

.This is my personal diary which contains resources, which I know, learned or people have told me to experiment with. I started writing this diary in Mar' 2019. This diary is related to my work/learning in data science, AI, ML, NLP, DL, GNN, GAN, Statistics, etc. A few links related to Software Development, People Management, Project Management are also kept in this diary. The resources here are python/R library link, blog articles, YouTube video links, article links, AI products links, architecture, images etc. The Table of Content and Content of this diary is not any specific order. Whatever I was/am getting I keep adding into this, mostly towards the end but not always.

You can connect with me on linkedin: <https://www.linkedin.com/in/harithapliyal/>

My blog: <https://dasarpai.netlify.app>, <https://dasarpai.com>

You can refer my github repo about AI resource: [DataScience-Github-Repo](#)

This is a 390 page personal learning diary. I didn't put any effort into making this document grammatically and spelling correct. So, there may be grammar and spelling errors.

Online Location of this Diary

#####

Date Updated: Thu, 15-Dec-22

Table of Contents

1. [ITU-AI-ML-in-5g-challenge / GNN Challenge](#)
2. [Graph Neural Networks - a perspective from the ground up](#)
3. [sebastianraschka DS courses](#)
4. [github: Pycaret](#)
5. [TF GNN](#)
6. [Manifold Learning](#)
7. [Summary of Data Science](#)
8. [Dr Repair - Graph-based, Self-Supervised Program Repair from Diagnostic Feedback](#)
9. [Interesting way to analyze a time series](#)
10. [Scrum vs Kanban](#)
11. [Leadership Qualities](#)
12. [Graph for Timeseries](#)
13. [Merlion](#)
14. [GNN based Movie Trailer Generator](#)
15. [Notation — ML Glossary documentation](#)
16. [How to Improve](#)
17. [Machine Learning & Data Science Competitions - ML Contests](#)
18. [The evolution of the Employee](#)
19. [Yann LeCun's Deep Learning Course at CDS – NYU Center for Data Science](#)
20. [Taxonomy of GNN Learning Methods](#)
21. [GNN for Novice](#)
22. [Some free alternatives software to MATLAB](#)
23. [500 Page Book on Quantum Computing](#)
24. [2022 Top 22 Tech Trends](#)
25. [Acing Data Science Interview Questions](#)
26. [Garbage Features](#)
27. [SQL Training](#)
28. [Process Mining](#)
29. [Best Graph Neural Network architectures: GCN, GAT, MPNN and more \ AI Summer](#)
30. [Free AI, DL, Maths, Python Resources MarkTechPost.com](#)
31. [Merlion Architecture](#)
32. [70 Free Online Courses for Data Science to Advance Your Skills in 2022](#)
33. [Machine Learning Simplified](#)
34. [The most innovative AI algorithms released in 2022 \(till date\)](#)

35. [Transformer Takeover in AI](#)
36. [AIOPS](#)
37. [DS Books](#)
38. [Techprof free books](#)
39. [Autoencoders-tutorial](#)
40. [GNN](#)
41. [github: 500-AI-Machine-learning-Deep-learning-Computer-vision](#)
42. [The NLP Index: 3,000+ code repos](#)
43. [github: Nyandwi/machine_learning_complete](#)
44. [Leadership Lessons from APJ Kalam](#)
45. [Github: Python framework to explore, label, monitor data nlp projects](#)
46. [Maths for Data Science by IITM](#)
47. [Datarobot](#)
48. [github: SeldonIO](#)
49. [Internshala](#)
50. [konfhub](#)
51. [Machine Learning Projects in Python](#)
52. [github: Merlion](#)
53. [github: aws-samples ml-inference-using aws lambda](#)
54. [Traditional ML vs Transfer Learning](#)
55. [How To Use The 4 Types Of Intelligence To Make Better Decisions](#)
56. [Microsoft SDE Sheet - Top 35 Most Frequently asked](#)
57. [Global AI Hub – Build your future](#)
58. [Keras community built 100 concise and clear code examples](#)
59. [Intro to machine learning compilers and optimizers](#)
60. [Approaches to process and classify images in a new way.](#)
61. [Inductive link prediction in knowledge graphs](#)
62. [TensorFlow tutorials](#)
63. [DevOps - Azure Board Training](#)
64. [Intelligent Graph](#)
65. [Great NLP, Text Mining Talks post](#)
66. [Google Cloud Tools](#)
67. [Spark NLP \ State of the Art Natural Language Processing \ John Snow Labs](#)
68. [Channels to Learn Python](#)
69. [What is Neural Network? How does it understand things?](#)
70. [World's Leading AI and Technology Publication](#)
71. [Skills, Roles and Responsibilities in Data Science](#)
72. [Complexity Explorer](#)
73. [Network Science by Albert-László Barabási](#)
74. [Graph Neural Networks: Models and Applications](#)
75. [NLP in Notebooks Competition](#)
76. [GNN TimeSeries Intelligence](#)
77. [Tuning Neural Networks Part I: normalize your data](#)
78. [Multi-paragraph segmentation](#)
79. [Machine Learning 2020 summary: 84 interesting papers/articles](#)
80. [GitHub Announces GitHub Open-Source Grants Recipients In India](#)
81. [Watch "Python Tutorial For Beginners In Hindi \(With Notes\) 🔥" on YouTube](#)
82. [Quantum Computing Course](#)
83. [Optimum: The Optimization Toolkit for Transformers at Scale](#)
84. [CPU, GPU, TPU, FPGA](#)
85. [Textless NLP: Generating expressive speech from raw audio](#)
86. [Textless NLP: Generating expressive speech from raw audio](#)
87. [How Data-Centric Platforms Solve the Biggest Challenges for MLOps - The Databricks Blog](#)
88. [Best Resources to Learn Natural Language Processing in 2021 - KDnuggets](#)
89. [Index of Best AI/Machine Learning Resources \ Hacker Noon](#)
90. [Qwiklabs - Hands-On Cloud Training](#)
91. [Introduction to Graph Neural Network \(GNN\) \ Analytics Steps](#)
92. [Watch "Auto Tensorflow - End to End AutoML solution - Exclusive Launch" on YouTube](#)
93. [Graph Database, GraphQL and Machine Learning for Carbohydrate-Active Enzymes](#)
94. [Pydantic experiment](#)

95. [Graph ML](#)
96. [Reinforcement Learning Lecture Series 2021 \ DeepMind](#)
97. [Advances in Cybernetics, Cognition, and Machine Learning for Communication ... - Google Books](#)
98. [Visualization — Effective Python for Data Scientists](#)
99. [The NLP Index](#)
100. [Github: Anomaly Detection Open source](#)
101. [Patsy: Build Features with Arbitrary Python Code - Data Science Simplified](#)
102. [Introducing OpenSearch \ AWS Open Source Blog](#)
103. [End-to-End Differentiable Molecular Mechanics Force Field Construction](#)
104. [Learning resources \ DeepMind](#)
105. [Understanding LSTM Networks -- colah's blog](#)
106. [Machine Learning Models as Micro Services in Docker](#)
107. [Unsupervised Learning Techniques](#)
108. [A 10 free 'Springer' Books on the Topics of AI, Ethics, Machine Learning, Robotics, Cybersecurity, Philosophy, Science & Innovation](#)
109. [Deep Learning on Graphs: Past, Present, And Future :](#)
110. [Watch "Ask the Expert: VS Code: Development in the Open" on YouTube](#)
111. [Analytics Tools](#)
112. [Online Data Science Course from Univ](#)
113. [ML Powered Apps vs Traditional Programming](#)
114. [Task:Introduction to Deep Learning](#)
115. [How Pokémon GO scales to millions of requests? \ Google Cloud Blog](#)
116. [A programmers guide to data mining](#)
117. [Pokeman go architecture](#)
118. [GNN summary paper](#)
119. [Regression Cheatsheet](#)
120. [Github: Data deduplication tool](#)
121. [Kappa for model evaluation](#)
122. [Data science resources](#)
123. [IIT Bombay Invites Applications from Students for Free Online Course on Quantum Mechanics](#)
124. [ViT](#)
125. [How Storytelling Affects the Brain](#)
126. [Navigating the Dozens of Different Strategy Options](#)
127. [Pytorch](#)
128. [GNN](#)
129. [AutoML](#)
130. [Deep Graph Neural Networks](#)
131. [Meta's Neural Prophet](#)
132. [Summary of HTTP Errors](#)
133. [mlcontents.com](#)
134. [What is R used for?](#)
135. [Intermittent Demand Forecasting with Deep Renewal Processes – Deep & Shallow](#)
136. [Github: GNN Challenge 2021](#)
137. [Holographic Principles](#)
138. [GNN Notation](#)
139. [AI Consulting: In-depth Guide with Top AI Consulting Firms](#)
140. [4 Keys to Effective—and Honest—Data Visualizations](#)
141. [SVM](#)
142. [ML For Managers](#)
143. [How to Talk to Your Data Scientist \ HuffPost Impact](#)
144. [Simplex Method with NumPy and Matrix Operations](#)
145. [Optimization problem](#)
146. [Difference between Soft Computing and Hard Computing](#)
147. [A Global Breakdown of Greenhouse Gas Emissions by Sector](#)
148. [Turing Bletchley: A Universal Image Language Representation model by Microsoft - Microsoft Research](#)
149. [Timeseries Library](#)
150. [10 Mysterious And Rare Artifacts You Didn't Know Before - TopTenScience](#)
151. [Categorical Encoding Cheat Sheet](#)

152. [Quantum Computing](#)
153. [NeurIPS 2021—12 papers You Shouldn't miss](#)
154. [Architecture of Modules in Merlion](#)
155. [poetry python dependency and package manager](#)
156. [The Technique of English Word Syllable Division in Speech ...](#)
157. [Tutorial on reading large datasets \ Kaggle](#)
158. [NVIDIA NLP in Production](#)
159. [Glassdoor Data Science Career Path](#)
160. [Google Using Deep Learning to Design Chips](#)
161. [MLContests.com](#)
162. [280+ Machine Learning Projects with Python](#)
163. [Machine learning engineer R&R](#)
164. [CNN and Transformer](#)
165. [AI, ML Tools](#)
166. [SBI Innovate for Bank 2022 Hackathon Technology Partner Microsoft \ TechGig](#)
167. [Intro to Intel's Distribution of OpenVino Toolkit](#)
168. [A Practical Strategy to Share the Code among Different Notebooks \ by Angelica Lo Duca \ Towards AI](#)
169. [Chart type selection](#)
170. [Monitoring NLP Pipelines](#)
171. [The Best NLP with Deep Learning Course is Free](#)
172. [Yellowbrick Python Library for ML](#)
173. [GoogleNews pip](#)
174. [Spring 2021 Schedule - Full Stack Deep Learning](#)
175. [Book Recommender System](#)
176. [Airflow](#)
177. [Mathematics of Big Data and Machine Learning](#)
178. [Data integrity issues](#)
179. [MLOPS with AWS 9 Weeks Course](#)
180. [Streamlit Tutorials Playlist](#)
181. [Online AI projects](#)
182. [3D Vision Summer School](#)
183. [40 Open-Source Audio Datasets for ML](#)
184. [Intro to Intel's OpenAPI](#)
185. [HackerNoon](#)
186. [ML Tools](#)
187. [Data Science Competitions](#)
188. [PyScript - JavaScript's sweet cousin.](#)
189. [GraphWorld: A Methodology For Analyzing The Performance Of GNN Architectures](#)
190. [github: connorferster](#)
191. [SQL Joins](#)
192. [Handling Missing Data](#)
193. [ML Pipeline](#)
194. [Generative Process - Overview of Neural Testbed](#)
195. [github: SeldoneO](#)
196. [Tracking Progress in Natural Language Processing \ NLP-progress](#)
197. [Neural Voice Camouflage](#)
198. [9 Free Cloud Storage](#)
199. [Generative Adversarial Networks \ Generative Models](#)
200. [Watch "Markov Chains Clearly Explained! Part - 1" on YouTube](#)
201. [Statistical tests](#)
202. [Watch "How to write a report using LATEX" on YouTube](#)
203. [AIM Survey - State of AI in Indian Enterprises](#)
204. [Data-Centric AI Competition Submission Guide](#)
205. [Understanding the role of individual units in a deep neural network \ PNAS](#)
206. [TPOT](#)
207. [Alexnet Architecture \ Introduction to Architecture of Alexnet](#)
208. [Watch "Introduction to AWS Services" on YouTube](#)
209. [Watch "Microservices Interview questions \ Interview Preparation" on YouTube](#)
210. [Watch "12 Factors App \ MicroServices Architecture \ Cloud Native Best Practices" on YouTube](#)

211. [Watch "Microservices Design Patterns \ Microservices Architecture Patterns \ Edureka" on YouTube](#)
212. [Watch "Microservices vs API \ Differences Between Microservice and API \ Edureka" on YouTube](#)
213. [Watch "Build a custom ML model with Vertex AI" on YouTube](#)
214. [\[Infographics\] Data Science Skills Roadmap: Skills and Certifications to Have in 2021 \ SDS Club](#)
215. [Autoencoders in Python \ How to use Autoencoders in Python](#)
216. [How to Calculate Feature Importance With Python](#)
217. [MLOps - Machine Learning Operations](#)
218. [Ten Highest Paying Companies for Data Scientists in 2021](#)
219. [MLOps Course \ MLOps Online Training. India - 360DigiTMG](#)
220. [Continuous Integration and Deployment for Machine Learning](#)
221. [Why AI is a game-changer for renewable energy \ EY - US](#)
222. [Worldscholarship Forum](#)
223. [Bird Song Classification using Siamese Networks and Dilated Convolutions](#)
224. [Matplotlib vs. Plotly: Let's Decide Once and for All](#)
225. [Dispatch from Bangalore – TechCrunch](#)
226. [RoBERTa building for Italian lang](#)
227. [PyTorch 1.9 - Towards Distributed Training and Scientific Computing](#)
228. [Train New BERT Model on Any Language](#)
229. [How to Create Report-Ready Plots in Python](#)
230. [OpenAI Launches GitHub Copilot: AI Focused On Code Generation. Should We Be Worried Now?](#)
231. [What OpenAI and GitHub's 'AI pair programmer' means for the software industry](#)
232. [Importing/Reading Excel data into R using RStudio \(readxl\) \ R Tutorial 1.5b \ MarinStatsLectures](#)
233. [How I passed the AWS Solutions Architect Associate Exam in 1 month 2021](#)
234. [Hindawi Journals](#)
235. [Is it possible to use multithreading inside of Flask?](#)
236. [Introduction to Deep Learning](#)
237. [100+ Free Machine Learning Books](#)
238. [Free online book - Machine Learning from Scratch](#)
239. [Skill Basics](#)
240. [Operating System For Machine Learning & AI](#)
241. [From Pandas to PySpark with Koalas](#)
242. [Free Data Science. Machine Learning](#)
243. [NET Architecture Guides](#)
244. [Getting Started with GPT-3 in Power Platform](#)
245. [Netflix Recommendations](#)
246. [Techlearn](#)
247. [Data Ingestion with TensorFlow eXtended \(TFX\)](#)
248. [The ExampleGen TFX Pipeline Component](#)
249. [The School of AI](#)
250. [Github: GREMLIN: An Apache TinkerPop Tutorial](#)
251. [Quantum Machine Learning](#)
252. [Quantum Machine Learning MOOC](#)
253. [Quantum Mechanics and Quantum Computation](#)
254. [Quantum Computation](#)
255. [Qiskit](#)
256. [TensorFlow Quantum](#)
257. [CirqBasics.ipynb - Colaboratory](#)
258. [QpiAI - AI and Quantum Simplified](#)
259. [MadewithML](#)
260. [Kick Start - Google's Coding Competitions](#)
261. [Online Python Challenges - Python Principles](#)
262. [Gaining access to the best machine-learning methods - O'Reilly Radar](#)
263. [Data Science Blogathon - 11](#)
264. [The Machine & Deep Learning Compendium](#)
265. [Github: NLP Resources dlq4nlp](#)

266. [Machine Learning Life Cycle 2.0 - Mind Map](#)
267. [Handbook on Data Protection and Privacy](#)
268. [6 low-code internal tool builders in 2021](#)
269. [AI books](#)
270. [Resources to learn DS](#)
271. [Parallel Processing and Pandas](#)
272. [Julia vs. Python](#)
273. [Introduction to Computational Thinking](#)
274. [A fun way to start learning Julia.](#)
275. [Data Science Tools](#)
276. [5 Most Popular Datasets](#)
277. [Drug Discovery Using Artificial Intelligence](#)
278. [DLG4NLP tutorial](#)
279. [Deep Reinforcement Learning by Pieter Abbeel](#)
280. [6 Highest Paying Data Science Certifications You Should Know](#)
281. [The Great Hack \ Netflix Official Site](#)
282. [Probability for Data Scientist](#)
283. [Quantum Computing, Artificial Intelligence, & Machine Learning for Drug Discovery](#)
284. [Gaussian processes \(1/3\) - From scratch](#)
285. [GRAPH4NLP](#)
286. [Computer Vision Skills](#)
287. [The Data Science Interview Study Guide - KDnuggets](#)
288. [GM VAE for NLP](#)
289. [NPTEL Courses on Computer Science and Engineering](#)
290. [VAE](#)
291. [AWS vs. Azure vs. Google Cloud - Price Comparison](#)
292. [Data Analytics and Visualization Made Easy - Juice Analytics](#)
293. [JupyterLab vs DataSpell](#)
294. [Top 15 YouTube Channels For Deep Learning](#)
295. [GitHub : TheAlgorithms/Python](#)
296. [Github: Label Studio Spacy](#)
297. [Free Data Science Courses - 11-Months Free IBM Certification Courses](#)
298. [Data Science Links](#)
299. [Free AI Resources - MarkTechPost](#)
300. [General and Scalable Parallelization for Neural Networks](#)
301. [Why Machine Learning Is Changing The Advertising Industry](#)
302. [Marketers: Here's Your Statistical Models Cheat Sheet](#)
303. [The Current Applications Of Artificial Intelligence In Mobile Advertising](#)
304. [Statistical Inference](#)
305. [CNN + Jraph in Colab](#)
306. [DAIR.ai YouTube Courses](#)
307. [Graph Neural Networks in Colab](#)
308. [Github: Must Practice this GNN notebook](#)
309. [Interview-question-data-science](#)
310. [How to Become Better Listener](#)
311. [Imbalance Training Data](#)
312. [Introduction to Probability for Data Science](#)
313. [Data retrieval with SQL, data manipulation MIT Software tool class](#)
314. [7 Ways how Data Scientists use Statistics](#)
315. [Google Data Science Interview Questions and Answers](#)
316. [Best Free Machine Learning Books](#)
317. [Machine Learning Algorithms Cheatsheet](#)
318. [Tools for Data sourcing, data wrangling, data application](#)
319. [Great Leaders Are Confident, Connected, Committed, and Courageous](#)
320. [25 Github Repositories Every Python Developer Should Know](#)
321. [11 Automatic Machine Learning Frameworks in 2022](#)
322. [Data Science Roles, Responsibilities, Tools](#)
323. [Debugging in Python](#)
324. [TensorFlow Introduces TensorFlow Graph Neural Networks \(TF-GNNs\)](#)
325. [India has wasted the potential of its large young population](#)

326. [The Power of Visualization in Data Science](#)
327. [Text Generation from Knowledge Graphs](#)
328. [Evidently AI - Open-Source Machine Learning Monitoring](#)
329. [ShaplyAI](#)
330. [BentoML](#)
331. [MLOps Toys \ A Curated List of Machine Learning Projects](#)
332. [Github: AI Fairness 360](#)
333. [Analytics for LinkedIn](#)
334. [Data Visualization Tools](#)
335. [Graph ML 2022](#)
336. [Graph Neural Networks through the lens of Differential Geometry and Algebraic Topology](#)
337. [Data universe](#)
338. [15 Graphs You Need to See to Understand AI in 2021](#)
339. [2021's Top Stories About AI - IEEE Spectrum](#)
340. [The Pattern of Success in the Digital Leadership World](#)
341. [Yann LeCun's Deep Learning Course at CDS](#)
342. [The best graph tool](#)
343. [Github: 365 days computer vision](#)
344. [Pytorch is a Deep Learning framework](#)
345. [Rising Against the Diminisher](#)
346. [Category - Programming Books](#)
347. [Designing Event-Driven Systems](#)
348. [Github: Seldon Core: Blazing Fast, Industry-Ready ML](#)
349. [Databricks](#)
350. [Deepfake using conv vision transformer](#)
351. [Detection of Fake Reviews on Online Review Platforms using Deep Learning Architectures](#)
352. [GNN Intro](#)
353. [50 Best Data Science Project Ideas -2022](#)
354. [60 Best Free Online Courses for Machine Learning & AI in 2022](#)
355. [Data Lifecycle](#)
356. [Github: HiPlot - High dimensional Interactive Plotting](#)
357. [6.10. Visualization — Effective Python for Data Scientists](#)
358. [Data Science Practices & Tools](#)
359. [Reproducibility in ML](#)
360. [Week 4 - Reproducibility in Machine Learning: From Theory to Practice - Koustuv Sinha](#)
361. [Designing, Visualizing, and Understanding Deep Neural Networks](#)
362. [YOLO v4 explained in full detail](#)
363. [Phases OF Dev-Ops](#)
364. [ML TOOLS & TECHNOLOGIES](#)
365. [PyCaret with FastAPI](#)
366. [18 Data Science Podcasts](#)
367. [Effective Testing for Machine Learning \(Part I\)](#)
368. [FREE DS MASTER CLASS SERIES \(PYTHON\)](#)
369. [Simple Explanation of LSTM \ Deep Learning Tutorial 36 \(Tensorflow, Keras & Python\)](#)
370. [Covid19 forecasting](#)
371. [COVID-19: Face Mask Detection using TensorFlow and OpenCV](#)
372. [14 Python Lib for Cyber Security](#)
373. [Roles in Data Science Project](#)
374. [Computer vision datasets: VisualData Discovery](#)
375. [DatAndroid Dataset](#)
376. [Pandas online learning](#)
377. [MongoDB Fundamentals Course](#)
378. [GroundedML — ICLR 2022](#)
379. [PyTorch vs. TensorFlow: Which Framework Is Best for Your Deep Learning Project?](#)
380. [MLOps \ Bring DevOps To Data Science With MLOps](#)
381. [Data science project life cycle](#)
382. [Devops life cycle](#)
383. [Continuous integration](#)
384. [3 ways to use data, analytics, and machine learning in test automation](#)
385. [Automated Functional Testing - Software Testing Tool](#)

386. [Using sub graph for more expressive GNN](#)
387. [HOW AI & MACHINE LEARNING IS CHANGING TEST AUTOMATION](#)
388. [NanoEdge AI Studio](#)
389. [How I Built an ML Algorithm to Improve Test Automation](#)
390. [AI and ML in Testing: X Tips To Make Test Automation Effective](#)
391. [KerGNNs: Interpretable Graph Neural Networks with Graph Kernels](#)
392. [TOP 10 GitHub Repositories for Data Science](#)
393. [Real-time machine learning: challenges and solutions](#)
394. [How to Read ML Papers Easily](#)
395. [Rulebased chatbot in Python](#)
396. [Forecasting: Principles and Practice](#)
397. [Chapter 10 Forecasting hierarchical or grouped time series](#)
398. [Will we see GPT-3 moment for computer vision?](#)
399. [Tiny ML, Quantum ML, Auto ML, MLOps, FUSLL Stack DL](#)
400. [List of Unicorns Startups in India \ Top Unicorns in India](#)
401. [SQL vs NoSQL Tools for Cloud](#)
402. [Free AI Introductory Course For All](#)
403. [Stanford CS224U Natural Language Understanding Spring 2021](#)
404. [Grouped multivariate and functional time series forecasting:](#)
405. [Cloud security assessment checklist](#)
406. [Python Cheat Sheet](#)
407. [Neural Network Types](#)
408. [Data Science Interview Questions](#)
409. [Ace the Data Science Interview book](#)
410. [Cracking-the-data-science-interview](#)
411. [Relationship Extraction for Knowledge Graph Creation From Biomedical Literature](#)
412. [DS Community](#)
413. [Metaverse – A New Era Emerging](#)
414. [Tensorflow data validation](#)
415. [Page rank Algo](#)
416. [Github: Auto Tensorflow](#)
417. [IIT Madras Offers Free Online 12- Week Certificate Course on AI](#)
418. [Top 10 Applications of Machine Learning in Cybersecurity](#)
419. [How to handle Emoji '😊' & Emoticon ':-\)' in text preprocessing?](#)
420. [Text Detection and Extraction using OpenCV and OCR](#)
421. [Devanagari Handwritten Character Dataset Data Set](#)
422. [Hindi-OCR](#)
423. [Stack exchange all rooms](#)
424. [How to improve Hindi text extraction?](#)
425. [What is Categorical Data \ Categorical Data Encoding Methods](#)
426. [DART](#)
427. [TimeSeries Forecasting models](#)
428. [Statistical Distances](#)
429. [AI Usecases](#)
430. [Architecture-to-operationalize-ML](#)
431. [neptune.ai](#)
432. [DeepMind shares a list of free AI & ML resources](#)
433. [AI course](#)
434. [TOC of AI course](#)
435. [Data Science Conceptual Map](#)
436. [Layers in NLP](#)
437. [Regression Analysis Estimation and Error Function](#)
438. [NLI for DB](#)
439. [Transformer models - Hugging Face Course](#)
440. [Learning From Data - Online Course \(MOOC\)](#)
441. [Find Leadership Courses](#)
442. [Wrapper to load dataset from remote PyTorch dataset](#)
443. [Advanced Certification in Data Science and AI by IIT Madras](#)
444. [How well do explanation methods for machine-learning models work?](#)
445. [Confusion Matrix for 2+ Classes](#)

446. [ML Algo Cheat Sheet](#)
447. [Machine Learning Algorithms Free course 8hours](#)
448. [AI Technologies in Construction](#)
449. [Github: Data drift what next](#)
450. [Cross Validation - Hold Out, K-Fold, LOOCV](#)
451. [Data Science Project Lifecycle](#)
452. [Data Science Tools, Algorithms, Pipeline](#)
453. [Graph neural networks \(GNNs\)](#)
454. [GNN](#)
455. [Clustering](#)
456. [How to master Streamlit for data science](#)
457. [\[ML News\] ConvNeXt: Convolutions return](#)
458. [Nature of Research and Questions](#)
459. [Tips for Sailing Data Science job](#)
460. [7 Interactive Bioinformatics Plots made in Python and R](#)
461. [Installing Python Packages from a Jupyter Notebook](#)
462. [Neural Networks on Graph](#)
463. [Live code Visualization notebook](#)
464. [What Is a Bitcoin ATM?](#)
465. [A New Reinforcement Learning Based Method - Dead-end-Discovery](#)
466. [Next Level of Data Visualization in Python](#)
467. [7 Plotly Graphs in 3D: Stocks, Cats, and Lakes](#)
468. [Github: Interactive graph in Python](#)
469. [Modern Deep Learning Techniques Applied to Natural Language Processing by Authors](#)
470. [NLP with Deep Learning Video playlist Stanford Online](#)
471. [Michigan University Fall 2020 Schedule](#)
472. [Deep Learning for Video, Master in Computer Vision Barcelona 2019](#)
473. [Computer Vision course at CTU in Prague](#)
474. [Github: DS Cheat sheets](#)
475. [ML Cheatsheet](#)
476. [IIT Khadakpur Sanskrit NLP](#)
477. [Making Sanskrit Accessible through AI-based Text Processing](#)
478. [Fundamentals of NLP research in Sanskrit](#)
479. [Is Sanskrit the most suitable language for natural language processing?](#)
480. [Sanskrit github](#)
481. [10 Data Analysis Methods](#)
482. [AI NLP Timeline and the Transformer Family](#)
483. [Introduction by Example - pytorch_geometric 2.0.4](#)
484. [Python Notebook to Webapp](#)
485. [18 Feb Trending Github](#)
486. [Real time machine-learning challenges and solutions](#)
487. [How Machine Learning Can Help Test Automation](#)
488. [Test Automation in the World of AI & ML](#)
489. [Maths for neuroscience](#)
490. [HyperSense AI Studio \ No Code Platform \ AutoML](#)
491. [Top 20 Websites for Data Science and ML](#)
492. [Websites for competitive programming](#)
493. [Raspberry pi](#)
494. [Library for Auto labelling](#)
495. [Github: Using AWS Lamda and EFS](#)
496. [Resources and tools](#)
497. [Paying the hidden technical debt](#)
498. [Hidden Technical Debt](#)
499. [Technical Debt in Machine Learning](#)
500. [Hidden technical debt in machine learning systems](#)
501. [Best Docket Cheatsheets](#)
502. [How to Scale AI in Your Organization?](#)
503. [CVPR 2022 paper Lifelong Graph Paper](#)
504. [DataOps Tools](#)
505. [NLP Repo](#)

506. [100+ data science Cheatsheet](#)
507. [Ignition enabled fast prototyping of GNN](#)
508. [Deep Learning with PyTorch. Full course](#)
509. [List of Language Models](#)
510. [AI Books Cover Pages](#)
511. [Solution approach for ai problems](#)
512. [Algorithms and their applications](#)
513. [What is Outlier \ PyOD For Outlier Detection in Python](#)
514. [Free ML tools from Microsoft](#)
515. [How to Easily Automate Emails with Python](#)
516. [Publicly Real-World Datasets To Evaluate Stream Learning Algorithms](#)
517. [Companies hiring data scientist](#)
518. [Google drive DS material](#)
519. [Microsoft Is Going Global With Speech Enablement](#)
520. [A topic-centric list of HQ open datasets](#)
521. [Top Explainable AI \(XAI\) Python Frameworks in 2022](#)
522. [MLOPs with AWS Cloud](#)
523. [The world of machine learning algorithms](#)
524. [Think Big Data](#)
525. [9 Reasons - Why Artificial Intelligence is So Essential Right Now?](#)
526. [Data Science Lifecycle](#)
527. [ML Deployment Tools](#)
528. [Machine Learning](#)
529. [New Generation Technologies](#)
530. [10 free websites To learn Web3 and Blockchain development](#)
531. [Blockchain Technologies](#)
532. [Web3 Stack](#)
533. [Machine Learning with Signal Processing Techniques](#)
534. [Nasa's SpaceML Tool](#)
535. [SpaceML Taps Satellite Images to Help Model Wildfire Risks](#)
536. [The Best Open Source Chatbot Platforms in 2022](#)
537. [13 Best AI Chatbot Development Framework & Platforms](#)
538. [10 Best Chatbot Development Frameworks to Build Powerful Bots](#)
539. [Learning Representations of Geographic Locations From Unlabeled GPS Trajectories](#)
540. [Deep Learning Streaming Platform](#)
541. [Fake data Library](#)
542. [Top Explainable AI Frameworks](#)
543. [The Top 5 AI/Machine Learning blogs](#)
544. [Industry application of NLP](#)
545. [Airflow vs Luigi vs Argo vs Kubeflow vs MLFlow](#)
546. [ML Platforms: Dataiku vs. Alteryx vs. Sagemaker vs. Datarobot](#)
547. [Machine Learning Applied to Bigdata](#)
548. [Top Responsible AI \(Artificial Intelligence\) Tools in 2022](#)
549. [Machine Learning Zoomcamp](#)
550. [90+ Data Science Projects You Can Try with Python](#)
551. [How to implement data-centric AI in NLP](#)
552. [Issue #190 - THE ML ENGINEER](#)
553. [Machine Learning in Alteryx with PyCaret](#)
554. [Machine Learning in Tableau with PyCaret](#)
555. [Machine Learning for Cyber Security](#)
556. [Google AI Introduces LocoProp](#)
557. [Distributed Training in TensorFlow with AI Platform & Docker](#)
558. [Serve hundreds to thousands of ML models : Architectures from industry](#)
559. [An Exhaustive Read & Watch List for AI](#)
560. [ML Concepts](#)
561. [Reco: Introduction to Probability for Data Science](#)
562. [Supervised Clustering: How to Use SHAP Values for Better Cluster Analysis](#)
563. [Github: 800 free #ComputerScience classes](#)
564. [Mathematics of Big Data and Machine Learning @MIT](#)
565. [Linear Algebra @MIT](#)

566. [CS109A Data Science @Harvard](#)
567. [NLP with Deep Learning by Stanford](#)
568. [Google Vizier: A Service for Black-Box Optimization](#)
569. [Tinker With a Neural Network in Browser](#)
570. [Hand Gesture Recognition - Pretrained](#)
571. [Blockchain Tech](#)
572. [Distributed GNN training](#)
573. [Serverless computing for GNN](#)
574. [Graph Analysis to Fight Fraud](#)
575. [Transfer Learning Colab Note](#)
576. [Introduction to Vertex AI \ Google Cloud](#)
577. [Continuous Integration for Machine Learning: Testing ML Models](#)
578. [Computer Vision Nanodegree](#)
579. [Sign up: Kaggle x Scale AI offer](#)
580. [Seminal Work in Data Science](#)
581. [Metaverse beyond the hype](#)
582. [Vertex AI: Qwik Start \ Google Cloud Skills Boost](#)
583. [How Azure Machine Learning works \(v2\) - Azure Machine Learning \ Microsoft Docs](#)
584. [Github probml/pml2-book](#)
585. [Probabilistic ML](#)
586. [Systematic Evaluation of Causal Discovery in Visual Model Based Reinforcement Learning](#)
587. [Hypothesis Testing](#)
588. [Architecture](#)
589. [The Basics - Holochain Docs](#)
590. [Understanding AlphaZero Neural Network's SuperHuman Chess Ability](#)
591. [Flink Vs. Spark: Difference Between Flink and Spark](#)
592. [Design Patterns](#)
593. [Web2 vs web3 platforms](#)
594. [100+ gpt3 projects](#)
595. [Python Design Patterns](#)
596. [Tutorial of fundamental remote sensing and GIS methodologies using open source software in python](#)
597. [Mining Trends in Data Science Blog Headlines](#)
598. [Best softwares for #latex like data visualisation programs](#)
599. [Knowledge Graph-based Perspective on Named Entity Disambiguation](#)
600. [Google Cloud Anthos Series - 1: Introduction to Anthos](#)
601. [Securing apps for Googlers using Anthos Service Mesh](#)
602. [Anthos supports NVIDIA GPUs](#)
603. [Animated Chart Presentation in Jupyter Notebook.](#)
604. [Word Error Rate in STT](#)
605. [Quality Engineering Microservices](#)
606. [Automate Machine Learning using Databricks AutoML](#)
607. [AI for Beginners](#)
608. [Bigdata architecture](#)
609. [Language-Vision Model To Leverage Text Medical Reports For Improved Segmentation](#)
610. [Information Theory & Bayesian ML](#)
611. [A generic, simple and fast implementation of Deepmind's AlphaZero algorithm.](#)
612. [Chart Suggestions Guide \ Tableau Public](#)
613. [Continuous Deployment of ML Models to the Edge](#)
614. [100+ data structure and algorithms](#)
615. [Microsoft Research - General-Purpose Multimodal Foundation Model BEiT-3](#)
616. [BEiT-3 Performance on Deep Learning Tasks](#)
617. [Data Analytics vs Data Science](#)
618. [Machine Learning Systems versus Machine Learning Models](#)
619. [Machine Learning Systems vs Machine Learning Models](#)
620. [Top 8 YouTube Channels to learn Web3](#)
621. [Learning timeseries can be tough.](#)
622. [Pen and Paper Exercises in Machine Learning](#)
623. [12 Best Online Courses for Machine Learning with Python-Bestseller 2022](#)
624. [Time Space Complexity in Machine Learning Algo](#)

625. [Activation Functions](#)
626. [Ace Your Data Science Interview in 6 Steps](#)
627. [Stanford University Free Online Courses 2022](#)
628. [Timeseries in Pycaret](#)
629. [Checklist - training deep neural networks](#)
630. [5 Checks during Training of ML Models](#)
631. [CS 229 - Supervised Learning Cheatsheet](#)
632. [Expert Data Analyst links](#)
633. [Learn Go](#)
634. [The AWS AI/ML Stack](#)
635. [Challau Metaverse](#)
636. [Data Quality Attributes](#)
637. [Which Visualization](#)
638. [How to Become a Data Analyst Intern - Infographic](#)
639. [Introduction to Causal Inference](#)
640. [AI Bank of Future](#)
641. [Atomic Habits](#)
642. [The Role of an AI Architect](#)
643. ["An Analysis of Deep Learning Neural Networks"](#)
644. [Comparing Model Evaluation Techniques](#)
645. [5 open-source frameworks for implementing GANs](#)
646. [PregEx : Write Human-Readable Regular Expressions in Python](#)
647. [FREE Node.js course](#)
648. [Seeing Theory : Statistics more accessible through interactive visualizations](#)
649. [Use MLNET to build custom machine learning solutions in dotnet](#)
650. [Conversational AI and Chatbot](#)
651. [Different Charts](#)
652. [10 Machine Learning Algorithms In Python \[A Beginners Guide\]](#)
653. [Gephi : Visualization and Exploration](#)
654. [Transformers are Graph Neural Networks](#)
655. [goodrahstar/draw-neural-network:](#)
656. [Quick, Draw!](#)
657. [Draw Neural Networks in Latex](#)
658. [Statistical Tests](#)
659. [Graph ordering attention networks](#)
660. [Bitcoin Prediction - FB-Prophet outperforms ARIMAX, XGBOOST & LSTM](#)
661. [introduction chapter on Geometric Deep Learning](#)
662. [Graph Transformer](#)
663. [Overview of GTransformer](#)
664. [Roadmap to Web3](#)
665. [Ace The SQL Interview](#)
666. [Summer school on Statistical Physics & Machine learning](#)
667. [Reproducible Deep Learning](#)
668. [MachineLearning & DeepLearning Concepts Playlist](#)
669. [Docker Integration with Other Softwares](#)
670. [Deck of slides on GNNs: foundations, challenges, and explainability](#)
671. [Structure-Aware Transformer for Graph Representation Learning](#)
672. [Machine Learning and Deep Learning frameworks and libraries for large-scale data mining](#)
673. [How Graph Neural Networks \(GNN\) work](#)
674. [What's New in v0.6.2: Continuous Deployment and a fresh CLI](#)
675. [Primers on training on neural network](#)
676. [8 Tips of StoryTelling](#)
677. [Upgini : Free automated data enrichment library for machine learning](#)
678. [Best Practices for Deploying Language Models](#)
679. [Distributed System Fundamentals](#)
680. [Learn web3 development by Building Projects](#)
681. [Adapts The Pretrained Language Image Models To Video Recognition](#)
682. [Data Analysis Video Playlists](#)
683. [Free Backend Development & Frontend Development Hosting Platforms](#)
684. [6 ways to use Jupyter notebook](#)

685. [21 Cyber Security Search Engines](#)
686. [What is Data Governance](#)
687. [Skills & Knowledge of Data Management](#)
688. [Machine learning systems design](#)
689. [Introduction to Machine Learning Interviews Book · MLIB](#)
690. [What is DALL-E, DALL-E2?](#)
691. [Four major smoothing techniques](#)
692. [b7leung/MLE-Flashcards](#)
693. [Starting Point of Data science for various people](#)
694. [Top Programming Languages of 2022](#)
695. [10 Programming Languages and their efficiency](#)
696. [maikherbig/AIDeveloper](#)
697. [Los Alamos Claims Quantum Machine Learning Breakthrough](#)
698. [How Google uses NLP to better understand search queries, content](#)
699. [Essential Charts of Data Analysis](#)
700. [CVPR 2022 notes with focus on Medical Imaging](#)
701. [Facebook AI Researchers Open-Source 'LLM.int8\(\)'](#)
702. [Data Structure - Cheat Sheet](#)
703. [Helpful AI tools for content writing and how to use them](#)
704. [PaLM: Scaling Language Modeling with Pathways](#)
705. [NLP - Explaining Joke](#)
706. [CS388 NLP Course](#)
707. [machine learning algorithms, data science infographics](#)
708. [Data Science Interview Questions and Answers Blogs - 360DigiTMG](#)
709. [Artificial Neural Network Questions to Test Your Skills](#)
710. [20 Questions to practice for machine learning interviews.](#)
711. [GAN from GNN To Correctly Predict the Coherent Orientations of Galaxies](#)
712. [Build Machine Learning Pipeline in Python and Deploy on Cloud easily](#)
713. [The Rise of Data Contracts](#)
714. [Workshop on Autonomous Driving - CVPR 2022](#)
715. [Google AI Introduces OptFormer](#)
716. [The ML Reproducibility Challenge is back!](#)
717. [Textualize/rich: Rich is a Python library for rich text and beautiful formatting in the terminal.](#)
718. [30 Projects Ideas You Can Build to Become a Blockchain & Web3 Master](#)
719. [Masked Language Model](#)
720. [Which Graph](#)
721. [40 Best Resources to Learn Natural Language Processing \[FREE\]- 2022](#)
722. [ConvNetJS MNIST demo](#)
723. [Google video QA](#)
724. [PyTorch Data Parallel Best Practices on Google Cloud](#)
725. [Causality for Machine Learning](#)
726. [Foundations of Causal Inference and its impact on Machine Learning](#)
727. [How Artificial Intelligence Is Changing The Law Industry for The Better • LegalScoops](#)
728. [Everything you've ever wanted to know about machine learning](#)
729. [Machine learning for sensors](#)
730. [Recent machine learning research tutorial focuses on Learning Theory](#)
731. [Top And Easy to use Open-Source Image Labelling Tools for Machine Learning Projects](#)
732. [An organized collection of 94 articles with NLP tips](#)
733. [Understanding AlphaZero Neural Network's SuperHuman Chess Ability - MarkTechPost](#)
734. [Want to learn more about Meta Learning?](#)
735. [Video lectures, UC Irvine CS 275](#)
736. [Towards Responsible Data Practices for Machine Learning in India: Health & Agriculture](#)
737. [Scalable Machine Learning System for Pre-Season Agriculture Yield Forecast](#)
738. [Model Evaluation Metrics in Machine Learning - Nucleusbox](#)
739. [190 Python Projects with Source Code \ by Aman Kharwal \ Medium](#)
740. [Dr. Rina Dechter @ UCI](#)
741. [TorchStudio, Deep Learning Toolkit](#)
742. [Free ML Engineering course](#)
743. [First Principles of Computer Vision, Columbia University](#)
744. [Are self-driving cars the future?](#)

745. [Guide to become Data Scientist](#)
746. [7 Advantage of social media marketing](#)
747. [Day 1 of 60 Days of Deep Learning with Projects Series](#)
748. [We Automated Attribute Tagging Using Deep Learning Models.\(1\)](#)
749. [9 Distance Measures in Data Science](#)
750. [Comparison of the Text Distance Metrics - DataScienceCentral.com](#)
751. [Author coder and VAE](#)
752. [Data Strategist - DataScienceCentral.com](#)
753. [Machine Learning Perceptron Implementation](#)
754. [HiLab-git/SSL4MIS: Semi Supervised Learning for Medical Image Segmentation](#)
755. [How do video live streamings work](#)
756. [Complete Guide to Data Visualization with Python](#)
757. [SDS 603: Geospatial Data and Unconventional Routes into Data Careers](#)
758. [7 tools for labeling of Audio, Image, Video and Textual Data](#)
759. [Free ML Engineering course starts on Monday!](#)
760. [12 Free video lectures on Deep Learning](#)
761. [What Are The 10 Best AI Consulting Firms](#)
762. [YOLOS](#)
763. [Top Parallel Processing Python Frameworks Data Scientists must know in 2022](#)
764. [Google AI Introduces a New Approach to Video-Text Learning Called Iterative Co-Tokenization](#)
765. [Meet Sipeed's TinyMaix](#)
766. [AI/CV/MLOps Tools](#)
767. [5 famous transformer architectures](#)
768. [Free courses to Learn Deep Learning](#)
769. [5 interesting #Python Projects with Source Code](#)
770. [UnMICST: Deep learning with real augmentation for robust segmentation](#)
771. [The First Workshop on Pre-training ICML22](#)
772. [Machine Learning Cheatsheets](#)
773. [Computational Linear Algebra](#)
774. [Guided-Generative Network](#)
775. [AlphaCode paper](#)
776. [How Do We Design and Develop Human-Centered AI?](#)
777. [Diplomacy: The AI Benchmark that Gets Us Closer to the Turing Test](#)
778. [ChatGPT vs LaMDA](#)
779. [Optimize Sales Funnel and Increase Revenue with Machine Learning](#)
780. [Create Observable and Reproducible Notebooks with Hex](#)
781. [Stanford Graph Learning Workshop 2022](#)
782. [Components of Research Paper](#)
783. [Software Design Patterns](#)
784. [Software Design Patterns2](#)
785. [How to Organize your Research Paper](#)
786. [Filter data using multiple columns - Microsoft Power BI Community](#)
787. [Edge#194: Masterful AI, the Training Platform for Automated Computer Vision](#)
788. [ML Algorithms and Training/Prediction/Space Complexity](#)
789. [12 Actions to improve Data Quality](#)
790. [Generative AI Application Landscape](#)
791. [MIT Deep Learning 6.S191](#)
792. [Welcome to Shapash's documentation ! — Shapash 2.2.0 documentation](#)
793. [Deep Learning Usecases](#)
794. [Research Paper Flow](#)
795. [Data Science Books](#)
796. [MLOps for production-grade ML](#)
797. [Westlake University Researchers Propose 'SimVP.'](#)
798. [Drexel University Introduces TorchFL:](#)
799. [Optimize Sales Funnel and Increase Revenue with Machine Learning in Python](#)
800. [Stanford Engineering Everywhere \ CS223A - Introduction to Robotics](#)
801. [Remote job in AI/ML](#)
802. [Algorithmic concepts \ Super Study Guide](#)
803. [Comprehensive Guide to Zero-Shot and K-Shot Learning](#)
804. [Learn Docker in 12 Minutes](#)

805. [The subversion process](#)
806. ['MEND' For Fast Model Editing At Scale](#)
807. [Data Structures and Algorithms](#)
808. [Digital Identify, Virtual World, Web3 Technologies](#)
809. [Lama Cleaner - Watermark Remover Open source](#)
810. [Dig Auto Profiler for Datasets](#)
811. [Learn from the best ML courses by Google](#)
812. [Basic Git Commands to get started - 14 days of Git](#)
813. [#Web3 and #Blockchain development](#)
814. [10 Top MLOps Books for Data Scientists](#)
815. [GAN Types](#)
816. [Machine learning resources YouTube](#)
817. [NLP resources for Indian languages](#)
818. [Feature Engineering with Neo4j and SageMaker](#)
819. [Books on Productivity Boost](#)
820. [7 FREE Web3 tutorials you need to know:](#)
821. [GitHub - dosisod/refurb: A tool for refurbishing and modernizing Python codebases](#)
822. [Researchers At IIIT Allahabad Propose T2CI GAN:](#)
823. [Deep Learning - UCB Berkeley 2021](#)
824. [Automated Machine Learning Hyperparameter Tuning in Python](#)
825. [Web3 Development and #Blockchain YouTube Courses](#)
826. [Top 10 Resources to get Free Datasets](#)
827. [Unsupervised anomaly detection using Anomalib library](#)
828. [A Framework for #web3 Infrastructure](#)
829. [Why building the metaverse is going to be a 'thrilling ride'](#)
830. [Awesome Diffusion Models](#)
831. [32 Important Datasets to Start Data Science](#)
832. [8 highly appreciated technical posts & videos](#)
833. [Salesforce Einstein, future of CRM](#)
834. [Machine Learning Models for Object Recognition in Underwater Video Images](#)
835. [Python for Bioinformatics - Drug Discovery Using Machine Learning and Data Analysis](#)
836. [An introduction to cheminformatics, data analysis and machine learning](#)
837. [AI to discover faster matrix multiplication](#)
838. [Free Certification Courses in Data Science using Python from IBM.](#)
839. [Aerial Vehicles detection Check](#)
840. [14 Data Science Skills](#)
841. [How to add a Subdomain in Github Pages- Tutorial 3](#)
842. [Another tool won't fix your MLOps problems \ by David Hershey \ Medium](#)
843. [MLOPS is not MLOPS Tools](#)
844. [Data science on Google cloud](#)
845. [Set of Techniques for Boosting Model Efficiencies](#)
846. [Researchers at Aveloop AI Introduce 'Deep Lake'](#)
847. [Big data pipeline on AWS Azure and GCP](#)
848. [DSC Webinar Series: How to Create Mathematical Optimization Models with Python](#)
849. [Model interpretability](#)
850. [Distributed Data Parallel in PyTorch - Video Tutorials](#)
851. [AI technologies for marketing](#)
852. [Why people give up?](#)
853. [Causal inference](#)
854. [Thesis Defence Questions](#)
855. [Katonic MLOps Platform \ Katonic](#)
856. [Neptune vs TensorBoard - neptune.ai](#)
857. [Synthetic Data - Machine Learning Techniques](#)
858. [Transformer Models](#)
859. [How to get better cell phone signal](#)
860. [What is GATO?](#)
861. [How do Google/Apple maps blur license plates](#)
862. [Sales funnel](#)
863. [How can banks become #AI/#Metaverse ready?](#)
864. [Fraud detection](#)

865. [Learn #Web3 for free](#)
866. [Digital detox](#)
867. [Into the metaverse: How conversational AI will build its experiential foundation](#)
868. [Application of iot](#)
869. [Hike cycle for artificial intelligence 2022](#)
870. [Technology Impact radar for 2022 - Garner](#)
871. [Free Websites to practice python](#)
872. [Understanding the Frisch-Waugh-Lovell Theorem](#)
873. [5 Data Science and Machine Learning competitions](#)
874. [The challenges of adversarial machine learning in constrained-feature applications](#)
875. [Hardware for Machine Learning spring 2021](#)
876. [FindIt: Generalized Object Localization with NaturalLanguage Queries](#)
877. [Serverless architecture](#)
878. [Metaverse trends by Gartner](#)
879. [Elements of metaverse](#)
880. [How I prepared for DeepMind and Google AI research](#)
881. [38 Free Courses on Coursera for Data Science You Must Know in 2022](#)
882. [Use machine learning to win the market.](#)
883. [Data Maturity Model](#)
884. [AR vs VR](#)
885. [Building-mlops-pipeline-problems.jpg](#)
886. [AI/ML Tools and Categories](#)
887. [free resources for learning Python and Data Science](#)
888. [Best System design resources](#)
889. [Building blocks of Data Science Project](#)
890. [Jaided AI: EasyOCR documentation](#)
891. [Developer-Y/cs-video-courses: List of Computer Science courses with video lectures.](#)
892. [Google AI Blog: Identifying Disfluencies in Natural Speech](#)
893. [Mastery in coding for data science is daily practice.](#)
894. [Research Paper Format](#)
895. [Computer Vision Research at Google and Boston University Proposes 'DreamBooth.'](#)
896. [3 Free Data science books for reference and learning.](#)
897. [Google sheet in Python Pandas](#)
898. [gsread library](#)
899. [Python vs Ruby](#)
900. [How to Read Research Abstract](#)
901. [Cost of AI Project and Estimating the Delivery Time - Addepto](#)
902. [Top 10 Open-Source Cybersecurity Project Ideas for Beginners](#)
903. [Top 10 Datasets For Cybersecurity Projects One Must Know](#)
904. [Top 7 cybersecurity projects for beginners - Great Learning](#)
905. [jivoi/awesome-ml-for-cybersecurity: Machine Learning for Cyber Security](#)
906. [Sipeed's TinyMaix Puts MNIST Digit Recognition on a Modest Microchip ATmega328 Microcontroller](#)
907. [Model Fitting and Line of Regression](#)
908. [The WeightWatcher tool for predicting the accuracy of Deep Neural Networks](#)
909. [Data Visualization Types: Everything a Marketer Needs to Know - TapClicks](#)
910. [Bitcoin Prediction - Using FBProphet](#)
911. [How to Generate Images from Text with Stable Diffusion Models](#)
912. [5 Exploding Areas Of The Metaverse That May Not Be On Your Radar Yet](#)
913. [Top free Books to Learn Data Science](#)
914. [Reading a Scholarly Article](#)
915. [Machine Learning Algorithms](#)
916. [Organizational Framework to Exploit the Data](#)
917. [AI artist who won competition says art world is 'in denial' about the tech](#)
918. [University of Washington MATH 340 Abstract Linear Algebra Summer 2020](#)
919. [Important links for writing research articles](#)
920. [Geometric Deep Learning - AMMI](#)
921. [Creating Mathematical Animations in Python using Manim -](#)
922. [Necessary libraries for Manim](#)
923. [manim library](#)

- 924. [Professional AI whisperers have launched a marketplace for DALL-E prompts](#)
- 925. [Dealing with Outliers Using Three Robust Linear Regression Models](#)
- 926. [solar Power and Surface Area Required](#)
- 927. [Cost of defect Leakage](#)
- 928. [5WH of Data Engineering](#)
- 929. [A Novel Approach to Creating Tailored Odors and Fragrances Using Machine Learning](#)
- 930. [Reservoir of data science and machine publications](#)
- 931. [How to Download, Install and Use Nvidia GPU For Tensorflow](#)
- 932. [Techniques for Training Large Neural Networks](#)
- 933. [microsoft/ML-For-Beginners - Iassic Machine Learning for all](#)
- 934. [SigmaQuan/Better-Python-59-Ways:](#)
- 935. [Top 20 Linked List Coding Problems](#)
- 936. [How Wireless Charger Works](#)
- 937. [Probability summary](#)
- 938. [Causal Inference and Statistical InferenceI](#)
- 939. [Why Zomato is goldmine](#)
- 940. [Quantum information science](#)
- 941. [AI Problems](#)
- 942. [AI governance framework](#)
- 943. [Agile Scrum Resources](#)
- 944. [Weather Data](#)
- 945. [How to built Positive Team Culture](#)
- 946. [Key Factors for Successful Goal Achievement](#)
- 947. [Application of Stats](#)
- 948. [Future of AI](#)
- 949. [what to do when I make a mistake](#)
- 950. [A new model that automatically generates movie trailers](#)
- 951. [Software Professions](#)
- 952. [Password Breaking Time](#)
- 953. [Adjacent Numbers in a Perfect Square 1-32](#)
- 954. [Evolution of Standard Oil](#)
- 955. [Watch Me Build an AI Startup](#)
- 956. [ML Frameworks/Tools](#)
- 957. [Data Science Project Stages](#)
- 958. [Machine Learning Resources](#)
- 959. [Understanding Skewed Distribution](#)
- 960. [Google Search Tech](#)
- 961. [Machine Learning Classical vs Quantum](#)
- 962. [New England Complex Systems Institute](#)
- 963. [Google Cloud Next](#)
- 964. [Why You Should Always Switch: The Monty Hall Problem](#)

~~~~~

### **ITU-AI-ML-in-5g-challenge / GNN Challenge**

1. <https://github.com/ITU-AI-ML-in-5G-Challenge/ITU-ML5G-PS-013-ATARI>
2. <https://bnn.upc.edu/challnge/gnnnet2021>

### **Graph Neural Networks - a perspective from the ground up**

<https://www.youtube.com/watch?v=GXhBEj1ZtE8>

### **sebastianraschka DS courses**

<https://sebastianraschka.com/blog/2021/dl-course.html>

### **github: Pycaret**

<https://github.com/pycaret/pycaret>

### **TF GNN**

Learn how to implement graph neural network models in TensorFlow with TF-GNN!

1. Read more → <https://blog.tensorflow.org/2021/11/introducing-tensorflow-gnn.html>

## Mainfold Learning

1. Manifold Learning: [http://perception.inrialpes.fr/people/Horaud/Courses/DAML\\_2011.html](http://perception.inrialpes.fr/people/Horaud/Courses/DAML_2011.html) most discussed on @kenreisman/machine-learning

#AI #MachineLearning #DataScience #GNN #ManifoldLearning

Data Analysis and Manifold Learning DAML

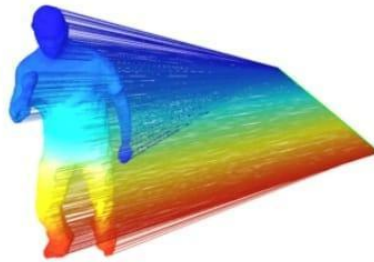
### Data Analysis and Manifold Learning (DAML)

Lecture series by [Radu HORAUD](#),

INRIA Grenoble Rhone-Alpes

Winter-Spring 2011

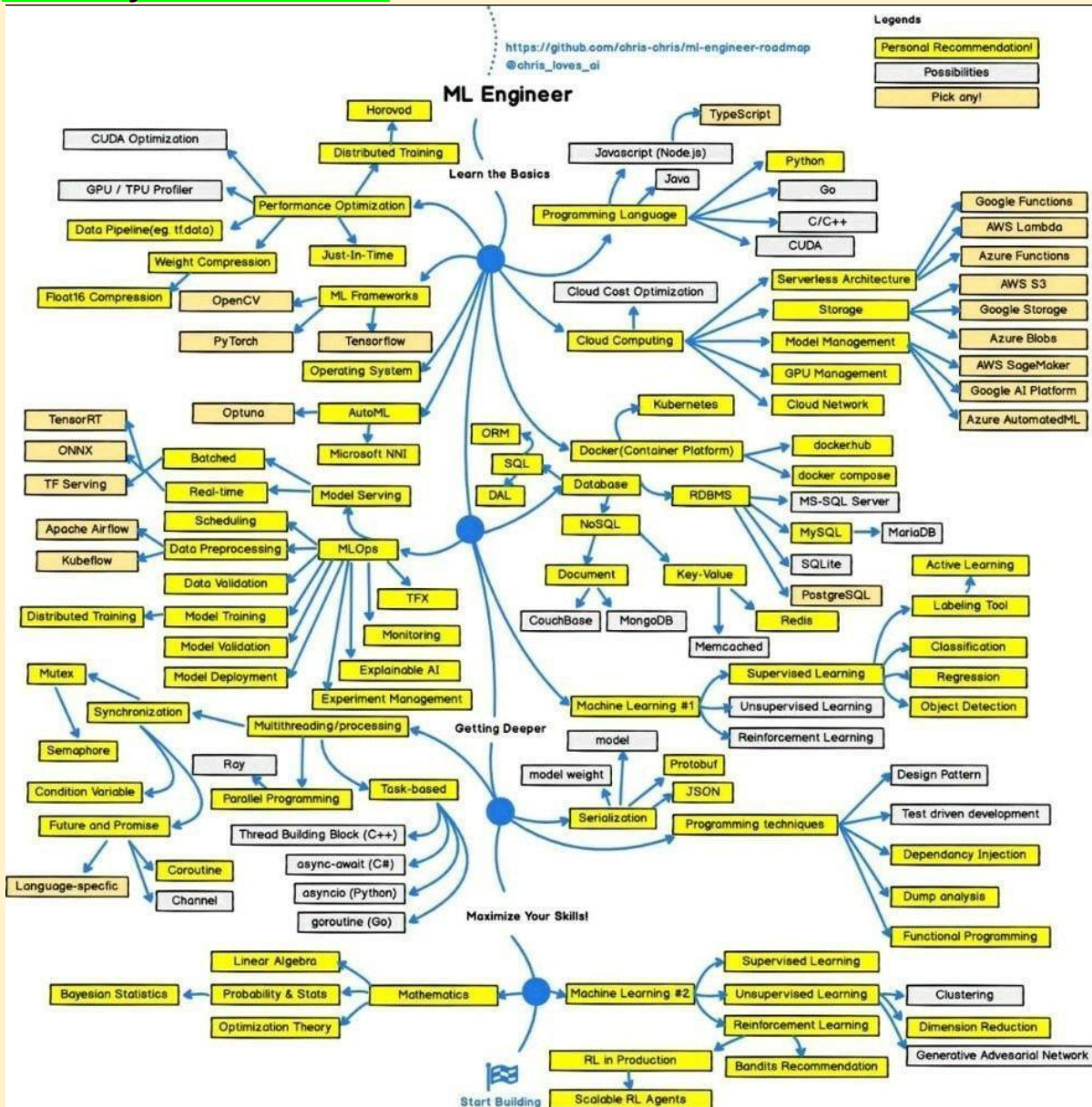
[Short course description \(pdf\)](#)



This is a 3D shape that is represented by an undirected weighted graph whose vertices correspond to 3D points and whose edges correspond to the local topology of the shape. Each graph vertex is projected onto the first non-null eigenvector of the graph's Laplacian matrix. This eigenvector may be viewed as the *intrinsic* principal direction of the shape, invariant to such shape deformations as articulated motion.

| Brief course description                                                               | Time and date              | Location                               | Material                          |
|----------------------------------------------------------------------------------------|----------------------------|----------------------------------------|-----------------------------------|
| <a href="#">Lecture #1</a> : Introduction to spectral and graph-based methods          | 14h-16h 7/1/2011 (Friday)  | UFR-IMA Grenoble: <a href="#">F113</a> | <a href="#">Horaud-DAML1.pdf</a>  |
| <a href="#">Lecture #2</a> : Symmetric matrices and their properties                   | 14h-16h 14/1/2011 (Friday) | UFR-IMA Grenoble: <a href="#">F113</a> | <a href="#">Horaud-DAML2.pdf</a>  |
| <a href="#">Lecture #3</a> : Graphs, graph matrices, and spectral embeddings of graphs | 14h-16h 21/1/2011 (Friday) | UFR-IMA Grenoble: <a href="#">F107</a> | <a href="#">Horaud-DAML3.pdf</a>  |
| <a href="#">Lecture #4</a> : Gaussian mixtures and the EM algorithm                    | 14h-16h 4/2/2011 (Friday)  | UFR-IMA Grenoble: <a href="#">F116</a> | <a href="#">Horaud-DAML4.pdf</a>  |
| <a href="#">Lecture #5</a> : Principal component analysis                              | 14h-16h 11/2/2011 (Friday) | UFR-IMA Grenoble: <a href="#">F114</a> | <a href="#">Horaud-DAML5.pdf</a>  |
| <a href="#">Lecture #6</a> : Bayesian PCA and factor analysis                          | 14h-16h 18/2/2011 (Friday) | UFR-IMA Grenoble: <a href="#">F114</a> | <a href="#">Horaud-DAML6.pdf</a>  |
| <a href="#">Lecture #7</a> : Laplacian embedding and spectral clustering               | 14h-16h 25/2/2011 (Friday) | UFR-IMA Grenoble: <a href="#">F114</a> | <a href="#">Horaud-DAML7.pdf</a>  |
| <a href="#">Lecture #8</a> : Introduction to kernel methods, kernel PCA                | 14h-16h 18/3/2011 (Friday) | UFR-IMA Grenoble: <a href="#">F114</a> | <a href="#">Horaud-DAML8.pdf</a>  |
| <a href="#">Lecture #9</a> : Diffusion kernels                                         | 14h-16h 25/3/2011 (Friday) | UFR-IMA Grenoble: <a href="#">F114</a> | <a href="#">Horaud-DAML9.pdf</a>  |
| <a href="#">Lecture #10</a> : Spectral matching                                        | 14h-16h 8/4/2011 (Friday)  | UFR-IMA Grenoble: <a href="#">F113</a> | <a href="#">Horaud-DAML10.pdf</a> |
| <a href="#">Lecture #11</a> : Other methods: LLE and LTSA                              | 14h-16h 22/4/2011 (Friday) | UFR-IMA Grenoble: <a href="#">F113</a> | <a href="#">Horaud-DAML11.pdf</a> |
| <a href="#">Lecture #12</a> : Manifold learning applications                           | 14h-16h 13/5/2011 (Friday) | UFR-IMA Grenoble: <a href="#">F114</a> | <a href="#">Horaud-DAML12.pdf</a> |

# Summary of Data Science



## Dr Repair - Graph-based, Self-Supervised Program Repair from Diagnostic Feedback

DrRepair: "Graph-based, Self-Supervised Program Repair from Diagnostic Feedback"!

1. <https://arxiv.org/abs/2005.10636> , #ICML2020

When writing code, we spend a lot of time debugging. Can we use machine learning to automatically repair programs from errors? [

We develop key insights into this problem:

- 1) the error message (feedback) is a crucial signal for learning repair
- 2) program-feedback graph: a joint graph of code & error message helps model the reasoning of repair (e.g. tracking the variable that caused the error).
- 3) self-supervised learning: freely-available, unlabeled programs (e.g. github, codeforces) can be turned into useful training examples of program repair

Final system achieves SOTA on DeepFix (program repair) and outperforms the prior best on SPoC (program synthesis)!

work also provides a general ML framework of "learning from feedback", which has various applications - editing documents based on comments, learning from users in interactive dialog, etc.

Paper: <https://t.co/jSZgTChKD9>

Code: <https://t.co/56Akn9Zqm7>

Slides: <https://t.co/15o6sFPZW8>

It's interesting...

#AI #MachineLearning #DataScience #DrRepair #NLP #SelfRepairML

### **Interesting way to analyze a time series**

An interesting way to analyze a time series is as a network (graph), not assuming that time order is synonymous with linearity.

Link to Research Article

<https://www.pnas.org/content/105/13/4972>

#AI #MachineLearning #TimeSeries #VisibilityGraphs #DataScience



Scrum vs Kanban

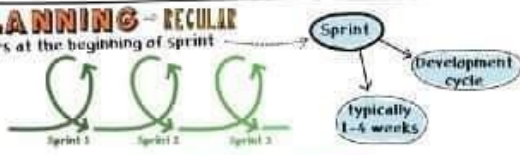
# SCRUM

VS

# KANBAN

## PLANNING - REGULAR

occurs at the beginning of sprint



## ESTIMATIONS OF TIME

BEFORE start of sprint



Items should be small to finish within sprint



## CHANGES TO WORK SCOPE

should wait for next sprint



## ROLES



## MEETINGS



## OWNERSHIP

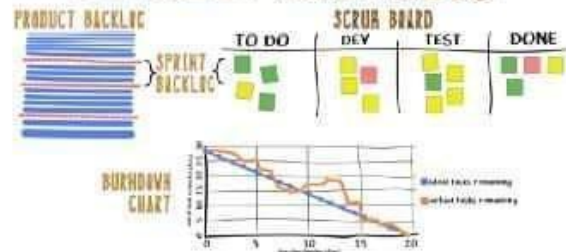
Product Owner

## WHEN TO USE

- Small items - small value
- adding increments is possible
- requirements in a good shape
- roadmap is clear
- more cross-dependent teams



## BOARDS / ARTIFACTS



## PLANNING

NOT PRECISE planning routine

PLAN WHEN they FINISH items

demand planning

CONTINUOUS FLOW



## ESTIMATIONS OF TIME

optional when items are completed



LIMIT

how many items can be in working columns at the same time

## CHANGES TO WORK SCOPE

added AS NEEDED



ROLES AS NEEDED



## MEETINGS

NONE REQUIRED



OWNERSHIP DEPENDS on defined roles and necessities

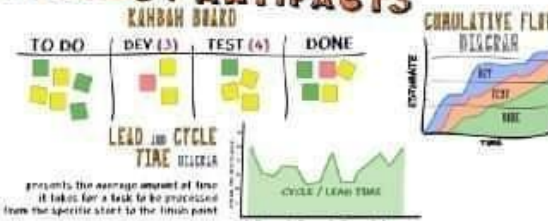


## WHEN TO USE

- Changes are too fast
- support/maintenance work (operational level)
- bugfix



## BOARDS / ARTIFACTS





# HEART-CENTERED LEADERSHIP QUALITIES

1. TRUTHFUL



2. TRUSTWORTHY



3. MODELS SELF-CARE



4. SERVANT LEADER



5. OPEN-MINDED



6. OWN CHARACTER



7. LIFELONG LEARNER



8. AVID READER



9. APPROACHABLE



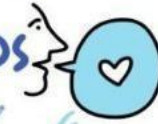
10. STRATEGIST



11. HELPS OTHERS RISE AND GIVE SUPPORT



12. MINDFUL OF WORDS



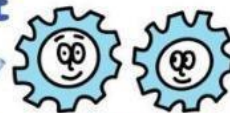
13. ATTENTIVE LISTENING



14. COMPASSIONATE FOR SELF AND OTHERS



15. RESPONSIBLE



16. COMMITTED



17. OFFER A SAFE, WELCOMING ENVIRONMENT



18. ABILITY TO FAIL FORWARD



19. EMPATHETIC

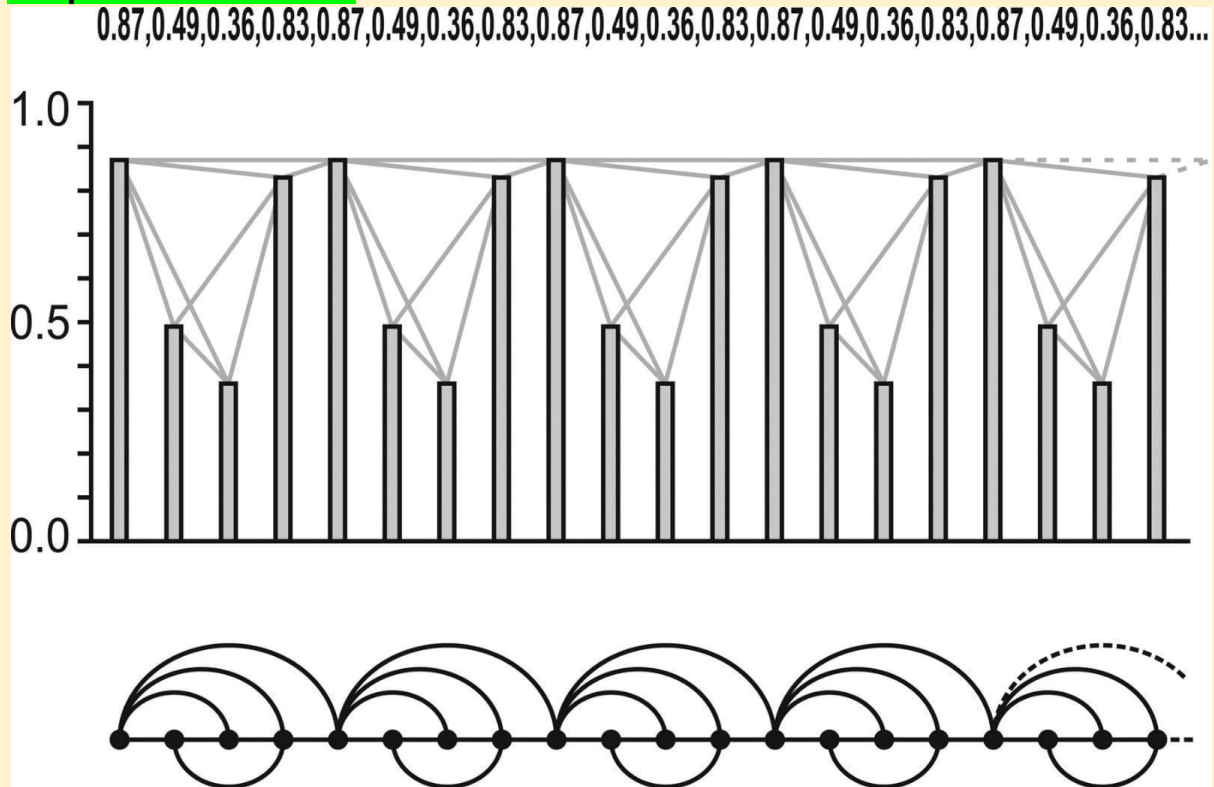


20. STRIVE TO MENTOR





## Graph for Timeseries



## Merlion

Salesforce Research Introduces 'Merlion', An Open-Source Machine Learning Library For Time Series Intelligence

Git: <https://lnkd.in/gvpEPQgy>

Paper: <https://lnkd.in/gayYxDds>

References:

<https://lnkd.in/dyeK3rnV>

<https://lnkd.in/gAW4PfuX>

<https://lnkd.in/gWsq8KpS>

#Merlion #Salesforce #TimeSeriesForecasting #AI #DataScience #AnamolyDetection #Forecasting #Prediction

**Salesforce** has opensourced **#MERLION** - a new Robust **#timeseries #forecasting** framework few weeks back.

**#Merlion** provides modules to preprocess univariate time-series data , Perform **#univariate** and **#multivariate** time series modeling using standard **#algorithms**

**#Merlion** also offers Anomaly detection algorithms , Ensembling features , **#automl #automactic #machinelearning** apis with various evaluation metrics.

It is the only **#TimeSeries** framework with multiple benchmark on **#popular #m4competetion #NREL** data and **#AUTOML** features

It has Flexible evaluation pipelines that simulate the live deployment & re-training of a model in production

It's user friendly code structures help in building Forecasting and Anomaly detection , **#visualizations** easier way.

## **GNN based Movie Trailer Generator**

1. <https://techxplore.com/news/2021-11-automatically-movie-trailers.html>



Ibrahim Sobh - PhD · 1st

Senior Expert of Artificial Intelligence, Valeo...

3h · 🌐

🔥 A new model that automatically generates movie trailers 🔥

💡 Movie trailers perform multiple functions: they introduce viewers to the story, convey the mood and artistic style of the film, and encourage audiences to see the movie.

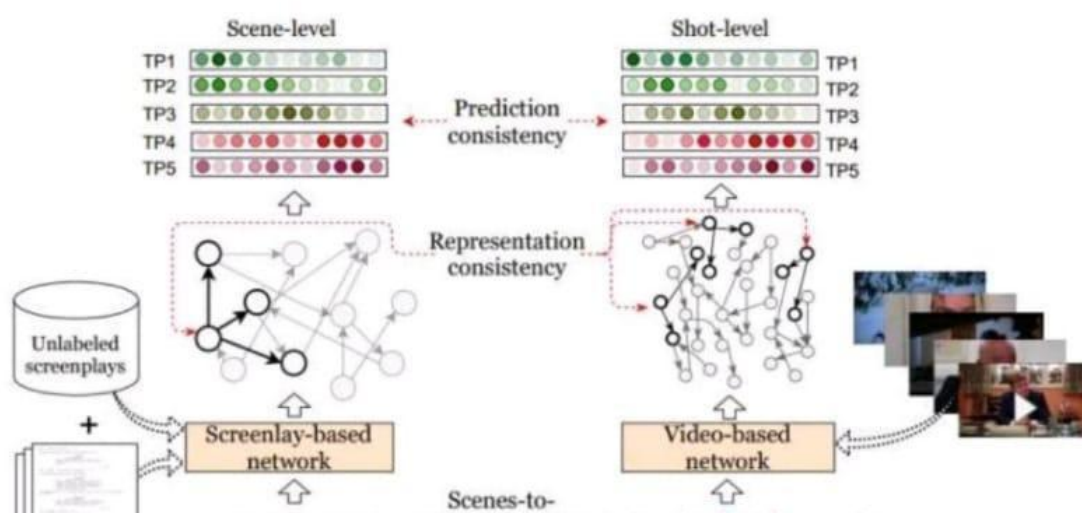
💡 These diverse functions make automatic trailer generation a challenging endeavor. We decompose it into two subtasks: narrative structure identification and sentiment prediction.

💡 We model movies as graphs, where nodes are shots and edges denote semantic relations between them. We learn these relations using joint contrastive training which leverages privileged textual information (e.g., characters, actions, situations) from screenplays.

💡 An unsupervised algorithm then traverses the graph and generates trailers that human judges prefer to ones generated by competitive supervised approaches.

👉 Paper <https://lnkd.in/dnkQ-egH>

👉 Article <https://lnkd.in/dznQmGhU>



## Notation — ML Glossary documentation

[https://ml-cheatsheet.readthedocs.io/en/latest/math\\_notation.html](https://ml-cheatsheet.readthedocs.io/en/latest/math_notation.html)

## How to Improve

# HOW TO IMPROVE

1. Meditate
2. Set goals
3. Read daily
4. Judge less
5. Eat healthy
6. Love yourself
7. Wake up early
8. Start a journal
9. Quit a bad habit
10. Ask for feedback
11. Learn a new skill
12. Believe in yourself
13. Start a new routine
14. Plan a new strategy
15. Learn how to invest
16. Don't make excuses
17. Stop procrastinating
18. Review your finances
19. Focus on the positive
20. Change your attitude
21. Learn a new language
22. Question your purpose
23. Focus one day at a time
24. Don't compare yourself
25. Automate and delegate
26. Acknowledge your flaws
27. Identify your blind spots
28. Leave your comfort zone
29. Change your social circle
30. See failure as opportunity
31. Follow an exercise routine
32. Make a business bucket list
33. Start being more organized
34. Quit things that aren't working
35. Write a letter to your future self
36. Watch educational videos instead

## Machine Learning & Data Science Competitions - ML Contests

<https://mlcontests.com/>






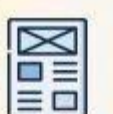









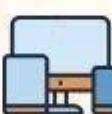








# THE EVOLUTION OF THE EMPLOYEE

PAST



FUTURE

- ←... Work 9-5 
- ←... Work in a corporate office 
- ←... Use company equipment 
- ←... Focused on inputs 
- ←... Climb the corporate ladder 
- ←... Pre-defined work 
- ←... Hoards information 
- ←... No voice 
- ←... Relies on email 
- ←... Focused on knowledge 
- ←... Corporate learning and teaching 

- ... Work anytime 
- ... Work anywhere 
- ... Use any device 
- ... Focused on outputs 
- ... Create your own ladder 
- ... Customized work 
- ... Shares information 
- ... Can become a leader 
- ... Relies on collaboration technologies 
- ... Focused on adaptive learning 
- ... Democratized learning and teaching 

by Jacob Morgan

thefutureorganization.com

# Taxonomy of GNN Learning Methods

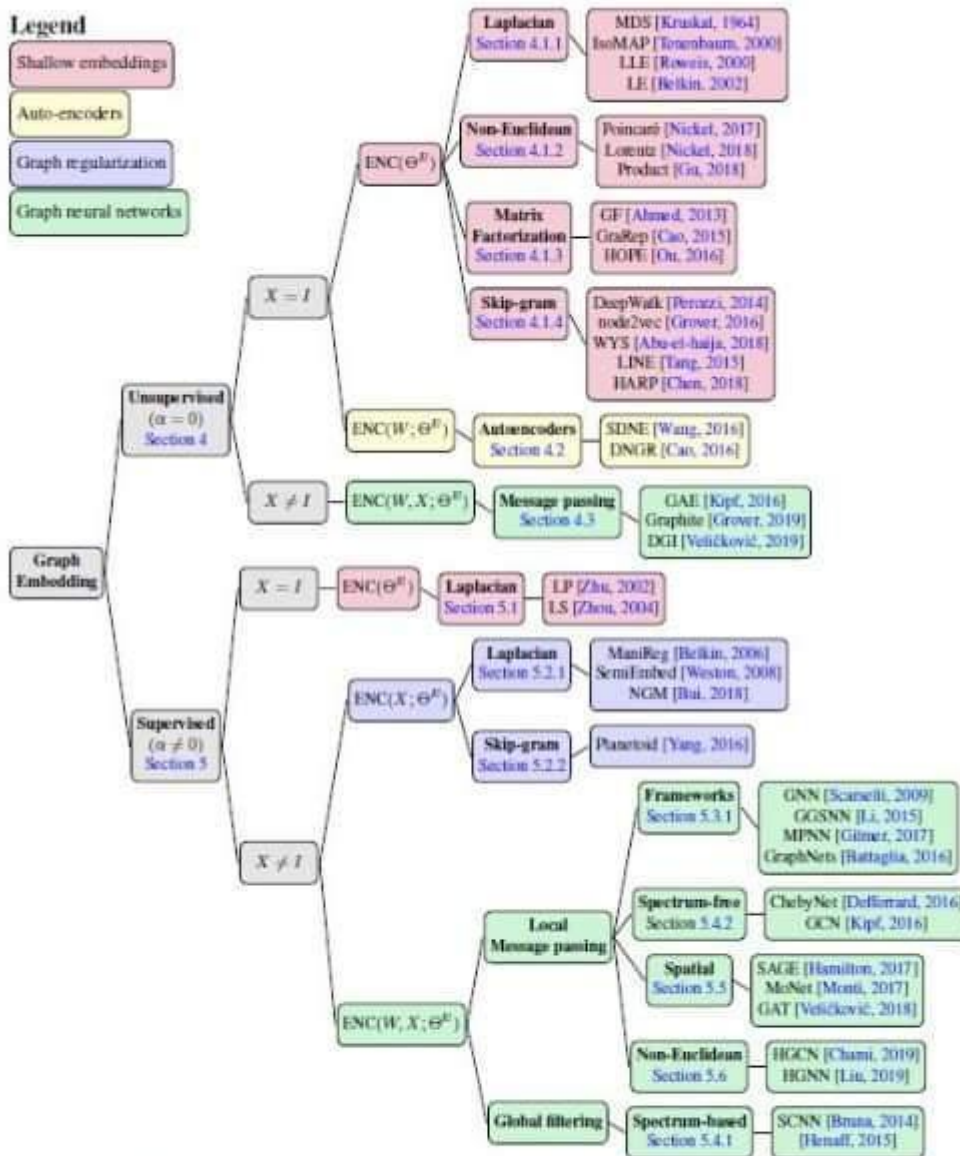


Figure 3: Taxonomy of graph representation learning methods. Based on what information is used in the encoder network, we categorize graph embedding approaches into four categories: shallow embeddings, graph auto-encoders, graph-based regularization and graph neural networks. Note that message passing methods can also be viewed as spatial convolution, since messages are computed over local neighborhood in the graph domain. Reciprocally, spatial convolutions can also be described using message passing frameworks.

## GNN for Novice

<https://rish16.notion.site/Graph-Neural-Networks-for-Novice-Math-Fanatics-c51b922a595b4efd8647788475461d57>

## Some free alternatives software to MATLAB

<https://perso.crans.org/besson/matlab-clones.en.html>



## 500 Page Book on Quantum Computing

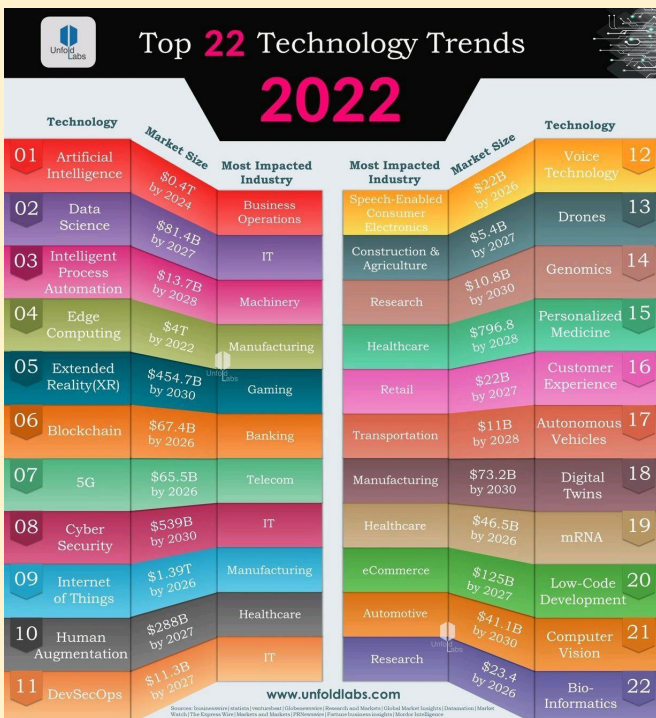
**Shlomo Kashani** · 2nd  
 Quantum Algorithms Researcher (11)  
 20h · Edited · 🌐

Jozef Gruska, made his 500-page QC book publicly available:  
<https://lnkd.in/ejbgkvHD>  
<https://lnkd.in/eBY8DDKW>

"In quantum computing, we witness an exciting and very promising merge of two of the deepest and most successful scientific and technological developments of this century: quantum physics and computer science. In spite of the fact that its experimental developments are in their infancy, there has already been a variety of concepts, models, methods and results obtained at the theoretical level that clearly have lasting value. These concepts, methods and results are the main subject of the book. Knowledge from two areas is of importance for understanding the basic developments in quantum computing: namely, quantum physics and theoretical computer science. The book provides elements of both, and concentrates on the presentation of concepts, models, methods and results mainly from a computing point of view. No previous knowledge of quantum mechanics is required."

#quantummechanics  
 #quantumcomputing #quantumalgorithms  
 #quantumapplications #quantumcomputer  
 #quantum #quantuminformationscience  
 #quantumresearch #quantumhardware  
 SPIN quantum

## 2022 Top 22 Tech Trends



## Acing Data Science Interview Questions



## How to Ace Your Data Science Interview in 6 Steps

By Catherine Adenle  
[www.catherinescareercorner.com](http://www.catherinescareercorner.com)

**Are you looking to break into the data science industry?** Data science seems to be the most popular and one of the highly paid jobs on the market. Any interviews can be daunting. For your Data Science interview, in 6 steps you can set yourself aside from the rest by showing the interviewer you are the right candidate for the job.

### 1. Research the role and the business you are applying to

Data Scientist is a blanket term, and there are many roles in the data science ecosystem. Such as Data Engineer, Data Analyst, Data Tester, Statistician, Data Architect, and many more. When you are applying for a data science job, research the requirement of the specific role and the organization.



### 2. Be able to back your Resume

Revise your resume several times a day before going for the interview. Filter out the things you can do without, mention the most applicable projects and skills required for the role.



### 3. Make your digital presence work for you

It is a fact that more than 80% of recruiters will first check a candidate's LinkedIn profile before scheduling an interview. Shows the value of your digital presence! We are in the digital age, which means a piece of paper with your resume on it will only take you so far.



### 4. Take a crack at puzzles and riddles

Data science recruiters love puzzles! It's highly likely when in the interview process, you'd be asked to solve puzzles, patterns, number sequences, and intelligence questions, etc.



### 5. Go through Data Science Glossary & Terms

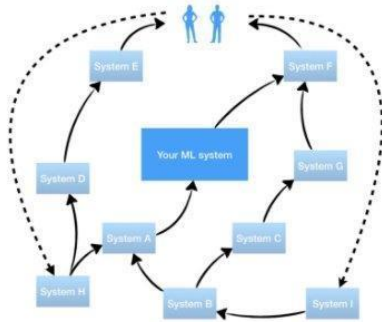


## Garbage Features

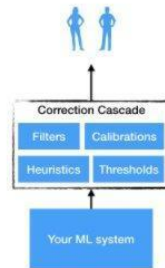
### Technical debt in AI/ML functions [2, 4]

20

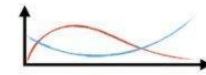
#### Garbage features



#### Feedback loops



Metrics after correction cascade  
(what you report to the stakeholders)



Metrics before correction cascade  
(what ML model optimizes for)

#### Correction cascades

Source: <https://towardsdatascience.com/technical-debt-in-machine-learning-8b0fae938657>

## SQL Training

<https://8weeksqlchallenge.com/getting-started/>

[https://youtube.com/playlist?list=PLdrw9\\_aIADIPAMJW8I\\_S-S747oyiRtzpS](https://youtube.com/playlist?list=PLdrw9_aIADIPAMJW8I_S-S747oyiRtzpS)

## Process Mining

<https://research.aimultiple.com/process-mining/>

<https://www.gartner.com/reviews/market/process-mining>

## Best Graph Neural Network architectures: GCN, GAT, MPNN and more \ AI Summer

<https://theaisummer.com/gnn-architectures/>

# Checkout Our List of Free Artificial Intelligence, Machine Learning, Data Science, Deep Learning, Mathematics, Python Programming Resources/Courses/Books

- Show All
- Data Science Resources
- Machine Learning Resources
- Deep Learning Resources
- Mathematics
- Reinforcement Learning
- Python



## Data Science Resources

- CS109 Data Science- Harvard 3
- Data Science Essentials- Microsoft 1
- Learning from Data – California Institute of Technology 1
- The Mathematics of Machine Learning by UC Berkeley 1
- "Foundations of Data Science" Book by Avrim Blum, John Hopcroft, and Ravindran Kannan 1
- Python Data Science Handbook by Jake VanderPlas 1

## Machine Learning Resources

- Introduction to Statistical Machine Learning Notes by Andreas Lindholm, Uppsala University 1
- Machine Learning Crash Course with TensorFlow APIs by Google 1
- The Mathematics of Machine Learning by UC Berkeley 1
- Understanding the major types of machine learning 1
- Self Supervised Learning 122 page lecture notebook by Andrew Zisserman from Deepmind and University of Oxford 1
- Stanford Machine Learning Notes From Andrew NG 1

MARKTECHPOST.COM  
Free AI Resources



## Merlion Architecture

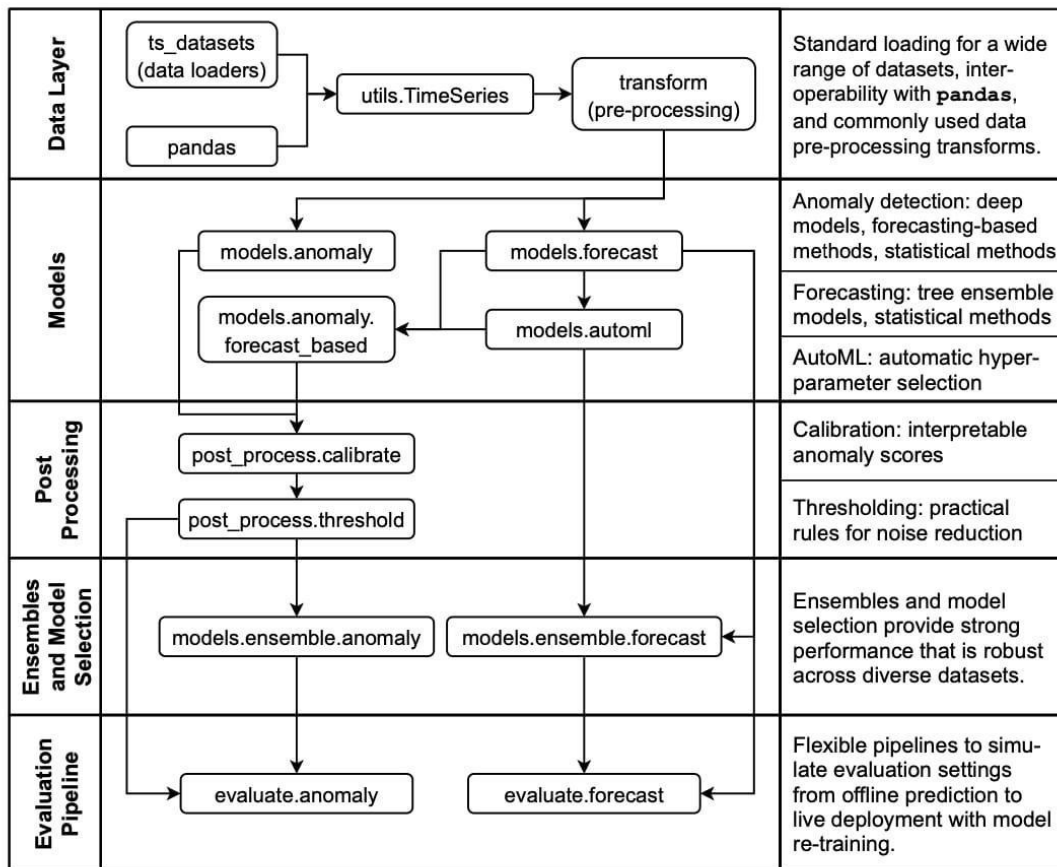


Figure 1: Architecture of modules in Merlion.

## 70 Free Online Courses for Data Science to Advance Your Skills in 2022

<https://www.mltut.com/free-online-courses-for-data-science/>

## Machine Learning Simplified

<https://www.themlsbook.com/>

## The most innovative AI algorithms released in 2022 (till date)

<https://analyticsindiamag.com/the-most-innovative-ai-algorithms-released-in-2022-till-date/>

## Transformer Takeover in AI

A powerful new approach to artificial intelligence allows computers to process and classify images in a new way.

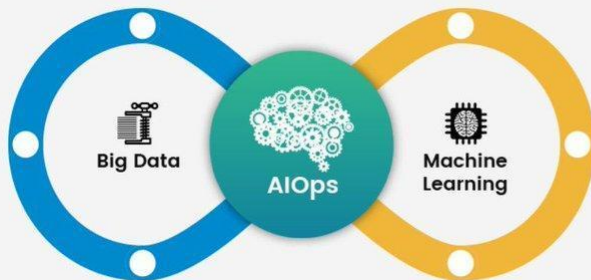
#AI #CNN #MachineLearning #DataScience #AITransformers

<https://quantamagazine.org/will-transformers-take-over-artificial-intelligence-20220310/>

## AIOPS

# WHAT IS AIOPS AND WHY DOES YOUR BUSINESS NEED IT?

AIops uses analytics and machine learning in order to analyze big data from various IT operation tools and devices. As a result, it is able to automatically spot and more importantly react to issues in real time.



Through digital transformation, you can add more business value by saving tons of time and effort, and spend more time in innovation. AIops adoption can help your organization to get end-to-end visibility into infrastructure and applications.



Infographic @ingliguori

source Botmetric

## AIops is not ITOA!

| ITOA                                                                              | AIops                                                     |
|-----------------------------------------------------------------------------------|-----------------------------------------------------------|
| Preceded AIops                                                                    | Evolved from ITOA                                         |
| IT operations analytics                                                           | Algorithmic IT operations                                 |
| Monitors and analyses historical operational data collected from multiple sources | Deals with real time data from various network conditions |

## DevOps vs AIops

| DevOps                                                                                                                  | AIops                                                                                                                                                                                                             |
|-------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Combination of cultural philosophies, practices, and tools                                                              | Multi-layered technology platforms that automates and enhances IT operations                                                                                                                                      |
| Delivers applications and services at high velocity                                                                     | Eliminates human errors and saves time                                                                                                                                                                            |
| Enables fast releases and deployment cycles by agile development methodologies                                          | Automatically spots and reacts to issues in real time and automation Smart DevOps and CloudOps                                                                                                                    |
| Operations collaborate with Developments to monitor self-service solutions                                              | Key components include Big Data and Machine Learning                                                                                                                                                              |
| Principles used by DevOps include Holistic system thinking, No silos, Rapid, useful feedback and Automate drudgery away | The principle used by AIops include taking inputs from existing monitoring tools, applying algorithmic techniques, analysing them, and producing an output that is an insight action item for the operations team |

## DS Books

- <https://jakevdp.github.io/PythonDataScienceHandbook/>
- <https://github.com/jakevdp/PythonDataScienceHandbook>
- <https://github.dev/jakevdp/PythonDataScienceHandbook/tree/master/notebooks>
- <https://colab.research.google.com/github/jakevdp/PythonDataScienceHandbook/blob/master/notebooks/Index.ipynb>
- [https://www.amazon.com/\\_/dp/1491912057?tag=oreilly20-20](https://www.amazon.com/_/dp/1491912057?tag=oreilly20-20)

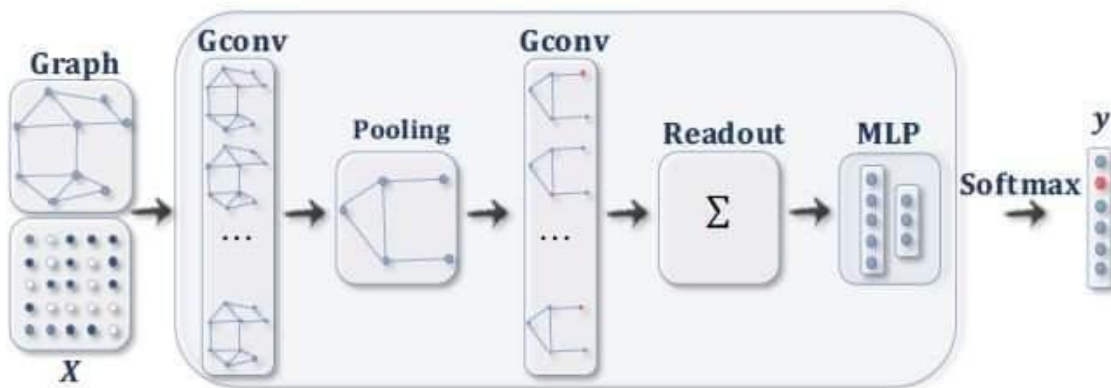
## Techprof free books

<https://www.techprofree.com/category/books/>

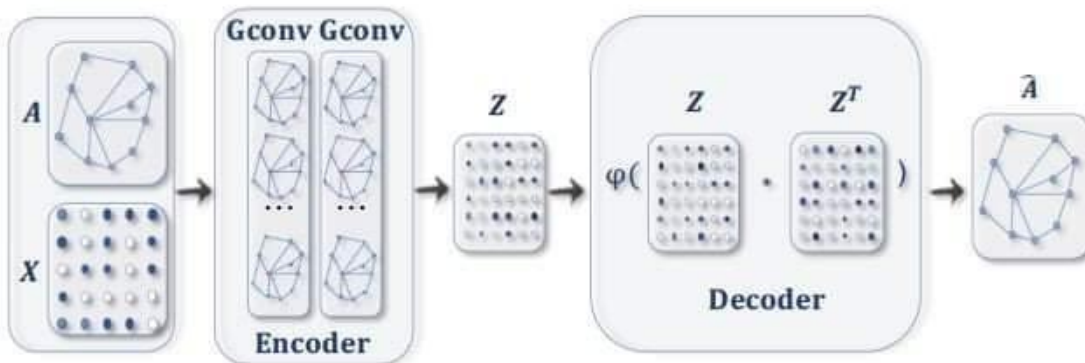
## Autoencoders-tutorial

1. <https://pythonprogramming.net/autoencoders-tutorial/>

## GNN



(b) A ConvGNN with pooling and readout layers for graph classification [21]. A graph convolutional layer is followed by a pooling layer to coarsen a graph into sub-graphs so that node representations on coarsened graphs represent higher graph-level representations. A readout layer summarizes the final graph representation by taking the sum/mean of hidden representations of sub-graphs.



(c) A GAE for network embedding [61]. The encoder uses graph convolutional layers to get a network embedding for each node. The decoder computes the pair-wise distance given network embeddings. After applying a non-linear activation function, the decoder reconstructs the graph adjacency matrix. The network is trained by minimizing the discrepancy between the real adjacency matrix and the reconstructed adjacency matrix.

Graph neural networks are rapidly advancing progress in ML for complex graph data applications.

In this week's newsletter, recent uses of GNNs ranging from protein interface prediction to collaborative filtering! Read on below:

<https://paperswithcode.com/newsletter/19/>

#GNN #AI #MachineLearning #DataScience

[github: 500-AI-Machine-learning-Deep-learning-Computer-vision](https://github.com/ashishpatel26/500-AI-Machine-learning-Deep-learning-Computer-vision-NLP-Projects-with-code)

<https://github.com/ashishpatel26/500-AI-Machine-learning-Deep-learning-Computer-vision-NLP-Projects-with-code>



## The NLP Index: 3,000+ code repos

The NLP Index: 3,000+ code repos for hackers and researchers. [Project]

[https://www.reddit.com/r/MachineLearning/comments/mzor46/the\\_nlp\\_index\\_3000\\_code\\_repos\\_for\\_hackers\\_and/](https://www.reddit.com/r/MachineLearning/comments/mzor46/the_nlp_index_3000_code_repos_for_hackers_and/)

#MachineLearning

<https://towardsdatascience.com/a-checklist-for-submitting-your-research-to-arxiv-64f31b4127d2?source=social.tw>

## github: Nyandwi/machine\_learning\_complete

[https://github.com/Nyandwi/machine\\_learning\\_complete](https://github.com/Nyandwi/machine_learning_complete)

## Leadership Lessons from APJ Kalam

**Leadership Lessons**  
FROM: DR. A.P.J. ABDUL KALAM  
BASED ON AN INTERVIEW WITH KNOWLEDGE@WHARTON 2009

1931 - 2015

WITHOUT **VISION** there is NO LEADERSHIP

a leader MUST BE ABLE TO TRAVEL INTO AN unexplored path

LEADERS MUST KNOW how to handle **SUCCESS** AND MORE IMPORTANTLY **failure**

LEADER MUST HAVE **courage** to make DECISIONS

TOUGH DECISIONS AHEAD

Leader should have **NOBILITY** in MANAGEMENT

EVERY ACTION OF A LEADER SHOULD BE **transparent**

A LEADER MUST WORK AND SUCCEED WITH **INTEGRITY**

@tnvora TANMAY VORA ~ QASPIRE.COM



## **Github: Python framework to explore, label, monitor data nlp projects**

<https://github.com/recognai/rubrix>

## **Maths for Data Science by IITM**

If anyone is looking for revising basic Mathematics , here is a beautiful playlist of Foundational BSc Data Science Maths course by IIT Madras :

[https://www.youtube.com/watch?v=8z04uTycZpE&list=PLZ2ps\\_\\_7DhBZYDZo9A0pZ\\_i0xhstrk5cR&index=5](https://www.youtube.com/watch?v=8z04uTycZpE&list=PLZ2ps__7DhBZYDZo9A0pZ_i0xhstrk5cR&index=5)

## **Datarobot**

<https://community.datarobot.com/t5/knowledge-base/introducing-drum/ta-p/9703/jump-to/first-unread-message>

## **github: SeldonIO**

<https://github.com/SeldonIO/seldon-core>

## **Internshala**

<https://internshala.com/>

## **konfhub**

<https://cc.konfhub.com/>

## **Machine Learning Projects in Python**

[https://m.youtube.com/playlist?list=PLKYEe2WisBTFSLeKs0aELKUkezOsYj\\_24](https://m.youtube.com/playlist?list=PLKYEe2WisBTFSLeKs0aELKUkezOsYj_24)

## **github: Merlion**

<https://github.com/salesforce/Merlion>

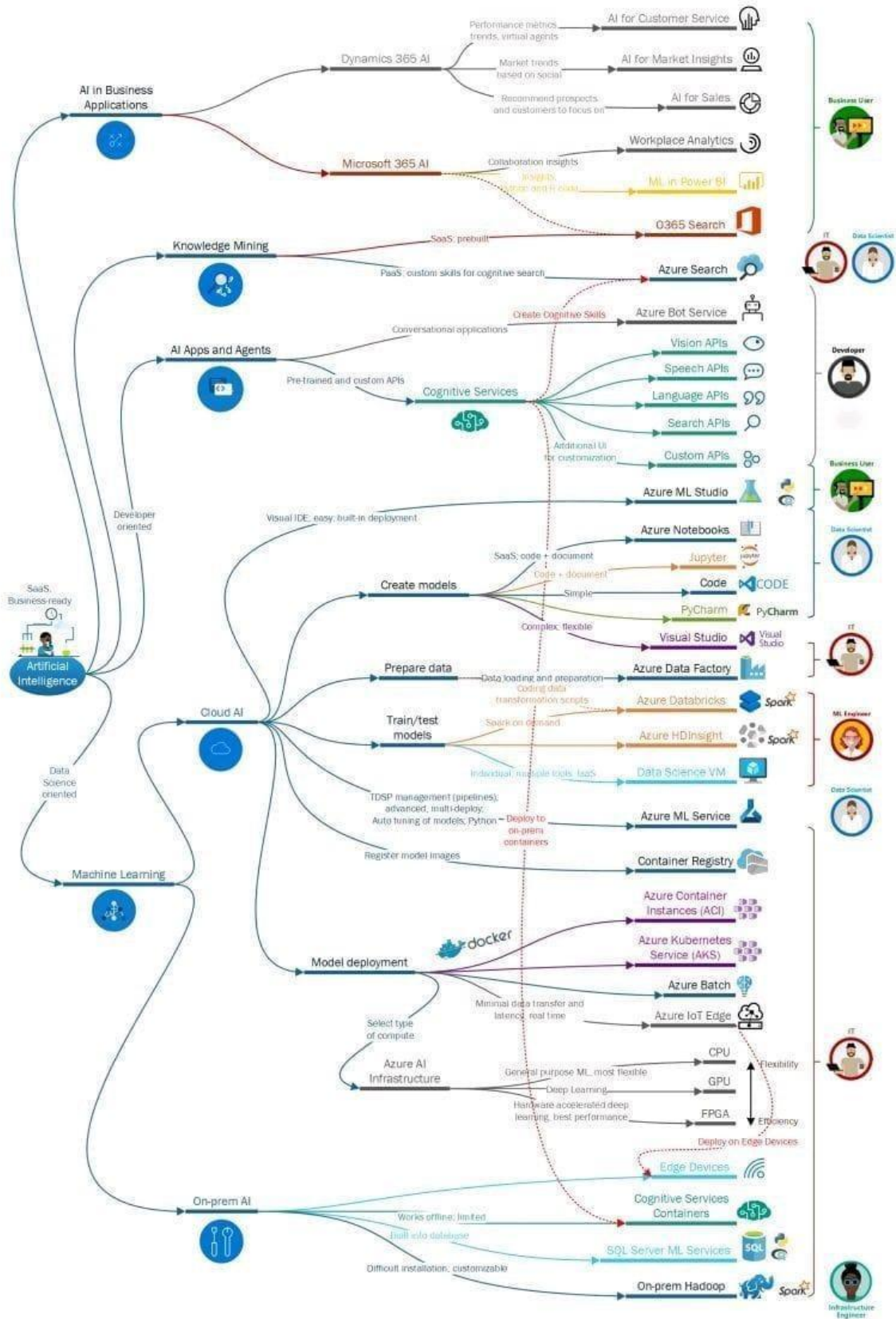
## **github: aws-samples ml-inference-using aws lambda**

<https://github.com/aws-samples/ml-inference-using-aws-lambda-and-amazon-efs>

## **AI Pipeline Tools**

# Artificial Intelligence Decision Tree

Version 2.02 February 16, 2019, Author: Ivan Kosyakov, Ph.D., Technical Architect in Microsoft Technology Center New York

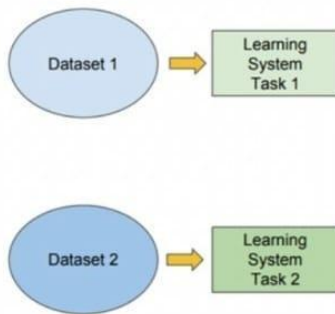


Download Artificial Intelligence Decision Tree (Mind Map) at <http://bizexcellence.com/2019/02/15/ai-dt>

## Traditional ML vs Transfer Learning

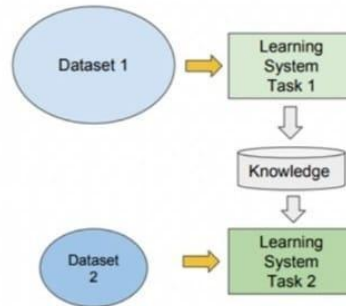
## Traditional ML

- Isolated, single task learning:
  - Knowledge is not retained or accumulated. Learning is performed w.o. considering past learned knowledge in other tasks



## Transfer Learning

- Learning of a new tasks relies on the previous learned tasks:
  - Learning process can be faster, more accurate and/or need less training data



Traditional Learning vs Transfer Learning

Source: <https://towardsdatascience.com/a-comprehensive-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a>

## How To Use The 4 Types Of Intelligence To Make Better Decisions

<https://www.mindbodygreen.com/articles/four-types-of-intelligence>

## Microsoft SDE Sheet - Top 35 Most Frequently asked

Complete SDE Sheet by Arsh with 280 Questions for preparing for Product Based Companies  
Subscribe to Arsh Goyal for more amazing content!

1. <https://practice.geeksforgeeks.org/problems/e015cb4d3f354b035d9665e7c8a54a7aefb1901b/1/>
2. <https://leetcode.com/problems/reverse-nodes-in-k-group/>
3. <https://leetcode.com/problems/copy-list-with-random-pointer/>
4. <https://practice.geeksforgeeks.org/problems/remove-every-kth-node/1/>
5. <https://leetcode.com/problems/spiral-matrix-ii/>
6. <https://leetcode.com/problems/edit-distance/> (Similar to Solar System Question)
7. <https://leetcode.com/problems/longest-palindromic-substring/>
  1. <https://leetcode.com/problems/encode-and-decode-tinyurl/> (Similar to Jack and his team)
  2. <https://leetcode.com/problems/string-to-integer-atoi/>
10. <https://leetcode.com/problems/longest-increasing-subsequence/>
11. <https://leetcode.com/problems/edit-distance/>
12. <https://leetcode.com/problems/number-of-digit-one/>
13. <https://leetcode.com/problems/number-of-provinces/> (Similar to Redmond Tech Park)
14. <https://leetcode.com/problems/reverse-nodes-in-k-group/>
15. <https://www.interviewbit.com/problems/hotel-bookings-possible/>
16. <https://leetcode.com/problems/longest-arithmetic-subsequence/>
17. <https://practice.geeksforgeeks.org/problems/is-binary-number-multiple-of-30654/1>
18. <https://leetcode.com/problems/distribute-candies/>

19. <https://leetcode.com/problems/satisfiability-of-equality-equations/>
20. <https://leetcode.com/problems/gas-station/>
21. <https://leetcode.com/problems/flatten-binary-tree-to-linked-list/>
22. <https://leetcode.com/problems/text-justification/>
23. <https://leetcode.com/problems/non-overlapping-intervals/>
24. <https://leetcode.com/problems/diagonal-traverse/> (Similar to Boss and colleagues)
25. <https://www.interviewbit.com/problems/hotel-bookings-possible/>
26. <https://leetcode.com/problems/sliding-window-median/>
27. <https://www.interviewbit.com/problems/meeting-rooms/> (Similar to Interview Rooms and Candidates)
28. <https://leetcode.com/problems/median-of-two-sorted-arrays/>
29. <https://leetcode.com/problems/satisfiability-of-equality-equations/>
  1. 30. <https://leetcode.com/problems/unique-paths/> /

## **Global AI Hub – Build your future**

<https://globalaihub.com/>

## **Keras community built 100 concise and clear code examples**

Keras community built 100 concise and clear code examples that demonstrate the latest deep learning models and techniques for computer vision, NLP, structured data, time series, generative DL, reinforcement learning, etc..

1. <https://keras.io/examples/>

## **Intro to machine learning compilers and optimizers**

1. A friendly introduction to machine learning compilers and optimizers  
<https://huyenchip.com/2021/09/07/a-friendly-introduction-to-machine-learning-compilers-and-optimizers.html>

#AI #DataScience #MachineLearning #DeepLearning

## **Approaches to process and classify images in a new way.**

A powerful new approach to artificial intelligence allows computers to process and classify images in a new way.

#AI #CNN #MachineLearning #DataScience #AITransformers

<https://quantamagazine.org/will-transformers-take-over-artificial-intelligence-20220310/>

## **Inductive link prediction in knowledge graphs**

1. <https://towardsdatascience.com/inductive-link-prediction-in-knowledge-graphs-23f249c31961>

## **TensorFlow tutorials**

- <https://youtube.com/playlist?list=PLZsOBAyNTZwbljGnolFydAN33ggyGP7IT>
- <https://youtu.be/qFJeN9V1ZsI>
- [https://youtube.com/playlist?list=PLeo1K3hjS3uu7CxAacxVndI4bE\\_o3BDtO](https://youtube.com/playlist?list=PLeo1K3hjS3uu7CxAacxVndI4bE_o3BDtO)
- Check these videos and also you can check tensorflow website.





| Project Management                         |                                         |                         |
|--------------------------------------------|-----------------------------------------|-------------------------|
| Topics                                     | Subtopics                               | Hrs                     |
| Introduction To Azure Boards               | Azure DevOps Services overview          |                         |
|                                            | Azure Boards overview                   |                         |
|                                            | The team project                        |                         |
|                                            | Creating and configuring a team project |                         |
|                                            | Securing a team project                 |                         |
|                                            | Teams, profiles, and notifications      |                         |
|                                            | Extending Azure Boards                  |                         |
|                                            | Creating and using a Wiki               |                         |
|                                            | Hands-on                                |                         |
|                                            | The Product Backlog                     | Process overview        |
| Choosing a process                         |                                         |                         |
| Working with the backlog                   |                                         |                         |
| Product Backlog Item vs Bug work item type |                                         |                         |
| Features and Epics backlogs                |                                         |                         |
| Decomposing large items                    |                                         |                         |
| Kanban board                               |                                         |                         |
| Tagging, querying, and charting work items |                                         |                         |
| Hands-on                                   |                                         |                         |
| Planning And Managing A Sprint             |                                         | Setting up a new sprint |
|                                            | Forecasting work in the backlog         |                         |
|                                            | The sprint backlog                      |                         |
|                                            | Creating a sprint plan                  |                         |
|                                            | The Task work item type                 |                         |
|                                            | The Test Case work item type            |                         |
|                                            | Working with the Task Board             |                         |
|                                            | Collaborating as a team                 |                         |
|                                            | Daily activities                        |                         |
|                                            | Implementing a definition of "Done"     |                         |

## Intelligent Graph

Getting Started with #Jupyter and #IntelligentGraph: <https://t.co/t3eXzp1L4a>

#abds #SPARQL #LinkedData #RDF #GraphDB #KnowledgeGraphs #BigData #DataScience #AI #MachineLearning #GraphAnalytics #Coding #Python #DataScientists

## Great NLP, Text Mining Talks post

Great NLP, Text Mining Talks post

Interested in NLP, text mining & AI applied to food, nutrition & health ?have a look at IberHeLT workshop talks exploring scientific literature, EHRs and social media for use cases related to food safety, food production, food-drug interactions,...

[https://www.linkedin.com/posts/martin-krallinger-85495920\\_the-presentations-of-the-language-technologies-activity-6861301475163811840-Oy\\_v?utm\\_source=linkedin\\_share&utm\\_medium=android\\_app](https://www.linkedin.com/posts/martin-krallinger-85495920_the-presentations-of-the-language-technologies-activity-6861301475163811840-Oy_v?utm_source=linkedin_share&utm_medium=android_app)

## Google Cloud Tools

- Cloud Storage
- Google Compute Engine (GCE Jupyter Notebook)
- Basic Google Cloud with Cloud Identity and Access Management (IAM)
- Cloud Functions
- Cloud Logging
- Dockerization (Basics)
- Cloud Run
- Cloud Build with Artifact Registry
- Cloud Scheduler
- BigQuery
- Kubeflow
- Vertex AI
- Airflow (DGAs, Variables)
- Pub/Sub

These ML models should be used in the example created while explaining other concepts (XGBoost, FBProphet, Simple Moving Average)

AI Platform (Jupyter Notebook) (Deploy using notebooks)

TFX basics

Basics Dataprep API

## 19 Datascience Podcasts

19 Podcasts to listen....

<https://theartistsofdatascience.fireside.fm/>

## Multivariate Time Series

MIT Researchers Create Tool to Forecast Multivariate Time Series: <https://t.co/6qMT221C8>

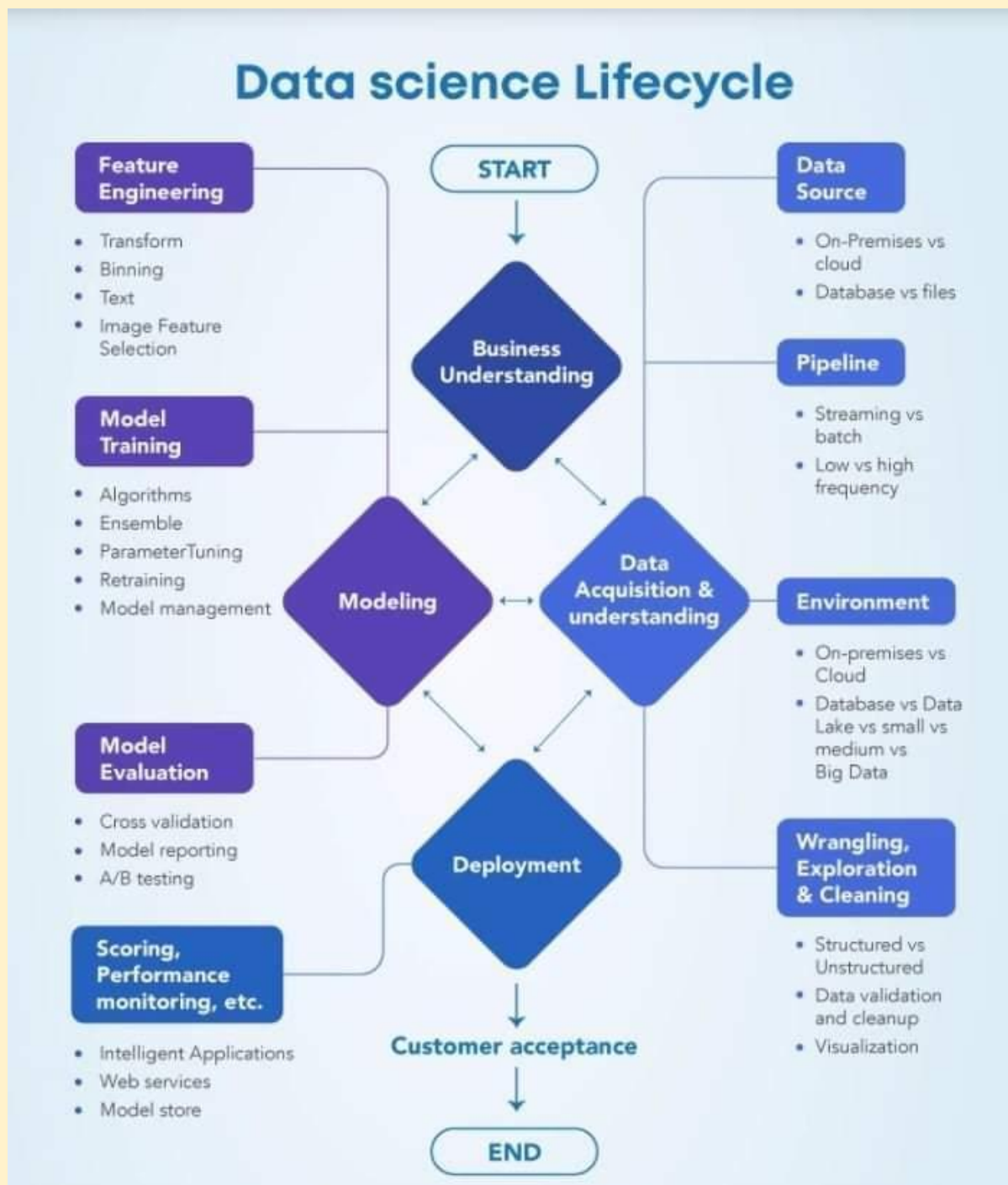
+

Research paper: <https://t.co/zC6WentTFo>

#TimeSeries #Forecasting #BigData #DataScience #MachineLearning #AI #PredictiveAnalytics

## SBI Innovate for Bank 2022 Hackathon Technology Partner Microsoft \ TechGig

<https://www.techgig.com/digital/sbi-microsoft>



## Data Science Lifecycle



## Building a Backtesting Service to Measure Model Performance at Uber-scale - Uber Engineering Blog

<https://eng.uber.com/backtesting-at-scale/>

## Gender Bias in Machine Translation

Google AI introduces Dataset for studying Gender Bias in Machine Translation.

<https://ai.googleblog.com/2021/06/a-dataset-for-studying-gender-bias-in.html>

## 10 Statistical Techniques for DS

<https://medium.com/nerd-for-tech/10-statistical-techniques-data-scientists-should-master-c52772f8e1cc>

## Spark NLP \ State of the Art Natural Language Processing \ John Snow Labs

<https://www.johnsnowlabs.com/spark-nlp/>

## Channels to Learn Python

A graphic with a black background. On the left is the YouTube logo. To its right, the text 'CHANNELS TO LEARN PYTHON' is written in large, bold, white and yellow letters. Below this, two columns of channel names are listed, each preceded by a white bullet point.

- *Al Sweigart*
- *Anaconda Inc.*
- *Christian Thompson*
- *Clever Programmer*
- *CodingEntrepreneurs*
- *Corey Schafer*
- *Chris Hawkes*
- *CS Dojo*
- *Data School (Kevin Markham)*
- *David Beazley*
- *Smaller Python Conferences*
- *Enthought*
- *Michael Kennedy (Talk Python)*
- *PrettyPrinted*
- *PyCon Session Recordings*
- *PyData*
- *Real Python*
- *Reuven Lerner*
- *Sentdex (Harrison Kinsley)*
- *Socratica*
- *TheNewBoston (Bucky Roberts)*

## What is Neural Network? How does it understand things?

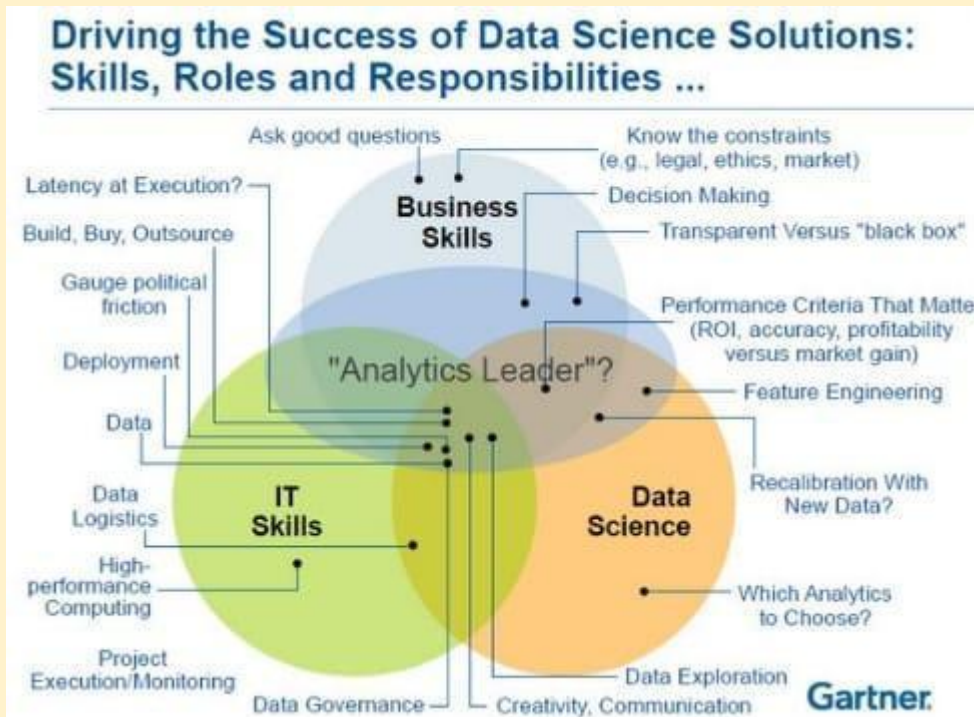
<https://glweb.eu/blog/digital-transformation/187-what-is-neural-network-how-does-it-understand-things>

## World's Leading AI and Technology Publication

This University Released a Free, Online Data Science Course (With Credits) – Towards AI — The World's Leading AI and Technology Publication

<https://towardsai.net/p/education/this-university-released-a-free-online-data-science-course-with-credits>

## Skills, Roles and Responsibilities in Data Science



## Complexity Explorer

<https://www.complexityexplorer.org/>

## Network Science by Albert-László Barabási

<http://networksciencebook.com/>

## Graph Neural Networks: Models and Applications

<http://cse.msu.edu/~mayao4/tutorials/aaai2021/>

78mb ppt

## NLP in Notebooks Competition

<https://deepnote.notion.site/NLP-in-Notebooks-Competition-6616e415f0a44e5c95982e7bc1cb89dd>

## GNN TimeSeries Intelligence

Let us explore below with Graph Neural Networks (GNNs) too.

#TimeSeries #ML #Merlion #GNN

<https://m.youtube.com/watch?v=rAYV1faUNuo&feature=youtu.be>

## Tuning Neural Networks Part I: normalize your data

1. <https://link.medium.com/tl7h7W8Eikb>

## Multi-paragraph segmentation

Check Multi-paragraph segmentation of expository text

(1994) : Hearst, Marti A

DOI: <https://t.co/n6ucaGOomo>

#text\_tiling #NLP #algorithm #discourse

#my\_bibtex

[15/09, 13:22] Bhaskar@upga: txtai now has a segmentation pipeline that splits text into sentences, lines and/or paragraphs. This can be used to customize how text is searched, indexed and processed.

<https://t.co/VJc0KyJrLc>

#machinelearning #ai #nlp #datascience #python #javascript #java #rustlang #golang

### **Machine Learning 2020 summary: 84 interesting papers/articles**

<https://link.medium.com/CLCnGCxWyjb>

### **GitHub Announces GitHub Open-Source Grants Recipients In India**

<https://analyticsindiamag.com/github-announces-github-open-source-grants-recipients-in-india/>

### **Watch "Python Tutorial For Beginners In Hindi (With Notes) 🔥" on YouTube**

<https://youtu.be/gfDE2a7MKjA>

### **Quantum Computing Course**

<https://quantum-aera.thinkific.com/courses/quantum-computing-with-qiskit-masterclass>

## Optimum: The Optimization Toolkit for Transformers at Scale

📣 Introducing 😊 Optimum

A new open-source library to optimize 😊  
#Transformers for production performance. 🚀  
Hugging Face

#deeplearning #artificialintelligence  
#machinelearning



Hugging Face

36,545 followers

1d • 🌐

📣 Introducing 😊 Optimum

A new open source library to optimize 😊  
Transformers for production performance. 🚀

Quantize, Prune, Optimize models easily, targeting  
hardware from our partners [Intel Corporation](#)  
[Graphcore](#) [Qualcomm](#) ! 🤗

<https://lnkd.in/gzZEbDDS>

#ai #machinelearning #hardware

### Introducing 😊 Optimum: The Optimization Toolkit for Transformers at Scale

Published September 14, 2021.

[Update on GitHub](#)



[mfuntowicz](#)  
Morgan Funtowicz



[echarlaix](#)  
Ella Charlaix



[michaelbenayoun](#)  
Michael Benayoun



[jeffboudier](#)  
Jeff Boudier

This post is the first step of a journey for Hugging Face to democratize state-of-the-art **Machine Learning production performance**. To get there, we will work hand in hand with our Hardware Partners, as we have with Intel below. Join us in this journey, and follow [Optimum](#), our new open source library!

### Introducing Optimum: The Optimization Toolkit for Transformers at Scale

[huggingface.co](https://huggingface.co) • 5 min read

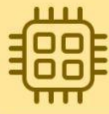


## CPU, GPU, TPU, FPGA



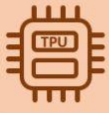
### CPU

- Small models
- Small datasets
- Useful for design space exploration



### GPU

- Medium-to-large models, datasets
- Image, video processing
- Application on CUDA or OpenCL



### TPU

- Matrix computations
- Dense vector processing
- No custom TensorFlow operations



### FPGA

- Large datasets, models
- Compute intensive applications
- High performance, high perf./cost ratio

## Textless NLP: Generating expressive speech from raw audio

<https://ai.facebook.com/blog/textless-nlp-generating-expressive-speech-from-raw-audio/>

## Textless NLP: Generating expressive speech from raw audio

<https://ai.facebook.com/blog/textless-nlp-generating-expressive-speech-from-raw-audio/>

## How Data-Centric Platforms Solve the Biggest Challenges for MLOps - The Databricks Blog

<https://databricks.com/blog/2021/06/23/need-for-data-centric-ml-platforms.html>

## Best Resources to Learn Natural Language Processing in 2021 - KDnuggets

<https://www.kdnuggets.com/2021/09/best-resources-learn-natural-language-processing-2021.html>

## Index of Best AI/Machine Learning Resources \ Hacker Noon

[https://hackernoon.com/index-of-best-ai-machine-learning-resources-71ba0c73e34d?utm\\_content=bufferd589f&utm\\_medium=social&utm\\_source=twitter.com&utm\\_campaign=buffer#.8ilaqcc4w](https://hackernoon.com/index-of-best-ai-machine-learning-resources-71ba0c73e34d?utm_content=bufferd589f&utm_medium=social&utm_source=twitter.com&utm_campaign=buffer#.8ilaqcc4w)

## Qwiklabs - Hands-On Cloud Training

<https://www.qwiklabs.com/>

## Introduction to Graph Neural Network (GNN) \ Analytics Steps

<https://www.analyticssteps.com/blogs/introduction-graph-neural-network-gnn?s=09>

## Watch "Auto Tensorflow - End to End AutoML solution - Exclusive Launch" on YouTube

<https://youtu.be/sil-RbuckG0>

## **Graph Database, GraphQL and Machine Learning for Carbohydrate-Active Enzymes**

<https://link.medium.com/INxjf5C2Fjb>

## **Pydantic experiment**

<https://link.medium.com/V8roCDkdHjb>

## **Graph ML**

<https://link.medium.com/EdDYUtW4Fjb>

## **Reinforcement Learning Lecture Series 2021 \ DeepMind**

<https://deepmind.com/learning-resources/reinforcement-learning-series-2021>

## **Advances in Cybernetics, Cognition, and Machine Learning for Communication ... - Google Books**

[https://books.google.com/books/about/Advances\\_in\\_Cybernetics\\_Cognition\\_and\\_Ma.html?id=v3HgDwAAQBAJ](https://books.google.com/books/about/Advances_in_Cybernetics_Cognition_and_Ma.html?id=v3HgDwAAQBAJ)

## **Visualization — Effective Python for Data Scientists**

<https://khuyentran1401.github.io/Efficient Python tricks and tools for data scientists/Chapter5/visualization.html#hiplot-high-dimensional-interactive-plotting>

## **The NLP Index**

<https://index.quantumstat.com/>

## **Github: Anomaly Detection Open source**

<https://github.com/opensearch-project/anomaly-detection>

## **Patsy: Build Features with Arbitrary Python Code - Data Science Simplified**

<https://mathdatasimplified.com/2021/10/08/patsy-build-features-with-arbitrary-python-code/>

## **Introducing OpenSearch \ AWS Open Source Blog**

<https://aws.amazon.com/blogs/opensource/introducing-opensearch/>

## **End-to-End Differentiable Molecular Mechanics Force Field Construction**

<https://arxiv.org/abs/2010.01196>

## **Learning resources \ DeepMind**

<https://deepmind.com/learning-resources>

## **Understanding LSTM Networks -- colah's blog**

<http://colah.github.io/posts/2015-08-Understanding-LSTMs/>

## **Machine Learning Models as Micro Services in Docker**

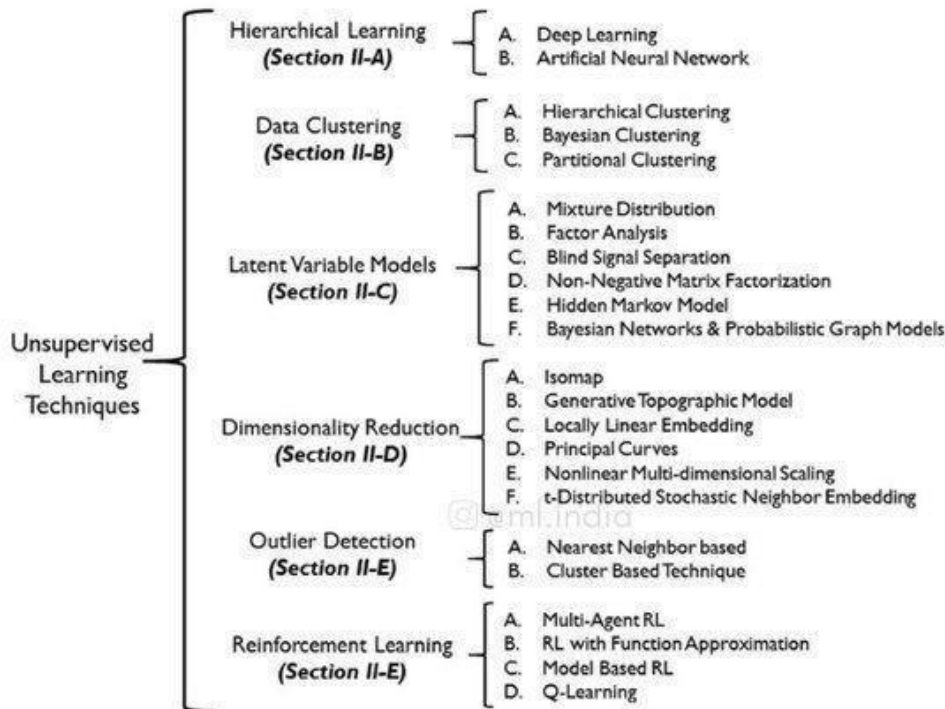
1. <https://link.medium.com/OAUeS2c6hkb>

## Unsupervised Learning Techniques

HIT LIKE TO SUPPORT!

SAVE FOR LATER!

# Unsupervised Learning Techniques



Source: Unsupervised Machine Learning for Networking: Techniques, Applications and Research Challenges, [www.researchgate.net](http://www.researchgate.net)



## A 10 free 'Springer' Books on the Topics of AI, Ethics, Machine Learning, Robotics, Cybersecurity, Philosophy, Science & Innovation

1. <https://link.medium.com/mhufpB4Eikb>

## Deep Learning on Graphs: Past, Present, And Future :

Knowledge Connexions Conference 2020 by Knowledge Connexions

<https://www.knowledgeconnexions.world/talks/deep-learning-on-graphs-past-present-and-future/>

## Watch "Ask the Expert: VS Code: Development in the Open" on YouTube

[https://youtu.be/Aopij\\_AxWdQ](https://youtu.be/Aopij_AxWdQ)

## Analytics Tools

SAS

SAS is a software suite that can mine, alter, manage and retrieve data from a variety of sources and perform statistical analysis on it. SAS provides a graphical point-and-click user interface for non-technical

users and more advanced options through the SAS programming language. SAS programs have a DATA step, which retrieves and manipulates data, usually creating a SAS data set, and a PROC step, which analyses the data.

#### WPS

WPS can use programs written in the language of SAS without the need for translating them into any other language. In this regard WPS is compatible with the SAS system. WPS is a language interpreter able to process the language of SAS and produce similar results. It is sometimes used as an alternative to SAS as it is relatively cheaper.

#### MS Excel

Microsoft Excel is a spreadsheet application developed by Microsoft for Microsoft Windows and Mac OS. It features calculation, graphing tools, pivot tables, and a macro programming language called Visual Basic for Applications. It has been a very widely applied spreadsheet for these platforms, especially since version 5 in 1993, and it has replaced Lotus 1-2-3 as the industry standard for spreadsheets. Excel forms part of Microsoft Office.

#### Tableau

Tableau Software is an American computer software company headquartered in Seattle, Washington. It produces a family of interactive data visualization products focused on business intelligence. Tableau offers five main products: Tableau Desktop, Tableau Server, Tableau Online, Tableau Reader and Tableau Public.

#### Pentaho

Pentaho is a company that offers Pentaho Business Analytics, a suite of open source Business Intelligence (BI) products which provide data integration, OLAP services, reporting, dash boarding, data mining and ETL capabilities. Pentaho was founded in 2004 by five founders and is headquartered in Orlando, FL, USA. The Pentaho suite consists of two offerings, an enterprise and community edition. The enterprise edition contains extra features not found in the community edition.

#### Statistica

STATISTICA is a statistics and analytics software package developed by StatSoft. STATISTICA provides data analysis, data management, statistics, data mining, and data visualization procedures. STATISTICA product categories include Enterprise (for use across a site or organization), Web-Based (for use with a server and web browser), Concurrent Network Desktop, and Single-User Desktop.

#### Qlikview

Qlikview is a business intelligence software from Qlik. It helps its users understand the business in a better way by providing them features like consolidating relevant data from multiple sources, exploring the various associations in the data, enabling social decision making through secure, real-time collaboration etc.

Qlikview is a business intelligence software from Qlik. It helps its users understand the business in a better way by providing them features like consolidating relevant data from multiple sources, exploring the various associations in the data, enabling social decision making through secure, real-time collaboration etc.

#### KISSmetrics

KISSmetrics is a person-based analytics product that helps users identify, understand, and improve the metrics that drive their online business. They make it simple to get the information users need to make better product and marketing decisions.

#### WeKa

The Weka workbench contains a collection of visualization tools and algorithms for data analysis and predictive modelling, together with graphical user interfaces for easy access to this functionality. The original non-Java version of Weka was a TCL/TK front-end to (mostly third-party) modelling algorithms implemented in other programming languages, plus data pre-processing utilities in C, and a Make

file-based system for running machine learning experiments. This original version was primarily designed as a tool for analysing data from agricultural domains, but the more recent fully Java-based version (Weka 3), for which development started in 1997, is now used in many different application areas, in particular for educational purposes and research.

## BigML

BigML is a Corvallis, Ore.-based startup with a SaaS-based machine learning platform that allows everyday business users to create actionable predictive models within minutes.

## R

R provides a wide variety of statistical and graphical techniques, including linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, and others. R is easily extensible through functions and extensions, and the R community is noted for its active contributions in terms of packages. There are some important differences, but much code written for S runs unaltered. Many of R's standard functions are written in R itself, which makes it easy for users to follow the algorithmic choices made. For computationally intensive tasks, C, C++, and FORTRAN code can be linked and called at run time. Advanced users can write C, C++, Java or Python code to manipulate R objects directly.

## Google Analytics

Google Analytics is a service offered by Google that generates detailed statistics about a website's traffic and traffic sources and measures conversions and sales. The product is aimed at marketers as opposed to webmasters and technologists from which the industry of web analytics originally grew. It's the most widely used website statistics service.

## Python

Pandas (python data analysis library) is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language. It provides tools for reading and writing data between in-memory data structures and different formats: CSV and text files, Microsoft Excel, SQL databases, and the fast HDF5 format and also intelligent data alignment and integrated handling of missing data: gain automatic label-based alignment in computations and easily manipulate messy data into an orderly form can be done using python

## Spotfire

TIBCO Spotfire is an analytics and business intelligence platform for analysis of data by predictive and complex statistics. During the 2010 World Cup, FIFA used this software to give viewers analytics on country teams' past performances.

## Online Data Science Course from Univ

This University Released a Free, Online Data Science Course (With Credits)

<https://link.medium.com/hZfXeFCrGkb>

## ML Powered Apps vs Traditional Programming

ML Powered Apps vs Traditional Programming

Source of errors in building traditional programs:

- ◆ Wrong syntaxes
- ◆ Inefficient codes



Source of errors in machine learning:

- ◆ Solving a wrong problem
- ◆ Using a wrong evaluation metric
- ◆ Not being aware of a skewed data
- ◆ Inconsistent data preprocessing functions

More sources of errors in ML:

- ◆ Putting too much emphasis on models than data
- ◆ Data leakage
- ◆ Training on the test data
- ◆ Model and data drifts

The difference between errors in traditional software and machine learning-powered applications:

The errors in traditional systems are obvious but not always the case in the latter. In machine learning, you may not get any errors but still, get misleading predictions.

How to minimize errors in machine learning:

- ◆ Use the right metric relevant to the problem
- ◆ Use consistent data preprocessing functions between training and test set
- ◆ Get the data right before the model
- ◆ Do not leak the test data to the model
- ◆ Avoid data leakage in general.
  
- ◆ Monitor the predictions and set up automatic actions(ex: retraining) in case of data and mode drifts.

Released a complete machine learning package containing over 30 end to end notebooks for:

- ◆ Data analysis
- ◆ Data visualization
- ◆ Data cleaning
- ◆ Classical ML
- ◆ Computer vision
- ◆ Natural language processing

Everything is now accessible here:

[https://github.com/Nyandwi/machine\\_learning\\_complete](https://github.com/Nyandwi/machine_learning_complete)

#MLPoweredAppsvsTraditionalPrograms

**Task: Introduction to Deep Learning**

<https://sebastianraschka.com/blog/2021/dl-course.html>

Must finish this task for my course

## How Pokémon GO scales to millions of requests? \ Google Cloud Blog

<https://cloud.google.com/blog/topics/developers-practitioners/how-pok%C3%A9mon-go-scales-millions-requests>

## A programmers guide to data mining

[www.guidetodatamining.com](http://www.guidetodatamining.com)

These methods of finding relevant stuff— friends, experts, the thing itself—are still present today but we need some computational help to transform them into the 21st century where we have billions of choices.

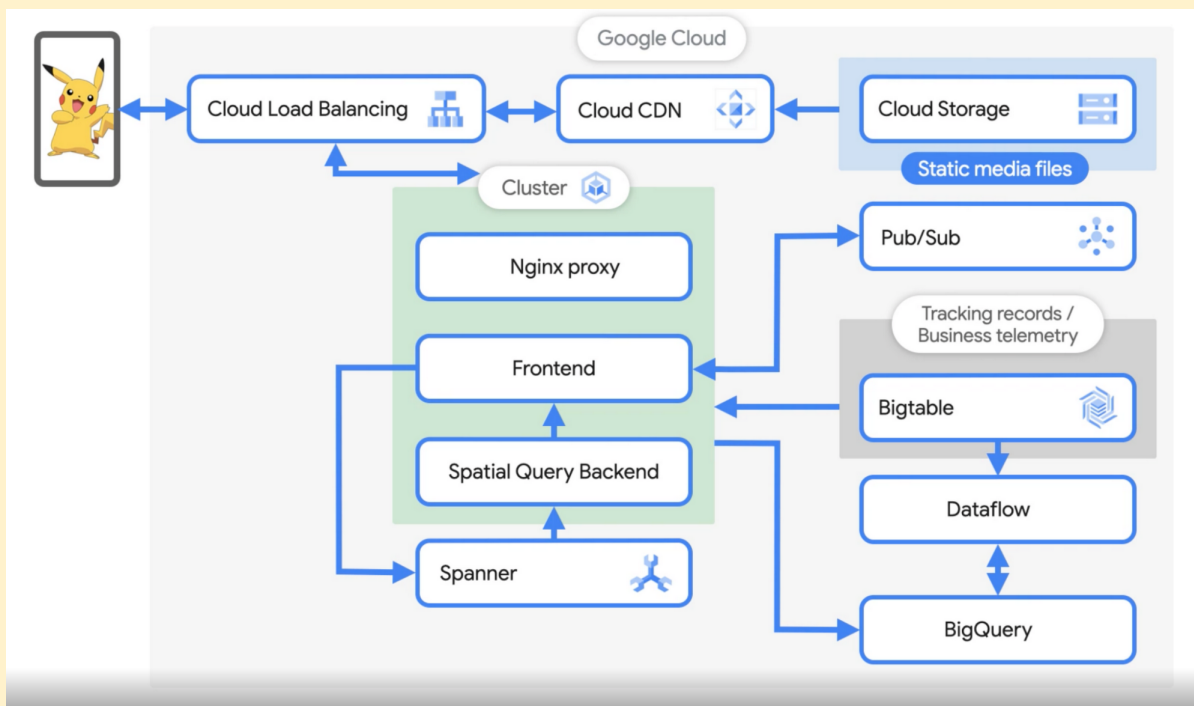
Every minute 24 hours of video is uploaded to YouTube. Every hour 180 new books are published

On October 11, 2001 the US government passed the USA Patriot Act (short for Uniting and Strengthening America by Providing Appropriate Tools Required to Intercept and Obstruct Terrorism).

Data mining is focused on finding patterns in data. At the small scale, we are expert at building mental models and finding patterns. I want to watch a movie tonight with my wife. I have a mental model of what she likes.

Data science ai data mining helping us knowing about everything.after somethings about a old women standing at one side of road you want help her and you make her cross the road. She was not interested.

## Pokeman go architecture



## **GNN summary paper**

A Comprehensive Survey on Graph Neural Networks (Wu 2019)

Paper: <https://arxiv.org/abs/1901.00596>

Abstract:

"Deep learning has revolutionized many machine learning tasks in recent years, ranging from image classification and video processing to speech recognition and natural language understanding. The data in these tasks are typically represented in the Euclidean space. However, there is an increasing number of applications where data are generated from non-Euclidean domains and are represented as graphs with complex relationships and interdependency between objects. The complexity of graph data has imposed significant challenges on existing machine learning algorithms. Recently, many studies on extending deep learning approaches for graph data have emerged. In this survey, we provide a comprehensive overview of graph neural networks (GNNs) in data mining and machine learning fields. We propose a new taxonomy to divide the state-of-the-art graph neural networks into four categories, namely recurrent graph neural networks, convolutional graph neural networks, graph autoencoders, and spatial-temporal graph neural networks. We further discuss the applications of graph neural networks across various domains and summarize the open source codes, benchmark data sets, and model evaluation of graph neural networks. Finally, we propose potential research directions in this rapidly growing field."

Graph Models: <https://paperswithcode.com/methods/category/graph-models>

## **Regression Cheatsheet**

# Cheat Sheet – Regression Analysis

## What is Regression Analysis?

Fitting a function  $f(\cdot)$  to datapoints  $y_i=f(x_i)$  under some error function. Based on the estimated function and error, we have the following types of regression

### 1. Linear Regression:

Fits a **line** minimizing the **sum of mean-squared error** for each datapoint.

$$\min_{\beta} \sum_i \|y_i - f_{\beta}^{\text{linear}}(x_i)\|^2$$

$$f_{\beta}^{\text{linear}}(x_i) = \beta_0 + \beta_1 x_i$$

### 2. Polynomial Regression:

Fits a **polynomial** of order  $k$  ( $k+1$  unknowns) minimizing the **sum of mean-squared error** for each datapoint.

$$\min_{\beta} \sum_{i=0}^m \|y_i - f_{\beta}^{\text{poly}}(x_i)\|^2$$

$$f_{\beta}^{\text{poly}}(x_i) = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \dots + \beta_k x_i^k$$

### 3. Bayesian Regression:

For each datapoint, fits a **gaussian distribution** by minimizing the **mean-squared error**. As the number of data points  $x_i$  increases, it converges to point estimates i.e.  $n \rightarrow \infty, \sigma^2 \rightarrow 0$

$$\min_{\beta} \sum_i \|y_i - \mathcal{N}(f_{\beta}(x_i), \sigma^2)\|^2$$

$$f_{\beta}(x_i) = f_{\beta}^{\text{poly}}(x_i) \text{ or } f_{\beta}^{\text{linear}}(x_i)$$

$$\mathcal{N}(\mu, \sigma^2) \rightarrow \text{Gaussian with mean } \mu \text{ and variance } \sigma^2$$

### 4. Ridge Regression:

Can fit either a **line**, or **polynomial** minimizing the **sum of mean-squared error** for each datapoint and the **weighted L2 norm** of the function parameters beta.

$$\min_{\beta} \sum_{i=0}^m \|y_i - f_{\beta}(x_i)\|^2 + \sum_{j=0}^k \beta_j^2$$

$$f_{\beta}(x_i) = f_{\beta}^{\text{poly}}(x_i) \text{ or } f_{\beta}^{\text{linear}}(x_i)$$

### 5. LASSO Regression:

Can fit either a **line**, or **polynomial** minimizing the the **sum of mean-squared error** for each datapoint and the **weighted L1 norm** of the function parameters beta.

$$\min_{\beta} \sum_{i=0}^m \|y_i - f_{\beta}(x_i)\|^2 + \sum_{j=0}^k |\beta_j|$$

$$f_{\beta}(x_i) = f_{\beta}^{\text{poly}}(x_i) \text{ or } f_{\beta}^{\text{linear}}(x_i)$$

### 6. Logistic Regression (NOT regression, but classification):

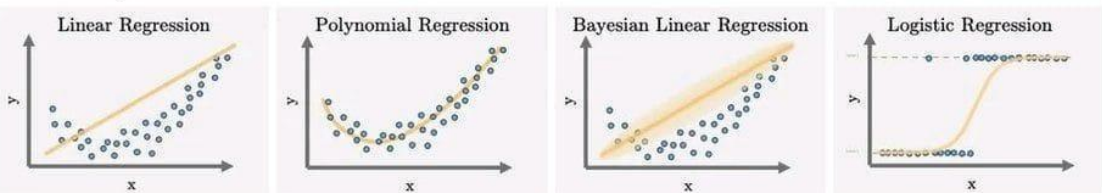
Can fit either a **line**, or **polynomial** with **sigmoid activation** minimizing the **sum of mean-squared error** for each datapoint. The labels  $y$  are binary class labels.

$$\min_{\beta} \sum_i \|y_i - \sigma(f_{\beta}(x_i))\|^2$$

$$f_{\beta}(x_i) = f_{\beta}^{\text{poly}}(x_i) \text{ or } f_{\beta}^{\text{linear}}(x_i)$$

$$\sigma(t) = \frac{1}{1 + e^{-t}}$$

## Visual Representation:



## Summary:

|                 | What does it fit?                    | Estimated function                                                             | Error Function                                                     |
|-----------------|--------------------------------------|--------------------------------------------------------------------------------|--------------------------------------------------------------------|
| Linear          | A line in $n$ dimensions             | $f_{\beta}^{\text{linear}}(x_i) = \beta_0 + \beta_1 x_i$                       | $\sum_{i=0}^m \ y_i - f_{\beta}(x_i)\ ^2$                          |
| Polynomial      | A polynomial of order $k$            | $f_{\beta}^{\text{poly}}(x_i) = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \dots$ | $\sum_{i=0}^m \ y_i - f_{\beta}(x_i)\ ^2$                          |
| Bayesian Linear | Gaussian distribution for each point | $\mathcal{N}(f_{\beta}(x_i), \sigma^2)$                                        | $\sum_i \ y_i - \mathcal{N}(f_{\beta}(x_i), \sigma^2)\ ^2$         |
| Ridge           | Linear/polynomial                    | $f_{\beta}^{\text{poly}}(x_i) \text{ or } f_{\beta}^{\text{linear}}(x_i)$      | $\sum_{i=0}^m \ y_i - f_{\beta}(x_i)\ ^2 + \sum_{j=0}^k \beta_j^2$ |
| LASSO           | Linear/polynomial                    | $f_{\beta}^{\text{poly}}(x_i) \text{ or } f_{\beta}^{\text{linear}}(x_i)$      | $\sum_{i=0}^m \ y_i - f_{\beta}(x_i)\ ^2 + \sum_{j=0}^k  \beta_j $ |
| Logistic        | Linear/polynomial with sigmoid       | $\sigma(f_{\beta}(x_i))$                                                       | $\sum_{i=0}^m \ y_i - f_{\beta}(x_i)\ ^2$                          |

## Github: Data deduplication tool

<https://github.com/zinggAI/zingg>

<https://www.github.com/zinggAI/zingg/tree/main/docs%2Fpatient.md>

## Kappa for model evaluation

A dataset of 600 students

It contains 10 movies rating (10 columns) by these 600 students.

Target column contains the stream name selected by these phy, chem, maths, bio

You build a predictor who can predict the subject selected by these.

Now you want to test the effectiveness of this model.

Calculate frequency using random classifier

Calculate accuracy of this random classifier

Calculate accuracy of real classifier

Calculate kappa = (a of real - a of random) / (1 - a of random)

Higher the kappa better the model.

## Data science resources

1. Scikit Learn : <https://scikit-learn.org/stable/tutorial/index.html>. Examples that illustrate the main features of the releases of scikit-learn <https://t.co/cBibhvyHeD>
2. Numpy: For mathematical and scientific computing. <https://t.co/vMBwQhQ8iF>
3. Complete Python Bootcamp 2021 : Beginner to Advanced. Learn Python fast like a pro with Hands On Advanced training: Python Code Implementations and Exercises with Instructor. <https://t.co/vX2oGkxCmt>
4. Pandas: Complete Hands On Pandas Bootcamp with Real World Project. <https://t.co/UeQSozNjb5>
5. Matplotlib : <https://t.co/N8qFx4S7li>
6. Tableau. Create amazing charts maps : <https://t.co/R32F3YB5Va>
7. Open CV: <https://t.co/Xazz5pr4mr>
8. Google's Machine Learning Zero to Hero. <https://t.co/lCu5muKmXg>
9. @lexfridman course Deep Learning Basics: Introduction and Overview. <https://t.co/Db8ErGPv8C>
10. Neural Network. @3blue1brown videos are amazing! <https://t.co/wXCBy48Tk3>
11. MLOPs. <https://t.co/54XrDUxN2B>
12. Projects. Learn how to Cluster Geolocation Data in Python using DBSCAN and K-Means. <https://t.co/gZID7EgwYy>
13. Datasets. 5,037 machine learning datasets. <https://t.co/lv0A3RsO6a>
14. Methods. 1,783 machine learning components. <https://t.co/leMq3EHn68>
15. Libraries and Models. 944 machine learning models. <https://t.co/W0ex0qBGRz>
16. Research Papers: <https://t.co/LlebMYBeck>
17. 60 days of Data Science and Machine Learning Series — Day 1. Deep dive into Important topics, code, resources and implementations. <https://t.co/izJ5kxVy3t>
18. Day 15–60 days of Data Science and Machine Learning. <https://t.co/b8hMgb8iod>
19. Pytorch tutorial with Examples. <https://t.co/iATDpCTXte>
20. Tensorboard. <https://t.co/qz051IXbye>
21. @Google Machine Learning Crash Course. <https://t.co/arj28SeJmu>
22. Kaggle <https://t.co/zR97g9CgXd>

## IIT Bombay Invites Applications from Students for Free Online Course on Quantum Mechanics

<https://www.dqindia.com/iit-bombay-invites-applications-from-students-for-free-online-course-on-quantum-mechanics/>

## ViT

🔥 #Transformers: from #NLP to #CV

This is a practical introduction to #Transformers and related #deepneuralnetworks architectures from #NaturalLanguageProcessing to #ComputerVision with #sourcecode #collab

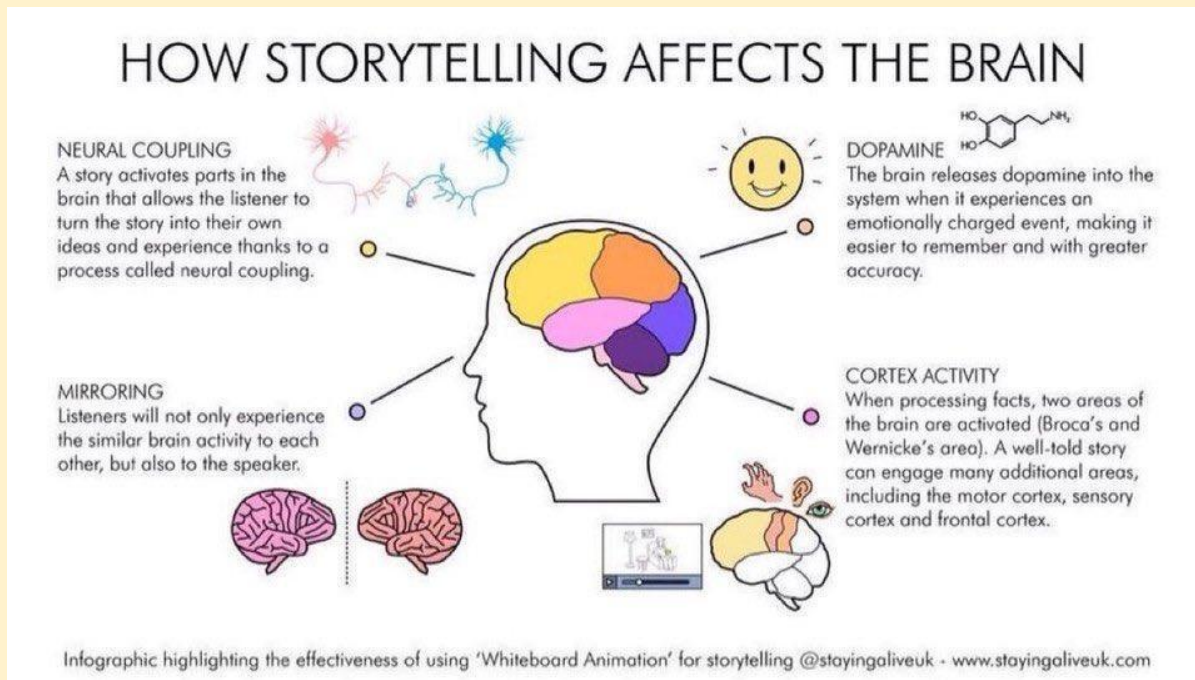
💡 Introduction



- 💡 ViT: Transformers for Computer Vision
- 💡 Visualizing the attention
- 💡 MLP-Mixer
- 💡 Hybrid MLP-Mixer + ViT
- 💡 ConvMixer
- 💡 Hybrid ConvMixer + MLP-Mixer

<https://ibrahimsobh.github.io/Transformers/>

## How Storytelling Affects the Brain



## Navigating the Dozens of Different Strategy Options

<https://hbr.org/2015/06/navigating-the-dozens-of-different-strategy-options>

## Pytorch

[https://pytorch.org/tutorials/recipes/recipes/saving\\_and\\_loading\\_models\\_for\\_inference.html](https://pytorch.org/tutorials/recipes/recipes/saving_and_loading_models_for_inference.html)

## GNN

\*Neural Networks for Data Science\* lecture 6 is out!

It's finally time to talk about graph neural networks. 🎉

Covered a number of basic ideas, from graph convolutions to message-passing, edge predictions, use cases.

Material is here: <https://t.co/TMMdmlqf46>

The slides: <https://t.co/KTwkxQ10Dp>

## AutoML

automating data collection and experiment design; automating data cleanup and missing data imputation; automating feature selection and transformation; automating model discovery, criticism, and explanation;

automating the allocation of computational resources; automating hyperparameter optimization; automating inference; and automating model monitoring and anomaly detection.

## Deep Graph Neural Networks

State of the art in deep graph neural networks - article

1. <https://t.co/e7y2GNoYOT>

## Meta's Neural Prophet

Meta's Neural Prophet - a neural twist on Prophet for time-series

Site: <https://t.co/oTPGbdyH2n>

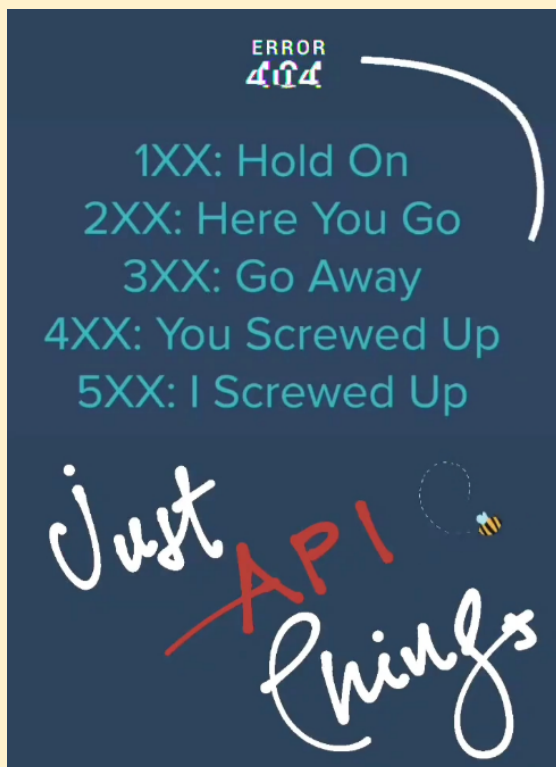
Python GitHub: <https://t.co/TTpQdeOFNT>

Article: <https://t.co/pGpYmbAtxu>

Video: <https://t.co/ExrZM6BQSP>

Comparison with ARIMA: <https://t.co/jQzcjYzbJn>

## Summary of HTTP Errors





Parul Pandey · 1st

Data Science @H2O.ai | Kaggle...

3d · 🌐

Machine Learning Contests(<https://mlcontents.com/>) is a data science competition aggregator site that lists ongoing machine learning competitions/data science contests across **#Kaggle**, **#DrivenData**, **#AICrowd**, and others.

It now also includes a dedicated page that compares cloud GPU prices across many providers, along with functionality. Super helpful for the **#deeplearning** enthusiasts on a budget.

### Cloud GPU Comparison

Find the right cloud GPU provider for your workflow.

Toggle column: 1080Ti - K80 - V100 - A100 - A6000 - P100 - T4 - P4 - 2080 - 3090

| Name                        | Credits | 1080Ti/h | K80/h  | 3090/h | Regions     | Images | Notebooks | SSH | Persistence |
|-----------------------------|---------|----------|--------|--------|-------------|--------|-----------|-----|-------------|
| Vast.ai                     | \$0     | \$0.40   | \$0.60 | \$0.80 | Worldwide   | ✓      | ✓         | ✓   | ✗           |
| LeaderGPU                   | \$0     | \$0.30   | -      | \$2.40 | Netherlands | ✗      | ✗         | ✓   | ✓           |
| Lambda Cloud                | \$0     | -        | -      | -      | US          | ✓      | ✓         | ✓   | ✗           |
| DataCrunch.io               | \$0     | -        | -      | -      | Finland     | ✓      | ✓         | ✓   | ✓           |
| Jarvis Labs                 | \$0     | -        | -      | -      | India       | ✓      | ✓         | ✓   | ✓           |
| Paperspace                  | \$10    | -        | -      | -      | US/Europe   | ✓      | ✓         | ✓   | ✓           |
| Genesis Cloud               | \$50    | \$0.30   | -      | \$0.85 | Iceland     | ✓      | ✓         | ✓   | ✓           |
| Microsoft Azure             | \$200   | -        | \$0.90 | -      | Worldwide   | ✓      | ✓         | ✓   | ✓           |
| Amazon Web Services (AWS)   | \$300   | -        | \$0.90 | -      | Worldwide   | ✓      | ✓         | ✓   | ✓           |
| Google Cloud Platform (GCP) | \$350   | -        | \$0.45 | -      | Worldwide   | ✓      | ✓         | ✓   | ✓           |

### What is R used for?

<https://www.codecademy.com/resources/blog/what-is-r-used-for/>

### Intermittent Demand Forecasting with Deep Renewal Processes – Deep & Shallow

<https://deep-and-shallow.com/2020/10/13/intermittent-demand-forecasting-with-deep-renewal-processes/>

## Github: GNN Challenge 2021

GNN Graph Neural Networks Challenge 2021

Solution: <https://github.com/ITU-AI-ML-in-5G-Challenge/ITU-ML5G-PS-001-PARANA>

Video: <https://m.youtube.com/watch?v=RmiLVl8yBZs>

### What's New at AWS re:Invent?

[https://aws.amazon.com/new/?whats-new-content-all.sort-by=item.additionalFields.postDateTime&whats-new-content-all.sort-order=desc&awsf.whats-new-analytics=\\*all&awsf.whats-new-app-integration=\\*all&awsf.whats-new-arvr=\\*all&awsf.whats-new-blockchain=\\*all&awsf.whats-new-business-applications=\\*all&awsf.whats-new-cloud-financial-management=\\*all&awsf.whats-new-compute=\\*all&awsf.whats-new-containers=\\*all&awsf.whats-new-customer-enablement=\\*all&awsf.whats-new-customer%20engagement=\\*all&awsf.whats-new-database=\\*all&awsf.whats-new-developer-tools=\\*all&awsf.whats-new-end-user-computing=\\*all&awsf.whats-new-mobile=\\*all&awsf.whats-new-gametech=\\*all&awsf.whats-new-iot=\\*all&awsf.whats-new-machine-learning=\\*all&awsf.whats-new-management-governance=\\*all&awsf.whats-new-media-services=\\*all&awsf.whats-new-migration-transfer=\\*all&awsf.whats-new-networking-content-delivery=\\*all&awsf.whats-new-quantum-tech=\\*all&awsf.whats-new-robotics=\\*all&awsf.whats-new-satellite=\\*all&awsf.whats-new-security-id-compliance=\\*all&awsf.whats-new-serverless=\\*all&awsf.whats-new-storage=\\*all](https://aws.amazon.com/new/?whats-new-content-all.sort-by=item.additionalFields.postDateTime&whats-new-content-all.sort-order=desc&awsf.whats-new-analytics=*all&awsf.whats-new-app-integration=*all&awsf.whats-new-arvr=*all&awsf.whats-new-blockchain=*all&awsf.whats-new-business-applications=*all&awsf.whats-new-cloud-financial-management=*all&awsf.whats-new-compute=*all&awsf.whats-new-containers=*all&awsf.whats-new-customer-enablement=*all&awsf.whats-new-customer%20engagement=*all&awsf.whats-new-database=*all&awsf.whats-new-developer-tools=*all&awsf.whats-new-end-user-computing=*all&awsf.whats-new-mobile=*all&awsf.whats-new-gametech=*all&awsf.whats-new-iot=*all&awsf.whats-new-machine-learning=*all&awsf.whats-new-management-governance=*all&awsf.whats-new-media-services=*all&awsf.whats-new-migration-transfer=*all&awsf.whats-new-networking-content-delivery=*all&awsf.whats-new-quantum-tech=*all&awsf.whats-new-robotics=*all&awsf.whats-new-satellite=*all&awsf.whats-new-security-id-compliance=*all&awsf.whats-new-serverless=*all&awsf.whats-new-storage=*all)

# Holographic Principles



Vipul Patel • Following

Chief Data Scientist at SAP | Practitioner, Evangelis...

2h • 🌐

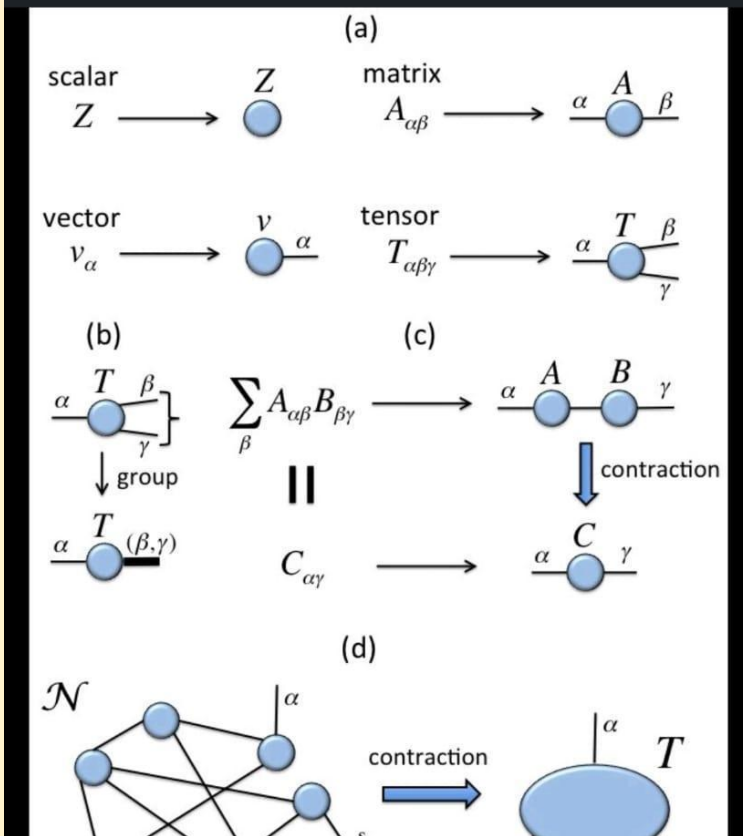
The Holographic Principle: Why Deep Learning Works

Other Popular Posts:

❤️ The Ultimate Guide to AI, Data Science & Machine Learning (free download) : <https://lnkd.in/dfnAkrSq>

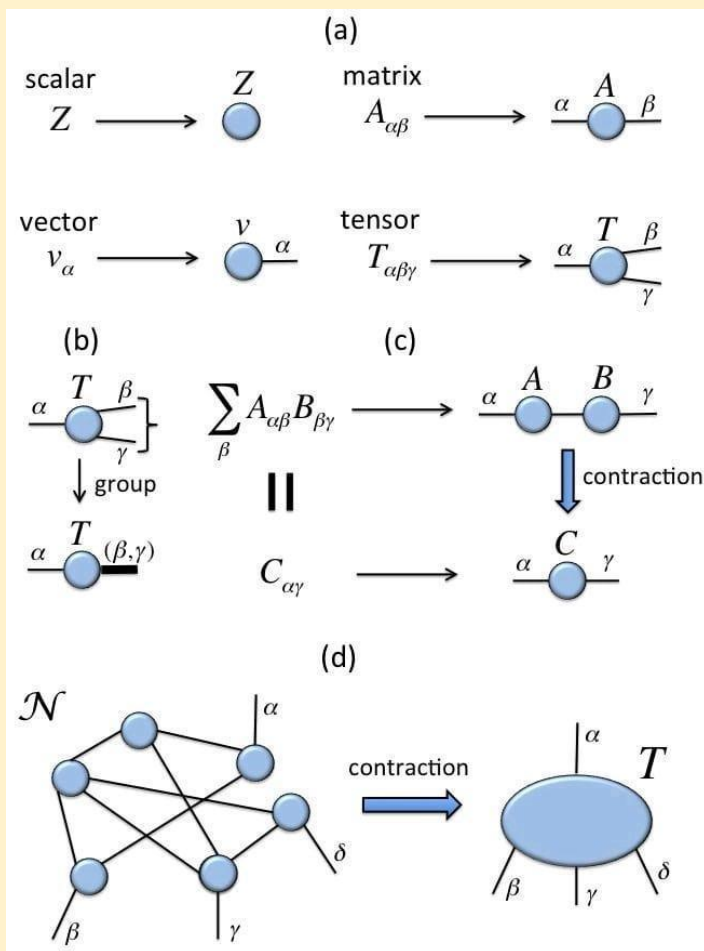
#data #businessintelligence #analytics  
#datascience #machinelearning  
#artificialintelligence #deeplearning

✓ If You Like the posts You can Follow Me here:  
LinkedIn <https://lnkd.in/dm3Mmk8>



# GNN Notation





## AI Consulting: In-depth Guide with Top AI Consulting Firms

<https://research.aimultiple.com/ai-consulting/>

What are the typical AI consulting activities?

### 1- Strategy formulation

- Understanding your company's current status: Using solutions like process mining and analyzing the company's strategy to understand the company's current status.
- Creating a portfolio of potential AI initiatives: This step involves identifying pain points and understanding how AI can contribute to the business.
- Predicting the value of a project: AI consulting firms help businesses predict the value of the project so that businesses don't invest more than the expected outcome.
- Choosing AI solutions and data sets to feed machine learning models: AI consultants should be aware of the abilities and limitations of each technology in the space. According to business needs, consultants choose the right AI solution and required data training sets for implementation. If the organization does not have the required data for AI to work properly, consultants can help find or label data with their extensive market knowledge. To acquire relevant data, businesses can rely on data collectors such as Bright Data. Bright Data's data collector extracts real-time public data from any marketplace or eCommerce platform and delivers it to businesses on autopilot in the designated format.
- Launching pilot projects: Consultants help launch initial small-scale projects to prove the viability of the idea.
- Identification of scaling challenges and coming up with a solution: After analyzing the results of pilot projects, artificial intelligence consultants provide businesses with a roadmap regarding scaling goals.

2- Commercial due diligence: companies are investing in AI-specific due diligence capabilities. These include understanding and evaluating data sources relevant for AI models, formulating fast approaches to benchmarking different AI vendors' solutions and getting familiar with AI-related academic and work credentials.

### 3- Implementation

The strategy will result in a number of initiatives. Implementation should also be considered as multiple activities such as planning, vendor selection if needed, project management, development, improvement of business processes impacted by the project, change management and so on.

### 4- Training

Ideally, consulting projects should improve the culture and skills of the client. This is especially relevant in the field of AI where talent is scarce. AI consulting projects need to ensure that client teams are capable and knowledgeable about the technologies they will be working on.

## **4 Keys to Effective—and Honest—Data Visualizations**

<https://insight.kellogg.northwestern.edu/article/data-visualization-honesty-infographics>

# SUPPORT VECTOR MACHINES



## What is SVM?

Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for both classification or regression. However, it is mostly used in classification problems.

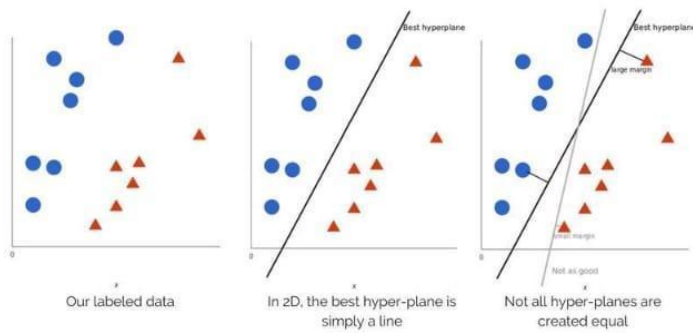
In this algorithm, we plot each data item as a point in n-dimensional space (where n is the number of features) with the value of each feature being the value of a particular coordinate.

## How is the data classified?

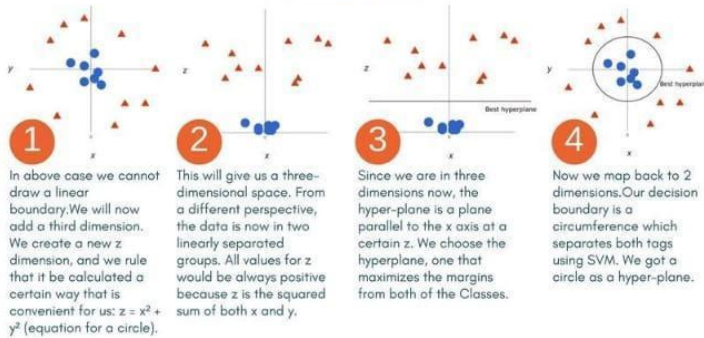
We perform classification by finding the hyperplane that differentiates the two classes very well. In other words the algorithm outputs an optimal hyperplane which categorizes new examples.

## What is an optimal Hyper-Plane?

For SVM, it's the one that maximizes the margins from both tags. In other words: the hyperplane whose distance to the nearest element of each tag is the largest.



## Nonlinear data



1 In above case we cannot draw a linear boundary. We will now add a third dimension. We create a new z dimension, and we rule that it be calculated a certain way that is convenient for us:  $z = x^2 + y^2$  (equation for a circle).

2 This will give us a three-dimensional space. From a different perspective, the data is now in two linearly separated groups. All values for z would be always positive because z is the squared sum of both x and y.

3 Since we are in three dimensions now, the hyper-plane is a plane parallel to the x axis at a certain z. We choose the hyperplane, one that maximizes the margins from both of the Classes.

4 Now we map back to 2 dimensions. Our decision boundary is a circumference which separates both tags using SVM. We got a circle as a hyper-plane.

# TUNING PARAMETERS

## KERNEL

The learning of the hyperplane in linear SVM is done by transforming the problem using some linear algebra. This is where the kernel plays role. Polynomial and exponential kernels calculates separation line in higher dimension. This is called kernel trick.

## GAMMA

The gamma parameter defines how far the influence of a single training set reaches. With low gamma, points far away from the possible separation line are considered in calculation for the separation line. Where as high gamma means the points close to possible line are considered in calculation.

## REGULARIZATION

For large values of this parameter, the optimization will choose a smaller-margin hyperplane if that hyperplane does a better job of getting all the training points classified correctly. Conversely, a very small value of it will cause the optimizer to look for a larger-margin separating hyperplane, even if that hyperplane misclassifies more points.

## MARGIN

A margin is a separation of line to the closest class points. A good margin is one where this separation is larger for both the classes. A good margin allows the points to be in their respective classes without crossing to other class.



## Top Machine Learning Terms for Managers

### Data

Data is the integral part of ML

- **Big Data** - Large volumes of structured & unstructured data
- **Training Data** - used to train model
- **Validation Data** - used to evaluate and fine tune the model
- **Test Data** - used to test the final model



### Feature

Features are variables a model will use to build a model

- **Feature Extraction** - process of figuring out what variables the model will use.
- **Feature Selection** - process where you automatically or manually select features
- **Feature Engineering** - process of using domain knowledge to extract features



### Overfitting Underfitting

- When a model learns the information and noise in the training data to the point that it degrades the model's output on new data, this is known as **overfitting**.
- **Underfitting** is defined as a model that cannot both model and generalise to new data.



### Supervised Learning

It is a form of ML in which machines are taught using "pre-labeled training data and then predict the performance using that data.

- **Classification** - predict a category or class
- **Regression** - predict a number

\*input data has already been tagged with the appropriate output.



### Unsupervised Learning

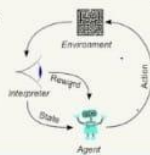
It is a ML technique in which models are trained without being supervised. On the other hand, these models find hidden trends & insights in the data they are given.

- **Clustering** - Segregate data by similarity
- **Dimension Reduction** - Identify hidden dependencies & simplify data
- **Association** - Find interesting relations or sequence



### Reinforcement Learning

It is an ML technique that allows an agent to learn by trial and error in an interactive environment using feedback from its own actions and experiences. It is similar to training a dog, where a dog gets its reward if it executes certain tasks correctly & punishment if it does not



### Natural Language Processing

NLP is a ML technique used to read, decipher, understand, and make sense of the human languages in a manner that is valuable

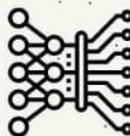
- **Syntax** refers to the arrangement of words in a sentence such that they make grammatical sense
- **Semantics** refers to the meaning that is conveyed by a text



### Deep Learning

It is subset of ML that is capable of learning unsupervised from data that is unstructured via networks similar to human brain called artificial **neural networks**

- **Computer Vision** is the analysis and high-level understanding of image and video data
- **Image classification** is the task of teaching a model to identify what is on a given image
- **Object recognition** is the task of identifying and locating objects in images and videos
- **Image segmentation** is the task of training a model to assign a class to each pixel



## How to Talk to Your Data Scientist \ HuffPost Impact

[https://www.huffpost.com/entry/how-to-talk-to-your-data\\_b\\_13464338](https://www.huffpost.com/entry/how-to-talk-to-your-data_b_13464338)

## Simplex Method with NumPy and Matrix Operations

<https://towardsdatascience.com/developing-the-simplex-method-with-numpy-and-matrix-operations-16321fd82c85>

<https://explain-that.blogspot.com/2011/06/logic-of-how-simplex-method-works.html>

[http://www.phpsimplex.com/en/simplex\\_method\\_example.htm](http://www.phpsimplex.com/en/simplex_method_example.htm)

## Optimization problem

- An optimization problem has three basic components: (1) an objective function, which we either want to maximize or minimize; (2) a set of variables which control the objective function; (3) a set of constraints that allows the variables to take certain values while excluding others.
- It then follows that the optimization problem is to find the set of values for the variables that maximize or minimize the objective function whilst satisfying the set of constraints.
- Linear Programming — a.k.a Linear Optimization is a technique to find the best outcome in a mathematical model where the objective function and the constraints are represented by linear relationships.
- Decision variables: what we'd like to determine. The unknowns.
- Objective function: the linear function representing the quantities to be maximized or minimized.
- Constraints: the system of equalities or inequalities representing the restrictions on the decision variables.
- Non-negativity restrictions: the values of the decision variables should be greater than or equal to zero.
- <https://towardsdatascience.com/linear-programming-for-data-scientists-e9cf1ddf1331>

## Difference between Soft Computing and Hard Computing

<https://www.geeksforgeeks.org/difference-between-soft-computing-and-hard-computing/?ref=lbp>

## A Global Breakdown of Greenhouse Gas Emissions by Sector

<https://www.visualcapitalist.com/cp/a-global-breakdown-of-greenhouse-gas-emissions-by-sector/>

## Turing Bletchley: A Universal Image Language Representation model by Microsoft - Microsoft Research

<https://www.microsoft.com/en-us/research/blog/turing-bletchley-a-universal-image-language-representation-model-by-microsoft/>



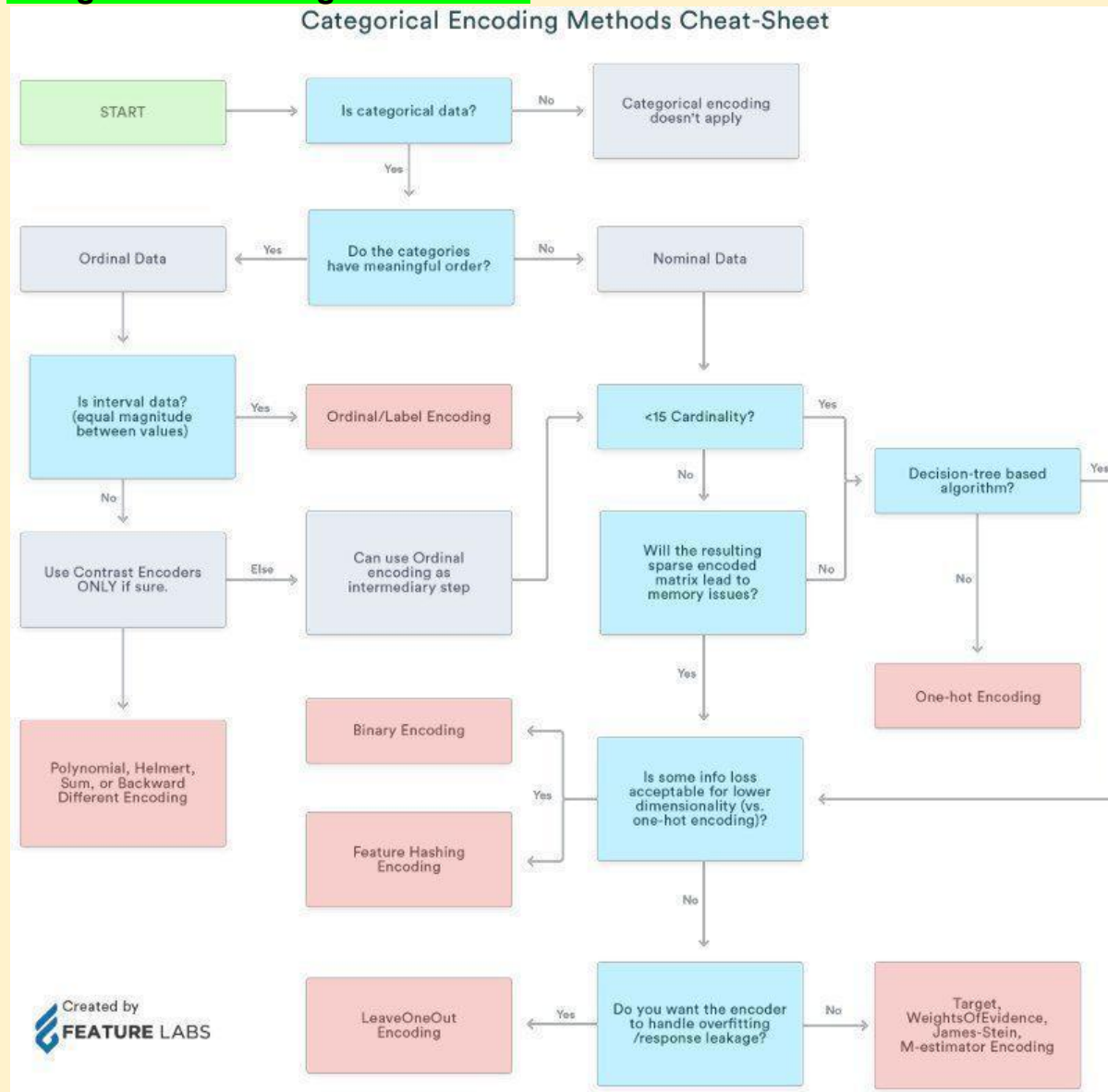
## Timeseries Library

|                                | Merlion | Alibi Detect | Kats | statsmodels | GluonTS | RRCF | STUMPY | Greykite | F |
|--------------------------------|---------|--------------|------|-------------|---------|------|--------|----------|---|
| Univariate Forecasting         | ✓       |              | ✓    | ✓           | ✓       |      |        | ✓        |   |
| Multivariate Forecasting       | ✓       |              | ✓    | ✓           | ✓       |      |        |          |   |
| Univariate Anomaly Detection   | ✓       | ✓            | ✓    |             |         | ✓    | ✓      | ✓        |   |
| Multivariate Anomaly Detection | ✓       | ✓            | ✓    |             |         | ✓    | ✓      |          |   |
| AutoML                         | ✓       |              | ✓    |             |         |      |        | ✓        |   |
| Ensembles                      | ✓       |              |      |             |         | ✓    |        |          |   |
| Benchmarking                   | ✓       |              |      |             | ✓       |      |        |          |   |
| Visualization                  | ✓       |              | ✓    |             |         |      |        | ✓        |   |

## 10 Mysterious And Rare Artifacts You Didn't Know Before - TopTenScience

<https://toptenscience.com/mysterious-and-rare-artifacts/>

## Categorical Encoding Cheat Sheet



## Quantum Computing

<https://www.fi.muni.cz/usr/gruska/quantum/overview.html>

## NeurIPS 2021—12 papers You Shouldn't miss

2334 papers, 60 workshops, 8 keynote speakers, 15k+ attendees. A dense landscape that's hard to navigate without a good guide and map, so here are some of our ideas!

1. Learning to Draw: Emergent Communication through Sketching \ NeurIPS 2021 Link: <https://youtu.be/mH9v-3AHmSQ>
2. Fixes That Fail: Self-Defeating Improvements in Machine-Learning Systems \ NeurIPS 2021 Link: [https://youtu.be/oP\\_peT1cvFw](https://youtu.be/oP_peT1cvFw)
3. Partition and Code: learning how to compress graphs \ NeurIPS 2021. Link: <https://youtu.be/Ed7xvAMGQ0A>
4. FLEX: Unifying Evaluation for Few-Shot NLP \ NeurIPS 2021. Link: <https://youtu.be/2CeUNW8IIZo>
5. Efficient Training of Retrieval Models using Negative Cache \ NeurIPS 2021 Link: <https://youtu.be/FncbQ2HGoEA>
6. Transformers for Multimodal Self Supervised Learning from Raw Video, Audio and Text \ NeurIPS 2021. Link: [https://youtu.be/V3gY\\_hyATU8](https://youtu.be/V3gY_hyATU8)
7. Robust Predictable Control \ NeurIPS 2021 Link: <https://youtu.be/6efUoVvhmpQ>
8. Multimodal Few-Shot Learning with Frozen Language Models 🌐 NeurIPS 2021 Link: <https://youtu.be/Il6GwutGpEM>

9. A 3D Generative Model for Structure-Based Drug Design \ NeurIPS 2021 Link: <https://youtu.be/vIBOfANRWRU>
10. Framing RNN as a kernel method: A neural ODE approach \ Oral Paper \ NeurIPS 2021 Link: [https://youtu.be/2\\_MF2LX9Q5E](https://youtu.be/2_MF2LX9Q5E)
11. MAUVE: Measuring Gap Between Neural and Human Text using Divergence Frontiers \ Oral \ NeurIPS 2021. Link: [https://youtu.be/sp\\_OjYUFKXo](https://youtu.be/sp_OjYUFKXo)
12. E(n) Equivariant Normalizing Flows \ Oral Paper \ NeurIPS 2021. Link: <https://youtu.be/RfzcyH7N7kw>

#MachineLearning #MontrealAI #ComputerVision #AI #neuralnetworks #NeurIPS2021 #artificialintelligence #deeplearning #ReinforcementLearning

## Architecture of Modules in Merlion

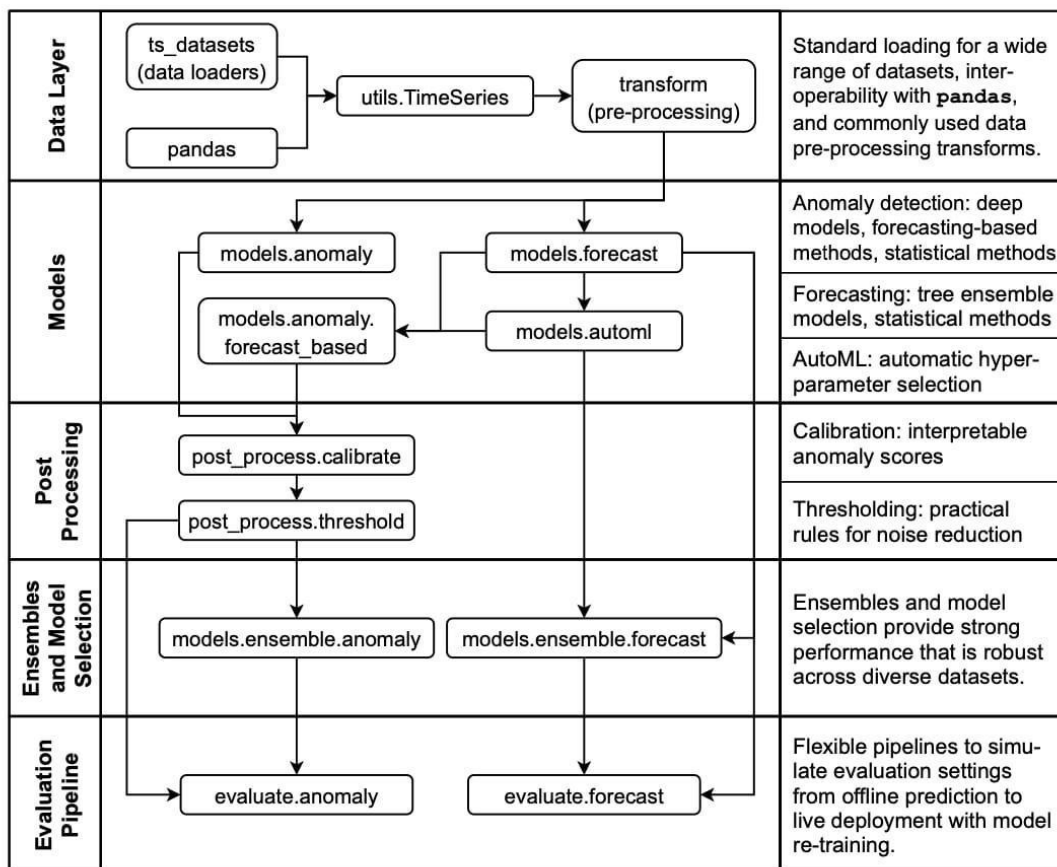


Figure 1: Architecture of modules in Merlion.

## poetry python dependency and package manager

```
example.sh

$ poetry add pandas 'numpy<1.18'
"""
SolverProblemError
Because pandas (1.4.1) depends on
numpy (≥1.18.5) and no versions
of pandas match >1.4.1,<2.0.0,
pandas (≥1.4.1,<2.0.0) requires
numpy (≥1.18.5).

So, because chapter6 depends on
both pandas (^1.4.1) and numpy
(<1.18), version solving failed.
"""

$ poetry remove pandas
"""
• Removing pandas (1.4.1)
• Removing python-dateutil (2.8.2)
• Removing pytz (2021.3)
• Removing six (1.16.0)
"""
```

Khuyen Tran

 khuyentran1401

## The Technique of English Word Syllable Division in Speech ...

[hindawi.com/journals/mpe/2021/4270035/](https://hindawi.com/journals/mpe/2021/4270035/)

## Tutorial on reading large datasets \ Kaggle

<https://www.kaggle.com/rohanrao/tutorial-on-reading-large-datasets>

## NVIDIA NLP in Production

Do you want to learn about state-of-the-art techniques for Search and NLP in production (FOR FREE, on-demand 🙄). Check the sessions below at [#gtc22](#) from March 21st - March 24th, register and search for session code.

Registration Link: <https://lnkd.in/ethTGJrk>

[#deeplearning](#) [#machinelearning](#) [#ai](#) [#ecommerce](#)  
[#nlp](#)

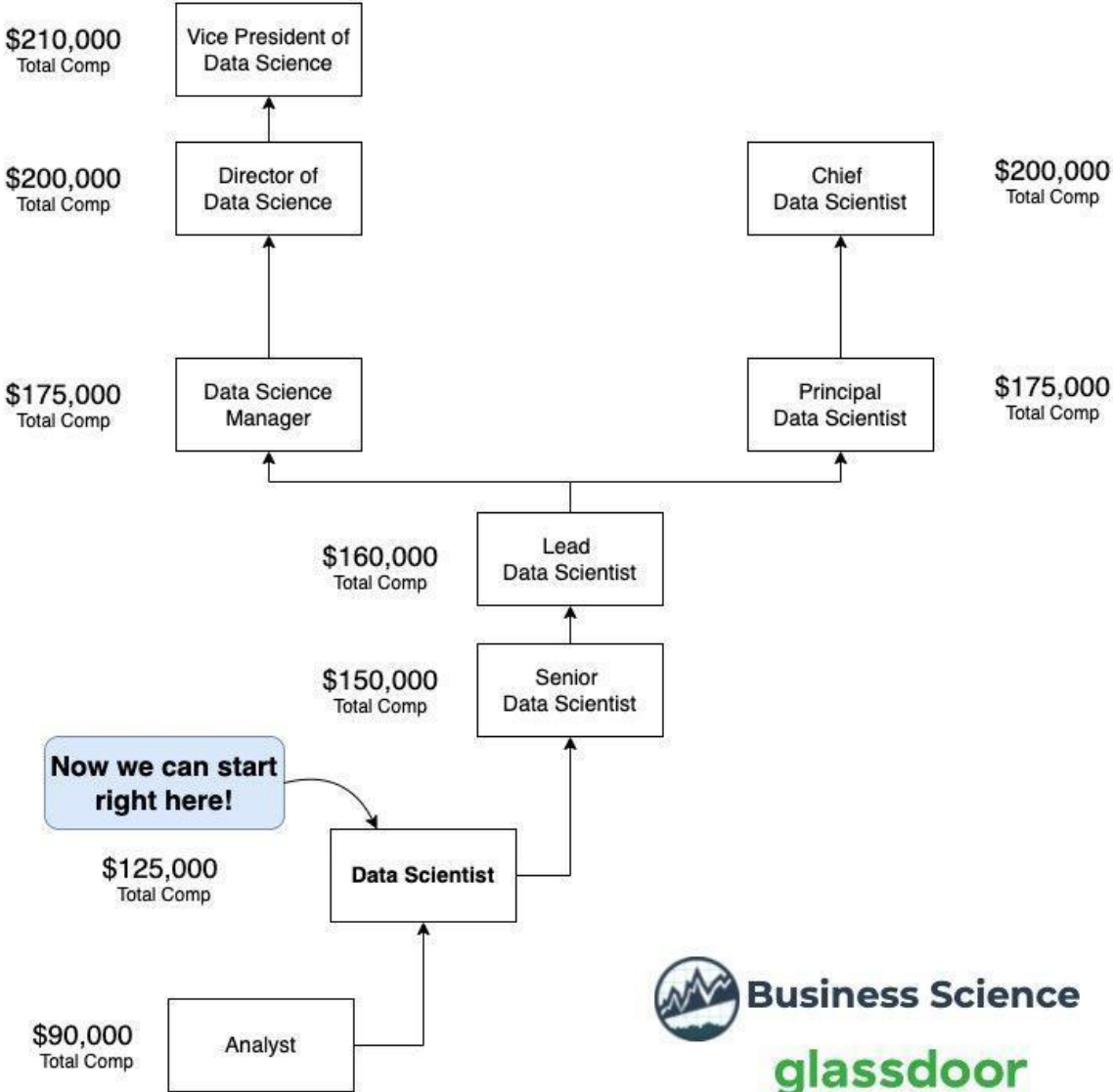
### NVIDIA NLP and Search Talks

|                                                                                                                                            |          |
|--------------------------------------------------------------------------------------------------------------------------------------------|----------|
| <b>Natural Language Understanding in Practice: Lessons Learned from Successful Enterprise Deployments</b>                                  | S41769   |
| <b>Big NLP Demystified: Business Impact of Large Language Models</b>                                                                       | S42016   |
| <b>How Transformers and ASR Change the Way of Search</b>                                                                                   | S41212   |
| <b>Conversational AI for the Next 500 million?</b>                                                                                         | S41564   |
| <b>NLP Beyond English: Training Extreme-scale Language Models with Megatron for French, and More</b>                                       | S41610   |
| <b>One Large GPT Model to Rule Them All: Building Single GPT-3 Language Model for Nordic Languages</b>                                     | S41711   |
| <b>Conversational AI: Build Applications with Riva and Train Multibillion Parameter Language Models using Nemo-Megatron</b>                | CWE41824 |
| <b>Conversational AI Demystified</b>                                                                                                       | S41956   |
| <b>Using NeMo-Megatron to Build Large-scale, Localized Language Models: From Data Preparation to Training and Deployment to Production</b> | S42018   |
| <b>When Neural Machine Translation Encounters Pre-trained Models</b>                                                                       | S42153   |

## Glassdoor Data Science Career Path

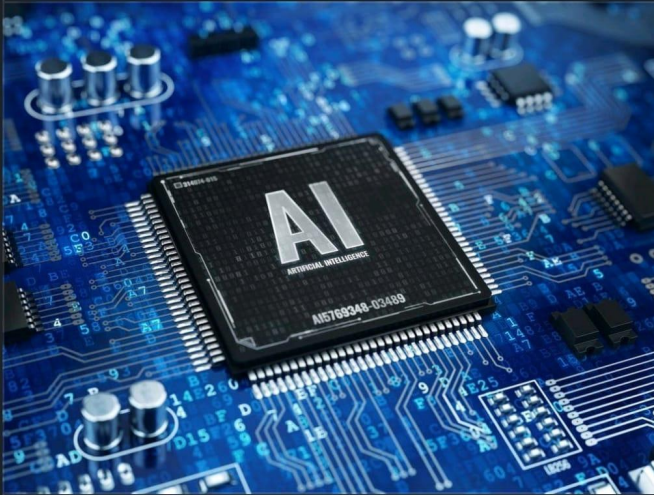


# The Data Science Career Path



## Google Using Deep Learning to Design Chips

Google uses deep learning to design faster, smaller AI chips



Google uses deep learning to design faster, smaller AI chips

[theregister.com](http://theregister.com) • 3 min read





**Giannis Tolios** · Following

Data Scientist | Book Author at Leanpub...

9h · Edited · 🌐

Machine learning competitions are an excellent way to practice your skills by solving challenging problems, and even earn some money! Kaggle is the leading platform for such competitions, but there are many others, including DrivenData and Alcrowd. Furthermore, ML Contests is a website that provides up-to-date information for ongoing competitions, making it a valuable resource for data scientists. You can visit the link below to check the latest information, and share your thoughts in the comments!

Machine Learning Contests website:

<https://mlcontests.com/>

**#datascience #machinelearning #deeplearning  
#python #kaggle #linkedin**

| Title                                                     | Type                    | Deadline    | Prize Pool | Platform   |
|-----------------------------------------------------------|-------------------------|-------------|------------|------------|
| Predict Air Quality from Satellite Images and Sensor Data | Computer Vision         | 21 Mar 2022 | \$50,000   | DrivenData |
| UltraMNIST: Extract Sum of Digits from Images             | Computer Vision         | 10 Apr 2022 | \$5,500    | Kaggle     |
| Predict Investment Returns                                | Time Series Forecasting | 11 Apr 2022 | \$100,000  | Kaggle     |
| AI Progress Essay Contest                                 | Writing                 | 16 Apr 2022 | \$6,500    | Metaculus  |
| Analyse Martian Mass Spectrometry Data                    | Supervised Learning     | 18 Apr 2022 | \$30,000   | DrivenData |
| Happywhale -                                              | Computer Vision         | 18 Apr 2022 | \$25,000   | Kaggle     |



Parul Pandey · 1st

Data Science @H2O.ai | Kaggle...

3d · 🌐

Machine Learning Contests(<https://mlcontests.com/>) is a data science competition aggregator site that lists ongoing machine learning competitions/data science contests across [#Kaggle](#), [#DrivenData](#), [#AICrowd](#), and others.

It now also includes a dedicated page that compares cloud GPU prices across many providers, along with functionality. Super helpful for the [#deeplearning](#) enthusiasts on a budget.

### Cloud GPU Comparison

Find the right cloud GPU provider for your workflow.

Toggle column: 1080Ti - K80 - V100 - A100 - A6000 - P100 - T4 - P4 - 2080 - 3090

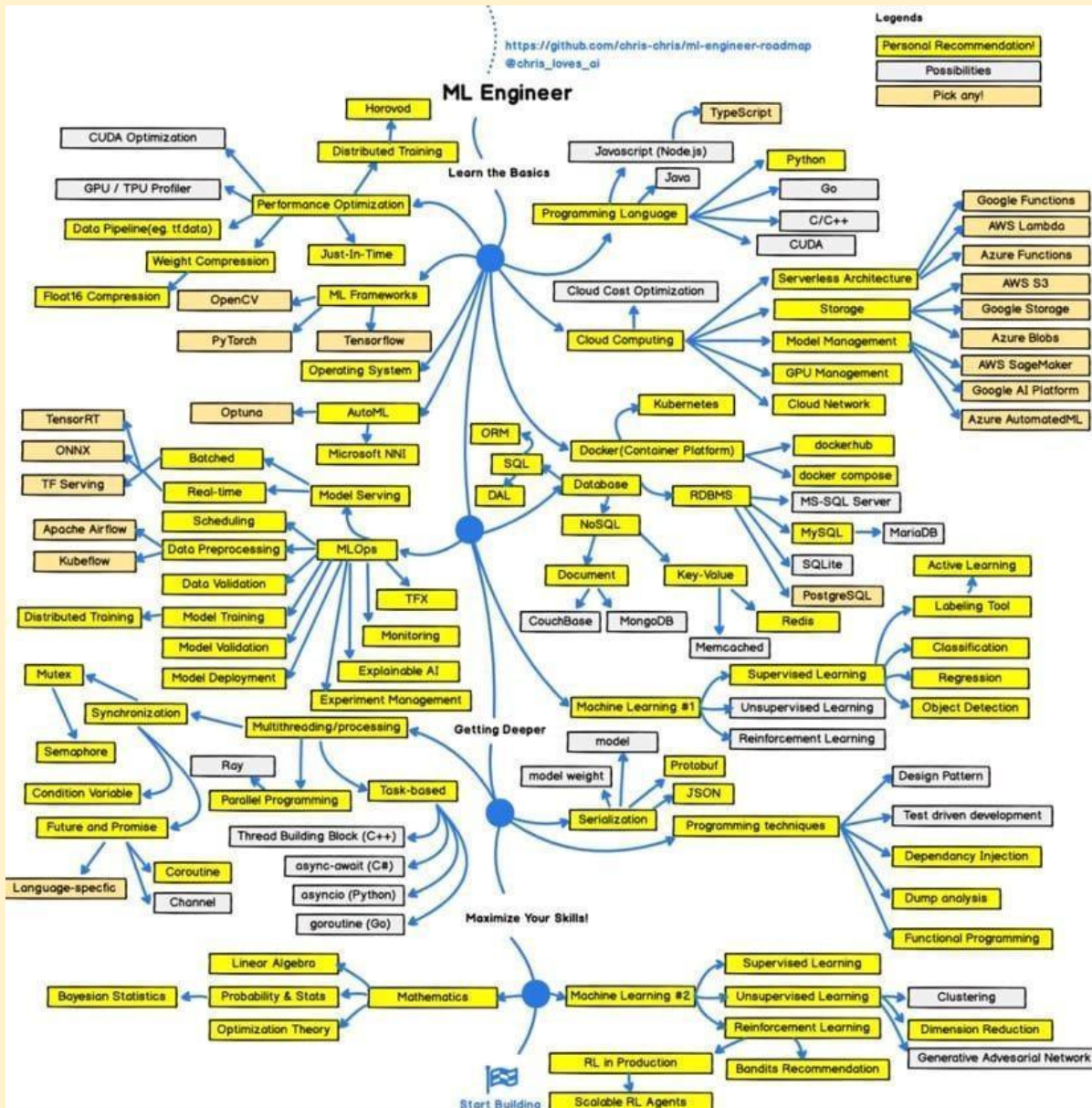
| Name                        | Credits | 1080Ti/h | K80/h  | 3090/h | Regions     | Images | Notebooks | SSH | Persistence |
|-----------------------------|---------|----------|--------|--------|-------------|--------|-----------|-----|-------------|
| Vast.ai                     | \$0     | \$0.40   | \$0.60 | \$0.80 | Worldwide   | ✓      | ✓         | ✓   | ✗           |
| LeaderGPU                   | \$0     | \$0.30   | -      | \$2.40 | Netherlands | ✗      | ✗         | ✓   | ✓           |
| Lambda Cloud                | \$0     | -        | -      | -      | US          | ✓      | ✓         | ✓   | ✗           |
| DataCrunch.io               | \$0     | -        | -      | -      | Finland     | ✓      | ✓         | ✓   | ✓           |
| Jarvis Labs                 | \$0     | -        | -      | -      | India       | ✓      | ✓         | ✓   | ✓           |
| Paperspace                  | \$10    | -        | -      | -      | US/Europe   | ✓      | ✓         | ✓   | ✓           |
| Genesis Cloud               | \$50    | \$0.30   | -      | \$0.85 | Iceland     | ✓      | ✓         | ✓   | ✓           |
| Microsoft Azure             | \$200   | -        | \$0.90 | -      | Worldwide   | ✓      | ✓         | ✓   | ✓           |
| Amazon Web Services (AWS)   | \$300   | -        | \$0.90 | -      | Worldwide   | ✓      | ✓         | ✓   | ✓           |
| Google Cloud Platform (GCP) | \$350   | -        | \$0.45 | -      | Worldwide   | ✓      | ✓         | ✓   | ✓           |

### 280+ Machine Learning Projects with Python

283 Machine Learning Projects Solved & Explained using Python programming language.

<https://medium.com/coders-camp/230-machine-learning-projects-with-python-5d0c7abf8265>





## Machine learning engineer R&R

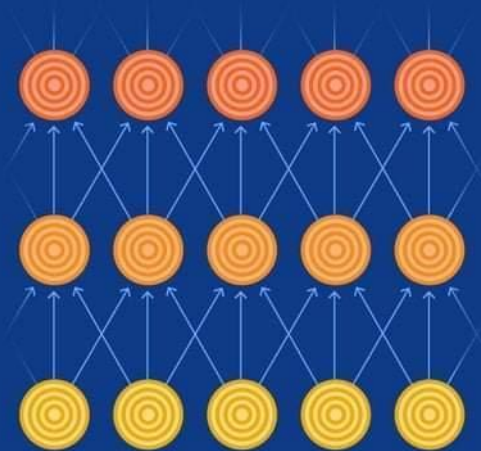
- <https://www.techtarget.com/searchenterpriseai/definition/machine-learning-engineer-ML-engineer>
- Designing ML systems.
- Researching and implementing [ML algorithms](#) and tools.
- Selecting appropriate data sets.
- Picking appropriate data representation methods.
- Identifying differences in data distribution that affects model performance.
- Verifying [data quality](#).
- Transforming and converting data science prototypes.
- Performing [statistical analysis](#).
- Running machine learning tests.
- Using results to improve models.
- Training and retraining systems when needed.
- Extending machine learning libraries.
- Developing machine learning apps according to client requirements.

## A New Way to See

For years, convolutional neural networks (CNNs) have been the dominant method of processing and classifying images. Now, transformers are performing as well as CNNs in many tasks. The two have different approaches to computer vision.

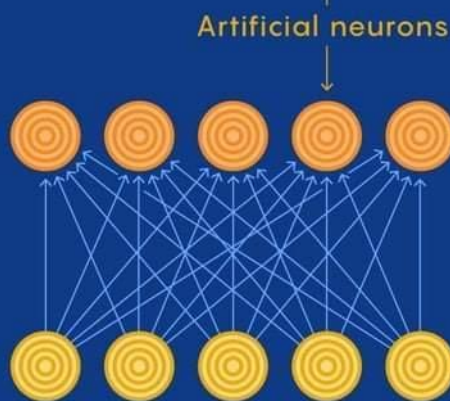
### CNN

A CNN repeatedly filters small portions of an image, using mathematical computations to map features and build up a fuller, more complex picture.



### TRANSFORMERS

A transformer instead starts by connecting every element to every other element, creating a global, if incomplete, representation from the first layer.



## AI, ML Tools



**SBI Innovate for Bank 2022 Hackathon Technology Partner Microsoft \ TechGig**

<https://www.techgig.com/digital/sbi-microsoft>

**Intro to Intel's Distribution of OpenVino Toolkit**

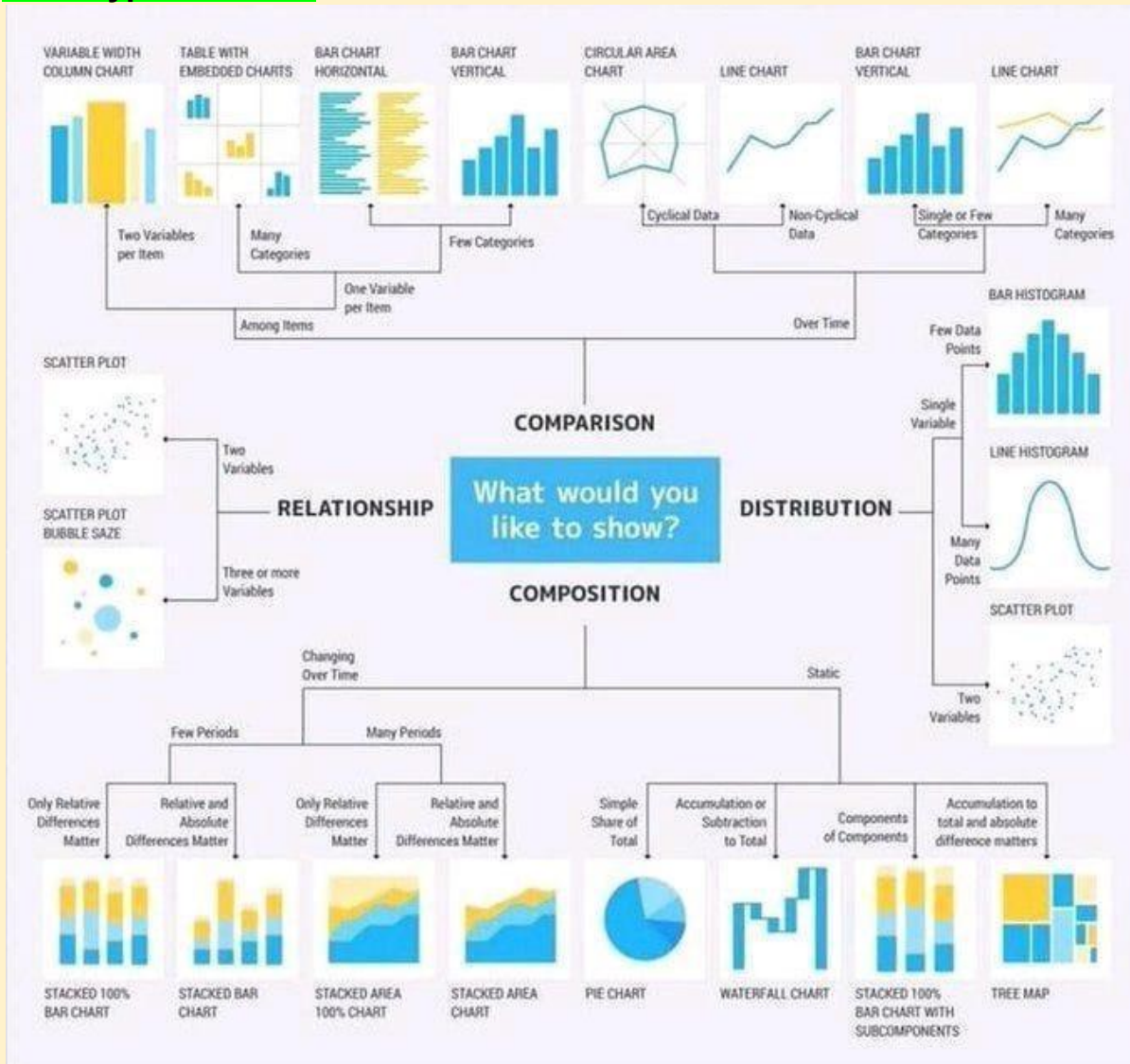
<https://pub.towardsai.net/introduction-to-intel-distribution-of-openvino-toolkit-b1ba5b0cf24f>

**A Practical Strategy to Share the Code among Different Notebooks \ by Angelica Lo Duca \ Towards AI**

<https://pub.towardsai.net/a-practical-strategy-to-share-the-code-among-different-notebooks-9f9bd78aa3a7>



## Chart type selection



## Monitoring NLP Pipelines

Rubrix currently gives users several ways to monitor and observe model predictions.

<https://rubrix.readthedocs.io/en/v0.14.0/guides/monitoring.html#Using-rb.log-in-background-mode>

## The Best NLP with Deep Learning Course is Free

<https://www.kdnuggets.com/2020/05/best-nlp-deep-learning-course-free.html>

<https://nlp.stanford.edu/~manning/>

<https://web.stanford.edu/class/cs224n/>

YouTube: Natural Language Processing with Deep Learning Course

1. [Stanford CS224N: Natural Language Processing with Deep Learning Course \ Winter 2019](https://www.kdnuggets.com/2020/05/best-nlp-deep-learning-course-free.html)

NLP DL Projects Reports

<https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1194/project.html>

Free NLP courses

<https://www.kdnuggets.com/2019/10/10-free-top-notch-courses-natural-language-processing.html>

NLP Best practices and examples

<https://www.kdnuggets.com/2020/05/natural-language-processing-recipes-best-practices-examples.html>

5 NLP Resources

<https://www.kdnuggets.com/2018/02/5-fantastic-practical-natural-language-processing-resources.html>

## **Yellowbrick Python Library for ML**

Yellowbrick Python Library for Machine Learning Visualizations

<https://www.scikit-yb.org/en/latest/>

## **GoogleNews pip**

If you want to get Google news in #Python, use GoogleNews. GoogleNews allows you to get search results for a keyword in a specific time interval.

Link to GoogleNews: <https://t.co/iG3b5UMYix>

## **Spring 2021 Schedule - Full Stack Deep Learning**

<https://fullstackdeeplearning.com/spring2021/>

## **Book Recommender System**

[https://youtu.be/1YoD0fg3\\_EM](https://youtu.be/1YoD0fg3_EM)

## **Airflow**

<https://youtube.com/playlist?list=PLwFJcsJ61oujAqYpMp1kdUBcPG0sE0QMT>

## **What is Apache Airflow \ Introduction to Apache Airflow**

<https://www.analyticsvidhya.com/blog/2020/11/getting-started-with-apache-airflow/>

## **Monitaur Announces The General Availability of 'GovernML' Platform to Bring Responsible AI/ML Systems into Practice - MarkTechPost**

<https://www.marktechpost.com/2022/04/08/monitaur-announces-the-general-availability-of-governml-platform-to-bring-responsible-ai-ml-systems-into-practice/>

## **Data science data repository**

<https://www.kdnuggets.com/2022/04/complete-collection-data-repositories-part-2.html>

## **Mathematics of Big Data and Machine Learning**

Mathematics of Big Data and Machine Learning \ Supplemental Resources \ MIT OpenCourseWare

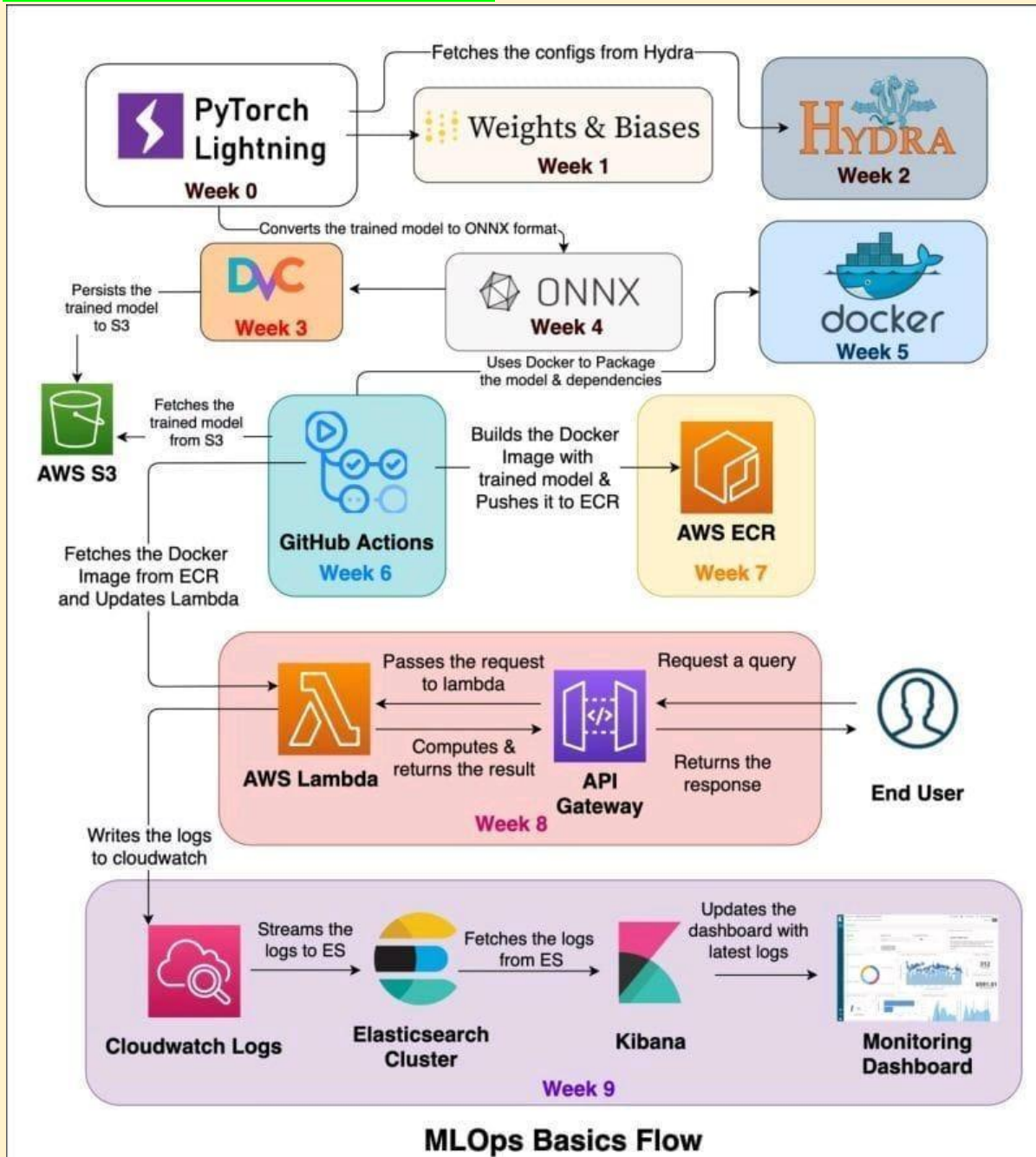
<https://ocw.mit.edu/courses/res-ll-005-mathematics-of-big-data-and-machine-learning-january-iap-2020/>



## Data quality attributes

| Attribute    | What it means                                                                                         | Example of good practice                                                                                                                         | Example of bad practice                                                                                                                  | Metrics                                                                   |
|--------------|-------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------|
| Consistency  | No matter where you look in the database, you won't find any contradictions in your data.             | Your payment system shows that Jane Brown has made 5 purchases this month, and CRM system contains the same information.                         | Your payment system shows that Jane Brown has made 5 purchases this month, while CRM system shows she has made only 4.                   | The number of inconsistencies.                                            |
| Accuracy     | The information your data contains corresponds to reality.                                            | Your customer's name is Jane Brown. And this is exactly how it's reflected in your CRM.                                                          | In your CRM, the customer's name is spelled Jane Brawn, though her actual name is Jane Brown.                                            | The ratio of data to errors.                                              |
| Completeness | All available elements of the data have found their way to the database.                              | You know that Jane Brown is born on 11/04/1975.                                                                                                  | You have no idea how old Jane Brown is, as the date of birth cell is empty.                                                              | The number of missing values.                                             |
| Auditability | Data is accessible and it's possible to trace introduced changes.                                     | You can track down the changes made in Jane's data record. For example, on 12/5/2018, her phone number was changed.                              | It's impossible to trace down the changes in Jane's record.                                                                              | % of cells where the metadata about introduced changes is not accessible. |
| Orderliness  | The data entered has the required format and structure.                                               | The entry for December 11, 2018 is in the format 12/11/2018.                                                                                     | The entry for December 11, 2018 is in the format 12/11/18, 12/11/2018 and even 11/12/18 (in your European stores).                       | The ratio of data of inappropriate format.                                |
| Uniqueness   | A data record with specific details appears only once in the database.                                | You have only one record for Jane Brown, born on 11/04/1975, who lives in Seattle.                                                               | You have multiple duplicate records for Jane Brown.                                                                                      | The number of duplicates revealed.                                        |
| Timeliness   | Data represents reality within a reasonable period of time or in accordance with corporate standards. | On 02/15/2018, the customer informed you that her name is misspelled in the emails you send her. The customer's name was corrected the next day. | On 02/15/2018, the customer informed you that her name is misspelled in the emails you send her. Her name was corrected only in a month. | Number of records with delayed changes.                                   |

## MLOPS with AWS 9 Weeks Course



## Streamlit Tutorials Playlist

8 days of Streamlit Tutorials Playlist

<https://youtube.com/playlist?list=PLpdmBGJ6ELUKVcOnN3PrIICc50MZ7qor6>

## Online AI projects

<https://docs.google.com/document/d/14FY45os1zjR4iVtZ8wlwET6RkzG5rDJkdVfFLBarK4g>

## 3D Vision Summer School

<https://cvit.iiit.ac.in/workshops/3dvision/>

## 40 Open-Source Audio Datasets for ML

<https://t.co/JQTOPXMHVg>

#AI #MachineLearning #DeepLearning #DataScience

<https://towardsdatascience.com/40-open-source-audio-datasets-for-ml-59dc39d48f06>

## **Intro to Intel's OpenAPI**

<https://pub.towardsai.net/introduction-to-intels-oneapi-ai-analytics-toolkit-8dd873925b96>

## **HackerNoon**

<https://hackernoon.com/ai>

# Machine Learning Tools

## Machine Learning Languages



Python



R



C++

## Data Analysis and Visualisation tools



Pandas



Matplotlib



Jupyter Notebook

## General Machine Learning Frameworks



Numpy



Scikit-learn



NLTK



Tableau



Weka

## ML frameworks for neural network modelling



Tensorflow & Tensorboard



Pytorch



Keras

## Big Data Tools



Apache Spark



MemSQL



Caffe2

## Some other tools



### Accord.Net

Accord.Net is a .Net machine learning framework combined with audio and image processing libraries written in C#.



### Shogun

It provides data structures and algorithms for machine learning problems and supports many languages like Python, R, Octave, Java, C#, Ruby, Lua, etc.



### Apache Mahout

It is a distributed linear algebra framework and mathematically expressive Scala DSL.



### Singa

This software is primarily used in natural language processing (NLP) and image recognition and supports a wide range of deep learning models.



### PredictionIO

Helps in building predictive engines for AI and ML tasks.

## Statistical, Mathematical Tools

R  
SciLab



SAS  
SPSS  
PSPP  
StatsModels  
SciPy  
NumPy

### **General Purpose**

Scikit-Learn  
Weka3  
MOA  
RapidMiner  
MatLab  
Octave  
Shogun

### **Narrowed Solutions**

XGBoost  
Vowpal Wabbit  
LibLINEAR  
LibSVM  
NLTK  
Gensim

### **Interactive Platforms**

Jupyter  
Zeppelin  
Orange  
Kibana  
Grafana  
Tableau

### **ML/DL with MapReduce**

Deeplearning4J  
H2O  
Deep Water  
Spark ML  
ML Lib  
FlinkML  
Oryx2 ML

### **DL with GPU**

TensorFlow  
CNTK  
Caffe  
Caffe2  
MXNet  
Theano  
chainer  
PaddlePaddle  
MatConvNet

### **DL Wrapper Libraries**

Keras  
Gluon  
TensorLayer  
TFLearn  
Lasagne  
KNIME  
NVidia Digits





## Data Science Competitions

# Data Science Competitions



Julia



SPSS

Take the Challenge!

Kaggle

Hackathon

## Why most people hesitate to participate ?

1. They belittle their skills & knowledge
2. They fall into the trap of lucrative prize money
3. Lack of structured guidance and resources to follow



## Why should you participate now ?

1. You learn from your peer data scientists
2. You get experience of working on problems from different domains
3. You can compare your skills to know your strong & weak areas



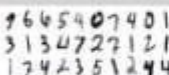
## List of Kaggle Problems



Titanic : Machine Learning from Disaster

Difficulty level

- a) Machine Learning Skills - Easy
- b) Coding skills - Easy
- c) Acquiring Domain Skills - Easy
- d) Tutorial available - Very comprehensive



Digit Recognizer

Difficulty level

- a) Machine Learning Skills - Medium
- b) Coding skills - Medium
- c) Acquiring Domain Skills - Easy
- d) Tutorial available - Available but no hand holding

for creating this corpus, do not contain Spanish text is a widespread major language used to create a corpus tasks. These tasks are:

Denosing Dirty Documents

Difficulty level

- a) Machine Learning Skills - Difficult
- b) Coding skills - Difficult
- c) Acquiring Domain Skills - Difficult
- d) Tutorial available - No



Taxi Trajectory Prediction Time / Location

Difficulty level

- a) Machine Learning Skills - Easy
- b) Coding skills - Difficult
- c) Acquiring Domain Skills - Medium
- d) Tutorial available - A few benchmark codes available



First Step with Julia

Difficulty level

- a) Machine Learning Skills - Easy
- b) Coding skills - Medium
- c) Acquiring Domain Skills - Easy
- d) Tutorial available - Comprehensive



Bag of Words meets Bag of Popcorn

Difficulty level

- a) Machine Learning Skills - Difficult
- b) Coding skills - Medium
- c) Acquiring Domain Skills - Easy
- d) Tutorial available - Available but no hand holding



San Francisco Crime Classification

Difficulty level

- a) Machine Learning Skills - Very Difficult
- b) Coding skills - Very Difficult
- c) Acquiring Domain Skills - Difficult
- d) Tutorial available - No



Facebook Recruiting - Human or Bot

Difficulty level

- a) Machine Learning Skills - Medium
- b) Coding skills - Medium
- c) Acquiring Domain Skills - Medium
- d) Tutorial available - No support available

\*Some of the competitions might be closed but you can still download the datasets

What should be the right approach to



## **PyScript - JavaScript's sweet cousin.**

<https://t.co/7J7miFbgBK>

🐍 Python in browser movement gained some momentum nowadays. All thanks to the PyScript library. See how easy it is to bring it into a simple HTML page with no additional setup

## **GraphWorld: A Methodology For Analyzing The Performance Of GNN Architectures**

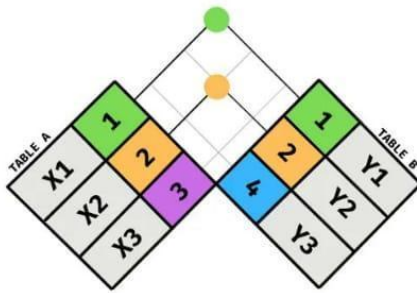
Google AI Introduces GraphWorld: A Methodology For Analyzing The Performance Of GNN Architectures On Millions Of Synthetic Benchmark Datasets - MarkTechPost

<https://www.marktechpost.com/2022/05/11/google-ai-introduces-graphworld-a-methodology-for-analyzing-the-performance-of-gnn-architectures-on-millions-of-synthetic-benchmark-datasets/>

## **github: connorferster**

<https://github.com/connorferster/handcalcs>

# SQL Joins

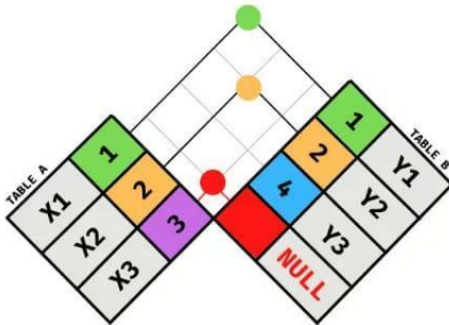


## INNER JOIN



```
SELECT
<SELECT LIST>
FROM TABLE_A A
INNER JOIN TABLE_B B
ON A.KEY = B.KEY
```

| KEY | VAL_X | VAL_Y |
|-----|-------|-------|
| 1   | X1    | Y1    |
| 2   | X2    | Y2    |

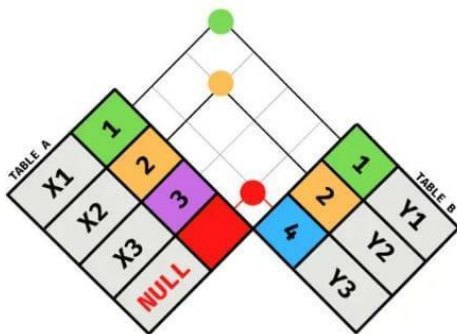


## LEFT JOIN



```
SELECT
<SELECT LIST>
FROM TABLE_A A
LEFT JOIN TABLE_B B
ON A.KEY = B.KEY
```

| KEY | VAL_X | VAL_Y |
|-----|-------|-------|
| 1   | X1    | Y1    |
| 2   | X2    | Y2    |
| 3   | X3    | NULL  |

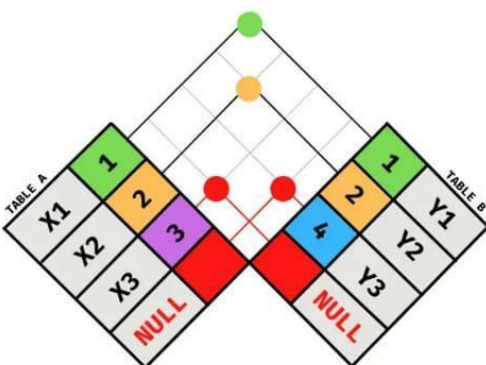


## RIGHT JOIN



```
SELECT
<SELECT LIST>
FROM TABLE_A A
RIGHT JOIN TABLE_B B
ON A.KEY = B.KEY
```

| KEY | VAL_X | VAL_Y |
|-----|-------|-------|
| 1   | X1    | Y1    |
| 2   | X2    | Y2    |
| 4   | NULL  | Y3    |



## FULL OUTER JOIN



```
SELECT
<SELECT LIST>
FROM TABLE_A A
FULL OUTER JOIN TABLE_B B
ON A.KEY = B.KEY
```

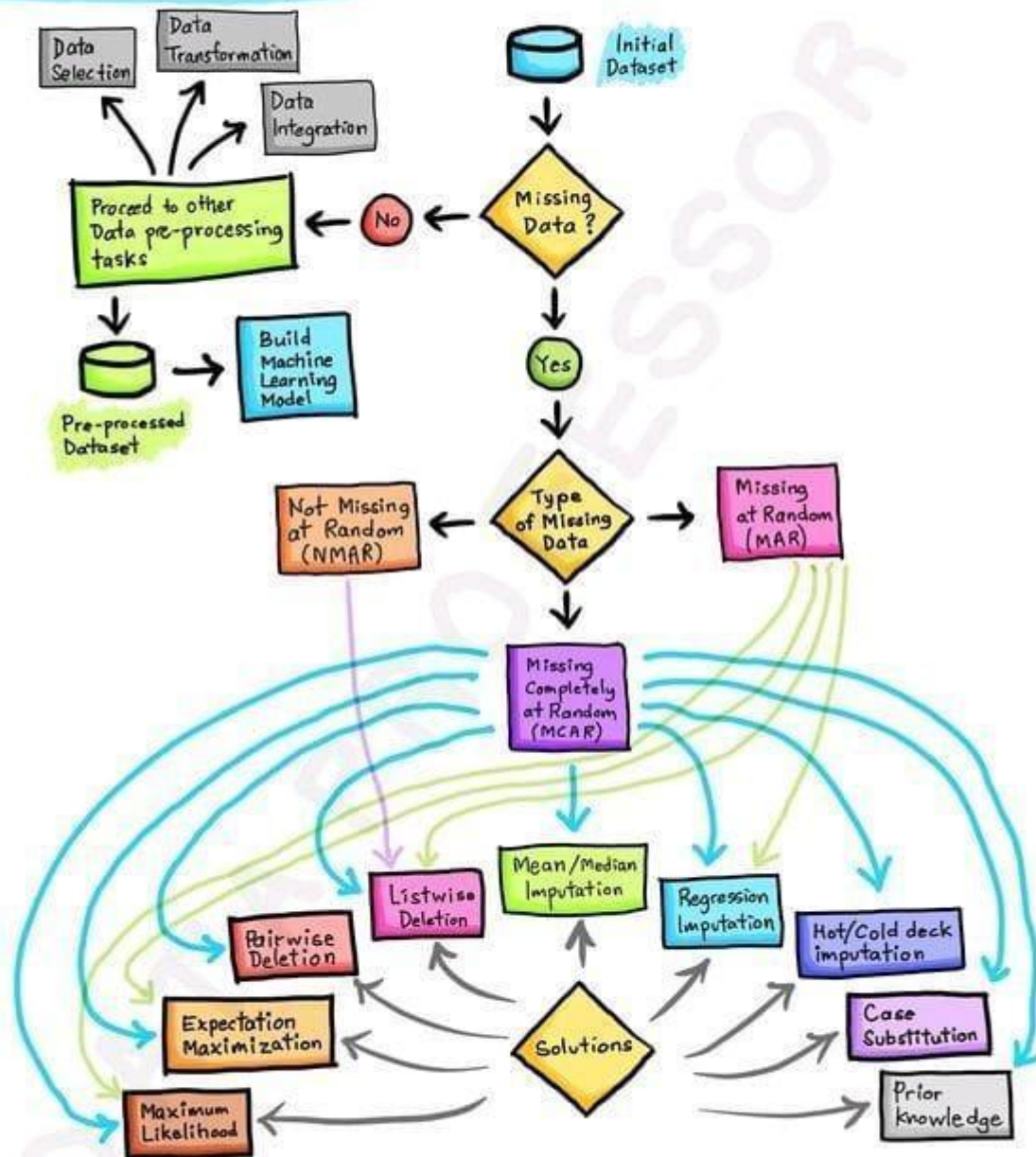
| KEY | VAL_X | VAL_Y |
|-----|-------|-------|
| 1   | X1    | Y1    |
| 2   | X2    | Y2    |
| 3   | X3    | NULL  |
| 4   | NULL  | Y3    |



# Handling Missing Data

## HANDLING MISSING DATA

By: Chanin Nantasenamat

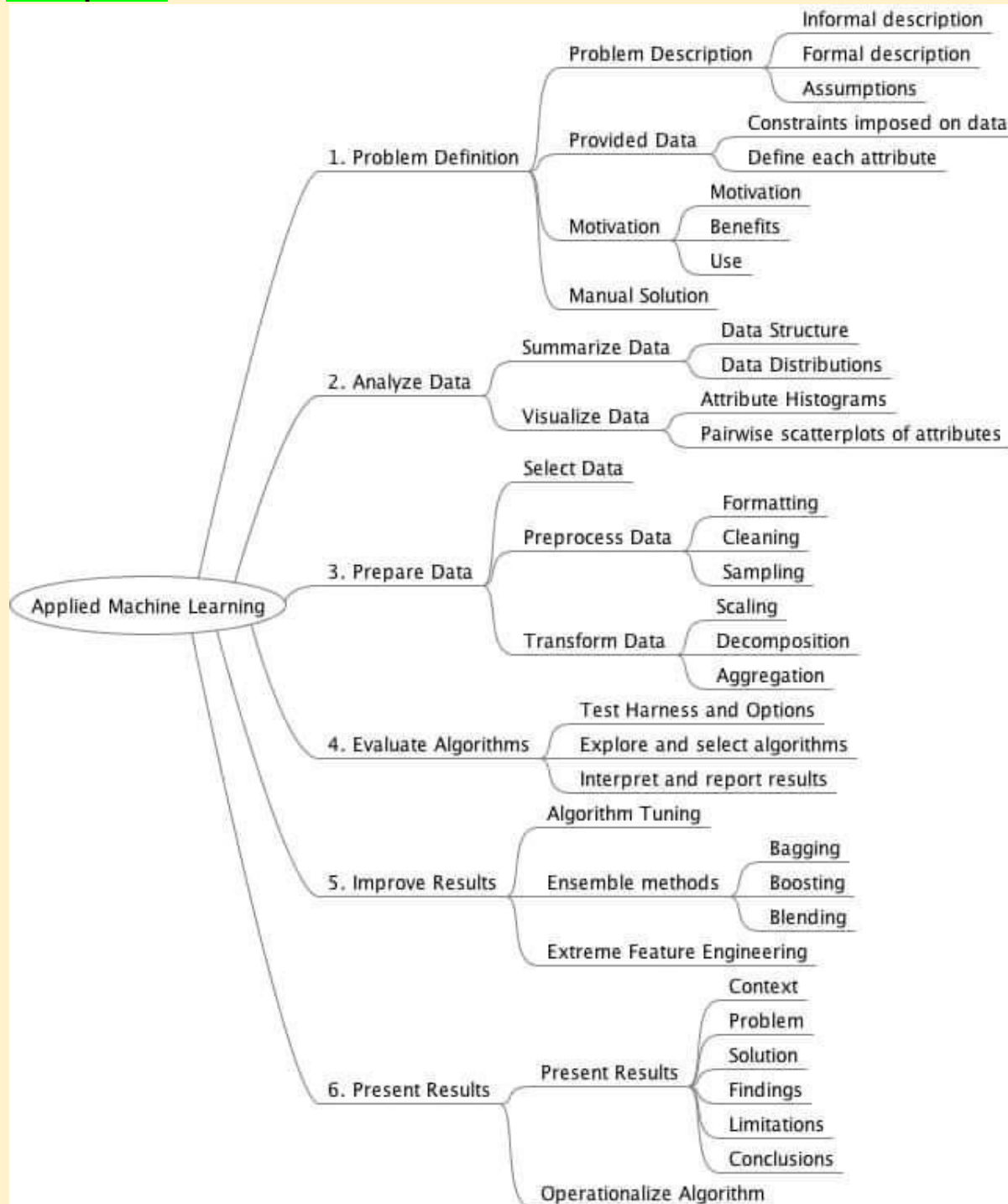


DATA PROFESSOR

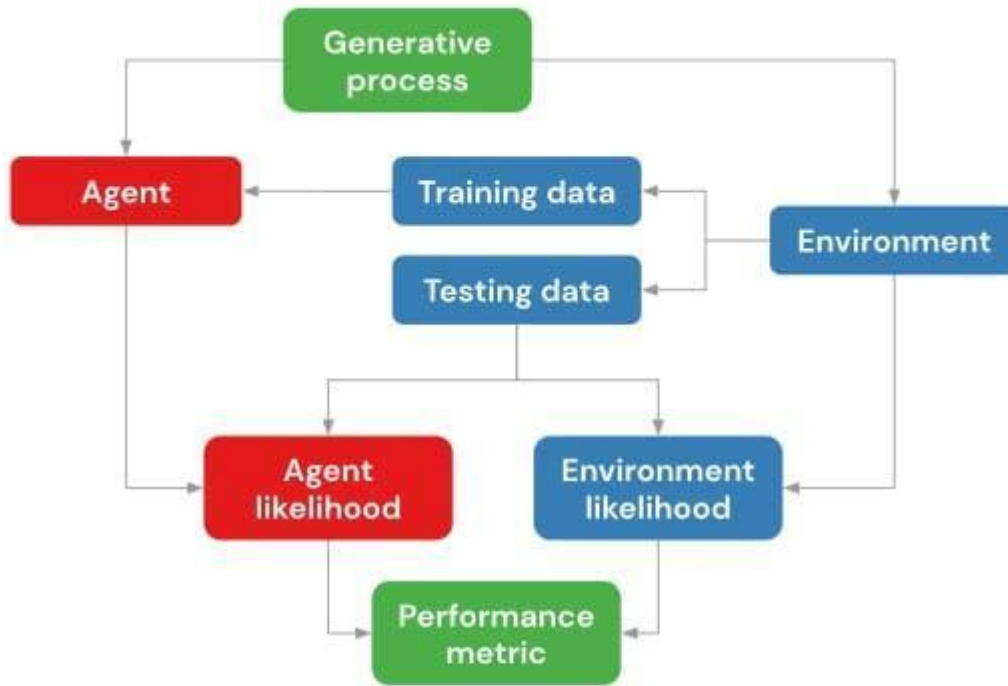


<http://youtube.com/dataprofessor>

# ML Pipeline



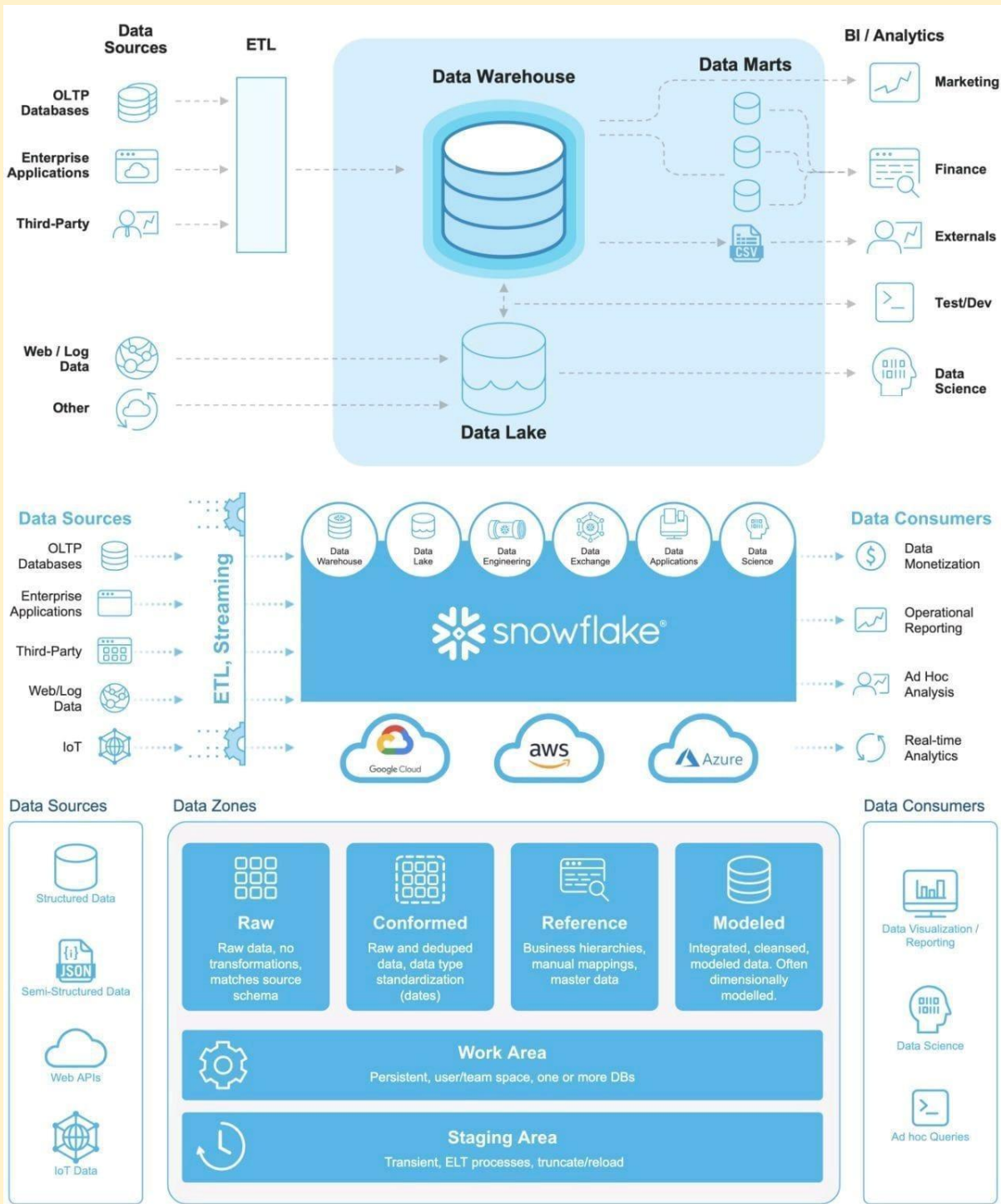
## Generative Process - Overview of Neural Testbed



*Figure 1.* Overview of the Neural Testbed. For each random seed the testbed samples an environment realization with training and testing data. The testbed then compares the estimated log-likelihood of the true environment and that of a trained agent to estimate KL divergence.

github: **SeldoneO**

<https://github.com/SeldonIO/alibi-detect>



## Tracking Progress in Natural Language Processing \ NLP-progress

<http://nlpprogress.com/>

### Neural Voice Camouflage

Researchers From Columbia University Propose, Neural Voice Camouflage: An Adversarial Attack-Based Approach That Disrupts Automatic Speech Recognition Systems In Real-Time

<https://www.marktechpost.com/2022/06/05/researchers-from-columbia-university-propose-neural-voice-camouflage-an-adversarial-attack-based-approach-that-disrupts-automatic-speech-recognition-systems-in-real-time/>

## 9 Free Cloud Storage

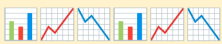


## Generative Adversarial Networks \ Generative Models

<https://www.analyticsvidhya.com/blog/2017/06/introductory-generative-adversarial-networks-gans/>

## Watch "Markov Chains Clearly Explained! Part - 1" on YouTube

<https://youtu.be/i3AkTO9HLXo>



## Watch "Python to Replace VBA in Excel" on YouTube

<https://youtu.be/ubqsRcCUcB4>

## Statistical tests

<https://youtu.be/rullUANOU3w>

## Watch "How to write a report using LATEX" on YouTube

<https://youtu.be/YLm3sXIKpHQ>

Miktex, texmaker

## AIM Survey - State of AI in Indian Enterprises

[https://docs.google.com/forms/d/e/1FAIpQLSfQk0FK4SrfuQBgreM\\_6SLcE1TbwI3cSI\\_vs79m07AP87vwA/viewform](https://docs.google.com/forms/d/e/1FAIpQLSfQk0FK4SrfuQBgreM_6SLcE1TbwI3cSI_vs79m07AP87vwA/viewform)

## Data-Centric AI Competition Submission Guide

<https://worksheets.codalab.org/worksheets/0x7a8721f11e61436e93ac8f76da83f0e6>

## Understanding the role of individual units in a deep neural network \ PNAS

<https://www.pnas.org/content/117/48/30071>

<https://link.medium.com/FAo0huUkghb>

## TPOT

<https://machinelearningmastery.com/tpot-for-automated-machine-learning-in-python/>

<https://news.towardsai.net/laptops>



## **Alexnet Architecture \ Introduction to Architecture of Alexnet**

<https://www.analyticsvidhya.com/blog/2021/03/introduction-to-the-architecture-of-alexnet/>

## **Watch "Introduction to AWS Services" on YouTube**

<https://youtu.be/Z3SYDTMP3ME>

## **Watch "Microservices Interview questions \ Interview Preparation" on YouTube**

<https://youtu.be/mM-VXVwjtPo>

## **Watch "12 Factors App \ MicroServices Architecture \ Cloud Native Best Practices" on YouTube**

<https://youtu.be/N45VJ20wvCw>

## **Watch "Microservices Design Patterns \ Microservices Architecture Patterns \ Edureka" on YouTube**

<https://youtu.be/xuH81XGWeGQ>

## **Watch "Microservices vs API \ Differences Between Microservice and API \ Edureka" on YouTube**

<https://youtu.be/GwafAoXKV8A>

## **Watch "Build a custom ML model with Vertex AI" on YouTube**

<https://youtu.be/l2CJL3tQasw>

[https://cnvrg.io/mlcon/?utm\\_source=thesequence&utm\\_medium=email&utm\\_campaign=thesequence-newsletter](https://cnvrg.io/mlcon/?utm_source=thesequence&utm_medium=email&utm_campaign=thesequence-newsletter)

## **[Infographics] Data Science Skills Roadmap: Skills and Certifications to Have in 2021 \ SDS Club**

<https://sdsclub.com/data-science-skills-roadmap-skills-and-certifications-to-have-in-2021/>

## **Autoencoders in Python \ How to use Autoencoders in Python**

<https://www.analyticsvidhya.com/blog/2021/06/complete-guide-on-how-to-use-autoencoders-in-python/>

## **How to Calculate Feature Importance With Python**

<https://machinelearningmastery.com/calculate-feature-importance-with-python/>

<https://link.medium.com/ND3WDMVquhb>

Enable service

Get key

Open notebook

Enter key

Upload img  
Train  
Test pred with dictionary

Cumputer vision  
Custom object  
Object  
Face  
Form

<https://www.itexams.com/exam/AI-900>

## **MLOps - Machine Learning Operations**

<https://valohai.com/mlops/>

## **Ten Highest Paying Companies for Data Scientists in 2021**

<https://www.analyticsinsight.net/ten-highest-paying-companies-for-data-scientists-in-2021/>

## **MLOps Course \ MLOps Online Training, India - 360DigiTMG**

<https://360digitmg.com/india/mlops-engineering-course-training>

## **Continuous Integration and Deployment for Machine Learning**

Continuous Integration and Deployment for Machine Learning Online Serving and Models \ Uber Engineering Blog

<https://eng.uber.com/continuous-integration-deployment-ml/>

## **Why AI is a game-changer for renewable energy \ EY - US**

[https://www.ey.com/en\\_us/power-utilities/why-artificial-intelligence-is-a-game-changer-for-renewable-energy](https://www.ey.com/en_us/power-utilities/why-artificial-intelligence-is-a-game-changer-for-renewable-energy)  
y

## **Worldscholarship Forum**

[https://www.google.com/url?sa=t&source=web&rct=j&url=https://worldscholarshipforum.com/phd-data-science-scholarships/&ved=2ahUKEwic3NTXv8zxAhU59nMBHZ\\_cDeAQFjATegQIEhAC&usq=AOvVaw3\\_IAHejcIJwN0N2o87LwMg](https://www.google.com/url?sa=t&source=web&rct=j&url=https://worldscholarshipforum.com/phd-data-science-scholarships/&ved=2ahUKEwic3NTXv8zxAhU59nMBHZ_cDeAQFjATegQIEhAC&usq=AOvVaw3_IAHejcIJwN0N2o87LwMg)

## **Bird Song Classification using Siamese Networks and Dilated Convolutions**

<https://towardsdatascience.com/bird-song-classification-using-siamese-networks-and-dilated-convolutions-3b38a115bc1>

## **Matplotlib vs. Plotly: Let's Decide Once and for All**

<https://towardsdatascience.com/matplotlib-vs-plotly-lets-decide-once-and-for-all-ad25a5e43322>

## **Dispatch from Bangalore – TechCrunch**

<https://techcrunch.com/2021/07/06/indian-startups-fundraise-spree/>

## **RoBERTa building for Italian lang**

<https://link.medium.com/uQE5M5Vdlhb>

## **PyTorch 1.9 - Towards Distributed Training and Scientific Computing**

<https://www.analyticsvidhya.com/blog/2021/07/pytorch-1-9-towards-distributed-training-and-scientific-computing/>

## **Train New BERT Model on Any Language**

<https://towardsdatascience.com/how-to-train-a-bert-model-from-scratch-72cfce554fc6>

## **How to Create Report-Ready Plots in Python**

<https://builtin.com/data-science/report-ready-plots-python>

## **OpenAI Launches GitHub Copilot: AI Focused On Code Generation. Should We Be Worried Now?**

<https://towardsdatascience.com/should-we-be-worried-now-that-github-copilot-is-out-12f59551cd95>

## **What OpenAI and GitHub's 'AI pair programmer' means for the software industry**

<https://venturebeat.com/2021/07/06/what-openai-and-githubs-ai-pair-programmer-means-for-the-software-industry/>

## **Importing/Reading Excel data into R using RStudio (readxl) \ R Tutorial 1.5b \ MarinStatsLectures**

<https://youtu.be/JYVWufSQ4OI>

## **How I passed the AWS Solutions Architect Associate Exam in 1 month 2021**

<https://youtu.be/jypuayQpvao>

<https://joss.theoj.org/>

## **Hindawi Journals**

<https://www.hindawi.com/journals/complexity/2021/5545491/>

## **Is it possible to use multithreading inside of Flask?**

[https://www.reddit.com/r/flask/comments/flrv0e/is\\_it\\_possible\\_to\\_use\\_multithreading\\_inside\\_of/?utm\\_medium=android\\_app&utm\\_source=share](https://www.reddit.com/r/flask/comments/flrv0e/is_it_possible_to_use_multithreading_inside_of/?utm_medium=android_app&utm_source=share)

## **Introduction to Deep Learning**

<https://sebastianraschka.com/blog/2021/dl-course.html>

## **100+ Free Machine Learning Books**

<https://www.theinsaneapp.com/2020/12/download-free-machine-learning-books.html>

<https://www.theinsaneapp.com/2020/12/free-data-science-books-pdf.html>

## **Free online book - Machine Learning from Scratch**

<https://www.datasciencecentral.com/profiles/blogs/free-online-book-machine-learning-from-scratch>

## **Skill Basics**

<https://www.skillbasics.com/>

## **Operating System For Machine Learning & AI**

<https://cnvrg.io/>

## **From Pandas to PySpark with Koalas**

<https://towardsdatascience.com/from-pandas-to-pyspark-with-koalas-e40f293be7c8>

## **Free Data Science, Machine Learning**

Free Data Science, Machine Learning & Statistics eBooks for 2021 - KDnuggets

1. <https://www.kdnuggets.com/2020/12/15-free-data-science-machine-learning-statistics-ebooks-2021.html>
2. <https://www.kaggle.com/general/95287>
3. <https://towardsdatascience.com/the-best-free-data-science-ebooks-b671691e5231>

4. <https://www.theinsaneapp.com/2020/12/free-data-science-books-pdf.html?m=1>
5. <https://www.datasciencecentral.com/profiles/blogs/5-free-data-science-ebooks-for-your-summer-reading-list>
6. <https://www.ubuntupit.com/best-data-science-books-available-online/?amp=1>
7. <https://www.theinsaneapp.com/2020/11/free-machine-learning-data-science-and-python-books.html?m=1>
8. <https://towardsdatascience.com/the-best-free-data-science-ebooks-b671691e5231>
9. <https://www.kdnuggets.com/2020/n38.html>
10. <https://towardsdatascience.com/the-best-free-data-science-ebooks-2020-update-dac5e170a478>
11. [https://www.analyticsvidhya.com/blog/2018/02/10-free-must-read-machine-learning-e-books/?utm\\_source=twitter.com&utm\\_medium=social](https://www.analyticsvidhya.com/blog/2018/02/10-free-must-read-machine-learning-e-books/?utm_source=twitter.com&utm_medium=social)
12. <http://brainstormingbox.org/10-free-must-read-data-science-and-machine-learning-e-books-in-2020/>
13. <https://www.datasciencecentral.com/profiles/blogs/5-free-data-science-ebooks-for-your-summer-reading-list>

## **NET Architecture Guides**

You can make lots more than just "apps" with @dotnet. You can do Data Science with MLNET, build distributed systems with k8s, Orleans, Akka, dapr, and more. You can build incredibly fast things and build amazing frameworks.

<https://dotnet.microsoft.com/learn/dotnet/architecture-guides>

## **Getting Started with GPT-3 in Power Platform**

<https://devblogs.microsoft.com/premier-developer/getting-started-with-gpt-3-in-power-platform/>

## **Netflix Recommendations**

<http://techblog.netflix.com/2012/04/netflix-recommendations-beyond-5-stars.html>

## **Techlearn**

<https://www.techlearn.live/netflix-recommendation-how-do-they-know-what-you-like>

## **Data Ingestion with TensorFlow eXtended (TFX)**

<https://dztlab.github.io/ml/2020/09/13/tfx-data-ingestion/>

## **The ExampleGen TFX Pipeline Component**

<https://www.tensorflow.org/tfx/guide/examplegen>

## **The School of AI**

<https://theschoolof.ai/>

## **Github: GREMLIN: An Apache TinkerPop Tutorial**

<http://www.kelvinlawrence.net/book/PracticalGremlin.html>

<https://github.com/deepklarity/jupyter-text2code>

## **Quantum Machine Learning**

<https://www.edx.org/course/quantum-machine-learning>

## **Quantum Machine Learning MOOC**

[https://youtube.com/playlist?list=PLmRxcgFnCIhaMgvot-Xuym\\_hn69lmzlokq](https://youtube.com/playlist?list=PLmRxcgFnCIhaMgvot-Xuym_hn69lmzlokq)

## **Quantum Mechanics and Quantum Computation**

<https://www.edx.org/course/quantum-mechanics-and-quantum-computation>

## Quantum Computation

[https://youtube.com/playlist?list=PL74Rel4lAsETUwZS\\_Se\\_P-fSEyEVQwni7](https://youtube.com/playlist?list=PL74Rel4lAsETUwZS_Se_P-fSEyEVQwni7)

## Qiskit

<https://qiskit.org/>

## TensorFlow Quantum

<https://www.tensorflow.org/quantum>

## CirqBasics.ipynb - Colaboratory

<https://colab.research.google.com/drive/17ysMYVSj1KYSbOZ2YAe7DuVJW5L0cDgD?usp=sharing>

## QpiAI - AI and Quantum Simplified

<https://qpi.ai.tech/>

<https://www.investopedia.com/articles/active-trading/092914/normal-distribution-table-explained.asp>

## MadewithML

<https://madewithml.com/#mlops>

## Kick Start - Google's Coding Competitions

<https://codingcompetitions.withgoogle.com/kickstart>

## Online Python Challenges - Python Principles

<https://pythonprinciples.com/challenges/>

## Gaining access to the best machine-learning methods - O'Reilly Radar

<http://radar.oreilly.com/2013/09/gaining-access-to-the-best-machine-learning-methods.html>

## Data Science Blogathon - 11

- [https://datahack.analyticsvidhya.com/contest/data-science-blogathon-11/?utm\\_source=webpush&utm\\_medium=notification&utm\\_campaign=blogathon11](https://datahack.analyticsvidhya.com/contest/data-science-blogathon-11/?utm_source=webpush&utm_medium=notification&utm_campaign=blogathon11)

## The Machine & Deep Learning Compendium

- <https://towardsdatascience.com/the-last-machine-deep-learning-compendium-youll-ever-need-dc973643c4e1>
- <https://github.com/orico/www.mlcompendium.com>
- <https://mlcompendium.gitbook.io/machine-and-deep-learning-compendium/>
- <https://book.mlcompendium.com/>
- <https://towardsdatascience.com/the-last-machine-deep-learning-compendium-youll-ever-need-dc973643c4e1?gi=ff8d2179ae8b>

## Github: NLP Resources dlq4nlp

Website: <https://dlq4nlp.github.io/index.html>

IJCAI'21 Slides: [https://drive.google.com/file/d/1ta1G\\_a-WnJ9MpxuKYBvNABs38TM3ts/view](https://drive.google.com/file/d/1ta1G_a-WnJ9MpxuKYBvNABs38TM3ts/view)

Survey: <https://arxiv.org/abs/2106.06090>

Library: <https://github.com/graph4ai/graph4nlp>

Demo: [https://github.com/graph4ai/graph4nlp\\_demo](https://github.com/graph4ai/graph4nlp_demo)

Github: [https://github.com/graph4ai/graph4nlp\\_demo](https://github.com/graph4ai/graph4nlp_demo)

## Machine Learning Life Cycle 2.0 - Mind Map

<https://youtu.be/CWnMJoaUsG0>

## Handbook on Data Protection and Privacy

Handbook on Data Protection and Privacy for Developers of Artificial Intelligence (AI) in India: Practical Guidelines for Responsible Development of AI \ Data Security Council of India

<https://www.dsci.in/content/privacy-handbook-for-ai-developers>



## 6 low-code internal tool builders in 2021

<https://www.analyticsinsight.net/5-best-no-code-and-low-code-internal-tool-builders/>

## AI books

Five FREE Books to learn Machine Learning and Data Science

### 1. Understanding Machine Learning: From Theory to Algorithms

The book provides a theoretical account of the fundamentals underlying machine learning and the mathematical derivations that transform these principles into practical algorithms. <https://t.co/6Z90ju7nTI>

### 2. Think Stats — Probability and Statistics for Programmers

This book written by Allen B Downey and published by O'Reilly Media is an introduction to Probability and Statistics for Python programmers. <https://t.co/0PcIIA0omi>

### 3. Machine Learning Yearning

Author @AndrewYNg states that the book's objective is to "teach one how to make the numerous decisions needed with organizing a machine learning project." <https://t.co/bT0Fu6td83>

### 4. Data-Driven

Succeeding with data isn't just a matter of putting Hadoop in your machine room, or hiring some physicists with crazy math skills. It requires you to develop a data culture that involves people throughout the organization. <https://t.co/R2Y3I2jIBy>

### 5. A Brief Introduction to Neural Networks

From the history of Neural Networks to their training, author D Kriesel systematically explains the subject in this book. <https://t.co/6UjIdP10Yw1>

## Resources to learn DS

All the Machine Learning Resources in one place.

\* Table of Contents \*

Data Quality

Data Engineering

Data Discovery

Feature Stores

Classification

Regression

Forecasting

Recommendation

Search & Ranking

Embeddings

Natural Language Processing

Sequence Modelling

Computer Vision

Reinforcement Learning Anomaly Detection

Graph Optimization

Information Extraction Weak Supervision

Generation

Audio

Validation and A/B Testing

Model Management

Efficiency

Ethics

MLOps Platforms

Practices

Team Structure  
Fails

## Parallel Processing and Pandas

Long execution time is due to several reasons

1. Cluster config -- too less nodes or too less RAM
2. Language ur using -- for parallel computing always DON'T use python (scala or Java is fast, next is sql or pyspark)
3. If u still wanna use python/pandas try koalas which is a wrapper on python n lets u use power of distributed computing

There are pandas like apis in python which support parallel computing Dask, Vaex etc ..do check them out Koalas run on spark architecture and similar to pandas api.. if you hate spark syntax then koalas will come to rescue

This course on "Introduction to Computational Thinking" is exceptional.

Computational Thinking is useful for CS and machine learning students.

Topics include image analysis, statistics, differential equations,...

A fun way to start learning Julia.

<https://t.co/TcX9Cs1yc7>

Things to Know Before Using Julia for Machine-Learning. #BigData #Analytics #DataScience #AI #MachineLearning #IoT #IIoT #PyTorch #JuliaLang #Python #RStats #TensorFlow #Java

<https://t.co/T3HAXeZESf>

## Julia vs. Python

Julia vs. Python: Which is best for data science

<https://www.infoworld.com/article/3241107/julia-vs-python-which-is-best-for-data-science.html>

## Introduction to Computational Thinking

This course on "Introduction to Computational Thinking" is exceptional.

Computational Thinking is useful for CS and machine learning students.

Topics include image analysis, statistics, differential equations,

## A fun way to start learning Julia.

<https://t.co/TcX9Cs1yc7>

Github: Stats with Julia

<https://github.com/h-Klok/StatsWithJuliaBook>






<https://github.com/StatisticalRethinkingJulia/StatisticsWithJuliaPlutoNotebooks.jl>

## Data Science Tools



## 5 Most Popular Datasets

### Five of the most popular ML dataset resources:

-  → Google's Dataset Search
-  Microsoft → Microsoft Research Open Data
-  → Amazon Datasets
-  → UCI Machine Learning Repository
-  → Government Datasets

## Drug Discovery Using Artificial Intelligence

- Rethinking Drug Discovery in the Era of Digital Biology \ Invited Talk \ Daphne Koller \ ICML 2021: <https://youtu.be/Fd0jvfMegvM>
- Discovering Genetic Medicines with our AI Drug Discovery Platform \ NeurIPS 2020: [https://youtu.be/\\_uo5fEjrNyU](https://youtu.be/_uo5fEjrNyU)
- Challenges and Opportunities for Machine Learning in Drug Discovery \ NeurIPS 2020: <https://youtu.be/H3AaVj0bopM>
- Real-World Application Of Machine Learning In Drug Discovery \ NeurIPS 2020: <https://youtu.be/vuRkovLXLi8>
- Artificial Intelligence for Drug Discovery Tutorial AAAI - 2021: <https://youtu.be/SSaiWxvMbg0>
- Advanced Graph and Sequence Neural Networks for Molecular Property Prediction and Drug Dis Part 5: <https://youtu.be/iGHQuk4btJo>

- Multi-view Graph Contrastive Representation Learning for Drug Interaction Prediction Part 4:  
<https://youtu.be/DZkC0I6jmYY>

#DrugDiscovery #DeepLearning #ArtificialIntelligence

## **DLG4NLP tutorial**

DLG4NLP tutorial

KDD'21 Slides: <https://t.co/9BQCYSK6SE>

Website: <https://t.co/7GW6nsWoQ3>

#DeepLearning #GNNUpdates #GNN #NLProc

## **Deep Reinforcement Learning by Pieter Abbeel**

6-lecture series on Foundations of Deep Reinforcement Learning by Pieter Abbeel

<https://youtube.com/playlist?list=PLwRJQ4m4UJjNymuBM9RdmB3Z9N5-0IIY0>

MDP basics, value & policy iteration, max-ent, DQN, policy gradient, TRPO, PPO, DDPG, SAC, model-based RL.

Slides: <https://www.dropbox.com/s/to43tskzsydqhel/foundations-deep-rl-abbeel.zip?dl=0>

<https://youtube.com/playlist?list=PLwRJQ4m4UJjNymuBM9RdmB3Z9N5-0IIY0>

## **6 Highest Paying Data Science Certifications You Should Know**

<https://www.analyticsinsight.net/6-highest-paying-data-science-certifications-you-should-know/>

## **The Great Hack \ Netflix Official Site**

<https://www.netflix.com/in/title/80117542>

## Probability for Data Scientist



**Kalyan KS** · 1st

PhD candidate | Clinical Natural Language...

19h · 🌐

Probability for Computer Scientists (Course from **Stanford University**) (link in the comments)

**#probability #datascience #ai #course #  
#machinelearning #artificialintelligence  
#deeplearning**

**CS 109: Probability for Computer Scientists, Summer 2021**

### Materials

- Schedule
- Syllabus
- Course Staff (Office Hours)
- Zoom (Canvas)

### Resources

- Lecture Slides
- 5-Min Videos
- Gradescope
- Discussion
- Board (Edstem)

### Announcements and Updates

- [6/21] Welcome to CS109! The course is open for shopping on Canvas - instructions on how to add yourself to Gradescope/Edstem are on the front page there. Please email us if you are having trouble accessing any materials.

### Schedule

| Week 1                  | Topic                                                | Materials                                                                                                                                  | Assignments                                                                    |
|-------------------------|------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------|
| Lecture 1 (Mon, Jun 21) | 1.1 So You Think You Can Count?<br>1.2 More Counting | 1.1 <a href="#">slides</a> <a href="#">notes</a> <a href="#">cc</a><br>1.2 <a href="#">slides</a> <a href="#">notes</a> <a href="#">cc</a> | PSet 1 Out:<br><a href="#">PDF</a> <a href="#">LaTeX Template/Starter Code</a> |



Like



Comment



Share



Send



**Kalyan KS** **Author**

PhD candidate | Clinical Natural Language Proces...

<http://web.stanford.edu/class/cs109/>

## Quantum Computing, Artificial Intelligence, & Machine Learning for Drug Discovery

[https://www.pmlive.com/pmhub/pharma\\_training/impetus\\_digital/videos/quantum\\_computing\\_artificial\\_intelligence\\_and\\_machine\\_learning\\_for\\_drug\\_discovery?SQ\\_DESIGN\\_NAME=2](https://www.pmlive.com/pmhub/pharma_training/impetus_digital/videos/quantum_computing_artificial_intelligence_and_machine_learning_for_drug_discovery?SQ_DESIGN_NAME=2)

## Gaussian processes (1/3) - From scratch

<https://peterroelants.github.io/posts/gaussian-process-tutorial/>

## GRAPH4NLP

"Graph4NLP is an easy-to-use library for R&D at the intersection of Deep Learning on Graphs and Natural Language Processing (i.e., DLG4NLP).



It provides both full implementations of state-of-the-art models for data scientists and also flexible interfaces to build customized models for researchers and developers with whole-pipeline support. Built upon highly-optimized runtime libraries including DGL , Graph4NLP has both high running efficiency and great extensibility.

The architecture of Graph4NLP is shown in the following figure, where boxes with dashed lines represents the features under development."

## Graph4NLP\_Literature

This is an impressive repository of literature regarding Deep Learning on Graphs for NLP. Topics include:

- Dialogue Generation
- Open-domain Question Answering
- Commonsense
- Knowledge Graph Alignment
- Named-entity Recognition
- Math Word Problem
- Topic Modeling
- Dialogue State Tracking
- Entity Typing in KB
- Dependency Parsing
- Text Matching
- Question Generation
- Next Utterance Prediction
- Code Summarization
- Natural Language Inference
- Semantic Role Labeling
- Fact Verification
- Sentiment Analysis
- Semantic Parsing
- Knowledge Graph Embedding
- Knowledge Base Completion
- Summarization
- Text Classification
- Relation Extraction
- Machine Reading Comprehension
- Community Question Answering
- Machine Translation
- Knowledge Base Completion/Reasoning

Github: Graph4NLP Repository:

<https://github.com/graph4ai/graph4nlp>

Github: Graph4NLP\_Literature:

[https://github.com/graph4ai/graph4nlp\\_literature](https://github.com/graph4ai/graph4nlp_literature)



# WE ARE HIRING

## COMPUTER VISION EXPERT

NO. OF VACANCIES:02

- 4 + YEARS EXPERIENCE WITH DEVELOPMENT AND IMPLEMENTATION OF COMPUTER VISION OR DEEP LEARNING ALGORITHMS.
- SIGNIFICANT, DEMONSTRABLE EXPERIENCE WITH STATE-OF-THE-ART DEEP LEARNING NETWORKS IN
  - REAL-TIME OBJECT DETECTION TECHNIQUES FOR 2D/3D OBJECTS,
  - SEMANTIC/INSTANCE SEGMENTATION,
  - HAND GESTURE RECOGNITION,
  - HUMAN POSE ESTIMATION,
  - TRAINING DATASET DESIGN, AUGMENTATION, AND VALIDATION,
  - CLASSIFICATION/SEGMENTATION PERFORMANCE METRICS.
- EXPERIENCE IN WORKING WITH LARGE DATA SETS AND DEVELOPING CLIENT INFRASTRUCTURE PIPELINES.
- STRONG EXPERIENCE IN NEURAL NETWORK OPTIMIZATION (PRUNING, QUANTIZATION AND COMPRESSION).
- STRONG EXPERIENCE WITH PYTHON, C++, AND OPENCV REQUIRED.
- STRONG EXPERTISE WITH DL FRAMEWORKS LIKE TENSORFLOW, PYTORCH, MXNET, ONNX ETC.
- STRONG EXPERTISE WITH SOFTWARE DEVELOPMENT LIFECYCLE - CODING, DEBUGGING, OPTIMIZATION, TESTING, INTEGRATION
- EXPERIENCE WITH DL INFERENCE ACCELERATOR TOOLKITS LIKE TVM, OPENVINO, TENSORRT, PLAIDML (OPTIONAL)
  - DISEASE RECOGNITION
  - ALGORITHMS IMPLEMENTATION
  - REST APIs IMPLEMENTATION
  - GOOGLE ANALYTICS AND AWS SERVER IMPLEMENTATION

KINDLY SHARE YOUR DETAIL RESUME ' ON BELOW MENTIONED E-MAIL ADDRESS



HR@POULTA.COM



- ◆ In the paper the authors propose a new angle on using VAEs in the context of NLP. Namely, they propose to use **Gaussian Mixture VAE** (GMVAE) as a probabilistic **regularizer** layer within BERT architecture. 💎
- ◆ The main advantage of VAE over other regularization techniques (like dropout) is that VAE is **not uniform across embeddings** - more noisy embeddings will lose more information during the reconstruction than their more typical counterparts.
- ◆ As a result - the authors conclude - the network will **rely less** on such **noisy embeddings** during the training - a clear regularizing effect.
- ◆ The authors demonstrate that **NoRBERT** outperforms **BERT** on sentence reconstruction task in terms of BLEU and accuracy using SNLI.

PS: Check the paper for more details on NoRBERT variations, an interesting sequential training procedure and additional results for seq2seq models!

---

💡 Paper: <https://lnkd.in/e9T8Apm6>

💡 Authors: [Aurora Cobo Aguilera](#), [Pablo Martinez Olmos](#), [Antonio Artes-Rodriguez](#), [Fernando Perez-Cruz](#)

📍 Follow [#SundayAiPapers](#) for more posts in this series.

[#nlp](#) [#transformers](#) [#vae](#) [#generativemodels](#)  
[#machinelearning](#) [#ai](#)

- In the paper the authors propose a new angle on using VAEs in the context of NLP. Namely, they propose to use Gaussian Mixture VAE (GMVAE) as a probabilistic regularizer layer within BERT architecture.

The main advantage of VAE over other regularization techniques (like dropout) is that VAE is not uniform across embeddings - more noisy embeddings will lose more information during the reconstruction than their more typical counterparts.

- As a result - the authors conclude - the network will rely less on such noisy embeddings during the training - a clear regularizing effect.

The authors demonstrate that NORBERT outperforms BERT on sentence reconstruction task in terms of BLEU and accuracy using SNLI.

PS: Check the paper for more details on NORBERT variations, an interesting sequential training procedure and additional results for seq2seq models!

Paper: <https://lnkd.in/e9T8Apm6>

Authors: Aurora Cobo Aguilera, Pablo Martinez Olmos, Antonio Artes-Rodriguez, Fernando Perez-Cruz

Follow #SundayAiPapers for more posts in this series.

#nlp #transformers #vae #generativemodels

#machinelearning #ai

#deeplearning #artificialintelligence

<https://link.medium.com/sP5veuqy8ib>

## **NPTel Courses on Computer Science and Engineering**

<https://nptel.ac.in/noc/courses/106/#upcoming>

## **VAE**

<https://arxiv.org/pdf/2108.10764.pdf>

## **AWS vs. Azure vs. Google Cloud - Price Comparison**

<https://cast.ai/blog/ultimate-cloud-pricing-comparison-aws-vs-azure-vs-google-cloud-in-2021/>

## **Data Analytics and Visualization Made Easy - Juice Analytics**

<https://www.juiceanalytics.com/>

## **JupyterLab vs DataSpell**



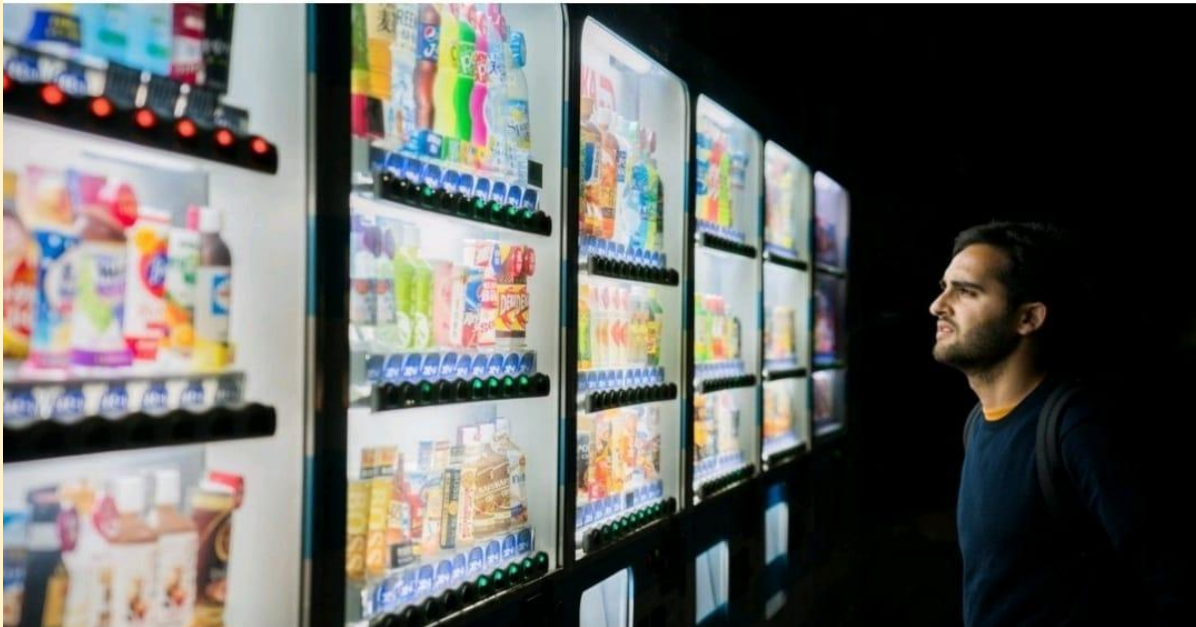


**Eric Feuilleaubois (Ph.D)** • Following

Deep Learner / Machine Learner (PhD in Neural...

6h • 🌐

Tired of JupyterLab? Try DataSpell – A New Amazing IDE for Data Science



Tired of JupyterLab? Try DataSpell – A New Amazing IDE for Data Science

towardsdatascience.com • 1 min read

### **Top 15 YouTube Channels For Deep Learning**

<https://analyticsindiamag.com/top-15-youtube-channels-to-follow-for-deep-learning-enthusiasts/>

### **GitHub : TheAlgorithms/Python**

<https://github.com/TheAlgorithms/Python>

### **Github: Label Studio Spacy**

<https://github.com/iAveshh/Label-Studio-Spacy-2.0>

### **Free Data Science Courses - 11-Months Free IBM Certification Courses**

[https://youtu.be/1e\\_BWnOeIM8](https://youtu.be/1e_BWnOeIM8)

### **Data Science Links**

<https://blog.cambridgespark.com/unit-testing-with-pyspark-fb31671b1ad8>

<https://towardsdatascience.com/time-series-forecasting-arma-models-7f221e9eee06>

## **Free AI Resources - MarkTechPost**

<https://www.marktechpost.com/free-resources/>

## **General and Scalable Parallelization for Neural Networks**

<http://ai.googleblog.com/2021/12/general-and-scalable-parallelization.html>

### **Data Science Tools**

SAS: It is one of those data science tools which are specifically designed for statistical operations. SAS is a closed source proprietary software that is used by large organizations to analyze data.

Matplotlib is a plotting and visualization library developed for Python. It is the most popular tool for generating graphs with the analyzed data. Mainly used for plotting complex graphs.

D3.js: Javascript is mainly used as a client-side scripting language. D3.js, a Javascript library allows you to make interactive visualizations on your web-browser.

Apache Spark or simply Spark is an all-powerful analytics engine and it is the most used Data Science tool. Spark is specifically designed to handle batch processing and Stream Processing.

TensorFlow has become a standard tool for Machine Learning. It is widely used for advanced machine learning algorithms like Deep Learning.

MATLAB is a multi-paradigm environment for processing mathematical information. It is a closed-source software that facilitates matrix, algorithmic implementation, statistical modeling of data.

Jupyter is a tool based on IPython for helping developers in making software and interactive computing. Jupyter supports multiple languages like Julia, Python, R.

BigML, it is another widely used Data Science Tool. It provides a fully interactable, cloud-based GUI environment that you can use for processing Machine Learning Algorithms.

Excel: Most widely used Data Analysis tool. Microsoft developed Excel mostly for spreadsheet calculations and today, it is widely used for data processing, visualization, and complex calculations.

## **Why Machine Learning Is Changing The Advertising Industry**

<https://blog.adext.com/machine-learning-changing-digital-advertising-industry/>

## **Marketers: Here's Your Statistical Models Cheat Sheet**

<http://www.cmswire.com/analytics/marketers-heres-your-statistical-models-cheat-sheet/>

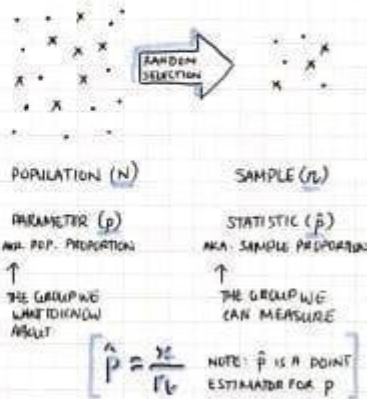
## **The Current Applications Of Artificial Intelligence In Mobile Advertising**

<https://www.forbes.com/sites/andrewarnold/2018/12/24/the-current-applications-of-artificial-intelligence-in-mobile-advertising/>

# Statistical Inference

## RANDOM SAMPLING

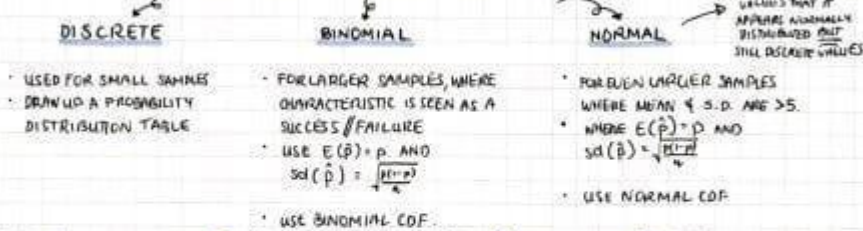
- RANDOM SAMPLING IS OFTEN DONE BECAUSE:
  - POPULATION MAY BE TOO LARGE
  - IT MAY BE DIFFICULT TO ACCESS THE POPULATION
  - DEALING WITH THE ENTIRE POPULATION MAY DESTROY IT.
- A SIMPLE RANDOM SAMPLE IS A RANDOM SAMPLE WHERE:
  - ALL POSSIBLE SAMPLES ARE EQUALLY LIKELY TO BE CHOSEN
  - EVERY POSSIBLE MEMBER OF POP. HAS EQUAL CHANCE TO BE CHOSEN.
  - EACH OBJECT IN SAMPLE IS CHOSEN INDEPENDENTLY



## TYPES OF SAMPLES

- ALMOST AS GOOD AS A RANDOM SAMPLE
- SYSTEMATIC SAMPLE: CHOOSING EVERY  $k^{th}$  MEMBER OF POPULATION
- STRATIFIED RANDOM SAMPLE: MAKING SURE SAMPLES ARE PROPORTIONATE TO SUBGROUPS IN THE POPULATION (OR REPRESENTED IN SIMILAR PROPORTION)
- SELF-SELECTED SAMPLE: WHERE PARTICIPANTS CHOOSE TO PARTICIPATE THEMSELVES
- BARELY REPRESENTATIVE OF POPULATION

## DISTRIBUTION OF SAMPLE PROPORTION



## CONFIDENCE INTERVALS

- "AN INTERVAL OR RANGE OF VALUES WITHIN WHICH WE HAVE CONFIDENCE THE TRUE VALUE OF p LIES."
  - IT IS GIVEN BY  $(\hat{p} - z \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}, \hat{p} + z \sqrt{\frac{\hat{p}(1-\hat{p})}{n}})$ 
    - \* ONLY USE IF MEAN & S.D. > 5 !! OTHERWISE USE BINOMIAL CDF.
- Z VALUES
- 90% CI = 1.64
  - 95% CI = 1.96
  - 99% CI = 2.58

## MARGIN OF ERROR

- "DISTANCE B/W ENDPOINTS OF CONFIDENCE INTERVAL"
- IT IS GIVEN BY  $M = z \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$
- USE THE SAME Z-VALUES



Petar Veličković · 2nd

Staff Research Scientist at DeepMind | Affiliated...

3d · 🌐

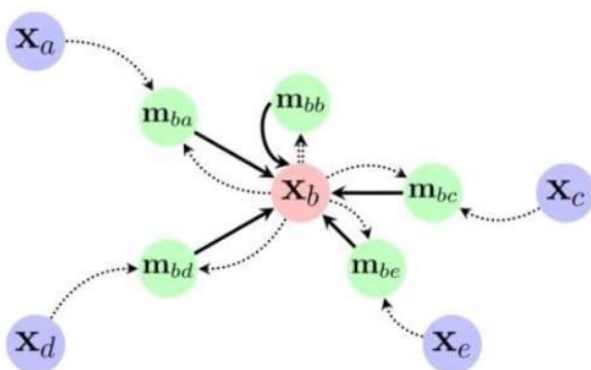
Excited to share a Colab tutorial on graph neural networks with the JAX+Jraph stack, which we use in-house at **DeepMind!**

Developed by **Lisa Wang** and **Nikola Jovanović**.

<https://lnkd.in/dywh7Pwm>

Should be a fantastic resource (theory + code!) for newcomers to either graph representation learning or jraph. 🎉

Many standard models (e.g. GCN / GAT), settings (node classification, graph classification, link prediction) and toy datasets (Karate Club, MUTAG, Cora) are covered.



Graph Neural Networks



Jraph





# ML YouTube Courses

---

At DAIR.AI we ❤️ open education. We are excited to share some of the best and most recent machine learning courses available on YouTube.

### Course List:

- [Stanford CS229: Machine Learning](#)
- [Applied Machine Learning](#)
- [Machine Learning with Graphs \(Stanford\)](#)
- [Probabilistic Machine Learning](#)
- [Introduction to Deep Learning \(MIT\)](#)
- [Deep Learning: CS 182](#)
- [Deep Unsupervised Learning](#)
- [NYU Deep Learning SP21](#)
- [CS224N: Natural Language Processing with Deep Learning](#)
- [CMU Neural Networks for NLP](#)
- [CMU Advanced NLP](#)
- [Multilingual NLP](#)
- [Advanced NLP](#)
- [Deep Learning for Computer Vision](#)
- [Deep Reinforcement Learning](#)
- [Full Stack Deep Learning](#)
- [AMMI Geometric Deep Learning Course \(2021\)](#)





Petar Veličković · 2nd

Staff Research Scientist at DeepMind | Affiliated...  
3d · 🌐

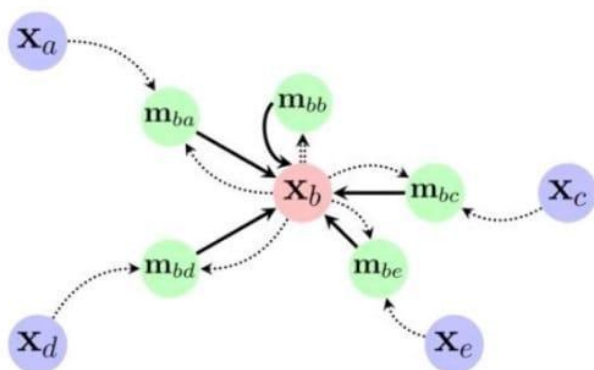
Excited to share a Colab tutorial on graph neural networks with the JAX+Jraph stack, which we use in-house at **DeepMind!**

Developed by **Lisa Wang** and **Nikola Jovanović**.

<https://lnkd.in/dywh7Pwm>

Should be a fantastic resource (theory + code!) for newcomers to either graph representation learning or jraph. 🦄

Many standard models (e.g. GCN / GAT), settings (node classification, graph classification, link prediction) and toy datasets (Karate Club, MUTAG, Cora) are covered.



Graph Neural Networks



Jraph

## **Github: Must Practice this GNN notebook**

[https://github.com/deepmind/educational/blob/f44b133905a96e1887f5909aedcae6811b2c37a3/colabs/summer\\_schools/intro\\_to\\_graph\\_nets\\_tutorial\\_with\\_jraph.ipynb](https://github.com/deepmind/educational/blob/f44b133905a96e1887f5909aedcae6811b2c37a3/colabs/summer_schools/intro_to_graph_nets_tutorial_with_jraph.ipynb)

## **Interview-question-data-science**

<https://github.com/iNeuronai/interview-question-data-science-/tree/master>

## **How to Become Better Listener**

[https://hbr.org/2021/12/how-to-become-a-better-listener?utm\\_source=pocket&utm\\_medium=email&utm\\_campaign=pockethits](https://hbr.org/2021/12/how-to-become-a-better-listener?utm_source=pocket&utm_medium=email&utm_campaign=pockethits)

## Imbalance Training Data

### Imbalance training data



Classifier that always predicts label blue



$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}}$$

yields prediction accuracy of 90%

### Accuracy doesn't always give the correct insight about your trained model

|                                            |                                                     |                              |
|--------------------------------------------|-----------------------------------------------------|------------------------------|
| <b>Accuracy:</b> %age correct prediction   | Correct prediction over total predictions           | One value for entire network |
| <b>Precision:</b> Exactness of model       | From the detected cats, how many were actually cats | Each class/label has a value |
| <b>Recall:</b> Completeness of model       | Correctly detected cats over total cats             | Each class/label has a value |
| <b>F1 Score:</b> Combines Precision/Recall | Harmonic mean of Precision and Recall               | Each class/label has a value |

### Performance metrics associated with Label 1

|                  |   | Actual Labels  |                |
|------------------|---|----------------|----------------|
|                  |   | 1              | 0              |
| Predicted Labels | 1 | True Positive  | False Positive |
|                  | 0 | False Negative | True Negative  |

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{F1 score} = 2 \times \frac{(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})}$$

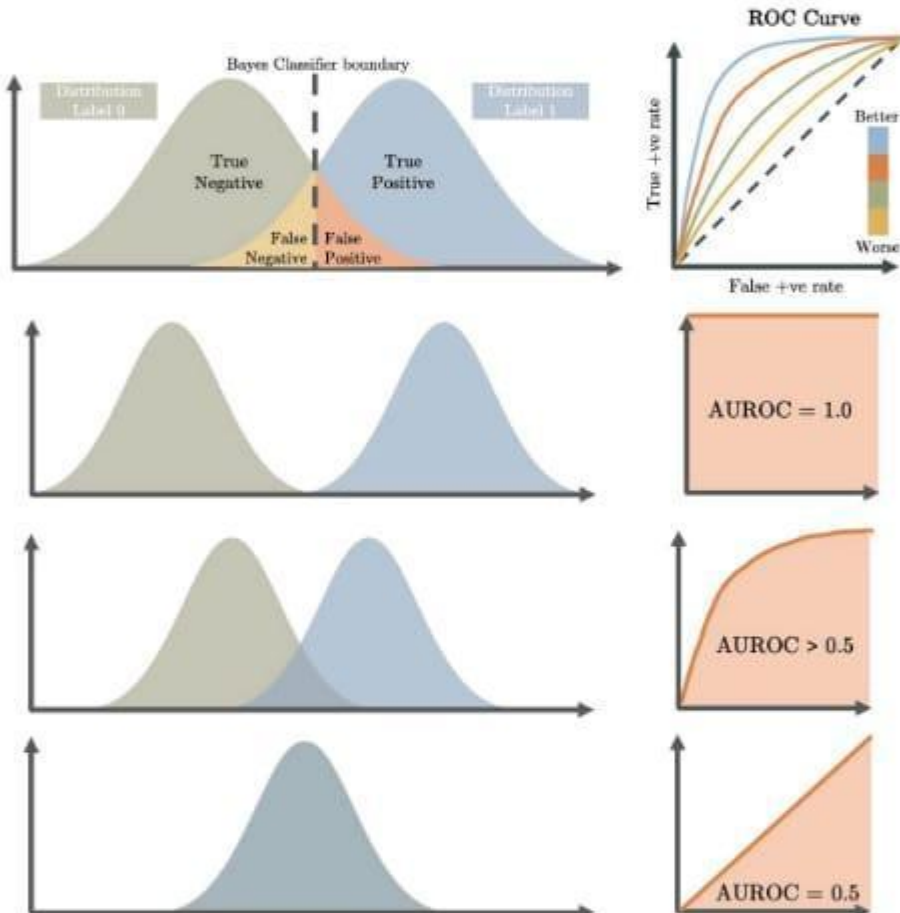
$$\text{Accuracy} = \frac{TP + TN}{TP + FN + FP + TN}$$

$$\text{Recall, Sensitivity} = \frac{TP}{TP + FN} \quad \text{Specificity} = \frac{TN}{TN + FP}$$

$$\frac{FP}{TN + FP} = \text{False +ve rate}$$

(Is your prediction correct?) (What did you predict)

### Receiver Operating Characteristic Explained



### Multi-class Example

Actual Labels

$$\text{Accuracy} = \frac{30+20+1}{30+2+1+1+20+3+4+4+1} = 77.2\%$$

## Introduction to Probability for Data Science

### Intro to Probability for Data Science

[Home](#)

[Author](#)

[Why another probability textbook?](#)

[Endorsement](#)

[Download / Purchase](#)

### Chapters

[Table of Content](#)

[Ch1 Mathematical Background](#)

[Ch2 Probability](#)

[Ch3 Discrete Random Variables](#)

[Ch4 Continuous Random Variables](#)

[Ch5 Joint Distributions](#)

[Ch6 Sample Statistics](#)

[Ch7 Regression](#)

[Ch8 Estimation](#)

[Ch9 Confidence and Hypothesis](#)

[Ch10 Random Processes](#)

### Lecture Videos + Slides

[Lecture Videos + Slides](#)

### Selected Exercise Solutions

[Exercise Ch1](#)

[Exercise Ch2](#)

[Exercise Ch3](#)

[Exercise Ch4](#)

[Exercise Ch5](#)

[Exercise Ch6](#)

[Exercise Ch8](#)

[Exercise Ch10](#)

### Code and Data

[MATLAB, Python, Julia, R Ch1](#)

[MATLAB, Python, Julia, R Ch3](#)

[MATLAB, Python, Julia, R Ch4](#)

[MATLAB, Python, Julia, R Ch5](#)

[MATLAB, Python, Julia, R Ch6](#)

[MATLAB, Python, Julia, R Ch7](#)

[MATLAB, Python, Julia, R Ch8](#)

[MATLAB, Python, Julia, R Ch9](#)

[MATLAB, Python, Julia, R Ch10](#)

[Data](#)

### Purdue Courses

[ECE 302](#)

[ECE 595](#)

## Introduction to Probability for Data Science

Stanley H. Chan

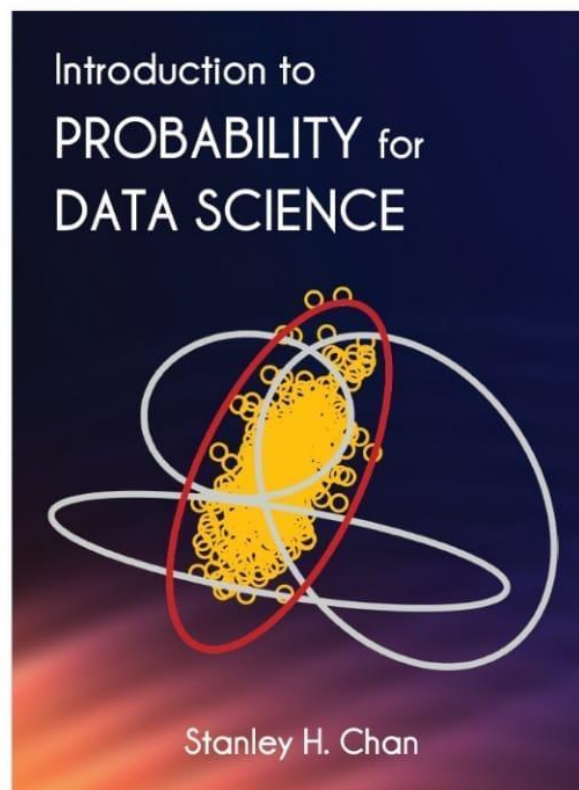
An undergraduate textbook on probability for data science.

Michigan Publishing, 2021

ISBN 978-1-60785-746-4 (hardcover):

[Purchase from Amazon](#)

ISBN 978-1-60785-747-1 (electronic) Free download from Univ. Michigan Publishing



Page generated 2021-12-03 14:35:37 Eastern Standard Time, by jemdoc.

## Data retrieval with SQL, data manipulation MIT Software tool class

Illustrated Data Science study guides covering MIT's 15.003 Software Tools class.

Topics: **data retrieval with SQL**, **data manipulation** with R and Python, tips with Bash and Git.

Web: <https://www.mit.edu/~amidi/teaching/data-science-tools/>

PDF version: <https://github.com/shervinea/mit-15-003-data-science-tools>

## 7 Ways how Data Scientists use Statistics

- Design and interpret experiments to inform product decisions.



- Build models that predict signal, not noise.
- Turn big data into the big picture.
- Understand user engagement, retention, conversion, and leads.
- Give your users what they want.
- Estimate intelligently.
- Tell the story with the data.

### **Google Data Science Interview Questions and Answers**

<https://www.credosystemz.com/google-data-science-interview-questions/>

### **Best Free Machine Learning Books**

<https://www.theinsaneapp.com/2020/11/free-machine-learning-data-science-and-python-books.html>

<https://www.theinsaneapp.com/2020/12/free-data-science-books-pdf.html>

### **Machine Learning Algorithms Cheatsheet**

<https://www.kaggle.com/getting-started/156497>

## Tools for Data sourcing, data wrangling, data application



## Great Leaders Are Confident, Connected, Committed, and Courageous

<https://hbr.org/2018/07/great-leaders-are-confident-connected-committed-and-courageous>

## 25 Github Repositories Every Python Developer Should Know

[https://www.kdnuggets.com/2021/11/25-github-repositories-python-developer.html#.Ychj4FBD\\_VI.linkedin](https://www.kdnuggets.com/2021/11/25-github-repositories-python-developer.html#.Ychj4FBD_VI.linkedin)

## 11 Automatic Machine Learning Frameworks in 2022

<https://www.naijatechguide.com/2021/12/automatic-machine-learning-frameworks.html>

## Data Science Roles, Responsibilities, Tools



© 2018 Maggie Giust

## Debugging in Python

<https://towardsdatascience.com/debug-python-scripts-like-a-pro-78df2f3a9b05>

## TensorFlow Introduces TensorFlow Graph Neural Networks (TF-GNNs)

<https://www.marktechpost.com/2021/11/22/tensorflow-introduces-tensorflow-graph-neural-networks-tf-gnns/>

## India has wasted the potential of its large young population

<https://qz.com/india/2104191/india-has-wasted-the-potential-of-its-large-young-population/>

## The Power of Visualization in Data Science

<https://towardsdatascience.com/the-power-of-visualization-in-data-science-1995d56e4208>

## Text Generation from Knowledge Graphs

Modeling Global and Local Node Contexts for Text Generation from Knowledge Graphs

<https://arxiv.org/abs/2001.11003>

Large improvements from GNN encoder using specialized embeddings for a Knowledge Graph to Text Generation Task. Interesting set of tasks.

GitHub : <https://github.com/UKPLab/kg2text>

## **Evidently AI - Open-Source Machine Learning Monitoring**

<https://evidentlyai.com/>

## **ShaplyAI**

<https://pypi.org/project/Shapely/>

<https://towardsdatascience.com/tagged/shapley-values>

## **BentoML**

<https://www.bentoml.ai/>

## **MLOps Toys \ A Curated List of Machine Learning Projects**

<https://mlops.toys/explainability>

## **Github: AI Fairness 360**

<https://github.com/Trusted-AI/AIF360>

## **Github: AI Fairness 360 Examples**

<https://github.com/IBM/AIF360/blob/master/examples/README.md>

## **Analytics for LinkedIn**

<https://www.shieldapp.ai/>

## **Data Visualization Tools**

<https://gallery.keshif.me/VisTools?s=09>

## **Graph ML 2022**

<https://towardsdatascience.com/graph-ml-in-2022-where-are-we-now-f7f8242599e0>

## **Graph Neural Networks through the lens of Differential Geometry and Algebraic Topology**

<https://towardsdatascience.com/graph-neural-networks-through-the-lens-of-differential-geometry-and-algebraic-topology-3a7c3c22d5f>

## **Data universe**

The data universe comprises data management architectures such as data fabric, data lakehouse, and data mesh.

## **15 Graphs You Need to See to Understand AI in 2021**

<https://spectrum.ieee.org/the-state-of-ai-in-15-graphs>

## **2021's Top Stories About AI - IEEE Spectrum**

<https://spectrum.ieee.org/artificial-intelligence-2021>

## **The Pattern of Success in the Digital Leadership World**

The Pattern of Success in the Digital Leadership World

<https://www.linkedin.com/pulse/pattern-success-digital-leadership-world-jorge-zuazola>

## **Yann LeCun's Deep Learning Course at CDS**

<https://cds.nyu.edu/deep-learning/>

## **The best graph tool**

<https://subhash-achutha.medium.com/best-graph-tools-to-follow-every-data-scientist-and-aspirant-must-know-b3fa44f31a7f>

## **Github: 365 days computer vision**

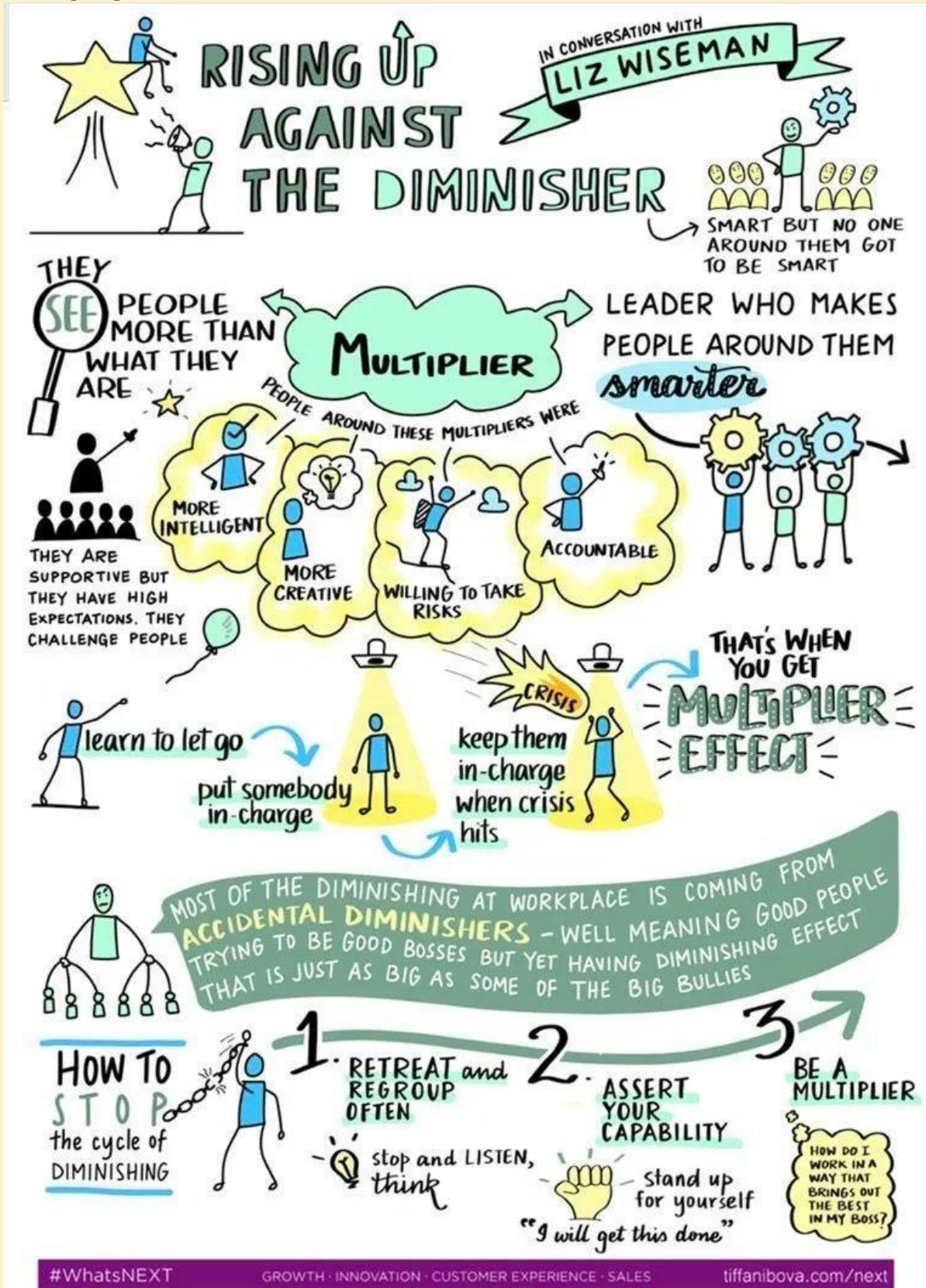
<https://github.com/ashishpatel26/365-Days-Computer-Vision-Learning-Linkedin-Post>

## **Pytorch is a Deep Learning framework**

<https://pub.towardsai.net/pytorch-tutorial-for-beginners-8331afc552c4>



# Rising Against the Diminisher



## Category - Programming Books

<https://www.techprofree.com/category/books/>

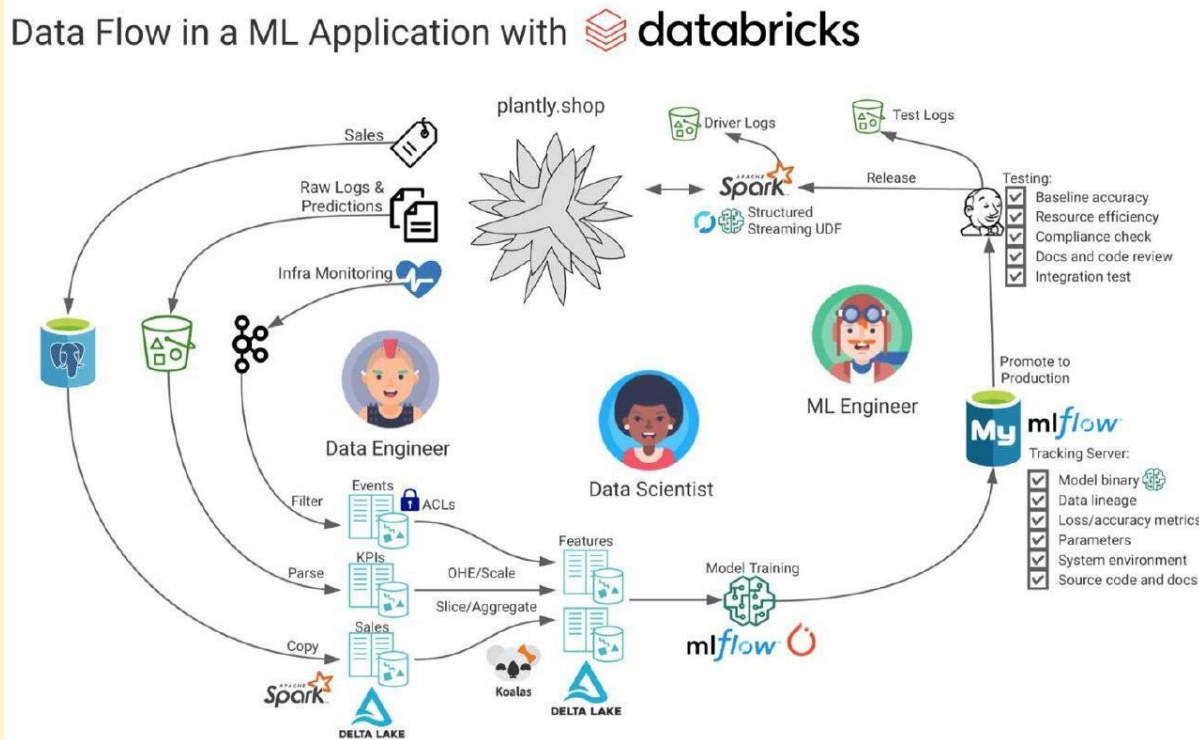
## Designing Event-Driven Systems

[https://www.confluent.io/designing-event-driven-systems/?utm\\_medium=paidsocial&utm\\_source=facebook&utm\\_campaign=ch.paidsocial\\_tp.prs\\_tgt.8pct-LAL-v2-high-scored-content-leads\\_rgn.india\\_lgn.eng\\_con.deds&creative=deds1&fbclid=IwAR1SJI8sF53zbZXNcNXJdHIZ1eQ7MZb6mt1f9RXbufhZB6uZ1wcYRL0ca4k](https://www.confluent.io/designing-event-driven-systems/?utm_medium=paidsocial&utm_source=facebook&utm_campaign=ch.paidsocial_tp.prs_tgt.8pct-LAL-v2-high-scored-content-leads_rgn.india_lgn.eng_con.deds&creative=deds1&fbclid=IwAR1SJI8sF53zbZXNcNXJdHIZ1eQ7MZb6mt1f9RXbufhZB6uZ1wcYRL0ca4k)

## Github: Seldon Core: Blazing Fast, Industry-Ready ML

<https://github.com/SeldonIO/seldon-core>

## Databricks



## Deepfake using conv vision transformer

I am excited to share my paper with Solomon Atnafu on "Deepfake Video Detection Using Convolutional Vision Transformer"

<https://t.co/llzU7YH5hY>

Implementation Code is available on GitHub. (<https://t.co/n0hKGFxiDM>)

## Detection of fake reviews

### Detection of Fake Reviews on Online Review Platforms using Deep Learning Architectures

<https://github.com/ashishsalunkhe/DeepSpamReview-Detection-of-Fake-Reviews-on-Online-Review-Platforms-using-DeepLearning-Architectures>

Dataset: [https://s3.amazonaws.com/fast-ai-nlp/yelp\\_review\\_polarity\\_csv.tgz](https://s3.amazonaws.com/fast-ai-nlp/yelp_review_polarity_csv.tgz)

<https://www.kaggle.com/rtatman/deceptive-opinion-spam-corpus>

The data includes 1,569,264 samples from the Yelp Dataset Challenge 2015. This subset has 280,000 training samples and 19,000 test samples in each polarity.

## GNN Intro

Graph Neural Networks are rapidly advancing progress in machine learning for complex graph data applications.

Here is a nice introduction to GNNs with starter code: <https://t.co/qlzrgp5KTq>

And here is a great summary of Graph ML: <https://t.co/LkZZ4L78Dc>

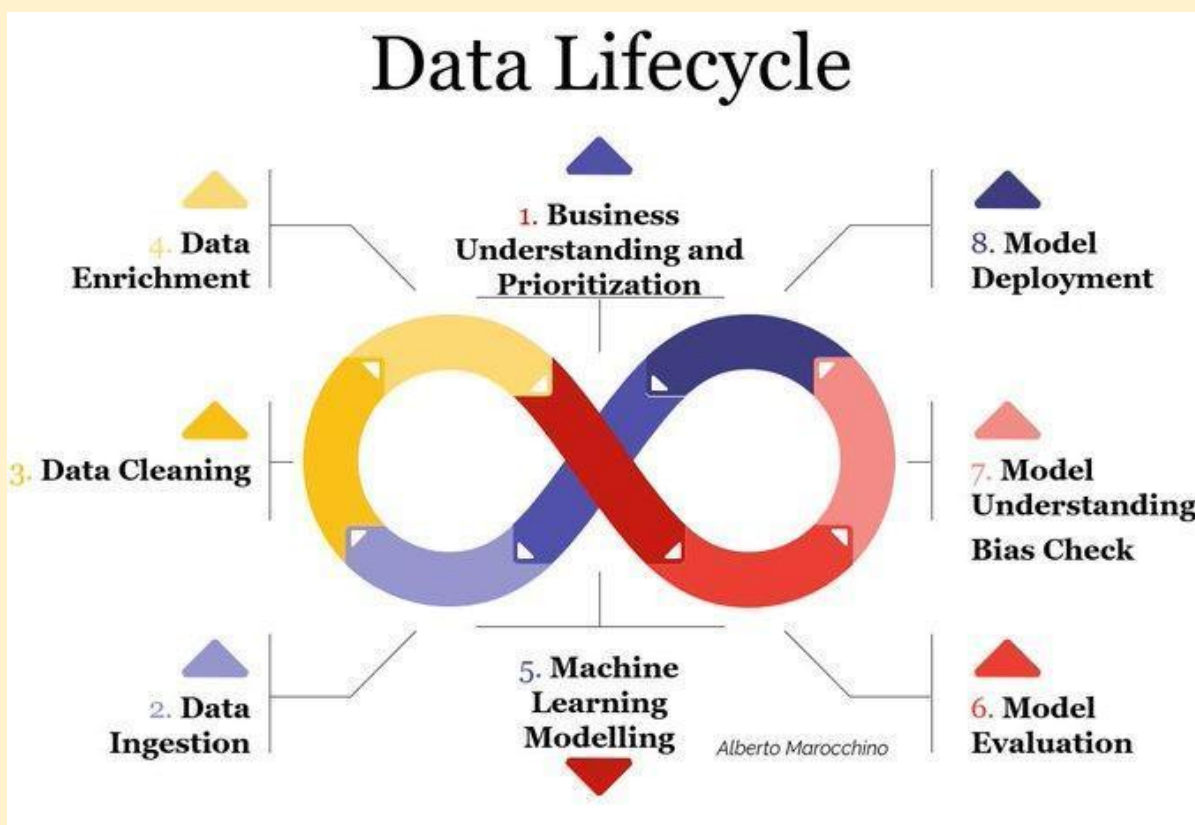
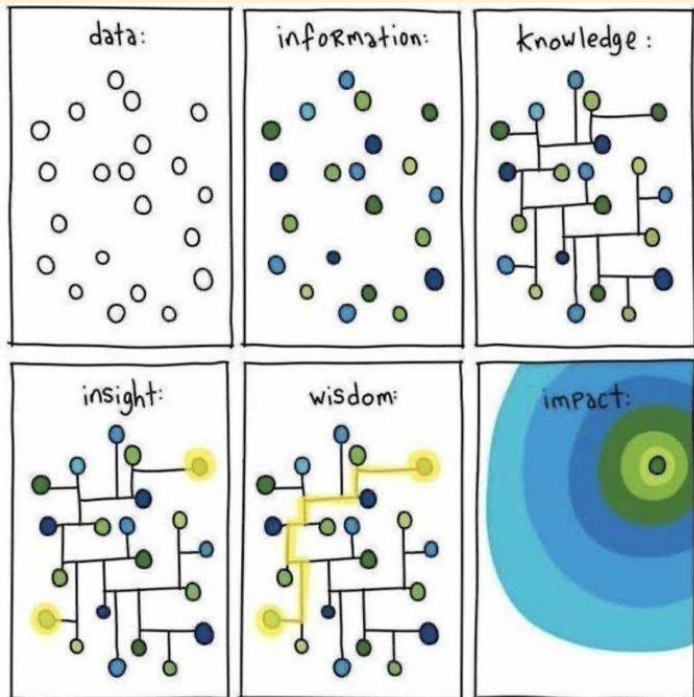
## 50 Best Data Science Project Ideas -2022

<https://www.mltut.com/best-data-science-project-ideas/>

## 60 Best Free Online Courses for Machine Learning & AI in 2022

<https://www.mltut.com/best-free-online-courses-for-machine-learning-and-ai/>

### Data Lifecycle



## Github: HiPlot - High dimensional Interactive Plotting

<https://github.com/facebookresearch/hiplot>



## 6.10. Visualization — Effective Python for Data Scientists

[https://khuyentran1401.github.io/Efficient\\_Python\\_tricks\\_and\\_tools\\_for\\_data\\_scientists/Chapter5/visualization.html#hiplot-high-dimensional-interactive-plotting](https://khuyentran1401.github.io/Efficient_Python_tricks_and_tools_for_data_scientists/Chapter5/visualization.html#hiplot-high-dimensional-interactive-plotting)

### Data Science Practices & Tools

1. Config Management: Hydra, OmegaConf, Pytorch Lightning
2. Checkpoint Management: Pytorch Lightning, TestTube
3. Logging: Tensorboard, Comet.ML, Weights & Biases, MLFlow, Visdom, Neptune
4. Experiment Management: Pytorch Lightning, MLFlow
5. Versioning: Github, Gitlab
6. Data Management: DVC, CML
7. Data analysis: Jupyter Notebook, papermill, JupyterLab, Google Colab
8. Reporting: Matplotlib, Seaborn, Pandas, Overleaf
9. Dependency Management: pip, conda, Poetry, Docker, Singularity, repo2docker
10. Open Source Release: Squash Commits, Binder
11. Effective Communication: ML Code Completeness Checklist, ML Reproducibility Checklist
12. Test and Validate: AWS, GCP, CodeOcean

### Reproducibility in ML

"Reproducibility in Machine Learning: From Theory to Practice", at Deep Learning for Science School, Berkeley <https://sites.google.com/lbl.gov/dl4sci/speakers> (this page has 13 speaker and their recorded talk)

### Week 4 - Reproducibility in Machine Learning: From Theory to Practice - Koustuv Sinha

Video: <https://www.youtube.com/watch?v=se7LNICECqI>

Slides : [https://drive.google.com/file/d/1spuQo08\\_qnq0To-bRIkUFQfu3VhhV4Ww/view](https://drive.google.com/file/d/1spuQo08_qnq0To-bRIkUFQfu3VhhV4Ww/view)

Blog post : [https://www.cs.mcgill.ca/~ksinha4/practices\\_for\\_reproducibility/](https://www.cs.mcgill.ca/~ksinha4/practices_for_reproducibility/)

### Designing, Visualizing, and Understanding Deep Neural Networks

CS W182 / 282A at UC Berkeley \ Designing, Visualizing, and Understanding Deep Neural Networks

<https://youtube.com/playlist?list=PLki3HkfgNEsK1PNSxxPi1YuUKi2cvz5S5>

### YOLO v4 explained in full detail

<https://medium.com/aiguys/yolo-v4-explained-in-full-detail-5200b77aa825>

### Phases OF Dev-Ops

1. Plan
2. Design
3. Code
4. Build
5. Test
6. Package
7. Release
8. Configure
9. Operate
10. Monitor
11. Feedback

### ML TOOLS & TECHNOLOGIES

1. SOURCE CODE MANAGEMENT (GitHub, GitLab, Bitbucket)

2. DATABASE AUTOMATION (Flyway DB, DBMeastro, Redgate DB)
3. CONTINUOUS INTEGRATION (Jenkins, Concourse, Bamboo, Travis, TeamCity, Circle CI)
4. SOFTWARE TESTING (JUnit, JMeter, Selenium)
5. CONFIGURATION MGNT (Chef, Puppet, Ansible, Packer, Saltstack)
6. DEPLOYMENT: (Automic, UrbanCode, Octopus Deploy, Electric Cloud)
7. CONTAINER & ORCHESTRATION (Helm, Docker, Kubernetes, Mesos)
8. RELEASE ORCHESTRATION: (Spinnaker, Plutora, UrbanCode, XL Release, OpenMake)
9. CLOUD PLATFORMS: (AWS, Azure, GCP, Oracle, IBM Cloud, Cloud Foundry, OpenStack)
10. ARTIFICIAL INTELLIGENCE: (Splunk, Fluentd, Logstash, Prometheus, SumoLogic)
11. ANALYTICS: (DatadogHQ, AppDynamics, Dynatrace, New Relic, Kana, ElasticSearch, ExtraHop)
12. MONITORING & OBSERVATIONS: (Nagios, Zenoss, AppDynamics, Dynatrace, Zabbix)
13. SECURITY & SECRETS MANAGEMENT (SonarQube, HashCorp Vault, CyberArk Conjur, Fortify, TripWire, Blackduck)
14. COLLABORATION & COMMUNICATION (Slack, Trello, PagerDuty, AlertOps)
15. ARTIFACT REPOSITORY Page 4 (Frog Artifactory, Nexus Sonatype)
16. PROJECT & ISSUE MANAGEMENT: (ServiceNow, JIRA)
17. SOFTWARE DESIGN & DOCS (VisualParadigm, LucidSpark, LucidCharts, Gliffy)
18. API & MICRO-SERVICES (KongHQ, Stoplight, Swagger, Postman, Tyk, OpsLevel)
19. MESSAGING & REPLICATION (Kafka, ActiveMQ)
20. CHOAS TESTING (ChoasMonkey, Gremlin)
21. FUNCTIONS AS A SERVICE (GCP Engine, Azure Functions, AWS Lambdas)
22. BIG-DATA TOOLS: (Apache Hadoop, Apache Spark, Apache Solr)
23. NON RELATIONAL DATA STORAGE (Cassandra, MongoDB, RedisDB, Neo4J)
24. RELATIONAL DATA STORAGE (DBVisualizer, EnterpriseDB, Oracle DB, PostgreSQL DB)
25. CONTINUOUS DELIVERY PIPELINE (Harness, Plutora)
26. INTEGRATED DEV ENVIRONMENTS (Eclipse IDE, DataGrip, IntelliJ IDEA....)
27. PACKAGE MANAGEMENT (Npm, Yum, Maven, RubyGems, CPAN, Yarn)
28. APPLICATION ORCHESTRATION (Keptn, Redhat Automation)
29. OPERATING SYSTEMS (Linux, Unix, Fedora)
30. SCRIPTING LANGUAGES (Perl, Python, Ruby)
31. WEB, APP SERVERS : (Nginx, Nutanix, Tomcat, JBoss Server, Oracle WebLogic Server)

#developer #software #softwarequalityassurance #cicd #devops #devopsworld #devopstools  
#devopsculture #testautomation #softwaretesting #observability #configurationmanagement #automation  
#jenkins #gitlab #versioncontrol #integrationtesting #softwaresecurity #database #cloud #github #docker







**Moez Ali** · Following

from `pycaret.classification import *`

1d · Edited · 🌐

🚀 Now you can create a web API of your entire ML Pipeline right from your Notebook. Have you ever seen this level of abstraction before?

A new function `create_api` is added to **PyCaret 2.3.6** (coming soon). This function takes the entire ML pipeline and deploys it as a web API using the FastAPI framework, right from your notebook.

Look at this:

```
>>> s = setup(data, target, session_id = 123)
>>> best_model = compare_models(sort='MAE')
>>> create_api(best_model, 'my_first_api')
>>> !python my_first_api.py
```

That's it. Now you can dockerize this notebook for your production. 🤖

Watch this demo 🖱️

**#datascience #machinelearning #opensource  
#python #pycaret #fastapi**

The screenshot shows a Jupyter Notebook interface with the following code in the first cell:

```
In [1]: 1 from pycaret.datasets import get_data
        2 data = get_data('boston')
```

The output of the code is a table with 15 columns and 5 rows of data:

|   | crim    | zn   | indus | chas | nox   | rm    | age  | dis    | rad | tax | pbratio | black  | lstat | medv |
|---|---------|------|-------|------|-------|-------|------|--------|-----|-----|---------|--------|-------|------|
| 0 | 0.00632 | 18.0 | 2.31  | 0    | 0.538 | 6.575 | 65.2 | 4.0900 | 1   | 296 | 15.3    | 396.90 | 4.98  | 24.0 |
| 1 | 0.02731 | 0.0  | 7.07  | 0    | 0.469 | 6.421 | 78.9 | 4.9671 | 2   | 242 | 17.8    | 396.90 | 9.14  | 21.0 |
| 2 | 0.02729 | 0.0  | 7.07  | 0    | 0.469 | 7.185 | 61.1 | 4.9671 | 2   | 242 | 17.8    | 392.63 | 4.03  | 34.7 |
| 3 | 0.03237 | 0.0  | 2.18  | 0    | 0.458 | 6.998 | 45.8 | 6.0622 | 3   | 222 | 18.7    | 394.63 | 2.94  | 33.4 |
| 4 | 0.06905 | 0.0  | 2.18  | 0    | 0.458 | 7.147 | 54.2 | 6.0622 | 3   | 222 | 18.7    | 396.90 | 5.33  | 36.2 |

## 18 Data Science Podcasts

<https://online.datasciencedojo.com/blogs/the-18-best-data-science-podcasts-on-soundcloud-apple-podcast-and-spotify>

**Data Science Dojo**  
149,467 followers  
2h • 🌐

🌟 Start your new year with these 18 podcasts that can help you to begin your data science journey in 2022: <https://hubs.la/Q0118Npv0>

#DataScience #Podcasts #DSDojo

**18 BEST DATA SCIENCE PODCASTS**

- Data Skeptic
- Data Science Ethics
- Data Science at Home
- Data Crunch
- SuperData Science
- Data Cast
- Data Science Imposters
- Women in Data Science
- Linear Digressions
- Data Futurology
- Data Science in Production
- Cyentia Podcast
- Talk Python To Me
- O'Reilly Radar - The O'Reilly Data Show Podcast
- The AI Podcast
- DATAHACK RADIO
- Half Stack Data Science
- This Week in Machine Learning & AI (TWIML & AI)

datasciencedojo  
data science for everyone

### Data Science Dojo

1. Data Skeptic Podcast
2. Data Science Imposters Podcast
3. Data Science at Home Podcast
4. Data Crunch Podcast
5. SuperData Science Podcast
6. Data Cast Podcast
7. Data Science Ethics Podcast
8. Women in Data Science Podcast
9. Linear Digressions Podcast
10. Data Futurology Podcast

11. Data Science in Production Podcast
12. Cyentia Podcast Podcast
13. Talk Python To Me Podcast
14. O'Reilly Radar - The O'Reilly Data Show Podcast
15. The AI Podcast
16. DATAHACK RADIO Podcast
17. Half Stack Data Science Podcast
18. This Week in Machine Learning & AI (TWIML & AI) Podcast

### **Effective Testing for Machine Learning (Part I)**

<https://towardsdatascience.com/effective-testing-for-machine-learning-part-i-e6b5aeb51421>

### **FREE DS MASTER CLASS SERIES (PYTHON)**

**PANTECH-Free Online Webinar**

<https://docs.google.com/forms/d/e/1FAIpQLSfbyy-L47QcdODPFLFC4GGFBvmVIUBy3mC0oBqIhJAXZEteq/w/viewform>

### **Simple Explanation of LSTM \ Deep Learning Tutorial 36 (Tensorflow, Keras & Python)**

<https://youtu.be/LfnrRPFhkuY>

### **Covid19 forecasting**

<https://medium.data4sci.com/covid-19-forecasting-the-death-toll-1c3b284537fc>

### **COVID-19: Face Mask Detection using TensorFlow and OpenCV**

<https://towardsdatascience.com/covid-19-face-mask-detection-using-tensorflow-and-opencv-702dd833515b>

<https://towardsdatascience.com/creating-a-powerful-covid-19-mask-detection-tool-with-pytorch-d961b31dcd45>

**IJERPH \ Free Full-Text \ Path Analysis to Assess Socio-Economic and Mitigation Measure Determinants for Daily Coronavirus Infections \ HTML**

<https://www.mdpi.com/1660-4601/18/19/10071/htm>



# THE 14 BEST PYTHON LIBRARIES FOR CYBERSECURITY



1

  
**pandas**  
**Pandas**

- Natural language processing
- Data analysis
- Data visualization
- Malware analysis

2

  
**NumPy**  
**NumPy**

- Natural language processing
- Data analysis
- Data visualization
- Malware analysis

3

  
**NLTK**  
**NLTK**

- Natural language processing
- Data analysis
- Data visualization
- Malware analysis

4

  
**Scikit**  
**Scikit**

- Implementing machine learning
- Machine learning algorithms

5

  
**NMap**  
**NMap**

- Scanning and implementing transport layer and application layer protocols

6



**Twisted**  
**Twisted**

- Scanning and implementing transport layer and application layer protocols

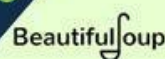
7



**Scapy**  
**Scapy**

- Packet processing
- Decoding packets
- Forging packets
- Analyzing packet information

8

  
**Beautiful Soup**  
**Beautiful Soup**

- Scraping data from web pages

9



**Cryptography Library**  
**Cryptography Library**

- Implementing cryptographic algorithms

10



**YARA**  
**YARA**

- Malware identification
- Malware classification

11



**Pymetasploit3**  
**Pymetasploit3**

- Discovers hidden vulnerabilities

12



**Mechanize**  
**Mechanize**

- Data acquisition
- Interacting with web pages
- Checking SQL injection
- Cross site scripting

13



**Socket**  
**Socket**

- Socket programming
- Port scanning
- Sniffing

14



**Requests**  
**Requests**

- API calling
- Grabbing HTML web pages

**Q:** Do cybersecurity professionals need to know Python?

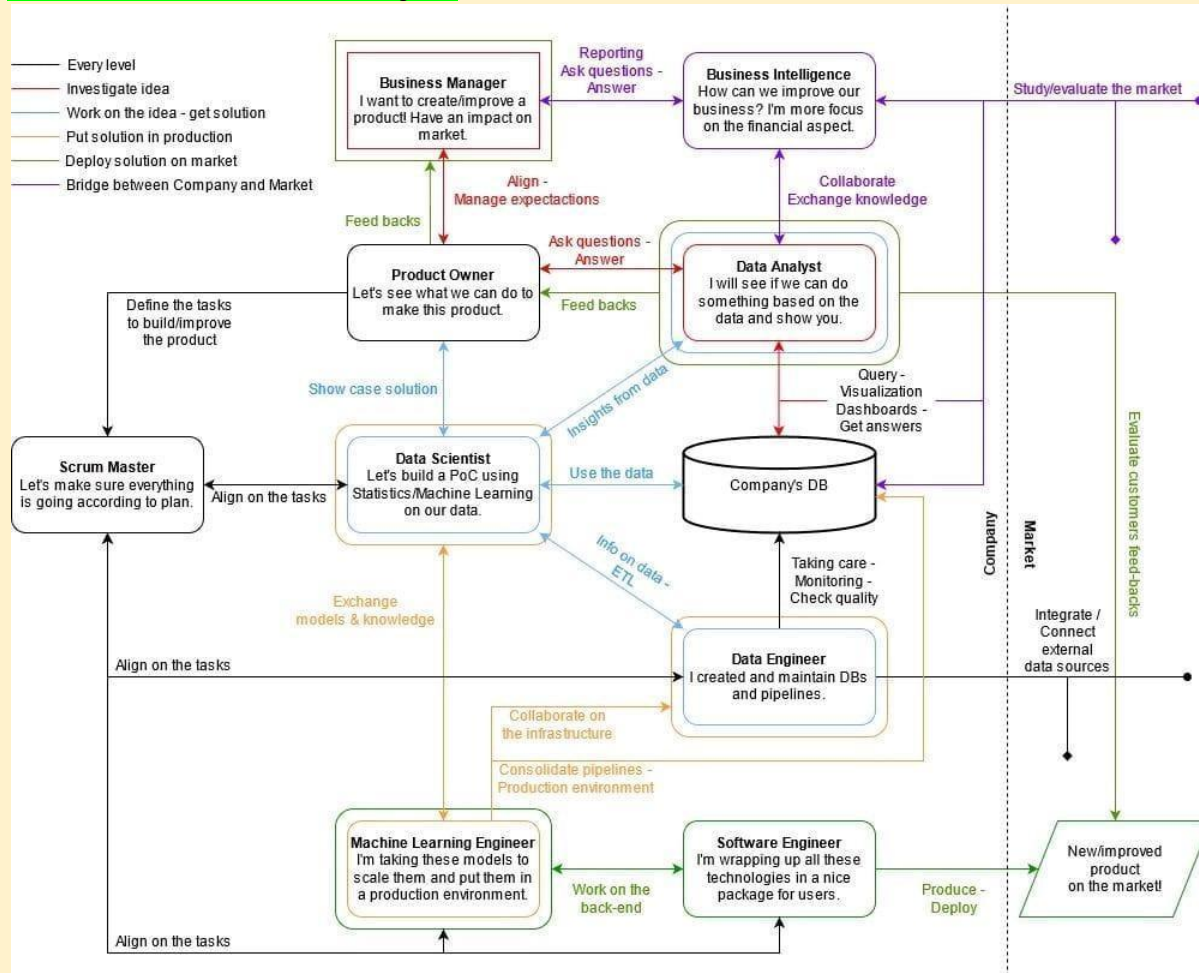


**A:** "It's not *mandatory* to know Python, but knowledge of Python opens many career doors. Having an understanding of Python helps a cybersecurity professional in interviews, leading to a higher salary, more responsibility, and future opportunity."

• Aastha Sahni, Flatiron School Instructor



## Roles in Data Science Project



## Computer vision datasets: VisualData Discovery

DatAndroid is a benchmark dataset for Visual GUI Testing (VGT) on Android. VGT techniques are based on image recognition that is able to identify the same widgets rendered across multiple devices.

Dataset: <https://visualdata.io/discovery/dataset/435c6084ab3051612f7aa36cb86ccc23be7c8e55>

Number of datasets: 840

## DatAndroid Dataset

DatAndroid is a benchmark dataset dedicated to the development and assessment of Visual GUI Testing (VGT) techniques in the Android domain.

<https://zenodo.org/record/4605429#.Yv91GnZBw5s>

The dataset contains 95 apps downloaded from [Uptodown](https://uptodown.com) emulated on 14 different devices

## Pandas online learning

<https://learning.oreilly.com/library/view/pandas-1x-cookbook/9781839213106/>

## MongoDB Fundamentals Course

<https://learndataengineering.com/p/mongodb-fundamentals-course>

## GroundedML — ICLR 2022

<https://sites.google.com/view/groundedml2022?s=09>

## PyTorch vs. TensorFlow: Which Framework Is Best for Your Deep Learning Project?

<https://builtin.com/data-science/pytorch-vs-tensorflow>

## **MLOps \ Bring DevOps To Data Science With MLOps**

<https://www.analyticsvidhya.com/blog/2021/04/bring-devops-to-data-science-with-continuous-mlops/>

## Data science project life cycle

# DATA SCIENCE

## THE LIFECYCLE

### BUSINESS UNDERSTANDING

As in any project irrespective of whether it is data science or not, understanding of business is crucial. This is where you define a business problem, identify the key metrics (Key Performance Indicators) and success criteria's. Everything that follows in subsequent stages are means to address and solve this problem.

### EDA & VISUALIZATION

In this stage we try to identify the relation between variables, catch outliers, apply statistical methods to find patterns and in the end - visualize our findings with a storyline. It is highly analytical and at the same time demands creativity in visualizing the complex data insights.

### MODEL BUILDING

Once data pre-processing is completed, create a baseline model. We then gradually increase the complexity, apply various algorithms, tune the parameters to find the right mix and make sure the model is not overfitted.

### DATA AQUISION

Once we understand the business and its objectives. We need to understand the data sources, its quality, relevance to our problem statement and other needs.

### FEATURE ENGINEERING & EXTRACTION

This stage involves application of domain knowledge and techniques to transform data into a form where models predictive ability is increases. E.g.: categorical encoding, log transforms, scaling and power functions

### MODEL DEPLOYMENT

Model are deployed to in order to use it for practical purpose. Generally models are deployment on cloud and in some case on-premise based on business requirements

### MODEL

# DEVOPS

## THE LIFECYCLE

### CONTINUOUS DEVELOPMENT

The first phase where the planning and software development takes place

### CONTINUOUS TESTING

The developed functionality is continuously tested for bugs. The test environments are simulated using docker containers.

### CONTINUOUS DEPLOYMENT

Finalized application code is deployed to the production servers. The consistency in the application's performance and functional conditions is established and curated.

### CONTINUOUS INTEGRATION

All new codes that support add-on functionalities are built and integrated into the existing code

### CONTINUOUS FEEDBACK

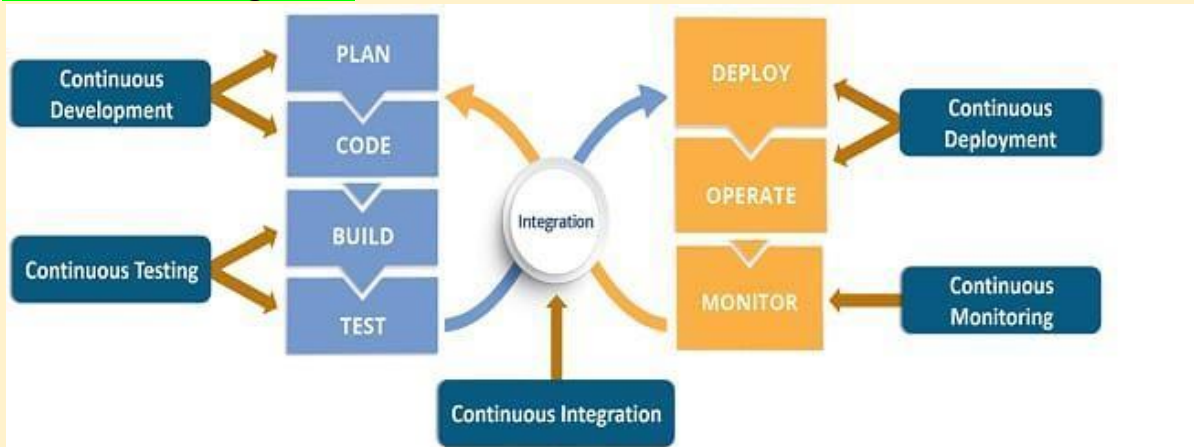
A phase for accomodating the feedback for continuous improvement in code ad also crucial phase for releasing newer version of the software

### CONTINUOUS MONITORING

For sustaining the availability of services in the application. It also helps in tracking any unusual system behavior



## Continuous integration



## 3 ways to use data, analytics, and machine learning in test automation

<https://www.infoworld.com/article/3636217/3-ways-to-use-data-analytics-and-machine-learning-in-test-automation.html>

## Automated Functional Testing - Software Testing Tool

<https://www.testim.io/>

## Using sub graph for more expressive GNN

<https://towardsdatascience.com/using-subgraphs-for-more-expressive-gnns-8d06418d5ab>

## HOW AI & MACHINE LEARNING IS CHANGING TEST AUTOMATION

<https://youtu.be/76oMGqjJQkU>

## NanoEdge AI Studio

NanoEdge AI Studio - V3 Released - Download Free Trial - STMicroelectronics

[https://www.st.com/content/st\\_com/en/campaigns/nanoedgeaistudio.html](https://www.st.com/content/st_com/en/campaigns/nanoedgeaistudio.html)

## How I Built an ML Algorithm to Improve Test Automation

<https://www.perfecto.io/blog/ml-in-testing-automation>

## AI and ML in Testing: X Tips To Make Test Automation Effective

<https://axiomq.com/blog/ai-and-ml-in-testing-x-tips-to-make-test-automation-effective/>

## KerGNNs: Interpretable Graph Neural Networks with Graph Kernels

<https://arxiv.org/abs/2201.00491>

## TOP 10 GitHub Repositories for Data Science

<https://www.analyticsvidhya.com/blog/2022/01/top-10-github-repositories-for-data-science/>

## Real-time machine learning: challenges and solutions

<https://huyenchip.com/2022/01/02/real-time-machine-learning-challenges-and-solutions.html>

## **How to Read ML Papers Easily**

<https://towardsdatascience.com/how-to-read-machine-learning-papers-easily-2555deb78d80>

## **Rulebased chatbot in Python**

Learn about How to Make a Rule-based Chatbot in Python using Flask.

Building an AI-based chatbot is very easy nowadays due to the rise in advanced Natural languaging processing techniques.

<https://pythonscholar.com/python-projects/rule-based-chatbot-in-python-using-flask/>

## **Forecasting: Principles and Practice**

<https://otexts.com/fpp2/hierarchical.html/>

## **Chapter 10 Forecasting hierarchical or grouped time series**

Forecasting: Principles and Practice (2nd ed)

<https://otexts.com/fpp2/hierarchical.html>

## **Will we see GPT-3 moment for computer vision?**

<https://analyticsindiamag.com/will-we-see-gpt-3-moment-for-computer-vision/>

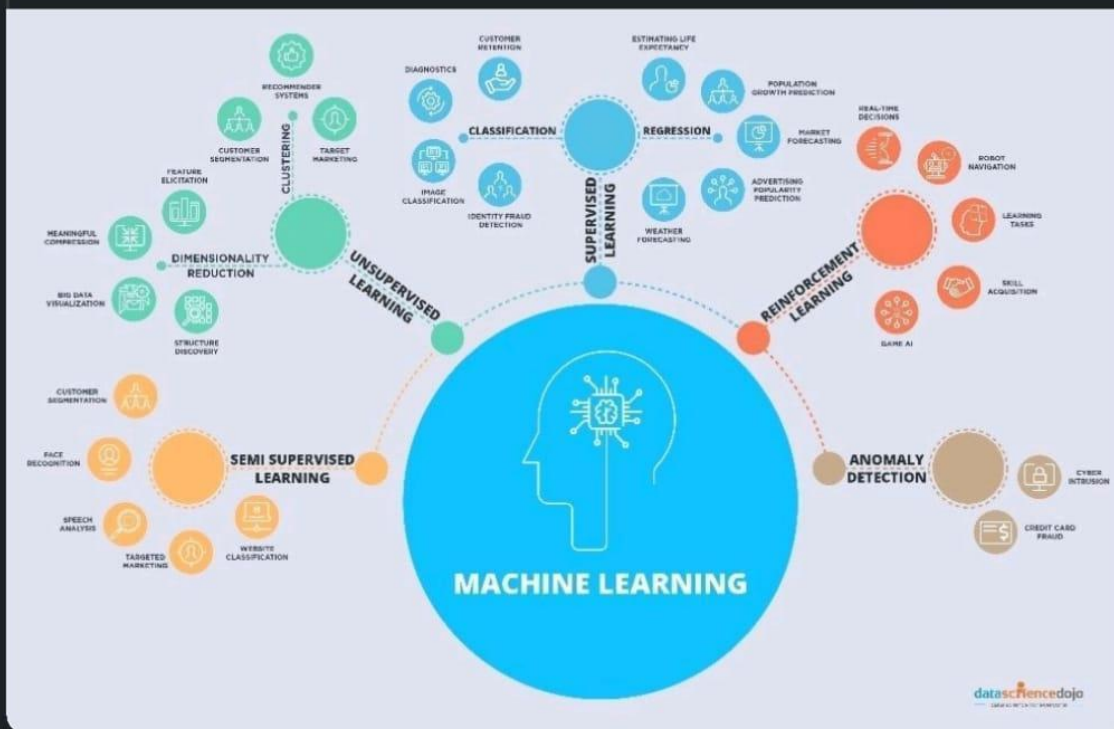
# Top 5 best #MachineLearning #Innovations in 2022.

<https://bit.ly/3HKyBeM> via Giuliano Liguori

- ◆ Tiny ML
- ◆ Quantum ML
- ◆ Auto ML
- ◆ MLOps
- ◆ Full stack deep learning

What do you think?

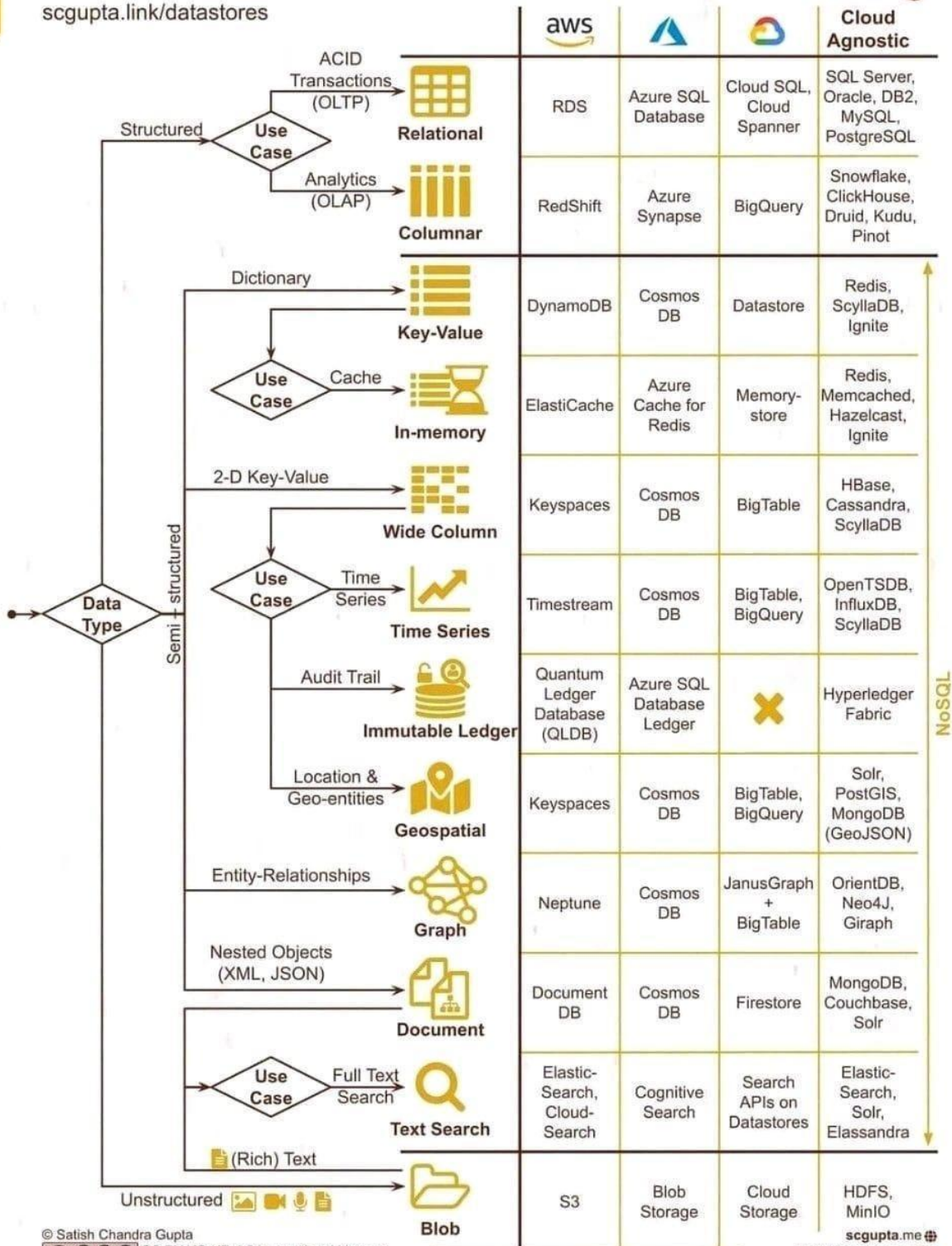
#innovation #Python #AI #PyTorch #DeepLearning  
#DataScience #MLOps #DigitalTransformation  
#technology



# SQL vs NoSQL Tools for Cloud

## SQL vs. NoSQL: Cheatsheet for AWS, Azure, and Google Cloud

scgupta.link/datastores



© Satish Chandra Gupta  
 CC BY-NC-ND 4.0 International Licence  
[creativecommons.org/licenses/by-nc-nd/4.0/](https://creativecommons.org/licenses/by-nc-nd/4.0/)

scgupta.me  
 twitter.com/scgupta  
 linkedin.com/in/scgupta

### Free AI Introductory Course For All

<https://www.marktechpost.com/free-ai-introductory-course-for-all/>

### Stanford CS224U Natural Language Understanding Spring 2021

[https://youtube.com/playlist?list=PLoROMvodv4rPt5D0zs3YhbWSZA8Q\\_DyiJ](https://youtube.com/playlist?list=PLoROMvodv4rPt5D0zs3YhbWSZA8Q_DyiJ)

## **Grouped multivariate and functional time series forecasting:**

An application to annuity pricing. <https://arxiv.org/abs/1705.08001>

## **Cloud security assessment checklist**

<https://relevant.software/blog/cloud-security-assessment/>



# Python Cheat Sheet

## Math

### Number Theoretic

ceil(x) copysign(x,y)  
fabs(x) factorial(x)  
floor(x) fmod(x,y)  
frexp(x) fsum(iterable)  
lrint(x) isnan(x)  
ldexp(x,i) modf() trunc()

### Power & Logarithmic

exp(x) log(x, base)  
log1p(x) log10(x)  
pow(x,y) sqrt(x)

### Trigonometric Functions

acos(x) asin(x) atan(x)  
atan2(y,x) cos(x) hypot(x,y)  
sin(x) tan(x)

### Angular Conversion

degrees(x) radians(x)

### Hyperbolic Functions

acosh(x) asinh(x)  
atanh(x) cosh(x)  
sinh(x) tanh(x)

### Constants

#### math.pi

The mathematical constant of pi = 3.141592... up to the available precision

#### math.e

The mathematical constant e = 2.718281... up to the available precision

## String Formatting

### Formatting Operations

'd' Signed integer decimal 'i' Signed integer decimal 'o' Signed octal value 'u' Obsolete type - it was identical to 'd'  
'x' Signed hexadecimal (lowercase) 'X' Signed hexadecimal (uppercase) 'e' Floating point exponential format (lowercase)  
'E' Floating point exponential format (uppercase) 'f' Floating point decimal format 'F' Floating point decimal format  
'g' Floating point format. Uses the lowercase exponential format if the exponent is less than -4 or not less than precision, otherwise it uses the decimal format  
'G' Floating point format. Uses the uppercase exponential format if the exponent is less than -4 or not less than precision, otherwise it uses the decimal format  
'c' Single character (accepts either integer or single character string) 'r' String (converts any Python object using repr())  
's' String (converts any Python object using str()) '%' No argument is converted, adds a % character in the end result

## File

### Methods

close() flush() fileno()  
isatty() next() read([size])  
readline([size]) readlines([size])  
xreadlines() seek(offset[, whence])  
tell() truncate([size])  
write(str) writelines(sequence)

### Attributes

closed encoding  
errors mode  
name newlines  
softspace

## Class

### Special Methods

\_\_new\_\_(cls) \_\_lt\_\_(self, other) \_\_init\_\_(self, args)  
\_\_le\_\_(self, other) \_\_del\_\_(self) \_\_gt\_\_(self, other)  
\_\_repr\_\_(self) \_\_ge\_\_(self, other) \_\_str\_\_(self)  
\_\_eq\_\_(self, other) \_\_cmp\_\_(self, other)  
\_\_ne\_\_(self, other) \_\_index\_\_(self) \_\_nonzero\_\_(self)  
\_\_hash\_\_(self) \_\_getattr\_\_(self, name)  
\_\_getattribute\_\_(self, name) \_\_setattr\_\_(self, name, attr)  
\_\_delattr\_\_(self, name) \_\_call\_\_(self, args, kwargs)

## Random

### Functions

seed([x]) getstate() vonmisesvariate(mu, kappa)  
setstate(state) jumpahead(n) paretovariate(alpha)  
getrandbits(k) randint(a,b) weibullvariate(alpha,beta)  
randrange([start, stop[, step]]) lognormvariate(mu, sigma)  
choice(seq) shuffle(x, random) normalvariate(mu, sigma)  
sample(population, k) random() gammavariate(alpha, beta)  
uniform(a, b) triangular(low, high, mode) gauss(mu, sigma)  
betavariate(alpha, beta) expovariate(lambda)

## Array

### Array Methods

append(x) buffer\_info()  
byteswap() count(x)  
extend(iterable) fromfile(f,n)  
fromlist(list) fromstring(s)  
fromunicode(s) index(x)  
insert(i,x) pop([i]) remove(x)  
reverse() tofile(f) tolist()  
tostring() tounicode()

### Indexes & Slices

a=[0,1,2,3,4,5]  
b=a[:] Shallow copy of a  
a[1:] [1,2,3,4,5]  
a[5:] [0,1,2,3,4]  
a[-2:] [0,1,2,3] len(a) 6  
a[1:3] [1,2] a[0] 0  
a[-1:-1] [1,2,3,4] a[5] 5  
a[-1] 5  
a[-2] 4

## OS

### OS Variables

altsep Alternative separator  
curdir Current dir string  
defpath Default search path  
devnull Path of null device  
extsep Extension separator  
pardir Parent dir string  
pathsep Patch separator  
sep Path separator  
name name of OS  
linesep Line separator

## SYS

### SYS Variables

argv Command line args  
builtin\_module\_names Linked C modules  
check\_interval Signal check frequency  
exec\_prefix Root directory  
executable Name of Executable  
exitfunc Exit function name  
modules Loaded modules  
path Search path

### platform

Current platform  
stdin, stdout, stderr File objects for I/O  
version\_info Python version info  
winver Version number

### SYS Arg V

sys.argv[0] foo.py  
sys.argv[1] bar  
sys.argv[2] -c  
sys.argv[3] qux  
sys.argv[4] -h

## String

### String Methods

capitalize() center(width[, fillchar]) count(sub[, start, end])  
decode encode([encoding[, errors]]) isalnum()  
endswith(suffix[, start[, end]]) expandtabs([tabsize])  
find(sub[, start[, end]]) format(\*args, \*\*kwargs) isalpha()  
index(sub[, start[, end]]) isdigit() islower() isspace() istitle()  
isupper() join(iterable) ljust(width[, fillchar]) lower()  
lstrip([chars]) partition(sep) replace(old, new[, count])  
rfind(sub[, start[, end]]) rindex(sub[, start[, end]])  
rjust(width[, fillchar]) rpartition(sep) rsplit([sep, maxsplit])  
rstrip([chars]) split([sep[, maxsplit]]) splitlines([keepends])  
startswith(prefix[, start[, end]]) strip([chars]). swapcase(), title()  
translate(table[, deletechars]), upper() zfill(width)  
isnumeric() isdecimal()

## Set & Mapping

### Mapping Types

len(d) d[key] d[key]=value  
del d[key] key in d key not in d  
iter(d) clear() copy() items()  
fromkeys(seq[, value]) keys()  
get(key[, default]) has\_key(key)  
iteritems() iterkeys()  
interval() popitem()  
pop(key[, default])  
setdefault(key[, default])  
update([other])  
values

### Set Types

len(s) x in s x not in s isdisjoint(other)  
issubset(others) issuperset union(other...)  
intersection(other...) difference(other...)  
symmetric\_difference(other) copy() update()  
intersection\_update() difference\_update()  
symmetric\_difference\_update() add(elem)  
remove() discard(elem) pop() clear()

## Date Time

### Date Object

replace(year, month, day) timetuple()  
toordinal() weekday() isoweekday()  
isocalendar() isoformat() \_\_str\_\_  
ctime() strftime()

### Time Object

## Date Formatting

%a Abbreviated weekday (Mon) %A Weekday (Monday)  
%b Abbreviated month name (Nov) %B Month name (November)

# Neural Network Types

| Network Type               | Architecture                                    | Network Model                             | Training Type               | Training Algorithm                              | Implementation Sample                                                  | Common Application                                                           | Popular Dataset Sample | DL Framework (sample)                                                                                |
|----------------------------|-------------------------------------------------|-------------------------------------------|-----------------------------|-------------------------------------------------|------------------------------------------------------------------------|------------------------------------------------------------------------------|------------------------|------------------------------------------------------------------------------------------------------|
| Feedforward Neural Network | CNN                                             | Discriminative                            | Supervised                  | Gradient Descent based Backpropagation          | Siamese Network, Deep CNN                                              | Image recognition/classification                                             | MNIST                  | TensorFlow, Caffe, Theano, Torch, Deeplearning4j, Microsoft Cognitive Toolkit, Keras, MXNet, PyTorch |
|                            | Residual Network                                | Discriminative                            | Supervised                  | Gradient Descent based Backpropagation          | Deep ResNet; HighwayNet; DenseNet                                      | Image recognition                                                            | ImageNet               | TensorFlow, PyTorch, Keras                                                                           |
|                            | Autoencoder                                     | Generative                                | Unsupervised                | Backpropagation                                 | Sparse Autoencoders, Variational Autoencoders                          | Dimensionality Reduction; Encoding                                           | MNIST                  | TensorFlow, Deeplearning4j, Keras                                                                    |
|                            | Adversarial Networks                            | Generative & Discriminative               | Unsupervised                | Backpropagation                                 | Generative Adversarial Network                                         | Generate realistic fake data; Reconstruction of 3D models; Image improvement | CIFAR10                | TensorFlow, Keras                                                                                    |
|                            | RBM                                             | Generative with Discriminative finetuning | Unsupervised                | Gradient Descent based Contrastive divergence   | Deep Belief Network; Deep Boltzmann Machine                            | Dimensionality Reduction; Feature learning; Topic modeling                   | MNIST                  | TensorFlow, Deeplearning4j, Keras, MXNet, Theano, Torch                                              |
| Recurrent Neural Network   | LSTM                                            | Discriminative                            | Supervised                  | Gradient Descent & Backpropagation through Time | Deep RNN, Gated Recurrent Unit (GRU), Neural Machine Translation (NMT) | Natural Language Processing; Language Translation                            | MNIST Stroke Sequence  | TensorFlow, Caffe, Theano, Torch, Deeplearning4j, Microsoft Cognitive Toolkit, Keras, MXNet, PyTorch |
| Radial Basis Function NN   | RBF Network                                     | Discriminative                            | Supervised and Unsupervised | K-means Clustering; Least Square Function       | Radial Basis Function NN                                               | Function approximation ; Time series prediction                              | Fisher's Iris data set | TensorFlow                                                                                           |
| Kohonen Self Organizing NN | Nodes arranged in hexagonal or rectangular grid | Generative                                | Unsupervised                | Competitive Learning                            | Kohonen Self Organizing NN                                             | Dimensionality Reduction; Optimization problems; Clustering analysis         | SPAMbase               | TensorFlow                                                                                           |

## Data Science Interview Questions



# DATA SCIENCE INTERVIEW QUESTIONS

DatabaseTown.com



## 1 WHAT IS DATA SCIENCE?

Data Science deals with the processes of data mining, cleansing, analysis, visualization, and actionable insight generation. Data Science is the mining and analysis of relevant information from data to solve analytically complicated problems. It is most widely used technique between Artificial Intelligence and Machine Learning. For example, when you logged on any e-commerce website and browsed some categories and products before purchase, you are generating data, which will be helpful for analysts to know your behavior about purchase.

## 2 WHAT ARE RECOMMENDER SYSTEMS?

A subclass of data sifting frameworks that are intended to anticipate the inclinations or evaluations that a client would provide for an item. Recommender systems are generally utilized in music, pictures, research, news, articles, social labels, and so on.

## 3 CAN YOU UTILIZE MACHINE LEARNING FOR TIME SERIES ANALYSIS?

Yes, machine learning can be utilized for time series analysis but it depends on the applications.

## 4 HOW WILL YOU ASSESS THE STATISTICAL IMPORTANCE OF AN INSIGHT?

By utilizing Hypothesis Testing, we can assess the statistical significance of an insight.

## 5 FOR TEXT ANALYTICS, PYTHON OR R WHICH ONE WOULD YOU GIVE THE PREFERENCE?

Python is the best choice for text analytics as it has Pandas library that provides easy to use data structures and better performance data analysis gadgets.

## 6 WHICH METHOD IS UTILIZED TO FORECAST CATEGORICAL RESPONSES?

Supervised machine learning i.e. Classification technique is widely utilized in mining for classifying data sets.

## 7 WHAT ARE THE BASIC EXPECTATIONS TO BE MADE FOR LINEAR REGRESSION?

Statistical independence of errors, normality of error distribution, linearity and additivity.

## 8 WHAT IS THE DIFFERENCE BETWEEN DATA SCIENCE AND MACHINE LEARNING?

Data Science deals with the processes of data mining, cleansing, analysis, visualization, and actionable insight generation, whereas, machine Learning is the part of Data Science which enables the system to process datasets autonomously without any human interference by utilizing various algorithms to work on massive volume of data generated and extracted from numerous sources.

## 9 WHAT IS THE FORMULA TO CALCULATE R-SQUARE?

R-Square can be calculated as:-  
 $1 - (\text{Residual Sum of Squares} / \text{Total Sum of Squares})$

## 10 WHAT BASIC KNOWLEDGE REQUIRED FOR DATA SCIENTIST?

Data Scientist must have the basic knowledge of mathematics, computer programming and statistics to solve the complex data problems in an efficient way to boost the business revenue.

## 11 NAMES OF BASIC MODELS OF MACHINE LEARNING?





Two books to help you prepare for your next Machine Learning interview:

### **Ace the Data Science Interview book**

<https://t.co/QwesE2FVEJ> <https://t.co/k8zqln5dgP>

### **Cracking-the-data-science-interview**

<https://github.com/khanhname1994/cracking-the-data-science-interview>

10 Statistical Concepts You Should Know For Data Science Interviews

<https://towardsdatascience.com/10-statistical-concepts-you-should-know-for-data-science-interviews-373f417e7d11>

### **Relationship Extraction for Knowledge Graph Creation From Biomedical Literature**

Paper: <https://arxiv.org/abs/2201.01647>

Abstract:

"Biomedical research is growing in such an exponential pace that scientists, researchers and practitioners are no more able to cope with the amount of published literature in the domain. The knowledge presented in the literature needs to be systematized in such a way that claims and hypothesis can be easily found, accessed and validated. Knowledge graphs can provide such framework for semantic knowledge representation from literature. However, in order to build knowledge graph, it is necessary to extract knowledge in form of relationships between biomedical entities and normalize both entities and relationship types. In this paper, we present and compare few rule-based and machine learning-based (Naïve Bayes, Random Forests as examples of traditional machine learning methods and T5-based model as an example of modern deep learning) methods for scalable relationship extraction from biomedical literature for the integration into the knowledge graphs. We examine how resilient are these various methods to unbalanced and fairly small datasets, showing that T5 model handles well both small datasets, due to its pre-training on large C4 dataset as well as unbalanced data. The best performing model was T5 model fine-tuned on balanced data, with reported F1-score of 0.88.

### **DS Community**

[Data Science Learner Community](#)

### **Metaverse – A New Era Emerging**

It's been a long day. Meera spent the morning attending back-to-back meetings, stole 10 minutes in the cafeteria to talk to her colleague about a movie she's going to that evening. She reaches the theatre and watches the latest release - The Matrix Resurrections. After the movie ends, she buys some groceries and thinks about what she's going to eat for the dinner. An ordinary day.

Twist in the story: Meera did it all without leaving her house. Welcome to the Metaverse.

Future of Investing: Did you miss the early internet boom in the 90s? Wish you were an early investor? Don't worry. If you missed the first bus, another lucrative internet boom is coming your way. Gen-Z and Gen-X investors are already breaking away from the traditional investing ecosystems and are likely to drive the future of the Metaverse. So, let's dive into it today to see the future!

What is a Metaverse?

If present-day internet is a 2D experience where you surf and scroll through a screen, the Metaverse will be 3D! Yes. Your digital twin will be 'walking' through it via connected headsets or glasses. As Mark

Zuckerberg says, "Instead of just viewing content — you are in it." The Metaverse is an immersive next-generation version of the internet, likely rendered by virtual or augmented reality technology. The Metaverse is part of Web 3.0, the successor to Web 1.0 and Web 2.0.

Web 3.0: Web 1.0 marked the arrival of the Internet. This period saw the dot-com boom and grabbed interest in e-commerce. Then came Web 2.0. It breathed new life into the internet. This time, Facebook, Uber, and others prospered. And boy, what we now cherish as a comfortable online environment falls short of even brushing the surface of the possibilities that Web 3.0 has to offer. It will essentially be an internet that will be decentralized and open-sourced.

Virtual reality (VR): VR is an immersive experience via a headset. You can see, and operate within a digital world. Remember those Samsung VR headsets?

Augmented reality (AR): If you've played Pokemon Go and caught a creature that you can only see through your phone, that's augmented reality.

Going forward, the lines between VR, and AR would blur as the metaverse becomes a reality!

The Unimaginable Possibilities, Cryptocurrencies, Gaming and the Economy

As a child, I always wondered if I could go inside the TV. It seems my naïve thought just became real! You will be able to teleport yourself anywhere, for a vacation or a meeting. One of the most important elements is the payments infrastructure, which will hold a significant place in the metaverse. The ability to buy, sell and own things in the metaverse could be the most transformative part. A virtual economy will need currency, and cryptocurrency and NFTs (non-fungible tokens) are answers to it.

While Metaverse is slated to enter through the gaming space, it will not only be restricted to the IT or gaming. Training, entertainment, sports, and the healthcare sector will change in unimaginable ways!

Which are the Key Players?

There are a few key players in this race, including Facebook, Microsoft, Amazon, and Snap. Recently, Mark Zuckerberg renamed his company - Meta and believes India will play a major role in building the proposed Metaverse ecosystem.

The social media giant has already invested in Meesho and Unacademy, both of which are now unicorns. Apart from the startups, established tech players like Reliance Jio are also betting huge on AR, MR, and overall blockchain technology.

Roadblocks to the Metaverse Universe

The full functionality of the Metaverse is likely still decades away. The Metaverse raises some legal questions as well. Further, creating it will also require technical cooperation among companies. While we continue to solve issues of data privacy and data security, there are worries that Metaverse could further compromise digital security.

Investing in the Metaverse

However, the momentum is there, and the pandemic has hastened the development of the Metaverse. Still in its infancy, as per the Bloomberg Intelligence report, Metaverse could be a market worth \$800 billion as early as 2024. That's huge! As an investor, you can focus initially on the hardware needed to build the Metaverse. Then the software that will design and host it, and finally the businesses that create use cases on it.

So, are you ready for the Metaverse?

### **Tensorflow data validation**

[https://colab.research.google.com/github/tensorflow/tfx/blob/master/docs/tutorials/data\\_validation/tfdv\\_basics.ipynb#scrollTo=8Ftd5k6AMkPV](https://colab.research.google.com/github/tensorflow/tfx/blob/master/docs/tutorials/data_validation/tfdv_basics.ipynb#scrollTo=8Ftd5k6AMkPV)

### **Page rank Algo**

<https://www.google.com/amp/s/www.geeksforgeeks.org/page-rank-algorithm-implementation/amp/>

### **Github: Auto Tensorflow**

<https://github.com/rafiqhasan/auto-tensorflow>

### **IIT Madras Offers Free Online 12- Week Certificate Course on AI**

<https://www.thebetterindia.com/270155/iit-madras-free-online-course-with-certificate-ai/>

### **Top 10 Applications of Machine Learning in Cybersecurity**

<https://www.analyticsinsight.net/top-10-applications-of-machine-learning-in-cybersecurity/>

### **How to handle Emoji '😊' & Emoticon ' :-)' in text preprocessing?**

<https://medium.com/geekculture/text-preprocessing-how-to-handle-emoji-emoticon-641bbfa6e9e7#:~:text=A%20better%20approach%20is%20to.it%20preserves%20the%20emoji%20information.&text=In%20the%20above%20review%2C%20we.texts%20with%20the%20word%20format>

### **Text Detection and Extraction using OpenCV and OCR**

<https://www.geeksforgeeks.org/text-detection-and-extraction-using-opencv-and-ocr/>

### **Devanagari Handwritten Character Dataset Data Set**

<https://archive.ics.uci.edu/ml/datasets/Devanagari+Handwritten+Character+Dataset>

### **Hindi-OCR**

This model involves recognition of hindi hand written characters using Convolutional neural network.

<https://github.com/darklord0303/Hindi-OCR>

### **Stack exchange all rooms**

<https://chat.meta.stackexchange.com/rooms?tab=all&sort=active>

### **How to improve Hindi text extraction?**

<https://stackoverflow.com/questions/67816185/how-to-improve-hindi-text-extraction>

### **What is Categorical Data \ Categorical Data Encoding Methods**

<https://www.analyticsvidhya.com/blog/2020/08/types-of-categorical-data-encoding/>

### **DART**

<https://medium.com/analytics-vidhya/darts-a-new-approach-simplifying-time-series-analysis-and-forecasting-in-machine-learning-8edcc177ad89>

<https://github.com/unit8co/darts>

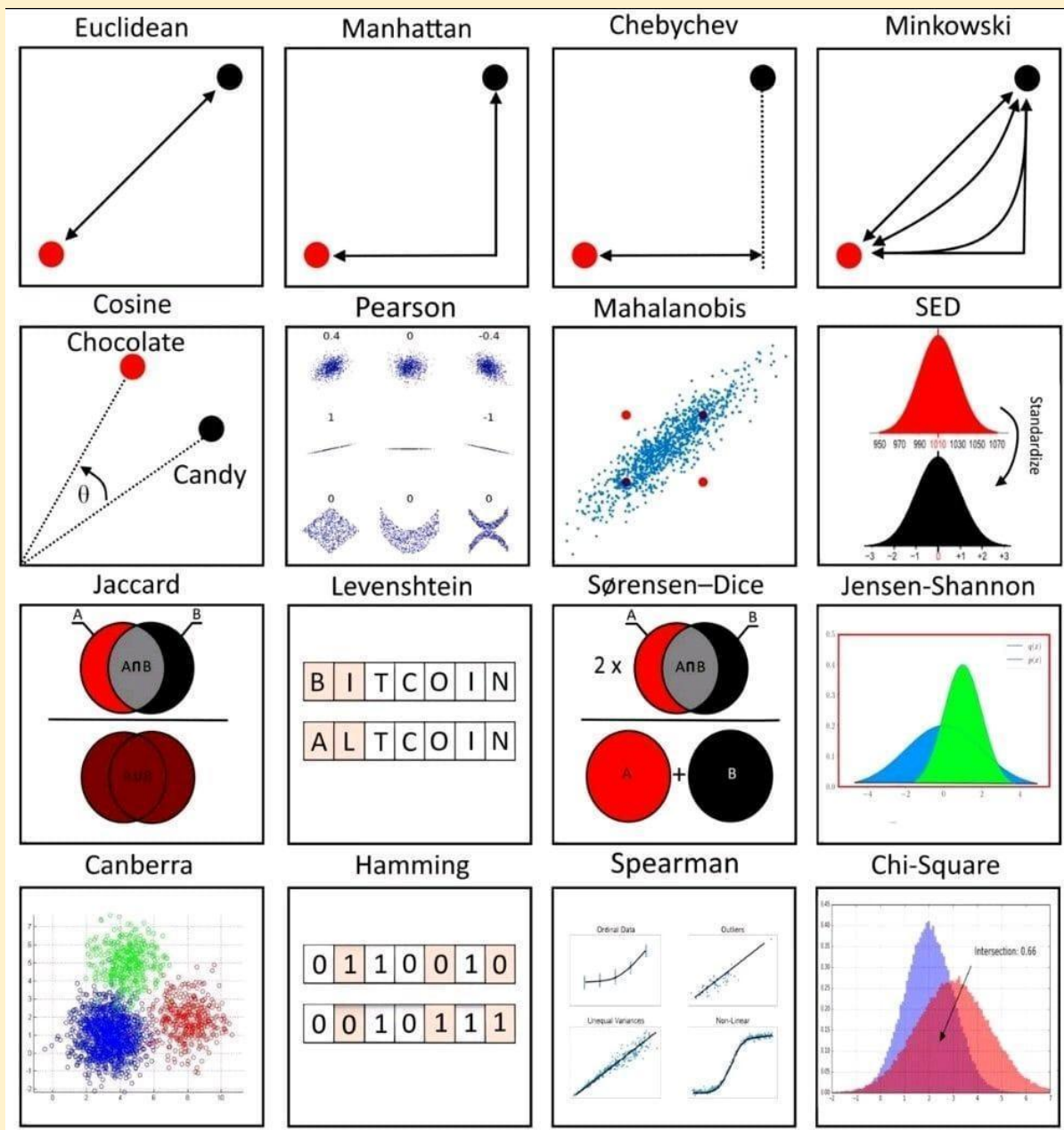
## TimeSeries Forecasting models

### Forecasting Models

Here's a breakdown of the forecasting models currently implemented in Darts. We are constantly working on bringing more models and features.

| Model                                                                            | Univariate | Multivariate | Probabilistic | Multiple-series training | Past-observed covariates support | Future-known covariates support |
|----------------------------------------------------------------------------------|------------|--------------|---------------|--------------------------|----------------------------------|---------------------------------|
| ARIMA                                                                            | ✓          |              | ✓             |                          |                                  | ✓                               |
| VARIMA                                                                           | ✓          | ✓            |               |                          |                                  | ✓                               |
| AutoARIMA                                                                        | ✓          |              |               |                          |                                  | ✓                               |
| ExponentialSmoothing                                                             | ✓          |              | ✓             |                          |                                  |                                 |
| Theta and FourTheta                                                              | ✓          |              |               |                          |                                  |                                 |
| Prophet                                                                          | ✓          |              | ✓             |                          |                                  | ✓                               |
| FFT (Fast Fourier Transform)                                                     | ✓          |              |               |                          |                                  |                                 |
| RegressionModel (incl. RandomForest, LinearRegressionModel and LightGBMModel)    | ✓          | ✓            |               | ✓                        | ✓                                | ✓                               |
| RNNModel (incl. LSTM and GRU); equivalent to DeepAR in its probabilistic version | ✓          | ✓            | ✓             | ✓                        |                                  | ✓                               |
| BlockRNNModel (incl. LSTM and GRU)                                               | ✓          | ✓            | ✓             | ✓                        | ✓                                |                                 |
| NBEATSModel                                                                      | ✓          | ✓            | ✓             | ✓                        | ✓                                |                                 |
| TCNModel                                                                         | ✓          | ✓            | ✓             | ✓                        | ✓                                |                                 |
| TransformerModel                                                                 | ✓          | ✓            | ✓             | ✓                        | ✓                                |                                 |
| TFModel (Temporal Fusion Transformer)                                            | ✓          | ✓            | ✓             | ✓                        | ✓                                | ✓                               |

## Statistical Distances



## AI Usecases

Agriculture: <https://www.datasciencecentral.com/top-3-ai-use-cases-in-agriculture-which-will-impact-yield/>

Administration: <https://www.datasciencecentral.com/top-7-data-science-use-cases-in-administration/>

Banking: <https://www.datasciencecentral.com/top-9-data-science-use-cases-in-banking/>

Construction: <https://t.co/opNntbvU8P>

Customer Support: <https://t.co/hnpXI8RFIb>

Design: <https://t.co/XNgnC8m99b>

Energy and Utilities: <https://t.co/MU9iYz06xU>

Finance: <https://t.co/tnelhXwA21>

Gaming: <https://t.co/QIWn5yfoy3>

Government: <https://t.co/1MtUpaapfW>

Healthcare: <https://t.co/O7AV575PuG>

Human Resources: <https://t.co/BHgg1TNDiS>

Insurance: <https://t.co/LzRqYf6RDP>

Manufacturing: <https://t.co/croKC2euke>

Marketing: <https://t.co/Sj53aWugMT>

Pharma: <https://t.co/u8ZGKo0xyd>

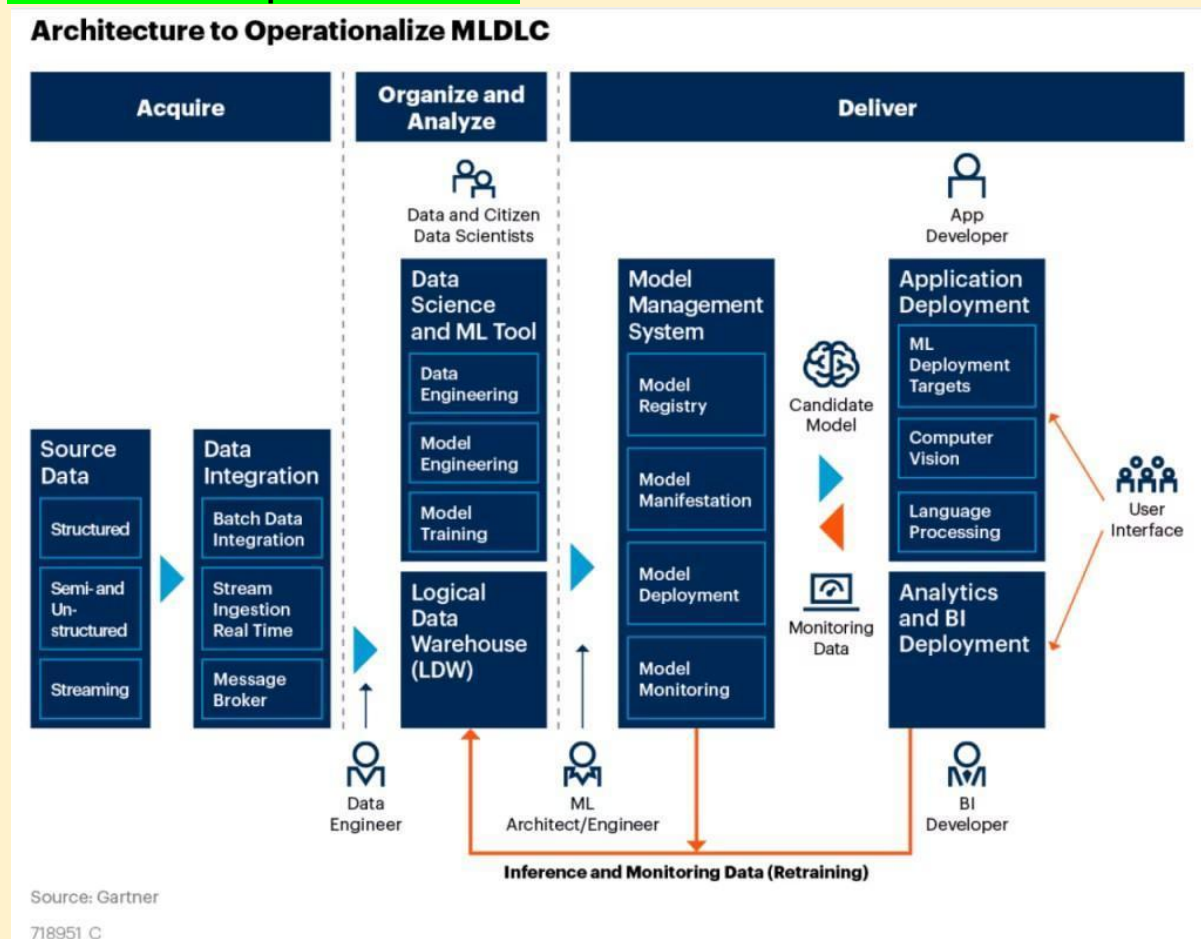
Production: <https://t.co/jmE91PHMIH>



Retail: <https://t.co/FmtiPCXnYK>  
 Sales: <https://t.co/mneWsNjwmM>  
 Telecom: <https://t.co/pFEpoOuk0X>  
 Transportation: <https://t.co/mJCoPwAh1G>  
 Travel: <https://t.co/uHPdHVIX2d>  
 and more... <https://t.co/jWIAEZuwmB>

<https://huggingface.co/ProsusAI/finbert>

## Architecture-to-operationalize-ML



## neptune.ai

<https://neptune.ai/blog/packaging-ml-models/amp>  
<https://github.com/alexlenail/NN-SVG>

## DeepMind shares a list of free AI & ML resources

<https://analyticsindiamag.com/deepmind-shares-a-list-of-free-ai-ml-resource/>

## AI course

<https://www.themlco.com/Academy/mgp>

## TOC of AI course

- ✓ Day-1: Introduction to Artificial Intelligence, Data Analytics & Road Map to become a Data Scientist
- ✓ Day-2: Data Preparation - Power Query & Tables
- ✓ Day-3: Data analytics- Formula & Pivot Table

- ✓ Day-4: Story Telling - Charts & Dashboard
- ✓ Day-5: Introduction to Python & Installing Python and its Libraries
- ✓ Day-6: Basic Python Programming for Data Analytics
- ✓ Day-7: Introduction to Statistics & Use Case of Statistics on Data
- ✓ Day-8: Population and Sampling
- ✓ Day-9: Connect Tableau to a Variety of Datasets
- ✓ Day-10: Visualize Data in the Form of Various Charts, Plots, and Maps
- ✓ Day-11: Connect Tableau to a Variety of Datasets
- ✓ Day-12: Visualize Data in the Form of Various Charts, Plots, and Maps and Calculate Data

## NUMPY

- ✓ Day-13: Python Numpy functions
- ✓ Day-14: Pandas for Data analytics in Python
- ✓ Day-15: Matplotlib for data visualization
- ✓ Day-16: Seaborn for data visualization
- ✓ Day-17: SQL basics for Data analytics
- ✓ Day-18: MongoDB basics for Data analytics
- ✓ Day-19: Introduction to Machine Learning & its libraries
- ✓ Day-20: Salary Estimation using K-NEAREST NEIGHBOR - SUPERVISED LEARNING
- ✓ Day-21: House Price Prediction using LINEAR REGRESSION - SUPERVISED LEARNING
- ✓ Day-22: Identifying the Pattern of the Customer spent using K-MEANS CLUSTERING
- ✓ Day-23: Market Basket Analysis using APRIORI
- ✓ Day-24: Web Ads. Click through Rate optimization using UPPER BOUND CONFIDENCE
- ✓ Day-25: Sentimental Analysis using Natural Language Processing
- ✓ Day-26: Introduction to Deep Learning & its libraries
- ✓ Day-27: Diabetes detection using Artificial Neural Network (MLP)
- ✓ Day-28: Object Recognition using Pre Trained Model – Caffe
- ✓ Day-29: Brain Tumor Detection using CNN
- ✓ Day-30: Stock Price prediction using LSTM

# Data Science Conceptual Map

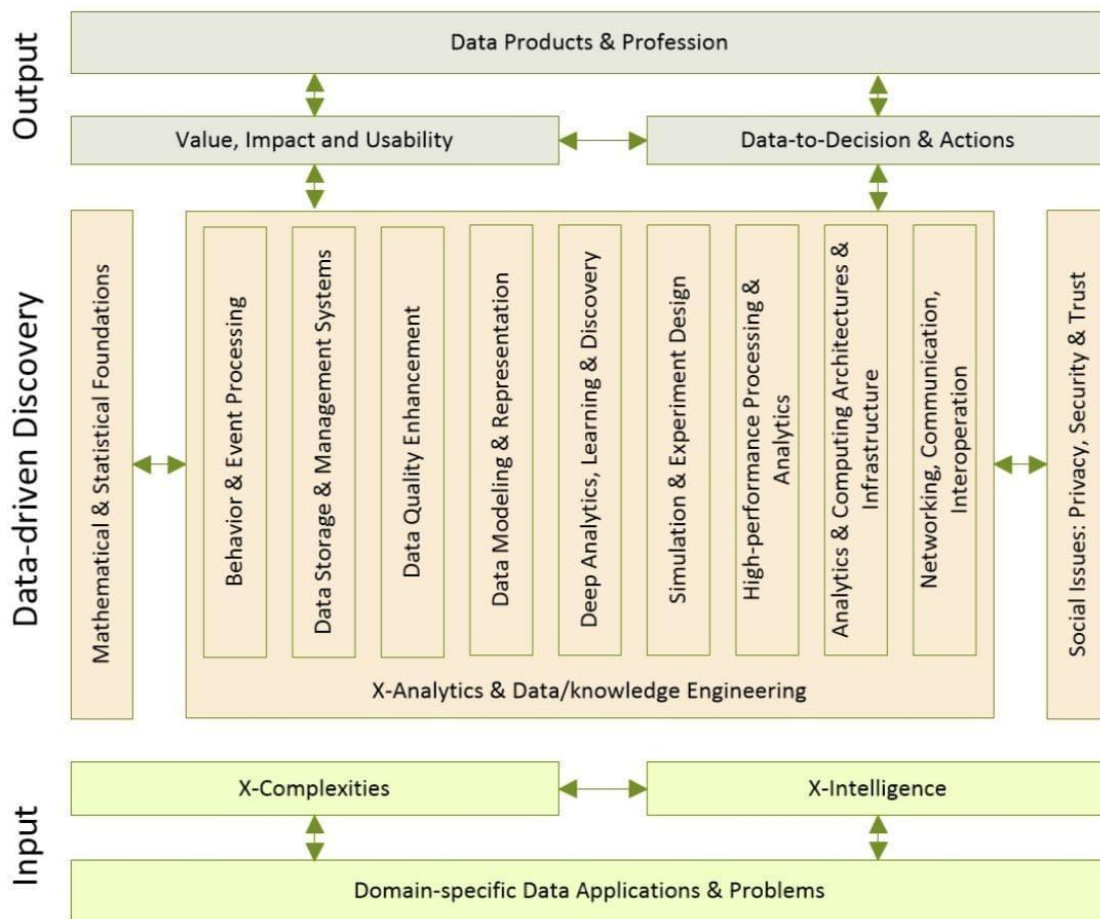
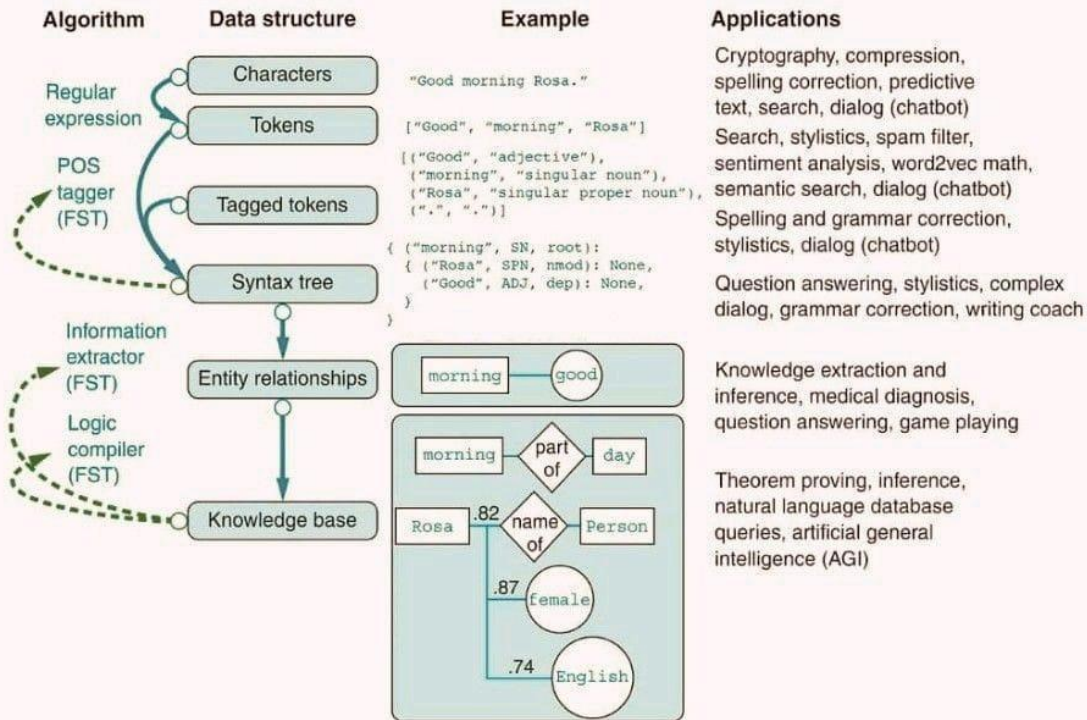


Fig. 6. Data science conceptual map.

# Layers in a NLP pipeline:

Source: Manning Publications





# Regression Analysis Estimation and Error Function

## Cheat Sheet – Regression Analysis

What is Regression Analysis?

Fitting a function  $f(\cdot)$  to datapoints  $y_i=f(x_i)$  under some error function. Based on the estimated function and error, we have the following types of regression

**1. Linear Regression:**

Fits a **line** minimizing the **sum of mean-squared error** for each datapoint.

$$\min_{\beta} \sum_i \|y_i - f_{\beta}^{linear}(x_i)\|^2$$

$$f_{\beta}^{linear}(x_i) = \beta_0 + \beta_1 x_i$$

**2. Polynomial Regression:**

Fits a **polynomial** of order  $k$  ( $k+1$  unknowns) minimizing the **sum of mean-squared error** for each datapoint.

$$\min_{\beta} \sum_{i=0}^m \|y_i - f_{\beta}^{poly}(x_i)\|^2$$

$$f_{\beta}^{poly}(x_i) = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \dots + \beta_k x_i^k$$

**3. Bayesian Regression:**

For each datapoint, fits a **gaussian distribution** by minimizing the **mean-squared error**. As the number of data points  $x_i$  increases, it converges to point estimates i.e.  $n \rightarrow \infty, \sigma^2 \rightarrow 0$

$$\min_{\beta} \sum_i \|y_i - \mathcal{N}(f_{\beta}(x_i), \sigma^2)\|^2$$

$$f_{\beta}(x_i) = f_{\beta}^{poly}(x_i) \text{ or } f_{\beta}^{linear}(x_i)$$

$$\mathcal{N}(\mu, \sigma^2) \rightarrow \text{Gaussian with mean } \mu \text{ and variance } \sigma^2$$

**4. Ridge Regression:**

Can fit either a **line**, or **polynomial** minimizing the **sum of mean-squared error** for each datapoint and the **weighted L2 norm** of the function parameters beta.

$$\min_{\beta} \sum_{i=0}^m \|y_i - f_{\beta}(x_i)\|^2 + \sum_{j=0}^k \beta_j^2$$

$$f_{\beta}(x_i) = f_{\beta}^{poly}(x_i) \text{ or } f_{\beta}^{linear}(x_i)$$

**5. LASSO Regression:**

Can fit either a **line**, or **polynomial** minimizing the **sum of mean-squared error** for each datapoint and the **weighted L1 norm** of the function parameters beta.

$$\min_{\beta} \sum_{i=0}^m \|y_i - f_{\beta}(x_i)\|^2 + \sum_{j=0}^k |\beta_j|$$

$$f_{\beta}(x_i) = f_{\beta}^{poly}(x_i) \text{ or } f_{\beta}^{linear}(x_i)$$

**6. Logistic Regression (NOT regression, but classification):**

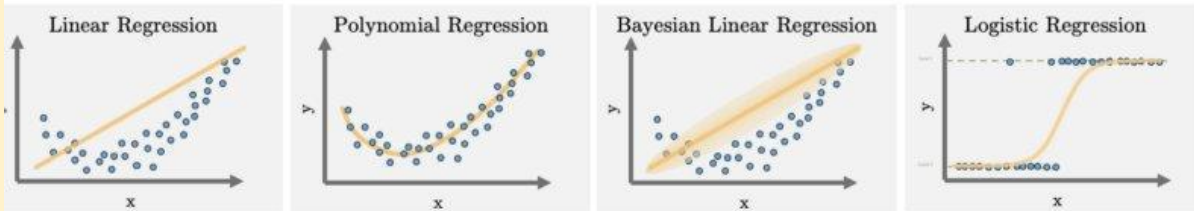
Can fit either a **line**, or **polynomial with sigmoid activation** minimizing the **sum of mean-squared error** for each datapoint. The labels  $y$  are binary class labels.

$$\min_{\beta} \sum_i \|y_i - \sigma(f_{\beta}(x_i))\|^2$$

$$f_{\beta}(x_i) = f_{\beta}^{poly}(x_i) \text{ or } f_{\beta}^{linear}(x_i)$$

$$\sigma(t) = \frac{1}{1 + e^{-t}}$$

Visual Representation:



Summary:

|                 | What does it fit?                    | Estimated function                                                      | Error Function                                                     |
|-----------------|--------------------------------------|-------------------------------------------------------------------------|--------------------------------------------------------------------|
| Linear          | A line in n dimensions               | $f_{\beta}^{linear}(x_i) = \beta_0 + \beta_1 x_i$                       | $\sum_{i=0}^m \ y_i - f_{\beta}(x_i)\ ^2$                          |
| Polynomial      | A polynomial of order k              | $f_{\beta}^{poly}(x_i) = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \dots$ | $\sum_{i=0}^m \ y_i - f_{\beta}(x_i)\ ^2$                          |
| Bayesian Linear | Gaussian distribution for each point | $\mathcal{N}(f_{\beta}(x_i), \sigma^2)$                                 | $\sum_i \ y_i - \mathcal{N}(f_{\beta}(x_i), \sigma^2)\ ^2$         |
| Ridge           | Linear/polynomial                    | $f_{\beta}^{poly}(x_i) \text{ or } f_{\beta}^{linear}(x_i)$             | $\sum_{i=0}^m \ y_i - f_{\beta}(x_i)\ ^2 + \sum_{j=0}^n \beta_j^2$ |
| LASSO           | Linear/polynomial                    | $f_{\beta}^{poly}(x_i) \text{ or } f_{\beta}^{linear}(x_i)$             | $\sum_{i=0}^m \ y_i - f_{\beta}(x_i)\ ^2 + \sum_{j=0}^n  \beta_j $ |
| Logistic        | Linear/polynomial with sigmoid       | $\sigma(f_{\beta}(x_i))$                                                | $\sum_{i=0}^m \ y_i - f_{\beta}(x_i)\ ^2$                          |

### NLI for DB

<https://link.medium.com/HweLYqgcpcb>

### Transformer models - Hugging Face Course

<https://huggingface.co/course/chapter1/1>

### Learning From Data - Online Course (MOOC)

<https://home.work.caltech.edu/telecourse.html>



## Find Leadership Courses

<https://www.coursary.com/search?fromfld=pn&q=leadership&pntype=6S> leadership

## Wrapper to load dataset from remote PyTorch dataset

Simple wrapper classes to serve and consume PyTorch datasets remotely

Handy when you have large datasets and the training computer doesn't have storage for them.

Docs: <https://docs.labml.ai/api/helpers.html#remote>

Github: [https://github.com/labmlai/labml/tree/master/helpers/labml\\_helpers/datasets/remote](https://github.com/labmlai/labml/tree/master/helpers/labml_helpers/datasets/remote)

Although it loads data across the network it doesn't usually slow down the training because the data loader workers fetch the data in background.

## Advanced Certification in Data Science and AI by IIT Madras

<https://intellipaat.com/advanced-certification-data-science-artificial-intelligence-iit-madras/>

## How well do explanation methods for machine-learning models work?

<https://techxplore.com/news/2022-01-explanation-methods-machine-learning.html>

## Confusion Matrix for 2+ Classes

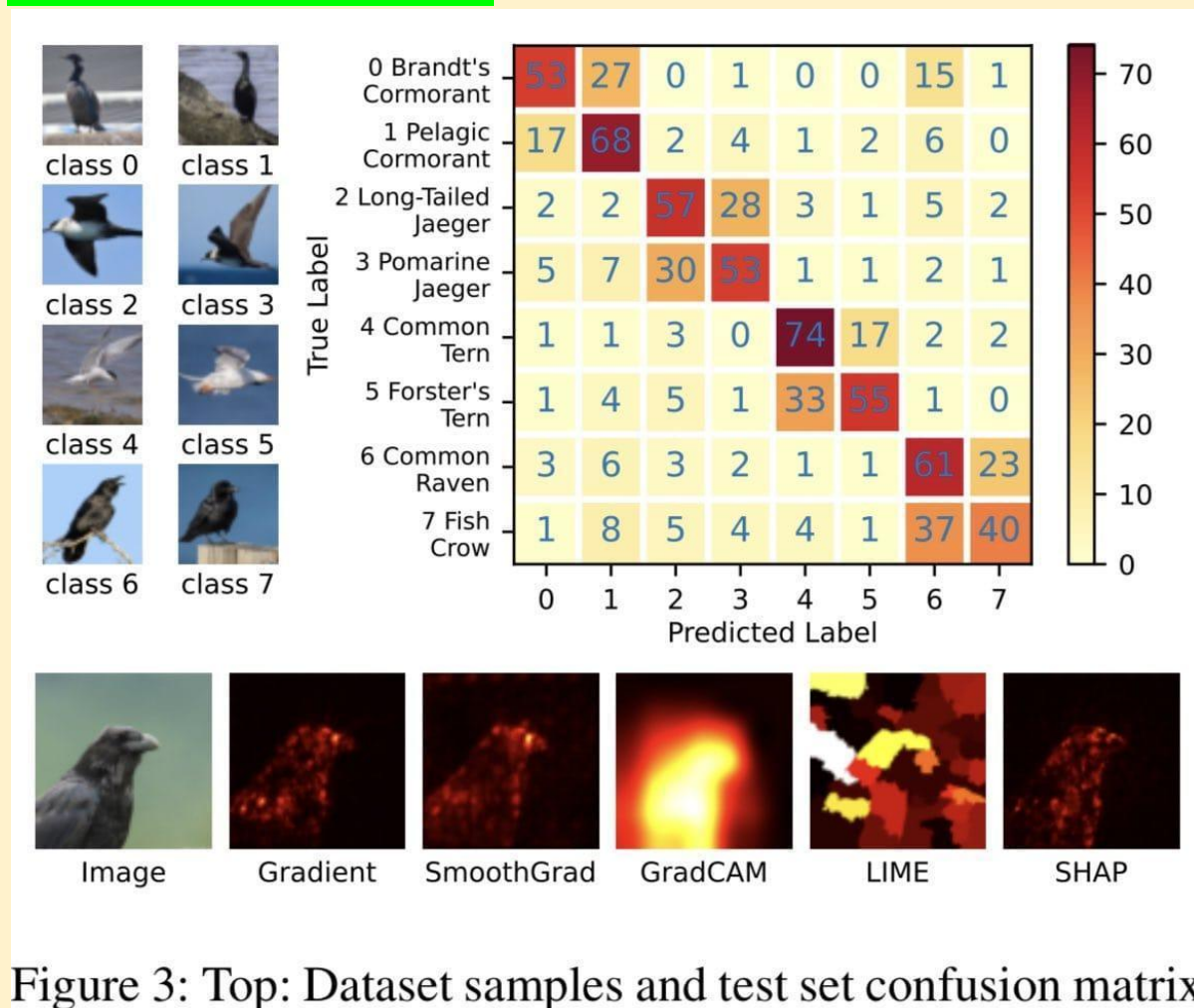


Figure 3: Top: Dataset samples and test set confusion matrix



Arjun Panwar • 1st

Freelancer | Tutor | Python | Data Science | Data...

1h • 🧑‍🎓

Hi, have a look at this ML algo CheatSheet!

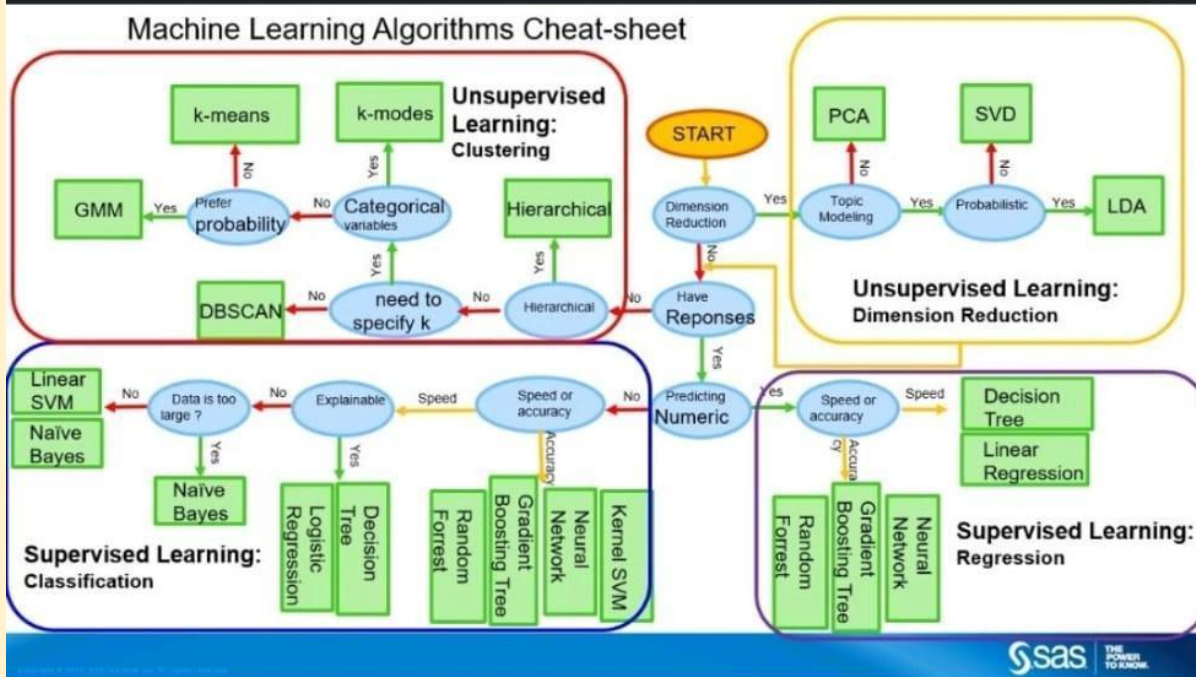
Image courtesy of [blogs.sas.com](https://blogs.sas.com)

Wanna learn Python, Machine Learning, Deep Learning and Statistics in one-on-one classes 🤔

Drop me a message to discuss your requirements 💬

Follow [Arjun Panwar](#) for more such resources 💡

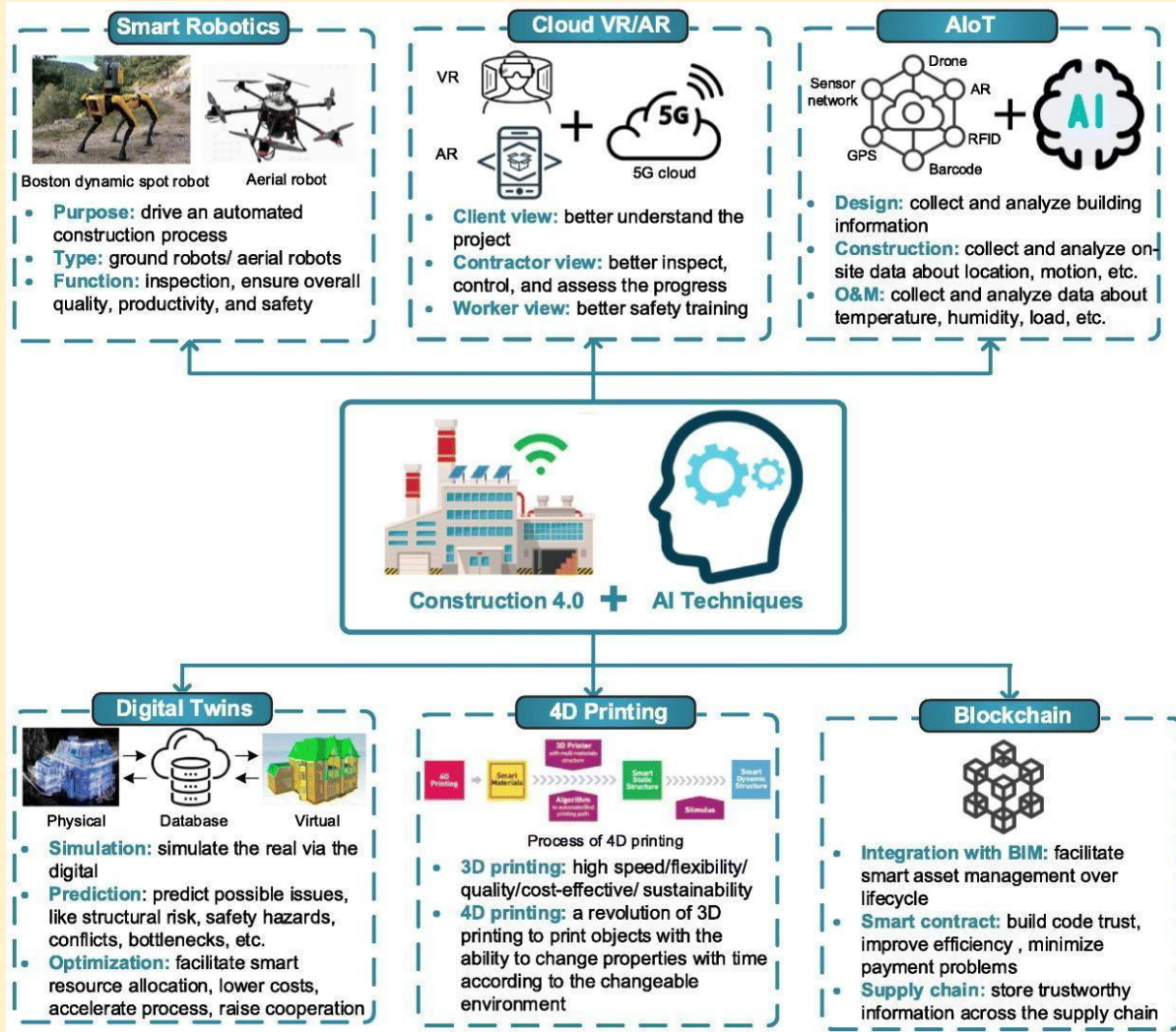
#python #machinelearning #deeplearning #statistics



Machine Learning Algorithms Free course 8hours

<https://youtu.be/7kyNYMwfYdw>

# AI Technologies in Construction

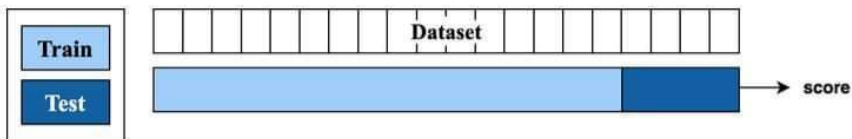


## Github: Data drift what next

<https://link.medium.com/GCMXMYKwrnb>

<https://github.com/codebasics/deep-learning-keras-tf-tutorial>

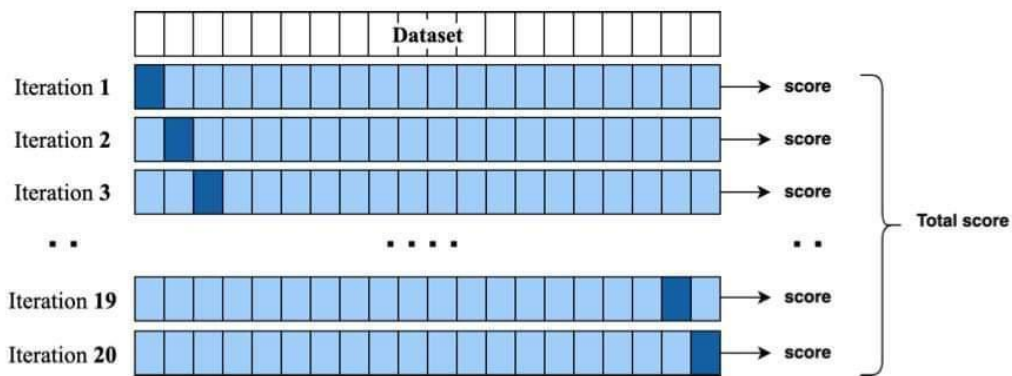
# Cross Validation - Hold Out, K-Fold, LOOCV



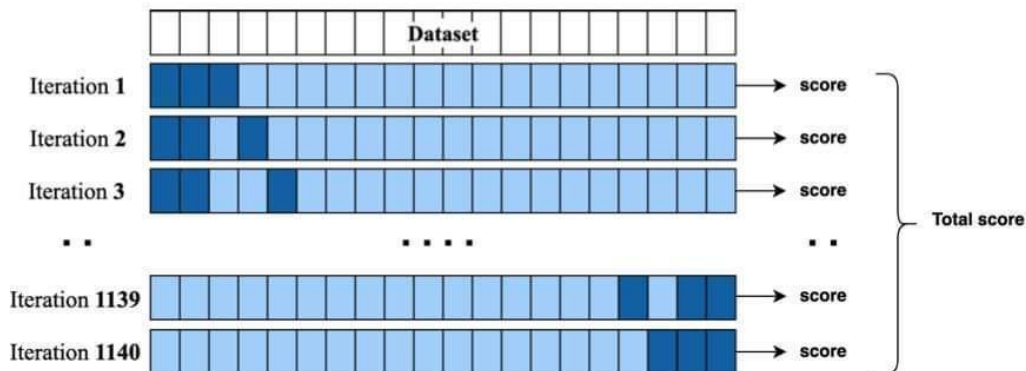
(a) Hold-out validation



(b)  $k$ -fold cross validation with  $k = 5$

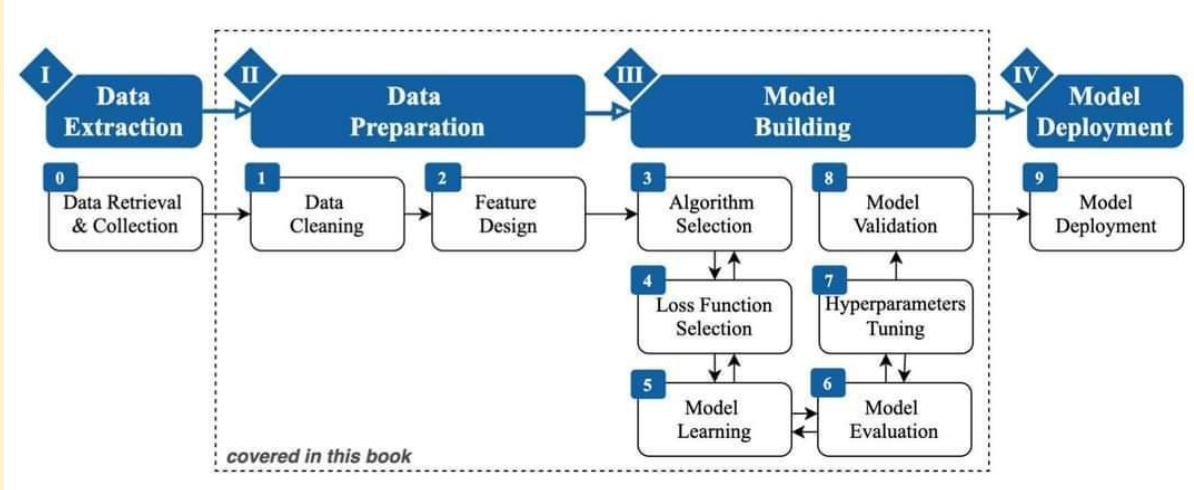


(c) Leave one out cross validation (LOOCV)





# Data Science Project Lifecycle



# Data Science Tools, Algorithms, Pipeline



# Graph neural networks (GNNs)

Let's have a look at some resources to help you learn and keep up-to-date with GNNs ↓

## 👉 Gentle Introduction to GNNs

- Gentle intro to GNNs (by @distillpub): <https://t.co/4v3be1Wf95>
- Foundations of GNNs (by @PetarV\_93): <https://t.co/rZzkXwhiSE>
- Graph Convolutional Networks (by @thomaskipf): <https://t.co/49cWWKzHdi>

## 👉 Diving Deep into GNNs



Some resources to dive deep:

- Geometric Deep Learning (by @mbronstein et al): <https://t.co/8u3VzYne07>
- Graph Representation Learning Book (by William Hamilton): <https://t.co/Okp6GuDW0C>
- CS224W: ML with Graphs (by @jure): <https://t.co/gaq42LH2Yw>

## Survey Papers

Explore survey papers for quick overviews:

- Graph Neural Networks: Methods, Applications, and Opportunities: <https://t.co/hl0aZWyxMk>
- A Comprehensive Survey on Graph Neural Networks: <https://t.co/4lmjEbhaj>

## GNN Papers and Implementations

If you want to keep up-to-date with popular methods and paper implementations for GNNs, check below

<https://t.co/qeDo19qExA>

## Benchmarks and Datasets

You can also find graph-related datasets, benchmarks, and leaderboards for graph-related data and GNNs on @paperswithcode:

<https://t.co/oPuWG0DzoX>

## Tools

Here are a few useful tools to get started with GNNs:

- PyTorch Geometric: <https://t.co/9pCoforoC6>
- Deep Graph Library: <https://t.co/Y0itY05Aau>
- jraph: <https://t.co/hGNXu4Uiyq>
- Spektral: <https://t.co/lWrit2yTCG>

Github: Framework - GraphSage

<https://github.com/williamleif/GraphSAGE>

<http://snap.stanford.edu/graphsage/>

Brilliant book by Kevin P. Murphy!

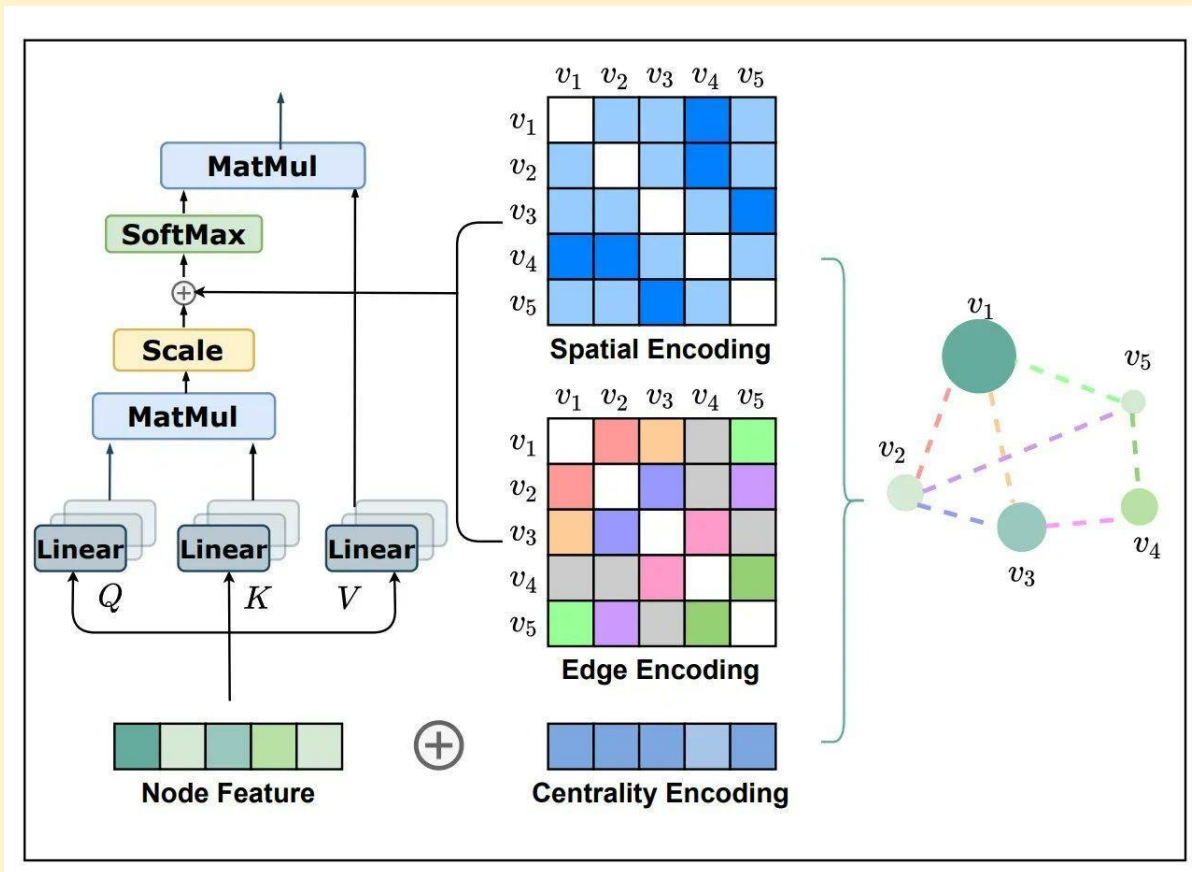
Probabilistic #MachineLearning (2nd Ed, 2022, PDF is free): <https://t.co/3T7n57qBjG>

+

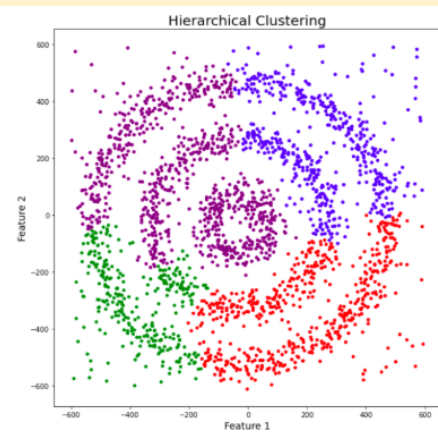
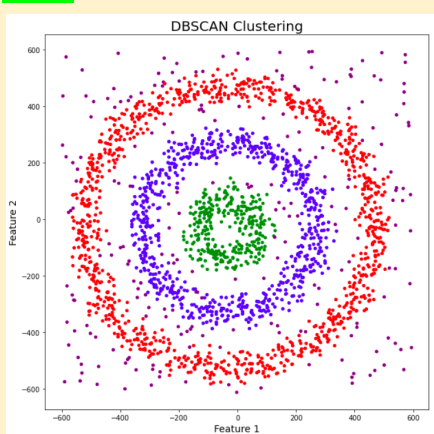
Read about it: <https://t.co/jEZar1nkWZ>

---

#AI #DeepLearning #BigData #DataScience #Mathematics #Probability #Statistics #LinearAlgebra  
#NeuralNetworks



## GNN



## Clustering

<https://www.analyticsvidhya.com/blog/2020/09/how-dbscan-clustering-works/>

<https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/>

<https://www.analyticsvidhya.com/blog/2017/02/test-data-scientist-clustering/>

## How to master Streamlit for data science

<https://blog.streamlit.io/how-to-master-streamlit-for-data-science/>

## [ML News] ConvNeXt: Convolutions return

<https://youtu.be/yVKiMh2vEWQ>

Good tools mentioned in this video.

- YourTTS: SOTA zero-shot Text-to-Speech

- MT3: Multi-Track Music Transcription
- Ru-DOLPH: HyperModal Text-to-Image-to-Text model

## **Nature of Research and Questions**

Exploratory... What.. Observe.. Identify some variables. Hypothetical or theoretical idea

Taking well defined theories and applying them in your area

Developing your own theory from scratch

University students mostly it is exploratory

Descriptive.. How.. Providing additional info about the topic. Build on exploratory research, requires a lot of data. Correlation.

Explanatory.. Explain.. Why.. Causality build on exploratory and descriptive research.

What is happening

How things work together

Why these things are happening.

Choosing research topic

Between current work, which you enjoy, which is valuable, your passion

Social media and hotel

Computer Gaming and business

Communications and social media

Using Different scripts to convey idea

Using different language to speak your mind/feelings.

Digitising Sanskrit documents

Translating Sanskrit documents Indian languages.

Transcribing Sanskrit documents to Indian languages

Knowledge base of Sanskrit documents.

Example

Most general - social media description

Less general - how skilled resources using social media

Concrete - mostly it is being used for brand awareness

Research Rational - my research gives an approach to skilled resources to increase brand awareness. Why it is worth. Why spending time is worth.

## **Tips for Sailing Data Science job**

Choose a business domain

Identify business problem

Look for dataset.

Solve the problem using algorithm you know.

Create an interesting story around above journey.

During interview connect with people using that story.

## **7 Interactive Bioinformatics Plots made in Python and R**

<https://www.r-bloggers.com/2017/10/7-interactive-bioinformatics-plots-made-in-python-and-r/>

## **Installing Python Packages from a Jupyter Notebook**

<http://jakevdp.github.io/blog/2017/12/05/installing-python-packages-from-jupyter/>

## **Neural Networks on Graph**

<https://medium.com/mlearning-ai/neural-networks-on-graphs-e06fc7f2b934>

## **Live code Visualization notebook**

<https://github.com/javascriptdata/dnotebook>

## **What Is a Bitcoin ATM?**

What Is a Bitcoin ATM?

Bitcoin ATMs work in a similar fashion to conventional cash machines — but instead of being shown their bank account details, users are presented with a series of options for trading Bitcoin in exchange for cash.

BATMs that only offer buying options are referred to as unidirectional, while those that offer both buy and sell functionality are known as bidirectional. Behind the scenes, BATMs are distinct from conventional ATMs. Most importantly, they do not connect to a user's bank or to a banking network. Instead, they are essentially internet-enabled interfaces that allow the user to interact with a specific exchange.

#BTM is the #cryptocurrency that powers Bytom. #Bytom is an interactive protocol of multiple byte assets. ... The Bytom price page is part of [Crypto.com](#) Price Index that features price history, price ticker, market cap and live charts for the top cryptocurrencies.

## **A New Reinforcement Learning Based Method - Dead-end-Discovery**

Microsoft AI Research Introduces A New Reinforcement Learning Based Method, Called 'Dead-end Discovery' (DeD), To Identify the High-Risk States And Treatments In Healthcare Using Machine Learning - MarkTechPost

<https://www.marktechpost.com/2022/02/09/microsoft-ai-research-introduces-a-new-reinforcement-learning-based-method-called-dead-end-discovery-ded-to-identify-the-high-risk-states-and-treatments-in-healthcare-using-machine-learning/>

## **Next Level of Data Visualization in Python**

<https://towardsdatascience.com/the-next-level-of-data-visualization-in-python-dd6e99039d5e>

## **7 Plotly Graphs in 3D: Stocks, Cats, and Lakes**

<https://blog.plotly.com/post/1013600482177-plotly-graphs-in-3d-stocks-cats-and-lakes>

<https://plotly.com/python/>

## **Github: Interactive graph in Python**

<https://github.com/matplotlib/ipymp>

## **Modern Deep Learning Techniques Applied to Natural Language Processing by Authors**

<https://nlpoverview.com/>

## **NLP with Deep Learning Video playlist Stanford Online**

<https://youtube.com/playlist?list=PLoROMvodv4rOSH4v6133s9LFPRHjEmbMJ>

## **Michigan University Fall 2020 Schedule**

**EECS 498-007 / 598-005: Deep Learning for Computer Vision**

<https://web.eecs.umich.edu/~justincj/teaching/eecs498/FA2020/schedule.html>

## **Deep Learning for Video, Master in Computer Vision Barcelona 2019**

<https://m6-video.github.io/deepvideo-2019/>

## **Computer Vision course at CTU in Prague**

Image retrieval lectures by @giotolias for Computer Vision course at CTU in Prague:

1) Classical image retrieval (yes, from our ancestors from hilltop villages ;) ) <https://t.co/3GuC2SD6mp>

2) Deep retrieval: <https://t.co/OqB5XmenzK>

Classical retrieval part 1: <https://t.co/UDTcOIQM4f>

Classical retrieval part 2: <https://t.co/neKuot0G02>

Deep retrieval : <https://t.co/y4Q4gqaJPc>

## **Github: DS Cheat sheets**

An important collection of the 15 best machine learning cheat sheets.

1- Supervised Learning

<https://github.com/afshinea/stanford-cs-229-machine-learning/blob/master/en/cheatsheet-supervised-learning.pdf>

2- Unsupervised Learning

<https://github.com/afshinea/stanford-cs-229-machine-learning/blob/master/en/cheatsheet-unsupervised-learning.pdf>

3- Deep Learning

<https://github.com/afshinea/stanford-cs-229-machine-learning/blob/master/en/cheatsheet-deep-learning.pdf>

4- Machine Learning Tips and Tricks

<https://github.com/afshinea/stanford-cs-229-machine-learning/blob/master/en/cheatsheet-machine-learning-tips-and-tricks.pdf>

5- Probabilities and Statistics

<https://github.com/afshinea/stanford-cs-229-machine-learning/blob/master/en/refresher-probabilities-statistics.pdf>

6- Comprehensive Stanford Master Cheat Sheet

<https://github.com/afshinea/stanford-cs-229-machine-learning/blob/master/en/super-cheatsheet-machine-learning.pdf>

7- Linear Algebra and Calculus

<https://github.com/afshinea/stanford-cs-229-machine-learning/blob/master/en/refresher-algebra-calculus.pdf>

8- Data Science Cheat Sheet



[https://s3.amazonaws.com/assets.datacamp.com/blog\\_assets/PythonForDataScience.pdf](https://s3.amazonaws.com/assets.datacamp.com/blog_assets/PythonForDataScience.pdf)

9- Keras Cheat Sheet

[https://s3.amazonaws.com/assets.datacamp.com/blog\\_assets/Keras\\_Cheat\\_Sheet\\_Python.pdf](https://s3.amazonaws.com/assets.datacamp.com/blog_assets/Keras_Cheat_Sheet_Python.pdf)

10- Deep Learning with Keras Cheat Sheet

<https://github.com/rstudio/cheatsheets/raw/master/keras.pdf>

11- Visual Guide to Neural Network Infrastructures

<http://www.asimovinstitute.org/wp-content/uploads/2016/09/neuralnetworks.png>

12- Scikit-Learn Python Cheat Sheet

[https://s3.amazonaws.com/assets.datacamp.com/blog\\_assets/Scikit\\_Learn\\_Cheat\\_Sheet\\_Python.pdf](https://s3.amazonaws.com/assets.datacamp.com/blog_assets/Scikit_Learn_Cheat_Sheet_Python.pdf)

13- Scikit-learn Cheat Sheet: Choosing the Right Estimator

[https://scikit-learn.org/stable/tutorial/machine\\_learning\\_map/](https://scikit-learn.org/stable/tutorial/machine_learning_map/)

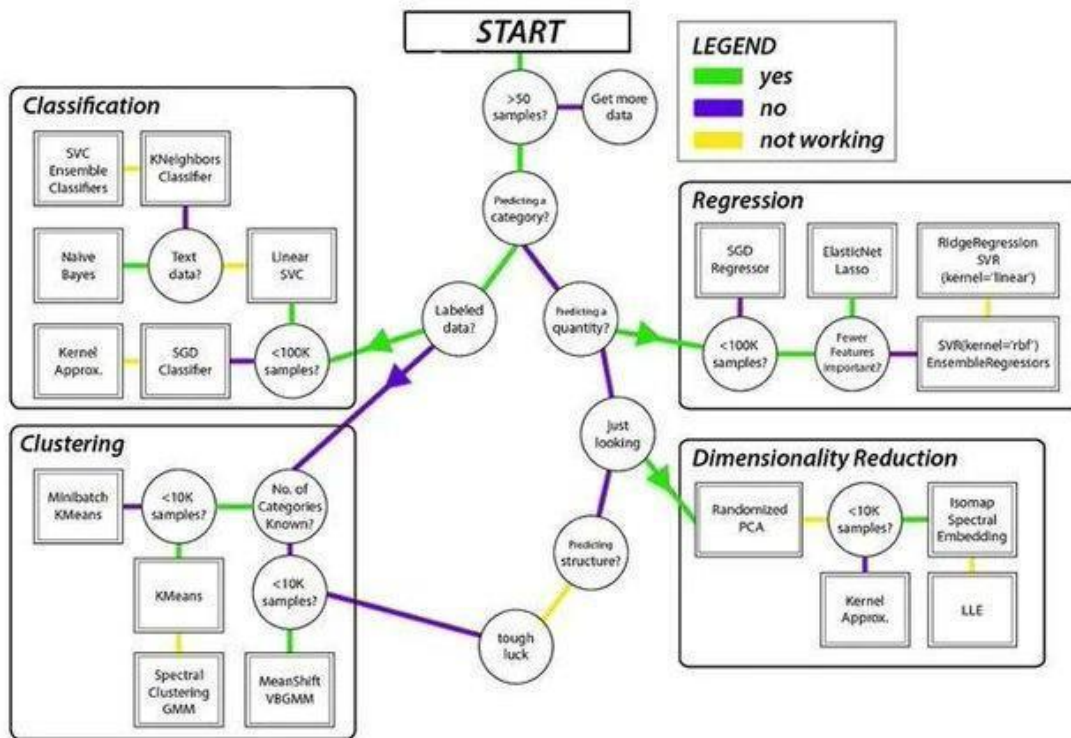
14- Tensorflow Cheat Sheet

<https://github.com/kailashahirwar/cheatsheets-ai/blob/master/PDFs/Tensorflow.pdf>

15- Machine Learning Test Cheat Sheet

<https://www.cheatography.com/lulu-0012/cheat-sheets/test-ml/pdf/>

# The simplest machine learning cheat-sheet ever!



## IIT Khadakpur Sanskrit NLP

<https://www.dqindia.com/iit-kharagpur-researchers-make-sanskrit-accessible-artificial-intelligence/>

## Making Sanskrit Accessible through AI-based Text Processing

<https://kgpchronicle.iitkgp.ac.in/making-sanskrit-accessible-through-ai-based-text-processing/>

## Fundamentals of NLP research in Sanskrit

<https://indiaai.gov.in/article/fundamentals-of-nlp-research-in-sanskrit>

## Is Sanskrit the most suitable language for natural language processing?

<https://ieeexplore.ieee.org/abstract/document/7724257?reload=true>

## Sanskrit github

<https://github.com/goru001/nlp-for-sanskrit>

## 10 Data Analysis Methods

# 10 KEY TYPES OF DATA ANALYSIS METHODS

Data mining does not have own methods of data analysis. It uses the methodologies and techniques of other related areas of science.

## Mathematical and Statistical Methods



### DESCRIPTIVE ANALYSIS

It does what the name suggests -"Describe". It looks at data and analyzes past events for deciding how to approach the future.



### REGRESSION ANALYSIS

It allows modeling the relationship between a dependent variable and one or more independent variables.



### FACTOR ANALYSIS

Factor analysis is a regression based data analysis technique, used to find an underlying structure in a set of variables.



### DISPERSION ANALYSIS

Dispersion is the spread to which a set of data is stretched. It is a technique of describing how extended a set of data is.

## Methods Based on The Artificial Intelligence, Machine Learning and Heuristic Algorithms



### NEURAL NETWORKS

They present a brain metaphor for information processing.

These models are biologically inspired computational models. They consist of an interconnected group of artificial neurons and process information using computation approach.



### DECISION TREES

The decision tree is a tree-shaped diagram that represents classification or regression models.

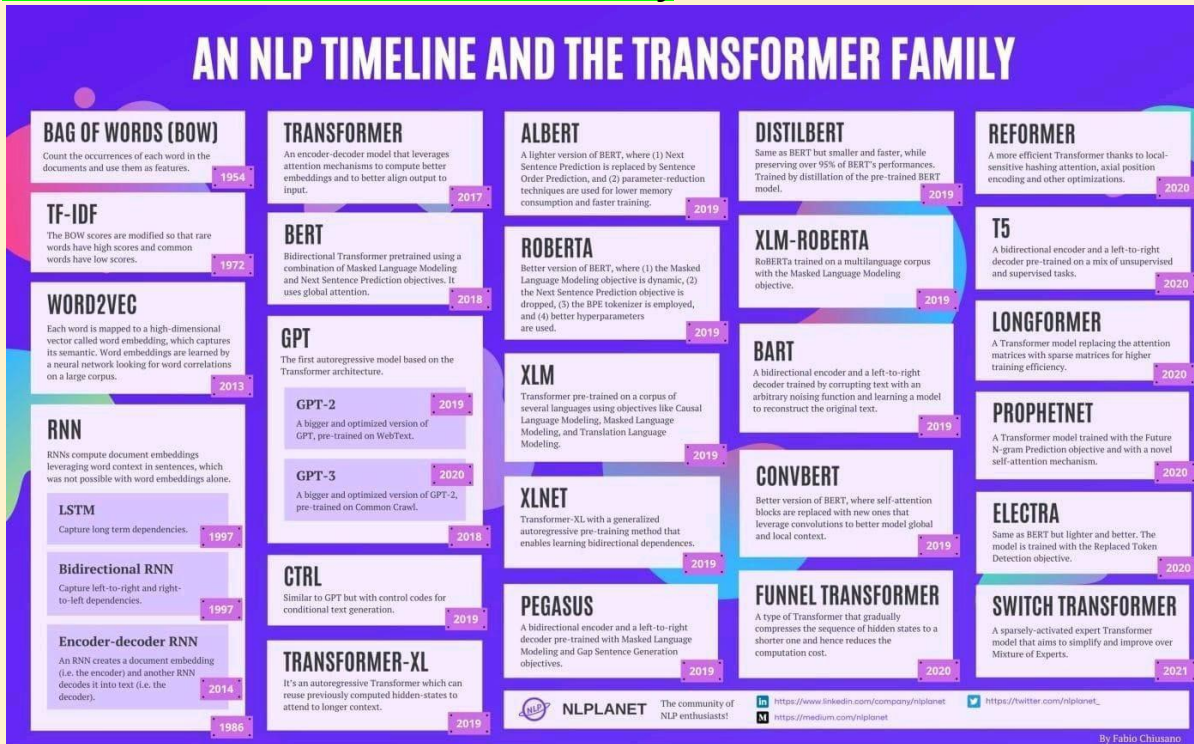
It divides a data set into smaller and smaller sub data sets while at the same time a related decision tree is continuously developed.



### EVOLUTIONARY ALGORITHMS

A common concept that combines

## AI NLP Timeline and the Transformer Family



## Introduction by Example - pytorch\_geometric 2.0.4

<https://pytorch-geometric.readthedocs.io/en/latest/notes/introduction.html>

## Python Notebook to Webapp

<https://github.com/mljar/mercury>

## 18 Feb Trending Github

- <https://github.com/probml/pml-book>
- <https://github.com/nlp-with-transformers/notebooks>
- <https://github.com/spmallick/learnopencv>
- [https://github.com/jeffheaton/t81\\_558\\_deep\\_learning](https://github.com/jeffheaton/t81_558_deep_learning)
- <https://github.com/dennybritz/reinforcement-learning>
- <https://github.com/understandable-machine-intelligence-lab/Quantus>
- <https://github.com/facebookresearch/Mask2Former>
- <https://github.com/TencentARC/GFPGAN>
- <https://github.com/PyTorchLightning/pytorch-lightning>
- <https://github.com/facebookresearch/moco>

## Real time machine-learning challenges and solutions

<https://huyenchip.com/2022/01/02/real-time-machine-learning-challenges-and-solutions.html>

## How Machine Learning Can Help Test Automation

<https://www.applause.com/blog/how-machine-learning-improves-test-automation>

## Test Automation in the World of AI & ML

<https://www.infoq.com/articles/test-automation-ai-ml/>



## Maths for neuroscience

<https://github.com/ebatty/MathToolsforNeuroscience>

## HyperSense AI Studio \ No Code Platform \ AutoML

<https://hypersense.subex.com/ai-studio/>

## Top 20 Websites for Data Science and ML

# TOP 20 WEBSITES FOR DATA SCIENCE AND ML

- ELITE DATA SCIENCE
- KDNUGGETS
- KAGGLE
- REDDIT – R/DATASCIENCE
- TOWARDS DATA SCIENCE
- DATAFLOQ
- DOMINO DATA SCIENCE BLOG
- MACHINE LEARNING MASTERY
- NATURE MACHINE INTELLIGENCE
- GEEKS FOR GEEKS – MACHINE LEARNING
- GOOGLE NEWS – DATA SCIENCE
- PETE WARDEN'S BLOG
- DATA36
- REVOLUTIONS
- EDWIN CHEN
- INSIDEBIGDATA
- GOOGLE AI BLOG
- ANALYTICS VIDHYA
- DATA SCIENCE DOJO
- DATA SCIENCE 101



## Websites for competitive programming

# WEBSITES FOR COMPETITIVE PROGRAMMING

| WEBSITES                                                                                               | HEADQUARTER                                                                               | RATING |
|--------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------|--------|
| 1) hackerrank.com     | INDIA    | ★★★★★  |
| 2) topcoder.com       | USA      | ★★★★☆  |
| 3) codechef.com       | INDIA    | ★★★★★  |
| 4) hackerearth.com  | INDIA  | ★★★★☆  |

## Real-time machine learning: challenges and solutions

<https://huyenchip.com/2022/01/02/real-time-machine-learning-challenges-and-solutions.html>

## Raspberry pi

<https://link.medium.com/Sh1cx11rSnb>

## Library for Auto labelling

Snorkel — Programmatically Build Training Data in Python. A Weak Supervision Approach to Label Your Training Dataset

<https://link.medium.com/uSW3wRTUYnb>

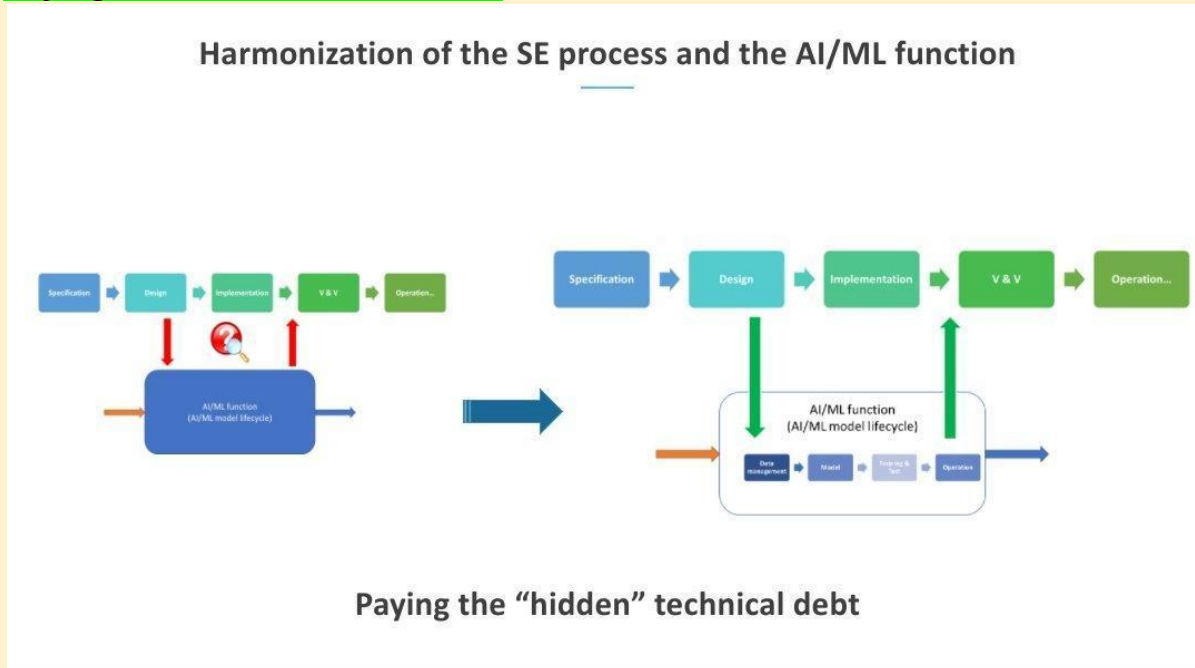
## Github: Using AWS Lamda and EFS

<https://github.com/aws-samples/ml-inference-using-aws-lambda-and-amazon-efs>

## Resources and tools

<https://bigscience.notion.site/Resources-and-tools-5a1d6e256087415484f42b43a4a14dde>

## Paying the hidden technical debt



## Hidden Technical Debt

Hidden #TechnicalDebt in #MachineLearning Systems — only a small fraction of real-world ML systems is composed of the ML code (the small box in the figure). The supporting infrastructure is large and complex:

<https://t.co/KHateaqtsK>

Identifying and Managing Technical Debt in Database Normalization Using Machine Learning and Trade-off Analysis. (paper :

<https://t.co/BTCE9AYazC>

Why your ML model is in DEBT!!! Hidden Technical Debt in Machine Learning is relevant threat to all intelligent decisions. #ArtificialIntelligence #AI #DataScience #Python #MachineLearning #BigData #DeepLearning #NLP #Robots #IoT <https://t.co/f0tHOt6kyA>

## Technical Debt in Machine Learning

<https://link.medium.com/937B7Liwaob>

[https://www.google.co.in/url?sa=t&source=web&rct=j&url=http://papers.neurips.cc/paper/5656-hidden-technical-debt-in-machine-learning-systems.pdf&ved=2ahUKEwivzs\\_q5LD2AhXI63MBHZwMA9MQFnoECAQQAQ&usq=AOvVaw3p0GJnRMO\\_JpbUVay4Kr4S](https://www.google.co.in/url?sa=t&source=web&rct=j&url=http://papers.neurips.cc/paper/5656-hidden-technical-debt-in-machine-learning-systems.pdf&ved=2ahUKEwivzs_q5LD2AhXI63MBHZwMA9MQFnoECAQQAQ&usq=AOvVaw3p0GJnRMO_JpbUVay4Kr4S)

What is hidden technical debt in machine learning systems?

These include boundary erosion, entanglement, hidden feedback loops, undeclared consumers, data dependencies, configuration issues, changes in the external world, and a variety of system-level anti-patterns.

## Hidden technical debt in machine learning systems

<https://link.medium.com/3toDff0vaob>

Paper : <https://papers.nips.cc/paper/2015/hash/86df7dcfd896fcdf2674f757a2463eba-Abstract.html>

Hidden Technical Debts and Solutions :  
<https://link.medium.com/vETOQ9Svaob>

How To Handle Hidden Technical Debt in a #MachineLearning Pipeline?  
<https://t.co/eU572S7BmN> #fintech #AI #ArtificialIntelligence #DeepLearning

Being able to select and design the most optimal architecture for your project is often what bridges the gap between #MachineLearning and operations, and ultimately what pays for the hidden technical debt in your ML system.

<https://t.co/vuxrdCckEK>

#TechnicalDebt #MachineLearning #AI #DataScience

### Best Docket Cheatsheets

<https://low-orbit.net/docker-cheat-sheet>

<https://dockerlabs.collabnix.com/docker/cheatsheet/>

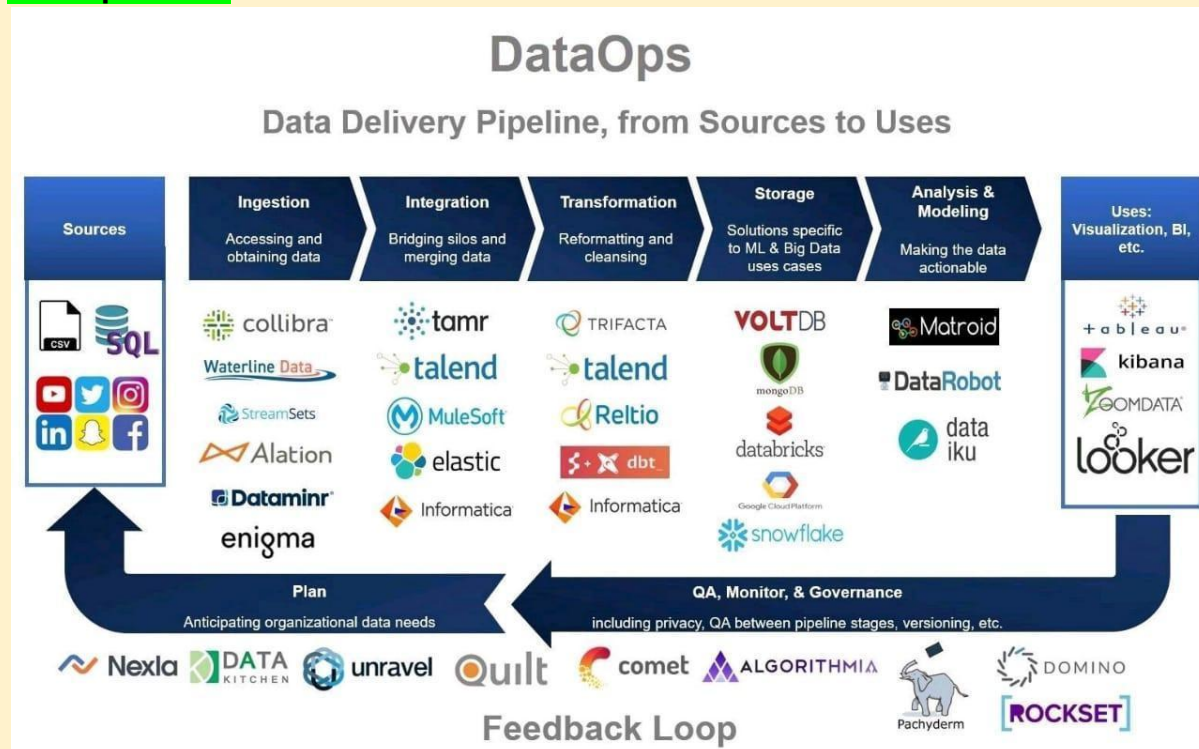
### How to Scale AI in Your Organization?

<https://hbr.org/2022/03/how-to-scale-ai-in-your-organization>

### CVPR 2022 paper Lifelong Graph Paper

Read CVPR 2022 paper "Lifelong Graph Learning", It converts features to nodes and turns nodes into graphs via a new graph topology. Go to project page: <https://t.co/MV3HHSSh17>

### DataOps Tools



### NLP Repo

<https://notebooks.quantumstat.com/>

## **100+ data science Cheatsheet**

<https://hubs.la/Q014LCQq0>

## **Ignition enabled fast prototyping of GNN**

Ignition enabled fast prototyping of Graph Neural Networks for Communication Networks:

[https://www.theinsaneapp.com/2020/12/machine-learning-and-data-science-cheat-sheets-pdf.html?utm\\_content=198825352&utm\\_medium=social&utm\\_source=linkedin&hss\\_channel=lcp-3740012](https://www.theinsaneapp.com/2020/12/machine-learning-and-data-science-cheat-sheets-pdf.html?utm_content=198825352&utm_medium=social&utm_source=linkedin&hss_channel=lcp-3740012)

#deeplearning #machinelearning #ai #gnn

## **Deep Learning with PyTorch, Full course**

Deep Learning with PyTorch, Full course

<https://www.youtube.com/watch?app=desktop&v=c36IUUr864M>




# LIST OF LARGE LANGUAGE MODELS

---

 **OpenAI** GPT-3

---

 **Google AI** SWITCH TRANSFORMER,  
GENERALIST LANGUAGE MODEL (GLAM), PATHWAYS LANGUAGE MODEL (PALM), LAMDA, T5, MT5

---

 **Microsoft** TURING NLG

---

 **DeepMind** GOPHER, CHINCHILLA

---

 **Meta** OPT, FAIRSEQ DENSE

---

**AI21 labs** JURASSIC

---

 **Baidu** ERNIE 3.0

---

 **LG** EXAONE

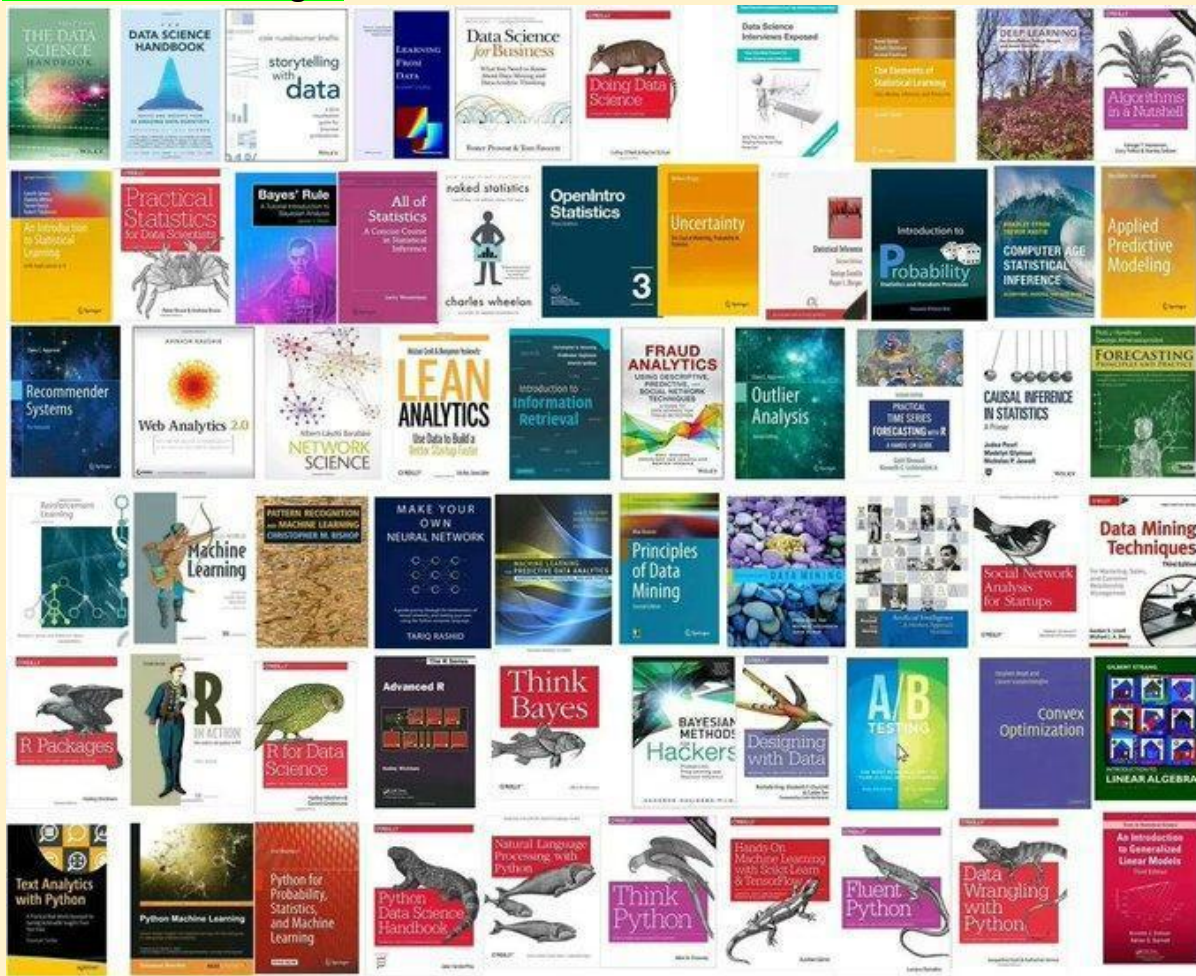
---

 **HUAWEI** PANGU ALPHA

"Large language models" are natural language processing systems which are trained on massive volumes of text, and are capable of answering reading comprehension questions or generating new text.



# AI Books Cover Pages



## Solution approach for ai problems





**Rithesh Sreenivasan** · Following  
Research Scientist, Educational YouTuber  
42m · Edited · 🌐

I always ask this question in my interviews. I give a scenario and ask for a solution approach

For instance, what would be the approach for leaf disease detection of tomato plant assuming images of leaf/leaves are captured with a camera.

95% of the times I get an answer as I will use deep learning, CNN and then do classification

Whereas my line of thinking would be

- 1) Why do i need to do disease detection?
- 2) What type of diseases are prevalent in tomato plants
- 3) Who is the intended user? A farmer or a drone pilot capturing images or an agriculture officer and so on...
- 4) How are the images captured? Is it from a mobile camera or a drone or a camera?
- 5) Should the algorithm be on device or will the images be sent to a central location for further processing
- 6) How do I take into account the various environmental conditions affecting the image capture
- 7) Should the image have a single leaf or bunch of leaves
- 8) What kind of image standardization should I do?
- 9) How many images needs to be captured to start development of a solution? Is there a similar dataset available?
- 10) How have others solved the problem?
- 11) Is classical image processing techniques good

## Algorithms and their applications

### Supported Algorithms

The following tables show the advised use cases for each algorithm. The column *Feature Level* indicates whether the detection can be done at the feature level, e.g. per pixel for an image. Check the [algorithm reference list](#) for more information with links to the documentation and original papers as well as examples for each of the detectors.

#### Outlier Detection

| Detector             | Tabular | Image | Time Series | Text | Categorical Features | Online | Feature Level |
|----------------------|---------|-------|-------------|------|----------------------|--------|---------------|
| Isolation Forest     | ✓       |       |             |      | ✓                    |        |               |
| Mahalanobis Distance | ✓       |       |             |      | ✓                    | ✓      |               |
| AE                   | ✓       | ✓     |             |      |                      |        | ✓             |
| VAE                  | ✓       | ✓     |             |      |                      |        | ✓             |
| AEGMM                | ✓       | ✓     |             |      |                      |        |               |
| VAEGMM               | ✓       | ✓     |             |      |                      |        |               |
| Likelihood Ratios    | ✓       | ✓     | ✓           |      | ✓                    |        | ✓             |
| Prophet              |         |       | ✓           |      |                      |        |               |
| Spectral Residual    |         |       | ✓           |      |                      | ✓      | ✓             |
| Seq2Seq              |         |       | ✓           |      |                      |        | ✓             |

#### Adversarial Detection

| Detector           | Tabular | Image | Time Series | Text | Categorical Features | Online | Feature Level |
|--------------------|---------|-------|-------------|------|----------------------|--------|---------------|
| Adversarial AE     | ✓       | ✓     |             |      |                      |        |               |
| Model distillation | ✓       | ✓     | ✓           | ✓    | ✓                    |        |               |

#### Drift Detection

| Detector           | Tabular | Image | Time Series | Text | Categorical Features | Online | Feature Level |
|--------------------|---------|-------|-------------|------|----------------------|--------|---------------|
| Kolmogorov-Smirnov | ✓       | ✓     |             | ✓    | ✓                    |        | ✓             |
| Cramér-von Mises   | ✓       | ✓     |             |      |                      | ✓      | ✓             |

## What is Outlier \ PyOD For Outlier Detection in Python

<https://www.analyticsvidhya.com/blog/2019/02/outlier-detection-python-pyod/>

## Free ML tools from Microsoft

Microsoft have free courses in data science & machine learning using python... 🌟

1. Data Science For Beginner : 12 weeks, 25 lessons, 50 quizzes, classic Machine Learning for all  
<https://github.com/microsoft/Data-Science-For-Beginners>

2. Machine Learning for Beginners



<https://github.com/microsoft/ML-For-Beginners>

3. For Self Learners, Below free resources helps a lot.

<https://github.com/microsoft/ai-for-beginners>

### **How To Build a Chatbot Using Rasa NLU?**

<https://www.analyticsvidhya.com/blog/2019/04/learn-build-chatbot-rasa-nlp-ipl/>

### **Researchers From China Introduce Vision GNN (ViG)**

A Graph Neural Network For Computer Vision Systems

<https://www.marktechpost.com/2022/06/08/researchers-from-china-introduce-vision-gnn-vig-a-graph-neural-network-for-computer-vision-systems/>

### **How to Easily Automate Emails with Python**

<https://towardsdatascience.com/how-to-easily-automate-emails-with-python-8b476045c151>

### **Fast Classification and Clustering via Image Convolution Filters**

<https://www.linkedin.com/pulse/fast-classification-clustering-via-image-convolution-granville>

### **Automate Zerodha login through Selenium Webdriver**

<https://medium.com/@algotrading0103/automate-zerodha-login-through-selenium-webdriver-dd73bb625c7f>

### **Publicly Real-World Datasets To Evaluate Stream Learning Algorithms**

<https://sites.google.com/view/uspdsrepository>

### **AI4Bharat Data Annotator Application**

<https://docs.google.com/forms/d/e/1FAIpQLSfK5yo1IUSEVB0WsCI92oLy5PLLGWEIGSewX0xhVFOCSyo5ZQ/viewform>

<https://neptune.ai/blog/packaging-ml-models>

### **CNN sign language detection action recognition**

<https://youtu.be/doDUihpj6ro>

<https://towardsdatascience.com/tutorial-using-deep-learning-and-cnns-to-make-a-hand-gesture-recognition-model-371770b63a51>

<https://www.google.com/amp/s/techvidvan.com/tutorials/hand-gesture-recognition-tensorflow-opencv/%3famp=1>

## Companies hiring data scientist



**Bhavishya Pandit** • Following

Associate Data Scientist @ Rakuten | 2.2 M+ views...

2h • 🌐

25 companies that hire freshers for data science roles in India 🔥

- American Express
- Atlan
- Bajaj Finserv
- Deloitte
- EY
- Fractal
- Gartner
- Genpact
- Greendeck
- Honeywell
- Hyperverge
- IBM
- OYO
- Mu sigma
- nference
- Nvidia
- Rakuten India
- Siemens
- Sony Research India
- Swiggy
- The Math Company
- Tiger Analytics
- Toppr
- Trell
- Udaan
- ZS

## Google drive DS material

[https://drive.google.com/drive/u/0/mobile/folders/1iW7IPrVUqsHumgXUMH\\_rgeBLpJjRDCmJ](https://drive.google.com/drive/u/0/mobile/folders/1iW7IPrVUqsHumgXUMH_rgeBLpJjRDCmJ)

## Microsoft Is Going Global With Speech Enablement

<https://cobusgreyling.medium.com/microsoft-is-going-global-with-speech-enablement-c5f43d819077>

## A topic-centric list of HQ open datasets

<https://github.com/awesomedata/awesome-public-datasets>

## Top Explainable AI (XAI) Python Frameworks in 2022

<https://moez-62905.medium.com/top-explainable-ai-xai-python-frameworks-in-2022-94ff4610b0f5>

## MLOPs with AWS Cloud

Here is a quick article about the MLOps with AWS Cloud along with its cool features.

[#aws](#) [#sagemaker](#) [#mlops](#) [#machinelearning](#)

### MLOps with AWS Cloud



Sagemaker



Quicksight



Cloudwatch



S3



API Gateway



Lambda

### MLOps with Cloud: AWS Sagemaker

Sridhar C R on LinkedIn  • 3 min read

Introduction This article talks about the several key features with cloud powered ML developments...

## The world of machine learning algorithms

### Regression

- Ordinary Least Squares Regression (OLSR) Stepwise Regression
- Linear Regression Logistic Regression
- Multivariate Adaptive Regression Splines (MARS)
- Locally filmated Scatterplot Smoothing (LOESS Jackkrufe Regression)

### Bayesian

- Naive Bayes
- Gausalan Naive Bayes
- Multinomial Naive Bayes Averaged One-Dependence Estimators (AOO)

- Bayesian Belief Network (BBN) Bayesian Network (BN)

#### Hidden Markov Models

- Conditional random fields (CRF)

#### Regularization

- Ridge Regression
- Least Absolute Shrinkage and Selection Operator (LASSO)
- Elastic Net Least-Angle Regression (LARS))

#### Decision tree

- instance based
- Classification and Regression Tree (CART)
- Nerative Dichotomiser
- Chi-squared Automatic interaction Detection (CHAID)
- Decision Stump
- Random Forests
- Conditional Decision Trees also called cake-based, memory-based
- K-Nearest Neighbour (ANN)
- Learning Vector Quantization (LVQ)
- Self-Organizing Map (SOM)
- Locally Weighted Learning (LWL)

#### Clustering

- Single-linkage clustering
- k-Means k-Medians
- Expectation Maximisation (EM) Hierarchical Clustering
- Fuzzy clustering
- DBSCAN
- OPTICS algorithm Non Negative Matrix Factorization
- Latent Dirichlet allocation (LDA)

#### Dimesionality reduction

- Principal Component Analysis (PCA)
- Principal Component Regression (PCR)
- Partial Least Squares Regression (PLSR) Sammon Mapping
- Multidisional Scaling (MDS)
- Projection Pursuit Discriminant Analysis (LDA, MOA, GDA, FDA)

#### Deep learning

- Neural networks
- Deep Boltzmann Machine (DBM)
- Deep Belief Networks (DBN)
- Convolutional Neural Network (CNN)
- Stacked Auto-Encoders
- Suff Organizing Map
- Perceptron
- Back-Propagation
- Hopfield Network Radial Basis Function Network (RBFM)
- Backpropagation
- Autoencoders Hopfield networks
- Boltzmann machines Restricted Boltzmann Machines
- Spiking Neural Networks Learning Vector quantization (LVQ)



Associated rule

- Aprior
- FP-Growth

Ensemble

- Logit Boost (Boosting) Bootstrapped Aggregation (Bagging)
- Stacked Generalization (blending)
- Gradient Boosting Machines (GEM)
- Gradient Boosted Regression Trees (GBRT) Randoms Forest

Support Vector Machines (SVM)

- Evolutionary Algorithms Inductive Logic Programming (ILP)
- Reinforcement Learning (Q-Learning. Temporal Difference.
- State-Action-Reward-State-Action (SARSA)) ANOVA
- Information Fuzzy Network (IFN) Page Rank
- Conditional Random Fields (CRF)

## Think Big Data

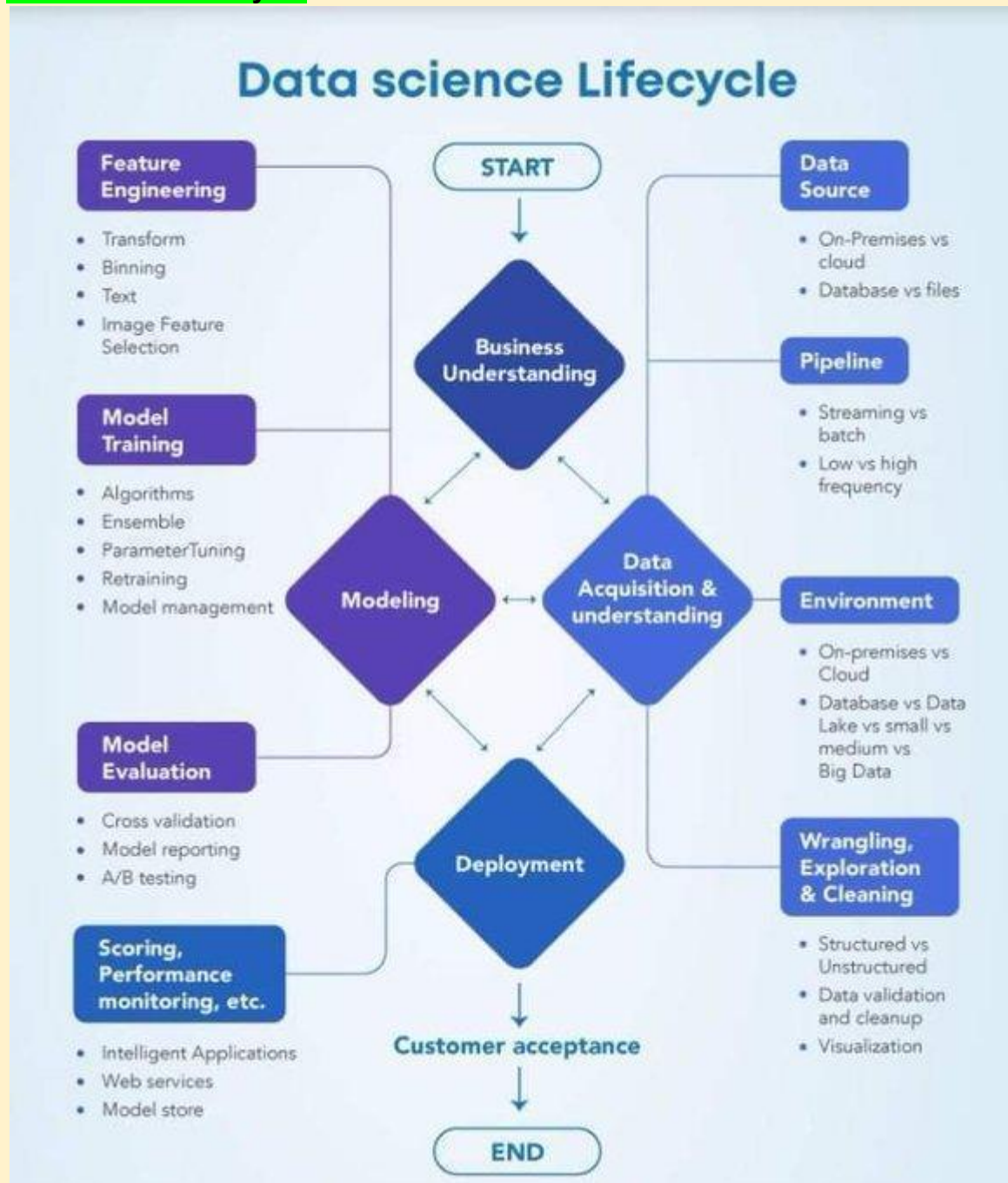
is a digital marketing consulting firm providing Search Engine Optimization (SEO), Content development, Data Analysis, and Digital Strategy planning services for organizations looking to promote and create awareness of their products and services online through digital communication channels including websites, newsletters, social media, mobile app's, and search engines to enhance or create a new competitive advantage and to increase market share.

<https://thinkbigdata.nyc/>

## 9 Reasons - Why Artificial Intelligence is So Essential Right Now?



## Data Science Lifecycle



## ML Deployment Tools

**Seldon:** <https://t.co/BCryKE60uE> offers Seldon core, which is an open-source framework. This framework simplifies and accelerates ML models and experiment deployment.

Seldon also offers the power to connect your project with (CI/CD) tools to scale and update model deployments.

<https://www.seldon.io/>

Why Seldon?

- Custom offline models.
- Real-time predictions exposing APIs to external clients.
- Simplifies the deployment process.

<https://t.co/lt0R2TOjvl>

**BentoML** : it simplifies the process of building ML API endpoints. It offers a standard, yet simplified architecture to migrate trained ML models to production & let you package trained models, using any ML framework to interpret them, for serving in a production environment.

Why BentoML?

- Supports high-performance model serving, model management, model packaging, and a unified model format.
- Supports deployment to multiple platforms.
- Flexible and modular design.

**AWS Sagemaker** : it is a powerful service provided by Amazon. It gives ML developers the ability to build, train, and deploy machine learning models quickly.

It simplifies the whole ML process by removing some of the complex steps, thus providing highly scalable ML models.

Why AWS Sagemaker ?

- The setup process is simple and can run with Jupyter Notebook. Hence, the management and deployment of scripts are simplified.
- The cost is modular, based on the feature you use.
- Model training is done on multiple servers.

**Kubeflow**: The main operations of Kubeflow include packages, and organizing docker containers that help maintain an entire machine learning system.

It simplifies the development and deployment of machine learning workflows, in turn making models traceable.

Why Kubeflow?

- it simplifies the onboarding of new team members.
- A standardized process helps establish security and better control over the infrastructure

**Cortex**: An open-source multi-framework tool that is flexible enough to be used as a model serving tool, as well as for purposes like model monitoring.

With its ability to address different machine learning workflows, it grants you full control over model management operations

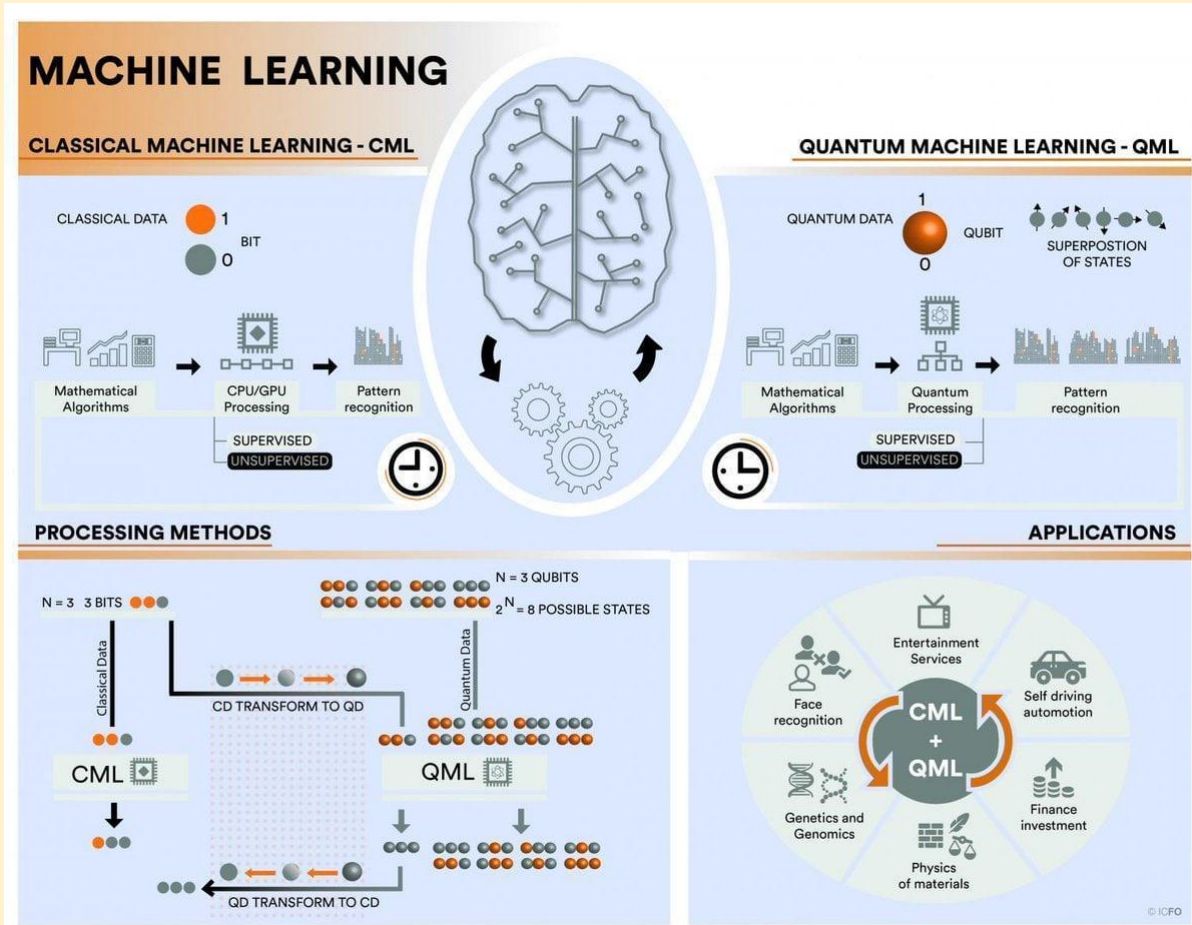
Why Cortex?

- Auto-scaling feature which allows APIs to be secure when network traffic fluctuates.
- Support for multiple platforms such as Keras, TensorFlow, Scikit-learn, PyTorch, etc.

- No downtime when models are being updated.

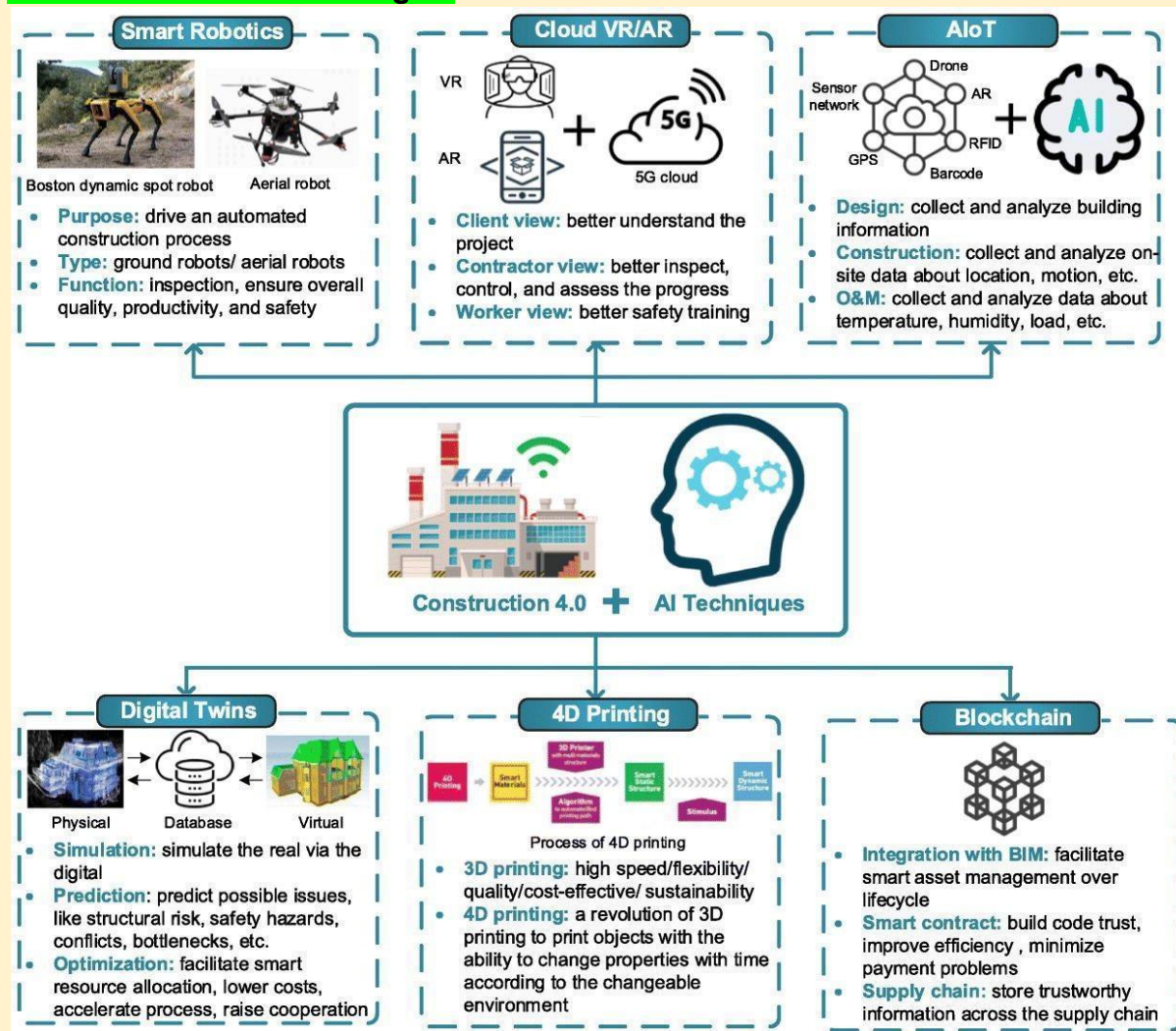
<https://lnkd.in/gTg4vi4J>

## Machine Learning





## New Generation Technologies



## 10 free websites To learn Web3 and Blockchain development

1. Pointer: Start learning the skills for the web3 programming economy and get paid while you learn.

<https://t.co/aQigudRk0U>

2. Learn Web3 DAO: LearnWeb3 is a high quality, holistic education platform to onboard you as a web3 developer. Join 40k+ builders and start your journey today

<https://t.co/UxVVFvbebi>

3. Buildspace: Home to the world's best web3 builders. Buildspace accelerates your builder journey into web3. Whether you're just starting out, a seasoned vet transitioning from web2, or thinking of building something cool, you can join Buildspace

<https://t.co/Tdtjvfesqw>

4. Solidity by Example: A great list of examples of Solidity programming languages.

<https://t.co/p54hKwILwM>

5. The Road to Web3: The Road to Web3 is a 10-week, self-paced program to bring you from Beginner to Advanced as a Web3 Blockchain Developer.

<https://t.co/m1rxUgkfkD>

6. Odyssey: Odyssey is one of the great platforms to learn Web3. They covers a range of topics such as NFTs, DeFi, DAO and etc..



<https://t.co/l83PwsaNng>

## 7. Questbook

Join 15,000+ developers in learning web3, earn crypto and NFTs on the way. It's a very good platform to learn blockchain and web3 development

<https://t.co/q69HkmYhx0>

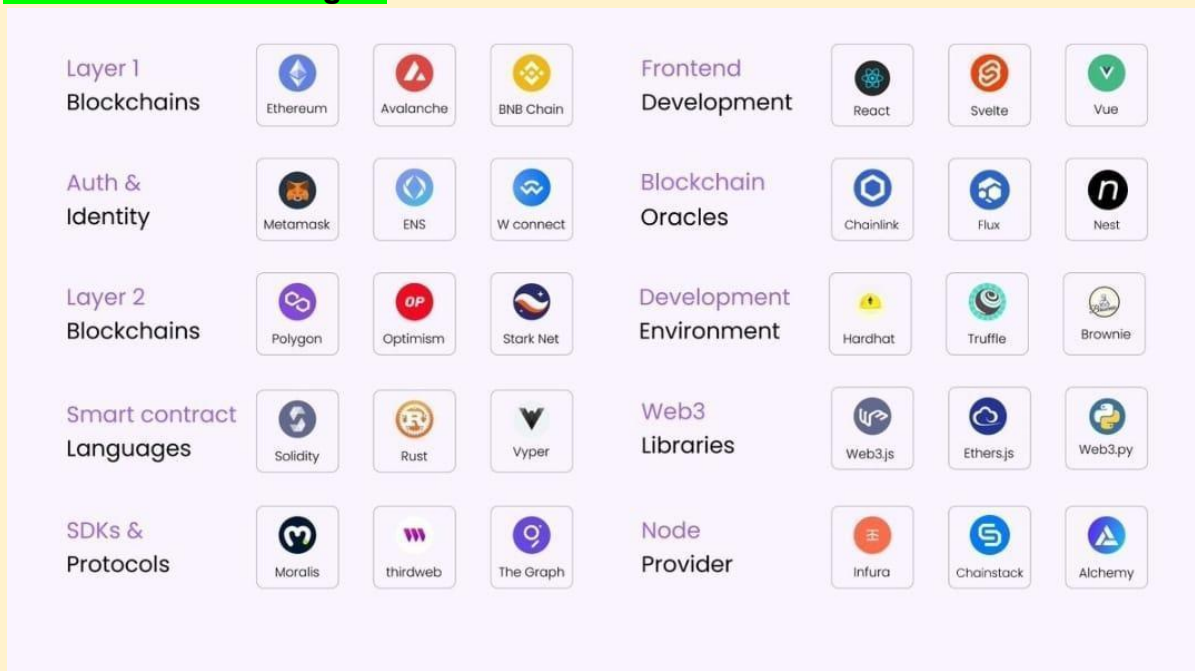
8. Smart Contract Best Practices: Smart contracts are complex and have the authority to allocate high-value resources between complex systems with huge financial loss at risk. This document provides a baseline knowledge of security considerations.

<https://t.co/UGuVRqdGHT>

10. Third Academy: Third Academy develops free courses for Web3 professions and business use cases. Whether you're looking to launch your own career or want to build directly in Web3, Third Academy guides you through engaging, non-traditional learning.

<https://t.co/dxMTkwx2WS>

## Blockchain Technologies



## Web3 Stack

Learn these to become as Web3 Professional.

1. Blockchains: The base of Web3 stack is the blockchains network. These are divided into 2 types. The first layer refers to the base level of blockchain(its main structure) such as Ethereum and BNB. However layer 2 refers to networks built on top of layer 1 blockchains.

2. Auth & Identity: Next is Authentication, these are the tools/wallets that is used to authenticate a user inside of applications. Metama\*k and Wallet connect are the most popular wallets.

3. Languages: In order to write smart contracts we need a programming Languages ... Popular ones are Solidity, Vyper and Rust. Each one has their own pros and cons.

4. SDKs: Building functionalities from scratch can be a little time-consuming, therefore we can use Web3 SDKs to make the development process much faster. thirdweb, Moralis are great examples.

5. Frontend Development: To develop the UI for Web3 application you would need a framework such as React, Angular or Vue. Developer can use any of them however for react there are many resources available online.

6. Blockchains Oracles: Blockchain oracles are tools that..... connect blockchains to external systems. They act as a bridge between blockchains and off chains.

7. Development Environment: This is the place that you write code. When it comes to writing smart contracts, you can either use Remix IDE a local development environment.

8. Web3 Libraries: These are used to interact with the blockchain from client. There are many libraries that does that but the most popular are Ethers.js, Web3.js, and Web3. py

9. Node Providers: Node provider offers a way to access information on blockchain without having your own local blockchain setup. Some of the fastest node providers are Chainstack, Infura and Alchemy.


Celestia: Celestia is the first modular consensus and data network blockchain that enables anyone deploy their own blockchain.


## Machine Learning with Signal Processing Techniques

ML Fundamentals

<https://ataspinar.com/2018/04/04/machine-learning-with-signal-processing-techniques/>

## Nasa's SpaceML Tool

Nasa's SpaceML Tool Introduces A New Six-Stage Pipeline To Automate The Classification Of Meteors From Non-Meteors Using Machine Learning. 

👉 First stage: Local devices at operator locations that capture sky data undertake local processing to evaluate whether a reported object is a meteor or a non-meteor.  Random Forest + CNN + LSTM

👉 Second Stage – Data Retrieval: The data is then retrieved from the remote site, which used to need bi-monthly in-person visits and the pickup of physical DVD media on which the data had been burned.

👉 Third stage – Processing: It is performed by Python scripts that interact with and automate CAMS' existing software stack, including MeteorCal, installed cameras, and star observations.

👉 Fourth stage – Calculation of coincidence: This process takes confirmed meteors and combines data from several cameras to create a trajectory, recognizing and automatically correcting irregularities in the video recording that could lead to inaccuracies.

👉 Fifth stage – Data clustering: It involves identifying outbursts and new showers that could suggest the presence of a long-period comet that was previously undiscovered. The pipeline can detect previously unidentified meteor shower groups and potential meteor outbursts

👉 Sixth Stage – Visualization Step: CAMS data that has gone through all five of the previous steps is transformed into a more accessible form to make it more widely available.

AI for Space 

#machinelearning #deeplearning #datascience #randomforest #convolutions #sequencemodel #spaceml #aiforspace #transformai #neuralverse #dataanalytics #BigData

### SpaceML Taps Satellite Images to Help Model Wildfire Risks

<https://blogs.nvidia.com/blog/2021/04/01/spaceml-taps-satellite-images-to-help-model-wildfire-risks/>

### The Best Open Source Chatbot Platforms in 2022

<https://verloop.io/blog/the-best-open-source-chatbot-platforms/>

### 13 Best AI Chatbot Development Framework & Platforms

<https://www.kommunicate.io/blog/chatbot-framework-platform/>

### 10 Best Chatbot Development Frameworks to Build Powerful Bots

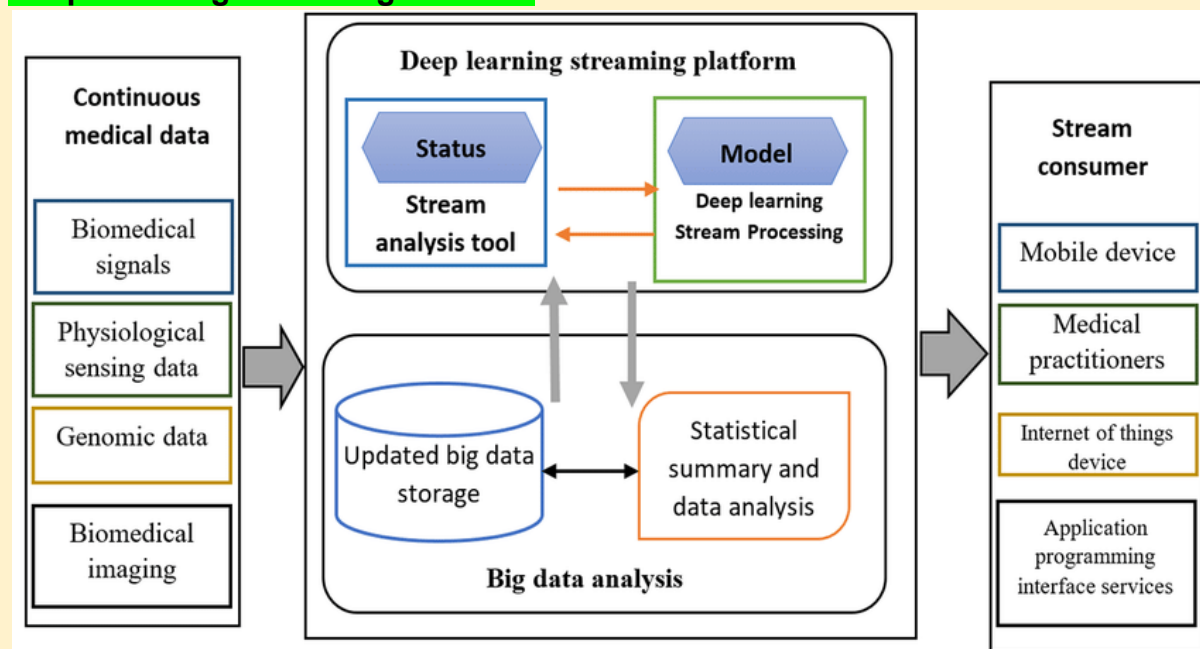
<https://geekflare.com/chatbot-development-frameworks/>

### Learning Representations of Geographic Locations From Unlabeled GPS Trajectories

Apple ML Researchers Propose A Self-Supervised Method For Learning Representations of Geographic Locations From Unlabeled GPS Trajectories to Solve Downstream Geospatial Computer Vision Tasks - MarkTechPost

<https://www.marktechpost.com/2022/08/09/apple-ml-researchers-propose-a-self-supervised-method-for-learning-representations-of-geographic-locations-from-unlabeled-gps-trajectories-to-solve-downstream-geospatial-computer-vision-tasks/>

### Deep Learning Streaming Platform



### Fake data Library

Do you ever need fake data for tests in #Python?

You should try the Faker package out then! It can generate all kinds of fake data!

```
>>> from faker import Faker
>>> fake = Faker()
```

```
>>> fake.name()
'Paul Lynn'
>>> fake.name()
'Keith Soto'
>>>fake.address() 'Unit 6944 Box 5854\nDPO AA 14829'
>>> fake.address()
'44817 Wallace Way Apt. 376\nSouth Ashleymouth, GA 03737'
```

### **Top Explainable AI Frameworks**

Top Explainable AI Frameworks For Transparency in Artificial Intelligence - MarkTechPost

<https://www.marktechpost.com/2022/08/09/top-explainable-ai-frameworks-for-transparency-in-artificial-intelligence/>

### **The Top 5 AI/Machine Learning blogs**









1. Google AI Blog - <https://t.co/VjyDvO6clr>
2. Bair by Berkely AI - <https://t.co/lJ89qFHgyk>
3. OpenAI - <https://t.co/5K6rbkWD1M>
4. Analytics Vidhya - <https://t.co/ykUsTNkLaT>
5. The AI Summer - <https://t.co/TvCZIGnQt0>

## Industry application of NLP



## Industry Application

---








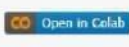







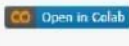
























-  Speech Recognition
-  Machine Translation
-  Natural Language Generation
-  Part-of-Speech Tagging
-  Parsing
-  Optical Character Recognition
-  Grammer Induction
-  Information Retrieval

## Courses

---

1. Natural Language Processing Specialization :  
<https://lnkd.in/dswF7VtT>
2. Introduction to Huggingface Course on NLP :  
[https://lnkd.in/dDP3V\\_wm](https://lnkd.in/dDP3V_wm)

[#naturallanguageprocessing](#) [#artificialintelligence](#)  
[#machinelearning](#)

|    |                               |      |                                                                                     |                                                                                     |                                                                                      |                                                                                                     |
|----|-------------------------------|------|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| 15 | VILBERT                       | 2019 |  |  |  |  Open in Colab |
| 16 | Sparse Transformer            | 2019 |  |  |  |  Open in Colab |
| 17 | Levenshtein Transformer       | 2019 |  |  |  |  Open in Colab |
| 18 | CTRL                          | 2019 |  |  |  |  Open in Colab |
| 19 | VideoBERT                     | 2019 |  |  |  |  Open in Colab |
| 20 | Compressive Transformer       | 2019 |  |  |  |  Open in Colab |
| 21 | CuBERT                        | 2019 |  |  |  |  Open in Colab |
| 22 | BP-Transformer                | 2019 |  |  |  |  Open in Colab |
| 23 | Adaptively Sparse Transformer | 2019 |  |  |  |  Open in Colab |
| 24 | Sandwich Transformer          | 2019 |  |  |  |  Open in Colab |

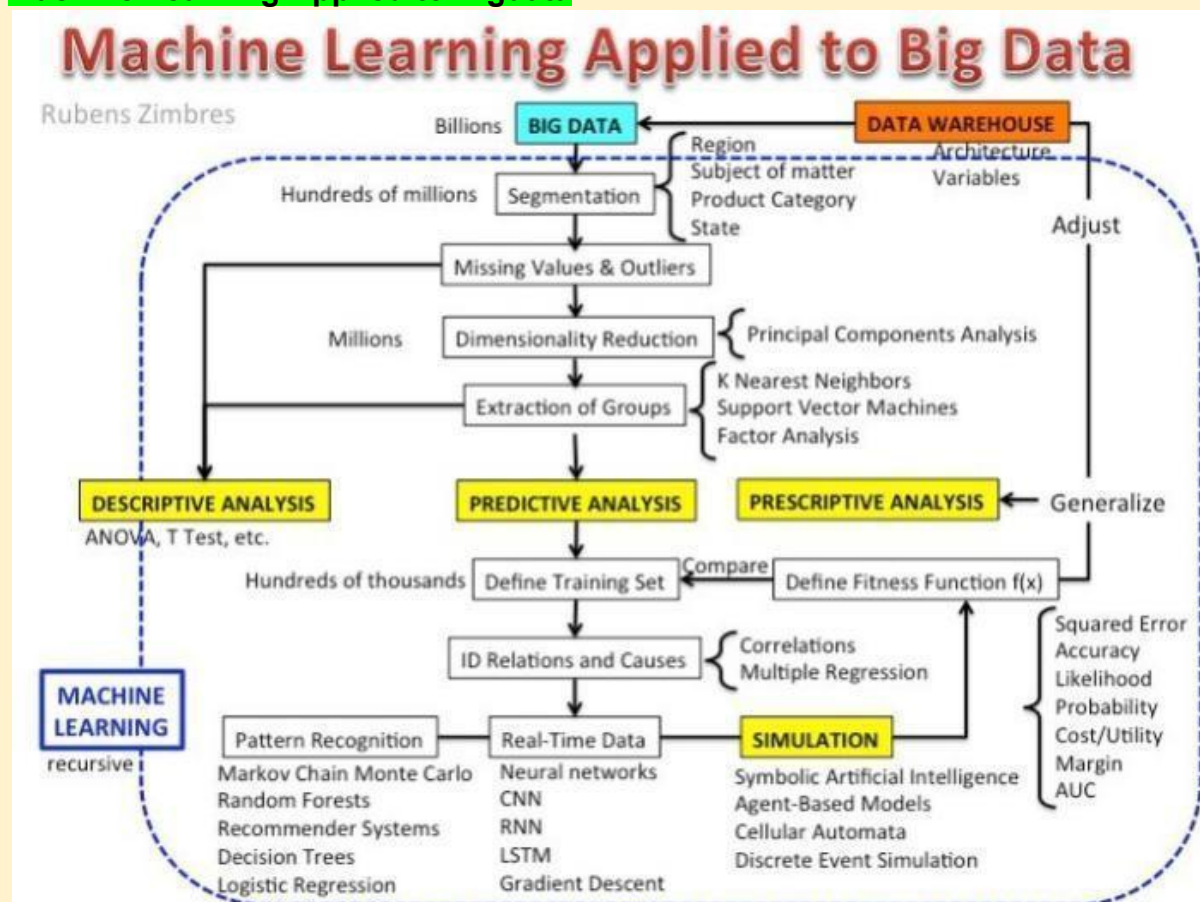
## Airflow vs Luigi vs Argo vs Kubeflow vs MLFlow

<https://www.datarevenue.com/en-blog/airflow-vs-luigi-vs-argo-vs-mlflow-vs-kubeflow>

## ML Platforms: Dataiku vs. Alteryx vs. Sagemaker vs. Datarobot

<https://www.datarevenue.com/en-blog/ml-platforms-dataiku-vs-alteryx-vs-sagemaker>

## Machine Learning Applied to Bigdata



## Top Responsible AI (Artificial Intelligence) Tools in 2022

<https://www.marktechpost.com/2022/08/06/top-responsible-ai-artificial-intelligence-tools-in-2022/>

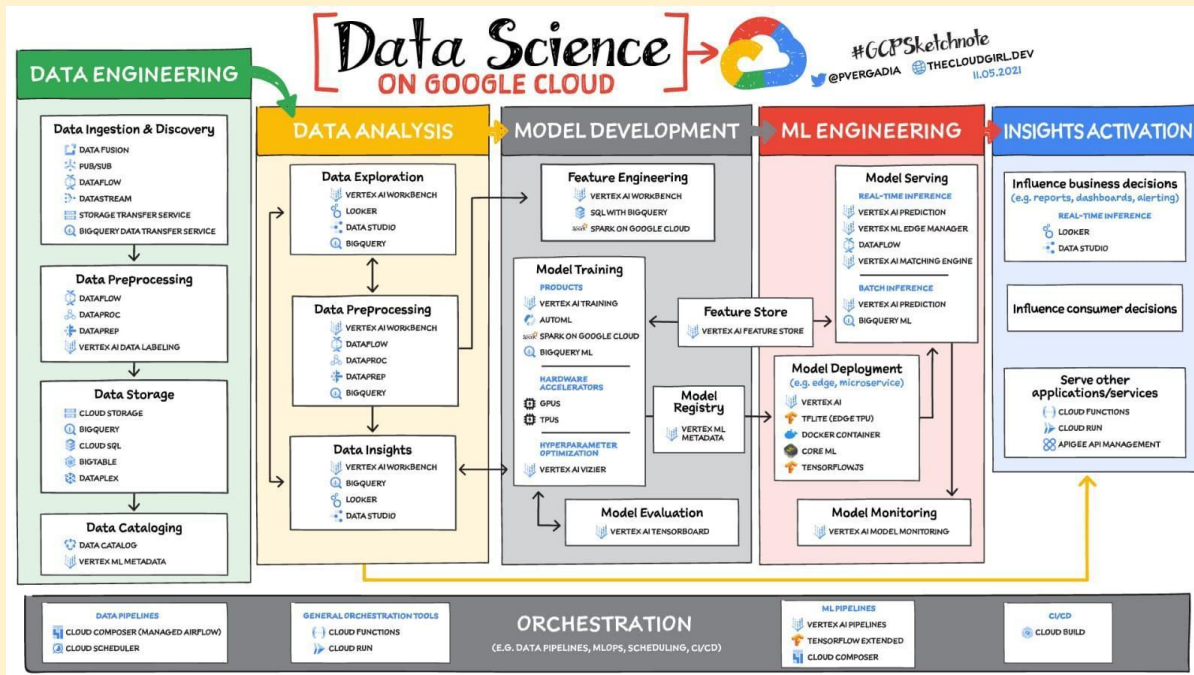
## Machine Learning Zoomcamp

<https://github.com/alexeygrigorev/mlbookcamp-code/tree/master/course-zoomcamp>

## 90+ Data Science Projects You Can Try with Python

<https://thecleverprogrammer.com/2022/03/09/data-science-projects/>

<https://plainenglish.io/blog/114-data-science-projects>



## How to implement data-centric AI in NLP

<https://utomorezadwi.medium.com/how-to-implement-data-centric-ai-in-nlp-512972a5676b>

## Issue #190 - THE ML ENGINEER

<https://www.linkedin.com/pulse/issue-190-ml-engineer-alejandro-saucedo>

## Machine Learning in Alteryx with PyCaret

[https://pycaret.gitbook.io/docs/learn-pycaret/official-blog/machine-learning-in-alteryx-with-pycaret?utm\\_medium=social&utm\\_source=twitter&utm\\_campaign=postfity&utm\\_content=postfity7f800](https://pycaret.gitbook.io/docs/learn-pycaret/official-blog/machine-learning-in-alteryx-with-pycaret?utm_medium=social&utm_source=twitter&utm_campaign=postfity&utm_content=postfity7f800)

## Machine Learning in Tableau with PyCaret

[https://pycaret.gitbook.io/docs/learn-pycaret/official-blog/machine-learning-in-tableau-with-pycaret?utm\\_medium=social&utm\\_source=linkedin&utm\\_campaign=postfity&utm\\_content=postfity59715](https://pycaret.gitbook.io/docs/learn-pycaret/official-blog/machine-learning-in-tableau-with-pycaret?utm_medium=social&utm_source=linkedin&utm_campaign=postfity&utm_content=postfity59715)

## Machine Learning for Cyber Security

Lectures: <https://www.youtube.com/playlist?list=PL74sw1ohGx7GHqDHckXZeqMQBVUTMrVLE>

#ArtificialIntelligence #DeepLearning #CyberSecurity

## Google AI Introduces LocoProp

Google AI Introduces 'LocoProp,' A New Machine Learning Framework That Reconceives A Neural Network As A Modular Composition of Layers

<https://www.marktechpost.com/2022/08/01/google-ai-introduces-locoprop-a-new-machine-learning-framework-that-reconceives-a-neural-network-as-a-modular-composition-of-layers/?amp>

## Distributed Training in TensorFlow with AI Platform & Docker

<https://sayak.dev/distributed-training/>

## Datashader

⚡ "Datashader, HoloViews, and hvPlot accept a variety of CPU or GPU data structures (e.g. Pandas/Dask/cuDF dataframes or Xarray/CuPy n-dimensional arrays).

For cuDF and CuPy, the main computations are then done on the GPU or cluster, without any extra work by the user."



See [https://datashader.org/user\\_guide/Performance.html](https://datashader.org/user_guide/Performance.html) for a chart listing which combinations of libraries and data structures are currently supported for CPUs and GPUs and distributed computing.

## Serve hundreds to thousands of ML models : Architectures from industry

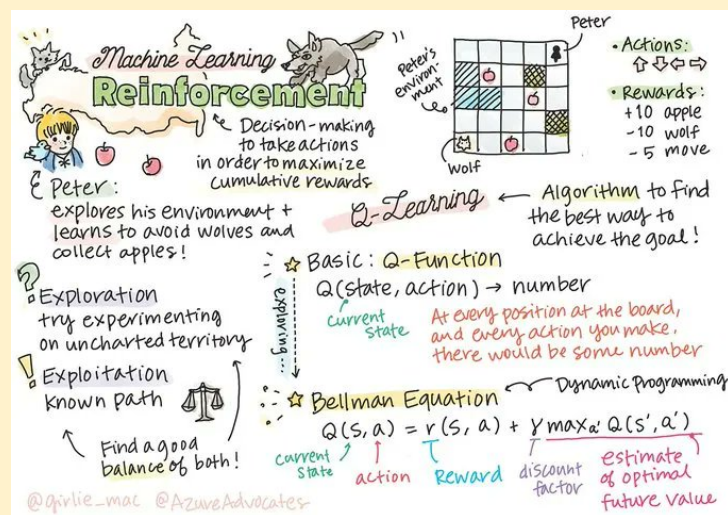
<https://towardsdatascience.com/serve-hundreds-to-thousands-of-ml-models-architectures-from-industry-bf3d9474d427>

## An Exhaustive Read & Watch List for AI

Read List: <https://aman.ai/read/>

Watch List: <https://aman.ai/watch/>

## ML Concepts



## Fairness in Machine Learning

**Fairness-related harms**

Unfairness = negative impacts for group of people such as those defined in terms of

- race
- age
- gender
- disability status

**Harms:**

- ✦ Allocation
- ✦ Quality of service
- ✦ Stereotyping
- ✦ Denigration
- ✦ Over- / under- representation

**Complex sociotechnical Challenges**

- ✗ purely social
- ✗ purely technical

**Assessment & mitigation**

- ♥ Identify the harm (+benefits)
- ♥ Identify the affected groups
- ♥ Define fairness metrics

**False negatives**  
**False positives**

|        | False | True | Loss |
|--------|-------|------|------|
| Actual | 0.35  | 0.27 | 6234 |
| Wanted | 0.24  | 0.35 | 3124 |

**Fairlearn**  
fairlearn.github.io

Training Set → input → BLACK Box → output → Oh no! Reject all racoons!

Bias: Cats are great! Racoons are bad!

@AzureAdvocates @girlie\_mac

## Machine Learning Time Series Forecasting

Looking at data overtime to forecast or predict future values based on patterns or recurring trends.

**Examples:**

- ☁ Weather forecast
- 🛒 Holiday season sales
- 📈 Stock market
- 📶 online user traffic

**Data Characteristics:**

- ♥ Seasonality
- ♥ Outliers
- ♥ long-run cycle
- ♥ Constant variance
- ♥ abrupt changes

**Using ARIMA**

- Auto regressive: previous values in your data "lags" long memory models
- integrated: Stationarity
- moving average: previous hidden shocks or errors in your model

**Short memory models**

**Differencing:** look at the  $\Delta$  from one period to another

**Stat models**  
Define model w/ SARIMAX()

LIB

@AzureAdvocates @girlie\_mac

### Reco: Introduction to Probability for Data Science

<https://probability4datascience.com/>

### Supervised Clustering: How to Use SHAP Values for Better Cluster Analysis

<https://www.aidancooper.co.uk/supervised-clustering-shap-values/>

### Github: 800 free #ComputerScience classes

800 free #ComputerScience classes you can take online right now, with video lectures:

<https://github.com/Developer-Y/cs-video-courses>

#BigData #DataScience #AI #ArtificialIntelligence #MachineLearning #DeepLearning #ComputerVision #Robotics #QuantumComputing

### Mathematics of Big Data and Machine Learning @MIT

Mathematics of Big Data and Machine Learning course materials by @MIT are free and open for everyone!

Learn here: <https://t.co/yDbtUEEEof>

### Linear Algebra @MIT

Linear Algebra course by @MIT is free and open for everyone!

Learn the basics: <https://t.co/WFXMliBUld>



## **CS109A Data Science @Harvard**

CS109A Data Science course materials @Harvard are free and open for everyone!

Learn here: <https://t.co/CriMdwDMN5>

## **NLP with Deep Learning by Stanford**

"NLP with Deep Learning" by Stanford is free and open!

<https://t.co/9juSBqZcxF>

## **Google Vizier: A Service for Black-Box Optimization**

<https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/46180.pdf>

## **Tinker With a Neural Network in Browser**

<https://playground.tensorflow.org/#activation=tanh&batchSize=10&dataset=circle&regDataset=reg-plane&learningRate=0.03&regularizationRate=0&noise=0&networkShape=4,2&seed=0.38574&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>

## **Hand Gesture Recognition - Pretrained**

<https://www.google.com/amp/s/techvidvan.com/tutorials/hand-gesture-recognition-tensorflow-opencv/%3famp=1>

<https://towardsdatascience.com/tutorial-using-deep-learning-and-cnns-to-make-a-hand-gesture-recognition-model-371770b63a51>

<https://youtu.be/doDUihpj6ro>

## **Blockchain Tech**

<https://thehackweekly.com/blockchain-technology-for-beginners/>

## **Distributed GNN training**

Large-scale graph datasets have called for distributed GNN training. However, existing approaches involve heavy communication and memory overhead while failing to protect data privacy in the process.

Researchers incorporated federated learning in distributed GNN algorithms to preserve privacy, significantly cut back on training overhead while maintaining desirable performance. For more information, please check out paper at

<https://t.co/hSokQuRgnd>.

## **Serverless computing for GNN**

Dorylus enables affordable, scalable, and accurate GNN training using distributed CPU servers and serverless threads. Through computation separation, Dorylus allows serverless computing to provide a scalable, efficient, and low-cost scheme for GNN training.

Paper: <https://t.co/BVJeXltSwy>

Code: <https://t.co/yLhiu53eDf>

Training graph neural networks is challenging, since the neural network computation relies on expensive high-end GPUs but the limited GPU memory cannot scale to today's billion-edge graphs.

Existing ML frameworks optimize DNN computation by considering only fully equivalent transformations. This approach preserves models' functionality but misses transformations that improve DNN performance but only maintain partial equivalence.

### **Graph Analysis to Fight Fraud**

How PayPal Uses Real-time Graph Database and Graph Analysis to Fight Fraud

<https://medium.com/paypal-tech/how-paypal-uses-real-time-graph-database-and-graph-analysis-to-fight-fraud-96a2b918619a>

### **Transfer Learning Colab Note**

<https://colab.research.google.com/drive/1y39PQs0AG5NLkCnDja6syYrcn2Fhx8Ek?usp=sharing>

### **Introduction to Vertex AI \ Google Cloud**

<https://cloud.google.com/vertex-ai/docs/start/introduction-unified-platform>

### **Continuous Integration for Machine Learning: Testing ML Models**

<https://ds3lab.ghost.io/ci/>

### **Computer Vision Nanodegree**

[https://drive.google.com/folderview?id=1770Z6Rw8tSq2iXjS2X\\_MoJwUd6EuZxCq](https://drive.google.com/folderview?id=1770Z6Rw8tSq2iXjS2X_MoJwUd6EuZxCq)

### **Sign up: Kaggle x Scale AI offer**

<https://scale.com/kaggle>

### **Seminal Work in Data Science**

- 1) The Design of Experiments By R.A Fisher - <https://home.iitk.ac.in/~shalab/anova/DOE-RAF.pdf>
- 2) The Fischer, Neyman-Pearson Theories of Testing Hypothesis - <https://digitalassets.lib.berkeley.edu/sdtr/ucb/text/333.pdf>
- 3) Statistical Inference - <http://home.ustc.edu.cn/~zt001062/MaStmaterials/George%20Casella&Roger%20L.Berger--Statistical%20Inference.pdf>
- 4) Generalized Linear Models (GLM) J.A Nelder et al - <https://www.utstat.toronto.edu/~brunner/oldclass/2201s11/readings/glmbook.pdf>
- 5) A Mathematical Theory of Communication - <https://people.math.harvard.edu/~ctm/home/text/others/shannon/entropy/entropy.pdf>, <https://web.mit.edu/6.933/www/Fall2001/Shannon1.pdf>
- 6) Latent Dirichlet Allocation - <https://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf>
- 7) Efficient Estimation of Word Representations in Vector Space (Word2vec) - <https://arxiv.org/abs/1301.3781?context=cs.CL>
- 8) Random-Effects Models for Serial Observations with Binary Response (Nan Laird et al) - <https://www.stat.cmu.edu/~brian/463-663/week06/some%20nice%20papers/stiratelli-laird-ware-biometrics-1984.pdf>
- 9) Maximum Likelihood from Incomplete Data via the EM Algorithm (Nan Laird et al) - <https://web.mit.edu/6.435/www/Dempster77.pdf>
- 10) Classification and Regression Trees - <https://pages.stat.wisc.edu/~loh/treeprogs/guide/LohISI14.pdf>
- 11) The Origins of Logistic Regression - <https://papers.tinbergen.nl/02119.pdf>
- 12) Random Forests (Leo Breiman) - <https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf>
- 13) Probability Theory: The Logic of Science (E T Jaynes) - <http://www.med.mcgill.ca/epidemiology/hanley/bios601/GaussianModel/JaynesProbabilityTheory.pdf>
- 14) The Elements of Statistical Learning - <https://hastie.su.domains/ElemStatLearn/>
- 15) An Introduction to Statistical Learning - <https://www.statlearning.com/>
- 16) Bayesian Data Analysis (Andrew Gelman) - <http://www.stat.columbia.edu/~gelman/book/>
- 17) Geoffrey Hinton's work (Google scholar link)- <https://scholar.google.com/citations?user=JicYPdAAAAAJ>

18) Robert Tibshirani's work (Google scholar link) -

[https://scholar.google.com/citations?user=ZpG\\_cJwAAAAJ&hl=en](https://scholar.google.com/citations?user=ZpG_cJwAAAAJ&hl=en)

19) Bradley Efron – Prediction, Estimation and Attribution -

<https://www.fox.temple.edu/wp-content/uploads/2021/07/Efron-2020-JASA-wdiscussion.pdf>

20) Statistical Modeling Two cultures : Leo Brieman -

<https://projecteuclid.org/journals/statistical-science/volume-16/issue-3/Statistical-Modeling--The-Two-Cultures-with-comments-and-a/10.1214/ss/1009213726.full>

21) What are the most important statistical ideas of the past 50 years (Andrew Gelman, Akivehtari)-

<https://arxiv.org/abs/2012.00174>

22) What is not what in statistics (Louis Guttman) -

<http://users.cla.umn.edu/~nwaller/prelim/guttmannotwhatinstatistics.pdf>

23) Forecasting: principles and practice - Rob J Hyndman and George Athanasopoulos -

<https://otexts.com/fpp3/>

24) Introduction to econometrics – Woolridge -

[https://cbpbu.ac.in/userfiles/file/2020/STUDY\\_MAT/ECO/2.pdf](https://cbpbu.ac.in/userfiles/file/2020/STUDY_MAT/ECO/2.pdf)

### **Metaverse beyond the hype**

Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy - ScienceDirect

<https://www.sciencedirect.com/science/article/pii/S0268401222000767>

### **Vertex AI: Qwik Start \ Google Cloud Skills Boost**

<https://www.cloudskillsboost.google/focuses/18940?parent=catalog>

### **How Azure Machine Learning works (v2) - Azure Machine Learning \ Microsoft Docs**

<https://docs.microsoft.com/en-us/azure/machine-learning/concept-azure-machine-learning-v2>

### **Github probml/pml2-book**

<https://github.com/probml/pml2-book/releases/tag/2022-08-15>

### **Probabilistic ML**

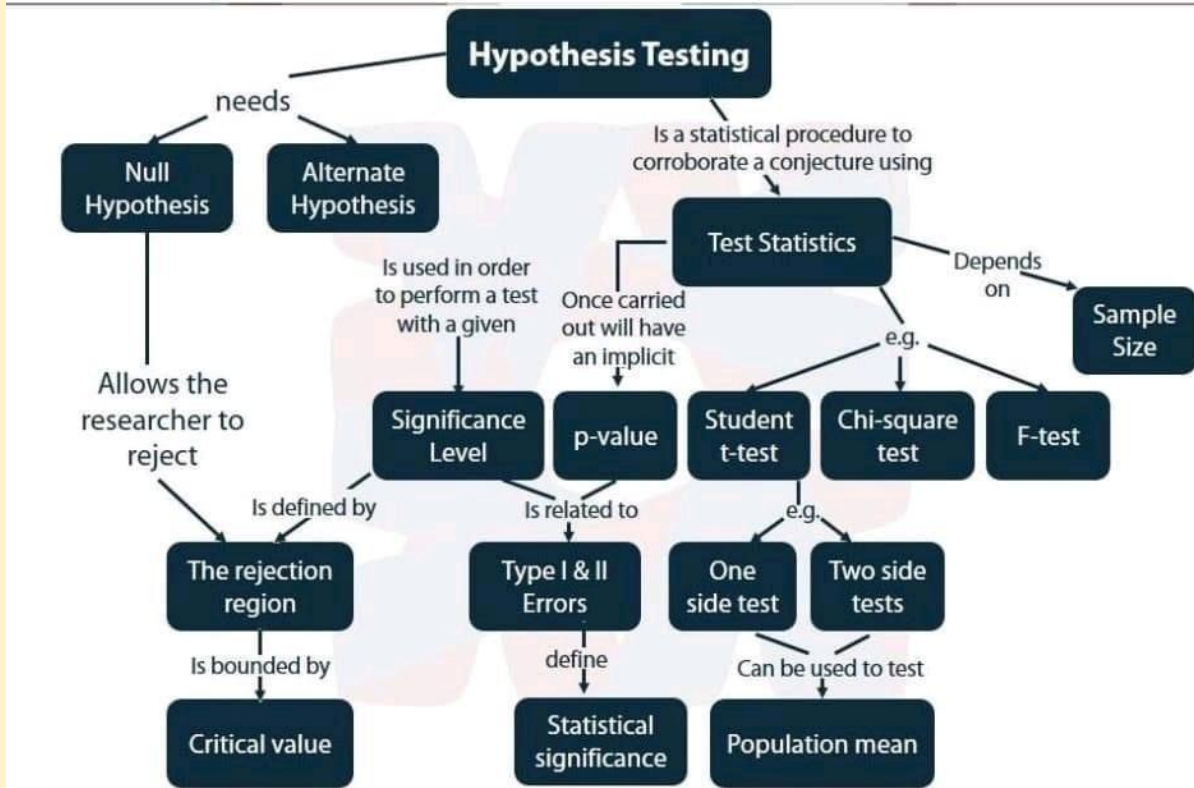
<https://probml.github.io/pml-book/book2.html>

### **Systematic Evaluation of Causal Discovery in Visual Model Based Reinforcement Learning**

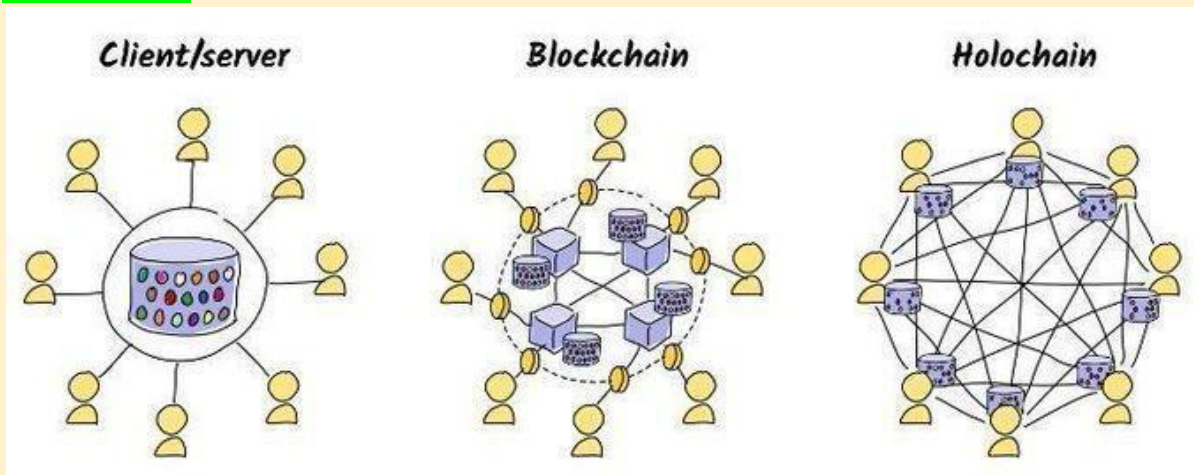
<https://arxiv.org/abs/2107.00848>

## Hypothesis Testing

# Hypothesis Testing



## Architecture



## The Basics - Holochain Docs

[https://developer.holochain.org/concepts/1\\_the\\_basics/](https://developer.holochain.org/concepts/1_the_basics/)

## Understanding AlphaZero Neural Network's SuperHuman Chess Ability

<https://www.marktechpost.com/2021/12/16/understanding-alphazero-neural-networks-superhuman-chess-ability/>

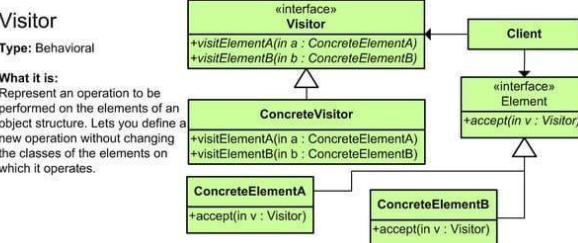
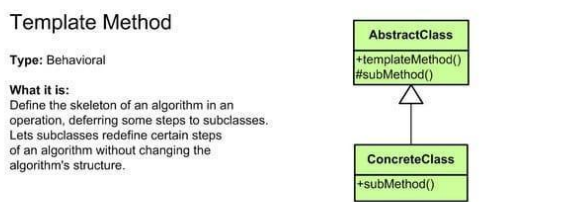
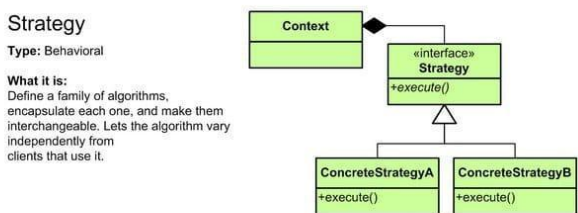
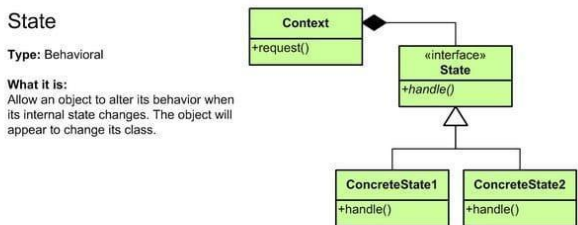
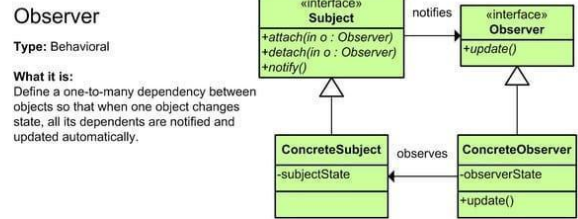
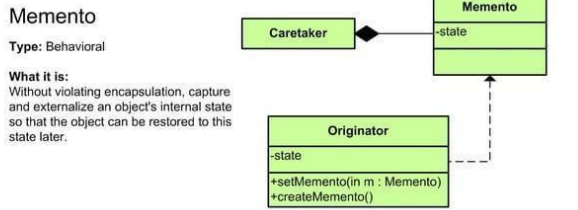
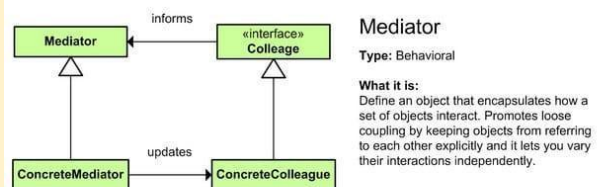
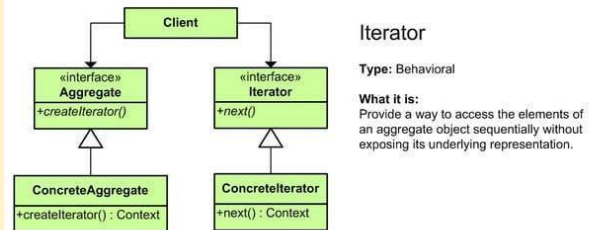
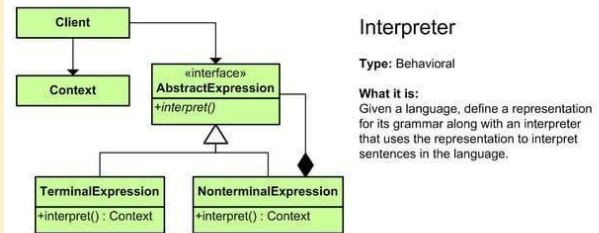
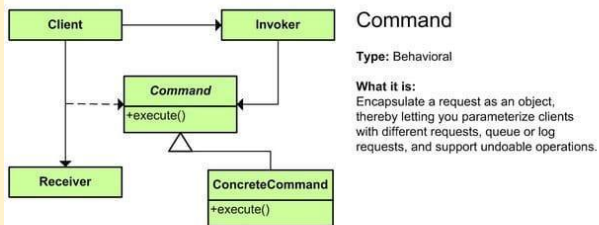
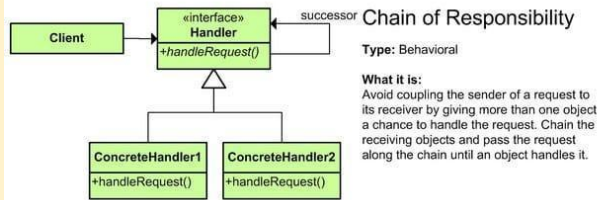


# Flink Vs. Spark: Difference Between Flink and Spark

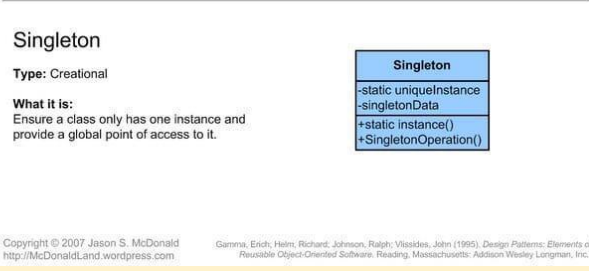
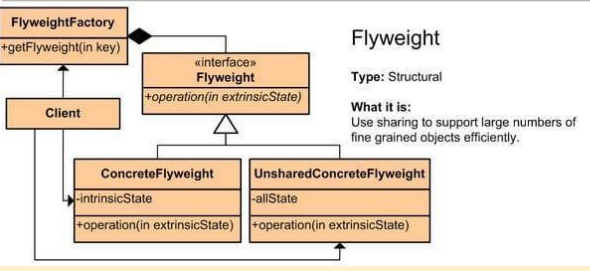
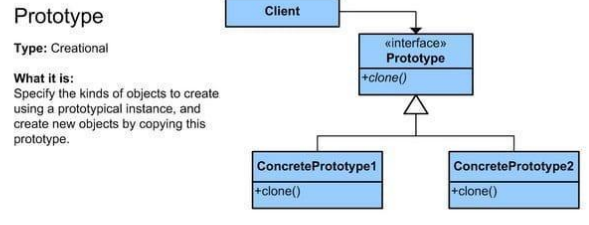
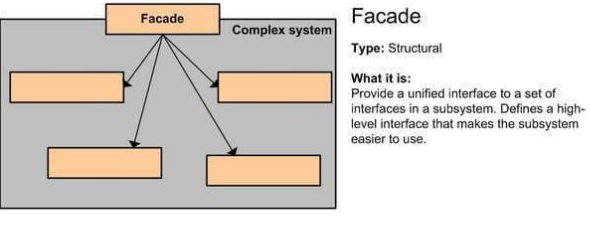
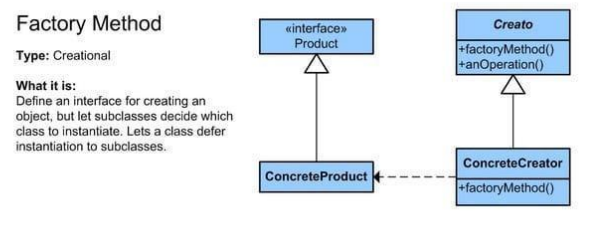
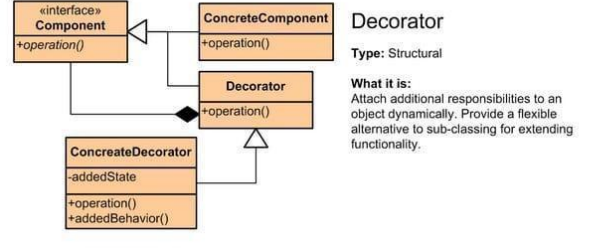
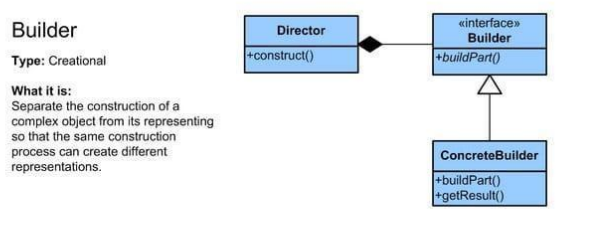
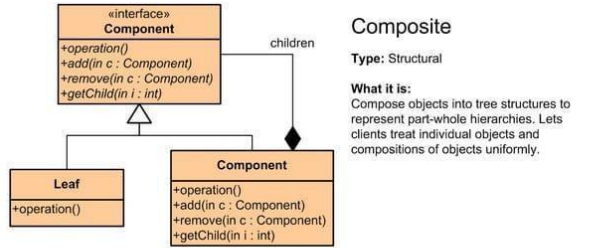
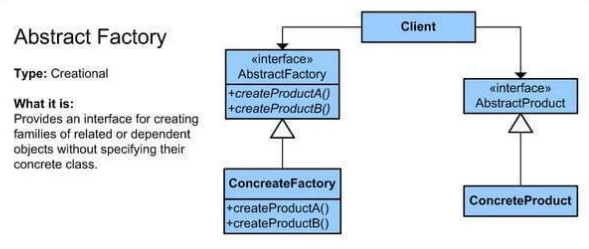
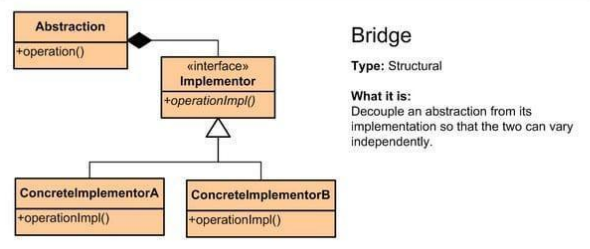
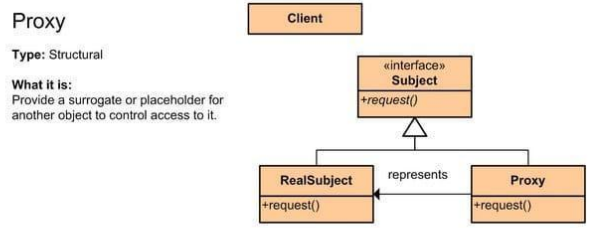
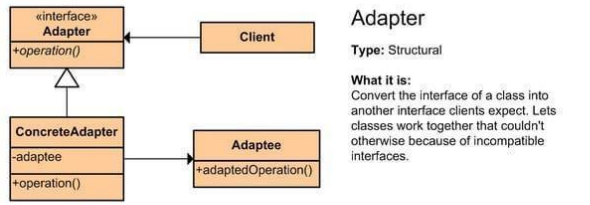
<https://www.upgrad.com/blog/flink-vs-spark/>

## Design Patterns

- |                                  |                         |                          |
|----------------------------------|-------------------------|--------------------------|
| <b>C</b> Abstract Factory        | <b>S</b> Facade         | <b>S</b> Proxy           |
| <b>S</b> Adapter                 | <b>C</b> Factory Method | <b>B</b> Observer        |
| <b>S</b> Bridge                  | <b>S</b> Flyweight      | <b>C</b> Singleton       |
| <b>C</b> Builder                 | <b>B</b> Interpreter    | <b>B</b> State           |
| <b>B</b> Chain of Responsibility | <b>B</b> Iterator       | <b>B</b> Strategy        |
| <b>B</b> Command                 | <b>B</b> Mediator       | <b>B</b> Template Method |
| <b>S</b> Composite               | <b>B</b> Memento        | <b>B</b> Visitor         |
| <b>S</b> Decorator               | <b>C</b> Prototype      |                          |







Next to writing cleaner code, knowing some popular **design patterns** also helps you quickly structure problems in your head when developing any kind of software solution. 🧠

Here are some **great free resources to brush up on design patterns** and their implementation in **#Python** :

- <https://lnkd.in/g5nZ7eMY> (my favorite! pic related)
- <https://lnkd.in/gtXnQZ-c> (github repo)
- <https://lnkd.in/g7N2umFw>
- <https://lnkd.in/gsk7-eNT>

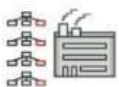
Have fun!

---

**Follow me** and click 🔔 for curated content on productionizing **#MachineLearning** and **#MLOps** .  
Let's take **#DataScience** from Notebook to Production!

## The Catalog of Python Examples

### Creational Patterns



#### Abstract Factory ★★★

Lets you produce families of related objects without specifying their concrete classes.

[Main article](#)

[Code example](#)

[Usage in Python](#)



#### Builder ★★★

Lets you construct complex objects step by step. The pattern allows you to produce different types and representations of an object using the same construction code.

[Main article](#)

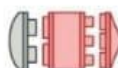
[Code example](#)

[Usage in Python](#)



#### Factory Method ★★★

### Structural Patterns



#### Adapter ★★★

Allows objects with incompatible interfaces to collaborate.

[Main article](#)

[Code example](#)

[Usage in Python](#)



#### Bridge ☆☆☆

Lets you split a large class or a set of closely related classes into two separate hierarchies—abstraction and implementation—which can be developed independently of each other.

[Main article](#)

[Code example](#)

[Usage in Python](#)



#### Composite ★★★

## Web2 vs web3 platforms

Web2 Web3 Supporting Platforms

AWS S3 → Arweave

Godaddy → ENS

Medium → Sigle

Paypal → Metamask

Spotify → Audius

Upwork → Braintrust

Google → Presearch

Twitter → Lenster

AWS Managed Blockchain → Chainstack

## 100+ gpt3 projects

<https://gptcrush.com>

## Python Design Patterns

Design Patterns - Python

<https://refactoring.guru/design-patterns/python>

A collection of design patterns/idioms in Python

<https://github.com/faif/python-patterns>

Python Design Patterns Guide

<https://python-patterns.guide/>

<https://www.geeksforgeeks.org/python-design-patterns/>

## Tutorial of fundamental remote sensing and GIS methodologies using open source software in python

<https://github.com/patrickcgray/open-geo-tutorial>

## Mining Trends in Data Science Blog Headlines

Synonym Extraction with Shallow Transfer Learning from CommonSense Knowledge graph embeddings and Fasttext embeddings pretrained on 1.3 billion Google Queries

<https://www.xn--knut-jgersberg-bib.com/blog/2022-08-08-mining-trends-in-data-science-blog-headlines/>

## Best softwares for #latex like data visualisation programs

Best softwares for #latex like data visualisation programs for Scientific Researchers and Data Scientists  
#DataScience #MachineLearning #Datavisualization

### 1. R studio

In addition to excel the best open source scientific tool is Rstudio. The libraries like ggplot2 make it unique for preparing the publication ready graphs. You can use Python, SQL , C++ in it as well in the form of Rmarkdown code chunks

### 2. Origin pro

Origin pro has a unique feature of tikz like picture graphical interface which comes in the form of a GUI without using the latex code. It can connect to other applications like MATLAB, Excel, Python, R. Peak and surface fitting techniques are added as an extra feature

### 3. Minitab

Minitab is very popular in statistics community alongwith other softwares such as SPSS,SAS,STATA, JMP. It has built in tools to deliver data visualisations and data driven decision process. It provides other optional features of minitab connect, predictive modeler etc

#### 4.Sigma Plot

If you want to integrate and convert your spreadsheets into publication ready graphs, sigma plot is best for you. In the financial and medical community it is already very popular. A User interface includes options such as symbols, graph planes, Reference lines etc.

#### 5.GraphPad prism

For the analysis of categorical and quantitative data this software as a drop down feature with 2 window named "which analysis and Analyze which dataset?" These two options are very handy when you can get p-value in minutes for all the statistical tests.

#### 6.gnuplot

If you are familiar with linux commands from terminal and you like using VIM, you will love this tool. You can tweak plot with just 1 line of code. Unfortunately it is not available in any software and only option is to run from command line.

#### 7.Matlab, Python

Of course, we cannot miss the popular tools such as MATLAB, Python for data visualisation. Python libraries such as Matplotlib, Seaborn, Altair, plotly are very popular in scientific community. Both of these can have inline, subplots and windowed graphs.

### **Knowledge Graph-based Perspective on Named Entity Disambiguation**

A Knowledge Graph-based Perspective on Named Entity Disambiguation

#AI #ML #IoT #DataScience #BigData #Analytics

<https://link.medium.com/o5s4Zrz9wsb>

<https://ieeexplore.ieee.org/document/8455418>

### **Google Cloud Anthos Series - 1: Introduction to Anthos**

<https://blog.searce.com/google-cloud-anthos-series-introduction-to-anthos-ee35a528b1e1>

### **Securing apps for Googlers using Anthos Service Mesh**

<https://cloud.google.com/blog/topics/developers-practitioners/securing-apps-googlers-using-anthos-service-mesh>

### **Anthos supports NVIDIA GPUs**

<https://cloud.google.com/blog/products/compute/anthos-supports-nvidia-gpus>

### **Animated Chart Presentation in Jupyter Notebook.**

<https://github.com/vizzuhq/ipyvizzu-story>



## Word Error Rate in STT



Cobus Greyling · 1st

NLP/NLU, LLM's, Chatbots, Voicebots...

31m · 🌐

Lately voicebots have received much attention, an integral part of voicebots is converting speech to text (ASR, STT). The commonly used measurement for success in TTS is Word Error Rate (WER).

Vendors often have startling claims when it comes to WER, in this article I try to define criteria which can be used when compiling a set of recordings for benchmarking and measuring WER. And how to have a representative WER measurement.

## Measuring Word Error Rate for ASR

Incorrectly identified words fall into three categories:

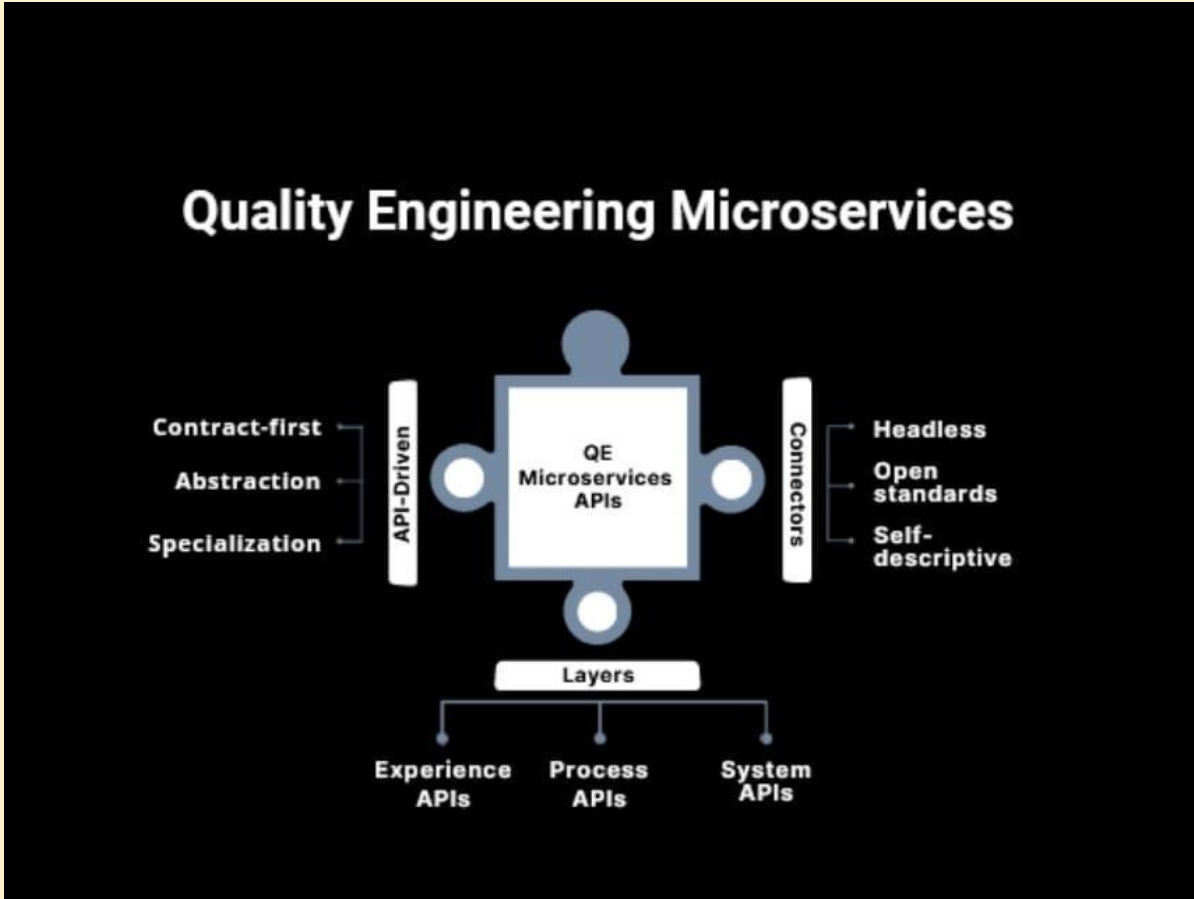
- **Insertion (I)**: Words that are incorrectly added in the hypothesis transcript
- **Deletion (D)**: Words that are undetected in the hypothesis transcript
- **Substitution (S)**: Words that were substituted between reference and hypothesis

$$WER = \frac{I + D + S}{N} * 100\%$$





## Quality Engineering Microservices



## Automate Machine Learning using Databricks AutoML

<https://towardsai.net/p//automate-machine-learning-using-databricks-automl%e2%80%8a-%e2%80%8aa-glass-box-approach-and-mlflow>

## AI for Beginners

# AI for Beginners

by microsoft

ethics

*Good Old Fashioned AI*

- ★ Knowledge Representation
- ★ Ontologies
- ★ Expert Systems
- ★ Genetic Algorithms

*Neural Networks / Deep Learning*

- ★ Computer Vision
- ★ Convolutional networks
- ★ GAN / VAE
- ★ NLP

*Machine Learning*

- Word2Vec
- Embedding
- Recurrent Networks
- Transformers

@girlie\_mac @shwars

aka.ms/ml-beginners

## Bigdata architecture

# Big Data Classification And Architecture

Two critical points to consider in big data architecture are

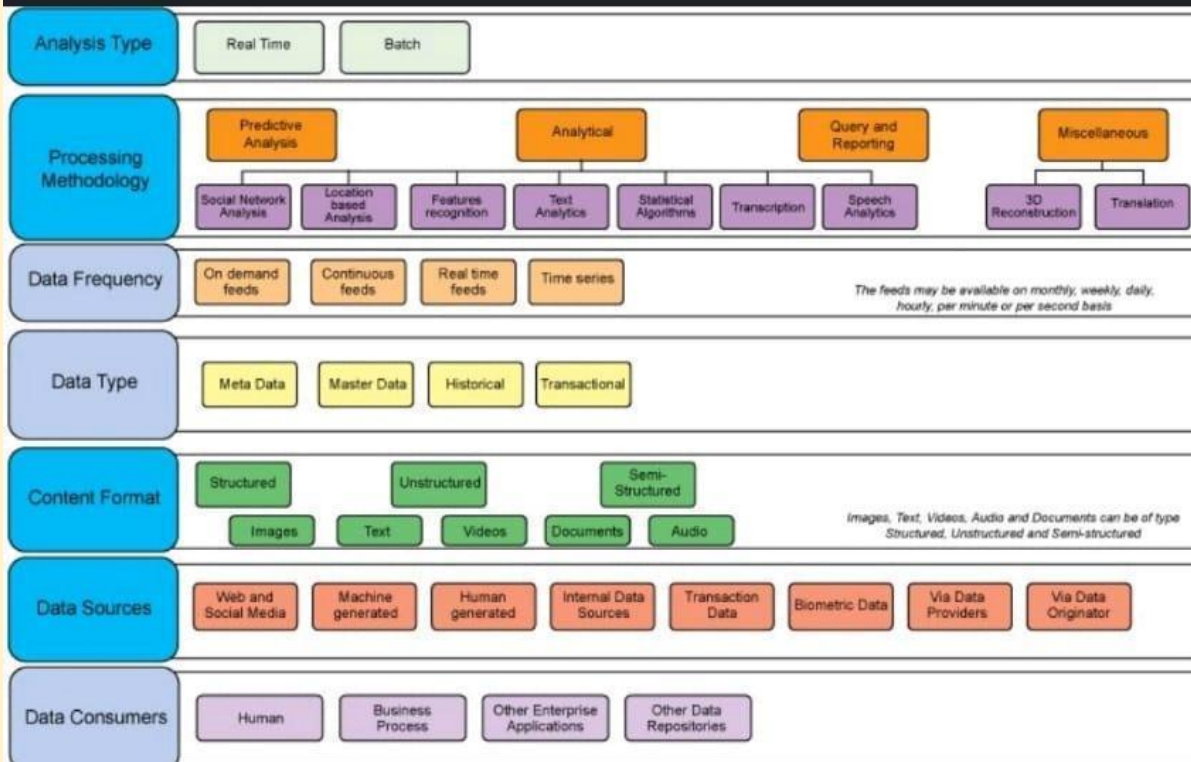
- Classifying business problems according to big data type
- Using big data type to classify big data characteristics

Business problems can be categorized into types of big data problems. Linking the business challenge to big data type is critical in defining the architecture.

It's helpful to look at the characteristics of the big data along certain lines – for example, how the data is collected, analyzed, and processed. Once the data is classified, it can be matched with the appropriate big data pattern.

[#bigdata](#) [#architecture](#)

Reference/Source: IBM



## Language-Vision Model To Leverage Text Medical Reports For Improved Segmentation

Researchers From China Propose 'LViT', A Language-Vision Model To Leverage Text Medical Reports For Improved Segmentation

- ✓ Researchers propose a new vision-language medical image segmentation model LViT (Language meets Vision Transformer).
- ✓ Medical text annotation is introduced to compensate for the quality deficiency in image data
- ✓ Experimental results show that the model has better segmentation performance in both fully and semi-supervised conditions
- ✓ Currently, the proposed model is only experimented on 2D medical data

Paper Summary:

<https://www.marktechpost.com/2022/08/02/researchers-from-china-propose-lvit-a-language-vision-model-to-leverage-text-medical-reports-for-improved-segmentation/>

Paper: <https://arxiv.org/pdf/2206.14718.pdf>

Github link: <https://github.com/HUANGLIZI/LViT>

## Information Theory & Bayesian ML

<https://sidbanerjee.github.io/courses/orie4742s21/>

## A generic, simple and fast implementation of Deepmind's AlphaZero algorithm.

<https://github.com/jonathan-laurent/AlphaZero.jl>

## Chart Suggestions Guide \ Tableau Public

<https://public.tableau.com/app/profile/adedamola8122/viz/ChartSelectionGuide2/Dashboard>

## Continuous Deployment of ML Models to the Edge

Real-life example: Worksite safety monitoring

<https://buff.ly/3Q3r2DJ>

## 100+ data structure and algorithms

1. Array - <https://t.co/wbp00Xm0QL>
2. String - <https://t.co/xm9dWnHrpM>
3. Binary tree - <https://t.co/4Cicp1VAOm>
4. Recursion - <https://t.co/hrENDp3DkB>
5. Linked List - <https://t.co/ljfQTL7p0X>
6. DP - <https://t.co/eSPf1EcJ18>

## Microsoft Research - General-Purpose Multimodal Foundation Model BEIT-3

Microsoft Research Introduces a General-Purpose Multimodal Foundation Model 'BEIT-3,' that Achieves State-of-the-Art Transfer Performance on Both Vision and Vision Language Tasks

Quick Read:

<https://www.marktechpost.com/2022/09/03/microsoft-research-introduces-a-general-purpose-multimodal-foundation-model-beit-3-that-achieves-state-of-the-art-transfer-performance-on-both-vision-and-vision-language-tasks/>

Paper: <https://arxiv.org/pdf/2208.10442.pdf>

Github: <https://github.com/microsoft/unilm/tree/master/beit>



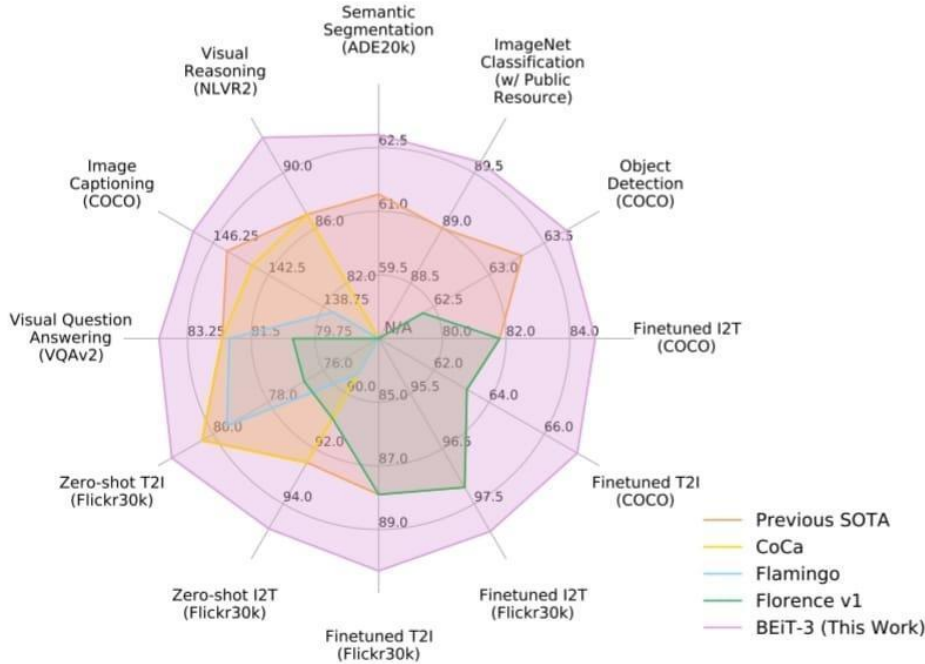
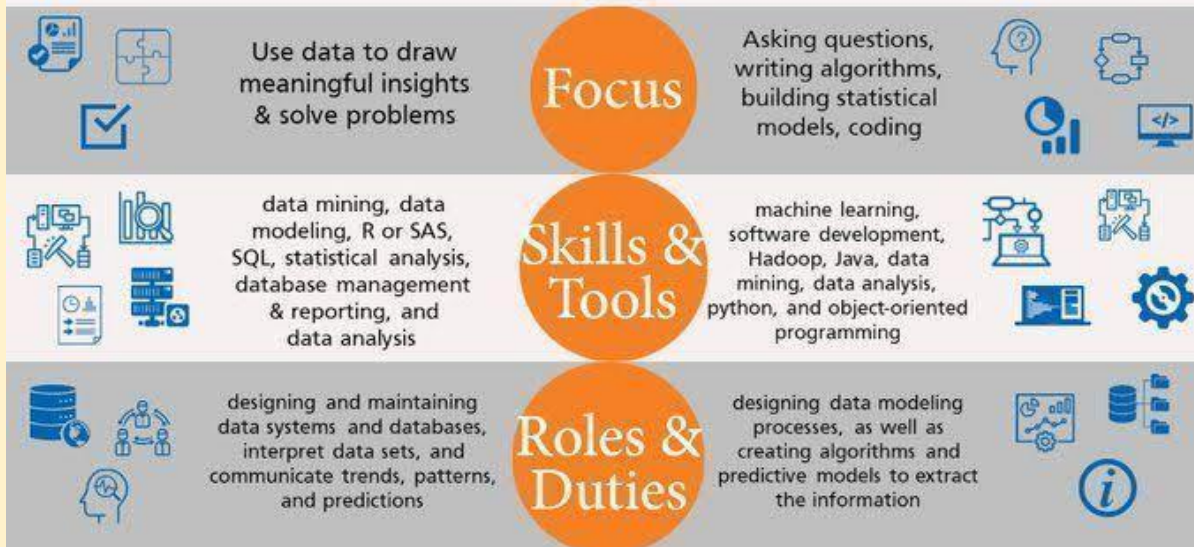


Figure 1: BEiT-3 achieves state-of-the-art performance on a broad range of tasks compared with other customized or foundation models. I2T/T2I is short for image-to-text/text-to-image retrieval.

## BEiT-3 Performance on Deep Learning Tasks

# Data Analytics VS. Data Science



## Data Analytics vs Data Science

Free Tableau Videos

1. Udemy: <https://lnkd.in/dE2GYCWj>
2. Tableau: <https://lnkd.in/dkybsvaW>
3. Skillup: <https://lnkd.in/dPr6Jk4j>
4. Analytics Vidhya: <https://lnkd.in/d4yyXQEA>
5. Learn Tableau Desktop for Accurate Business Analysis: <https://lnkd.in/dg6U57b5>



6. Data Visualization and Communication with Tableau: <https://lnkd.in/djNAe4q3>
7. Kickstarting Tableau: <https://lnkd.in/d2n-ntdz>
8. Data Visualization with Tableau Specialization: <https://lnkd.in/dgDtSwHv>
9. Tableau Fundamentals for Aspiring Data Scientists: <https://lnkd.in/d9UevMvv>
10. Noble Desktop: <https://lnkd.in/dZ7b-QES>
11. Udacity: <https://lnkd.in/ditqVfA2>

Free Tableau Cheatsheet

1. Intellipat: <https://lnkd.in/dBBRpGvz>
2. Tableau Cheat Sheet: <https://lnkd.in/dAfkudfP>
2. Datacamp: <https://lnkd.in/dVb3DKXT>
3. Tableau Desktop Keyboard Shortcuts Cheat Sheet: <https://lnkd.in/dyQsBRG7>

Free Tableau Blogs

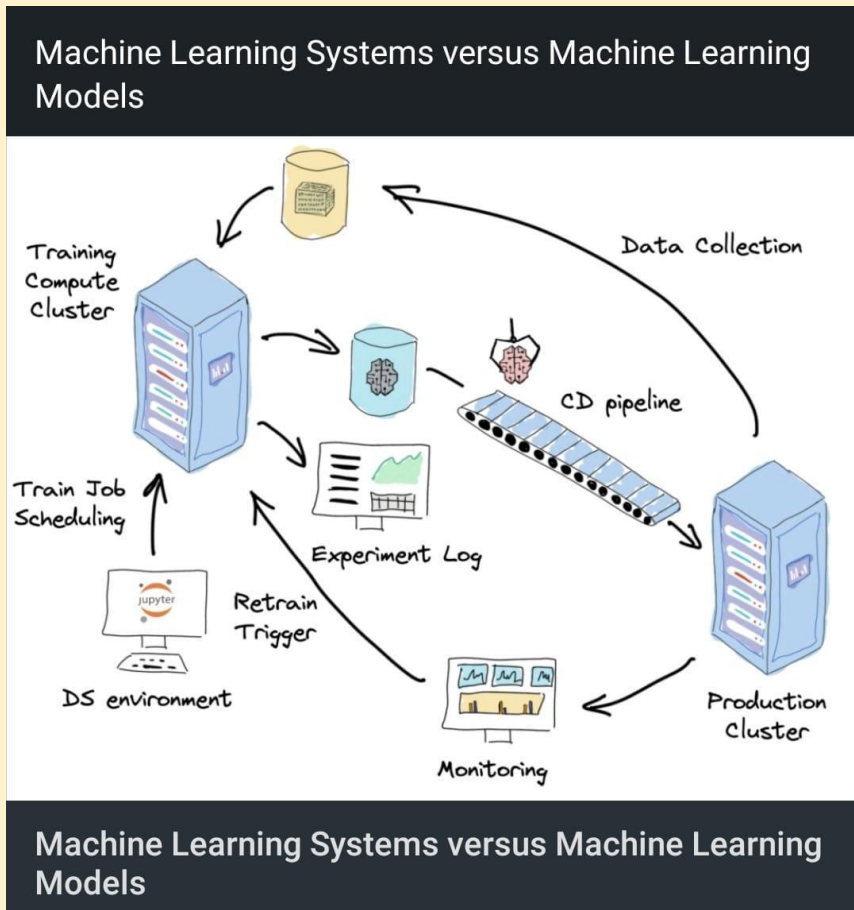
1. Tableau: <https://lnkd.in/dYS2jqqM>
2. Datavis: <https://datavis.blog/>
3. Zuar: <https://lnkd.in/duFncjQB>
4. Feedly: <https://lnkd.in/dPxUkAnH>

**Machine Learning Systems versus Machine Learning Models**

by Alexander Reshytko \ Aug, 2022 \ Towards Data Science

<https://towardsdatascience.com/machine-learning-systems-versus-machine-learning-models-3955d038ea1f>

**Machine Learning Systems vs Machine Learning Models**



## Top 8 YouTube Channels to learn Web3

Top 8 YouTube Channels to learn Web3, latest technologies development

- 1 <https://youtube.com/c/Freecodecamp>
- 2 <https://youtube.com/c/DappUniversity>
- 3 Moralis Web3: <https://t.co/E7ZNMyxzmK>
4. LearnWeb3: <https://t.co/nYLTvVpN1F>
5. BlockGeeks : <https://t.co/mzVSYXBn0W>
6. Eat The Blocks : <https://t.co/rgUgK5u3J4>
7. WhatTheFunc : <https://t.co/peexckSdgl>
8. Ethereum Foundation : <https://t.co/rgznLedrxo>

## Learning timeseries can be tough.

Learning #timeseries can be tough.

Here are the 7 #R packages that can help learn time series and improve forecasts.

#rstats

1. #timetk time series data wrangling + visualization. <https://t.co/nxbW8b9vQr>
2. #modeltime time series forecasting. <https://t.co/8eZM2PCyvm>
3. #modeltime.ensemble make ensemble forecasts <https://t.co/SfUMe3p8kh>
4. #modeltime.h2o use AutoML for forecasting <https://t.co/8bl0fsfdwB>
5. #modeltime.gluonts use deep learning for forecasting <https://t.co/hyDctyCvei>
6. #modeltime.resample backtest your forecast <https://t.co/xuOJ2AH4f4>
7. #boostime forecasting with lightgbm and catboost <https://t.co/4tRYhaxXdF>

All of these 7 R packages plus 93 more in one consolidated #cheatsheet, download ultimate #R cheat sheet.

<https://t.co/tThSsujzvl>

## Pen and Paper Exercises in Machine Learning

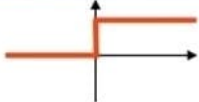

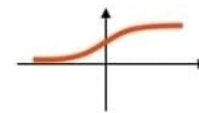

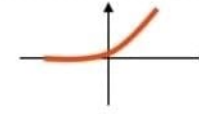
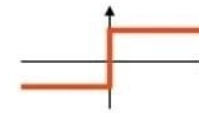
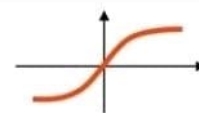
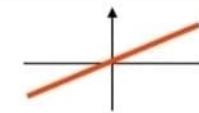
<https://arxiv.org/abs/2206.13446>

## 12 Best Online Courses for Machine Learning with Python-Bestseller 2022

<https://www.mltut.com/best-online-courses-for-machine-learning-with-python/>

| Algorithm               | Training (time complexity)         | Prediction (time complexity) | Auxiliary space complexity | Comment                                                                                                                                                                                                                                                         |
|-------------------------|------------------------------------|------------------------------|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Linear regression       | $O(nm^2+m^3)$ [1]                  | $O(m)$                       | $O(m)$                     | Low run time and space complexity!<br>Suitable for low latency problems                                                                                                                                                                                         |
| Logistic regression     | $O(nm)$ [2]                        | $O(m)$                       | $O(m)$                     | Low run time and space complexity!<br>Suitable for low latency problems                                                                                                                                                                                         |
| Naive Bayesian          | $O(n*m)$                           | $O(c*m)$                     | $O(c*m)$                   | Works well with high dimensional data. Mainly used in NLP problems with large dimensionality. Also used to compare with more complex algorithms used for NLP.                                                                                                   |
| Decision Trees          | $O(n\log(n)d)$                     | $O(d)$                       | $O(p)$                     |                                                                                                                                                                                                                                                                 |
| Gradient Boosting       | $O(n\log(n)dr)$                    | $O(dr)$                      | $O(pr+br)$                 | br could be dropped as $br \ll pr$                                                                                                                                                                                                                              |
| Random Forest           | $O(n\log(n)dn_{tree})$             | $O(dn_{tree})$               | $O(pn_{tree})$             |                                                                                                                                                                                                                                                                 |
| Support Vector Machines | $O(n^2m+n^3)$                      | $O(mn_{sv})$                 | $O(n_{sv})$                | High training time but low space complexity. Suitable for low latency problems                                                                                                                                                                                  |
| K Means                 | $O(nmki)$                          | $O(mk)$                      | $O(nm+km)$                 |                                                                                                                                                                                                                                                                 |
| KNN                     | $O(1)$                             | $O(nm)$                      | $O(nm)$                    |                                                                                                                                                                                                                                                                 |
| Hierarchical clustering | $O(n^3) \Rightarrow O(n^2\log(n))$ | $O(mn)$                      | $O(n^2)$                   | High training time, high runtime, and high space complexities. Not suitable for low latency.                                                                                                                                                                    |
| DBSCAN                  | $O(n^2) \Rightarrow O(n\log(n))$   | $O(mn)$                      | $O(n)$                     | Better than hierarchical clustering in terms of complexities                                                                                                                                                                                                    |
| PCA                     | $O(nm \times \min(n, m) + m^3)$    | $O(lm)$                      | $O(lm)$                    | The complexity of covariance matrix computation is $O(nm \times \min(n, m))$ . Its eigenvalue decomposition is $O(m^3)$ . $O(lm)$ is for the transformation matrix                                                                                              |
| t-SNE                   | $O(n^2)$                           | $O(nq)$                      | $O(n^2)$                   | t-SNE involves significant amount of computations and hence the algorithm takes a lot of time and space for the computation. t-SNE has a quadratic time and space complexity in the number of data points. <b>t-SNE requires <math>O(3n^2)</math> of memory</b> |

## Time Space Complexity in Machine Learning Algo

| Activation Function                           | Equation                                                                                                                                          | Example                                   | 1D Graph                                                                              |
|-----------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------|---------------------------------------------------------------------------------------|
| <b>Value between 0 to 1</b>                   |                                                                                                                                                   |                                           |                                                                                       |
| Unit step<br>(Heaviside)                      | $\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$                                                                 | Perceptron<br>variant                     |    |
| Piece-wise linear                             | $\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$ | Support vector<br>machine                 |    |
| Logistic (sigmoid)                            | $\phi(z) = \frac{1}{1 + e^{-z}}$                                                                                                                  | Logistic<br>regression,<br>Multi-layer NN |    |
| Rectifier, ReLU<br>(Rectified Linear<br>Unit) | $\phi(z) = \max(0, z)$                                                                                                                            | Multi-layer<br>Neural<br>Networks         |    |
| Rectifier, softplus                           | $\phi(z) = \ln(1 + e^z)$                                                                                                                          | Multi-layer<br>Neural<br>Networks         |    |
| <b>Value between -1 to 1/ -inf to +inf</b>    |                                                                                                                                                   |                                           |                                                                                       |
| Sign (Signum)                                 | $\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$                                                                  | Perceptron<br>variant                     |  |
| Hyperbolic tangent                            | $\phi(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$                                                                                                     | Multi-layer<br>Neural<br>Networks         |  |
| Linear                                        | $\phi(z) = z$                                                                                                                                     | Adaline, linear<br>regression             |  |

## Activation Functions

### How To Patent Artificial Intelligence And Machine Learning Models

<https://www.forbes.com/sites/forbesbusinesscouncil/2022/08/29/how-to-patent-artificial-intelligence-and-machine-learning-models/>

## Ace Your Data Science Interview in 6 Steps

By Catherine Adenie

[www.catherinescareerconet.com](http://www.catherinescareerconet.com)

### 1. Research the role and the business you are

Data Scientist is a blanket term, and there are many roles in the data science ecosystem. Such as Data Engineer, Data Analyst, Data Tester, Statistician, Data Architect, and many more.

When you are applying for a data science job, research the requirement of the specific role and the organization.

### 2. Be able to back your Resume

Have your resume several times a day before going for the interviews. Filter out the things you can do without mention the most applicable projects and s for the role

3. Make your digital presence work for you is a fact that more than 80% of recruiters will test check a candidate's LinkedIn profile before scheduling an interview Shows the value of your digital presence. We are in the digital age, which means a piece of paper with your resume on it will only take you applying to

4. Take a crack at puzzles and riddles

Data science recruiters love put highly when is the interview process, you'd be asked to solve see, patterns, number sequences, and alignments

5. Go through Data Science Glossary & Terms

Recruiters point-blank kordions and what you understand by specific data science terms and concepts, if such questions have previously cond you should prepare before the interview Freshen up on the data science-related terminology by quickly going through the relevant glossaries and adding up on the concepts.

6. Assignments & Case Studies

At times the interviewer all ask you to solve a case study or a particular assignment. These can be of two types, and they give you a take-home assignment which as the name suggests something you can do at your discretion at home and take your time. Another and most common is an on-site case study problem and can take 3-6 hours depending on the type of problem

### **Stanford University Free Online Courses 2022**

<https://opportunitiespedia.com/stanford-university-free-online-courses-2022/>



## Timeseries in Pycaret

```
#### Import libraries ----
from pycaret.datasets import get_data
from pycaret.time_series import TSForecastingExperiment

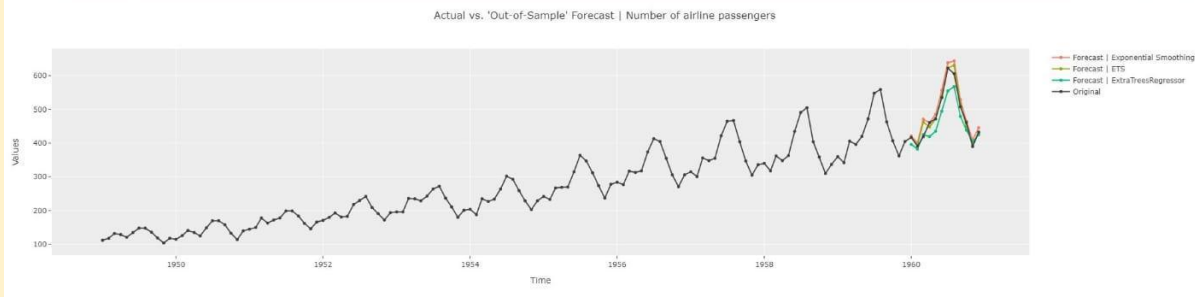
#### Get data ----
data = get_data("airline")

#### Create an forecasting experiment ----
exp = TSForecastingExperiment()
exp.setup(data=data, fh=12, session_id=42)

#### Select 3 best candidate models ----
best3 = exp.compare_models(n_select=3)

#### Tune the candidate models ----
tuned3 = [exp.tune_model(model) for model in best3]

#### Plot and compare multiple models ----
exp.plot_model(tuned3)
```



## Checklist - training deep neural networks

What are some of the basic things you should watch out for when training deep neural networks? Here are my top 5:

- 1) Make sure training loss converged
- 2) Check for overfitting
- 3) Compare accuracy to a zero-rule baseline
- 4) Look at failure cases
- 5) Plot at a confusion matrix

Bonus! some additional suggestions from the community:

6) Make sure your model is able to overfit to a small dataset (like 1000 examples) or a single minibatch. (Tip 3 from Andrej Karpathy's "A Recipe for Training Neural Networks" <https://lnkd.in/gvfgqxTQ>)

7) Check whether layers have converged to a good alpha (e.g., using weightwatcher). See <https://lnkd.in/gHDHDXAu>

8) Check how confident your model is on out-of-distribution data (a common problem for neural nets); one out of many papers on this topic:

<https://lnkd.in/gsjSVYcd>

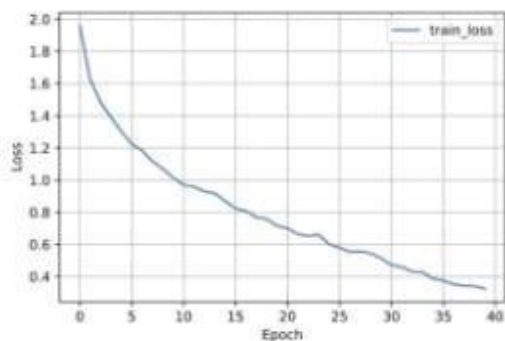
9) Apply your model to new data from the application domain (e.g., if you train a handwritten digit classifier, try your own handwritten digits)

## 5 Checks during Training of ML Models

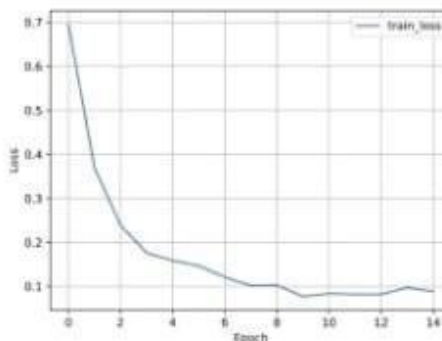
# My top 5 basic checks when training deep learning models

## 1) Make sure the training loss converged

This is a classic. We typically want to see that the loss plateaus



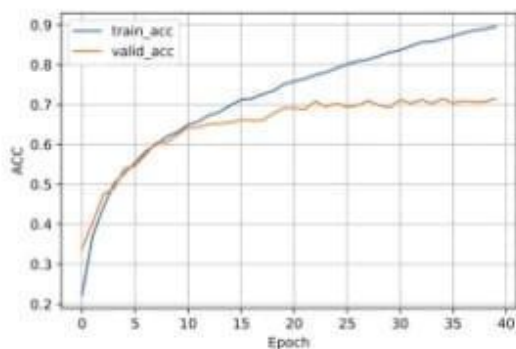
(Bad)



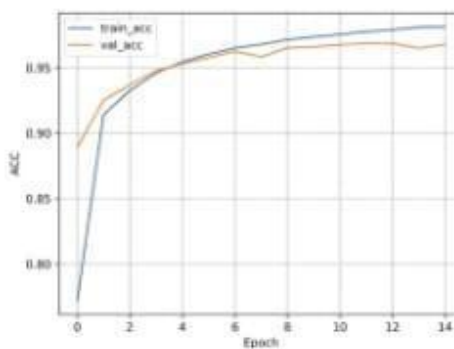
(Better)

## 2) Check for overfitting

We typically don't want the gap between training and validation accuracy to be too large



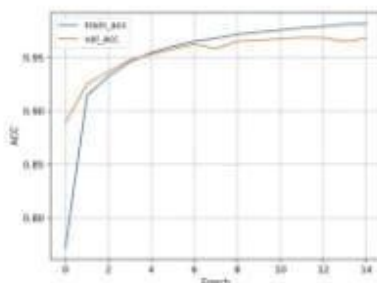
(Bad)



(Better)

## 3) Compare accuracy to a zero-rule baseline

Here, I check that the validation accuracy is substantially better than a baseline based on always predicting the majority class (aka zero-rule classifier)



### Check label distribution

From collections import Counter

```
test_counter = Counter()
for images, labels in test_loader:
    test_counter.update(labels.view(-1))
print('test label distribution')
print(test_counter)
```

test label distribution:  
(18, 980), (1, 120), (2, 980), (3, 980), (4, 980), (5, 980), (6, 980), (7, 980), (8, 980), (9, 980)

### Zero-rule baseline (majority class classifier)

```
majority_class = test_counter.most_common(1)[0]
```

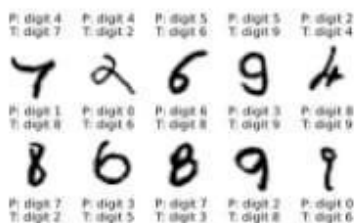
```
print('majority class:', majority_class[0])
```

```
baseline_acc = majority_class[1] / len(test_loader.dataset)
print('accuracy when always predicting the majority class:')
print(baseline_acc) # 0.9800000000000001
```

```
majority_class: 0
accuracy when always predicting the majority class:
0.12 (0.98)
```

## 4) Look at failure cases

It's always useful to check what cases the model gets wrong. Sometimes, there is something useful that can be learned from this analysis (incl. detecting mislabeled data)




## CS 229 - Supervised Learning Cheatsheet

<https://stanford.edu/~shervine/teaching/cs-229/cheatsheet-supervised-learning>

<https://learn.jatanshah.in/dashboard/>

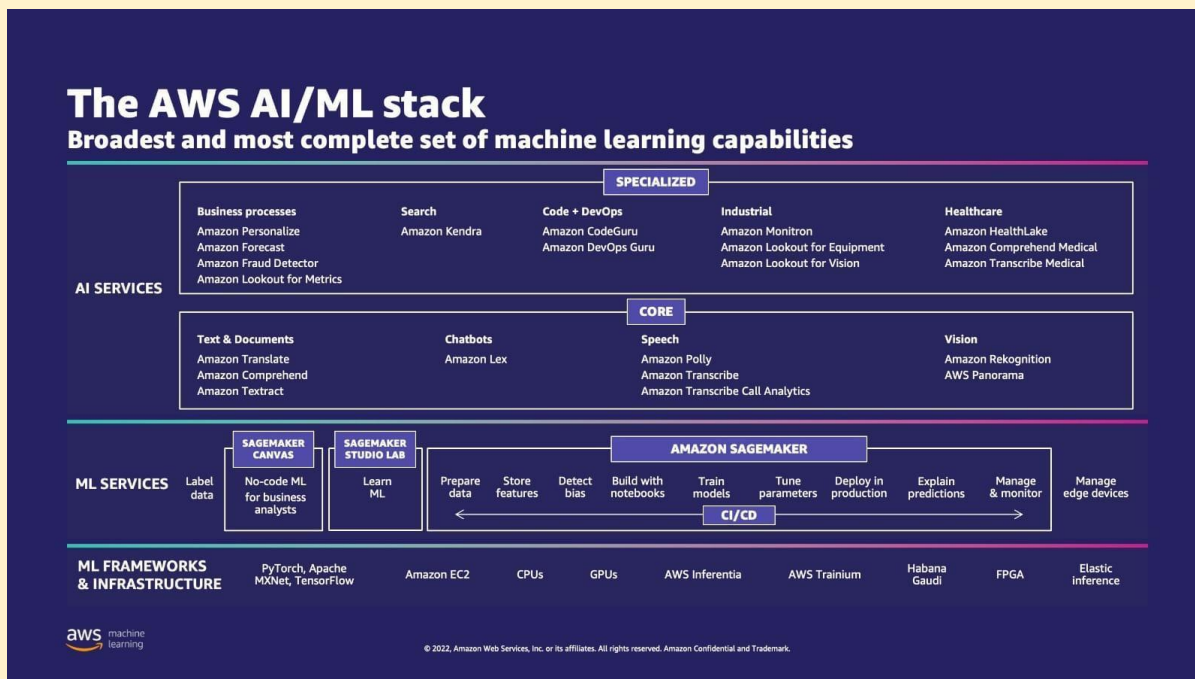
## Expert Data Analyst links

Master Data Visualisation 

1. Excel : <https://t.co/j9vCt9GRRo>
2. Tableau: <https://t.co/4ceaHYRqOr>
3. Power BI: <https://t.co/jjrSIGK6g0>
4. QlikView: <https://t.co/FQGtFmfrJQ>
5. Looker: <https://t.co/lWPztYNS1u>

## Learn Go

<https://karanpratapsingh.com/courses/go>



## The AWS AI/ML Stack

Tutorial: Statistical Tests of Hypothesis - [DataScienceCentral.com](https://www.datasciencecentral.com)

<https://www.datasciencecentral.com/tutorial-statistical-tests-of-hypothesis/>

DP-100 Exam Prep - How to become Azure Data Scientist Associate?

<https://intellipaat.com/blog/how-to-become-microsoft-certified-azure-data-scientist-associate-dp-100/>

DocQuery, a new #opensource query engine for analyzing documents using large language models (LLMs).



<https://www.impira.com/blog/hey-machine-whats-my-invoice-total>

The demo was hosted at <https://t.co/UKPxTnQQCR> and allows you to ask any question you'd like from a PDF document or image.

LLMs applied to #intelligentdocumentprocessing is a huge unlock.

An easy way to extract information from documents

<https://github.com/impira/docquery>

## **Challau Metaverse**

<https://www.challau.com/create-your-free-world?twclid=2-5wzwcwecye46zt1lio7ltgo80x>

## Data Quality Attributes

### Data quality attributes

| Attribute    | What it means                                                                                         | Example of good practice                                                                                                                         | Example of bad practice                                                                                                                  | Metrics                                                                   |
|--------------|-------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------|
| Consistency  | No matter where you look in the database, you won't find any contradictions in your data.             | Your payment system shows that Jane Brown has made 5 purchases this month, and CRM system contains the same information.                         | Your payment system shows that Jane Brown has made 5 purchases this month, while CRM system shows she has made only 4.                   | The number of inconsistencies.                                            |
| Accuracy     | The information your data contains corresponds to reality.                                            | Your customer's name is Jane Brown. And this is exactly how it's reflected in your CRM.                                                          | In your CRM, the customer's name is spelled Jane Brawn, though her actual name is Jane Brown.                                            | The ratio of data to errors.                                              |
| Completeness | All available elements of the data have found their way to the database.                              | You know that Jane Brown is born on 11/04/1975.                                                                                                  | You have no idea how old Jane Brown is, as the date of birth cell is empty.                                                              | The number of missing values.                                             |
| Auditability | Data is accessible and it's possible to trace introduced changes.                                     | You can track down the changes made in Jane's data record. For example, on 12/5/2018, her phone number was changed.                              | It's impossible to trace down the changes in Jane's record.                                                                              | % of cells where the metadata about introduced changes is not accessible. |
| Orderliness  | The data entered has the required format and structure.                                               | The entry for December 11, 2018 is in the format 12/11/2018.                                                                                     | The entry for December 11, 2018 is in the format 12/11/18, 12/11/2018 and even 11/12/18 (in your European stores).                       | The ratio of data of inappropriate format.                                |
| Uniqueness   | A data record with specific details appears only once in the database.                                | You have only one record for Jane Brown, born on 11/04/1975, who lives in Seattle.                                                               | You have multiple duplicate records for Jane Brown.                                                                                      | The number of duplicates revealed.                                        |
| Timeliness   | Data represents reality within a reasonable period of time or in accordance with corporate standards. | On 02/15/2018, the customer informed you that her name is misspelled in the emails you send her. The customer's name was corrected the next day. | On 02/15/2018, the customer informed you that her name is misspelled in the emails you send her. Her name was corrected only in a month. | Number of records with delayed changes.                                   |

# Which Visualization? A Quick Reference

You have the following data (sample):  
**Discrete Categories,**  
**Ordered categories,**  
**and Continuous Metrics**

| Categories | Ordered Cats | Continuous Metrics |
|------------|--------------|--------------------|
| City       | Class        | Price              |
| Alphaville | XeroTrip     | Coach \$           |
| Betastan   | YoloFly      | Business \$\$      |
| Chicago    | ZeusAir      | First \$\$\$       |
| ...        | ...          | ...                |

Here's how to plot them

**Discrete Categories**

**Ordered Categories**

**Continuous Metrics**

Metric, binned by 1 category

... by 2 categories

Part-to-Whole, binned by 1 category

... by 2 categories

by hierarchies

**Bar (Row)**

Rows allow readable labels, while columns awkwardly turn text

**Lollipop**

More focus on the positions of tops. Fun factor +1

**Dot Plot**

A non-zero y-axis base may be less misleading here

**Bar (Column)**

Histogram. Boxes help convey the underlying bins

**Bar (Column)**

Time moves horizontally. So use Column, not Row

**Area**

Adds continuity to x-axis.

**Line**

A non-zero y-axis base may be less misleading here

**Bar Table X,Y,Z,...**

Compare X to Y to Z. "Small multiples". Please use this more.

**Bar Table X,Y, Delta**

Comparisons are slow. Plot critical Deltas explicitly

**Mirror Bar**

Compare X to Y, leverages human symmetry perception

**Benchmark Bar**

Compare X to a benchmark

**Benchmarks Bar**

Compare X to Y. Fancyer version called a "Bullet graph"

**Interleaved Bar**

Interleaves two categories into one spatial dimension. Typically better to use Bar Table (above) instead

**2D Heat**

2D Histogram. Similar in spirit to a bar table, but ordered cats + color encoding

**Bar Table**

Compare a metric across an ordered category

**Bar Line Table**

Now one is continuous-er

**Line Table**

Trends visible, but use Lines (below) to compare heights

**Slopegraph**

With two values, slope encodes delta

**Dual Axis**

Use (above) instead. Crossings here are salient, but meaningless

**Lines**

Compare many. Getting spaghetti? Split into subset or Line-Table (above)

**Pie**

Screams "Percentages!"

**Stacked Bar (Row)**

More precise and flex, but less screaming

**Stacked Bar (Col)**

Now I'm standing

**Waterfall**

Waterfalls are vertical stacked bars that narrate financial values in a (typically) artificially imposed ordering across fantasy-time

**MultiPie**

Please don't... (not recommended)

**Stacked Bars (R)**

Stacked bars but now with X, Y, Z info. Here XYZ might be absolute values of a market, ABC are company % market shares

**[Mar]Mekko**

Stacked bars but now with X, Y, Z info. Here XYZ might be absolute values of a market, ABC are company % market shares

**Stacked Bars (C)**

Horizontal flow implies an ordering

**... with lines**

Added lines suggest continuity, help depict changes

**Stacked area**

Now it's definitely continuous

**Breakout Bar**

Let's zoom in here. Use different colors. Global at top or left.

**Treemap**

Hierarchy, ~3 levels max of bento boxes going all (ception) within other bentos. Size+Color better code different metrics. Typically misused. 95% sure you actually wanted a Bar Table (above)

**Scatter**

An elegant graph, from a civilized age

**Connected Scatter**

A scatterplot, connected into a journey over time

**Parallel Coordinates**

Beyond 2 perpendicular Cartesian axes, Parallel format allows more axes

**Map**

Maps and Rosings share the same DNA. Color = Flight Time. Size = Price

**Hans Rosling Scatter**

Watch Rosling's TED talk. Take XY scatter and adds two more metrics (color and size), and then moves in time

**Rosling Comet**

Show two or more X+Y history values for comparison over time

**Look at this number. Just look at it.**

**43%**

Huge Number

Connecting Lines imply continuous data

Same data, ask people: "What do you see?"

USA Dutch

USA Dutch

Zacks & Tversky, 1997

"Dutch people are taller than Americans"

"People get taller as they get more Dutch"

**Metrics: relationships to other metrics**

**Map**

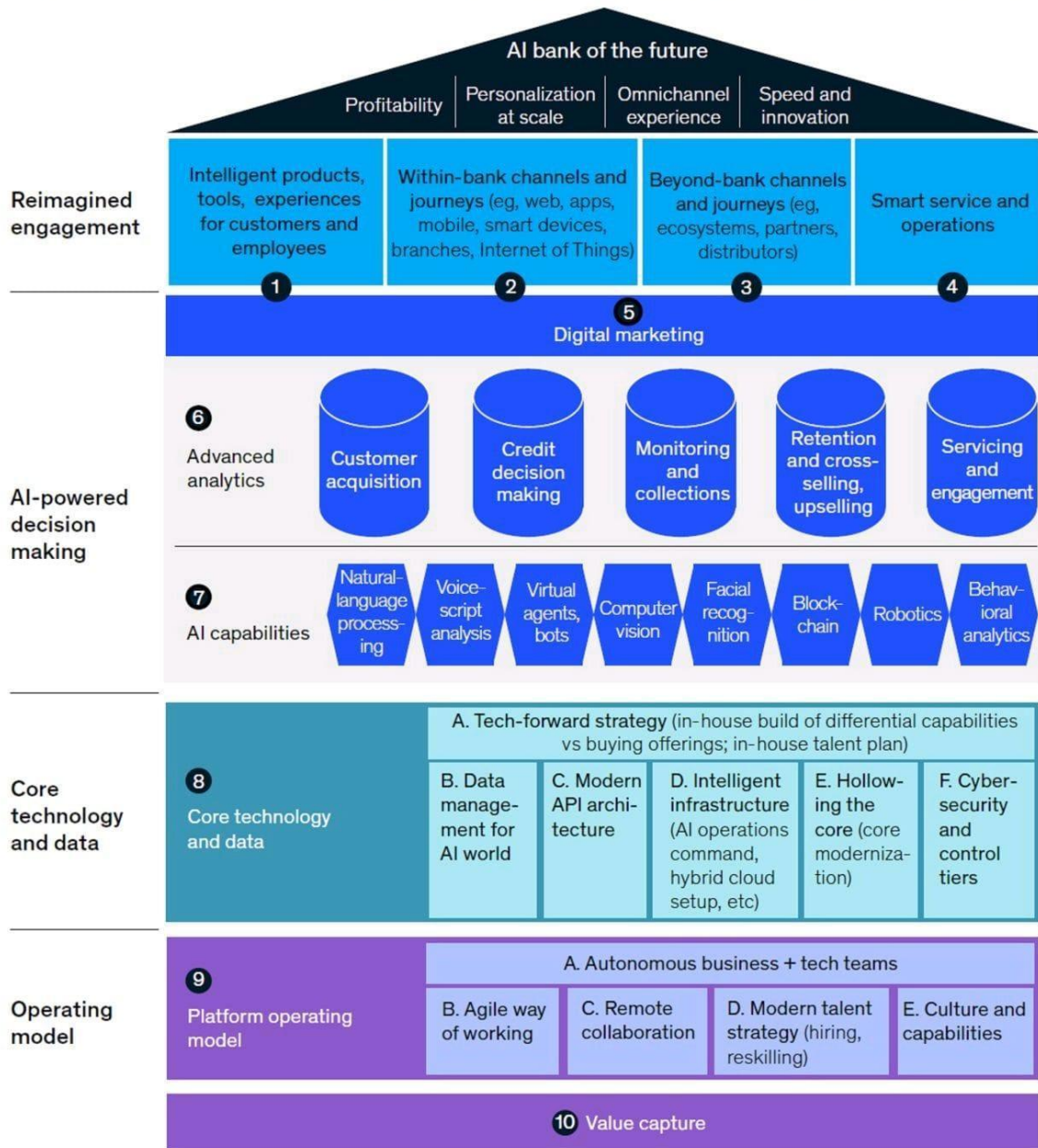
Maps and Rosings share the same DNA. Color = Flight Time. Size = Price



## Introduction to Causal Inference

<https://www.bradyneal.com/causal-inference-course>

# The AI Bank of the future



Source: McKinsey & Company

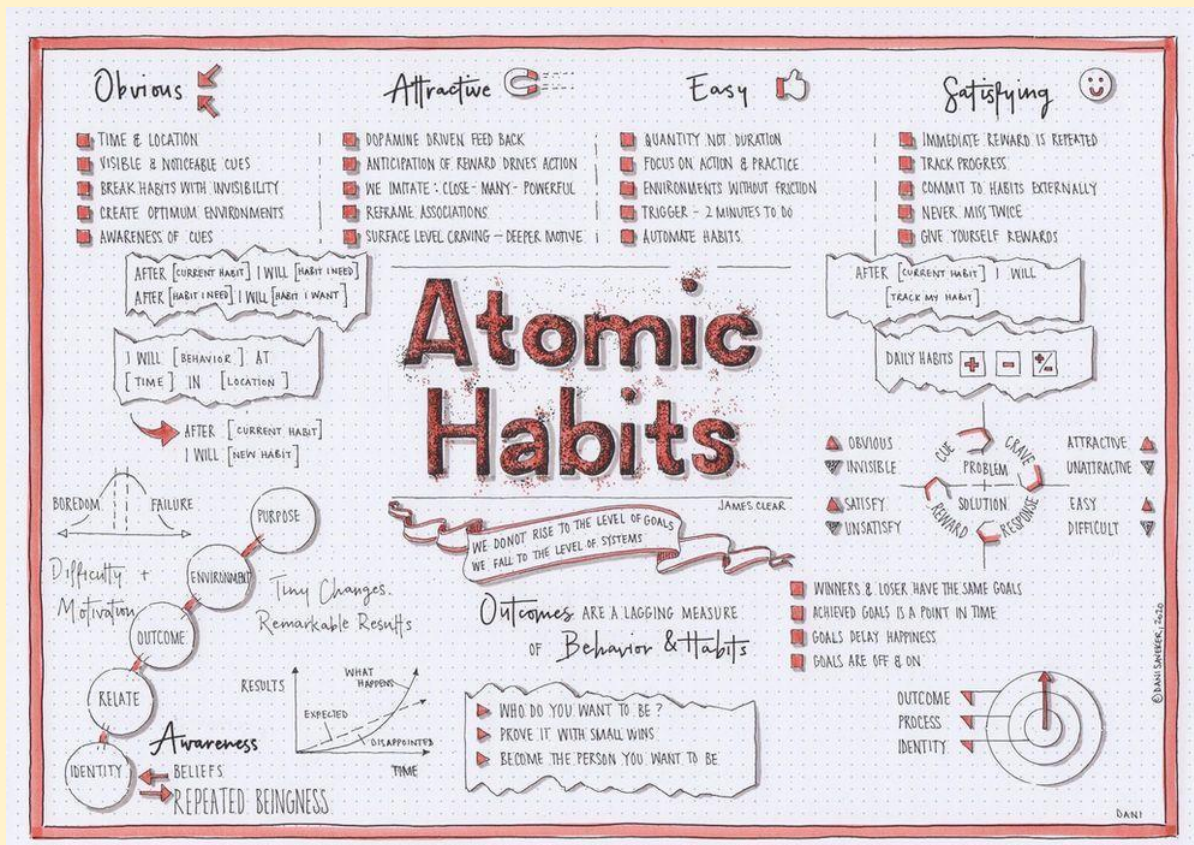
Edited by: Panagiotis Kriaris

## AI Bank of Future

Automate with Python – Full Course for Beginners

<https://youtu.be/PXMJ6FS7llk>

## Atomic Habits



## The Role of an AI Architect

<https://www.gartner.com/en/articles/what-are-ai-architects-and-what-do-they-do>

## "An Analysis of Deep Learning Neural Networks"

\ by Jitendra R Rathod \ Medium

<https://medium.com/@jiterockzz/an-analysis-of-deep-learning-neural-networks-e9e3e038426a>

## Comparing Model Evaluation Techniques

Comparing Model Evaluation Techniques.

Part 1: <https://t.co/JVY7fEkaIZ>

Part 2: <https://t.co/x9tOYoKiJS>

Part 3: <https://t.co/Pue3biAw6r>

#BigData #DataScience #Statistics #AI #MachineLearning #Algorithms #DataLiteracy #StatisticalLiteracy #Mathematics

## 5 open-source frameworks for implementing GANs

1. Deep Graph Library
2. Spektral
3. PyTorch Geometric
4. DeepMind's Jraph
5. TensorFlow TF-GNN



## **PregEx : Write Human-Readable Regular Expressions in Python**





Full article about PregEx: <https://t.co/v3kBjMm2Wb>

Link to PregEx: <https://t.co/l6cgWxxLIV>

## **FREE Node.js course**

Launched FREE Node.js course today

<https://nodejsbeginner.com/>

-  Learn best way to install Node.js
-  Learn to manage Node.js versions
-  Build & launch your first CLI
-  13+ vids 1hr of content

## **Seeing Theory : Statistics more accessible through interactive visualizations**

<https://seeing-theory.brown.edu/>

## **Use MLNET to build custom machine learning solutions in dotnet**

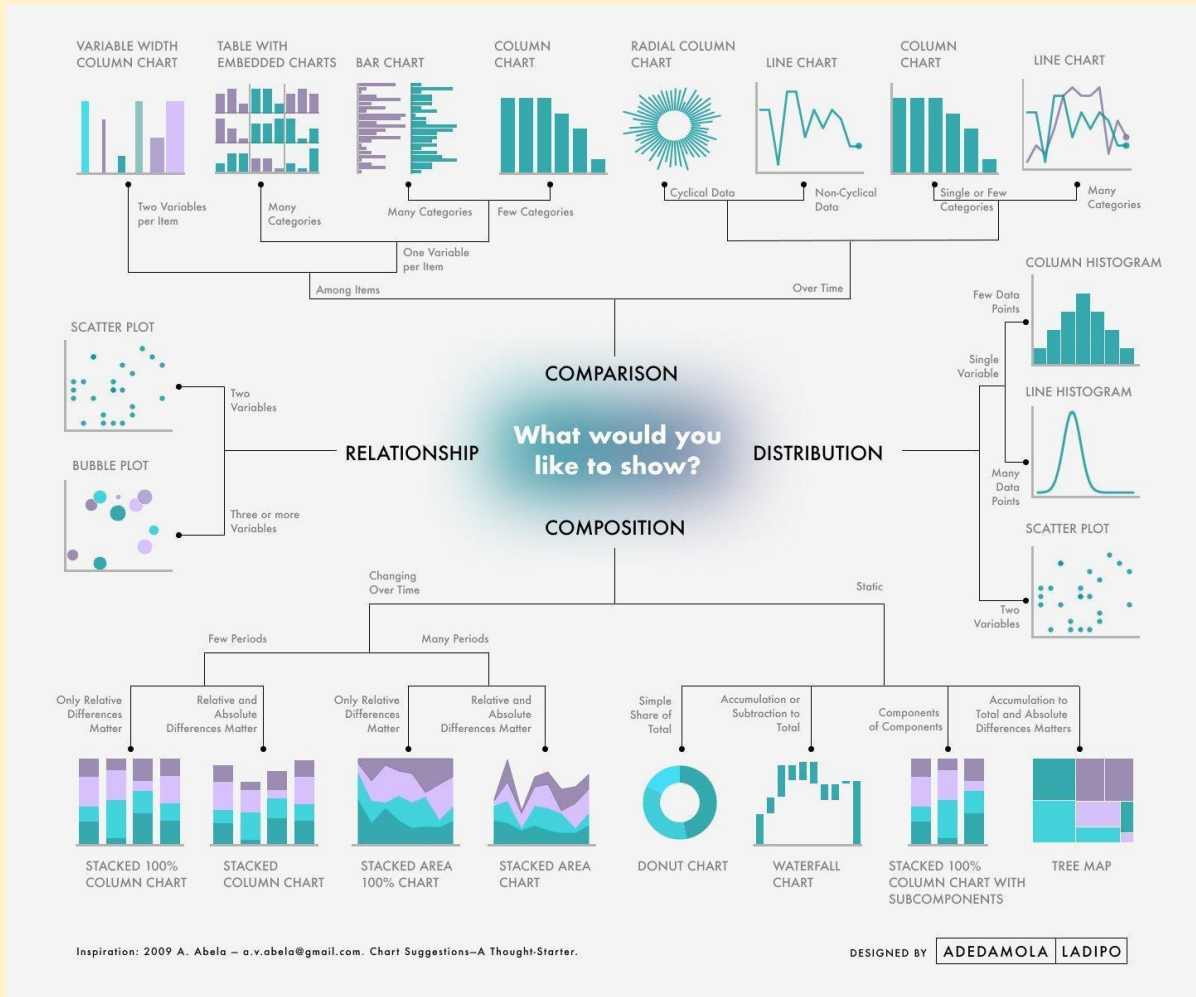
Use #MLNET to build custom machine learning solutions and integrate them into your #dotNET apps.

Apply sentiment analysis, GitHub issue classification, price predictor, image classification, and more. Follow the tutorials: <https://docs.microsoft.com/en-us/dotnet/machine-learning/tutorials/>

## **Conversational AI and Chatbot**

[https://youtube.com/playlist?list=PLAMHV77MSKJ7s4jE7F\\_k\\_Od8qZIFGf1BY](https://youtube.com/playlist?list=PLAMHV77MSKJ7s4jE7F_k_Od8qZIFGf1BY)

# Different Charts



# 10 Machine Learning Algorithms In Python [A Beginners Guide]

<https://www.theinsaneapp.com/2021/11/machine-learning-algorithms-for-beginners.html>

# GEPHI

“ Gephi is the leading visualization and exploration software for all kinds of graphs and networks. Gephi is open-source. ”



## APPLICATIONS

**Exploratory Data Analysis:** intuition-oriented analysis by networks manipulations in real time.

**Link Analysis:** revealing the underlying structures of associations between objects.

**Social Network Analysis:** easy creation of social data connectors to map community organizations and small-world networks.

**Biological Network analysis:** representing patterns of biological data.

**Poster creation:** scientific work promotion with hi-quality printable maps.



## PLUGINS

Customizable by plugins: layouts, metrics, data sources, manipulation tools, rendering presets and more.

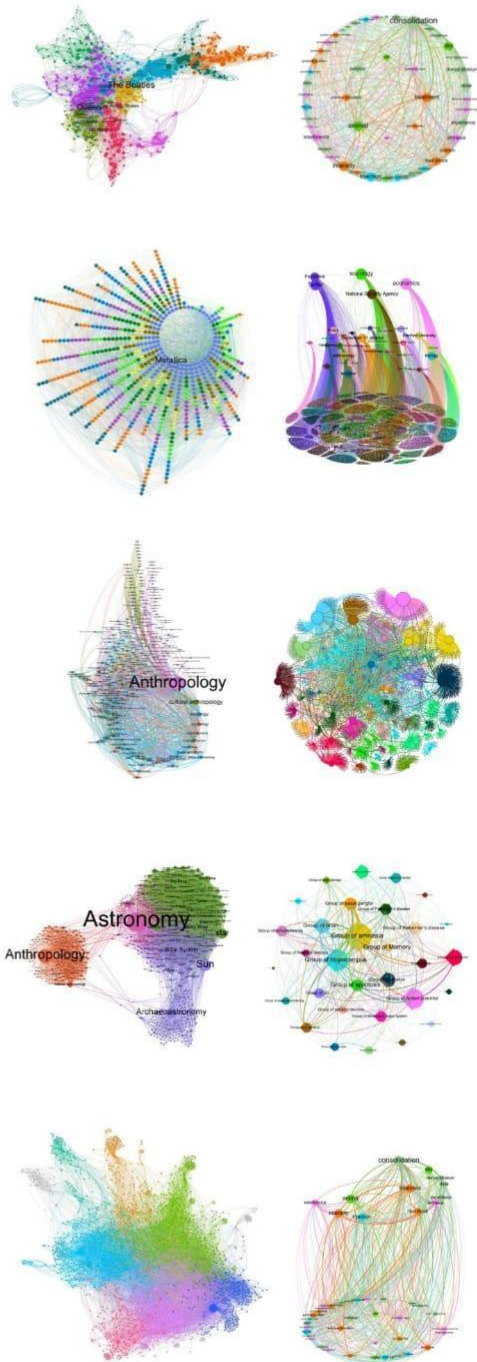


## GEPHI TEAM

Young engineers and researchers in computer sciences, particularly interested in web mining, network sciences / complex networks and information visualization.

 [gephi.org](http://gephi.org)

 Gephi



Poster and graphs made by  
PHD Verónica  
[@Verukita1](https://twitter.com/Verukita1)

## Gephi : Visualization and Exploration

[gephi.org](http://gephi.org)

Gephi - The Open Graph Viz Platform

<https://gephi.github.io/>

## **Transformers are Graph Neural Networks**

<https://graphdeeplearning.github.io/post/transformers-are-gnns/>

Read

Graph Transformer: A Generalization of Transformers to Graphs

<https://link.medium.com/moTRTjUkSsb>

GNNs and Transformers Workshop

<https://youtu.be/ZS00tnHVRbU>

## **goodrahstar/draw-neural-network:**

Quick tool to draw fully connected neural network architectures

<https://github.com/goodrahstar/draw-neural-network>

## **Quick, Draw!**

<https://quickdraw.withgoogle.com/>

## **Draw Neural Networks in Latex**

Super useful tool to plot/draw Neural Networks in Latex, based on TikZ package.  
PlotNeuralNet

<https://t.co/Qdp82eTQ5V>

#MachineLearning #DeepLearning #NeuralNetworks #plot

## Statistical Tests

| Hypothesis Testing                      |                                                                                                        |                                                                                                                                             |                                                                                                                                                                                                                                                               |                                                                                                                                                |                                                                                                                                                                               |
|-----------------------------------------|--------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Type Of Test                            | Purpose                                                                                                | Example                                                                                                                                     | Equation                                                                                                                                                                                                                                                      | Comment                                                                                                                                        | Excel Function                                                                                                                                                                |
| <b>Z Test</b>                           | Test if the average of a single population is equal to a target value                                  | Do babies born at this hospital weigh more than the city average                                                                            | $Z = \frac{\bar{x} - u_0}{\frac{\sigma}{\sqrt{n}}}$                                                                                                                                                                                                           | Z test does not need df<br>$\sigma$ = population standard deviation                                                                            | =Ztest(array,x,sigma)                                                                                                                                                         |
| <b>1 Sample T-Test</b>                  | Test if the average of a single population is equal to a target value                                  | Is the average height of male college students greater than 6.0 feet?                                                                       | $t = \frac{\bar{x} - u_0}{\frac{s}{\sqrt{n}}}$<br>$df = n - 1$                                                                                                                                                                                                | s = sample standard deviation                                                                                                                  | no built in equation<br>use =STDEVA for standard deviation<br>use =AVERAGE for mean<br>use =T.DIST.RT to get 1 tailed confidence<br>use =T.DIST.2T to get 2 tailed confidence |
| <b>Paired T-Test</b>                    | Test if the average of the differences between paired or dependent samples is equal to a target value. | Weigh a set of people. Put them on a diet plan. Weigh them after. Is the average weight loss significant enough to conclude the diet works? | $t = \frac{\bar{d}}{\frac{s}{\sqrt{n}}}$<br>$df = n - 1$                                                                                                                                                                                                      | d bar = average difference between samples<br>s = sample deviation of the difference<br>n = count of one set of the pairs (don't double count) | =TTEST(Array1,Array2,*,1)<br>* -> 1 for 1 tailed, 2 for 2 tailed                                                                                                              |
| <b>2 Sample T-Test Equal Variance</b>   | Test if the difference between the averages of two independent populations is equal to a target value  | Do cats eat more of type A food than type B food                                                                                            | $df = n_1 + n_2 - 2$<br>$t = \frac{(\bar{x}_1 - \bar{x}_2)}{\sqrt{\frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2} * \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}$                                                                                       | n1, n2 = count of sample 1, 2                                                                                                                  | =TTEST(Array1,Array2,*,2)                                                                                                                                                     |
| <b>2 Sample T-Test Unequal Variance</b> | Test if the difference between the averages of two independent populations is equal to a target value  | Is the average speed of cyclists during rush hour greater than the average speed of drivers                                                 | $t = \frac{(\bar{x}_1 - \bar{x}_2)}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$<br>$df = \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\frac{\left(\frac{s_1^2}{n_1}\right)^2}{n_1 - 1} + \frac{\left(\frac{s_2^2}{n_2}\right)^2}{n_2 - 1}}$ |                                                                                                                                                | =TTEST(Array1,Array2,*,3)                                                                                                                                                     |

## Graph ordering attention networks

The GOAT is a new graph layer where an LSTM is applied to a learnt node ordering (obtained by a simple attention layer + sorting).

Cool results!

Read paper: [https://www.researchgate.net/publication/359918143\\_Graph\\_Ordering\\_Attention\\_Networks](https://www.researchgate.net/publication/359918143_Graph_Ordering_Attention_Networks)

## Bitcoin Prediction - FB-Prophet outperforms ARIMAX, XGBOOST & LSTM

[https://m.youtube.com/watch?v=0gXeXtL\\_KjY&list=PLxqBkZuBynVS8mDTc8ZGermXiS-32pR2y&index=16](https://m.youtube.com/watch?v=0gXeXtL_KjY&list=PLxqBkZuBynVS8mDTc8ZGermXiS-32pR2y&index=16)

## introduction chapter on Geometric Deep Learning

An early version of the introduction chapter on Geometric Deep Learning

Part I: symmetry from Euclid to the standard model

<https://t.co/W243bRVXnI>

[28/08, 08:07] Bhaskar@upga: \*GroupViT: Semantic Segmentation Emerges from Text Supervision\*

GroupViT is a framework for learning semantic segmentation purely from text captions without using any mask supervision. It learns to perform bottom-up hierarchical spatial grouping of semantically-related visual regions.

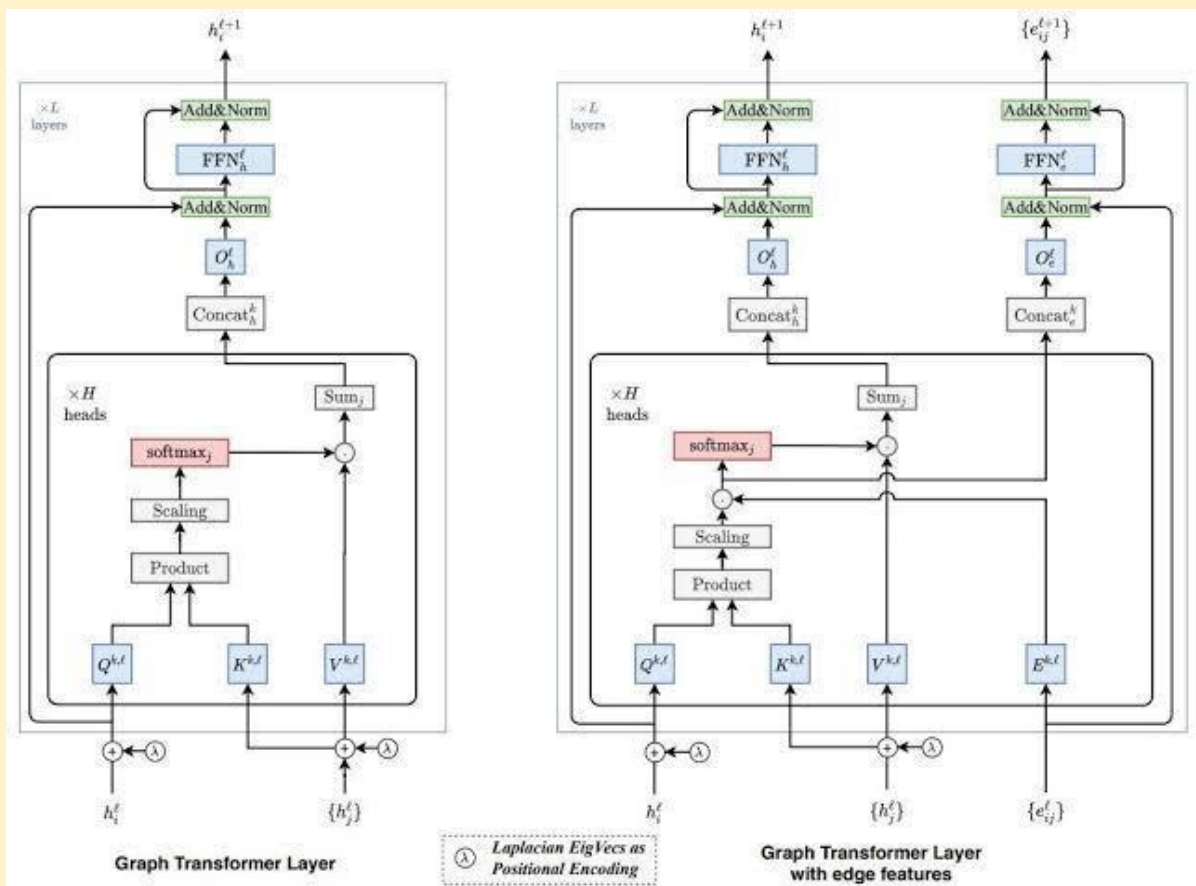
Paper : <https://arxiv.org/abs/2202.11094>

<https://github.com/NVlabs/GroupViT>



#CVPR architecture to learn zero-shot segmentation networks starting only from image-text pairs.

## Graph Transformer



## Overview of GTransformer

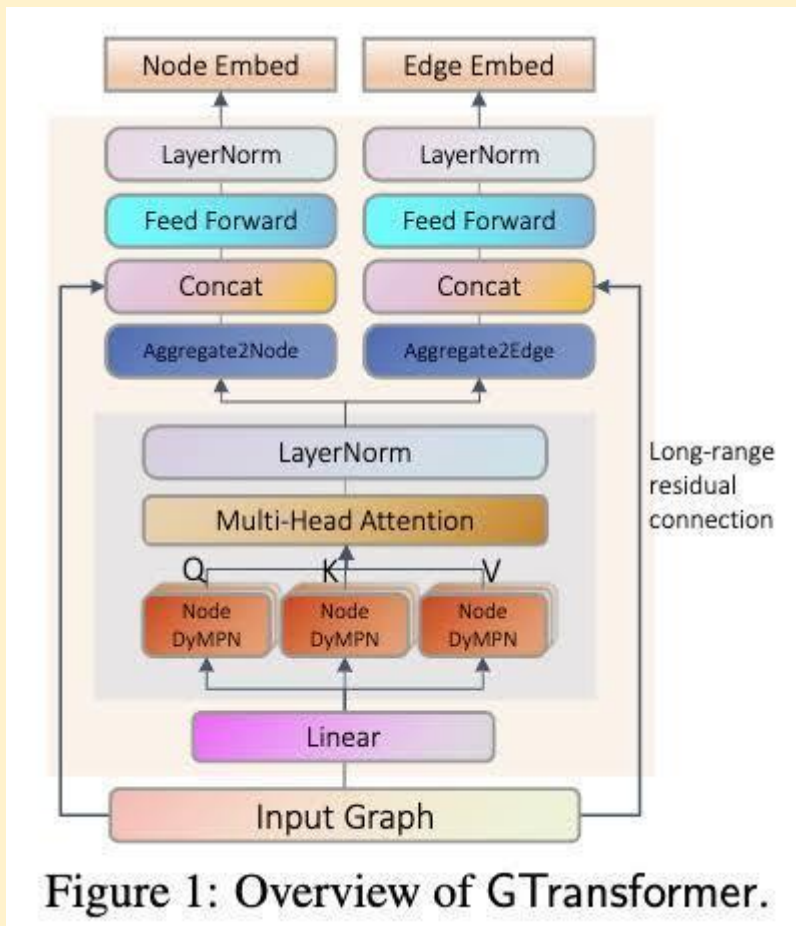


Figure 1: Overview of GTransformer.

## Roadmap to Web3



# Roadmap to Web3

- Website: [pointnetwork.io](https://pointnetwork.io)
- Run the Alpha: [pointnetwork.io/alpha](https://pointnetwork.io/alpha)
- Telegram Chat: [t.me/pointnetworkchat](https://t.me/pointnetworkchat)

## Main Zone

|                                                                                |                                                                                                                 |                                                                                                                                      |                                                                                                                                                                                  |
|--------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Point Node</b><br>Main software responsible for the network operations      | <b>Point Browser</b><br>Portable fork of Firefox with web3 tools enabled                                        | <b>Point Home</b><br>The first thing a user sees, hosted on <a href="https://point">https://point</a>                                | <b>Infrastructure</b><br>The rest of the components needed to launch beta                                                                                                        |
| <b>Core</b><br>Internal scaffolding                                            | <b>Firefox profile lifecycle management</b>                                                                     | <b>Point Wallet</b><br>UI for sending and receiving tokens and coins, transaction history etc. across multiple chains and currencies | <b>Point Network website</b><br><a href="https://pointnetwork.io">https://pointnetwork.io</a>                                                                                    |
| <b>Storage</b><br>Decentralized storage on Arweave                             | <b>Point SDK</b><br>Extra API for dApp developers                                                               | <b>Point Explorer</b><br>Explorer for Point Chain                                                                                    | <b>Faucet</b>                                                                                                                                                                    |
| <b>Blockchain</b><br>Connections to various blockchains                        | <b>Browser extension</b><br>For Point UI tools                                                                  | <b>Status/Debug</b><br>Control the state of the components from Dashboard                                                            | <b>Twitter oracle</b>                                                                                                                                                            |
| <b>Wallet</b><br>Send, receive, transaction history                            | <b>Updates</b><br>Receive browser security updates from Mozilla                                                 | <b>Software Packs</b><br>Web3 software installation for non-programmers by cloning IKV                                               | <b>Testnet infra</b>                                                                                                                                                             |
| <b>API &amp; Console</b><br>API keys for other software to interact with Point | <b>Confirmation window</b><br>To confirm transactions and actions                                               | <b>Notification Manager</b><br>Manage notifications from dApps                                                                       | <b>Arweave bundler support nodes</b>                                                                                                                                             |
| <b>Point Proxy</b><br>Local proxy that Point Browser interacts with            | <b>Mimicking Metamask API</b><br>Will allow to support all dApps on all Ethereum and Ethereum-compatible chains |                                                                                                                                      | <b>Web2 Gateway</b><br><a href="https://point.link">point.link</a>                                                                                                               |
| <b>Identities</b><br>Point DNS                                                 | <b>Notifications</b><br>Receive notifications right from dApps                                                  |                                                                                                                                      | <b>Bug tracker</b>                                                                                                                                                               |
| <b>Deployer</b><br>Tools to deploy dApps onto Point Network                    | <b>Status/Debug</b><br>Control the internal state of the network                                                |                                                                                                                                      | <b>Point Website on Web3</b>                                                                                                                                                     |
|                                                                                |                                                                                                                 |                                                                                                                                      | <b>Responsible disclosure/Bug hunting program</b><br><a href="https://pointnetwork.io/security">https://pointnetwork.io/security</a>                                             |
|                                                                                |                                                                                                                 |                                                                                                                                      | <b>Status (in the Explorer)</b>                                                                                                                                                  |
|                                                                                |                                                                                                                 |                                                                                                                                      | <b>POINT Token</b> <ul style="list-style-type: none"><li>• Point Chain</li><li>• Tokenomics</li><li>• Staking/Validators</li><li>• Token characteristics &amp; genesis</li></ul> |
|                                                                                |                                                                                                                 |                                                                                                                                      | <b>Documentation</b> <ul style="list-style-type: none"><li>• Wiki</li><li>• Whitepaper</li><li>• Litepaper</li><li>• Tutorials</li></ul>                                         |
|                                                                                |                                                                                                                 |                                                                                                                                      | <b>Bridges</b>                                                                                                                                                                   |

## Ace The SQL Interview

<https://datalemur.com/>

## Summer school on Statistical Physics & Machine learning

[https://leshouches2022.github.io/?r=21r14&utm\\_source=substack&utm\\_medium=email](https://leshouches2022.github.io/?r=21r14&utm_source=substack&utm_medium=email)

#AI #DeepLearning #DataScience #MachineLearning

## Reproducible Deep Learning

#MLOps #DeepLearning #MachineLearning #Cloud

<https://www.sscardapane.it/teaching/reproducible/>

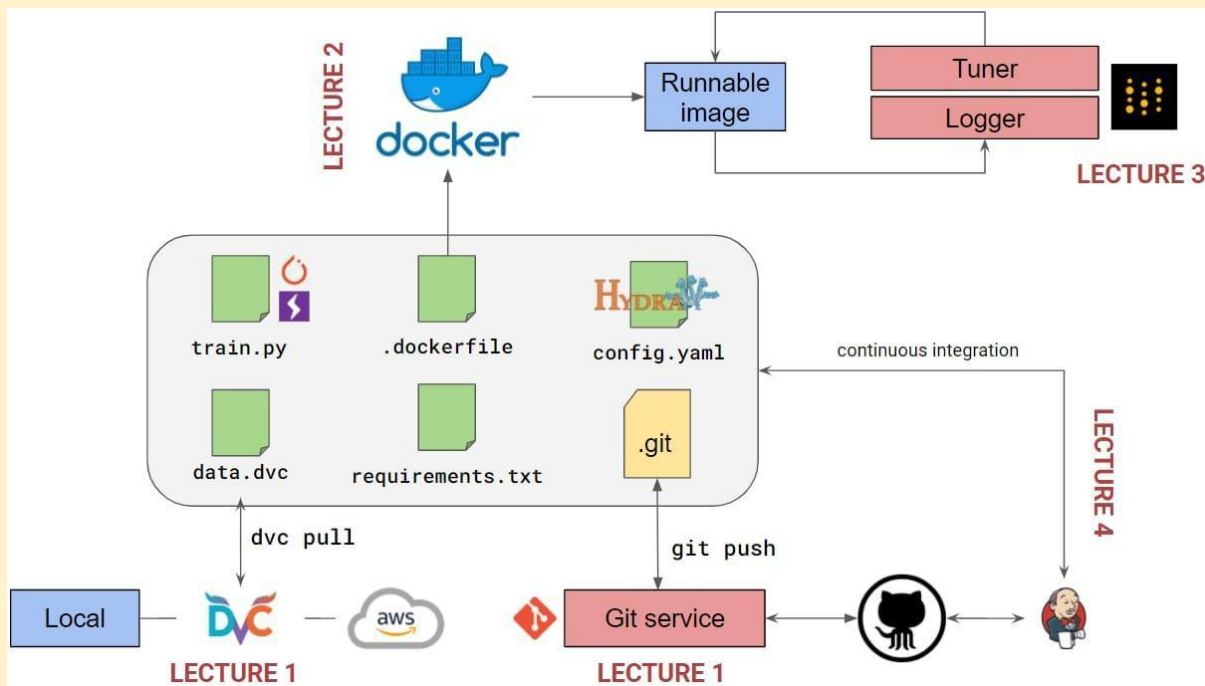
Paper : <https://arxiv.org/abs/2202.02326>

## MachineLearning & DeepLearning Concepts Playlist

Checkout #MachineLearning & #DeepLearning Concepts & #interview Question Playlist in my #YouTube channel - 62 Videos

[https://www.youtube.com/playlist?list=PLxqBkZuBynVS3RtJITF-DOxqfwT97PW\\_x](https://www.youtube.com/playlist?list=PLxqBkZuBynVS3RtJITF-DOxqfwT97PW_x)

## Docker Integration with Other Softwares



## Deck of slides on GNNs: foundations, challenges, and explainability

The key problem is how to process a graph with a layer which is (a) differentiable, (b) composable, and (c) scalable.

If you can do that, there is an absurd amount of interesting tasks you can solve, from anomaly detection on networks to forecasting, molecular conformation, etc.

Researchers or Practitioners core operator is a simple diffusion process over the graph, which is implemented by left-multiplying the (weighted) adjacency matrix. GNNs based on diffusions are simple examples of message-passing networks.

Any layer defined on a graph must satisfy permutation-equivariance, since permuting the nodes of a graph does not change its identity, and any layer should work accordingly. Message-passing is a simple way to do that.

The simplest example of a message-passing layer is graph convolution, where node features are updated and weighted-summed with a diffusion step.

From there, you can define a plethora of interesting message-passing mechanisms, from graph attentions to edge convolutions. The key idea underlying all of them is similar.

From there, you can define a plethora of interesting message-passing mechanisms, from graph attentions to edge convolutions. The key idea underlying all of them is similar.

Go thru : slides overviews many other topics, from geometrical DL to spatio-temporal graphs, learning the graph connectivity, graph transformers, and more!

Go thru 20MB Deck :  
<https://t.co/sCx39o9WP8>

And

The second set of slides talks about the fundamental issue of explaining the GNN output.

<https://t.co/ZLXC4U6PQp>

First, a warning about "explainability". Most XAI methods have serious limitations, and they are mostly useful for practitioners / experts to probe the model's output, not necessarily for the general public.

Researchers interpret explainability as defining masks over the graph layer, in order to select a relevant subgraph that is maximally correlated with the final output.

You can define these maps with classical methods (e.g., saliency maps) with small modifications. All relevant references are in the slides.

However, they tend to be hard to visualize and not compact enough.

Which leads us to tailored methods, ranging from GNNExplainer to amortized graph learning with PGExplainer.

There's more in the slides: Researchers work on meta-learning for XAI on graphs, and current XAI challenges from evaluation to finding valuable ground truths.

Slides were presented at the recent SSIE 2022 summer school, thanks again to the organizers and to all co-authors!

## **Structure-Aware Transformer for Graph Representation Learning**

<https://arxiv.org/abs/2202.03036>

**A Latest Machine Learning Research Brings A Novel Explanation For Performance Deterioration of Deeper Graph Neural Networks GNNs - MarkTechPost**

<https://www.marktechpost.com/2022/08/26/a-latest-machine-learning-research-brings-a-novel-explanation-for-performance-deterioration-of-deeper-graph-neural-networks-gnns/>

**Machine Learning and Deep Learning frameworks and libraries for large-scale data mining a survey \ SpringerLink**

<https://link.springer.com/article/10.1007/s10462-018-09679-z>

## **How Graph Neural Networks (GNN) work**

introduction to graph convolutions from scratch \ AI Summer

<https://theaisummer.com/graph-convolutional-networks/>



## What's New in v0.6.2: Continuous Deployment and a fresh CLI

ZenML Blog

<https://blog.zenml.io/zero-six-two-release/>

## Primers on training on neural network

[https://www.linkedin.com/posts/amanc\\_artificialintelligence-machinelearning-ai-activity-6969173312987942912-Oyv2?utm\\_source=share&utm\\_medium=member\\_android](https://www.linkedin.com/posts/amanc_artificialintelligence-machinelearning-ai-activity-6969173312987942912-Oyv2?utm_source=share&utm_medium=member_android)

## 8 Tips of StoryTelling



## Upgini : Free automated data enrichment library for machine learning

Only the accuracy improving features in 2 minutes

Automatically searches through thousands of ready-to-use features from public and community shared data sources and enriches your training dataset with only the relevant features

<https://github.com/upgini/upgini#readme>

## upgini Library

Upgini is a Python library that allows you to automatically search and enrich your dataset from public and scraped sources.

Among the features that can be automatically added to your dataset are: Market data, house prices for 44 countries, holidays, events, and weather.

Moreover, Upgini can automatically calculate model metrics/uplift from new features, providing you with the necessary insight into which external features are relevant to your usecase.

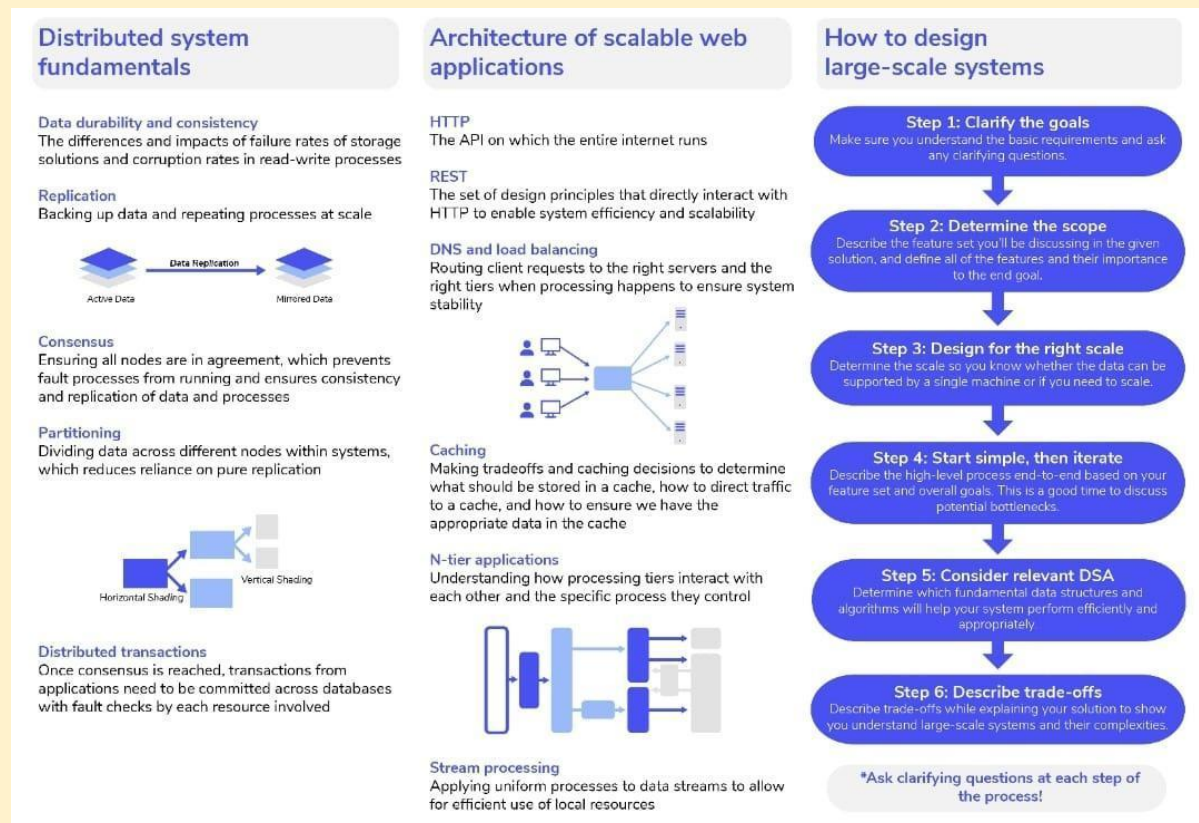
Upgini- Free automated data enrichment library for machine learning → searches through thousands of ready-to-use features from public and community shared...

<https://github.com/upgini> : Free automated data enrichment library for machine learning → searches... -

## Best Practices for Deploying Language Models

<https://txt.cohere.ai/best-practices-for-deploying-language-models/>

## Distributed System Fundamentals



## Learn web3 development by Building Projects

5 web3 Projects for Beginners.

Projects + Source code Links

1. NFT mint website

• Create an NFT from Scratch.

- learn ERC-721 token.
- Deploy on the Testnet.
- upload it to Opensea.

<https://youtu.be/OVGoJV5tcM8>

## 2. Create a ERC-20 Token

- Create your own cryptoCurrency

Add features :

- Burnable
- Mintable
- fees
- Gas fees
- Pausable
- Deflation
- Liquidity Fee

<https://docs.openzeppelin.com/contracts/4.x/erc20>

## 3. Create a DAO

Decentralized Autonomous Organization (DAO)

you can learn :

- How DAO work
- Voting process
- Governance token
- Proposal

<https://buildspace.so/daos>

## 4. UniSwap Clone

<https://youtu.be/xXxjRzdYIss>

## 5. Polling website

<https://github.com/doriancrutcher/blockvotetutorial2>

## **Adapts The Pretrained Language Image Models To Video Recognition**

Latest Computer Vision Research At Microsoft Explains How This Proposed Method Adapts The Pretrained Language Image Models To Video Recognition

Numerous vision applications heavily rely on video recognition, including autonomous driving, sports video analysis, and microvideo recommendation. A temporal video model is showcased in this research to make use of the temporal information in videos that consists of two essential parts: a multi-frame integration transformer and a cross-frame communication transformer. Additionally, the text encoder is pretrained in language image models and expanded with a video-specific prompting scheme to acquire discriminative text representation for a video.

This research utilizes text as the supervision because it contains more semantic information. Instead of starting from scratch, this approach builds on prior language-image models and expands them with video temporal modeling and video-adaptive textual prompts. An overview of the proposed framework is showcased in the figure below.

Read more in blog and paper.

<https://kaltak.in/latest-computer-vision-research-at-microsoft-explains-how-this-proposed-method-adapts-the-pretrained-language-image-models-to-video-recognition/>

Paper: <https://arxiv.org/pdf/2208.02816v1.pdf>

Github: <https://github.com/microsoft/VideoX/tree/master/X-CLIP>

## Data Analysis Video Playlists


<https://youtu.be/r-uOLxNrNk8>

<https://youtu.be/GPVsHOIRBBI>

<https://youtube.com/playlist?list=PLyMom0n-MBrpzC91Uo560S4VbsiLYtCwo>

<https://youtu.be/WcDaZ67TVRo>

## Free Backend Development & Frontend Development Hosting Platforms

 Free Backend Development & Frontend Development Hosting Platforms For Your Coding Projects

### 1. Firebase

Google's Firebase isn't just a hosting platform.

It's a BaaS (Backend as a service). It provides a variety of tools to speed up development & build quality apps 💡

So many of the tools are available in the free tier along with hosting.

 <https://t.co/QSXzTTUCvs>

### 2. Vercel

Developed by the team behind Next.js.

It's become increasingly popular, and for good reason. It's awesome — great features and great DX 😎

Deploy frontend and backend here. Generous free usage tier 🙌

 <https://t.co/XO4xEw0SX1>

### 3. AWS (Amazon Web Services)

All three IaaS (infrastructure as a service) leaders, AWS, GCP, and Azure offer a free tier for many of their tools. Including hosting — backend and frontend 💡

 <https://t.co/AHPHaSGpPs>

### 4. GCP (Google Cloud Platform)

 <https://t.co/9ETiz644ND>

## 5. Azure (Microsoft Azure)

<https://t.co/6itJR4a3wx>

## 6. Netlify

There's hosting, and there's high-quality hosting.

The dev community really loves Netlify, and for good reason.

Netlify takes care of so much and provides so many powerful features all on the free tier 😊

<https://t.co/45zkPCgv9O>

## 7. Github Pages

Arguably the easiest way to deploy your app's frontend. It is essentially an out of the box CMS 💡

Hosted directly from your GitHub repository. Just edit, push, and your changes are live 🚀

<https://t.co/bGHRnIYzAD>

## 8. Cloudflare Pages

Cloudflare Pages is a JAMstack platform for frontend developers to collaborate and deploy sites.

They're another modern hosting platform making frontend hosting super easy 🙌

<https://t.co/3ijV3KnBE5>

## 9. Surge

Really easy deployment without leaving your CLI.

Simple, single-command web publishing. Publish HTML, CSS, and JS for free, without leaving the command line.

Just install their CLI tool and deploy 🚀

<https://t.co/QHXpRnzMTP>

## 6 ways to use Jupyter notebook

Jupyter Notebook is the most powerful tool Python developers have.

Explore the hidden features.

Need a quick web app?

Or to create REST APIs?

Here's the 6 ways to use Jupyter notebook you never knew existed:

Jupyter Notebook is a web app for creating and sharing computational documents.

When I say powerful, here's what I mean:

- It supports 40 languages



- Produces rich, interactive output
- Can leverage big data tools like Spark

So, what else can we do with Jupyter Notebook?

## 1.Package Development

nbdev let's you develop and publish Python packages right from Jupyter Notebook.

It generates documentation and publishes on GitHub Pages. You can also write tests and setup CI with GitHub Actions.

<https://github.com/fastai/nbdev>

## 2.Web Apps

With Mercury, you can build an interactive web app.

Define the interactive widgets and deploy the app to any server. When users interact with the app, they can save the results as PDF or HTML.

<https://github.com/mljar/mercury>

## 3.Books

Jupyter Book is a tool for building publication-quality books and documents.

Start by building the notebook and finish by publishing as a website or exported as a PDF file.

<https://mljar.com/blog/how-to-use-jupyter-notebook/>

## 4.Blogs

Create a blog with Nikola.

Nikola reads your notebooks and produces the static HTML and CSS for the website. It comes with themes, feeds and comments.

<https://getnikola.com/>

## 5.Reports

Impress your colleagues by sending them a report built by nbconvert.

nbconvert converts notebooks into PDF, HTML, Markdown and Latex. Converting notebooks lets you share results with people not using notebooks.

<https://nbconvert.readthedocs.io/en/latest/>

## 6.REST API

Jupyter Kernal Gateway turns each cell into a REST API endpoint.

Use the Kernal Gateway to define many endpoints in a single notebook. It also supports Websockets.

<https://jupyter-kernel-gateway.readthedocs.io/en/latest/features.html>

Whether it's algorithmic trading, big data analysis or web development, use Jupyter Notebook for your next project.

- Package development

- Web apps
- REST API
- Reports
- Books
- Blogs

And that's a wrap!

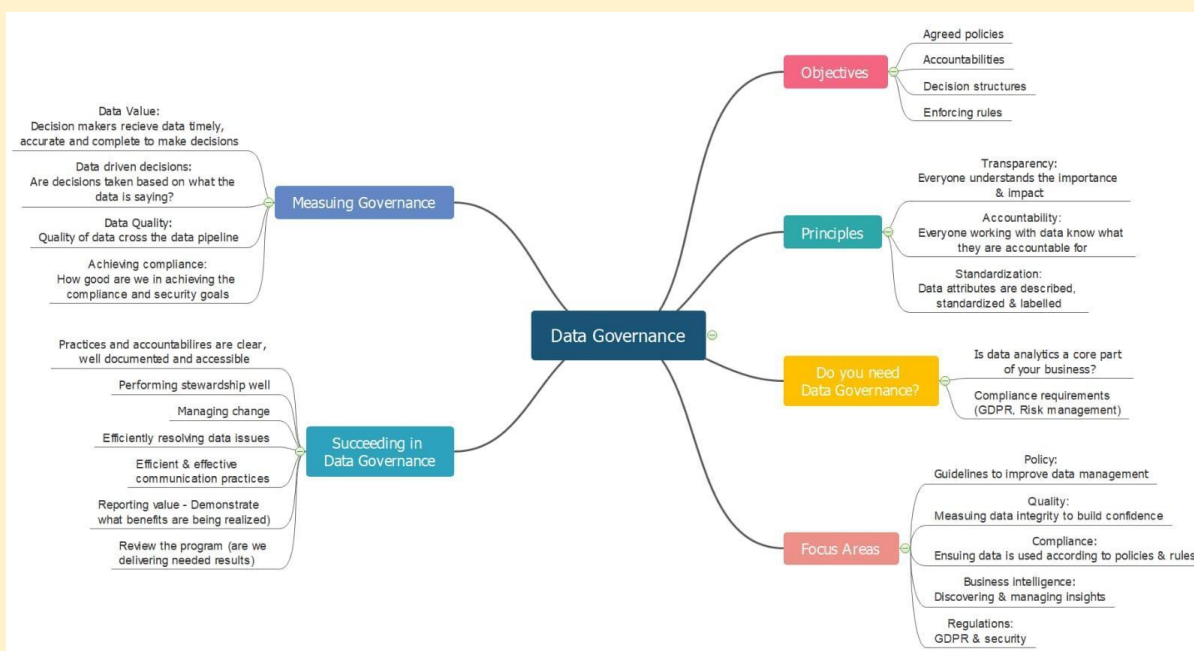
PyQuant News is the best place for resources for developers using Python for scientific computing and quantitative analysis.

Check out <https://pyquantnews.com/>

## 21 Cyber Security Search Engines

1. Shodan-Search for devices connected to the internet.
2. Wigle-Database of wireless networks, with statistics.
3. Grep App-Search across a half million git repos.
4. Binary Edge-Scans the internet for threat intelligence.
5. ONYPHE-Collects cyber-threat intelligence data.
6. GreyNoise-Search for devices connected to the internet.
7. Censys-Assessing attack surface for internet connected devices.
8. Hunter-Search for email addresses belonging to a website.
9. Fofa-Search for various threat intelligence.
10. ZoomEye-Gather information about targets.
11. LeakIX-Search publicly indexed information.
12. IntelligenceX-Search Tor, 12P, data leaks, domains, and emails.
13. Netlas-Search and monitor internet connected assets.
14. URL Scan-Free service to scan and analyse websites.
15. PublicWWW-Marketing and affiliate marketing research.
16. FullHunt-Search and discovery attack surfaces.
17. CRT sh-Search for certs that have been logged by CT.
18. Vulners-Search vulnerabilities in a large database.
- 19 Pulsedive-Search for threat intelligence.
20. Packet Storm Security-Browse latest vulnerabilities and exploits.
21. GrayHatWarefare-Search public S3 buckets.

## What is Data Governance



# Skills & Knowledge of Data Management



## Machine learning systems design

<https://huyenchip.com/machine-learning-systems-design/toc.html>

## Introduction to Machine Learning Interviews Book · MLIB

<https://huyenchip.com/ml-interviews-book/>

## What is DALL-E, DALL-E2?

What is DALL-E, DALL-E2?

DALL-E (stylized as DALL·E) and DALL-E 2 are machine learning models developed by OpenAI to generate digital images from natural language descriptions. DALL-E was revealed by OpenAI in a blog post in January 2021, and uses a version of GPT-3[1] modified to generate images. In April 2022, OpenAI announced DALL-E 2, a successor designed to generate more realistic images at higher resolutions that "can combine concepts, attributes, and styles".

<https://en.wikipedia.org/wiki/DALL-E>

DALL-E2 is a new AI system that can create realistic images and art from a description in natural language.

<https://openai.com/dall-e-2/>

DALL-E mini is an online text-to-image generator that has exploded in popularity on social media in recent weeks.

What does DALL-E 2 stand for?

On the most basic level, DALLE-2 is a function that maps text to images with remarkable accuracy, producing high quality and vibrant output images.

Videos:

[https://youtu.be/O\\_j\\_7Zdt7hg](https://youtu.be/O_j_7Zdt7hg)

How DALL-E 2 Actually Works?

<https://www.assemblyai.com/blog/how-dall-e-2-actually-works/>

How is it so good ? (DALL-E Explained Pt. 2)

By Charlie Snell

<https://ml.berkeley.edu/blog/posts/dalle2/>

The neural network architecture of @OpenAI DALL-E2 is now open source!

GitHub Repo - <https://t.co/lhN7xwiz8P>

#dalle #PyTorch #Python #openai #OpenSource

## Four major smoothing techniques

In a Latest Machine Learning Research, Salesforce AI Team Developed a New Time-Series Forecasting Model Called ETSformer Which Exploits The Principle of Exponential Smoothing in Improving Transformers for Time-Series Forecasting

How smoothing Methods can be used to forecast a time series?

Forecasts produced using exponential smoothing methods are weighted averages of past observations, with the weights decaying exponentially as the observations get older. In other words, the more recent the observation the higher the associated weight.

Exponential Smoothing (ETS) is a commonly-used local statistical algorithm for time-series forecasting.

The Amazon Forecast ETS algorithm calls the ets function in the Package 'forecast' of the Comprehensive



R Archive Network (CRAN).

Exponential smoothing is a rule of thumb technique for smoothing time series data using the exponential window function. Whereas in the simple moving average the past observations are weighted equally, exponential functions are used to assign exponentially decreasing weights over time.

Transformers and Time Series Forecasting

The Transformer learns an information-passing graph between its inputs. Because they do not analyze their input sequentially, Transformers largely solve the vanishing gradient problem that hinders Recurrent Neural Networks (RNNs) in long-term prediction.

Four major smoothing techniques:

1. Moving average smoothing.
2. Exponential smoothing.
3. Double exponential smoothing.
4. Triple exponential smoothing.

Paper: <https://arxiv.org/pdf/2202.01381.pdf>

Github: <https://github.com/salesforce/ETSformer>

Other References :

<https://docs.aws.amazon.com/forecast/latest/dg/aws-forecast-recipe-ets.html>

## **b7leung/ML-Flashcards**

**200+ detailed flashcards useful for reviewing topics in machine learning, computer vision, and computer science.**

<https://github.com/b7leung/ML-Flashcards>

## **Starting Point of Data science for various people**

You're super motivated to learn data science/analysis but don't know where to start?

Here's a start to get familiar with data area:

Read & open your mind: <https://t.co/LvG7SDMXGT>

There's tons of amazing articles, it's very very useful. It gives tons ideas of what you can do.

Skills to deepen: Python, Dataviz, Tableau, Excel...

For python, I like <https://t.co/ftxn8IzpBp> and <https://t.co/wdVDHeOluz> to develop python skills.

For dataviz, have a look at below resources:

1. For Javascript coders, Explore chart.js gallery to help you find perfect graph to tell your story <https://t.co/1dyabnnjML>. It's highly customizable, interactive and code snippets are very useful.

2. Another very cool Javascript library, D3.js: <https://t.co/XshCfnISBt>

3. For python coders, Plotly is a nice library to draw interactive charts: <https://t.co/hho7BqI981>

4. Another great python library: <https://t.co/INNTOFrYZT>

5. For non-coders, Tableau is one of the leader to create dashboards: <https://t.co/toLwrXjEk1>, and it's free



for students & teachers.

6. Microsoft Excel, it's worth mentioning it. I'm sure you have tons of spreadsheets in your laptop, why don't you try to draw charts?

7. Datawrapper,

Free tools for non-coders to create & export charts, maps.

<https://t.co/hiH2rdh2W8>

Practice!

8. Install @pycharm, anaconda, or whatever IDE you want, and PRACTICE what you learned.

Theory will never give you experience you get when you practice. You can execute code written in articles you read or by build your own personal projects.

9. Have fun with playbook resources:

<https://t.co/9zWJca4kiM>

10. Bookcamp

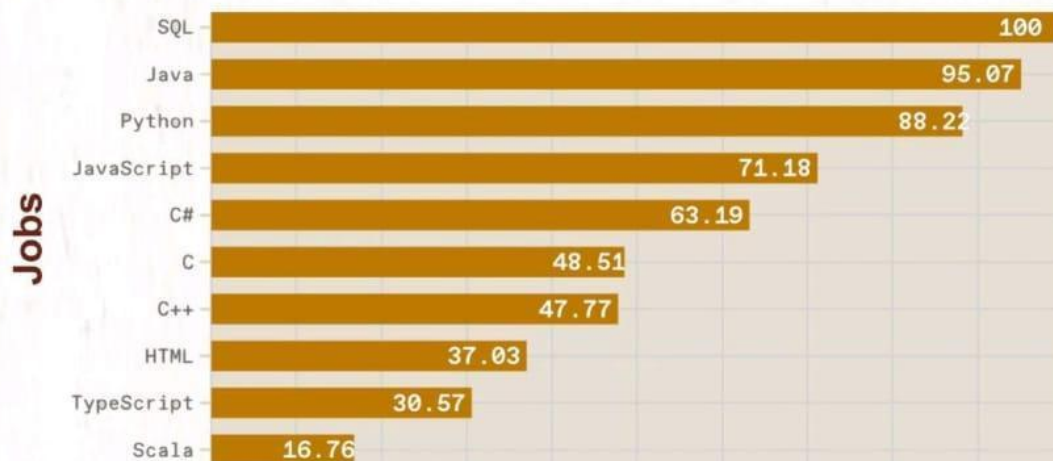
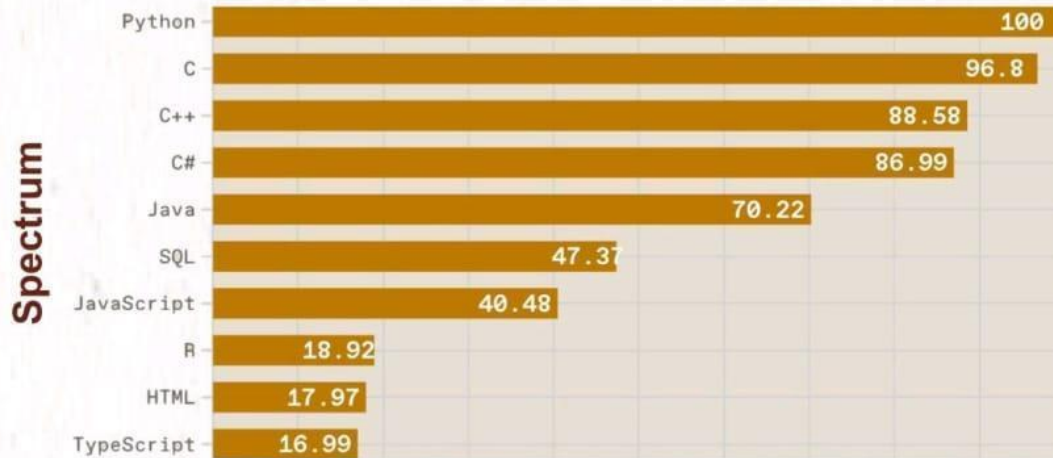
<https://github.com/alexeygrigorev/mlbookcamp-code/tree/master/course-zoomcamp>

# Top Programming Languages of 2022

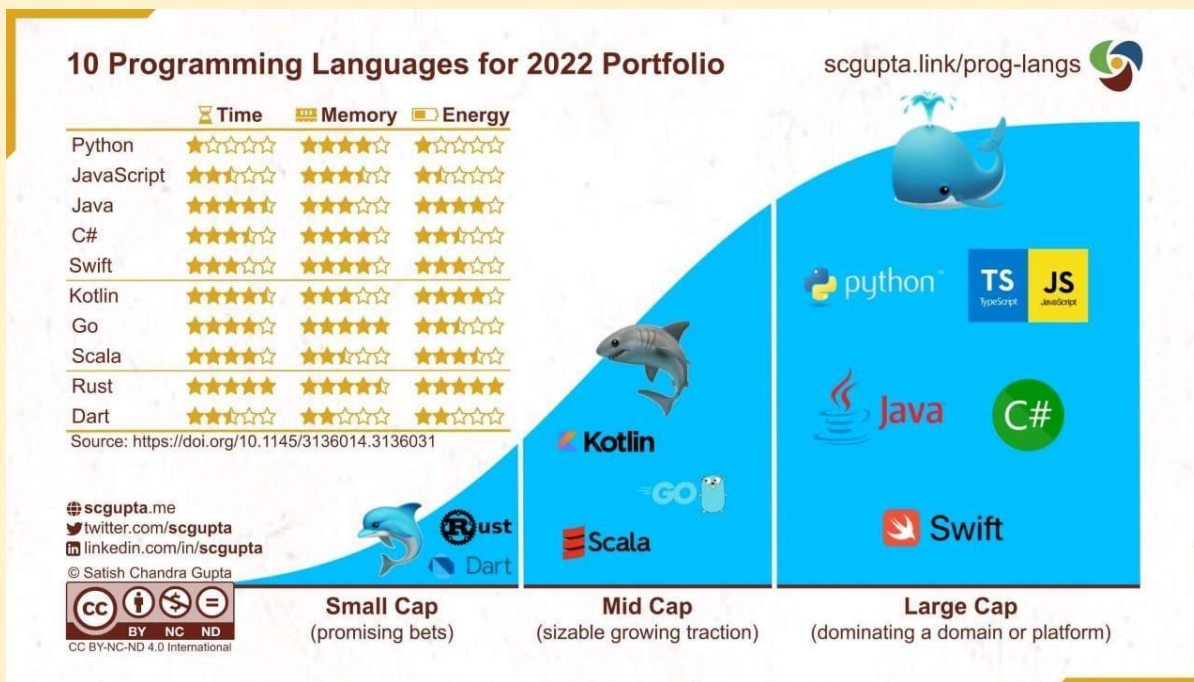
## Top Programming Languages 2022 – IEEE Spectrum



Source: <https://spectrum.ieee.org/top-programming-languages-2022>



## 10 Programming Languages and their efficiency



## maikherbig/AIDeveloper

GUI-based software for training, evaluating and applying deep neural nets for image classification

<https://github.com/maikherbig/AIDeveloper>

## Los Alamos Claims Quantum Machine Learning Breakthrough

Training with Small Amounts of Data - High-Performance Computing News Analysis \ insideHPC

<https://insidehpc.com/2022/08/los-alamos-claims-quantum-machine-learning-breakthrough-training-with-sm-all-amounts-of-data/>

## How Google uses NLP to better understand search queries, content

<https://searchengineland.com/how-google-uses-nlp-to-better-understand-search-queries-content-387340>

# Essential Charts for Data Analysis

## Raw Number

- 10** **Single Value Chart**  
Show a raw singular value
- 10** **Single Value w/ Indicator**  
Comparison of a single value against a previous value
- Bullet Chart**  
Comparison of a single value against a benchmark value
- Table**  
Show raw values for multiple data points on multiple variables

## Change over Time

- Line Chart**  
Change over time for a numeric variable or to compare 1-5 groups
- Sparkline**  
Miniature line charts to compare many groups

## Distribution

- Bar Chart**  
Comparison or distribution by a single categorical variable
- Histogram**  
Distribution by a binned single numeric variable
- Box Plot**  
Compare distribution summaries across a categorical variable

## Part-to-Whole

- Pie Chart**  
Part-to-whole breakdown by a single categorical variable
- Stacked Bar Chart**  
Bar chart with additional part-to-whole breakdown
- Stacked Area Chart**  
Line chart with additional part-to-whole breakdown

## Relationship

- Scatter Plot**  
Relationship between two numeric variables
- Bubble Chart**  
Relationship between three numeric variables
- Grouped Bar Chart**  
Comparison or distribution by two categorical variables
- Heatmap**  
Distribution by two binned variables (categorical or numeric)

## Geospatial

- Bubble Map**  
Bubble chart built on top of a geographic map
- Choropleth**  
Comparison between geopolitical regions by color

## Facebook AI Researchers Open-Source 'LLM.int8()'

Tool To Perform Inference In Large Language Models (LLMs)

Facebook AI Researchers Open-Source 'LLM.int8()' Tool To Perform Inference In Large Language Models (LLMs) With Up To 175B Parameters Without Any Performance Degradation

Large pretrained language models are frequently used in NLP, although inference requires substantial memory. The feed-forward and attention projection layers, along with associated matrix multiplication operations, are in charge of 95% of the consumed parameters and 65-85% of the total computation for large transformer language models at and beyond 6.7B parameters. Utilizing low-bit-precision matrix multiplication and quantizing the parameters to utilize fewer bits is one method of reducing their size. 8-bit quantization techniques for transformers have been created with this objective in mind. These techniques decrease memory use but also harm performance, need further quantization after training, and have only been tested for models with less than 350M parameters.

Up to 350M parameters, degradation-free quantization is challenging to understand, and multi-billion parameter quantization is still tricky. This research describes the first performance-unaffected multi-billion-scale Int8 quantization method for transformers. The method may change the feed-forward and attention projection layers from 16-bit to 32-bit weights. The resultant model can be used for inference immediately without suffering any performance penalties. By overcoming two significant obstacles: One is the need for higher quantization precision at scales above 1B parameters. Second, the requirement to explicitly represent the sparse but systematic extensive magnitude outlier features appearing in all transformer layers beginning at scales of 6.7B parameters can achieve this result.

Read more.....

Paper: <https://arxiv.org/pdf/2208.07339v1.pdf>

Github link: <https://github.com/timdettmers/bitsandbytes>

LLM.int8(): 8-bit Matrix Multiplication for Transformers at Scale

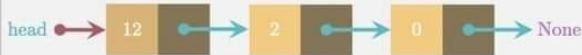
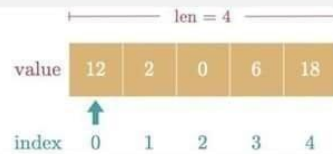
<https://deepai.org/publication/llm-int8-8-bit-matrix-multiplication-for-transformers-at-scale>



# Cheat Sheet – Data Structures

## 1. List

- Ordered collection of elements
- The position of each element is defined by the *index*
- The elements can be accessed in *any order*

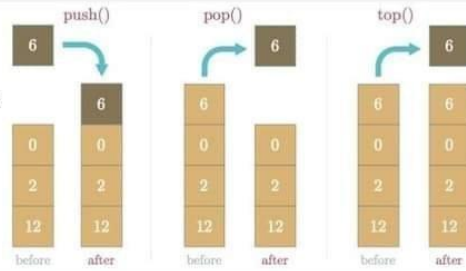


## 2. Linked List

- Each linked list element contains both the values and the address (pointer) to the next linked list element.
- Hence the linked list can only be traversed sequentially going through each element at a time
- Linked List does not have their order defined by their physical placement in the memory.
- Contiguous elements of the linked list are not placed adjacent to each other in the memory.

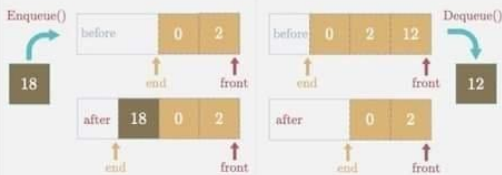
## 3. Stack

- Stack is a **sequential data structure** which maintains the order of elements as they were inserted in.
- Last In First Out (LIFO) order, which means that the elements can only be accessed in the *reverse order* as they were inserted into the stack.
- The element to be *inserted last*, will be the *first one to get removed* from the stack.
- **Push()** adds an element at the *head* of the stack, while **pop()** removes an element from the *head* of the stack
- A real-life example of a stack is a stack of kitchen plates



## 4. Queue

- A queue is a **sequential data structure** that maintains the order of elements as they were inserted in
- First In First Out (FIFO), the element to be *inserted first*, will be the *first one to get removed* from the queue
- Whenever an element is added (Enqueue()) it is added to the *end* of the queue. On the other hand, element removal (Dequeue()) is done from the *front* of the queue.
- A real-life example is a check-out line at a grocery store

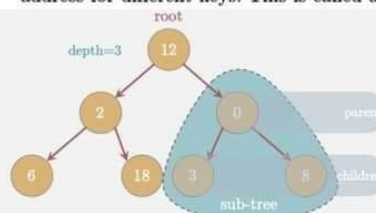


## 5. HashTable

- Creates *paired assignments* (key mapped to values) so the pairs can be accessed in constant time
- For each (key, value) pair, the key is passed through a hash function to create a unique physical address for the value to be stored in the memory.
- Hash function can end up generating the same physical address for different keys. This is called a *collision*.



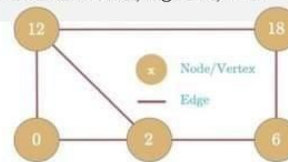
## 4. Tree



- Maintains a hierarchical relation between its elements.
- **Root Node** – The node at the top of the tree
- **Parent Node** – Any node that has at least one child
- **Child Node** – The successor of a parent node is known as a child node. A node can be both a parent and a child node. The root is never a child node.
- **Leaf Node** – The node which does not have any child node.
- **Traversing** – Passing through the nodes in a certain order, e.g BFS, DFS

## 4. Graph

- A graph is a pair of sets (V, E), where V is set of all the vertices, E is set of all edges.
- A neighbor of a node is set of all vertices connected with that node through an edge.
- As opposed to trees, a graph can be cyclic, which means starting from a node and following the edges, you can end up on the same node



Source: <https://www.cheatsheets.aqeel-anwar.com> Tutorial: [Click here](#)



## Helpful AI tools for content writing and how to use them

<https://aicontentmarketing.voiceoferos.cyou/helpful-ai-tools-for-content-writing-and-how-to-use-them/>

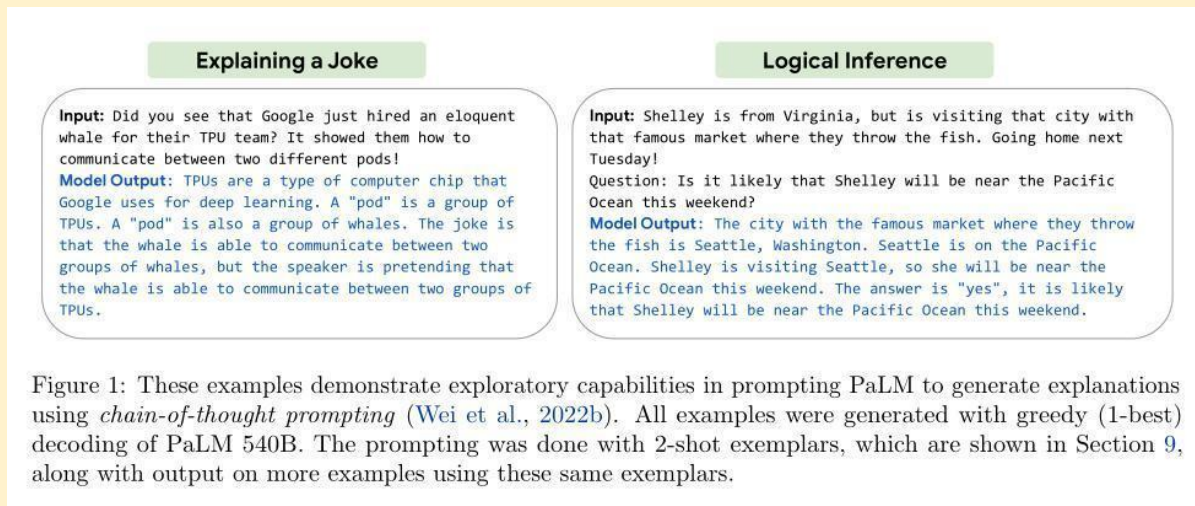
## PaLM: Scaling Language Modeling with Pathways

<https://t.co/A1rI0ITAh1>

Large language models have been shown to achieve remarkable performance across a variety of natural language tasks using few-shot learning. Researchers train a 540-billion parameter, densely activated, Transformer language model, which they call Pathways Language Model PaLM. Researchers demonstrate continued benefits of scaling by achieving state of the art model.

Introducing the 540 billion parameter Pathways Language Model. Trained on two Cloud #TPU v4 pods, it achieves state-of-the-art performance on benchmarks and shows exciting capabilities like mathematical reasoning, code writing, and even explaining jokes. <https://t.co/NFHFAHgUkB>

## NLP - Explaining Joke



## CS388 NLP Course

<https://www.cs.utexas.edu/~gdurrett/courses/online-course/materials.html>

## machine learning algorithms, data science infographics

<http://discuss.analyticsvidhya.com/t/download-full-cheatsheet-on-machine-learning-algorithms/4063/1>

## Data Science Interview Questions and Answers Blogs - 360DigiTMG

<https://360digitmg.com/blog-category/data-science-interview-questions>

## Artificial Neural Network Questions to Test Your Skills

<https://www.analyticsvidhya.com/blog/2021/05/artificial-neural-networks-25-questions-to-test-your-skills-on-ann/>

## 20 Questions to practice for machine learning interviews.

These questions focus mostly on neural networks. They cover some fundamental concepts you should know.

1. Why is it important to introduce non-linearities in a neural network?

2. What are the differences between a multi-class classification problem and a multi-label classification problem?
3. Why does the use of Dropout work as a regularizer?
4. Why you shouldn't use a softmax output activation function in a multi-label classification problem?
5. Does the use of Dropout in your model slow down or speed up the training process?
6. Is gradient descent guaranteed to find the global minima? Why so?
7. Explain the difference between Batch Gradient Descent, Stochastic Gradient Descent, and Mini-batch Gradient Descent.
8. What are the advantages of Convolution Neural Networks (CNN) over a fully connected network for image classification?
9. What are the advantages of Recurrent Neural Networks (RNN) over a fully connected network when working with text data?
10. How do you deal with the vanishing gradient problem?
11. How do you deal with the exploding gradient problem?
12. Are feature engineering and feature extraction still needed when applying Deep Learning?
13. How does Batch Normalization help?
14. The universal approximation theorem shows that any function can be approximated as closely as needed using a single nonlinearity. Then why do we use more?
15. What are some of the limitations of Deep Learning?
16. Why is initializing the weights of a network important?
17. The training loss of your model is too high. What does it mean? What can you do?
18. Assuming you are using Batch Gradient Descent, what advantage would you get from shuffling your training dataset?
19. Compare the following evaluation protocols: a hold-out validation set, K-fold cross-validation, and iterated K-fold validation. When would you use each one?
20. What are the main differences between Adam and the Gradient Descent optimization algorithms?

## **GAN from GNN To Correctly Predict the Coherent Orientations of Galaxies**

Using Graph Neural Networks, CMU Researchers Trained Generative Adversarial Networks to Correctly Predict the Coherent Orientations of Galaxies in a State-of-the-Art Cosmological Simulation

Blog :

<https://appsforpcfree.net/using-graph-neural-networks-cmu-researchers-trained-generative-adversarial-networks-to-correctly-predict-the-coherent-orientations-of-galaxies-in-a-state-of-the-art-cosmological-simulation/>

Galaxies on Graph Neural Networks

<https://blog.ml.cmu.edu/2022/08/19/galaxies-on-graph-neural-networks/>

Paper: <https://arxiv.org/pdf/2204.07077.pdf>

## **Build Machine Learning Pipeline in Python and Deploy on Cloud easily**

by Moez Ali \ Aug, 2022 \ Medium

<https://moez-62905.medium.com/build-machine-learning-pipeline-in-python-and-deploy-on-cloud-easily-2f907ff1af1f>

## **The Rise of Data Contracts**

by Chad Sanderson

<https://dataproducts.substack.com/p/the-rise-of-data-contracts>

## Workshop on Autonomous Driving - CVPR 2022

Workshop on Autonomous Driving(WAD) just shared the recordings of all talks. Might be useful for people interested in autonomous driving.

Recordings: <https://t.co/uL2pVhVIWZ>

About WAD :<https://cvpr2022.wad.vision/>

## Google AI Introduces OptFormer

Google AI Introduces OptFormer: The First Transformer-Based Framework For Hyperparameter Tuning

OpenML and other public machine learning data platforms, along with hyperparameter optimization (HPO) services like Google Vizier, Amazon SageMaker, and Microsoft Azure, have facilitated the availability of comprehensive datasets with hyperparameter assessments. Optimization of hyperparameters is crucial in machine learning since they can make or break a model's performance on a given task.

There is a growing interest in using this kind of information to meta-learn hyperparameter optimization (HPO) algorithms. Still, working with large datasets that include experimental trials in the wild can be difficult due to the wide variety of HPO problems and the text metadata that describes them. Consequently, most meta- and transfer-learning HPO approaches consider a constrained environment where all tasks must share the same hyperparameters so that the input data can be represented as fixed-sized vectors. As a result, the data used to learn priors using such methods is limited. This is a particularly serious problem for huge datasets that include valuable information.

Google AI has developed the OptFormer, one of the first Transformer-based frameworks for hyperparameter tuning, which can learn from massive amounts of optimization data by employing versatile textual representations.

Earlier works have shown the Transformer's versatility. However, not many pieces of research focused on its potential for optimization, particularly in the realm of text. The paper "Towards Learning Universal Hyperparameter Optimizers with Transformers" presents a meta-learning HPO system that is the first to learn policy and function priors from data in multiple search spaces simultaneously.

Unlike traditional approaches, which often only use numerical data, the proposed method uses concepts from natural language and depicts all of the research data as a series of tokens, including textual information from the original metadata.

The T5X codebase is used to train the OptFormer in a conventional encoder-decoder fashion with standard generative pretraining for several hyperparameter optimization objectives, including Google Vizier's real-world data and public hyperparameter (HPO-B) and black-box optimization benchmarks (BBOB). The OPTFORMER can generalize the habits of seven distinct black-box optimization algorithms (non-adaptive, evolutionary, and Bayesian).

Paper: <https://arxiv.org/pdf/2205.13320.pdf>

## The ML Reproducibility Challenge is back!

The 6th edition now expands to papers from 11 ML conferences:

1. NeurIPS,
2. ICML,
3. ICLR,
4. ACL,
5. EMNLP,
6. CVPR,
7. ECCV,

8. AAAI,
9. IJCAI-ECAI,
10. ACM
11. FAccT & SIGIR.

ML journals included this year:

1. JMLR,
2. TACL, &
3. TMLR.

Read more: <https://t.co/tTHElxo1XI>

**Textualize/rich: Rich is a Python library for rich text and beautiful formatting in the terminal.**

<https://github.com/Textualize/rich>

**30 Projects Ideas You Can Build to Become a Blockchain & Web3 Master**

Beginner Projects

- ERC-20 Token
- Crypto Wallet
- NFT collection
- Time-lock wallet
- NFT minting page
- Blockchain-based image storage
- DAO
- NFT Game
- Web3 Forum
- Todo-list Dapp

Intermediate Projects

- Crowdfunding App
- Peer To Peer Ridesharing
- Decentralised blog application
- NFT Marketplace
- A decentralized music application
- Uniswap clone
- Disk Space Renting System
- ICO
- E-commerce application using Solana Pay
- Domain service with Polygon
- Chat application

Advanced Projects

- Blockchain-Based Voting System
- Decentralized Crypto Exchange
- Social media application
- A Decentralized Web Hosting System
- Video sharing application
- Decentralized Notion clone
- Crypto Lending Platform
- Crypto trading bot
- DeFi application



# Masked Language Model

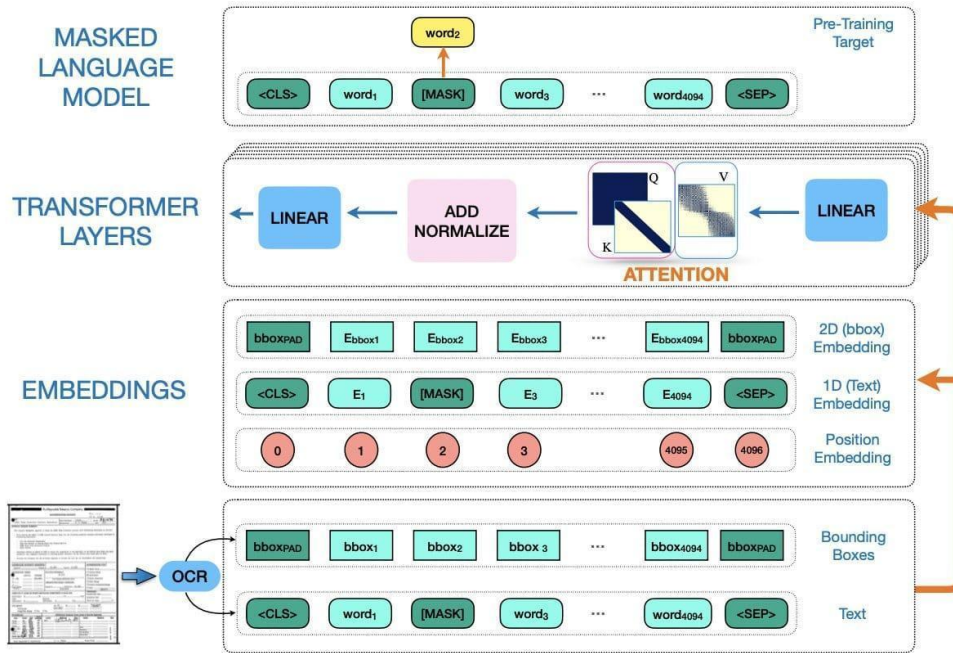
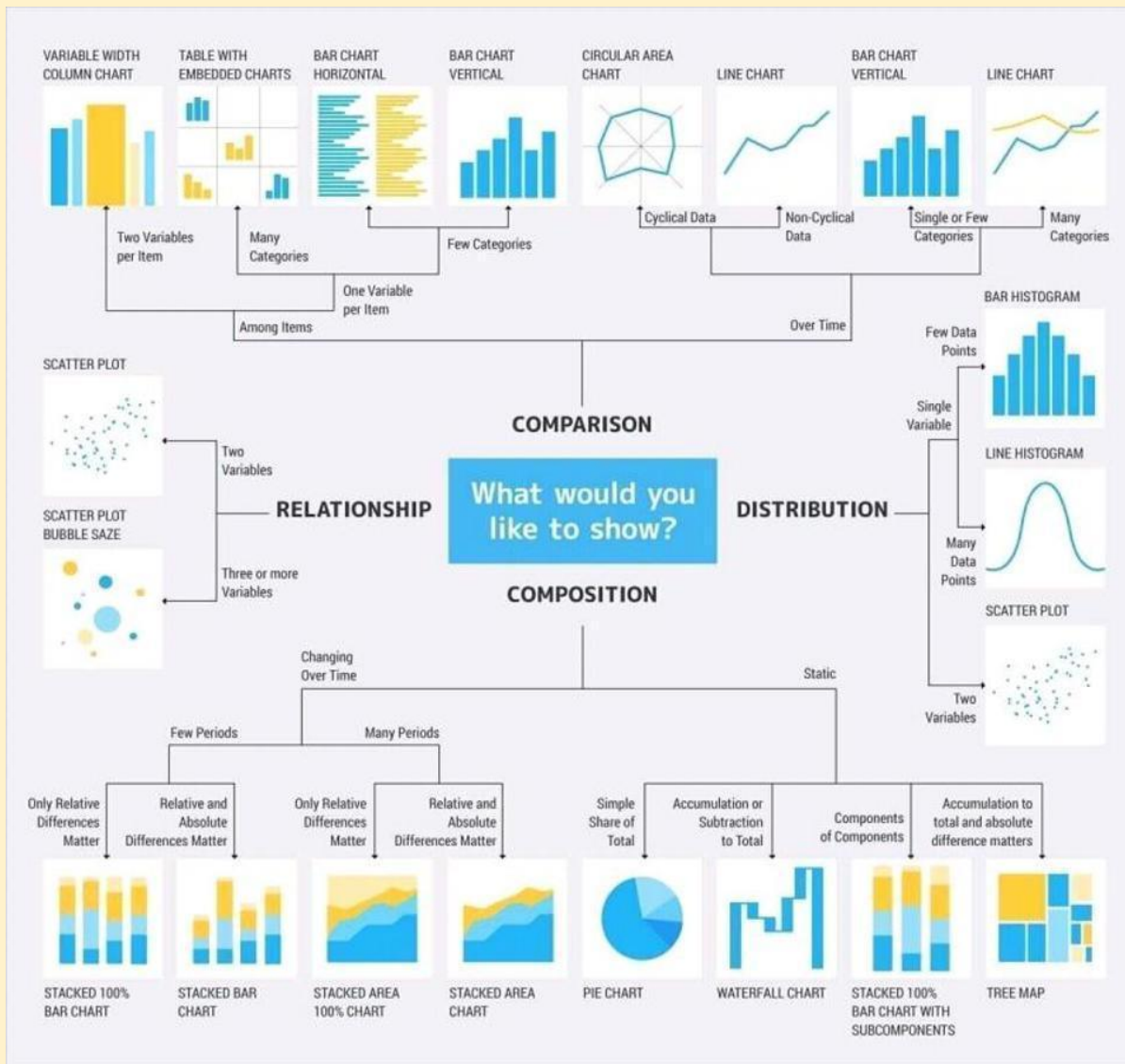


Figure 2: Our pretrain model architecture. Unlike other models for this task, we keep a simple approach by only employing a single MLM pretraining objective and do not employ extra overhead into multimodal embedding or encoding methods. Instead, we tackle the attention module directly and make necessary changes to deal with our focus on long documents, by flexibly using 1D and 2D input.

## Which Graph



## 40 Best Resources to Learn Natural Language Processing [FREE]- 2022

<https://www.mltut.com/best-resources-to-learn-natural-language-processing/>

## ConvNetJS MNIST demo

<https://cs.stanford.edu/~karpathy/convnetjs/demo/mnist.html>

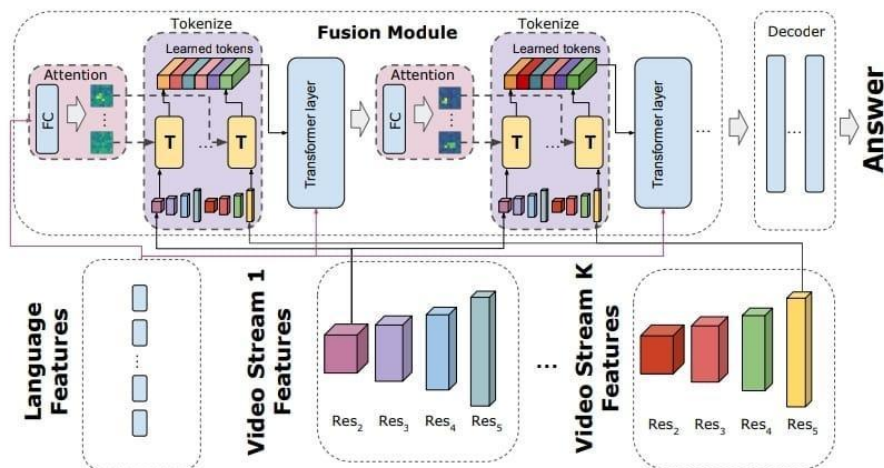


Fig. 3: Illustration of the iterative fusion module in detail. The features are used to generate  $N$  attention maps, which are applied to each stream and (optionally) scale of the video features. This results in a fixed number of tokens for the  $K$  streams and  $S$  scales ( $K$  and  $S$  are typically very small). The tokens are fed through a transformer layer, generating a new feature, which is used to generate new attention maps. This processes is repeated multiple times. The yellow boxes (denoted as 'T') represent the application of an attention map to a feature map.

**PyTorch Data Parallel Best Practices on Google Cloud**

by PyTorch \ PyTorch \ Medium

<https://medium.com/pytorch/pytorch-data-parallel-best-practices-on-google-cloud-6c8da2be180d>

Assessed the effectiveness of the DDP, Pipe and FSDP distributed training techniques available via PyTorch with different model sizes and network configurations. See the results

Data Parallel Best Practices on Google Cloud.  
 #AI #ML #ArtificialIntelligence #MachineLearning #PyTorch

**Causality for Machine Learning**

<https://ff13.fastforwardlabs.com/>

**Foundations of Causal Inference and its impact on Machine Learning**

<https://youtu.be/LALfQStONEc>

**How Artificial Intelligence Is Changing The Law Industry for The Better • LegalScoops**

<https://www.legalscoops.com/how-artificial-intelligence-is-changing-the-law-industry-for-the-better/>

## Everything you've ever wanted to know about machine learning

by Cassie Kozyrkov \ Aug, 2022 \ Medium

<https://kozyrkov.medium.com/everything-youve-ever-wanted-to-know-about-machine-learning-b396b0abee8c>

## Machine learning for sensors

<https://phys.org/news/2019-06-machine-sensors.html>

## Recent machine learning research tutorial focuses on Learning Theory

Recent machine learning research tutorial focuses on Learning Theory and is therefore math-intensive. To cater for people whom are new to ML, go thru a new set of foundational mathematics machine learning notes:

<https://github.com/roboticcam/machine-learning-notes>

containing the following topics: Model Evaluation, Decision Tree, Simple Bayes, Regression, Neural Network and Unsupervised Learning. Specifically:

Class 1: Model Evaluation

common concepts and techniques for classification model evaluation, including bootstrapping sampling, confusion matrices, receiver operating characteristic (ROC) curves.

Class 2: Decision Tree

In addition to all the basics of decision trees, I've added a  $\chi^2$  test section to this note.

Class 3: Simple Bayes

This note is intended to provide an intuitive explanation of the basic concepts of probability, Bayes' theorem, graphical models of probability.

Class 4: Regression

This note is to explain the century-old, simplest regression models: linear and polynomial regression, and some techniques for evaluating regression performance, especially the coefficient of determination (CoD) method.

Class 5: Neural Network

First I show three different last output layer models: logistic, multinomial, and linear regression. Then I show the concept of gradient descent. The main part is to show a basic fully connected neural network and finally a convolutional neural network.

Class 6: Unsupervised Learning

This note describes some common topics in unsupervised learning. From the most obvious methods like clustering, to topic modeling (Latent Dirichlet Allocation) and traditional word embeddings like the word2vec algorithm.

I used simple mathematics to explain them. Hope they are useful to you!

Github

<https://github.com/roboticcam/machine-learning-notes>

## Top And Easy to use Open-Source Image Labelling Tools for Machine Learning Projects

by MRINAL WALIA \ Becoming Human: Artificial Intelligence Magazine

<https://becominghuman.ai/top-and-easy-to-use-open-source-image-labelling-tools-for-machine-learning-projects-ffd9d5af4a20>

## **An organized collection of 94 articles with NLP tips**

by Fabio Chiusano \ NLPlanet \ Aug, 2022 \ Medium

<https://medium.com/nlplanet/an-organized-collection-of-94-articles-with-nlp-tips-5e0af1f3854a>

## **Understanding AlphaZero Neural Network's SuperHuman Chess Ability - MarkTechPost**

<https://www.marktechpost.com/2021/12/16/understanding-alphazero-neural-networks-superhuman-chess-ability/>

## **Want to learn more about Meta Learning?**

What is it? How do companies use it?

Read a blog post on Meta Learning and Neural Architecture Search:

<https://www.thinkautonomous.ai/blog/meta-learning/>

Hope this helps!

## **Video lectures, UC Irvine CS 275**

Video lectures, UC Irvine CS 295 Causal Reasoning spring 2021, by Rina Dechter

<https://t.co/cWKmaRiQuO>

Video lectures, UC Irvine CS 275 Constraint Networks fall 2020, by Rina Dechter

<https://t.co/fzGRFaVtmA>

Video lectures, UCLA CS 262A Learning and Reasoning with Bayesian Networks winter 2017, by Adnan Darwiche

<https://t.co/RZsiWXltnz>

Video lectures, UCLA CS 264A Automated Reasoning fall 2020, by Adnan Darwiche

<https://t.co/FsNBShNWaz>

Video lectures, UC Irvine CS 276 Reasoning in Graphical Models fall 2021, by Rina Dechter

<https://t.co/pNxliIEq7j>

Video lectures, Cornell ORIE 4742 Information Theory and Bayesian Machine Learning spring 2021, by Sid Banarjee

<https://t.co/13cSwxrMtj>

Video lectures, Cornell ORIE 4580 / 5580 Simulation Modeling and Analysis, ORIE 5581 Monte Carlo Simulation fall 2020, by Sid Banarjee

<https://t.co/8MpdOgcZcA>

Video lectures, UCB Info 290 Analyzing Big Data with Twitter fall 2012, by Marti Hearst & Gidal Mishne

<https://t.co/FrSvrWT8Ct>

Video lectures, Cornell ORIE 4741 / 5741 Learning with Big Messy Data fall 2020, by Madeleine Udell

<https://t.co/ena01QHXMV>

Video lectures, CMU 17-445 / 10-645 Software Engineering for AI-Enabled Systems summer 2020, by Christian Kästner



<https://t.co/IMud8z1djh>

Video lectures, CMU 17-445 / 10-645 Software Engineering for AI-Enabled Systems summer 2020, by Christian Kästner

<https://t.co/IMud8z1djh>

Video lectures, CMU 10-418 / 10-618 Machine Learning for Structured Data fall 2019, by Matt Gormley

<https://t.co/u48gmjv8jc>

Video lectures, UC Davis Algebraic and Geometric Ideas in the Theory of Discrete Optimization 2013, by Jesus A. De Loera

<https://t.co/EmyFaBVF MJ>

Video lectures, UC Davis Algebraic and Geometric Ideas in the Theory of Discrete Optimization 2013, by Jesus A. De Loera

<https://t.co/EmyFaBVF MJ>

Often, automated UI testing delays the release cycles because tests take too long to create or fail on small application changes. Join us to learn more about how to align your testing strategy, processes, and tools with your modern development processes to accelerate innovation.

Video lectures, UC Davis MAT 168 Optimization fall 2020, by Matthias Köppe

<https://t.co/FTmld8pkV>

Video lectures, UC Davis MAT 258B Discrete Optimization fall 2021, by Matthias Köppe

<https://t.co/KJ05mPdxNB>

Video lectures, University of Waterloo CO 452/652 Integer Programming Winter 2021, by Ricardo Fukasawa

<https://t.co/8rEGexRvrI>

Video lectures, University of Waterloo CO 450/650 Combinatorial Optimization Fall 2020, by Ricardo Fukasawa

<https://t.co/tmF9SeYZac>

Video lectures, CVPR 2022 Tutorial, Machine Learning for Remote Sensing: Agriculture and Food Security

<https://t.co/IY7Rnld8Zl>

Video lectures, CVPR 2022 Tutorial, Sparse Learning in Neural Networks and Robust Statistical Analysis

<https://t.co/foPi6AuZNN>

Video lectures, CVPR 2022 Tutorial, Multimodal Machine Learning

<https://t.co/Vz16rmYQcw>

Video lectures for EE512A - Advanced Inference in Graphical Models, Fall Quarter, 2014 by Jeff Bilmes

<https://t.co/TyvqlwaCrz>

Video lectures for EE596B - Submodular Functions, Optimization, and Applications to Machine Learning by Jeff Bilmes

<https://t.co/3toMMaaHDH>

Video lectures for EE514a-EE515a Information Theory 1-2 by Jeff Bilmes <https://t.co/w8TmdqnK56>

Video lectures, University of Chicago Computer Science 25300 / 35300 & Statistics 27700 Mathematical Foundations of Machine Learning fall 2019, fall 2021, by Rebecca Willett

<https://t.co/4sHq0zP47h>

Video lectures, University of Wisconsin-Madison CS/ECE 561 - Probability and Information Theory in Machine Learning fall 2020, by Matthew Malley

<https://t.co/cboT8fcoL9>

Video lectures, University of Wisconsin-Madison ECE/CS/ME 532 Matrix Methods in Machine Learning fall

2017, by Laurent Lessard  
<https://t.co/W8WMKLfgy8>

video lectures + lecture slides + Julia notebooks for my optimization course (advanced undergrad level; covers modeling, LP, convex, and integer programming)

<https://t.co/nW18tv41Jj>

## **Towards Responsible Data Practices for Machine Learning in India: Health & Agriculture**

<https://digitalfutureslab.notion.site/Towards-Responsible-Data-Practices-for-Machine-Learning-in-India-Health-Agriculture-f32d56ab8bfc43be8f7fb8d690061db9>

## **Scalable Machine Learning System for Pre-Season Agriculture Yield Forecast**

A Scalable Machine Learning System for Pre-Season Agriculture Yield Forecast. (arXiv:1806.09244v1  
[\[https://t.co/fz6pTNHN2N\]](https://t.co/fz6pTNHN2N))

Yield forecast is essential to agriculture stakeholders and can be obtained with the use of machine learning models and data coming from multiple sources. Most solutions for yield forecast rely on NDVI (Normalized Difference Vegetation Index) data, which is time-consuming to be acquired and processed. To bring scalability for yield forecast, in the present paper researchers describe a system that incorporates satellite-derived precipitation and soil properties datasets, seasonal climate forecasting data from physical models and other sources to produce a pre-season prediction of soybean/maize yield---with no need of NDVI data. This system provides significantly useful results by the exempting the need for high-resolution remote-sensing data and allowing farmers to prepare for adverse climate influence on the crop cycle. In our studies, we forecast the soybean and maize yields for Brazil and USA, which corresponded to 44% of the world's grain production in 2016. Results show the error metrics for soybean and maize yield forecasts are comparable to similar systems that only provide yield forecast information in the first weeks to months of the crop cycle.

## **Model Evaluation Metrics in Machine Learning - Nucleusbox**

<https://www.nucleusbox.com/model-evaluation-metrics-in-machine-learning/>

## **190 Python Projects with Source Code \ by Aman Kharwal \ Medium**

<https://thecleverprogrammer.com/2021/01/14/python-projects-with-source-code/>

## **Dr. Rina Dechter @ UCI**

<https://www.ics.uci.edu/~dechter/courses/ics-295cr/spring-2021/>

## **TorchStudio, Deep Learning Toolkit**

<https://youtu.be/aHmmtUWfnq0>

## Free ML Engineering course

- ◆ Regression & classification
- ◆ Evaluation metrics
- ◆ Tree-based models
- ◆ Deep learning
- ◆ AWS, Serverless, Kubernetes

👉 <https://github.com/alexeygrigorev/mlbookcamp-code/tree/master/course-zoomcamp>

## First Principles of Computer Vision, Columbia University

Really nice lectures on the physical and mathematical foundations of computer vision.

140 videos that you can watch at your pace. Slides are also provided to follow along.

#AI #ComputerVision #MachineLearning #DataScience

Videos: <https://t.co/GqPIV2SzSU>

Slides: <https://t.co/qliRqngnWR>

**Are self-driving cars the future?**

# ARE SELF-DRIVING CARS the Future?



Forbes reports that over 1,700 startups aim to disrupt the automotive industry with autonomous vehicles (AVs).



AVs could free up to 50 minutes a day in commuting time for users.



They could potentially reduce the need for parking space in the US by over 5.7 billion square meters.

## How do Self-Driving Cars Work?

Autonomous cars use Connected Intelligent Transport Systems (C-ITS) to detect their surroundings and to drive without human input.

### Driver Assistance

Car technology automates some tasks without human intervention such as steering or acceleration adjustments, but everything else is fully under human control.

Level 1

### Partial Automation

The car automates some safety actions, such as advanced cruise control, lane-centering and active collision avoidance, but the driver needs to stay alert at the wheel.

Level 2

### Conditional Automation

The car will drive itself for extended periods of time under certain conditions, but the driver must be ready to take control when prompted by the vehicle.

Level 3

### High Automation

The vehicle will drive itself without any human input almost all the time. The human driver may have to take the wheel in certain weather or geographical conditions.

Level 4

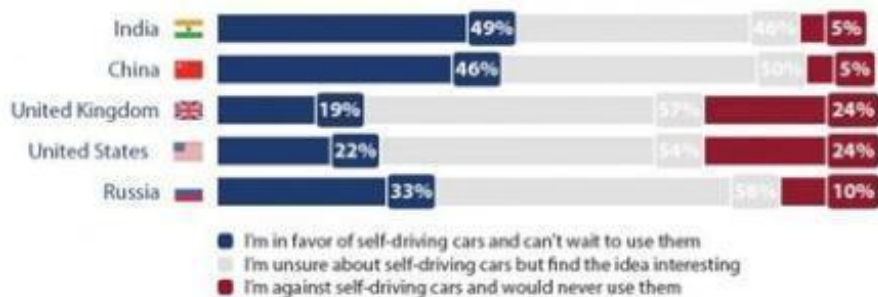
### Full Automation

Level 5

The car is fully automated in all conditions. It doesn't need a steering wheel or pedals and it can travel between locations without a driver or passengers.

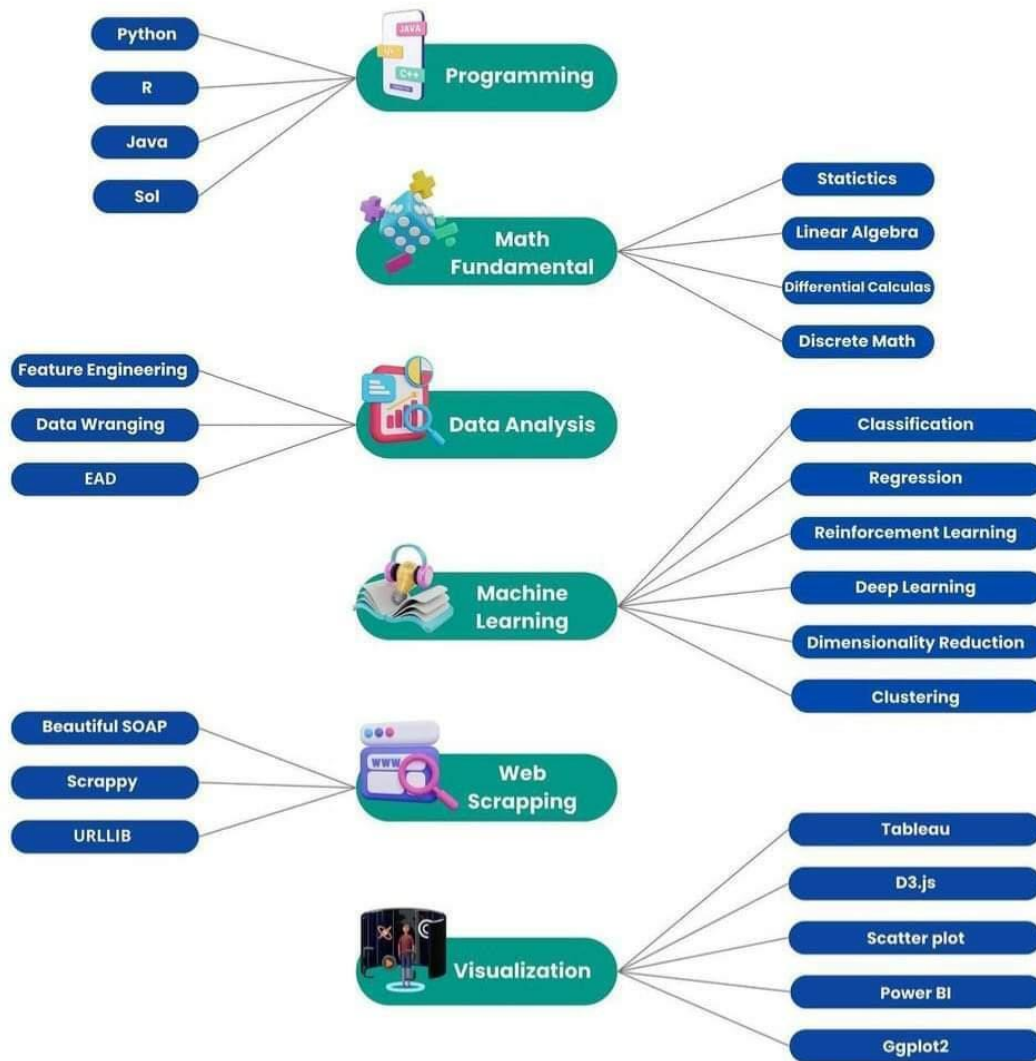
## Consumer Opinion

Only 13% of people say they would never use a self-driving car.





# GUIDE TO BECOME A DATA SCIENTIST



## 7 Advantage of social media marketing



# *7 Advantages of* **Social Media Marketing**

## **1. Brand Awareness**

---

Being present on social media helps your users recognize and recall your brand with ease

## **2. More Website Traffic**

---

Posting optimized content on social media helps drive more traffic to your business website

## **3. Cost Effective**

---

Social media marketing is a cost effective method of marketing and does not require huge investments

## **4. Build Online Community**

---

By sharing content online, you can create a virtual community for your brand or business

## **5. Increase Sales**

---

A stronger presence on social media can have a direct impact on the sales of your business

## **6. Expert Authority**

---

Sharing information on social media will help establish your business as credible and authoritative

## **7. Word of Mouth Marketing**

---

Users who share their experiences with your brand online act as word of mouth advertising

## **Day 1 of 60 Days of Deep Learning with Projects Series**

by Naina Chaturvedi \ Coders Mojo \ Aug, 2022 \ Medium

<https://medium.com/coders-mojo/day-1-of-60-days-of-deep-learning-with-projects-series-4a5caa305cf6>

## **We Automated Attribute Tagging Using Deep Learning Models.(1)**

[https://www.meesho.io/blog/tech-we-automated-attribute-tagging-using-deep-learning-models-data-science-machine-learning?utm\\_source=twitter-paid&utm\\_medium=image-post&twclid=2-5nnlohrwdd35vdrqz22lw7ii7](https://www.meesho.io/blog/tech-we-automated-attribute-tagging-using-deep-learning-models-data-science-machine-learning?utm_source=twitter-paid&utm_medium=image-post&twclid=2-5nnlohrwdd35vdrqz22lw7ii7)

## **9 Distance Measures in Data Science**

<https://towardsdatascience.com/9-distance-measures-in-data-science-918109d069fa>

## **Comparison of the Text Distance Metrics - DataScienceCentral.com**

<https://www.datasciencecentral.com/comparison-of-the-text-distance-metrics/>



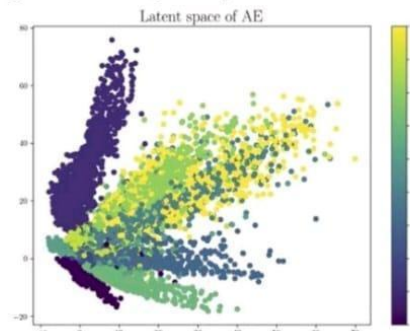
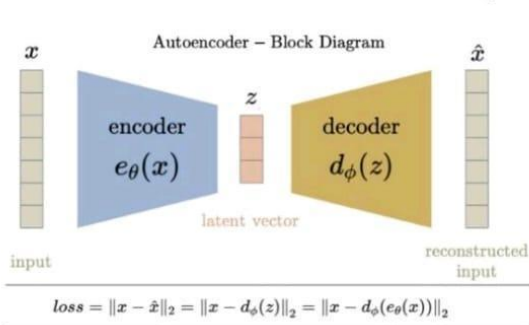
# Cheat Sheet – Autoencoder & Variational Autoencoder

## Context – Data Compression

- Data compression is an essential phase in training a network. The idea is to compress the data so that the same amount of information can be represented by fewer bits.

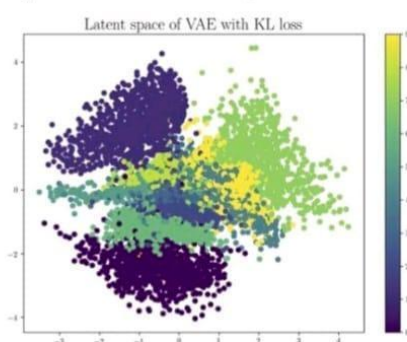
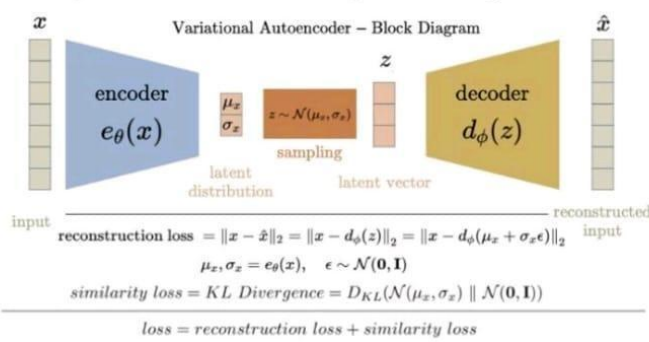
## Auto Encoder (AE)

- Autoencoder is used to learn efficient embeddings of unlabeled data for a given network configuration. It consists of two parts, an encoder, and a decoder.
- The encoder compresses the data from a higher-dimensional space to a lower-dimensional space (also called the latent space), while the decoder converts the latent space back to higher-dimensional space.
- The entire encoder-decoder architecture is collectively trained on the loss function which encourages that the input is reconstructed at the output. Hence the loss function is the mean squared error between the encoder input and the decoder output.
- The latent variable is not regularized. Picking a random latent variable will generate garbage output.
- Latent variable is deterministic values and the space lacks the generative capability



## Variational Auto Encoder (VAE)

- Variational autoencoder addresses the issue of non-regularized latent space in autoencoder and provides the generative capability to the entire space.
- Instead of outputting the vectors in the latent space, the encoder of VAE outputs parameters of a pre-defined distribution in the latent space for every input.
- The VAE then imposes a constraint on this latent distribution forcing it to be a normal distribution.
- The latent variable in the compressed form is mean and variance
- The training loss of VAE is defined as the sum of the reconstruction loss and the similarity loss (the KL divergence between the unit gaussian and decoder output distribution).
- The latent variable is smooth and continuous i.e., random values of latent variable generates meaningful output at the decoder, hence the latent space has generative capabilities.
- The input of the decoder is sampled from a gaussian with mean/variance of the output of encoder.



Source: <https://www.cheatsheets.aqeel-anwar.com> Tutorial: [Click here](#)





<https://towardsdatascience.com/machine-learning-perceptron-implementation-b867016269ec>

## **HiLab-git/SSL4MIS: Semi Supervised Learning for Medical Image Segmentation**

a collection of literature reviews and code implementations.

<https://github.com/HiLab-git/SSL4MIS>

## **How do video live streamings work**

1. How do video live streamings work on YouTube, TikTok live, or Twitter? The technique is called live streaming.

Livestreaming differs from regular streaming because the video content is sent via the internet in real-time, usually with a latency of just a few seconds.

2. The diagram below explains what happens behind the scenes to make this possible.

Step 1: The raw video data is captured by a microphone and camera. The data is sent to the server side.

3. Step 2: The video data is compressed and encoded. For example, the compressing algorithm separates the background and other video elements. After compression, the video is encoded to standards such as H.264.

The size of the video data is much smaller after this step.

4. Step 3: The encoded data is divided into smaller segments, usually seconds in length, so it takes much less time to download or stream.

5. Step 4: The segmented data is sent to the streaming server. The streaming server needs to support different devices and network conditions. This is called 'Adaptive Bitrate Streaming.' This means we need to produce multiple files at different bitrates in steps 2 and 3.

6. Step 5: The livestreaming data is pushed to edge servers supported by CDN (Content Delivery Network.) Millions of viewers can watch the video from an edge server nearby. CDN significantly lowers data transmission latency.

7. Step 6: The viewers' devices decode and decompress the video data and play the video in a video player.

Steps 7 and 8: If the video needs to be stored for replay, the encoded data is sent to a storage server, and viewers can request a replay from it later.

8. Common protocols for live streaming include:

- ◆ RTMP (Real-Time Messaging Protocol): This was originally developed by Macromedia to transmit data between a Flash player and a server. Now it is used for streaming video data over the internet.

9. Note that video conferencing applications like Skype use RTC protocol for low latency

- ◆ HLS (HTTP Live Streaming): It requires the H.264 or H.265 encoding. Apple devices accept only HLS format

- ◆ DASH (Dynamic Adaptive Streaming over HTTP): DASH does not support Apple devices

10. Both HLS and DASH support adaptive bitrate streaming.

Over to you: What are some of the optimizations that can be done in this process? Which type of storage is suitable for video persistence in step 7?

By Alex xu

## Complete Guide to Data Visualization with Python

<https://towardsdatascience.com/complete-guide-to-data-visualization-with-python-2dd74df12b5e>

## SDS 603: Geospatial Data and Unconventional Routes into Data Careers

Podcasts - SuperDataScience \ Machine Learning \ AI \ Data Science Career \ Analytics \ Success

<https://www.superdatascience.com/podcast/geospatial-data-and-unconventional-routes-into-data-careers>

Items mentioned in this podcast:

1. Pachyderm
2. Waze
3. IE Business School
4. SDS 502: Managing Imposter Syndrome
5. S2 Geometry
6. SQL
7. H3
8. Weapons of Math Destruction by Cathy O'Neil
9. Army of None by Paul Scharre

## 7 tools for labeling of Audio, Image, Video and Textual Data

Here are 7 tools that will ease the labeling of Audio, Image, Video and Textual Data for your next Data Science Project:

What is Data Labelling?

It is adding labels to raw data (images, videos, etc.) which gives some meaning so that a machine learning model can learn from it.

And We all know it is a tedious and time taking Process in a Data Science Lifecycle.

1. Labelbox: You can label images, videos, text, PDF documents, medical imagery, and audio data. <https://t.co/TRAKCJ4X8f>
2. Label Studio: This also offers data labeling for text, images, video, audio, time series, multi-domain data types, etc <https://t.co/O1oDsG1kzm>
3. Universal Data Tool: This also allows you to label any type of data in a simple desktop app <https://t.co/JqUgkX8ln8>
4. Prodigy: This is the Annotation tool from Explosion AI used to Annotate text, images, and other data <https://t.co/F32sd2k8RT>
5. CVAT: This is an Open Source labelling tool mostly used to annoate Image and video data. <https://t.co/9AHwASo5gA>
6. Tagtog: Tagtog is a labeling tool for textual data. <https://t.co/BvJON7LCGD>
7. Audino: This is also an Opensource tool that helps you to Annotate Audio data <https://t.co/YcYyzZUT6h>  
#AI #DataScience #MachineLearning #DataLabellingTool #AnnotationTool

## Free ML Engineering course starts on Monday!

- ◆ Linear & logistic regression
- ◆ Model evaluation
- ◆ Model deployment
- ◆ Virtual env, Docker, AWS
- ◆ Tree-based models
- ◆ Neural nets
- ◆ Serverless, Kubernetes

👉 <https://t.co/5vNMU02BFi>

## 12 Free video lectures on Deep Learning

12 Free video lectures on Deep Learning Machine Learning Data Science Natural Language Processing Computer Vision Artificial Intelligence.

1/12 How To Create A Mind By Ray Kurzweil <https://lnkd.in/dz3wyDAY>

2/12 Deep Learning, Self-Taught Learning, and Unsupervised Feature Learning By Andrew Ng <https://lnkd.in/dKNe4mba>

3/12 Recent Developments in Deep Learning By Geoff Hinton <https://lnkd.in/dkCKZ2aF>

4/12 The Unreasonable Effectiveness of Deep Learning by Yann LeCun <https://lnkd.in/dpMgZeVa>

5/12 Deep Learning of Representations by Yoshua bengio <https://lnkd.in/dqmJ2Gye>

6/12 Principles of Hierarchical Temporal Memory by Jeff Hawkins <https://lnkd.in/deNirsPC>

6/12 Principles of Hierarchical Temporal Memory by Jeff Hawkins <https://lnkd.in/deNirsPC>

7/12 Machine Learning Discussion Group - Deep Learning w/ Stanford AI Lab by Adam Coates <https://lnkd.in/dtgBFMU5>

8/12 Making Sense of the World with Deep Learning By Adam Coates <http://vimeo.com/80821560>

9/12 - Demystifying Unsupervised Feature Learning By Adam Coates <https://lnkd.in/dm9FNwgU>

10/10 - Visual Perception with Deep Learning By Yann LeCun <https://lnkd.in/d7AspigF>

11/12 - The Next Generation of Neural Networks By Geoffrey Hinton at GoogleTechTalks <https://lnkd.in/dZ4hBRcJ>

12/12 The wonderful and terrifying implications of computers that can learn By Jeremy Howard at TEDxBrussels <https://lnkd.in/d3t2dfBK>

## What Are The 10 Best AI Consulting Firms

<https://www.forbes.com/sites/bernardmarr/2022/08/17/what-are-the-10-best-ai-consulting-firms/>

## YOLOS

You Only Look at One Sequence: Rethinking Transformer in Vision through Object Detection

Using vanilla ViT that has been pre-trained on medium-sized Imagenet, YOLOS treats object detection as a

sequence-sequence problem. major DETR inspiration, but simply uses transformer encoder and MLP

The best performance on object detection benchmarks is not what YOLOS aims for. Researchers just wanted to adhere to the NLP philosophy (pre-training the task-agnostic transformer for learning general visual representations and fine-tuning on downstream tasks).

It looks like an intriguing route to pre-train for learning general visual representation first and then fine-tune on downstream tasks using the same model afterwards. We are hoping to see pure ViTs that match/outperform current detection-specific models.

YOLOS - You Only Look at One Sequence: Rethinking Transformer in Vision through Object Detection

Paper: <https://t.co/yclM32D2U5>

Code: <https://t.co/bvsiJKVJRH>

## Top Parallel Processing Python Frameworks Data Scientists must know in 2022

by Moez Ali \ Aug, 2022 \ Medium

<https://moez-62905.medium.com/top-parallel-processing-python-frameworks-data-scientists-must-know-in-2022-167212fefa76>

## Google AI Introduces a New Approach to Video-Text Learning Called Iterative Co-Tokenization

Which can Efficiently Fuse Spatial, Temporal, and Language Information for VideoQA - MarkTechPost

<https://www.marktechpost.com/2022/08/21/google-ai-introduces-a-new-approach-to-video-text-learning-called-iterative-co-tokenization-which-can-efficiently-fuse-spatial-temporal-and-language-information-for-videoqa/>

## Meet Sipeed's TinyMaix

An Open-Source Lightweight Machine Learning Library For Microcontrollers - MarkTechPost

<https://www.marktechpost.com/2022/08/21/meet-sipeeds-tinymaix-an-open-source-lightweight-machine-learning-library-for-microcontrollers/>

## AI/CV/MLOps Tools

MLOps Tools and Steps For Computer Vision:

Data and Feature Stage:

Data creation:

1. Labelling: <https://lnkd.in/ddZef4BQ>
2. Computer Vision Annotation Tools: <https://lnkd.in/diiKwFsF>

3. Label Studio: <https://lnkd.in/dhuTCwAt>
4. Visual Object Tagging Tool: <https://lnkd.in/dzUejr6B>

Data management:

1. Neptune: <https://lnkd.in/dzmUWytP>
2. DVC: <https://dvc.org/>
3. RoboFlow: <https://roboflow.com/>
4. Dataiku: <https://www.dataiku.com/>

Data verification:

1. Scale Nucleus: <https://scale.com/>
2. great\_expectation: <https://lnkd.in/dHrVVEba>
3. Soda Data Observability: <https://www.soda.io/>

Data processing:

1. Apache Storm: <https://storm.apache.org/>
2. Dagster: <https://dagster.io/>
3. Apache Spark: <https://lnkd.in/d9gNqPTg>

Feature store

1. butterfree: <https://lnkd.in/dauvwuAE>
2. Feast: <https://feast.dev/>
3. SageMaker: <https://lnkd.in/djgm5PrU>
4. ByteHub: <https://www.bytehub.ai/>

Model Development Stage:

Model Registry:

1. MLFlow: <https://lnkd.in/deq7UmZq>
2. Neptune.ai: <https://neptune.ai/>
3. SageMaker Model Registry:  
[https://lnkd.in/dkV\\_y3mr](https://lnkd.in/dkV_y3mr)

Model Training:



1. Vision AI: <https://lnkd.in/df7UtAWW>
2. DINO: [https://lnkd.in/d5Ay\\_PrY](https://lnkd.in/d5Ay_PrY) 3. Amazon Rekognition: [https://lnkd.in/d-A6n\\_X9](https://lnkd.in/d-A6n_X9)

Metadata management:

1. [Neptune.ai](https://neptune.ai)
2. SiaSearch: <https://www.siasearch.io/>
3. Tensorflow's ML MetaData:  
<https://lnkd.in/d6pWWUUJ>

Hyperparameter tuning:

1. HyperOpt: <https://lnkd.in/daDshC2h>
2. Dask: <https://lnkd.in/d5wWVy9T2> 3. Optuna: <https://optuna.org/>

Version Control:

1. [Github.com](https://github.com)
2. [gitlab.com](https://gitlab.com)

## 5 famous transformer architectures

1. #GoogleAI Switch Transformer
2. #GoogleAI BERT
3. #OpenAI GPT-3
4. #OpenAI ImageGPT
5. #DeepMind compressive transformer

Read more about each of them in comprehensive overviews: <https://t.co/Mf7BUJ7iml>

## Free courses to Learn Deep Learning

DeepMind, MIT, Stanford & Cornell Tech are offering comprehensive courses in Machine Learning for FREE!!

Here's a list of 8 YouTube Playlists you must save

### 1) DeepMind x UCL

Deep Learning Lecture series by DeepMind & University college London!

Check this out  
<https://t.co/GmDAFYQ2p0>

### 2) Stanford CS-229

Machine Learning full course by Andrew Ng.

Check this out

<https://t.co/QfrDvzNwKC>

③ MIT \ Intro to deep Learning

Check this out

<https://t.co/m4zciSfOnA>

④ Stanford CS231n

Started by Andrej Karpathy, arguably the best resource to get started with computer vision.

Check this out

<https://t.co/jgVAVheHtW>

⑤ Stanford \ Statistical Learning

Based on the book Intro to statistical learning (ISL).

A great lecture series

Check this out

<https://t.co/y2WUfmpwbA>

⑥ Deep Mind \ Reinforcement Learning course

One of the most trusted and comprehensive course on RL.

Check this out

<https://t.co/mqmOhtHPuD>

⑦ Cornell Tech \ Applied Machine Learning

Check this out

<https://t.co/Sz8JuzzjZN>

⑧ CMU \ Deep Learning - 2022

A comprehensive course on Deep Learning.

Both NLP & CV are covered 🚀

What you will learn:

- CNN
- RNN
- LSTM
- Graph Neural Networks
- Transformers
- Auto-encoders

Check this out

<https://t.co/75MObAmkOx>

## 5 interesting #Python Projects with Source Code

Here are 5 #DataScience and #MachineLearning Projects that you can add to your portfolio :

👉 1. WhatsApp Data Analysis using Python

🌐 <https://t.co/hkgRFd9H3A>

2. Extract Text from any website using Python

🌐 <https://t.co/yDWaw5Tyiv>

3. Extract Text from PDF file using Python

🌐 <https://t.co/T6AI5zA5kT>

4. Detect Emotion, Sentiment and Trends using AI

🌐 <https://t.co/Fpu5ZiPiXJ>

5. Generate Subtitles for YouTube Video

🌐 <https://t.co/7oDxFqPeCb>

## UnMICST: Deep learning with real augmentation for robust segmentation

UnMICST: Deep learning with real augmentation for robust segmentation of highly multiplexed images of human tissues

<https://buff.ly/3Vb4NPb>

## The First Workshop on Pre-training ICML22

The First Workshop on Pre-training: Perspectives, Pitfalls, Paths Forward - ICML 22' Videos

Through pre-training, we have been able to solve many downstream tasks using (large) pre-trained models.

Found a great series of videos of pre-training workshop.

<https://t.co/LARuhb4y3x>

All recordings of ICML 2022 conference can be found below on SlidesLive too.

<https://t.co/dmgugQFOgx>

## Machine Learning Cheatsheets

1. Supervised Learning cheatsheet

By Afshine Amidi and Shervine Amidi

Introduction to Supervised Learning

<https://stanford.edu/~shervine/teaching/cs-229/cheatsheet-supervised-learning>

2. Unsupervised Learning cheatsheet

By Afshine Amidi and Shervine Amidi

Introduction to Unsupervised Learning

<https://stanford.edu/~shervine/teaching/cs-229/cheatsheet-unsupervised-learning>

3. Deep Learning cheatsheet

By Afshine Amidi and Shervine Amidi

Neural Networks

Neural networks are a class of models that are built with layers. Commonly used types of neural networks include convolutional and recurrent neural networks.

<https://stanford.edu/~shervine/teaching/cs-229/cheatsheet-deep-learning>

4. Machine Learning tips and tricks cheatsheet

By Afshine Amidi and Shervine Amidi

<https://stanford.edu/~shervine/teaching/cs-229/cheatsheet-machine-learning-tips-and-tricks>

## **Computational Linear Algebra**

Computational Linear Algebra Videos Playlist

<https://www.youtube.com/playlist?list=PLtmWHNX-guklc92m1K0P6bIOnZb-mg0hY>

## **Guided-Generative Network**

Guided-Generative Network: A New Robust Deep Learning Architecture for Noise Characterization in Monte-Carlo Rendering

<https://buff.ly/3VQLuva>

## **AlphaCode paper**

DeepMind published the original AlphaCode paper!

It can generate code that requires deep reasoning on the problem.

Researchers also open-sourced the CodeContests dataset that was used to train AlphaCode.

<https://t.co/QEokoDMeyN>

## **How Do We Design and Develop Human-Centered AI?**

<https://hai.stanford.edu/news/how-do-we-design-and-develop-human-centered-ai>

## **Diplomacy: The AI Benchmark that Gets Us Closer to the Turing Test**

<https://thesequence.substack.com/p/the-ai-benchmark-that-gets-us-closer>

## **ChatGPT vs LaMDA**

<https://analyticsindiamag.com/chatgpt-vs-lamda/>

## **Optimize Sales Funnel and Increase Revenue with Machine Learning**

<https://moez-62905.medium.com/optimize-sales-funnel-and-increase-revenue-with-machine-learning-in-pyt-hon-9af57c1e5776>

## **Create Observable and Reproducible Notebooks with Hex**

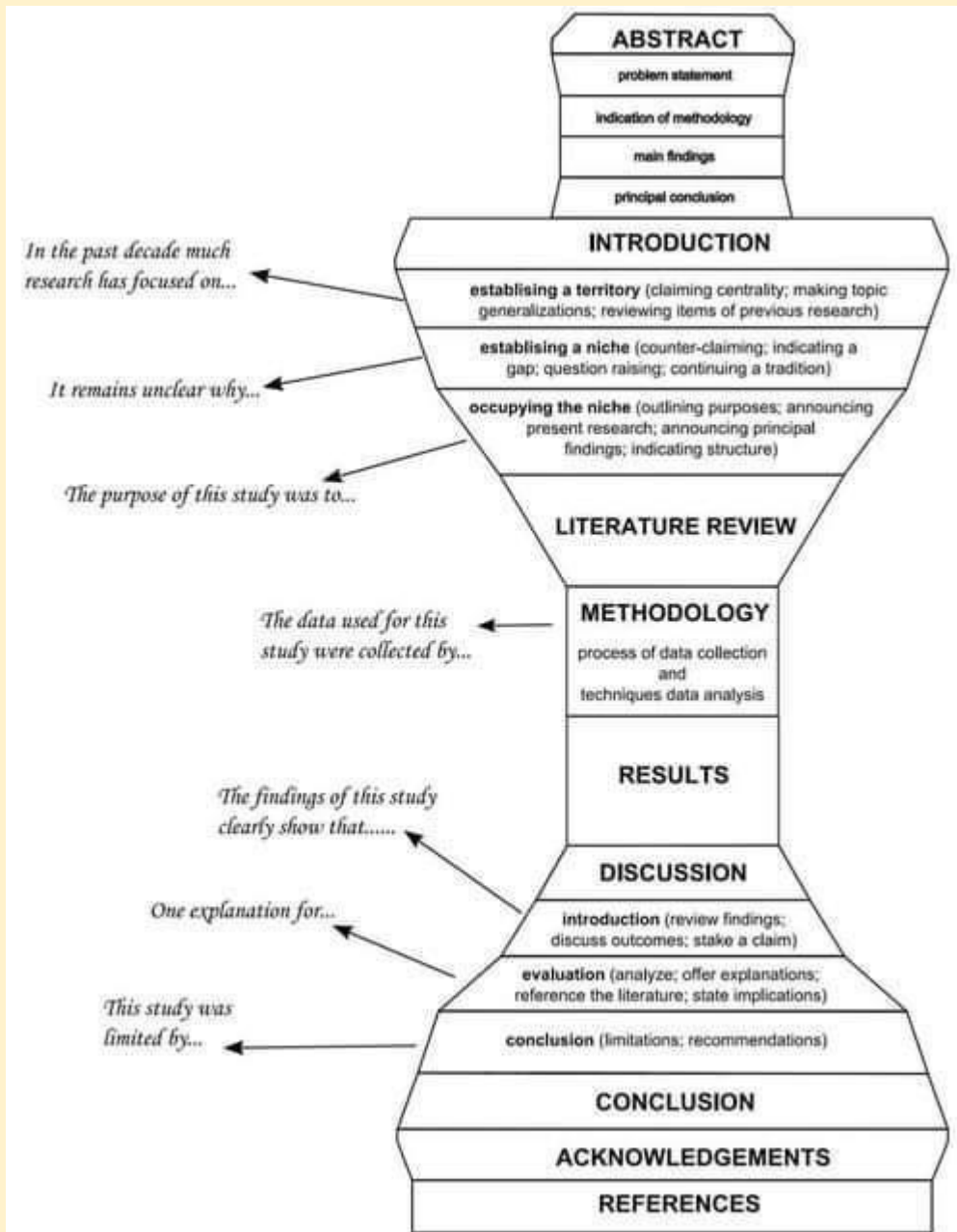
<https://towardsdatascience.com/create-observable-and-reproducible-notebooks-with-hex-460e75818a09>

## **Stanford Graph Learning Workshop 2022**

<http://snap.stanford.edu/graphlearning-workshop-2022/>

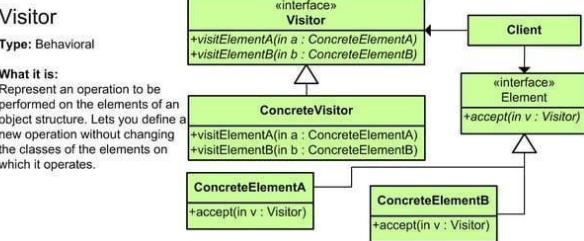
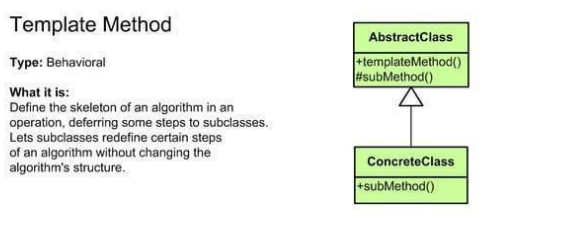
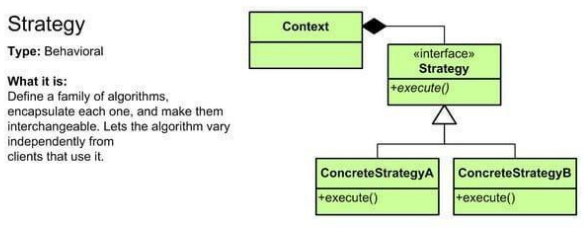
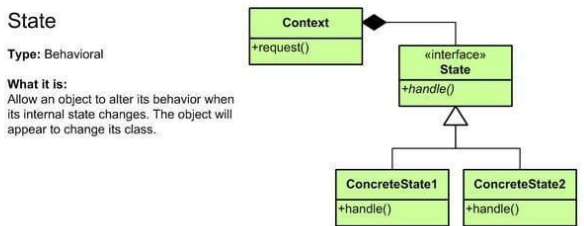
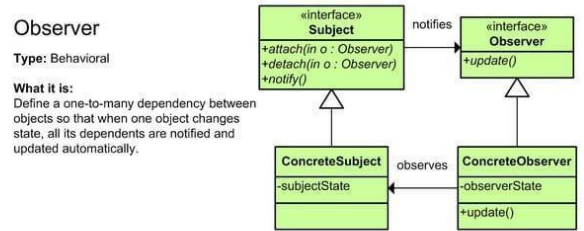
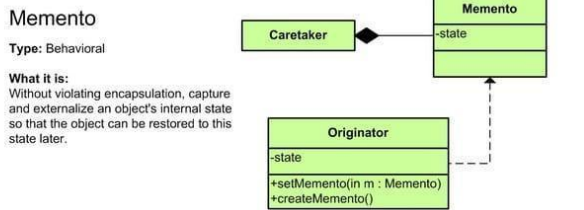
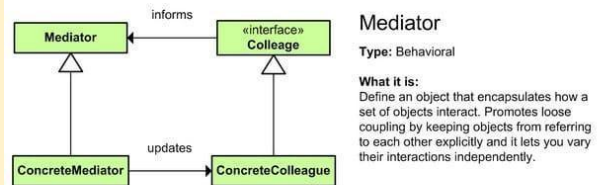
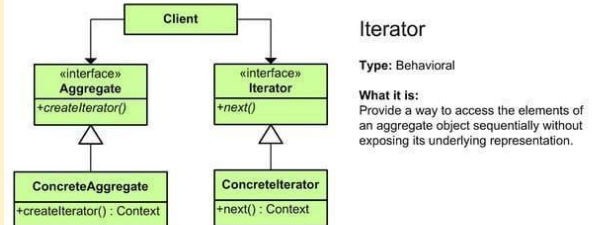
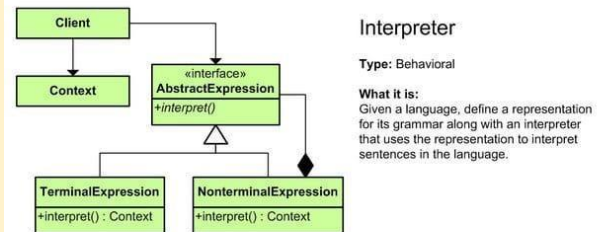
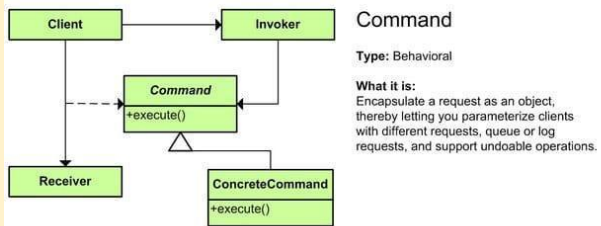
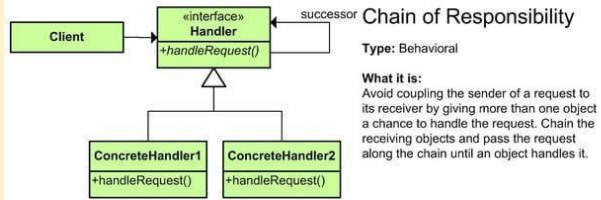
## **Components of Research Paper**





## Software Design Patterns

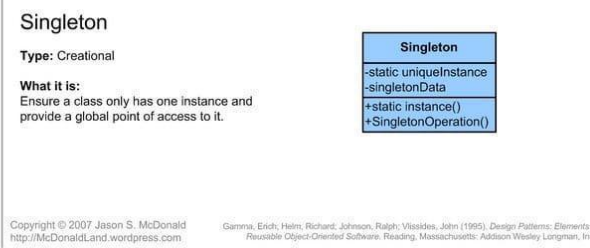
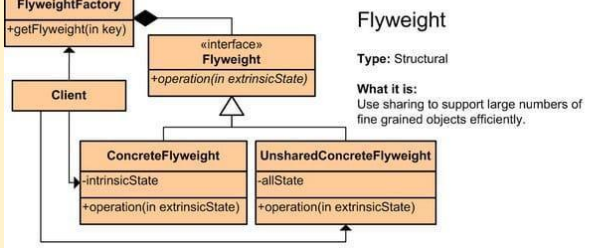
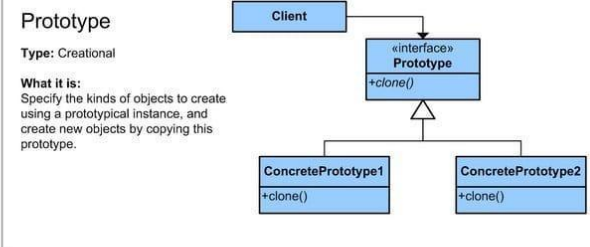
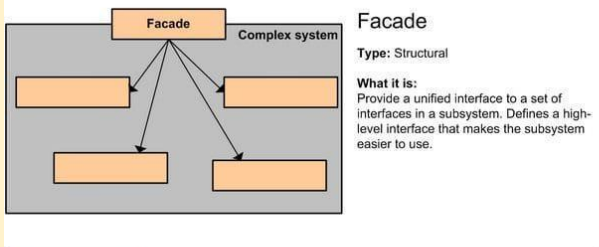
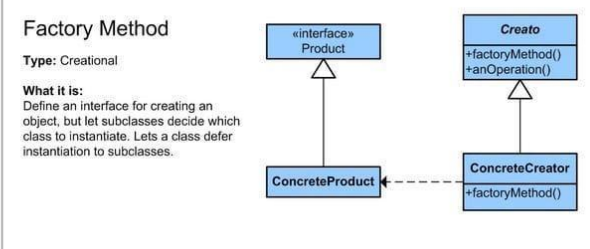
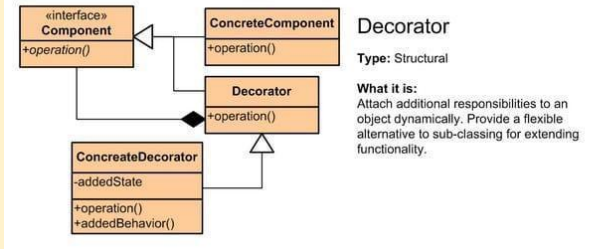
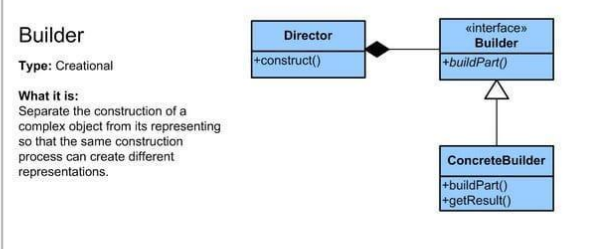
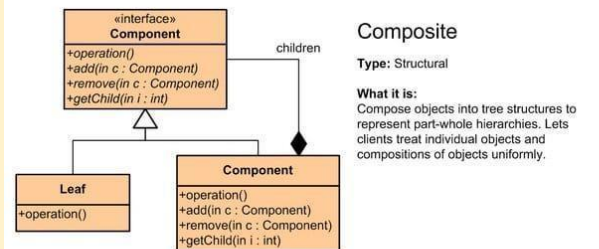
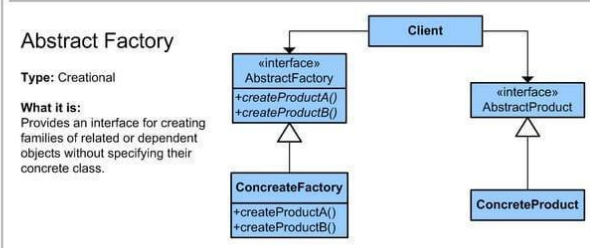
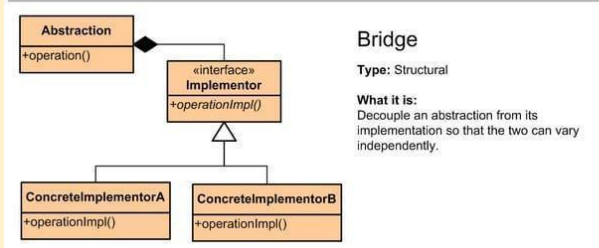
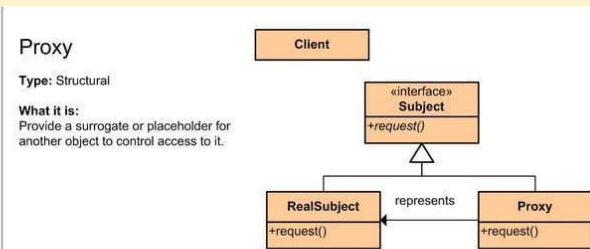
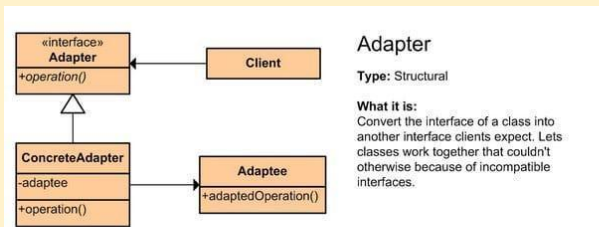
|   |                         |   |                |   |                 |
|---|-------------------------|---|----------------|---|-----------------|
| C | Abstract Factory        | S | Facade         | S | Proxy           |
| S | Adapter                 | C | Factory Method | B | Observer        |
| S | Bridge                  | S | Flyweight      | C | Singleton       |
| C | Builder                 | B | Interpreter    | B | State           |
| B | Chain of Responsibility | B | Iterator       | B | Strategy        |
| B | Command                 | B | Mediator       | B | Template Method |
| S | Composite               | B | Memento        | B | Visitor         |
| S | Decorator               | C | Prototype      |   |                 |



Copyright © 2007 Jason S. McDonald  
http://McDonaldJL.wordpress.com

Gamma, Erich; Helm, Richard; Johnson, Ralph; Vlissides, John (1995). Design Patterns: Elements of Reusable Object-Oriented Software. Reading, Massachusetts: Addison Wesley Longman, Inc.

## Software Design Patterns2



Copyright © 2007 Jason S. McDonald <http://McDonaldL.and.wordpress.com> Gamma, Erki, Helm, Richard, Johnson, Ralph, Vlissides, John (1995). Design Patterns: Elements of Reusable Object-Oriented Software. Reading, Massachusetts: Addison Wesley Longman, Inc.

## How to Organize your Research Paper

# A VISUAL GUIDE TO ORGANIZING YOUR PAPERS

## Suggestions for Introduction:

- Define related terms
- Discuss related history
- Use an anecdote
- Use a provocative quotation
- Ask a rhetorical question
- Relate a dramatic episode
- Use wit, humor (if appropriate)

- Does your title capture your reader's interest?
- Does it hint at the direction of your subject matter?

## TITLE

## INTRODUCTION

A BROAD CONTEXT  
\*SETS THE STAGE\*

NARROW CONTEXT  
State Thesis

## SUPPORTING PARAGRAPHS

- Write meaningful and sufficiently broad topic sentences.
- Support your topic sentences/paragraphs with facts, relevant information (evidence, etc.).
- Relate each sentence in a paragraph to the topic sentence to achieve unity.
- Choose transitional words and expressions carefully to achieve logical coherence.
- Vary your sentence pattern. ✓
- Choose precise diction and active verbs.
- Use passive voice sparingly.
- Use vivid and lively language, creating fresh comparisons when appropriate.
- In argumentation, present weakest argument first.

## CONCLUSION

NARROW CONTEXT  
Restate Thesis  
Restate Major Points

A BROAD CONTEXT  
\*wraps up\* your discussion.

You as a writer have concluded  
WHAT from your supporting  
paragraphs?

Bring your central idea to a  
logical end.

## Suggestions for Conclusion

- Suggest a larger pattern
- Use a "clinching" quotation
- Draw inferences from your presentation of ideas
- Look forward and make predictions
- Suggest solutions, resolutions

Quoted by Joyce M. Miller  
English Department, CCCO

\*Kupala\* from adapted from  
Pinto's Computer System

- REMINDEES:
- A thesis sets forth the main idea of your paper; a topic sentence sets forth the main idea of your paragraph.
  - An effective thesis has an "edge" to it, one that arouses your reader's interest.
  - To put it another way, a thesis statement sets up expectations that must be fulfilled.

## Filter data using multiple columns - Microsoft Power BI Community

<https://community.powerbi.com/t5/Desktop/Filter-data-using-multiple-columns/m-p/327714#M146290>

## Edge#194: Masterful AI, the Training Platform for Automated Computer Vision

<https://thesequence.substack.com/p/edge194>



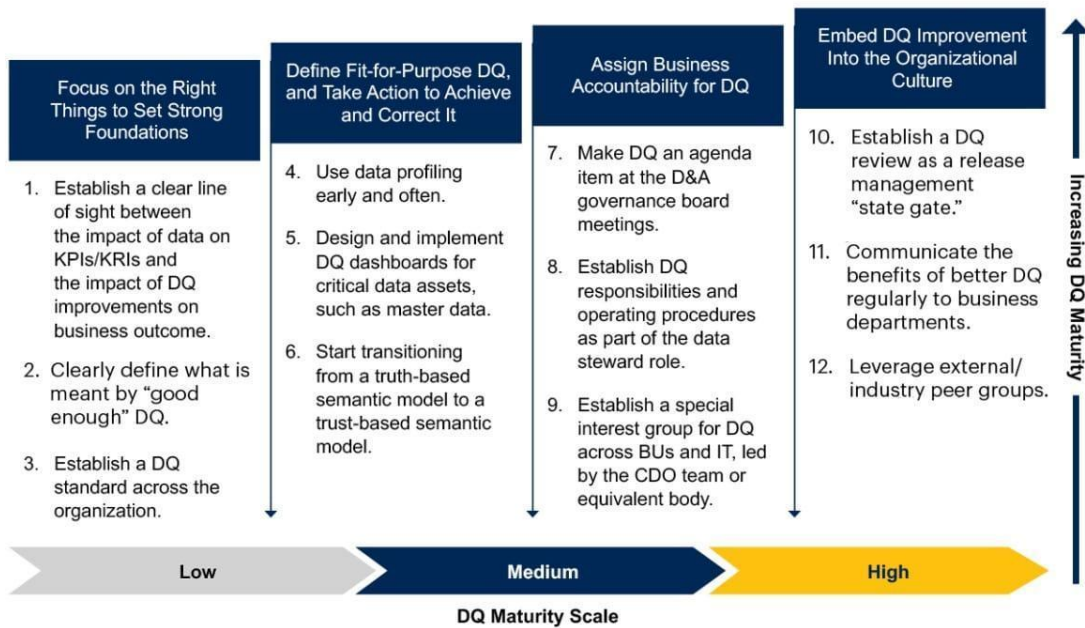
## ML Algorithms and Training/Prediction/Space Complexity

| Algorithm]              | Training (time complexity)         | Prediction (time complexity) | Auxiliary space complexity | Comment                                                                                                                                                                                                                                                         |
|-------------------------|------------------------------------|------------------------------|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Linear regression       | $O(nm^2+m^3)$ [1]                  | $O(m)$                       | $O(m)$                     | Low run time and space complexity!<br>Suitable for low latency problems                                                                                                                                                                                         |
| Logistic regression     | $O(nm)$ [2]                        | $O(m)$                       | $O(m)$                     | Low run time and space complexity!<br>Suitable for low latency problems                                                                                                                                                                                         |
| Naive Bayesian          | $O(n*m)$                           | $O(c*m)$                     | $O(c*m)$                   | Works well with high dimensional data. Mainly used in NLP problems with large dimensionality. Also used to compare with more complex algorithms used for NLP.                                                                                                   |
| Decision Trees          | $O(n\log(n)d)$                     | $O(d)$                       | $O(p)$                     |                                                                                                                                                                                                                                                                 |
| Gradient Boosting       | $O(n\log(n)dr)$                    | $O(dr)$                      | $O(pr+br)$                 | br could be dropped as $br \ll pr$                                                                                                                                                                                                                              |
| Random Forest           | $O(n\log(n)dn_{tree})$             | $O(dn_{tree})$               | $O(pn_{tree})$             |                                                                                                                                                                                                                                                                 |
| Support Vector Machines | $O(n^2m+n^3)$                      | $O(mn_{sv})$                 | $O(n_{sv})$                | High training time but low space complexity. Suitable for low latency problems                                                                                                                                                                                  |
| K Means                 | $O(nmki)$                          | $O(mk)$                      | $O(nm+km)$                 |                                                                                                                                                                                                                                                                 |
| KNN                     | $O(1)$                             | $O(nm)$                      | $O(nm)$                    |                                                                                                                                                                                                                                                                 |
| Hierarchical clustering | $O(n^3) \Rightarrow O(n^2\log(n))$ | $O(mn)$                      | $O(n^2)$                   | High training time, high runtime, and high space complexities. Not suitable for low latency.                                                                                                                                                                    |
| DBSCAN                  | $O(n^2) \Rightarrow O(n\log(n))$   | $O(mn)$                      | $O(n)$                     | Better than hierarchical clustering in terms of complexities                                                                                                                                                                                                    |
| PCA                     | $O(nm \times \min(n, m) + m^3)$    | $O(lm)$                      | $O(lm)$                    | The complexity of covariance matrix computation is $O(nm \times \min(n, m))$ . Its eigenvalue decomposition is $O(m^3)$ . $O(lm)$ is for the transformation matrix                                                                                              |
| t-SNE                   | $O(n^2)$                           | $O(nq)$                      | $O(n^2)$                   | t-SNE involves significant amount of computations and hence the algorithm takes a lot of time and space for the computation. t-SNE has a quadratic time and space complexity in the number of data points. <b>t-SNE requires <math>O(3n^2)</math> of memory</b> |

## 12 Actions to improve Data Quality



# 12 Actions to Improve Your DQ



[gartner.com](https://www.gartner.com)

Source: Gartner  
© 2021 Gartner, Inc. All rights reserved. PR\_1363600

**Gartner**

**Generative AI Application Landscape**

# The Generative AI Application Landscape v2



A work in progress



MIT Deep Learning 6.S191

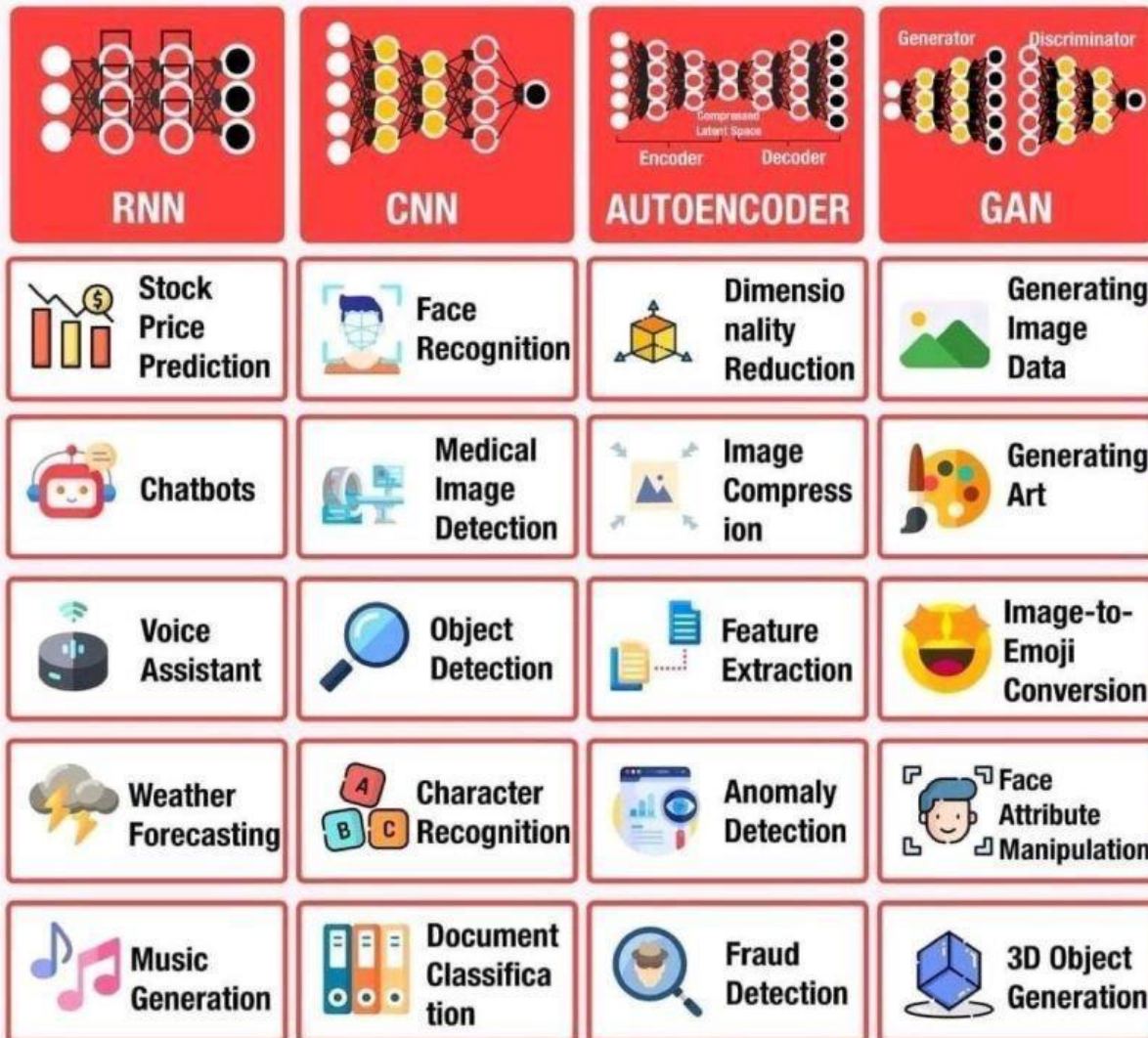
<http://introtodeeplearning.com/>

Welcome to Shapash's documentation ! — Shapash 2.2.0 documentation

<https://shapash.readthedocs.io/en/latest/index.html#>

## Deep Learning Usecases

# DEEP LEARNING USE CASES



## Research Paper Flow



# Academic Research Posters: Strategies and Principles For Good Design

Albert Einstein, Department of Physics and Astronomy, University of Sheffield

## Introduction

Academic research is often presented visually in the form of a research poster. This format of dissemination is a particularly effective way to get complex ideas across in a simple and accessible way and is commonly used as a method of networking for researchers at conferences and events. It can be a great way to present your research to a wider audience and to get it noticed by relevant specialists, policy-makers or the general public.

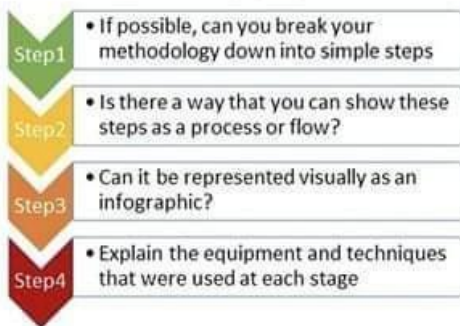
## Aims and Objectives

Can you break down the aims and objectives of your project into some simple statements? If so, try presenting these as bullet points to make it easy for your reader to find them:

- Produce a high quality academic poster
- Present my findings in an accessible and visually engaging way
- Raise awareness of my research among a wider audience

## Methodology

A brief objective description of the experimental approach, including equipment used and the protocol followed. Your methods should include techniques used both to generate and to analyse the data (i.e. which statistical methods were employed).



## Bibliography:

Lippman, Doris Troth et al. "Designing a research poster with impact." *Western Journal of Nursing Research* 11.4 (1989): 477-485.  
Rowe, Nicholas et al. "What impact do posters have on academic knowledge transfer?" *BMC medical education* 9.1 (2009): 71.

## Results/Findings

What are your key findings? Can you support these with graphical representations of your data? Think about simplicity and accessibility of the information and adopt a 'less is more' approach where possible.

Fig. 1: Delegate responses to the prompt: "Academic Conference Posters are commonly used with my professional field"

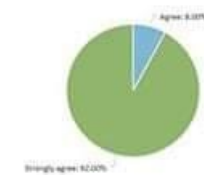
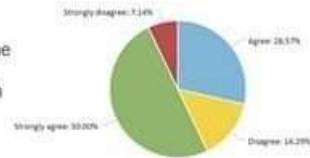


Fig. 2: Delegate responses to the prompt: "Catching viewers attention is reliant upon imagery and composition"

A lot of viewers of your poster will look at the images first (see Fig 2. above), so make sure they tell a story in their own right and are clearly annotated and labelled.

## Analysis/Discussion

What does your data mean? How does what you have presented fit in with the existing literature? Do your findings support other published research, or do they challenge it? If so, why, or why not? You may also wish to include the following sections:

- **Limitations: were there any problems or issues with your research project?**
- **Key themes or findings from your research summarised as succinctly as possible**

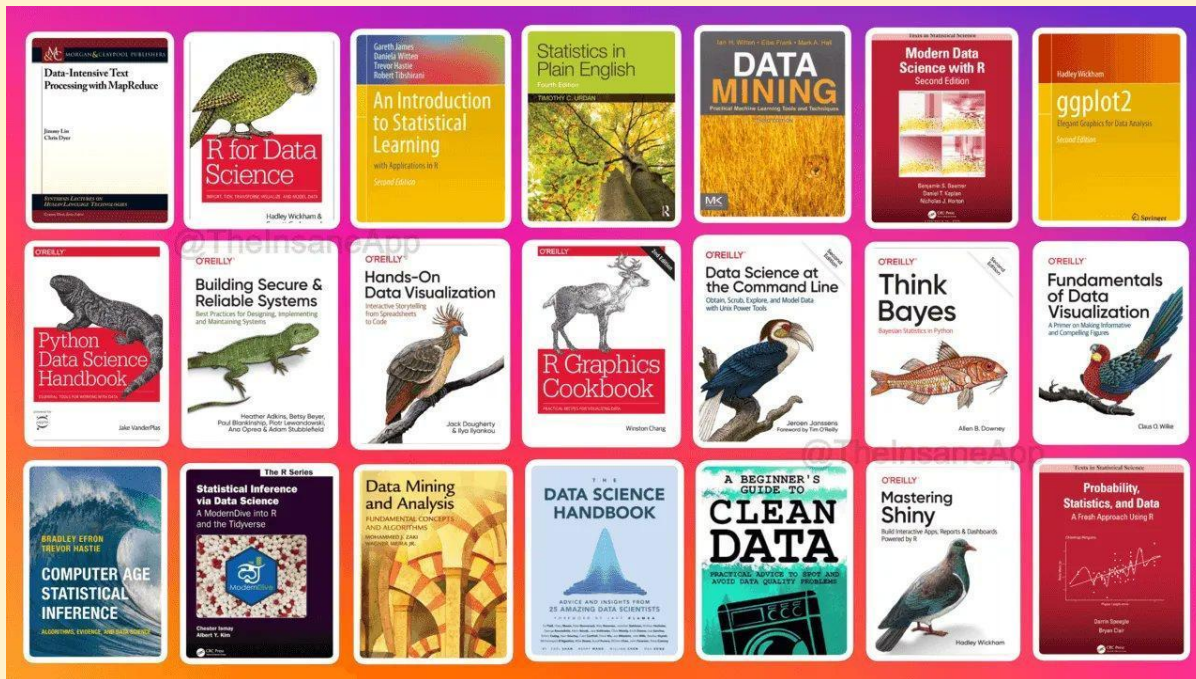
## Conclusion

What are the main take-home messages for your audience? Are there any recommendations, implications for practice, or is further research required to build a complete picture? If so, what are the next steps for you and/or other researchers in the field?

## Acknowledgements:

Sir Isaac Newton, Professor of Mathematics  
University of Cambridge  
Brian Cox, School of Physics and Astronomy,  
University of Manchester





## MLOps for production-grade ML

#MLOps for production-grade Machine Learning.

Here are five FREE courses to learn MLOps and start deploying your ML models in production.

### 1. MLOps Specialization by DeepLearning AI

In this course, you will learn how to conceptualize, build, and maintain integrated systems that continuously operate in production.

You can apply for financial aid to take this course for FREE.

👉 <https://t.co/VQ3f6d3Gwx>

### 2. Made with ML

One of the best places to learn about all things MLOPs.

With a community of 30k+ developers, you will learn how to combine ML with software engineering to develop, deploy & maintain production ML applications.

👉 <https://t.co/lbHoZxYzWe>

### 3. MLOps Fundamentals by Google

In this course, you will learn about MLOps tools and best practices for deploying, evaluating, and monitoring production ML systems on Google Cloud.

You can apply for financial aid to take this course for FREE.

👉 <https://t.co/JTU18V1rRZ>

### 4. MLOps with GitHub

A collection of resources on how to facilitate Machine Learning Ops with GitHub. Learn how to use GitHub for automation, collaboration and reproducibility in your machine learning workflows.



👉 <https://t.co/nGnuHUKVwi>

## 5. MLOps Guide by #INNOQ

Collection of articles that will help you learn end-to-end machine learning development process to design, build and manage reproducible, testable, and evolvable ML-powered software.

👉 <https://ml-ops.org/>

## Westlake University Researchers Propose 'SimVP,'

An Artificial Intelligence AI-Based Video Prediction Model that is Completely Built Upon CNN and Trained by MSE Loss in an End-to-End Fashion - MarkTechPost

<https://www.marktechpost.com/2022/11/18/westlake-university-researchers-propose-simvp-an-artificial-intelligence-ai-based-video-prediction-model-that-is-completely-built-upon-cnn-and-trained-by-mse-loss-in-an-end-to-end-fashion/>

## Drexel University Introduces TorchFL:

A Python Library for Prototyping and Experimentation of Federated Learning Using State-of-the-Art Models and Datasets - MarkTechPost

<https://www.marktechpost.com/2022/11/18/drexel-university-introduces-torchfl-a-python-library-for-prototyping-and-experimentation-of-federated-learning-using-state-of-the-art-models-and-datasets/>

## Optimize Sales Funnel and Increase Revenue with Machine Learning in Python

<https://moez-62905.medium.com/optimize-sales-funnel-and-increase-revenue-with-machine-learning-in-python-9af57c1e5776>

## Stanford Engineering Everywhere \ CS223A - Introduction to Robotics

<https://see.stanford.edu/course/cs223a>

## Remote job in AI/ML

How to get your first remote job in AI/ML that pays in USD?

1. RemoteOK

Find a remote job and work from anywhere.

👉 <https://t.co/KR4G1n6XoU>

2. NoDesk

Find remote jobs and connect with companies that offer you the freedom to work remotely from home or places around the world.

👉 <https://t.co/Cj171HntWe>

3. JustRemote

Discover fully and partially remote jobs from the best remote working companies.

👉 <https://t.co/7pASpurcqY>

#### 4. We Work Remotely

We Work Remotely is the largest remote work community in the world. With over 4.5M visitors, WWR is the number one destination to find and list incredible remote jobs.

👉 <https://t.co/cHSC7befl5>

#### 5. Remotive

Find your dream remote job without the hassle!

👉 <https://t.co/XCs8XH4FO4>

#### 6. Himalayas

Experience the remote ecosystem, and find a remote job you love!

👉 <https://t.co/RPC7TgxnvP>

#### 7. Remote Co

All things remote work:

- Learn about remote work
- Find Remote Jobs
- Post Remote Jobs

👉 <https://t.co/kt2eX68eqy>

### **Algorithmic concepts \ Super Study Guide**

<https://superstudy.guide/algorithms-data-structures/foundations/algorithmic-concepts>

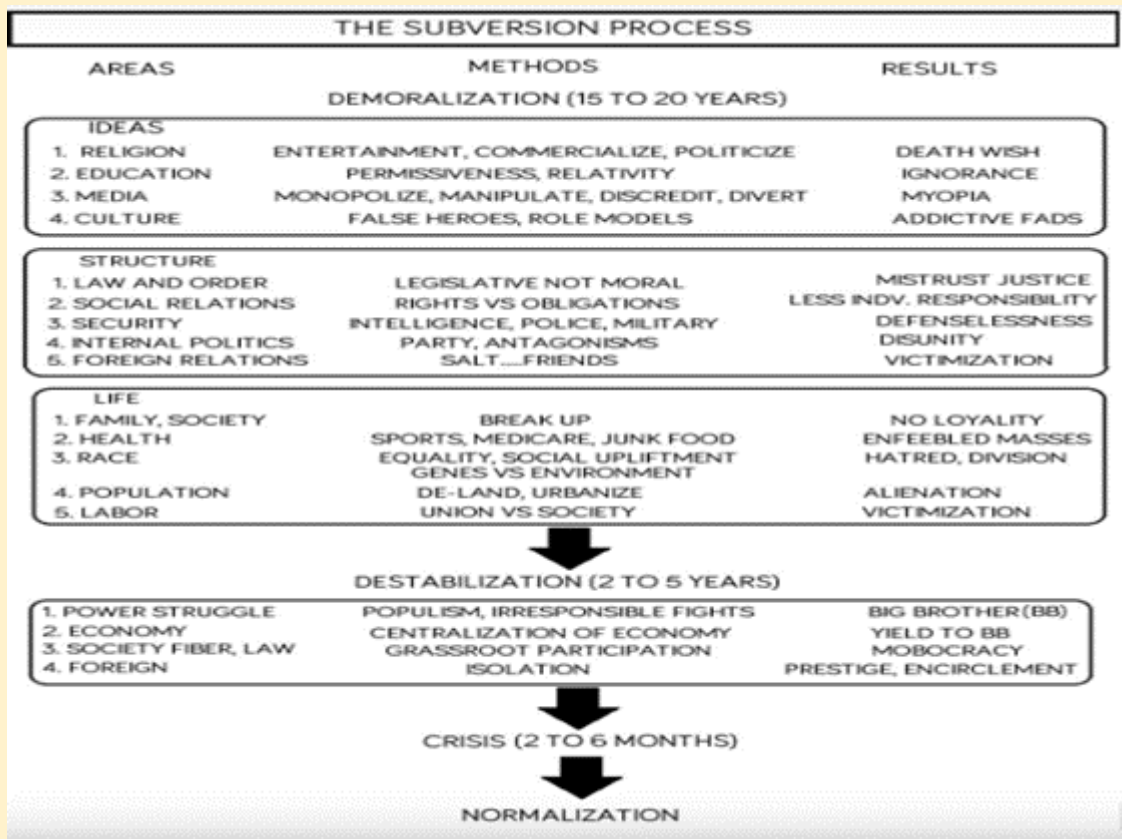
### **Comprehensive Guide to Zero-Shot and K-Shot Learning**

<https://www.datature.io/blog/zero-shot-and-k-shot-learning>

### **Learn Docker in 12 Minutes**

<https://youtu.be/YFI2mCHdv24>

## The subversion process



## 'MEND' For Fast Model Editing At Scale

Researchers At Stanford Have Developed An Artificial Intelligence (AI) Approach Called 'MEND' For Fast Model Editing At Scale - MarkTechPost

<https://www.marktechpost.com/2022/11/09/researchers-at-stanford-have-developed-an-artificial-intelligence-ai-approach-called-mend-for-fast-model-editing-at-scale/>

## Data Structures and Algorithms

Data Structures and Algorithms are building blocks of Programming,

Here are 5 Resources to Learn DSA in Python for Free :

1 - DSA in Python by Jovian ML :

<https://t.co/kPS8hUFh5E>

2 - Intro to Data Structures and Algorithms :

<https://t.co/7QDV1sptLM>

3 - DSA in Python for Beginners - 2022 \ Great Learning :

<https://t.co/688LjYtnHk>

4 - Introduction To Data Structures :

<https://t.co/IOtehX7rmK>

5 - Problem-solving with Data structures and Algorithms using Python :

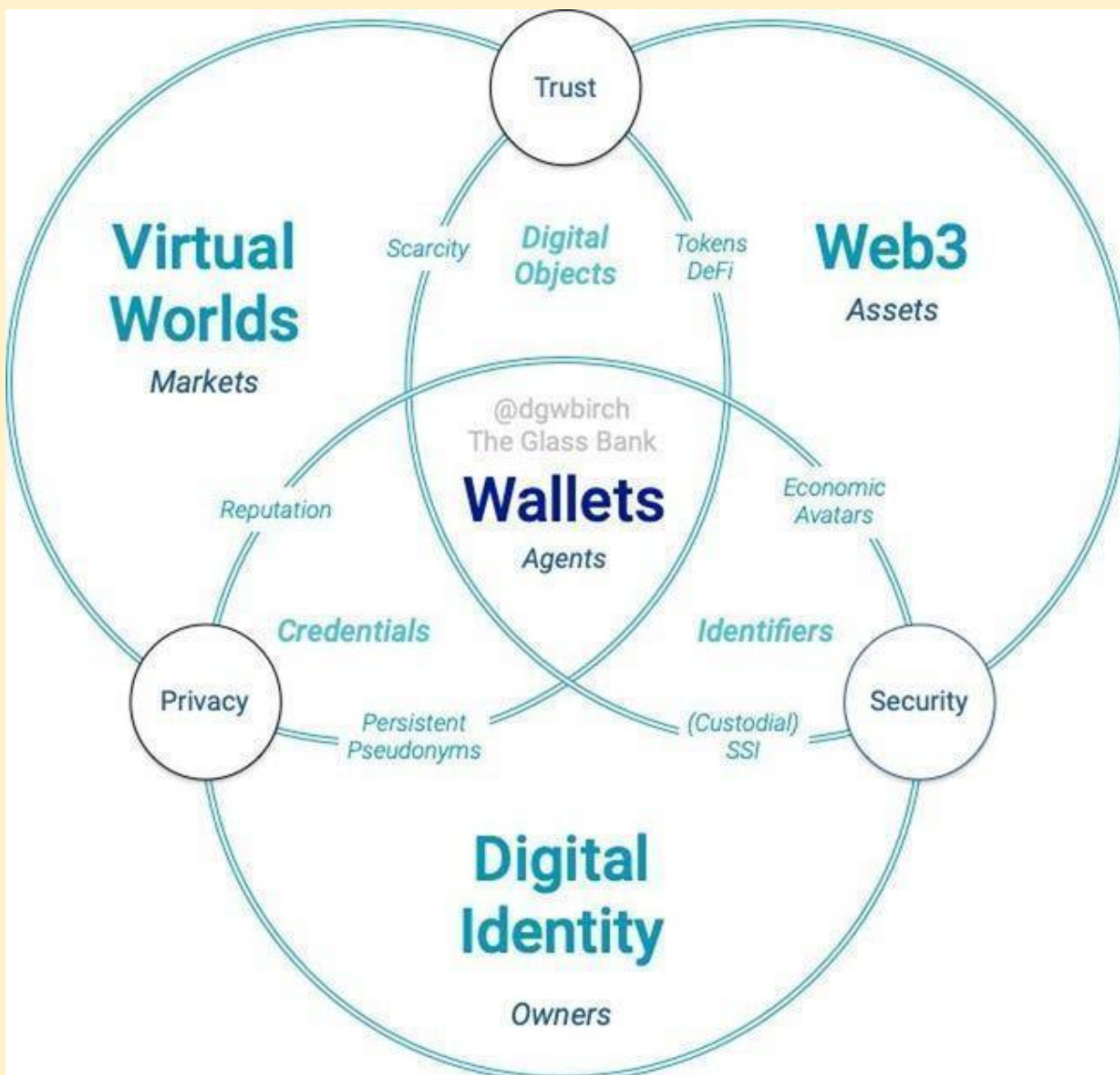
<https://t.co/XcJXlcBb0Y>

- Along with these courses, start practicing Easy to Medium problems in Leetcode and Hackerrank
- This helps to gain a solid understanding of Data Structures & Algorithms.

See attached links too.

#Datastructures #Algorithms #DSA #Programming

## Digital Identify, Virtual World, Web3 Technologies



**Lama Cleaner - Watermark Remover Open source**





**Prithivi Da** · Following

AI/ML Consulting @ Donkey Stereotype

11h · 🌐

Lama-Cleaner: A tool for Watermark removal and other inpainting features that you can self-host.

Image inpainting tool powered by SOTA AI Model. Remove any unwanted object, defect, people from your pictures or erase and replace (powered by stable diffusion) any thing on your pictures.

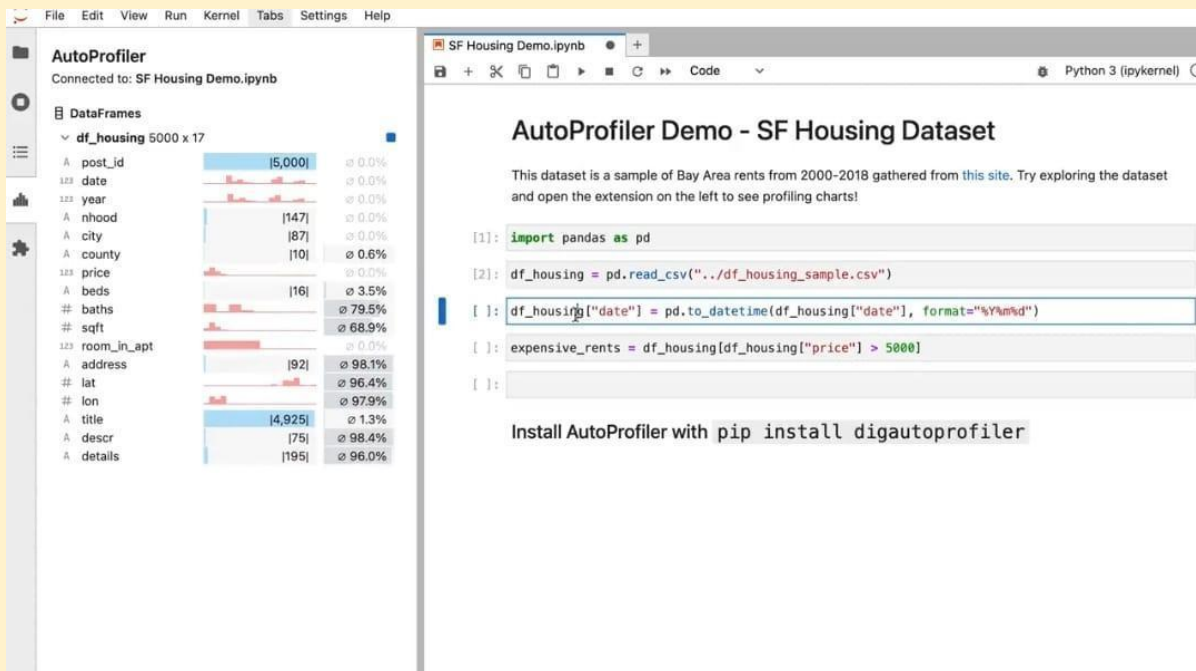
Features:

- Completely free and open-source
- Fully self-hosted
- Classical image inpainting algorithm powered by cv2
- Multiple SOTA AI models
- Support CPU & GPU
- Various inpainting strategy
- Run as a desktop APP

Repo: <https://lnkd.in/eYZX4Rqn>



## Dig Auto Profiler for Datasets



## Learn from the best ML courses by Google

Learn from the best ML courses by Google and master the art of data science 🧠

### 1) Introduction to machine learning

- get started with ML
- Learn about supervised ML

<https://t.co/4PRPpLHnYd>

### 2) machine learning crash course

- How ML differs from traditional programming
- How does gradient descent work?
- End-to-end ML engineering
- Production ML systems and more..

<https://t.co/6GofPqxiMR>

### 3) Introduction to ML problem framing

- Understand how to frame the ML problem

<https://developers.google.com/machine-learning/problem-framing>

### 4) Data preparation and FE in ML

- Data preparation
- Data transformation
- Feature engineering

<https://t.co/PuURww4Exf>

5) testing and debugging ML models

- Train and test ML models
- Debugging the ML models

<https://t.co/jlghgi3lF>

Mathematics foundations are a must to build solid ML skills

1) A friendly introduction to linear algebra by TensorFlow

<https://t.co/1y53KLJqIV>

2) Essence of Calculus playlist by 3blue1brown

<https://t.co/YZIKoSgOCL>

3) Essence of Linear algebra playlist by 3blue1brown

<https://t.co/SSVlkX9kHn>

4) Neural nets foundations

<https://t.co/l5xPVwXrSm>

5) Linear algebra by Gilbert Stange

<https://t.co/V2GMNBb1R5>

6) Statistics by Khan academy

<https://t.co/rszRuUvAWT>

7) Probability by Khan academy

<https://t.co/dUayWaMQIW>

If you can learn math skills, then you can build a stellar career in Natural language processing

## **Basic Git Commands to get started - 14 days of Git**

<https://www.techieclass.com/basic-git-commands-to-get-started/>

## **#Web3 and #Blockchain development**

#Web3 and #Blockchain development are one of the most in-demand skills, learn faster apart from existing technical stack :

1/ Learning Platforms

- Buildspace: Buildspace accelerates your builder journey into web3 - <https://t.co/Tdtjvfesqw>
- LearnWeb3: Build a Web3 career the quickest way possible - <https://t.co/UxVVFvbebi>

Odyssey DAO: It is a Web3 learning community helping one million people join the Web3 economy - <https://t.co/l83PwsaNng>

- Women Build Web3: Unlocking the next wave of web3 builders through education, opportunities, and funding - <https://t.co/rjtJKYp9Dn>
- Web3 University: The comprehensive educational resource on blockchain development with the most useful tutorials - <https://t.co/GcOECIYv29>
- Speed Run Ethereum: Learn how to build on Ethereum; the superpowers and the gotchas - <https://t.co/zQvrpSP9y7>
- Cryptozombies: CryptoZombies is an interactive school that teaches you all things technical about blockchains. Learn to write smart contracts by making your own crypto-collectibles game - Cryptozombies

## 2/ YouTube Channels

- Dapp University: <https://youtube.com/c/DappUniversity>
- Nader Dabit: <https://youtube.com/c/naderdabit>
- Learn Web3 DAO: <https://youtube.com/c/LearnWeb3DAO>
- Moralis: <https://youtube.com/c/MoralisWeb3>
- Patrick Collin: <https://youtube.com/c/PatrickCollins>
- Eat the block: <https://youtube.com/c/EatTheBlocks>

## 3/ Packages, Libraries, and SDK

- Ethers.js: The ethers.js library aims to be a complete and compact library for interacting with the Ethereum Blockchain and its ecosystem.

<https://t.co/GVxgSiliev>

- The Alchemy SDK is the easiest way to connect your dApp to the blockchain. Just download, write two lines of code, and you're ready to go.

<https://t.co/wTMhM6jkwd>

- Livepeer.js: Livepeer.js makes building with Livepeer effortless. It provides a core vanilla JS library to easily connect to a Livepeer provider, as well as React-specific hooks/components to provide memoization and DOM management.

<https://t.co/6V4eoC97Zi>

- WAGMI: wagmi is a collection of React Hooks containing everything you need to start working with Ethereum.

<https://t.co/HBJ5yAtwSN>

- RainbowKit: RainbowKit provides a fast, easy, and highly customizable way for developers to add a great wallet experience to their applications.

<https://t.co/K5yibKgsg8>

## 4/ Protocols and Networks

- Livepeer: It is an Ethereum-based protocol that distributes video transcoding work throughout its

decentralized network.

- Lens Protocol: It leverages the power of Web3 to facilitate building more fair and composable social applications.
- The Graph: The Graph is an indexing protocol for organizing blockchain data and making it easily accessible with GraphQL.
- Lit Protocol: Use Lit as a decentralized serverless function platform to create applications that write data to public state machines like blockchains.
- Push: Push Protocol is a web3 communication network, enabling cross-chain notifications and messaging for dapps, wallets, and services.
- IPFS: IPFS is a distributed system for storing and accessing files, websites, applications, and data.
- Ceramic Network: Ceramic is a decentralized data network that brings unlimited data composability to Web3 applications.
- Chainlink: Chainlink is a decentralized blockchain oracle network built on Ethereum.
- Arweave: It is a Decentralized Storage Network (DSN) that connects people who have extra available computer disk space with those who need more computer storage.

5/ Node Providers

- Chainstack: Chainstack is a single point of access for all your blockchain projects across any protocol.
- Alchemy: Alchemy is a powerful blockchain developer platform providing a suite of developer tools.
- QuickNode: QuikNode simplifies blockchain infrastructure for teams big & small. A lightning-fast global API provides instant access to Ethereum, Bitcoin & xDai networks.
- Ankr: Build on Web3 with a fast, reliable, and distributed network of blockchain nodes. Earn with Liquid Staking and integrate it to power your app.

## 10 Top MLOps Books for Data Scientists

<https://medium.com/geekculture/10-top-mlops-books-for-data-scientists-9b8bb222fd2c>

## GAN Types

1) Instance-Conditioned GAN by Meta AI

Introduced a technique that extends traditional GANs using a combination of local and overlapping clusters. <https://t.co/N2XMS4awvz>

2) A Style-Based Generator Architecture for Generative Adversarial Network by NVIDIA

Introduced StyleGAN model that has become widely used to generate photorealistic synthetic images. <https://t.co/vSRe3etqil>

3) StackGAN: Text to Photorealistic Image Synthesis with Stacked Generative Adversarial Networks by Baidu

Introduced StackGANs that have become one of the most popular text-to-image techniques in deep learning. <https://t.co/Xe2IJVPTXM>



## Machine learning resources YouTube

Machine learning resources(YouTube video Playlists) :

1. We will start with "Stanford CS229: Machine Learning" by Andrew Ng to start and learn the following ML concepts:

Linear & Logistic Regression,  
Naive Bayes, SVMs, Kernels  
Decision Trees, Introduction to Neural Networks  
Debugging ML Models.

<https://t.co/cMLzvscdT>

2.A series of mini-lectures (~5 mins) covering various introductory topics in ML by Cassie Kozyrkov, covering:

Explainability in AI, Precision vs. Recall, Statistical Significance, Clustering and K-means, and finally, Ensemble models. <https://t.co/LiujYMWfBT>

3.Beyond an AI genius, Andrej Karpathy is a brilliant teacher. His creative teaching methods make this intro to Neural Networks (NN): Zero to Hero makes one of the best ways to get introduced to NN.

<https://t.co/WaYzmyHYKU>

4."MIT: Deep Learning for Art, Aesthetics, and Creativity " covers the application of deep learning for art, aesthetics, and creativity, including Neural Abstractions, Efficient GANs, and explorations in AI for Creativity. <https://t.co/cANOWM1M2B>

5.(Striking again) Andrew Ng's "Stanford CS230: Deep Learning (2018)" covers:

The foundations of deep learning, how to build different neural networks (CNNs, RNNs, LSTMs, etc...), how to lead machine learning projects and finally - AI and Healthcare. <https://t.co/F1jBHejS5k>

6.Applied Machine Learning teaches some of the most widely used techniques in ML, including:

Optimization and Calculus, Overfitting and Underfitting, Regularization, Monte Carlo Estimation, and Maximum Likelihood Learning. <https://t.co/2znEMgrJvf>

7.The first part of 'Practical Deep Learning for Coders' teaches you how to:

Build & deploy deep learning models for vision & NLP. Use PyTorch, plus popular libraries like fastai.

<https://t.co/xTq00k7wrt>

8.The 2-hour second part of 'Practical Deep Learning for Coders' takes a deep dive into a recent hot ML topic - Diffusion Models. <https://t.co/82AHonifNK>

9.ML with graphs teaches some of the latest graph techniques in machine learning:

PageRank, Matrix Factorizing, Node Embeddings, Graph Neural Networks, Knowledge Graphs, and finally, Deep Generative Models for Graphs.

<https://t.co/hkgfoFoB9O>

10.This course focuses on the probability and maths behind ML, covering:

Reasoning about uncertainty, Continuous Variables, Sampling, and Markov Chain Monte Carlo.

<https://t.co/Z76gVxel3d>

11.This 12-part Deep Unsupervised Learning aims to teach the latest and most widely used techniques in deep unsupervised learning:

Autoregressive Models, Latent Variable Models, & Self-supervised learning.

<https://t.co/ywkSKC5r5w>

12.'Foundation Models' is a recent course (June 2022) that aims to teach about foundation models like GPT-3, CLIP, Flamingo, and cross-language generalization. <https://t.co/owLqaDXAwj>

8 out of 10 ML breakthroughs you recently heard of are likely based on transformers. "Stanford CS25 - Transformers United" aims to introduce us to the following:

Transformers, its applications in Language (GPT-3), vision & universal compute engines. <https://t.co/nkTtSCG854>

## **NLP resources for Indian languages**

Top NLP research labs (academic) in India MT-NLP Lab (IIIT Hyderabad)

Link: <https://lnkd.in/gP3QYUgZ>

✓ Machine and Language Learning Lab (Indian Institute of Science (IISC))

Link: <https://lnkd.in/gggMay9H>

Computation For Indian Language Technology (CFILT) Lab (Indian Institute of Technology, Bombay) Link:

[https://lnkd.in/gpd\\_gpj3](https://lnkd.in/gpd_gpj3)

Information Retrieval and Extraction Lab (IIIT Hyderabad)

Link: <https://irel.iiit.ac.in>

✓ LINGO - Computational Linguistics and Complex Social Networks Group (Indian Institute of Technology Gandhinagar)

Link: <https://lnkd.in/gVhsJva8>

AI-ML-NLP Group (AI-NLP-ML Research Lab IIT Patna)

Link: [https://lnkd.in/gWXz\\_EV8](https://lnkd.in/gWXz_EV8)

✓ Laboratory for Computational Social Sciences (LCS2) (Indraprastha Institute of Information Technology, Delhi)

Link: <https://www.lcs2.in>

✓ AI4Bharat (Indian Institute of Technology, Madras) Link: <https://ai4bharat.org>

MEDAIM Lab (Clinical NLP focused) (Indraprastha Institute of Information Technology, Delhi)

Link: <https://lnkd.in/gBYMdUvD>

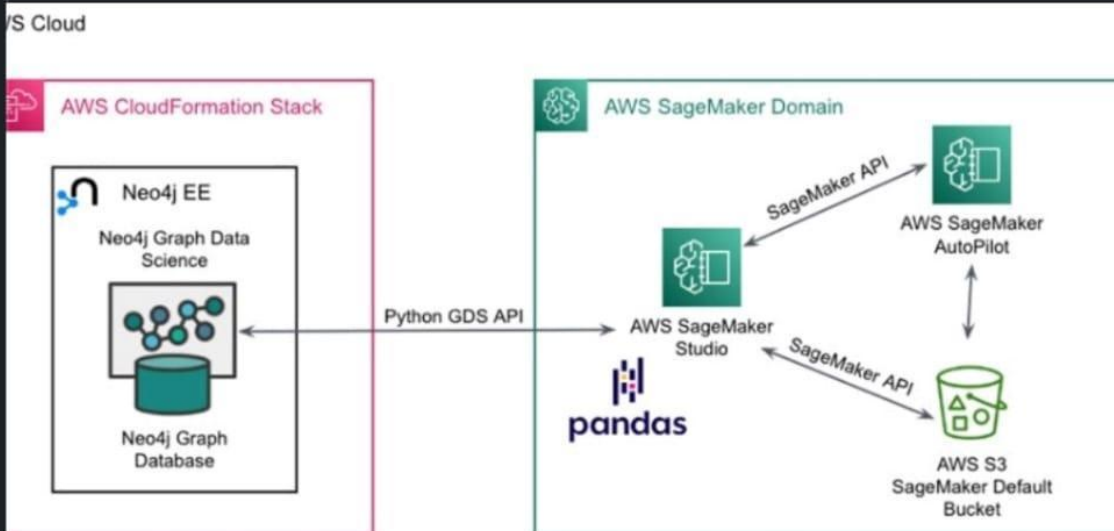
## Announcing Graph Feature Engineering with **Neo4j** and **#Amazon** SageMaker.

With over 65 pretuned graph algorithms and machine learning modeling to analyze your connected data, Neo4j Graph Data Science is the only connected data analysis platform that unifies the ML surface and graph database into a single workspace. And combined with Sagemaker, Neo4j and AWS provide a seamless solution for enhanced ML models through graph feature engineering.

Read more here:

<https://okt.to/RsalrO>

**#AWS #Neo4j #GraphDataScience**



Announcing Graph Feature Engineering with Neo4j and Amazon SageMaker - Neo4j Graph D...

neo4j.com • 3 min read

## Books on Productivity Boost

Books on Productivity Boost

1. Finish What You Start
2. Hyper Focus
3. The One Thing
4. Getting Things Done
5. Effortless
6. 80/20 Principle

## 7 FREE Web3 tutorials you need to know:

7 FREE Web3 tutorials

1. Blockchain Demo

The best and most clear visual explanation for what a blockchain is and how it works.

<https://t.co/vkPBUgI2F>

2. FreeCodeCamp 32 hours video

It will take a while, but it's worth every second. It took me 4 months so go at your own pace and try to understand and code along with Patrick Collins.

<https://t.co/YzGUevCyTy>

3. Cryptozombies

This a great tutorial to get your hands dirty.

Great gamification of the tutorial and great to live stream, too!

<https://t.co/vRfge9V4ay>

4. LearnWeb3

An amazing set of tutorials with different levels, and a great community to support you in case you get lost.

Recommended 100%

<https://t.co/GUMDW8X8h2>

5. Speed Run Ethereum

Cool Challenges to learn web3 "hands-on"

<https://t.co/yg9UzJSKt6>

6. Road to Web3

A 10-week program to learn from the basics of the Blockchain to more advanced

<https://t.co/nO5ADs0FXD>

7. Solidity by example

Not a real tutorial, but an incredible set of code snippets with explanations to get real implementations of

Blockchain use-cases

<https://t.co/kDnXyIkIKY>

## **GitHub - dosisod/refurb: A tool for refurbishing and modernizing Python codebases**

<https://github.com/dosisod/refurb>

## **Researchers At IIIT Allahabad Propose T2CI GAN:**

**A Deep Learning Model That Generates Compressed Images From Text - MarkTechPost**

<https://www.marktechpost.com/2022/10/29/researchers-at-iiit-allahabad-propose-t2ci-gan-a-deep-learning-model-that-generates-compressed-images-from-text/>

## **Deep Learning - UCB Berkeley 2021**

A great series of deep learning lectures that cover ML basics, error analysis, neural networks training mechanics, and architectures(in vision, NLP, generative, RL).

66 videos. Available freely. Self-paced

<https://t.co/IEKq0vVmF>

## **Automated Machine Learning Hyperparameter Tuning in Python**

<https://www.datasciencecentral.com/automated-machine-learning-hyperparameter-tuning-in-python/>

## **Web3 Development and #Blockchain YouTube Courses**

1/ Learn Blockchain, #Solidity, and Full Stack Web3 Development with #JavaScript

This course will give you a full introduction to all of the core concepts related to blockchain and Web3. Follow along with the videos and you'll be a blockchain wizard

<https://t.co/1DluUCpBAG>

2/ Learn Solidity: The Complete Beginner's Guide (Latest Version 0.8)

This is a ~1.5 hours straight to the point course by Dapp University. It teaches you the Solidity basics such as variables, data types, mapping and etc.

<https://t.co/3XQ17WAjwd>

3/ Master #Ethers.js for Blockchain Step-by-Step [Full Course]

ethers.js is a popular Ethereum library for JavaScript and is used by millions. This video shows you how to use ethers to interact with smart contracts, create transactions and etc.

<https://t.co/pmsouZMKd>



#### 4/ Build Spotify Blockchain Web3 dApp with Next.js, Solana, Phantom, QuickNode & Digital Ocean

Once you learned the basics of Web3 and Solidity, here is a great video by Clever Programmer, that will help you build a Web3 Spotify clone

<https://t.co/sUAe19D2zd>

#### 5/ The Complete Guide to Full Stack Web3 Development

In this, in-depth guide you'll learn the tools, protocols, and frameworks for building full-stack web3 apps.

<https://t.co/sFjGBKfsnx>

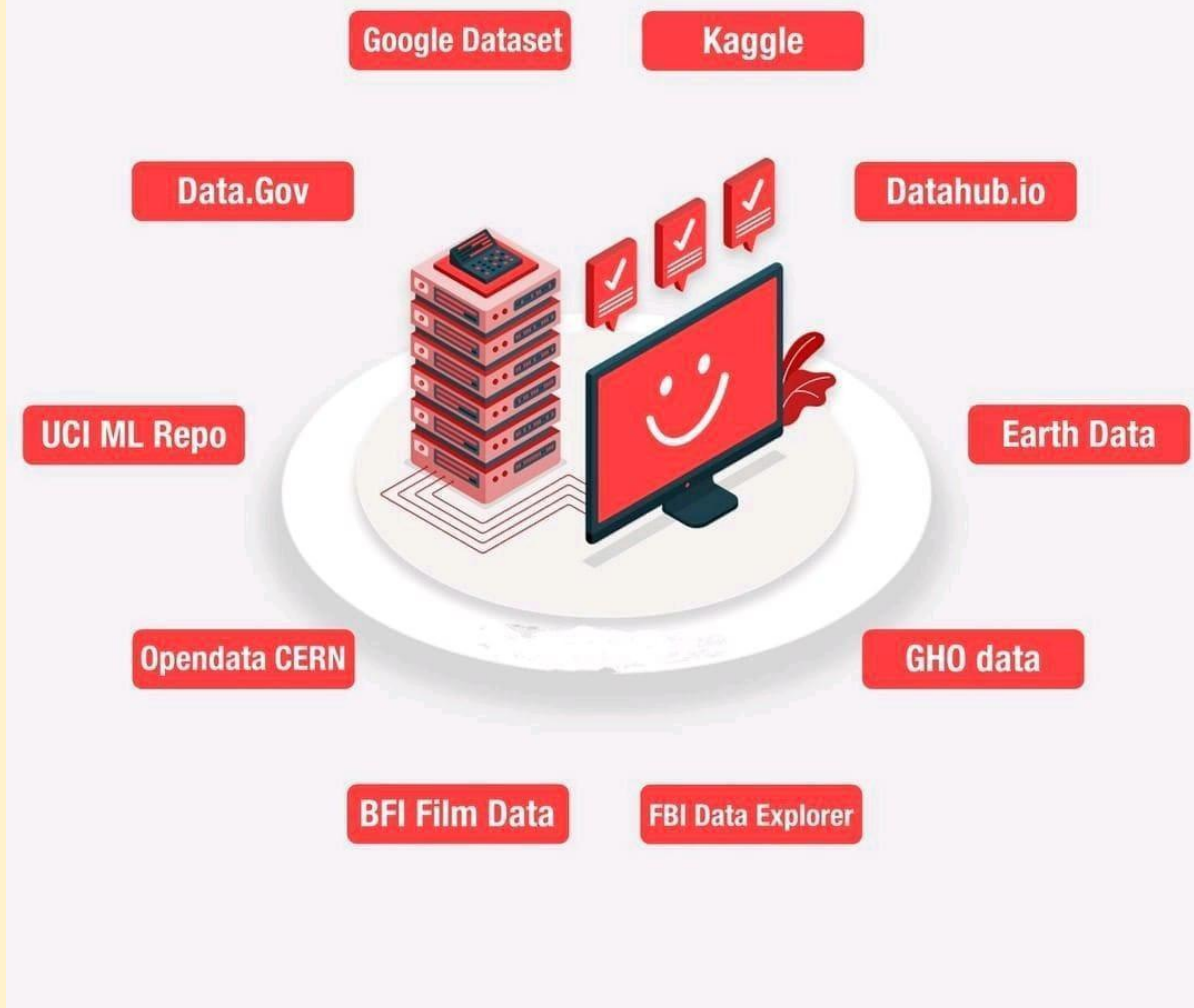
#### 6/ Building smart contracts With #Solana and Rust Lang

If you are a Solana fan, this video is for you. This video will demonstrate how Smart Contracts in Solana work and the rules around programming them.

<https://t.co/PmoTdckalk>

## Top 10 Resources to get Free Datasets

# TOP 10 RESOURCES TO GET FREE DATASETS FOR DATA SCIENCE



## Unsupervised anomaly detection using Anomalib library

Explore a repository developed about unsupervised anomaly detection using Anomalib library, it is in python and is easy to understand. I hope it will be useful for those who want to explore anomaly detection.

Link: <https://github.com/dennishnf/unsupervised-anomaly-detection>

## A Framework for #web3 Infrastructure

[https://mirror.xyz/ericyhu.eth/CGvbcLNZNkZMm-bYMCcyKA8\\_BAbeXBzflfbw8RXXRV2w](https://mirror.xyz/ericyhu.eth/CGvbcLNZNkZMm-bYMCcyKA8_BAbeXBzflfbw8RXXRV2w)

#Metaverse

## **Why building the metaverse is going to be a 'thrilling ride'**

<https://venturebeat.com/virtual/why-building-the-metaverse-is-going-to-be-a-thrilling-ride/>

## **Awesome Diffusion Models**

A fantastic and well-organized collection of learning resources on diffusion models such as introductory papers, survey papers, intro videos, long lectures, and blog posts. Papers in vision, natural language, tabular, graph, etc.

<https://t.co/Dfmkv9mmKD>

## 32 Important Datasets to Start Data Science

🔥 Go through this list of 32 datasets over the weekend to uplift your skills in data science:


[https://hubs.la/Q01n\\_rYW0](https://hubs.la/Q01n_rYW0)

Source: [Data Science Dojo](#)

#Datasets #DataScience #Weekend #DSDojo

# 32 Datasets

## to Uplift your Skills in Data Science

|                                                                                                                                                                |                                                                                                                                                                |                                                                                                                                                                 |                                                                                                                                                                  |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  <p>Find out the age of Abalone from physical measurements</p>                |  <p>Predict student knowledge level</p>                                       |  <p>Can you predict the price of a house?</p>                                  |  <p>Can you estimate location from WIFI signal strength</p>                   |
|  <p>Predict acceptability of a car</p>                                      |  <p>Predict seminal quality of an individual</p>                            |  <p>Estimate chance of bankruptcy from qualitative parameters by experts</p> |  <p>Can you predict the fuel-efficiency of a car?</p>                       |
|  <p>Was that chest pain an indicator of a heart disease</p>                 |  <p>Predict total number of demand of orders</p>                            |  <p>Find out if a donor will give blood in March 2007</p>                    |  <p>Forecast pollution level of a city</p>                                  |
|  <p>Will the patient survive for at least one year after a heart attack</p> |  <p>Estimate compressive strength of concrete</p>                           |  <p>Discover patterns relating liver disorder and alcohol consumption</p>    |  <p>Predict which stock will provide greatest rate of return</p>            |
|  <p>Assess heating and cooling load requirements of building</p>            |  <p>Determine the type of glass using oxide content</p>                     |  <p>Predict chance of survival</p>                                           |  <p>Find patterns from spending data at wholesale</p>                       |
|  <p>Group similar travel reviews</p>                                        |  <p>Relate returns of Istanbul SE with other international indices</p>      |  <p>Predict bike rental count based on the environmental settings</p>        |  <p>Detect Room Occupancy through Light, Temperature, &amp; CO2 sensors</p> |
|  <p>Estimate whether a person's income exceeds \$50K/year</p>               |  <p>Coronavirus (COVID-19) Dataset</p>                                      |  <p>Detect Autistic Spectrum Disorder (ASD) Cases</p>                        |  <p>Estimate the probability of Default</p>                                 |
|  <p>Predict if a note is genuine</p>                                        |  <p>Find a short term forecast on electric consumption of a single home</p> |  <p>On social networks Predict the number of shares</p>                      |  <p>Amazon Product Reviews Data</p>                                         |

## 8 highly appreciated technical posts & videos

Here is a list of 8 must know topics in field of Data Engineering.

SQL Internals -

<https://lnkd.in/dhHHCIES>

Databricks -

<https://lnkd.in/gRcjQXyB>

ACID properties -

<https://lnkd.in/gDms3N-t>

Delta Lake -

<https://lnkd.in/gAmY7BTv>

Normalization vs Denormalization -

<https://lnkd.in/grX8Ry7G>

Order of execution in SQL- <https://lnkd.in/gunTiiZW>

OLTP vs OLAP -

<https://lnkd.in/gNSs626E>

CTE (Common table Expression) <https://lnkd.in/gYZBApWj>

## Salesforce Einstein, future of CRM

AI, machine learning, deep learning, predictive analytics, smart data discovery <https://t.co/RhYTLEkyyy>

## Machine Learning Models for Object Recognition in Underwater Video Images

Performance Analysis of Machine Learning Models for Object Recognition in Underwater Video Images

<https://t.co/7pCHVcRzgR> #AI #MachineLearning #DataScience #ArtificialIntelligence

## Python for Bioinformatics - Drug Discovery Using Machine Learning and Data Analysis

Learn how to use Python and machine learning to build a bioinformatics project for drug discovery.

<https://t.co/LzXslwkVaK>

## An introduction to cheminformatics, data analysis and machine learning

20 08 21 - Filed in: Cheminformatics Workshop Tutorials RSC CICAG

The video of the latest RSC CICAG Open Source Tools for Chemistry is now online.

Part of the RSC CICAG Open Chemical Sciences workshop series. Workshop given by Pat Walters (<http://practicalcheminformatics.blogspot.com>) entitled "An introduction to cheminformatics, data analysis and machine learning".

A hands-on workshop on building and validating ML models, including:

Initial exploratory data analysis

ML model building

Model evaluation

Making predictions on a larger data set

The notebooks for the tutorial are here [https://mybinder.org/v2/gh/PatWalters/chem\\_tutorial/HEAD](https://mybinder.org/v2/gh/PatWalters/chem_tutorial/HEAD). If there are any problems, you should be able to just reload the page. The code and notebooks are in a GitHub



repo.

The video is here YouTube:[https://youtu.be/2ZjerAGS\\_IQ](https://youtu.be/2ZjerAGS_IQ)

## AI to discover faster matrix multiplication

DeepMind unveils first AI to discover faster matrix multiplication algorithms.

<https://venturebeat.com/ai/deepmind-unveils-first-ai-to-discover-faster-matrix-multiplication-algorithms/>

<https://www.newscientist.com/article/2340343-deepmind-ai-finds-new-way-to-multiply-numbers-and-speed-up-computers/>

## Free Certification Courses in Data Science using Python from IBM.

1. Python for Data Science

<https://t.co/w8KTgnXwnc>

2. Data Analysis with Python

<https://t.co/vmqqRAGpLK>

3. Data Visualization with Python

<https://t.co/nwjadv6nqU>

Top NLP online courses

[https://www.linkedin.com/posts/kalyankatikapallisubramanyam\\_nlproc-nlp-deeplearning-activity-6984349048073842688-7\\_Lu?utm\\_source=share&utm\\_medium=member\\_android](https://www.linkedin.com/posts/kalyankatikapallisubramanyam_nlproc-nlp-deeplearning-activity-6984349048073842688-7_Lu?utm_source=share&utm_medium=member_android)

Get latest info on ML by subscribe to

<https://alphasignal.ai/>

Building reusable Machine Learning workflows with Pipeline Templates

<https://cloud.google.com/blog/topics/developers-practitioners/building-reusable-machine-learning-workflows-pipeline-templates>

## Aerial Vehicles detection Check

Aerial Vehicles detection Check out open-source dataset "Aerial Multi-Vehicle Detection Dataset" at

<https://zenodo.org/record/7053442>.

This dataset can be used for:

- Training an aerial vehicle detection algorithm
- Enhance vehicle detection dataset
- Benchmark vehicle detectors
- Traffic monitoring analysis

Annotation formats:

[#Yolo](<https://www.linkedin.com/feed/hashtag/?keywords=yolo&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A6978701293166944257>),

[#COCO](https://www.linkedin.com/feed/hashtag/?keywords=coco&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A6978701293166944257),

[#VOC](https://www.linkedin.com/feed/hashtag/?keywords=voc&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A6978701293166944257)

Give it a try.

PS. Image depicts an example of applying detecting and tracking algorithms using this dataset.

## 14 Data Science Skills

The 14 most important data science skills to acquire.

[#datacaptain](#) [#datascience](#) [#machinelearning](#)  
[#learning](#) [#machinelearning](#)

| Plan                                     | Skills                                                                                                                                                                                                                                                |
|------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Machine Learning                         | Supervised Classification, Supervised Regression, Unsupervised Clustering, Dimensionality Reduction, Local Interpretable Model Explanation - H2O Automatic Machine Learning, parsnip (XGBoost, SVM, Random Forest, GLM), K-Means, UMAP, recipes, lime |
| Data Visualization                       | Interactive and Static Visualizations, ggplot2 and plotly                                                                                                                                                                                             |
| Data Wrangling & Cleaning                | Working with outliers, missing data, reshaping data, aggregation, filtering, selecting, calculating, and many more critical operations, dplyr and tidyr packages                                                                                      |
| Data Preprocessing & Feature Engineering | Preparing data for machine learning, Engineering Features (dates, text, aggregates), Recipes package                                                                                                                                                  |
| Time Series                              | Working with date/datetime data, aggregating, transforming, visualizing time series, timetk package                                                                                                                                                   |
| Forecasting                              | ARIMA, Exponential Smoothing, Prophet, Machine Learning (XGBoost, Random Forest, GLMnet, etc), Deep Learning (GluonTS), Ensembles, Hyperparameter Tuning, Scaling to 1000s of forecasts, Modeltime package                                            |
| Text                                     | Working with text data, Stringr                                                                                                                                                                                                                       |
| NLP                                      | Machine learning, Text Features                                                                                                                                                                                                                       |
| Functional Programming                   | Making reusable functions, sourcing code                                                                                                                                                                                                              |
| Iteration                                | Loops and Mapping, using Purrr package                                                                                                                                                                                                                |
| Reporting                                | Rmarkdown, Interactive HTML, Static PDF                                                                                                                                                                                                               |
| Applications                             | Building Shiny web applications, Flexdashboard, Bootstrap                                                                                                                                                                                             |
| Deployment                               | Cloud (AWS, Azure, GCP), Docker, Git                                                                                                                                                                                                                  |
| Databases                                | SQL (for data import), MongoDB (for apps)                                                                                                                                                                                                             |

## **How to add a Subdomain in Github Pages- Tutorial 3**

<https://youtu.be/-pJq3XZQbt0>

## **Another tool won't fix your MLOps problems \ by David Hershey \ Medium**

<https://dshersh.medium.com/too-many-mlops-tools-c590430ba81b>

# MLOPS is not MLOPS Tools

## MLOps ≠ MLOps Tools

MLOps needs to be built on top of a culture of delivering ML products.

Excited to re-open my Medium blog and start writing about the space! I've learned a lot in the last few years, and hopefully some of it can be useful to folks here.

Check it out, and let me know what you think!

[#ml](#) [#machinelearning](#) [#mlops](#) [#devops](#) [#data](#)  
[#culture](#) [#blogging](#)



Another tool won't fix your MLOps problems

[link.medium.com](https://link.medium.com)



Below shows break down of the 6 steps of #DataScience

- 1 Data Engineering
- 2 Data Analysis
- 3 Model Development
- 4 ML Engineering
- 5 Insights Activation
- 6 Orchestration

#### 1 Data Engineering 📌

- ◆ Ingest & Discover- Data Fusion, Pub/Sub, Dataflow, Datastream, Storage/BigQuery data transfer
- ◆ Process - Dataflow, Dataproc, Dataprep, Data Labeling
- ◆ Store - GCS, BigQuery, Cloud SQL, Bigtable, Dataplex
- ◆ Catalog - Data Catalog, Vertex Metadata

#### 2 Data Analysis 📌

- ◆ Exploration
- ◆ Processing
- ◆ Insights

Some #GoogleCloud tools that help in this stage are Vertex AI Workbench, BigQuery, Looker, Dataflow, Dataproc, Dataprep, Data Studio

#### 3 Model Development 📌

- ◆ Feature Engineering
- ◆ Model Training
- ◆ Feature Store
- ◆ Model Registry
- ◆ Model Evaluation

Some #GoogleCloud tools that help in this stage are BigQuery ML, Vertex AI Workbench, AutoML, Vizier, Tensorboard, Spark on Google Cloud.

#### 4 ML Engineering 📌

- ◆ Model Serving
- ◆ Feature Store
- ◆ Model Deployment
- ◆ Model Monitoring

Some #GoogleCloud tools that help in this stage are BigQuery ML, Vertex AI Workbench, AutoML, Prediction, Model Monitoring

#### 5 Insights Activation 📌

- ◆ Influence business & customer decisions
- ◆ Serve other applications & services

Some #GoogleCloud tools that help in this stage are BigQuery, Looker, Data studio, Apigee, Cloud Functions, Cloud Run.

#### 6 Orchestration

- ◆ Data Pipelines - Cloud Composer, Cloud Scheduler
- ◆ ML Pipelines - Vertex AI Pipelines, TFX, Cloud Composer
- ◆ General orchestration tools - Workflows, Cloud Run, Pub/Sub, Cloud Functions

## Set of Techniques for Boosting Model Efficiencies

A Latest Machine Intelligence Research Built A Stronger and Faster Baselines Based on a Set of Techniques for Boosting Model Efficiencies

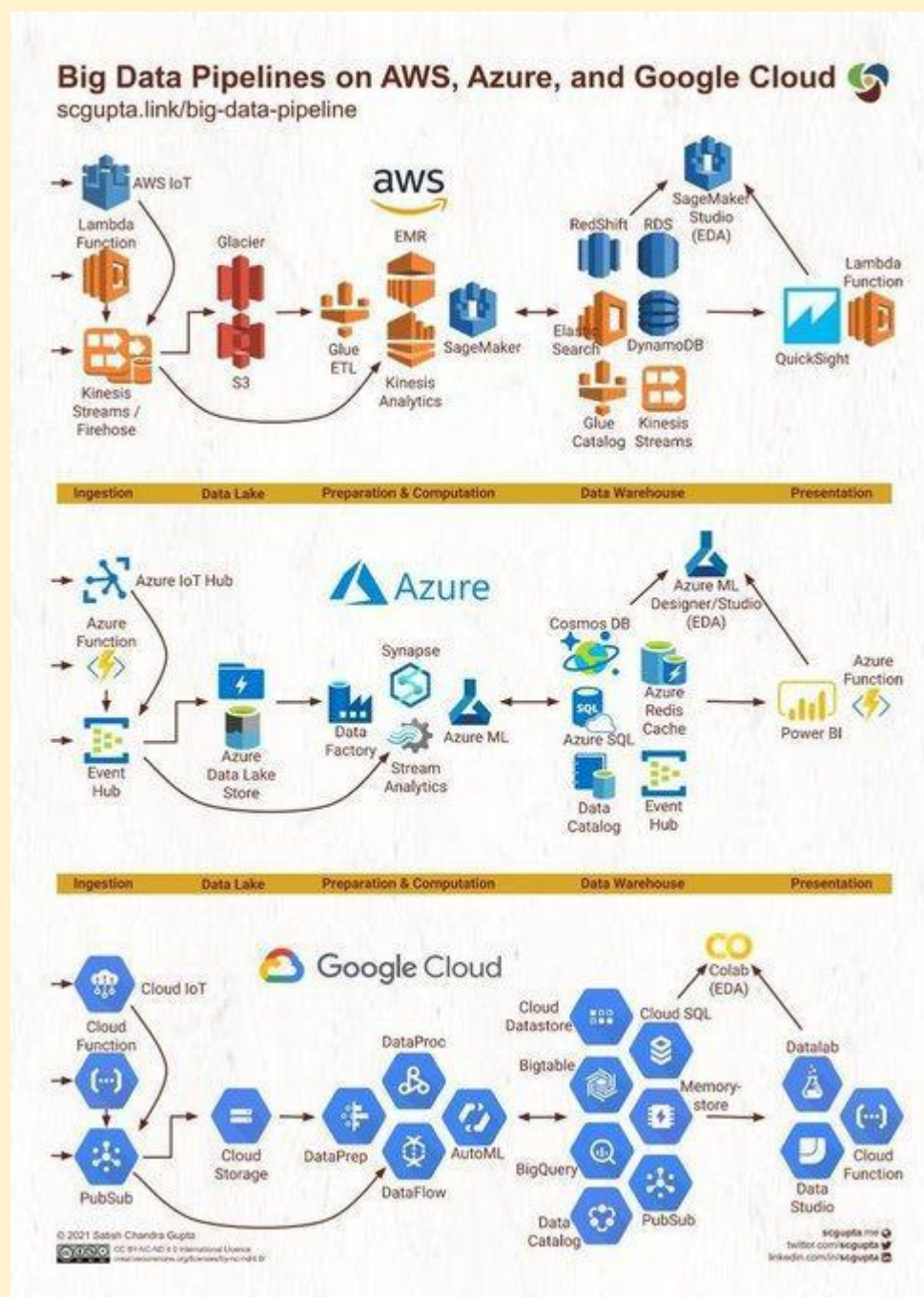
<https://www.marktechpost.com/2022/10/02/a-latest-machine-intelligence-research-built-a-stronger-and-faster-baselines-based-on-a-set-of-techniques-for-boosting-model-efficiencies/>

## Researchers at ActiLoop AI Introduce 'Deep Lake'

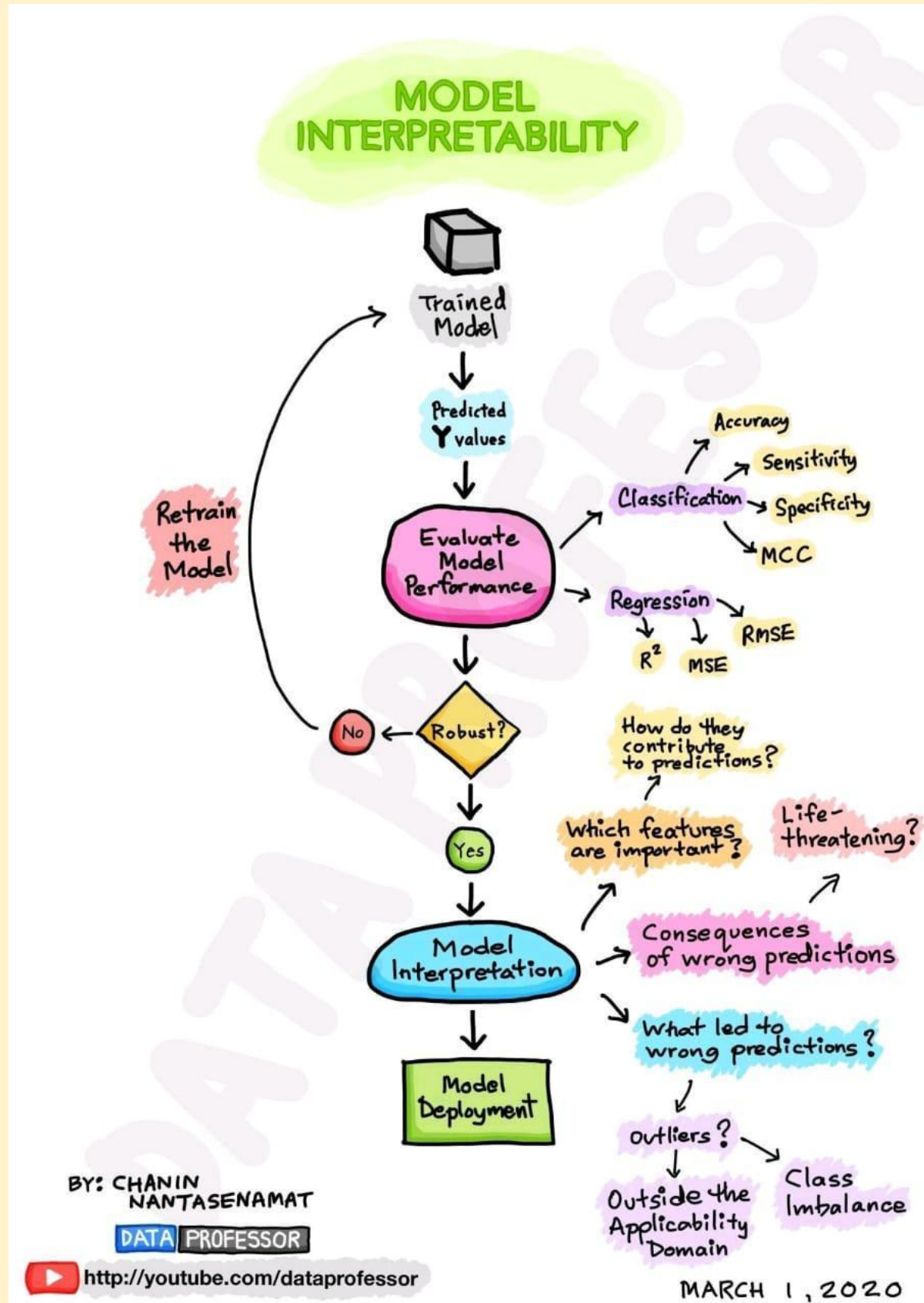
an Open-Source Lakehouse for Deep Learning Applications - MarkTechPost

<https://www.marktechpost.com/2022/10/02/researchers-at-actiLoop-ai-introduce-deep-lake-an-open-source-lakehouse-for-deep-learning-applications/>

## Big data pipeline on AWS Azure and GCP



Model interpretability

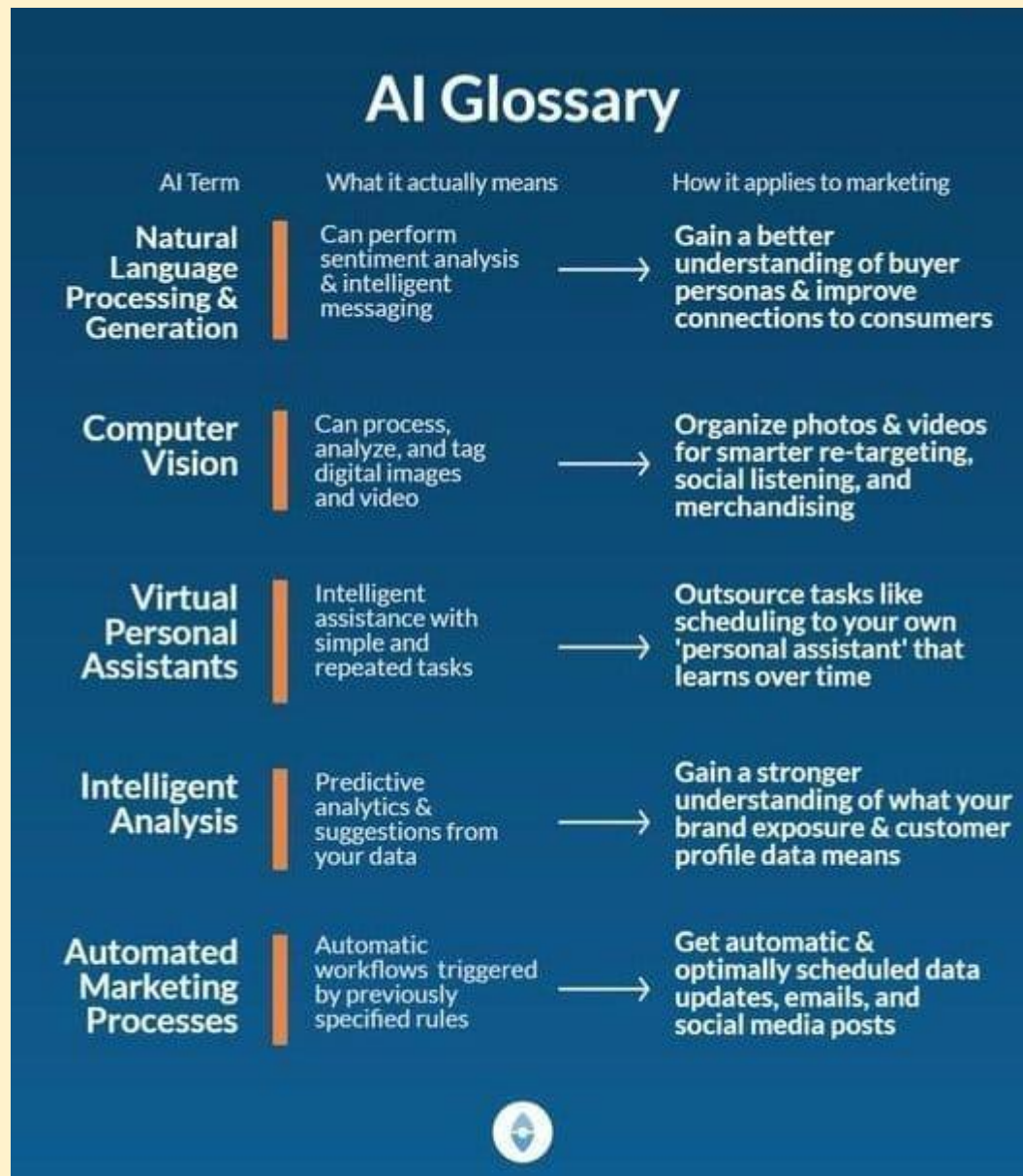


## Distributed Data Parallel in PyTorch - Video Tutorials

PyTorch Tutorials 1.12.1+cu102 documentation

[https://pytorch.org/tutorials/beginner/ddp\\_series\\_intro?utm\\_source=organic\\_social&utm\\_medium=linkedin&utm\\_campaign=distributed\\_data\\_parallel](https://pytorch.org/tutorials/beginner/ddp_series_intro?utm_source=organic_social&utm_medium=linkedin&utm_campaign=distributed_data_parallel)

## AI technologies for marketing





## Why people give up?



## Causal inference

Randomised control test  
Ab test

Select participant  
Create two groups  
Send two set of emails two two groups  
Analyse the purchase data  
Make the decision which email should be sent

Users are selected randomly among control group and treatment group

Control group get the treatment and treatment group get the Placebo

This sending email causes the purchase increase which is called causality

What about if I want to check the efficacy of my Billboard. Ab testing will not work

Because it is expensive  
Because it is time consuming



Because the experiment is entangled with to time

If an Alixir need to be tested for a flu

If Ace is lesser like 35 and treatment will automatically happen compared to when is high like 65

Group A which is having less age people people are having effect of the medicine then it may be because of confounder that is why we need to rendmaize the group

Does confounder variable is a challenge in causal inferencing that uses the prior data

Confronting variables like age or any other thing should be equal between two groups

These confounder variables need to be identified and addressed

This will cause selection buyers and it means the control group is not the representation of the population

Another problem is counter factuals.

When we say what would have happened when treatment group people could have got the treatment or control group people could not have got the treatment

This problem can be addressed by Apple to Apple pairing

Machine learning technique or matching technique can be used for this

Assumption 1

Causal Markov graph are directional graphs. Relationship between confounder treatment and outcome

Assumption 2 Sutva

Stable user treatment value assumption. To group does not affect each other at all during the treatment.

Assumption 3 ignoreability

It means there is no other important variable which influence the treatment. This is also called unknown confounder

## **Thesis Defence Questions**

# List Of 20 Common Thesis Defense Questions You Should Be Prepared For

1. The most common question you may be asked is what you learned from the study you have done. You have to sum up your entire study in a few sentences and remember the technical terms you have mentioned in your research because that is what your examiner wants to hear from you.
2. The next question to follow by default is why you chose this particular topic or what your inspiration behind this study was. This is one of the trickiest questions as you have to prove your convincing power to the panel of the teachers that what you did is valuable for the society and was worth their time. Tell about how zealous you were about this particular problem.
3. What is the importance of your study or how will it contribute or add up to the existing body of knowledge?
4. You may be asked to summarize your key findings of the research.
5. What type of background research have you done for the study?
6. What are the limitations you have faced while writing?
7. Why did you choose this particular method or sample for the study?
8. What will you include if you are told to add something extra to the study?
9. What are the recommendations of your study?
10. Who formed your sample and why you selected this particular age group?
11. What was your hypothesis and how you framed it?
12. If given a chance, would like to do something different with your work?
13. What are the limitations you faced while dealing with your samples?
14. How did you relate your study to the existing theories?
15. What is the future scope of this study?
16. What do you plan to do with your work after you have completed your degree?
17. What are the research variables you used?
18. Do you have any questions to be asked?
19. Did you evaluate your work?
20. How would you improve your work?

These are some of the very general but a bit complicated questions you may be asked during your interview.

## **Katonic MLOps Platform \ Katonic**

<https://www.katonic.ai/>

85% of AI models don't make it to production

Our MLOps platform helps you build and deploy Machine Learning Models into production quickly, securely, and cost-effectively

## **Neptune vs TensorBoard - [neptune.ai](https://neptune.ai)**

<https://neptune.ai/vs/tensorboard>

Home - [neptune.ai](https://neptune.ai)

<https://neptune.ai/home>

Log, organize, compare, register, and share all your ML model metadata in a single place.  
Automate and standardize as your modeling team grows.  
Hosted, on-premises, private cloud.

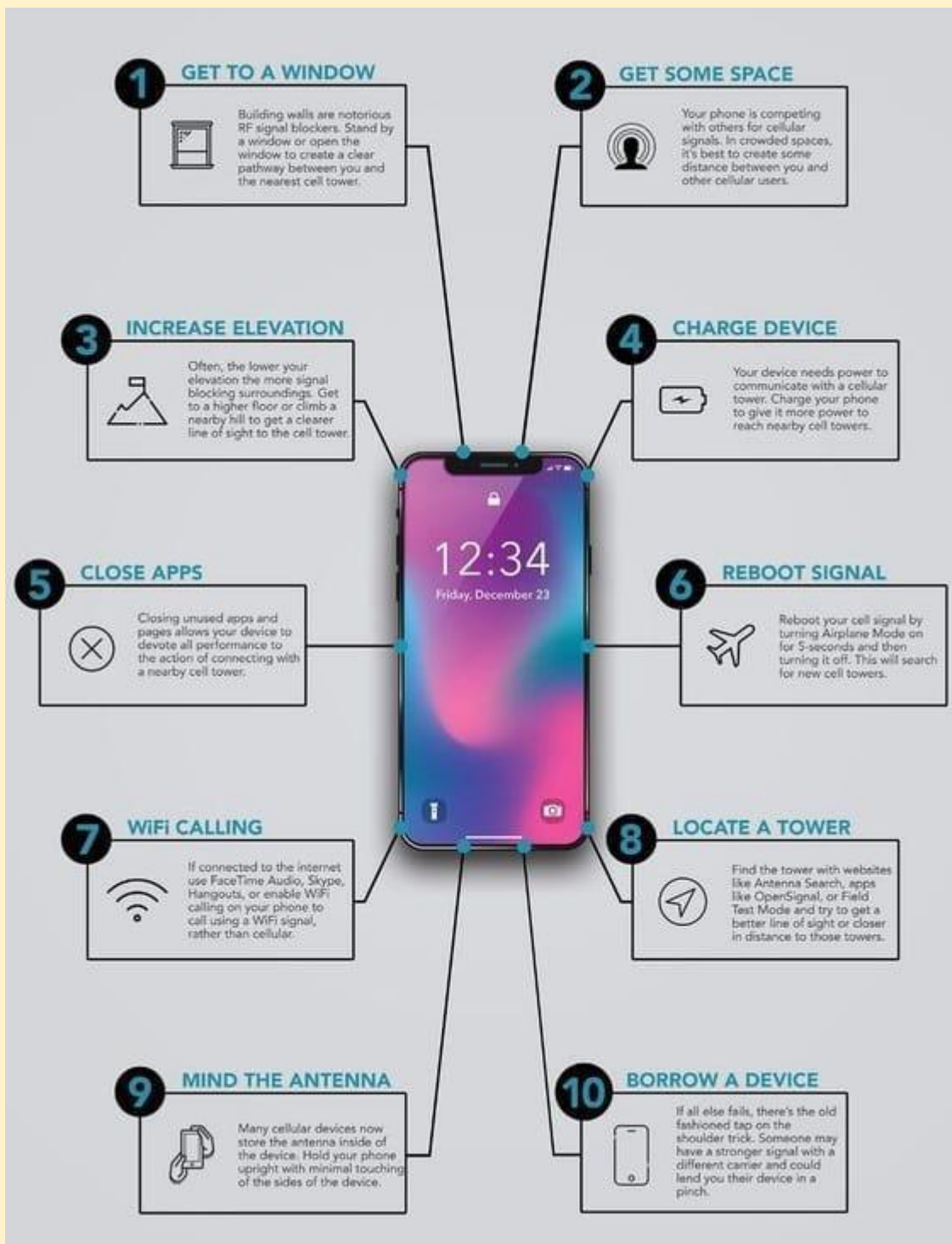
## **Synthetic Data - Machine Learning Techniques**

<https://mltechniques.com/category/synthetic-data/>

## **Transformer Models**

<https://bit.ly/3cXsPfB>

## How to get better cell phone signal



## What is GATO?

GATO is a deep neural network for a range of complex tasks that exhibits multimodality. It can perform tasks such as engaging in a dialogue, playing video games, controlling a robot arm to stack blocks, and more. It was created by researchers at London-based AI firm DeepMind. It is a transformer, like GPT-3. According to MIT Technology Review, the system "learns multiple different tasks at the same time, which means it can switch between them without having to forget one skill before learning another" whereas "[t]he AI systems of today are called "narrow," meaning they can only do a specific, restricted set of tasks such as generate text", and according to The Independent, it is a "'generalist agent' that can carry out a huge range of complex tasks, from stacking blocks to writing poetry". The technology has been described as "general purpose" artificial intelligence and a "step toward" artificial general intelligence.

Inspired by progress in large-scale language modeling, we apply a similar approach towards building a single generalist agent beyond the realm of text outputs. The agent, which we refer to as Gato, works as a multi-modal, multi-task, multi-embodiment generalist policy. The same network with the same weights can play Atari, caption images, chat, stack blocks with a real robot arm and much more, deciding based on its context whether to output text, joint torques, button presses, or other tokens. In this report we describe the model and the data, and document the current capabilities of Gato.

Paper : <https://arxiv.org/abs/2205.06175>

## How do Google/Apple maps blur license plates

How do Google/Apple maps blur license plates and human faces on Street View?

The diagram below presents a possible solution that might work.

The high-level architecture is broken down into three stages:

- a. Data pipeline - prepare training data set
- b. Blurring pipeline - extract and classify objects and blur relevant objects (license plates, faces)
- c. Serving pipeline - serve blurred street view images to users.

### ① Data pipeline

Step 1: We get the annotated dataset for training. The objects are marked in bounding boxes.

Steps 2-4: The dataset goes through preprocessing and augmentation to be normalized and scaled.

Steps 5-6: The annotated dataset is then used to train the machine learning model, a 2-stage network.

### ② Blurring pipeline

Steps 7-10: The street view images go through preprocessing, and object boundaries are detected in the images. Then sensitive objects are blurred, and the images are stored in an object store.

### ③ Serving pipeline

Step 11: The blurred images can now be retrieved by users.

Can you think of other objects that should be blurred from stored/live images?



## Sales funnel



## How can banks become #AI/#Metaverse ready?

Fraud Prevention in Banking and Finance With AI

Financial Services Review : Banks are now performing machine learning algorithms that can detect fraud transactions and much more.

Read More: <https://t.co/7jLEcGAizO>

#financialservices #finance #banking #algorithms #retail #fraud #artificialintelligence

How can financial service operators build trust in the digital age? Leveraging AI and machine learning, banks can create customer-first and personalised interactions. Read more in blog 🙌

<https://t.co/kBjx8WYvfQ> #AI #CustomerExperience

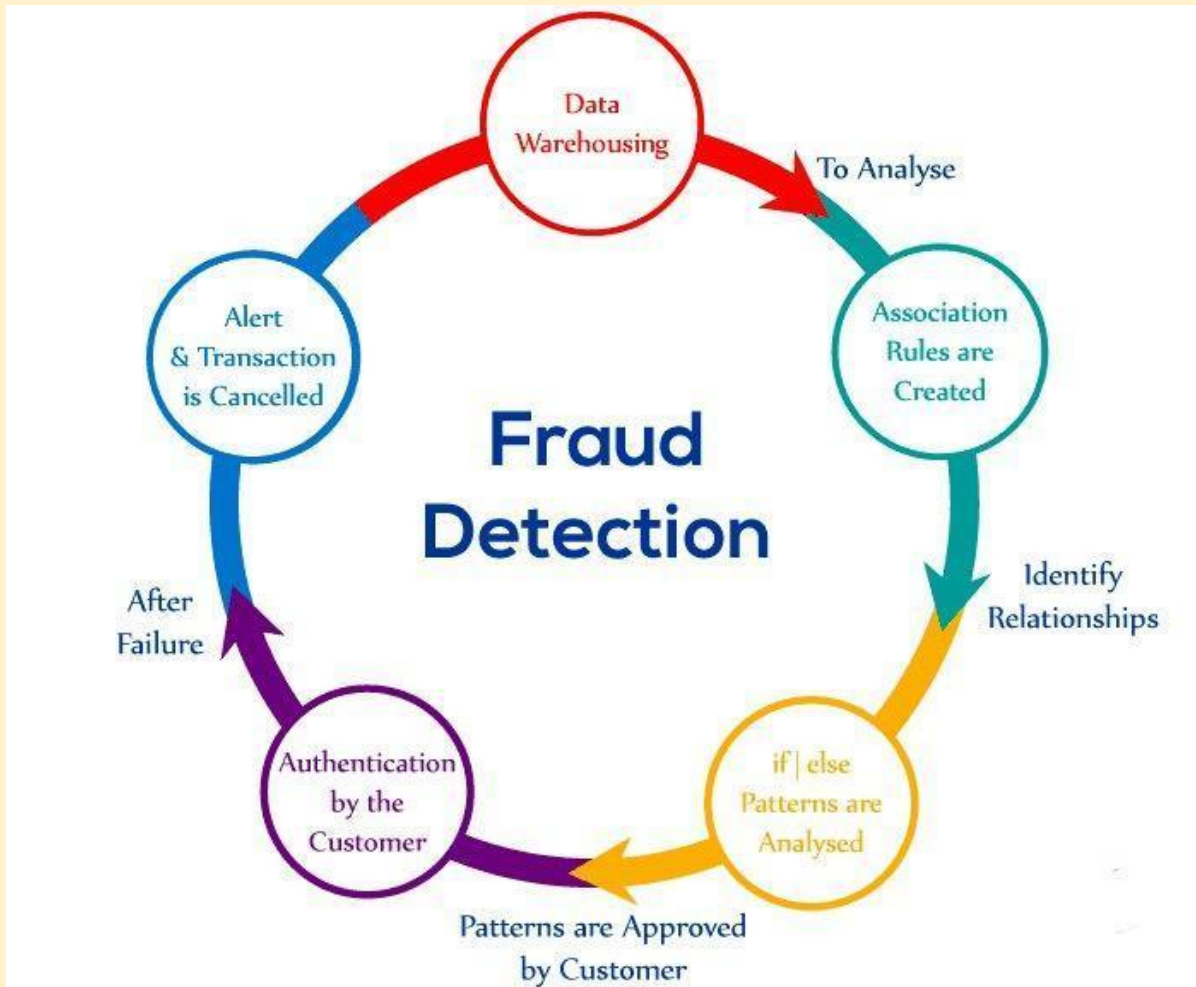
Putting AI and machine learning to work against fraud for banks, PSP's and merchants, POD cast  
:<https://brighterion.com/putting-ai-ml-to-work-against-fraud-for-banks-psps-and-merchants-podcast/>

How AI, Machine Learning & Big Data Can Help Shape Your Bank's Compliance Strategy

<https://bis.lexisnexis.co.uk/blog/categories/data-as-a-service/AI-MachineLearning-Banks-Compliance>

A machine learning program trains itself by associating words and phrases with image data. It then categorizes that image data into large banks to pull from when prompted. When printed it will use the complied image data as reference to create a new image.

## Fraud detection



## Learn #Web3 for free

Ethereum: <https://t.co/dm50GZ8IMP>

Solidity: <https://t.co/BfjoA5ynEt>

Oracles: <https://t.co/kzVyl8leHK>

Tutorial 1: <https://t.co/QRwZn0CvPI>

Tutorial 2: <https://t.co/GUMDW8WArU>

Project Ideas: <https://t.co/YN5mvKCTfV>

32 hours video: <https://t.co/YzGUevkY20>

# HOW TO BEGIN A DIGITAL DETOX?



If you feel that you are like me, a 21-century junky <br> . you'll find below a mix of personal advice to reduce your digital consumption.



01

## NO ELECTRONICS WHEN YOU JUST WAKE UP

The first reflex of my girlfriend when she wakes up is to check her Instagram and Facebook... When you wake up, it's the moment your brain is the freshest. Use this moment wisely.

## STOP ALL SMARTPHONE NOTIFICATIONS (EXCEPT CALLS OR IMPORTANT STUFF)

If you let all your notifications active (updates, messages, emails, notifications), you can't focus.

02



## SETTLE A TIME FRAME WHEN YOU DISCONNECT

After 10 pm, avoid any screens (especially in bed). The screens blue light might disturb your sleep.



03

## REDUCE THE NUMBER OF WEB TOOLS YOU NEED

Tech is an investment. Focus where you get the maximum satisfaction and results. The less platforms you use, the less waste of your time.

04



05

## ACCEPT THE FOMO

The fear-of-missing-out keeps us hooked. It's social anxiety led by our voyeurism or the desire to stay connected with what others are doing.



## TRY OFFLINE ACTIVITIES

Find your dopamine suppliers somewhere out of the web. A new sport? A new artistic hobby?

06

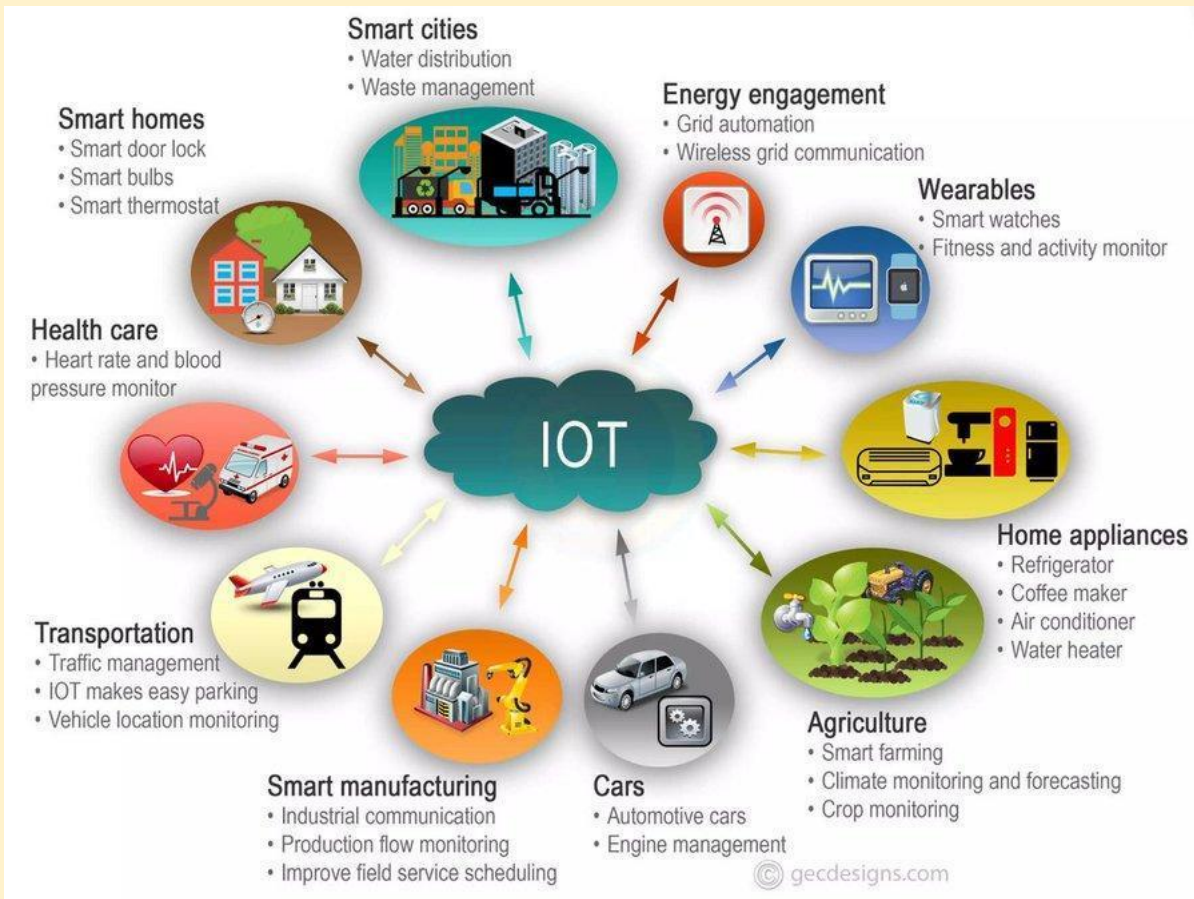


Into the metaverse: How conversational AI will build its experiential foundation

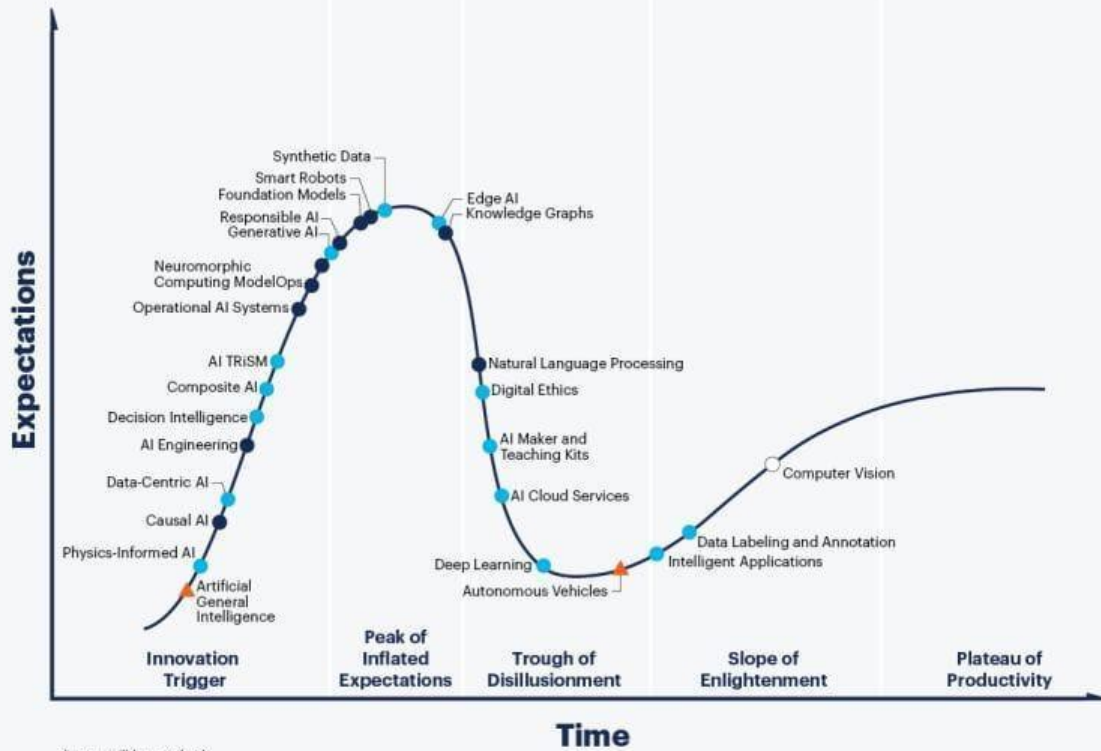
<https://venturebeat.com/ai/into-the-metaverse-conversational-ai-build-experiential-foundation/>

Application of iot





# Hype Cycle for Artificial Intelligence, 2022



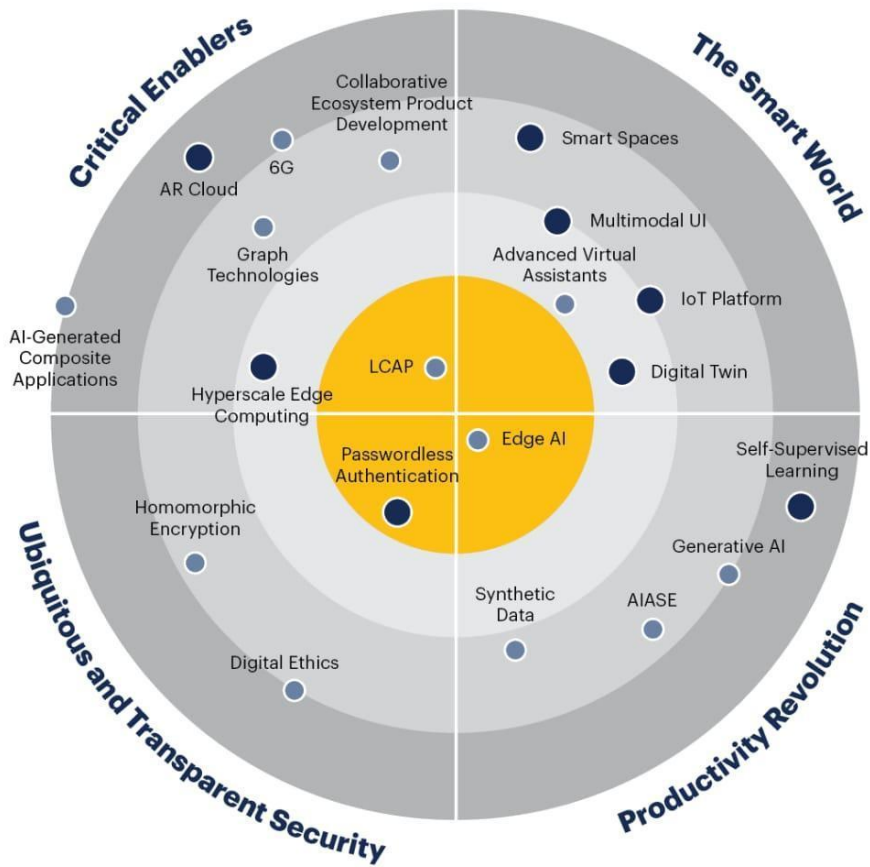
Plateau will be reached:  
○ less than 2 years   ● 2 to 5 years   ● 5 to 10 years   ▲ more than 10 years   ⊗ obsolete before plateau   As of July 2022

[gartner.com](https://www.gartner.com)

Sources: Gartner  
© 2022 Gartner, Inc. and/or its affiliates. All rights reserved. Gartner and Hype Cycle are registered trademarks of Gartner, Inc. and its affiliates in the U.S. 1957302



# Impact Radar for 2022



## Range

- 6 to 8 Years
- 3 to 6 Years
- 1 to 3 Years
- Now (0 to 1 Years)

## Mass

- Low
- Medium
- High
- Very High

[gartner.com](https://www.gartner.com)

Source: Gartner

© 2021 Gartner, Inc. and/or its affiliates. All rights reserved. CTMKT\_1574277

**Gartner**

## Free Websites to practice python

Here are the top Free Websites to practice coding (all programming languages) 🖥️ :

Thread 🗨️ 1 of 6

🗨️ 2 of 6 - Hacker Rank

<https://t.co/0iZS51F2HW>

📖 3 of 6 - Leet Code

<https://t.co/J9NpnwLuDY>

📖 4 of 6 - Code Chef

<https://t.co/eENshpZZAG>

📖 5 of 6 - Code Wars

<https://t.co/kVRXjvZXDZ>

📖 6 of 6 - Coderbyte

<https://t.co/DDDOHCkQ6Y>

Bonus: Exercism

<https://t.co/GNveWgyhD2>

## Understanding the Frisch-Waugh-Lovell Theorem

<https://towardsdatascience.com/the-fwl-theorem-or-how-to-make-all-regressions-intuitive-59f801eb3299>

## 5 Data Science and Machine Learning competitions

5 Data Science and Machine Learning competitions to get real-world experience and earn \$\$\$

Hurry up to participate!

(A thread) 📌 📖

1. House Price Prediction by @BowTied\_Raptor

Kaggle's most popular house price prediction is a great way for beginners to get hands-on data science experience.

Prize Money - \$1,000

👉 <https://t.co/Po4XAC3hbJ>

2. Image Classification by @MayoClinic

Build a machine learning model to classify the blood clot origins from digital pathology images.

Prize Money - \$3,000

👉 <https://t.co/BL68V9KYrb>

3. Stock price prediction by Japanese Exchange Group

Build machine learning models to predict future stock prices based on stock information metadata and

historical stock prices.

Prize Money - \$63,000

👉 <https://t.co/PoYHJ6d0lo>

4. Big Data Derby 2022 by @TheNYRA

Analyze horse racing data to improve the health of the horse and the strategy/tactics in the competition.

Prize Money - \$50,000

👉 <https://t.co/SWRnz50e6J>

5. Scene Detection by DFL

Build a computer vision model to automatically classify important events from long video recordings.

Prize Money - \$25,000

👉 <https://t.co/tJ7ZoNxmGK>

## **The challenges of adversarial machine learning in constrained-feature applications**

<https://bdtechtalks.com/2022/09/19/adversarial-machine-learning-constrained-features/>

## **Hardware for Machine Learning spring 2021**

Video lectures, UC Berkeley EE 290-2 Hardware for Machine Learning spring 2021, by Sophia Shao  
<https://t.co/gvaqsXhZtE>

Cool! There is more from her (with recordings), such as <https://t.co/44Y3h7P7pz> and <https://t.co/3VloyoT6xM> which may be possible pre-reqs for the EE 290 course.

Video lectures, CMU 10 708 Probabilistic Graphical Models spring 2017, 2019, 2020 by Eric Xing  
<https://t.co/oBJe2rIkI>  
<https://t.co/V6JFmPxRC9>  
<https://t.co/qyIMEavNuv>

Also from CMU... going through the links you've posted... found these -> <https://t.co/H8Z3MkJxAa> deep reinforcement learning

<https://t.co/swb5dRJWX3> 10-414/714: Deep Learning Systems

<https://t.co/37HhRsYCay> Introduction to Machine Learning

Convex Optimization: Fall 2018  
<https://t.co/qBYsD0Zb5p>

Another deep reinforcement learning one  
<https://t.co/8lEQ3OAK8>

Deep Reinforcement Learning  
10-403 • Spring 2021 • Carnegie Mellon University

## FindIt: Generalized Object Localization with Natural Language Queries

<http://ai.googleblog.com/2022/09/findit-generalized-object-localization.html>

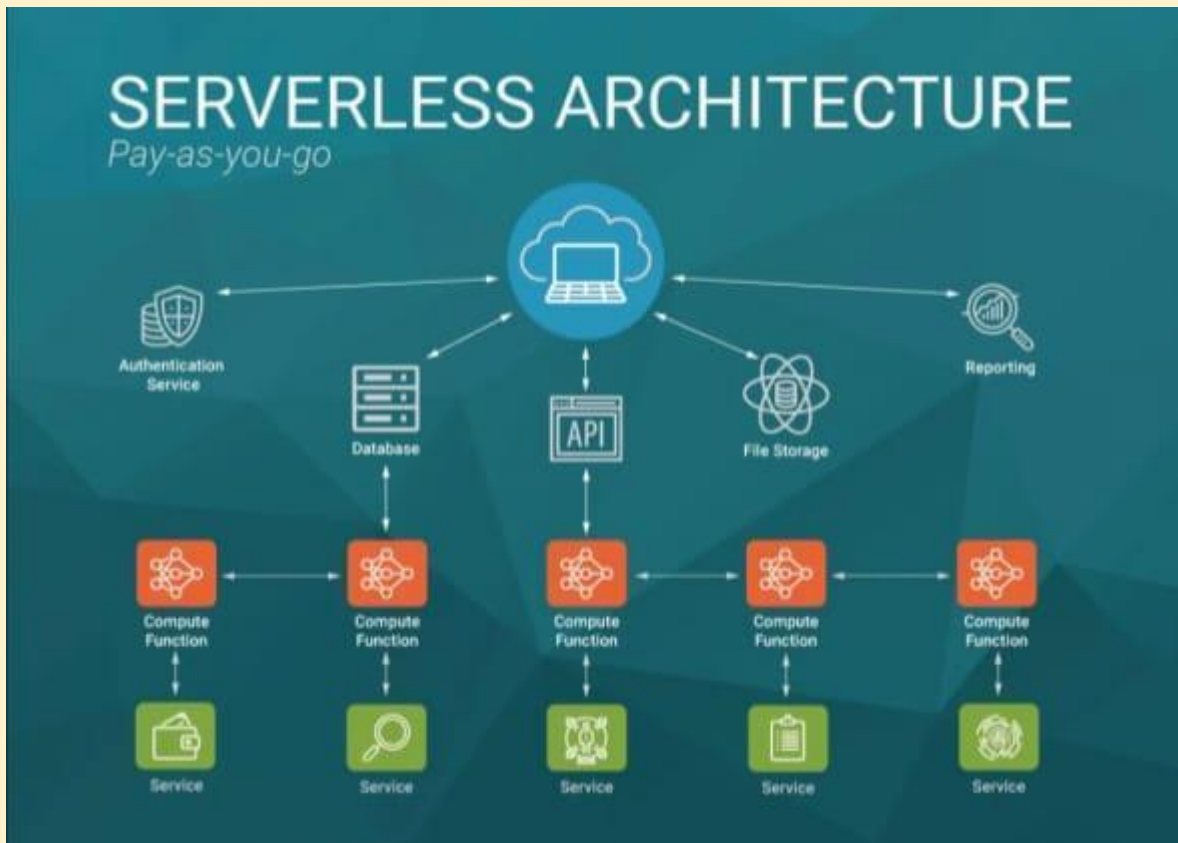
Natural language enables flexible descriptive image queries. The interplay between textual queries and images grounds linguistic meaning in the visual world, facilitating better understanding of relationships between objects, human intentions toward objects, and interactions with the environment. The research community has investigated object-level visual grounding through a range of tasks including referential understanding of the expression, text-based localization and wider object detection, each requiring different skills in the model. For example, object detection seeks to find all objects from a predefined set of classes, which requires accurate localization and classification, while referential expression understanding locates an object from a referencing text and often requires complex reasoning over salient objects. At the intersection of the two is text-based localization, where a simple category-based text query prompts the model to find the objects of interest.

Due to their different task properties, comprehension, referential expression detection, and text-based localization are mostly studied through separate benchmark tests, with most models designed for only one task. As a result, existing models have not adequately synthesized information from the three tasks to achieve a more holistic visual and linguistic understanding. Referential understanding models, for example, are trained to predict a single object in an image and often struggle to localize multiple objects, reject negative queries, or discover new categories. In addition, detection models are unable to handle text input, and text-based localization models often struggle to handle complex queries that refer to a single instance of an object, such as “Left Half Sandwich”. Finally, none of the models generalize well enough beyond their data and training categories

To address these limitations, researchers present “FindIt: Generic localization with natural language queries” at ECCV 2022. Here, they propose a unified, general-purpose, multitasking visual grounding model called FindIt, which can flexibly respond to different types of grounding and detection requests. The key to this architecture is a multi-level cross-modal fusion module that can perform complex reasoning to refer to expression understanding and simultaneously recognize small and challenging objects for text-based localization and detection. Furthermore, they find that the standard object detector and detection losses are sufficient and surprisingly efficient for all three tasks, without the need for task-specific design and losses common in existing works. Read more in paper.

#AI #MachineLearning #IoT #IIoT #CX #SelfDrivingCars #SmartCity #AutonomousVehicles #DataScience #BigData #Analytics #FindIt

## Serverless architecture



## Metaverse trends by Gartner

Gartner suspects that by 2026, 25% of all people will spend at least one hour per day in the metaverse for shopping, work, education, social, or entertainment.

This expected increase in usage, coupled with more advanced virtual reality (VR) environments and virtual avatar creation, has driven many brands to explore the sale of virtual goods. #VR

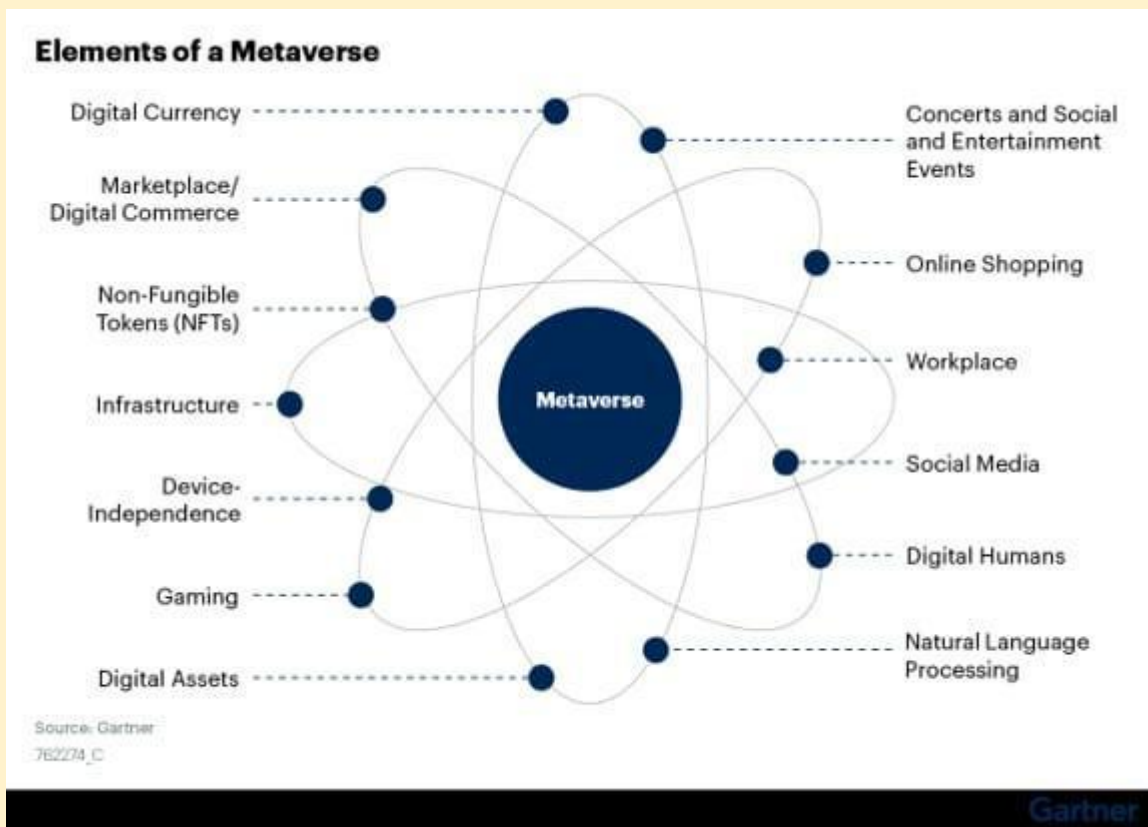
Check: <https://www.simon-kucher.com/en/metaverse/business-models/bridging-gap-between-physical-virtual-fashion>

Metaverse Implications for Retail Technology and Service Providers

<https://blogs.gartner.com/robert-hetu/metaverse-implications-for-retail-technology-and-service-providers/>



## Elements of metaverse



## How I prepared for DeepMind and Google AI research

internship interviews in 2019 • David Stutz

<https://davidstutz.de/how-i-prepared-for-deepmind-and-google-ai-research-internship-interviews-in-2019/>

## 38 Free Courses on Coursera for Data Science You Must Know in 2022

<https://www.mltut.com/free-courses-on-coursera-for-data-science/>

## Use machine learning to win the market.

How successful businesses use machine learning to win the market?

### 1. ML-driven customer segmentation

Rather than guesswork, use ML algorithms to segment customers based on their preferences, behavior, etc.

Two commonly used ML algorithms for customer segmentation are:

- K Means Clustering
- DBSCAN

### 2. Demand Forecasting

Making products available at the right place at the right time is crucial to any business. Where traditional methods would lack, ML models can identify hidden patterns to precisely forecast the demand.

ML algorithms that can be used:

- XGBoost
- LightGBM

### 3. Personalized Product Recommendation

ML algorithms can identify patterns in previous search history, location, etc for each customer to recommend relevant product.

ML algorithms that can be used:

- Collaborative Filtering & Matrix Factorization
- Deep Neural Networks

### 4. Chatbot that talk like human

AI powered chatbots can enhance user experience by curating conversations that feel personal and empathetic.

Models that can be used:

- GPT-3 by OpenAI
- OPT by Meta AI
- GPT-J by AI21 Labs

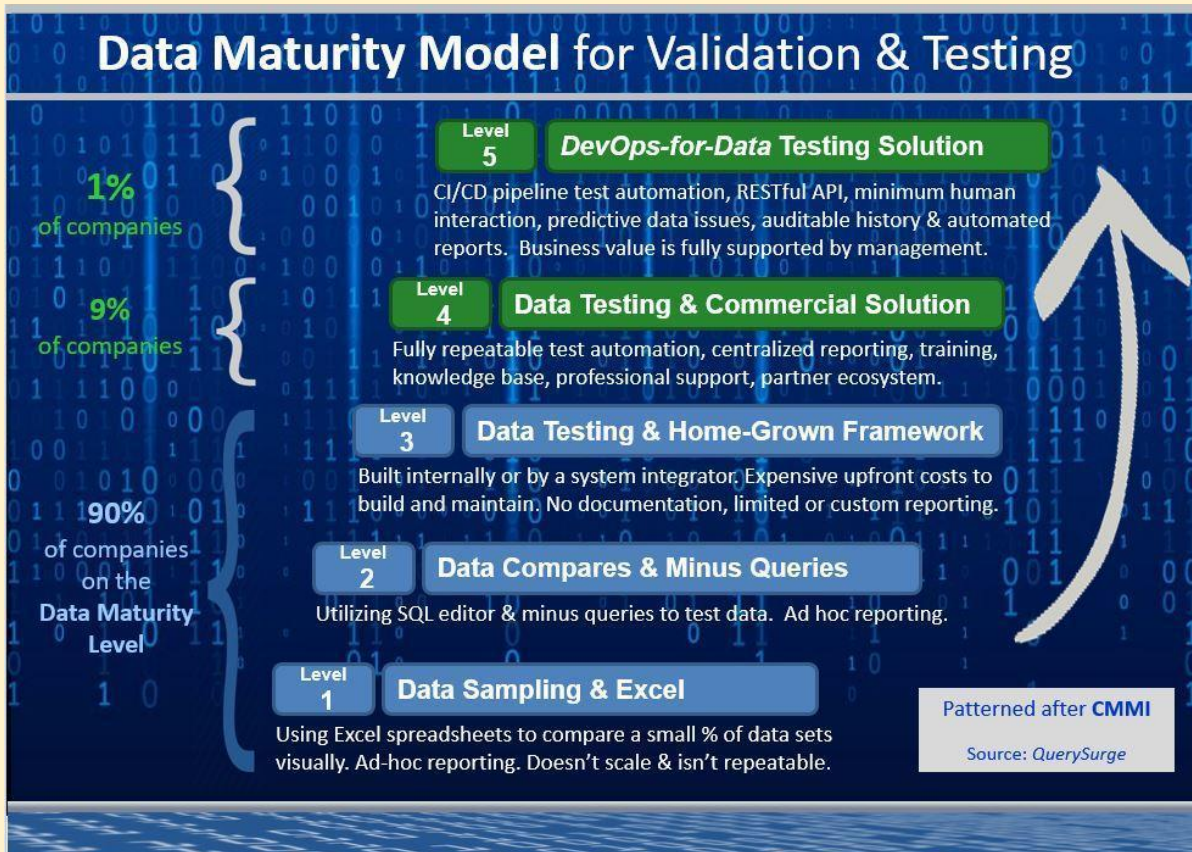
### 5. Marketing on Autopilot

AI can help you attract customers with limited manual inputs. Generate creative text with clear and concise messaging on the fly with the latest text generation models.

Models that can be used:

- GPT-3 by OpenAI
- OPT by Meta AI

## Data Maturity Model




# AR vs VR

## Virtual Reality (VR)

- VR replaces the real world scenarios with a simulated one
- You can't see real world around you everything is virtual
- VR is delivered through Head Mounted Display such as Oculus VR, playstation VR

## Augmented Reality (AR)

- In AR , you can see the real world and interact with it
- AR Enhances user's perception by using the existing environment
- AR is delivered through smartphones such as GPS and gesture recognition systems



### Healthcare

Used for product Simulation, pain Distraction, Diagnostics and More

### Education

Students watch 360-degrees Videos and learn through immersive in 3D environment

### Automobile

Engineers work in virtual environment to Test and Analyze the efficiency, safety, durability and other features of Vehicles

### Healthcare

Used in areas such as surgery training, anatomy knowledge, rehabilitation and more

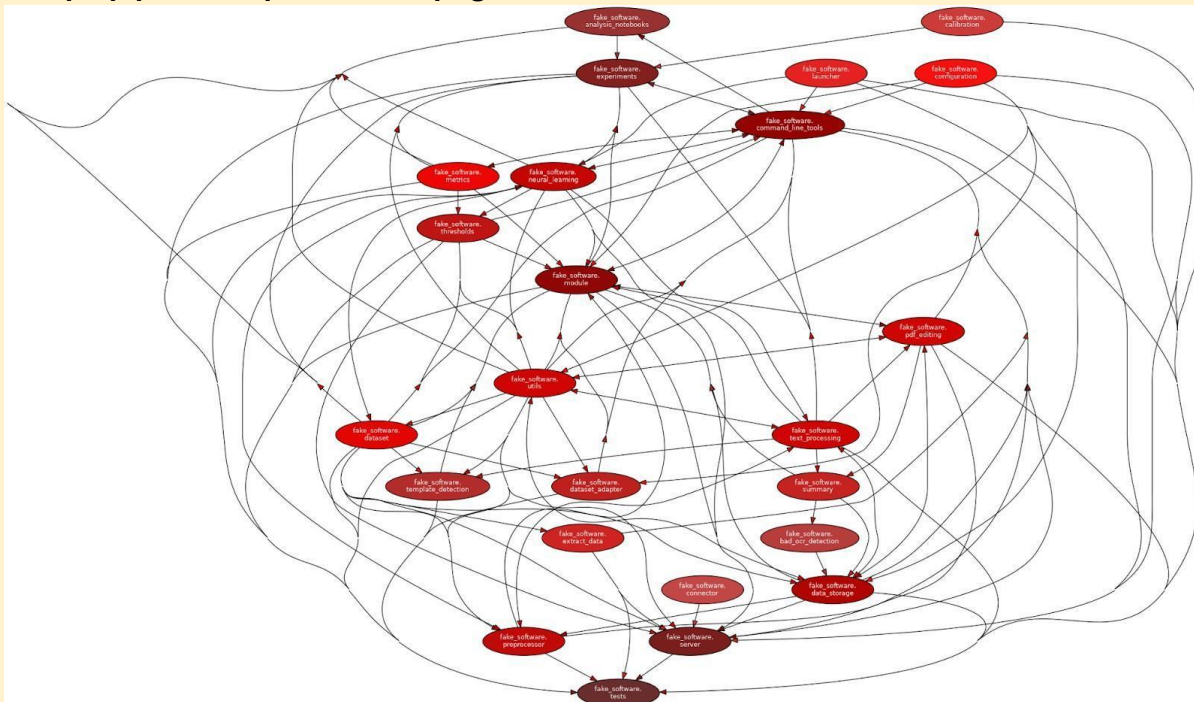
### Education

Students use interactive 3D models to learn diverse topics

### Automobile

AR offers Drivers the opportunity to experience a test drive-without actually setting foot in a car

MLOps-pipeline-dependencies.png





## Building-mlops-pipeline-problems.jpg

### Building MLOps Pipeline: 6 Problems & Solutions

| Problem                       | Solution               | Tools that helped                                            |
|-------------------------------|------------------------|--------------------------------------------------------------|
| 01 POC Code                   | Production code        | Typer, FastAPI                                               |
| 02 Entangled ML monolith      | ML Microservices       | Docker, Docker Compose                                       |
| 03 Low separation of concerns | Good code architecture | PyTorch, PyTorch Lightning, Scikit-learn, Optuna, Neptune.ai |
| 04 Dynamic configuration      | Good typed data model  | Pydantic                                                     |
| 05 Model roasting             | Automatic retraining   | Cron, Latex, Plotly, tmux                                    |
| 06 Low code quality           | High code quality      | Pylint, my[py], unittest, Sourcery, pydeps, Vulture, GitLab  |


 neptune.ai


## AI/ML Tools and Categories


 Design  
Product  
Engineering  
Project


 Data


- Labeling
- Exploration
- Preprocessing
- Splitting
- Augmentation


-  Modeling
- Baselines
- Evaluation
- Experiment tracking
- Optimization

-  Developing
- Packaging
- Organization
- Logging
- Documentation
- Styling
- Makefile

-  Serving
- Command-line
- RESTful API

-  Testing
- Code
- Data
- Models

-  Reproducibility
- Git
- Pre-commit
- Versioning
- Docker

-  Production
- Dashboard
- CI/CD
- Monitoring
- Systems design

- ✳ Data Engineering
- Data stack
- Orchestration
- Feature store

## free resources for learning Python and Data Science

Bookmark these 10 repositories to guarantee you learn from the best.

1. Machine Learning Complete (by @Jeande\_d)

Foundational Python, followed by every significant application of Machine Learning and Deep Learning.

Follow Jeande's course (based on his own experience) for practical learning for any experience level.

<https://t.co/tPHZMfVRVY>



## 2. Made with ML (by @GokuMohandas)

Foundations in coding and modeling through to deployment in a customer-facing application.

Goku's perfect course flow will take you from 0 to MLOps Engineer at the perfect level of difficulty.

<https://t.co/dFiwymFvO3>

## 3. freeCodeCamp

The premier platform for learning how to code; and they've added resources on stats and modeling.

Ignore the abundant resources for other coding languages and go deep with Python and ML.

<https://t.co/iPdimBzXGM>

## 4. Homemade Machine Learning

Popular machine learning algorithms implemented in Python with explanations of the underlying math.

Trust this resource based on lessons from @AndrewYNg - premier data science instructor and researcher.

<https://t.co/Yxe1e7zrE4>

## 5. Data Science IPython Notebooks

Notebook-based tutorials of every major Python library used for data science.

Perfect way to get a crash course in one library before using it your work.

<https://t.co/Q7LzCeFhVo>

## 6. Deep Learning Drizzle

Over 300 free YouTube lectures in Deep Learning, Reinforcement Learning, Machine Learning, Computer Vision, and NLP.

When you're ready to specialize, go deep and become an expert with these videos.

<https://t.co/0wEE6CePJU>

## 7. 500 AI-ML Projects

Over 500 projects in Deep Learning, Reinforcement Learning, Machine Learning, Computer Vision, and NLP.

Bookmark for inspiration and instruction with your own projects - this repo comes with code and is updated continuously.

<https://t.co/92B2xxk3jS>

## 8. Data Science Cheatsheets

70+ comprehensive cheatsheets in Python, SQL, R, math, business intuition.

You can't learn everything here, but bookmark this as a resource for your learning.

<https://t.co/bLuvW9pylQ>

## 9. Awesome Machine Learning

A curated list of awesome machine learning frameworks, libraries and software.

Use this as home base to discover resources for the next topic you want to learn.

<https://t.co/5W2ZVuOgbG>

10. Awesome Data Science

An open source data science repository to learn and apply towards real-world problems.

Use this as a list of courses and learning material to start learning data science.

<https://t.co/P2BkX7FS1n>

## **Best System design resources**

1. Modern System design - <https://t.co/6CBu13zkLC>
2. Grokking the System Design - <https://t.co/OxR5eCWFJk>
3. Software Architecture 101 - <https://t.co/GudhicBTUF>
4. System Design Cheat sheet
5. more - <https://t.co/VM0Kk2OMvy>

# Building blocks of Data Science Project

1. Requirements
2. Estimation
3. Storage schema (optional)
4. High-level design
5. APIs
6. Detailed design
7. Evaluation
8. Distinctive component/feature

## Building Blocks Glossary:

**Domain Name System:** Maps domain names to IP addresses.

**Load Balancers:** Distributes client requests among servers.

**Databases:** Stores, retrieves, modifies, & deletes data.

**Key-Value Store:** Stores data as key-value pairs.

**Content Delivery Network:** Distributes in-demand content to end users.

**Sequencer:** Generates unique IDs for events & database entries.

**Service Monitoring:** Analyzes system for failures & sends alerts.

**Distributed Caching:** Stores frequently accessed data.

**Distributed Messaging Queue:** Decouples messaging producers from consumers.

**Publish-Subscribe System:** Supports asynchronous service-to-service communication.

**Rate Limiter:** Throttles incoming requests for services.

**Blob Store:** Stores unstructured data.

**Distributed Search:** Returns relevant content for user queries.

**Distributed Logging:** Enables services to log events.

**Distributed Task Scheduling:** Allocates resources to tasks.

**Sharded Counters:** Counts concurrent read/write requests.

**Step 1: Requirements**  
Gather functional & non-functional requirements

**Consider:**

- System goals
- Key features
- System constraints
- User expectations

**Step 2: Estimation**  
Estimate hardware & infrastructure needed to implement at scale

**Consider requirements for:**

- Number of servers
- Daily storage
- Network

**Step 3: Storage schema (optional)\***  
Articulate data model

**Define:**

- Structure of data
- Tables to use
- Type of fields in tables
- Relationship between tables (optional)

**\*Relevant when you:**

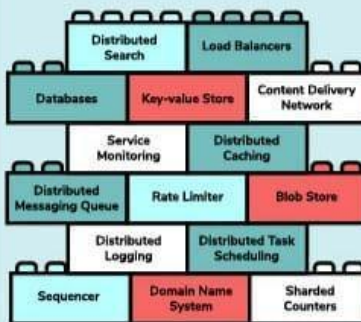
- Expect highly normalized data
- Will store different parts of data in various formats
- Face performance & efficiency concerns around storage

**Step 4: High-level design**

- Build high-level design
- Choose building blocks to meet functional requirements

**For each, identify:**

- **How** they work
- **Why** they're needed
- **How** they integrate



This layered visual shows dependencies between building blocks. Blocks in lower layers support those above.

**Step 5: APIs**  
Translate functional requirements into API calls

**E.g.:**

- **Requirement:** Users should be able to access all items
- **API call:** GET / items

**Step 6: Detailed design**

- Improve high-level design
- Consider all non-functional requirements & complete design

**Step 7: Evaluation**

- Evaluate design against requirements
- Explain trade offs & pros/cons of different solutions
- Address overlooked design problems

**(8\*) Distinctive component/feature**  
Discuss a distinctive feature that meets requirements

- E.g. Concurrency control in high-traffic apps

\* Timing varies. Best done after completing design (E.g. Step 6 & 7)

## Jaied AI: EasyOCR documentation

<https://www.jaied.ai/easyocr/documentation/>

Video lectures, UCLA Stats 131 Introduction to Probability Theory 2018, 2019 by Miles Chen  
<https://t.co/sMX6QidU7B>

Video lectures, UCLA Stats 15 Introduction to Data Science fall 2021, by Miles Chen  
<https://t.co/cPUk8YLSdp>

Video lectures, UCLA Stats 101C Introduction to Statistical Models and Data Mining summer 2019, by Miles Chen  
<https://t.co/eO8Oxk4sUK>

Video lectures, UCLA Stats 101A Introduction to Data Analysis and Regression fall 2015, by Miles Chen  
<https://t.co/i76suo2Wq>

Video lectures, UCLA Stats 231C Theories of Machine Learning spring 2022, by Arash Amini

<https://t.co/8ucUS0p8eQ>

Video lectures, UCLA Stats 200C High-dimensional Statistics spring 2021, by Arash Amini  
<https://t.co/qAmasdDYHw>

Video lectures, UCLA Stats 200B Theoretical Statistics winter 2022, by Arash Amini  
<https://t.co/AfUeGRfeJT>

Video lectures, UCLA Stats 100C Linear Models spring 2021, winter 2022, by Arash Amini  
<https://t.co/2cKvLpfaJL>

Video lectures, UCLA Stats 102C Monte Carlo Methods fall 2020, by Miles Chen  
<https://t.co/6xhIOKvLBZ>

Video lectures, UCLA Stats 102B Computation and Optimization for Statistics spring 2021, by Miles Chen  
<https://t.co/ueykT1wJfm>

Video lectures, UCLA Stats 102A Introduction to Computational Statistics with R winter 2021, by Miles Chen  
<https://t.co/14K7Mnik0D>

Video lectures, UCLA Stats 21 Python and Other Technologies for Data Science spring 2021, by Miles Chen  
<https://t.co/k6bUphcwTR>

Video lectures, UCLA Stats 13 Introductory Statistics summer 2021, by Miles Chen  
<https://t.co/2QhTxiqKOY>

## **Developer-Y/cs-video-courses: List of Computer Science courses with video lectures.**

<https://github.com/Developer-Y/cs-video-courses>

## **Google AI Blog: Identifying Disfluencies in Natural Speech**

<http://ai.googleblog.com/2022/06/identifying-disfluencies-in-natural.html>

## **Mastery in coding for data science is daily practice.**

The best way to reach mastery in coding for data science is with daily practice.

Use these 10 resources for unlimited exercises (SQL, Python, pandas, ML) so you can become a master.

### 1. Mode (SQL)

Learn SQL from the ground up by writing queries on real data.

Bonus: Mode is an analytics platform and has tutorials you can follow for complex projects!

<https://t.co/3yNgtGFOGu>

### 2. StrataScratch (SQL)

Keep your SQL skills fresh with real interview questions.

Bonus: Practice case studies and statistics questions too!

<https://t.co/ICY5dHkRdI>

### 3.HackerRank (Python)

Learn Python with practical exercises, not dense textbook material.

Bonus: Get a certification FAST, and start building your online reputation!

<https://t.co/RAPyaoMeLp>

### 4.Codewars (Python)

Keep your Python skills fresh with interview-style questions.

Bonus: Gamified learning for added excitement

<https://t.co/eD0HG9H1ux>

### 5.LeetCode (Python)

Level up your algorithmic problem solving in Python with difficult problems.

Bonus: These are tough - you're on the road to a future in Software or ML Engineering!

<https://t.co/q69KzNcS3E>

### 6.GeeksForGeeks (pandas)

When it comes to data, Python is nothing without pandas. Learn it with practical exercises.

Bonus: GeeksForGeeks has answers to all your common coding bugs

<https://t.co/C7jNY5zEw2>

### 7.Machine Learning Plus (pandas)

More pandas exercises - you can't have enough.

Bonus: Valuable articles and resources for learning more about Python and ML.

<https://t.co/WDV5SO4ufr>

### 8.W3 Resource (pandas)

Did I mention how important pandas is?

Bonus: Lots more tutorials and projects for Python and SQL!

<https://t.co/KXQPB7CXu0>

### 9.Kaggle (pandas)

Level up with all the most common operations needed in data science.

Bonus: Dive right into a project or competition with real data when you're done!

<https://t.co/a3U9bXA3Eh>



## 10. Binomial (Machine Learning)

1 machine learning question to your inbox every day, complete with explanation and resources for further reading.

<https://t.co/35iyX9fwib>

## Research Paper Format

### Title

The title is self-explanatory and the easiest element to identify.

### Abstract

The abstract is unique to journal articles. This is a brief, one paragraph overview of the article. The abstract is designed to help you to know whether the journal article is valuable for you to read or not. The abstract is usually on the first page of the piece, always before the introduction. You are likely to find it indented underneath the title.

### Introduction

The introduction also gives an overview of the piece. While the abstract explains the key arguments of the piece, the introduction is usually used to give context and a rationale to the piece. It usually also signposts what will be said in each section of the article.

### Literature Review /Theoretical Framework

The literature review summarizes and analyses previous research on the topic that the author is reporting on. This is an important section for showing readers what is already known about the topic, and how the study expands on existing knowledge by introducing new ideas and information that hasn't been discovered before.

### Methodology

Methodologies are important for providing the 'recipe' on how the research took place. The methodology answers the question 'How did you reach your conclusions?' This is what distinguishes research from hypothesis. Logic and reason are required in order to establish knowledge within academic research. The methodology should therefore demonstrate the logic used to reach the conclusion (Drew, Hardman & Hosp, 2008, p. 18).

### Findings

The findings section outlines what the study found out. It is often separated into themes, outlining some of the key ideas that have emerged out of the research project. This is where you finally find out what the authors actually discovered through their project. It is usually quite detailed, taking readers through the findings step-by-step.

### Discussion / Conclusion

The conclusion will sum up the findings and explain how and why the findings are important, relevant or interesting. It may explain what the implications of the findings are for practitioners, policy makers or researchers.

## **Computer Vision Research at Google and Boston University Proposes 'DreamBooth,'**

**A Technique for Fine-Tuning a Text-to-Image Model with a very Limited Set of Images - MarkTechPost**

<https://www.marktechpost.com/2022/09/14/latest-computer-vision-research-at-google-and-boston-university-proposes-dreambooth-a-technique-for-fine-tuning-a-text-to-image-model-with-a-very-limited-set-of-images/>

## **3 Free Data science books for reference and learning.**

1. Fundamentals of Data Visualization by @ClausWilke  
<https://t.co/jH2rVz7kSp>

2. Python Data Science Handbook by @jakevdp  
<https://t.co/MIvnJQw7Rz>

3. Hands-On Data Visualization by @HandsOnDataViz  
<https://t.co/qhSaEzxVrn>

## **Google sheet in Python Pandas**

```
"""
Link Google Sheets to Python directly
"""

import gspread
import pandas as pd

# Use gspread for authentication
gc = gspread.oauth()

# Open a document by URL
gsheets_url = "https://docs.google.com/spreadsheets/d/sampleid/"
doc = gc.open_by_url(gsheets_url)

# Open the Tab named "Friends" and store in pandas
sheet = doc.worksheet("Friends")
pd.DataFrame(sheet.get_all_records())

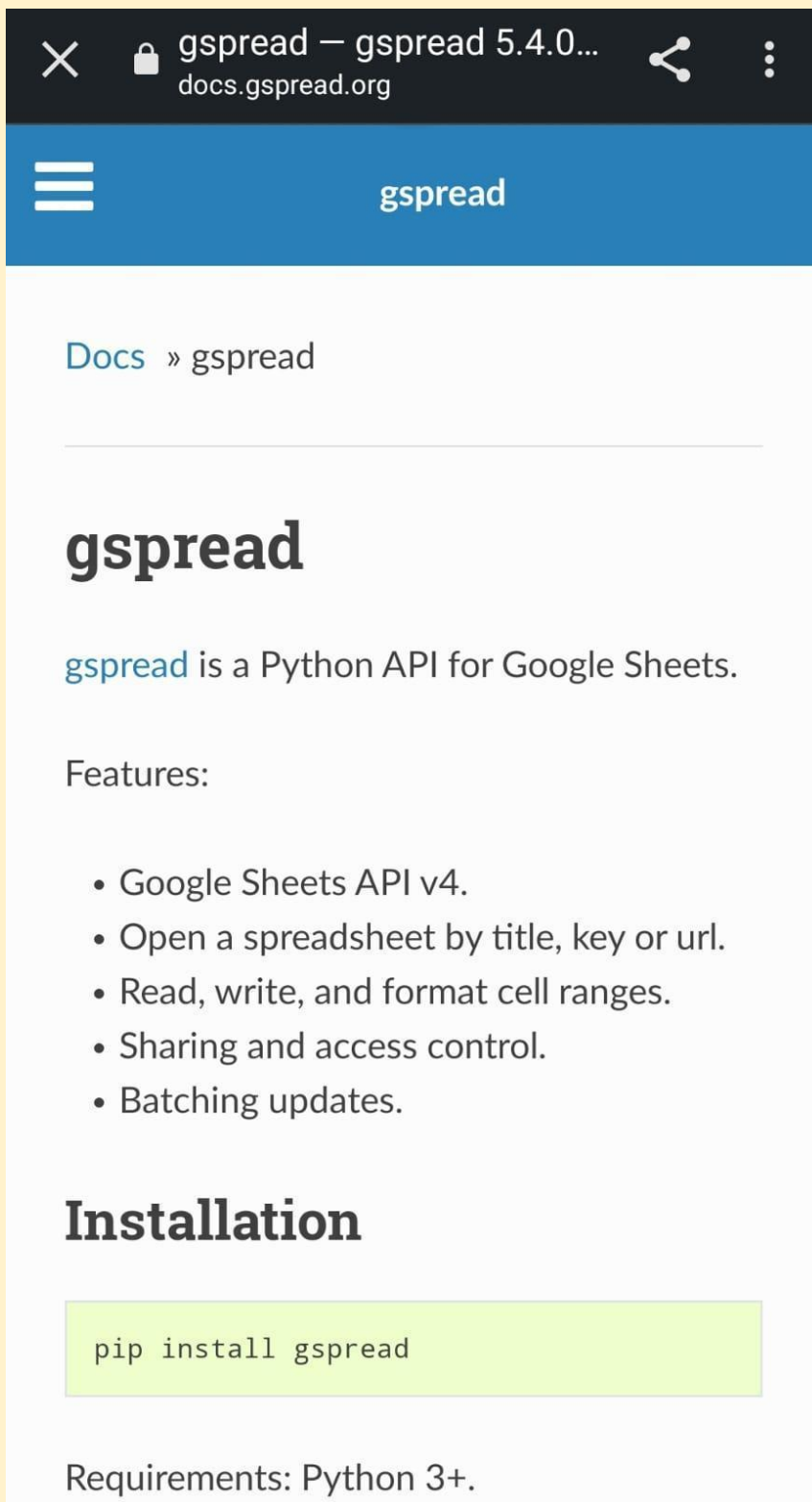
>>>   Name  Age
>>> Rachel  23
>>> Phoebe  24
>>>   Ross  25
>>> Monica  23

# Create a dataframe and update a sheet
data = {'Name': ['Michael', 'Jim', 'Pam'],
        'Age': [40, 30, 29]}
df = pd.DataFrame.from_dict(data)

# Update the sheet
worksheet.update([df.columns.values.tolist()] + \
                 df.values.tolist())

# Uploaded by Travis Tang
```

## gsread library



The screenshot shows a mobile browser view of the gsread library documentation. The address bar at the top displays 'gsread — gsread 5.4.0...' and 'docs.gsread.org'. Below the address bar is a blue header with a hamburger menu icon on the left and the text 'gsread' in the center. The main content area has a breadcrumb 'Docs » gsread' followed by a horizontal line. The title 'gsread' is displayed in a large, bold font. Below the title, a paragraph states 'gsread is a Python API for Google Sheets.' This is followed by a section titled 'Features:' which contains a bulleted list of five items: 'Google Sheets API v4.', 'Open a spreadsheet by title, key or url.', 'Read, write, and format cell ranges.', 'Sharing and access control.', and 'Batching updates.' Below the features is a section titled 'Installation' which contains a code block with the command 'pip install gsread'. At the bottom of the page, the text 'Requirements: Python 3+' is displayed.

gsread — gsread 5.4.0...  
docs.gsread.org

gsread

Docs » gsread

# gsread

gsread is a Python API for Google Sheets.

Features:

- Google Sheets API v4.
- Open a spreadsheet by title, key or url.
- Read, write, and format cell ranges.
- Sharing and access control.
- Batching updates.

## Installation

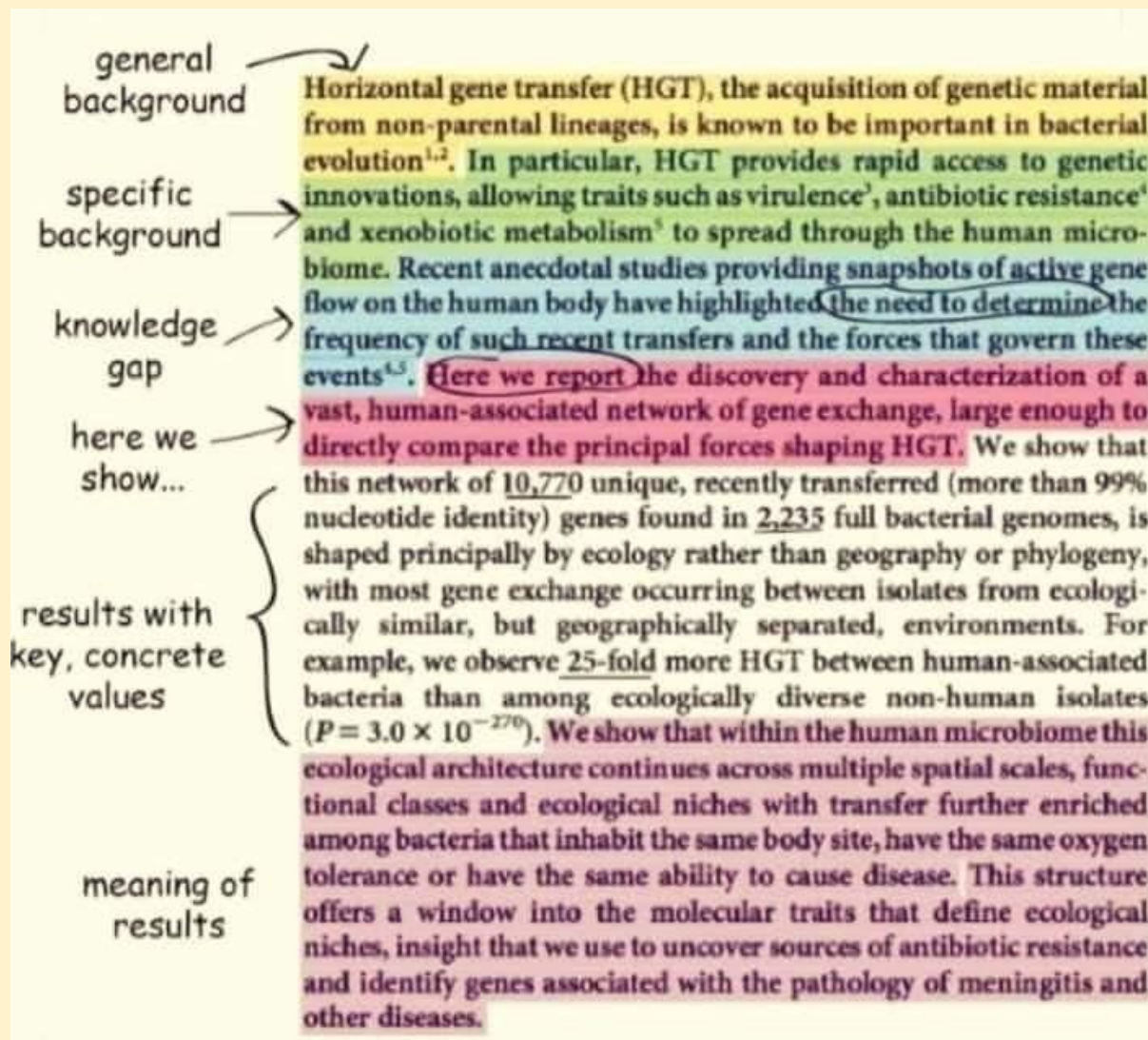
```
pip install gsread
```

Requirements: Python 3+.

## Python vs Ruby

<https://www.guru99.com/python-vs-ruby-difference.html#:~:text=Ruby%20is%20a%20server%2Dside,like%20Perl%20and%20C%2B%2B.>

## How to Read Research Abstract



## Cost of AI Project and Estimating the Delivery Time - Addepto

<https://addepto.com/blog/estimating-the-delivery-time-and-the-cost-of-an-ai-project/#>

## Top 10 Open-Source Cybersecurity Project Ideas for Beginners

<https://www.analyticsinsight.net/top-10-open-source-cybersecurity-project-ideas-for-beginners/>

## Top 10 Datasets For Cybersecurity Projects One Must Know

<https://analyticsindiamag.com/top-10-datasets-for-cybersecurity-projects/>

## Top 7 cybersecurity projects for beginners - Great Learning

<https://www.mygreatlearning.com/blog/top-cybersecurity-projects-for-beginners/>



## **[jivoi/awesome-ml-for-cybersecurity: Machine Learning for Cyber Security](https://github.com/jivoi/awesome-ml-for-cybersecurity)**

<https://github.com/jivoi/awesome-ml-for-cybersecurity>

## **Sipeed's TinyMaix Puts MNIST Digit Recognition on a Modest Microchip ATmega328 Microcontroller**

Open source project, written during a hackathon weekend, adds INT8 and FP32 machine learning model support to low-end microcontrollers.

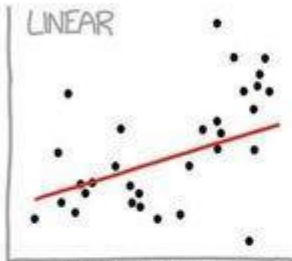
Sipeed is aiming to bring machine learning inference capabilities to even low-end microcontrollers with TinyMaix, written over a weekend — and has proven its capabilities by running an MNIST handwritten digit recognition model on a modest Microchip ATmega328 chip.

"TinyMaix is designed for running AI neural network models on resources limited MCUs [Microcontroller Units], which [is] usually called tinyML," the company explains of the software. "TinyMaix is a weekend hackathon's project, so it is simple enough to read though in 30 minutes and it will help tinyML newbies to understand how is it running."

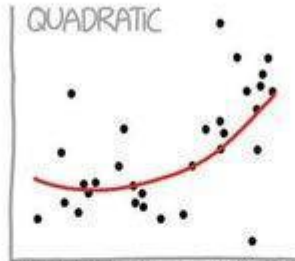
<https://www.hackster.io/news/sipeed-s-tinymaix-puts-mnist-digit-recognition-on-a-modest-microchip-atmega-328-microcontroller-9884bc3cc57c>

## Model Fitting and Line of Regression

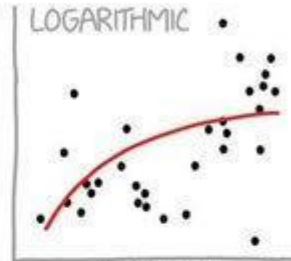
### CURVE-FITTING METHODS AND THE MESSAGES THEY SEND



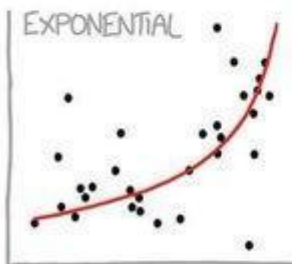
"HEY, I DID A  
REGRESSION."



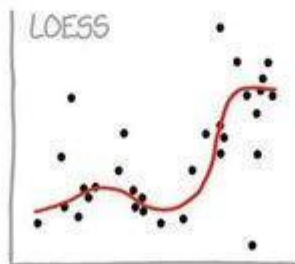
"I WANTED A CURVED  
LINE, SO I MADE ONE  
WITH MATH."



"LOOK, IT'S  
TAPERING OFF!"



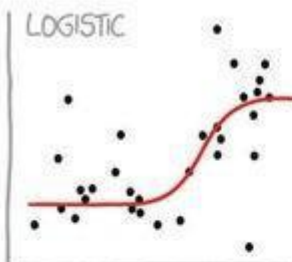
"LOOK, IT'S GROWING  
UNCONTROLLABLY!"



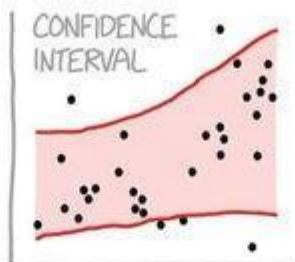
"I'M SOPHISTICATED, NOT  
LIKE THOSE BUMBLING  
POLYNOMIAL PEOPLE."



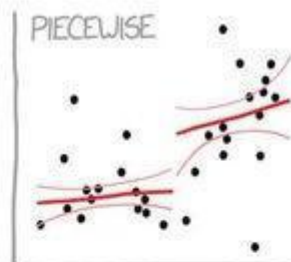
"I'M MAKING A  
SCATTER PLOT BUT  
I DON'T WANT TO."



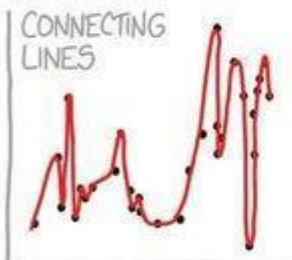
"I NEED TO CONNECT THESE  
TWO LINES, BUT MY FIRST IDEA  
DIDN'T HAVE ENOUGH MATH."



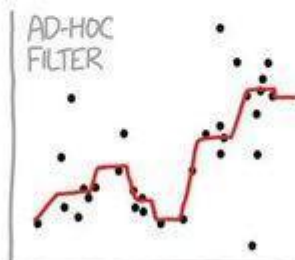
"LISTEN, SCIENCE IS HARD,  
BUT I'M A SERIOUS  
PERSON DOING MY BEST."



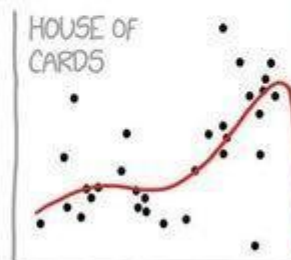
"I HAVE A THEORY,  
AND THIS IS THE ONLY  
DATA I COULD FIND."



"I CLICKED 'SMOOTH  
LINES' IN EXCEL."



"I HAD AN IDEA FOR HOW  
TO CLEAN UP THE DATA.  
WHAT DO YOU THINK?"



"AS YOU CAN SEE, THIS  
MODEL SMOOTHLY FITS  
THE- WAIT NO NO DON'T  
EXTEND IT AAAAAA!!!"

<https://github.com/CalculatedContent/WeightWatcher>

## **Data Visualization Types: Everything a Marketer Needs to Know - TapClicks**

<https://www.tapclicks.com/resources/blog/data-visualization-types/>

## **Bitcoin Prediction - Using FBProphet**

[https://m.youtube.com/watch?v=0gXeXtL\\_KjY&list=PLxqBkZuBynVS8mDTc8ZGermXiS-32pR2y&index=16](https://m.youtube.com/watch?v=0gXeXtL_KjY&list=PLxqBkZuBynVS8mDTc8ZGermXiS-32pR2y&index=16)

## **How to Generate Images from Text with Stable Diffusion Models**

<https://link.medium.com/82GopoMlctb>

Stability AI Open-Sources Image Generation Model Stable Diffusion

<https://www.infoq.com/news/2022/09/stable-diffusion-image-gen/>

Github:

<https://github.com/CompVis/stable-diffusion>

Try executing model by entering text :

<https://huggingface.co/spaces/stabilityai/stable-diffusion>

Stability AI Open-Sources Stable Diffusion: An Artificial Intelligence Model That Converts Text To Image

<https://www.marktechpost.com/2022/09/07/stability-ai-open-sources-stable-diffusion-an-artificial-intelligence-model-that-converts-text-to-image/>

References:

<https://analyticsindiamag.com/diffusion-models-from-art-to-state-of-the-art/>

[https://en.wikipedia.org/wiki/Stable\\_Diffusion](https://en.wikipedia.org/wiki/Stable_Diffusion)

Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation

<https://paperswithcode.com/paper/dreambooth-fine-tuning-text-to-image>

## **5 Exploding Areas Of The Metaverse That May Not Be On Your Radar Yet**

<https://www.forbes.com/sites/michellegreenwald/2022/09/06/5-exploding-areas-of-the-metaverse-that-may-not-yet-be-on-your-radar/>

## **Top free Books to Learn Data Science**

1- Data Science at the Command Line by Jeroen Janssens: <https://lnkd.in/gbjdkW9M>

2- Deep Learning on Graphs by Yao Ma and Jiliang Tang: <https://lnkd.in/g3g-puib>

- 3- Hands-on Machine Learning with Scikit-learn, Keras and Tensorflow by Aurelien Geron: <https://lnkd.in/gzeASHUd>
- 4- Practical Statistics for Data Science by Peter Bruce & Andrew Bruce <https://lnkd.in/gfUUfb6K>
- 5-An Introduction to Statistical Learning by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani: <https://lnkd.in/eBCkgBS>
- 6-Learning Deep Architectures for AI by Yoshua Bengio: <https://lnkd.in/gHNKMzE2>
- 7- Python for Data Science Handbook by Jake VanderPlas: <https://lnkd.in/bxTAdNY>
- 8- The Hundred-Page Machine Learning Book by Andriy Burkov: <https://lnkd.in/gdbbUuPH>
- 9- A Course in Machine Learning by Hal Daume III: <https://lnkd.in/gDr2C7qi>
- 10- Intuitive ML and Big Data in C++, Scala, Java, and Python by Kareem Alkaseer: <https://lnkd.in/eVanhXm>
- 11- Python Notes for Professionals book: <https://lnkd.in/g2cNnFjJ>
- 12- Learning Pandas <https://lnkd.in/gM9C2BvN>
- 13- Machine Learning - A First Course for Engineers and Scientists by Andreas Lindholm, Niklas Wahlström, Fredrik Lindsten, and Thomas B. Schön: <https://lnkd.in/gzuNxKi3>
- 14- Dive into Deep Learning by Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola: <https://d2l.ai/d2l-en.pdf>
- 15- A Comprehensive Guide to Machine Learning Soroush Nasiriany, Garrett Thomas, William Wang, Alex Yang, Jennifer Listgarten, Anant Sahai: <https://lnkd.in/gp3AKgMY>
- 16-SQL Notes for Professionals book: <https://lnkd.in/g5dNZCuD>
- 17-Algorithms Notes for Professionals book: <https://lnkd.in/eX6YkWv>
- 18- Deep Learning Interviews: Hundreds of fully solved job interview questions from a wide range of key topics in AI by Shlomo Kashani, Amir Ivry: <https://lnkd.in/aMFVTbrn>
- 17-Algorithms Notes for Professionals book: <https://lnkd.in/eX6YkWv>
- 18- Deep Learning Interviews: Hundreds of fully solved job interview questions from a wide range of key topics in AI by Shlomo Kashani, Amir Ivry: <https://lnkd.in/gMFVTbrn>
- 19- Understanding Machine Learning: From Theory to Algorithms by Shai Shalev-Shwartz and Shai Ben-David: <https://lnkd.in/gEJGTfB7>
- 20-Algorithms Notes for Professionals book: <https://lnkd.in/eX6YkWv>
- 21-Bayesian Reasoning and Machine Learning by David Barber: <https://lnkd.in/g5bH8Ynb>
- 22-800+ Q&As about: Stats, Python, ML, DL, NLP, CV, MLOps by Steve Nouri: [https://lnkd.in/g5vmvp\\_v](https://lnkd.in/g5vmvp_v)

- 23- An Introduction/A History of Data Science: <https://lnkd.in/g8ahuvgc>
- 24- Introduction to Machine Learning by Alex Smola and S.V.N. Vishwanathan: <https://lnkd.in/gHZrURPM>
- 25- Deep Learning by Ian Goodfellow, Yoshua Bengio, and Aaron Courville: <https://lnkd.in/gDPIS5Fc>
- 26- Natural Language Processing with Python by Steven Bird, Ewan Klein, and Edward Loper: <https://www.nltk.org/book/>
- 27- Learn Python the Right Way by Peter Wentworth, Jeffrey Elkner, Allen B. Downey, and Chris Meyers: <https://lnkd.in/guHPGGCC>
- 28- Pandas: Powerful Python Data Analysis Toolkit by Wes McKinney and the Pandas Development Team: [https://lnkd.in/gUWy\\_meU](https://lnkd.in/gUWy_meU)
- 29- Neural Networks and Deep Learning by Michael Nielsen: <https://lnkd.in/brReGHX>
- 30- Automate the Boring Stuff with Python by Al Sweigart: <https://lnkd.in/dkVr5-n>
- 31- Patterns, Predictions, and Actions; A story about machine learning by Moritz Hardt and Benjamin Recht: <https://lnkd.in/gVU7gf3h>
- 32- Advanced Excel by Towson University: [https://lnkd.in/gU\\_kkfCZ](https://lnkd.in/gU_kkfCZ)
- 33- The Data Science Handbook by Carl Shan, Henry Wang, William Chen, and Max Song: <https://lnkd.in/g5wqFey>
- 34- Kubernetes Up and Running by Brendan Burns,  
Joe Beda & Kelsey Hightower:  
[https://lnkd.in/eM\\_9arN3](https://lnkd.in/eM_9arN3)
- 35- Introducing MLOps by Mark Treveil, Nicolas Omont, Clement Stenac, Kenji Lefevre, Du Phan:  
<https://lnkd.in/gmrXWqV9>

Check the first post here: <https://lnkd.in/g3yVbJcX>

Most of them are wikis, please consider buying the book or donating to support the authors if you like them!

I share my learning journey into Data Science with my amazing LinkedIn friends, please follow me and let's grow together! Alex Wang

#artificialintelligence #datascience

#machinelearning



# Reading a Scholarly Article

## Structure of an article

The journal articles mostly follow the format of abstract, introduction, methods, results, discussion, conclusion, and bibliography.

### Abstract (Summary)

A concise summary of the article, including methodology, results and their implications, authors' interpretations and conclusions.

### Introduction (Why)

The authors introduce their topic, explain the purpose of the study, and present the statement of the research hypothesis. There is also a brief overview of the other literature that has been written about the same topic.

### Methods/Methodology / Materials (How)

It provides the technical details of how the study was conducted, the identification of participants, procedures, and materials.

### Results (What Happened)

Includes the data the authors use to reach their conclusions. These are often presented in figures, tables and graphs.

### Discussion / Analysis/ (What It Means)

The authors analyse the data, evaluate whether the results of their study answered their hypothesis and explain how results support the conclusions.

### Conclusion (What Was Learned)

Here the authors offer their final thoughts about how the study adds to the existing knowledge on the topic, the strengths and weaknesses of the study, and recommendations for future research.

### References

A list of sources cited in the journal article.

## Reading with the SQ3R method

The SQ3R (Survey, Question, Read, Recite, Review) method is an effective approach to reading scholarly articles.

### Survey

Survey entire journal article and note the article title, authors' affiliation, journal details: volume, issue number, year of publication. Then examine in the following order: Abstract - Discussion and Conclusion - Introduction - Methods - Results. Note any key words and new terms.

Develop and write out questions about each section of the reading in relation to your assessment task. Use these questions to guide your reading and your note-taking process.

### Question

### Read

Read chronologically one section at a time using a slower in-depth reading style. Make short notes in your own words and highlight main ideas that support the questions.

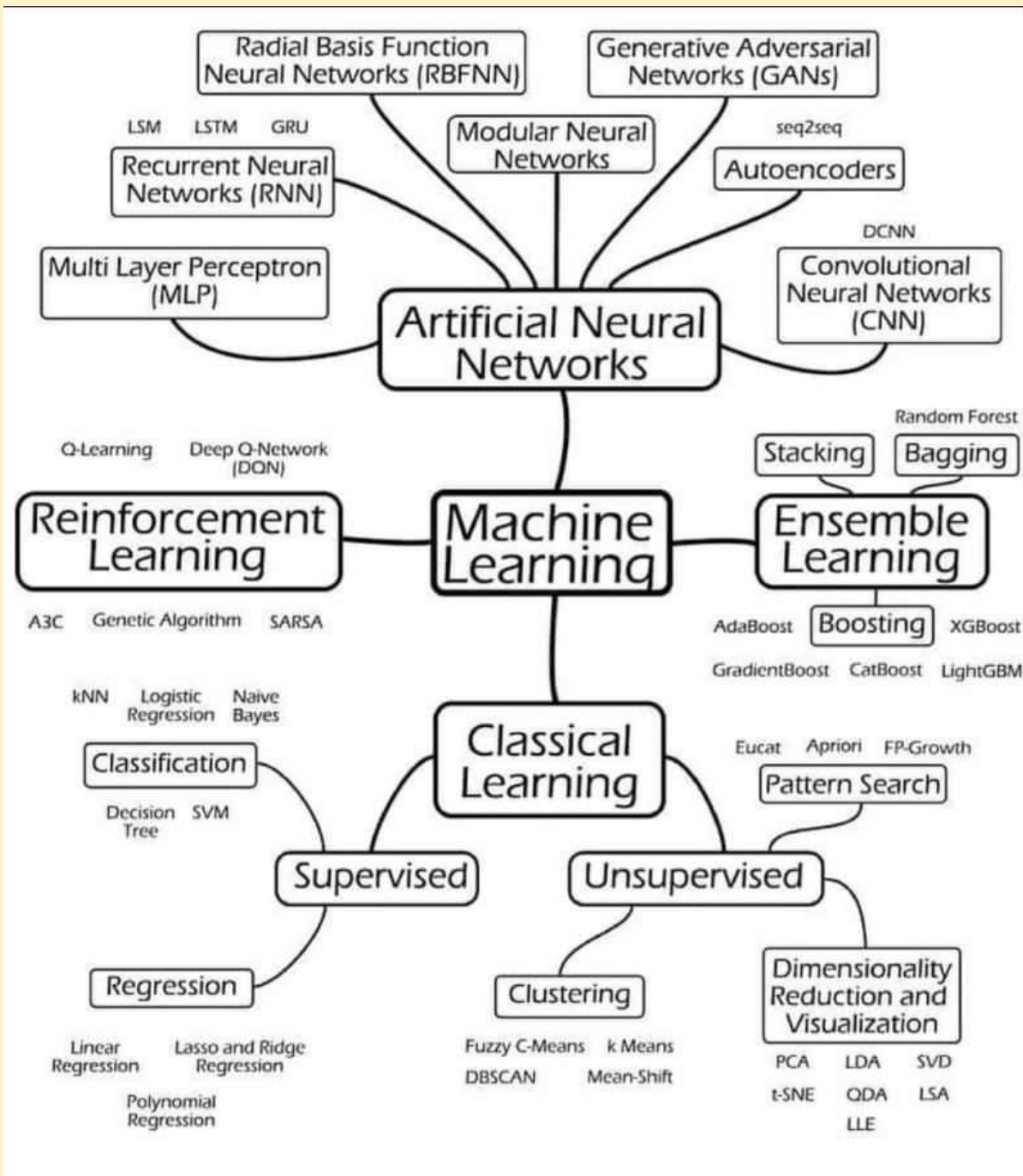
Recite and recall the information by summarising and paraphrasing. Try to answer the questions you wrote down earlier without turning back to the text.

### Recite

### Review

Review your notes against the section that you read. Note the concepts that you still don't understand and re-read the article.

# Machine Learning Algorithms



## Organizational Framework to Exploit the Data



## AI artist who won competition says art world is 'in denial' about the tech

<https://www.pcgamer.com/ai-artist-who-won-competition-says-art-world-is-in-denial-about-the-tech/>

## University of Washington MATH 340 Abstract Linear Algebra Summer 2020

Video lectures, University of Washington MATH 340 Abstract Linear Algebra Summer 2020, by Robert Won

<https://t.co/lB57B8xhaQ>

## Important links for writing research articles

Important links for writing research articles, proposals, papers etc, reference videos

How to Write a Research Article

<https://youtu.be/X3thGQPPf8>

Basics of Literature Review \ 5 Simple Steps

<https://youtu.be/9w5cqeUVpZ0>

How to Write a Review Article

<https://youtu.be/b72N-Qu7A90>

How to Develop a Good Research Topic

<https://youtu.be/gGdpXBCdxY>

How to write an Abstract

<https://youtu.be/EWdhfgC9DLY>

How to Write a Summary

<https://youtu.be/Dho56HtJPXQ>

How to Read a Journal Article

[https://youtu.be/5VxM\\_PG5sb0](https://youtu.be/5VxM_PG5sb0)

Problem Statement

<https://youtu.be/98vOS9yihRA>

How to add or update articles in Google Scholar ?

<https://youtu.be/fTIsRufXbDw>

How to Write a Strong Thesis Statement

<https://youtu.be/3y4tsQwHOUY>

How to identify a research gap ?

<https://youtu.be/gyimugAdYyo>

Five Simple Rules to Avoid Plagiarism [https://youtu.be/iBm\\_kj8jNZM](https://youtu.be/iBm_kj8jNZM)

Differences of Useful terms used in Research Methodology

<https://youtu.be/fcnKStMNVuQ>

Difference between Research Methods and Research Methodology <https://youtu.be/RSUfNkJFWeU>

Differences between Theoretical Framework and Conceptual Framework

<https://youtu.be/qXeWfhc-X0I>

Differences between Reliability and Validity

<https://youtu.be/qXeWfhc-X0I>

Differences between Review Paper vs Research Paper

<https://youtu.be/t5TTqfixL2A>

How to calculate sample size

<https://youtu.be/Zmpaak6iYv4>

Data Collection Websites

<https://youtu.be/fN7RXVg4Q9Q>

Interpretation of Regression Analysis

<https://youtu.be/E-Gd8WNa1N8>

Regression Analysis with Scientific Calculator

<https://youtu.be/NHYLthpxQnw>

How to install Add-ins

[https://youtu.be/yku\\_A6f4I78](https://youtu.be/yku_A6f4I78)

How to select Research Topic

<https://youtu.be/1QuZ0Dp9i3g>

How to cite and download articles from google scholar

<https://youtu.be/fwnjg7jkwvl>

Differences Between Google and Google Scholar

<https://youtu.be/gJFA7tZA4nU>

Literature Databases

<https://youtu.be/2WV0K8AaDqo>

Basics of MS Excel

[https://youtu.be/pbHbnHD\\_EyM](https://youtu.be/pbHbnHD_EyM)

Research Methods For Business: A Skill Building Approach By UMA SEKARAN

<https://youtu.be/IW-tYvM7Kbk>

Best Books of Research for beginners

<https://youtu.be/OeACYQwcvU>

Best Academic Paper Writing Books

<https://youtu.be/c3aB7rl2mv8>

How to Convert Word to PDF \ Without Software or Internet

[https://youtu.be/\\_ITupCafSyk](https://youtu.be/_ITupCafSyk)

How to Convert PDF to word \ Without Software or Internet

<https://youtu.be/5BzZGghbFJw>

Guidelines for Writing a Thesis or Dissertation

<https://youtu.be/JlumBcr21yU>

How to write the introduction?

<https://youtu.be/S-koanSBMOY>

MCQs on Research Methodology

Important for all Competitive Examinations

<https://youtu.be/xKh3SLWweWs>

Research Proposal Presentation Speech

[https://youtu.be/RO\\_V\\_UQ9NPE](https://youtu.be/RO_V_UQ9NPE)

Article Writing Pros and Cons

<https://youtu.be/HUYybGIHGEU>

## **Geometric Deep Learning - AMMI**

Graphs data are everywhere. Deep learning is increasingly showing remarkable results in processing and analyzing graphs and alike datasets.

Here are great lectures for people interested in geometric deep learning/graph ML.

<https://t.co/kSxRhcgrWl>

## **Creating Mathematical Animations in Python using Manim -**

<https://www.analyticsvidhya.com/blog/2022/08/creating-mathematical-animations-in-python-using-manim/>

## **Necessary libraries for Manim**

Necessary libraries are FFmpeg, OpenGL, and LaTeX (optional, if you want to use LaTeX). MikTeX is recommended.

## **manim library**

Manim is a Python library that is used for creating animation

## **Professional AI whisperers have launched a marketplace for DALL-E prompts**

<https://www.theverge.com/2022/9/2/23326868/dalle-midjourney-ai-promptbase-prompt-market-sales-artist-interview>

## **Dealing with Outliers Using Three Robust Linear Regression Models**

<https://towardsdatascience.com/dealing-with-outliers-using-three-robust-linear-regression-models-544cfbd00767>

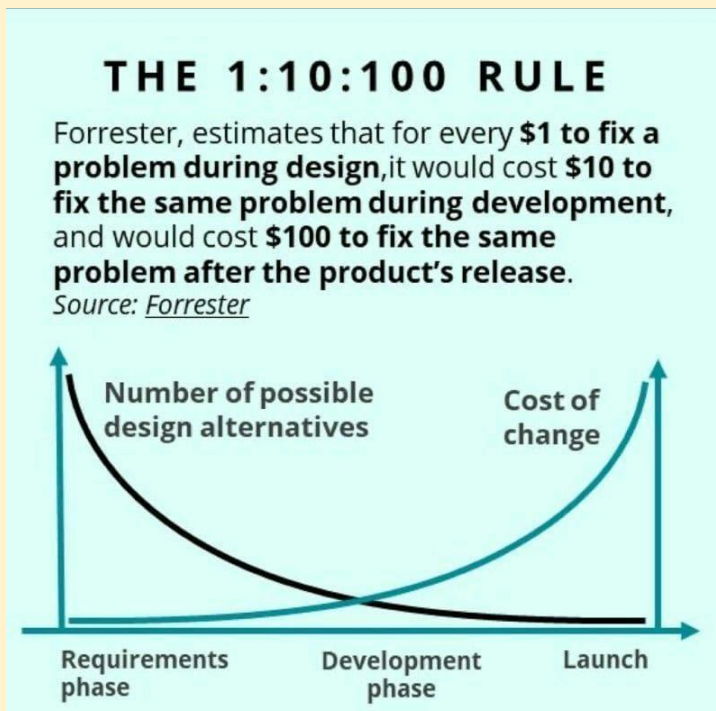


## solar Power and Surface Area Required



<https://landartgenerator.org/blagi/archives/127>

## Cost of defect Leakage



## 5WH of Data Engineering

5 Five WH questions that every data engineer must ask when building a pipeline.

🎯 **What** is the data source?

- It could be RDBMS systems or flat files.

🎯 **Where** to store data?

- SQL/NOSQL databases or storage buckets.

🎯 **Why** to store data here?

- To simplify data storage, create data lake, data security.

🎯 **How** to store the data?

- Store in form of files or tables.

🎯 **When** is data arriving?

- Time when data will arrive, and at what frequency.

🎯 **Who** is the consumer?

- End users like analyst or ml engineers for visualisations and model creation.

📌 The above given examples are just to give you an understanding, these could vary based on each one's experience.

## A Novel Approach to Creating Tailored Odors and Fragrances Using Machine Learning

<https://www.titech.ac.jp/english/news/2022/064776>

## Reservoir of data science and machine publications

Medium is a reservoir of data science and machine publications that publishes some of the best DS/ML articles from writers around the world

Here are the top 6 publications you must follow to read the best articles in the data domain

- 1) ODSC - Open data science <https://t.co/Mq9EcTr1ls>
- 2) Towards data science <https://t.co/mNYqJVksGi>
- 3) Mlearning-ai <https://t.co/Y4H1vCkC85>
- 4) Data-driven fiction <https://t.co/MTwUJLov8L>
- 5) Geek culture <https://t.co/teDKGI3tOQ>
- 6) Codex <https://t.co/Uoi3A9cUMd>

## How to Download, Install and Use Nvidia GPU For Tensorflow

<https://www.analyticsvidhya.com/blog/2020/11/how-to-download-install-and-use-nvidia-gpu-for-tensorflow-on-windows/>

## Techniques for Training Large Neural Networks

<https://openai.com/blog/techniques-for-training-large-neural-networks/>

## microsoft/ML-For-Beginners - Classic Machine Learning for all

microsoft/ML-For-Beginners: 12 weeks, 26 lessons, 52 quizzes, classic Machine Learning for all

<https://github.com/microsoft/ML-For-Beginners>

## SigmaQuan/Better-Python-59-Ways:

Code Sample of Book "Effective Python: 59 Specific Ways to Write Better Python" by Brett Slatkin

<https://github.com/SigmaQuan/Better-Python-59-Ways>

## Top 20 Linked List Coding Problems

10 Algorithm Books Every Programmer Should Read

<http://www.java67.com/2015/09/top-10-algorithm-books-every-programmer-read-learn.html>

Top 5 Data Structure and Algorithm Books for Java Developers

<http://javarevisited.blogspot.sg/2016/05/5-free-data-structure-and-algorithm-books-in-java.html#axzz4uXETWjmV>

20+ String Coding Problems from Interviews

<https://dev.to/javinpaul/top-20-string-coding-problems-from-programming-job-interviews-493m>

50+ Data Structure and Algorithms Problems from Interviews

<https://dev.to/javinpaul/50-data-structure-and-algorithms-problems-from-coding-interviews-4lh2>

30+ Array-based Problems from Coding Interviews

<https://javarevisited.blogspot.com/2015/06/top-20-array-interview-questions-and-answers.html>

From 0 to 1: Data Structures & Algorithms in Java

[https://click.linksynergy.com/fs-bin/click?id=JVFXdTr9V80&subid=0&offerid=323058.1&type=10&tmpid=14538&RD\\_PARM1=https%3A%2F%2Fwww.udemy.com%2Ffrom-0-to-1-data-structures%2F](https://click.linksynergy.com/fs-bin/click?id=JVFXdTr9V80&subid=0&offerid=323058.1&type=10&tmpid=14538&RD_PARM1=https%3A%2F%2Fwww.udemy.com%2Ffrom-0-to-1-data-structures%2F)

Data Structure and Algorithms Analysis — Job Interview

[https://click.linksynergy.com/fs-bin/click?id=JVFXdTr9V80&subid=0&offerid=323058.1&type=10&tmpid=14538&RD\\_PARM1=https%3A%2F%2Fwww.udemy.com%2Fdata-structure-and-algorithms-analysis%2F](https://click.linksynergy.com/fs-bin/click?id=JVFXdTr9V80&subid=0&offerid=323058.1&type=10&tmpid=14538&RD_PARM1=https%3A%2F%2Fwww.udemy.com%2Fdata-structure-and-algorithms-analysis%2F)

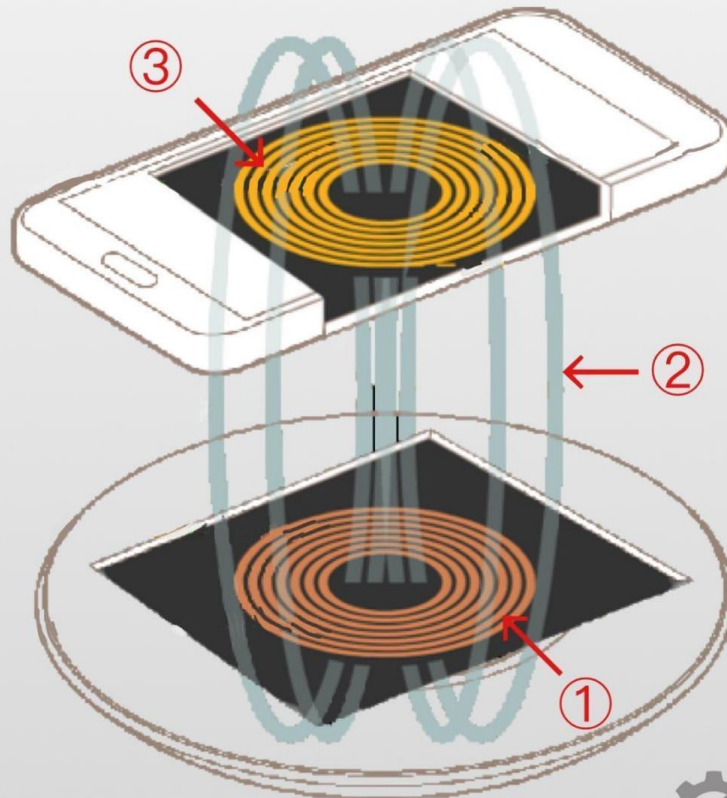
10 Books to Prepare Technical Programming/Coding Job Interviews

<http://www.java67.com/2017/06/10-books-to-prepare-technical-coding-job-interviews.html>

## How Wireless Charger Works

# HOW WIRELESS CHARGER WORKS

Current passes through an induction coil in the pad (1). It creates a magnetic field (2). This changing magnetic field creates an alternating electric current in the phone's induction coil (3), which passes through a rectifier to convert it to direct current.



Finally, the direct current charges a battery.



## Probability summary

Random variable - what turns to be upside of a coin

Event - a coin is thrown in air

Success - upside of a coin is head.

Probability of success- chances upside will be head after the event.  $P(h)$

Conditional probability - if outcome of last event was head then probability next will be head. (Entire seen of future)

Outcome of last event was head, what is the probability next will be head. (past and future).  $P(h/h)$

Joint probability - probability 2 coin thrown together and both are head.  $P(h)$  and  $P(h)$

Mutual exclusive events

## Causal Inference and Statistical Inference

What are the Causal Inference and Statistical Inference?

Causal inference is the process of determining the independent, actual effect of a particular phenomenon that is a component of a larger system.

Statistical inference is the process through which inferences about a population are made based on certain statistics calculated from a sample of data drawn from that population.

Causal inference refers to an intellectual discipline that considers the assumptions, study designs, and estimation strategies that allow researchers to draw causal conclusions based on data.

What are some examples of causal inference?

For a single person, the causal effect of taking vitamin C in this context would be the difference between the expected outcome of taking vitamin C and the expected outcome of not taking vitamin C.

What is the difference between causal inference and statistical inference?

Causal inference is the process of ascribing causal relationships to associations between variables. Statistical inference is the process of using statistical methods to characterize the association between variables. Causality is at the root of scientific explanation which is considered to be a causal explanation.

Can machine learning be used for causal inference?

When humans rationalize the world, we often think in terms of cause and effect — if we understand why something happened, we can change our behavior to improve future outcomes. Causal inference is a statistical tool that enables our AI and machine learning algorithms to reason in similar ways.

Why causal inference is important?

Causal Inference Demonstrates the Importance of the Random Allocation of Units. When a random allocation is not used in a study, units may be purposefully allocated to conditions. In that case, the simple comparison of average scores between groups may not produce an unbiased estimate of the treatment effect.

What is causal inference in data science?

Causal inference is a powerful methodology that allows a data scientist to identify causes from data, even when no experiment or test has been performed. Using causal methods increases the level of confidence in business decision-making by clearly connecting causes and effects.

Why is causal inference difficult?

Making valid causal inferences is challenging because it requires high-quality data and adequate statistical methods.

Read course materials online. If you're teaching a causal inference course, refer notes!

<https://t.co/fqSxhzGsnR>



## Why Zomato is goldmine

| Restaurant 50 seater                     |                     |
|------------------------------------------|---------------------|
| Rental cost                              | 6 Lakhs (2000 sqft) |
| Rental Deposit                           | 20 Lakhs            |
| Licenses                                 | 8-10 Lakhs          |
| Equipment                                | 8-15 Lakhs          |
| Working captial                          | 20 Lakhs            |
| <b>Break even revenue 20 Lakhs/month</b> |                     |



Amazon real estates

<https://youtu.be/fTuVWwTQegQ>

Understand customer profile

Understand what features makes product great

Owns online real estate to push up its own products

Same applies to Zomato

Best seasons and location of a food .

1. Cloud kitchen. Behros Biryani

## Quantum information science

Feynman may have had mostly quantum simulations in mind, but at present “quantum information science” includes sensing, cryptography, simulation, networking and computing.

A qubit is any two-level system that can be physically implemented by various methods, like the Josephson junction, polarisation encoding, quantum dots, nuclear magnetic resonance, and electronic spin.

## AI Problems

Problem1 - Area, resto name, cuisine1 rating, cuisine 2 rating, cuisine 10 rating

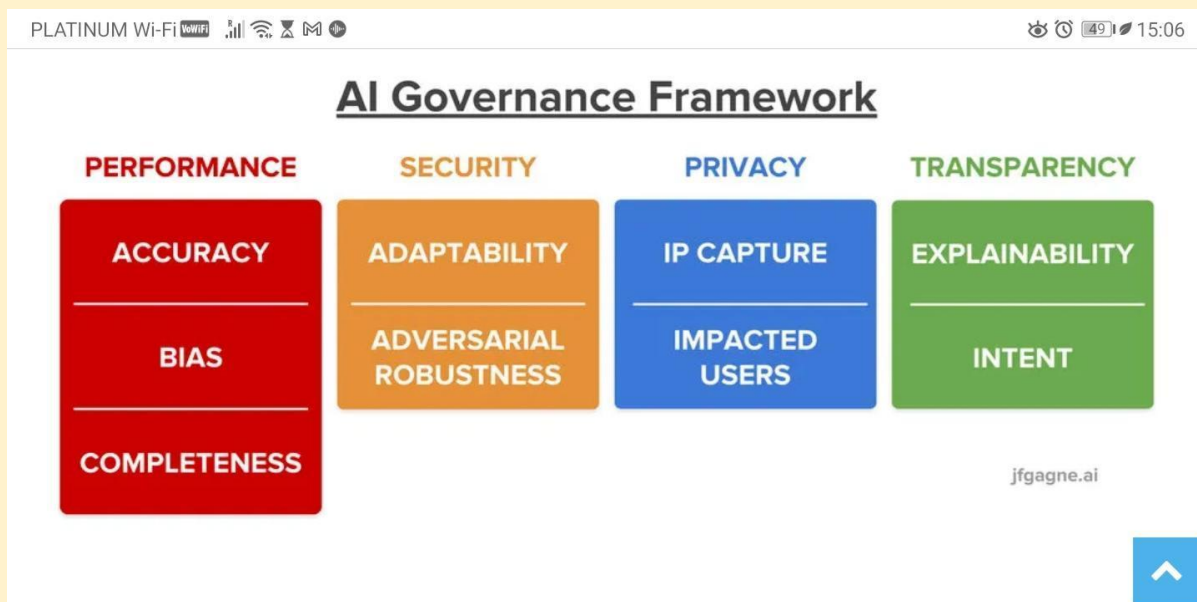
50 records of for each restaurant

Refresh dataset every 4 hrs and recommend people the best restaurants in the area for a specific cuisine type.

Algo: kbandit

Problem 2 - 2L stores need to be clustered into 10k clusters. Based on foot fall, invoices, sales, purchase, employee, number of type of categories sold etc.

## AI governance framework



## Agile Scrum Resources

<https://www.scrum.org/resources/blog/order-events-scrum-matters>

## Weather Data

Weatherapi.com,  
&nbsp;[openweathermap.org](https://openweathermap.org)

# HOW TO BUILD A POSITIVE TEAM CULTURE

**TIP 1**

Discuss 'success' – What does it mean? How would it feel? What would it look like?



**TIP 2**

Set goals – Work with the team to identify individual and team goals that they want to pursue



**TIP 3**

Discuss values – Find out from your team the values that they believe are of utmost importance



**TIP 4**

Respect – Build a culture where everyone is treated equally



**TIP 5**

Responsibility – Players must accept their role and be responsible for their actions



**TIP 6**

Commitment – Players must be committed to the team, training and games



**TIP 7**

Continuous development – Building a positive team culture takes time. Are you committed to keep developing it?



**TIP 8**

Praise – Make sure that coaches are appropriately praising players. Athletes can also praise each other



**TIP 9**

Support – Coaches and athletes can support each other when things become hard in training or games



**TIP 10**

Work together – It is key that everyone works together when things become tough.



**TIP 11**

Team building – Develop activities outside of sport so players can get to know each other



**TIP 12**

Lead by example – Have a number of key role models who people can look up to



**TIP 13**

Clear expectations – Identify clear expectations to the team. Ask them of their expectations of you as a coach



**TIP 14**

Be patient – Building a team culture takes time and it is important that you are patient with the process



**TIP 15**

Communicate – Ask the right questions and make sure that you communicate effectively with everyone on the team



## Key Factors for Successful Goal Achievement



95% data has height of 5 years children  $\leq 2\text{ft}$ . 5% data says height is  $>2\text{ft}$ .

95% data says salary of Indian is  $\geq 2000\text{\$}$  and  $\leq 10000\text{\$}$ . 2.5% says  $<2\text{k}\text{\$}$  and 2.5% says  $>10\text{k}\text{\$}$

95% data says weight of dry fruit in chocolate is  $\geq 10\text{gm}$ . 5% says it is  $<10\text{gm}$ .

This is Hybrid meeting

Those who cannot join physically can join on Zoom link

<https://exiweb.in/indo-european-course>

## Application of Stats

I mentioned in the last session that data scientist are paid highly for asking unique, insightful questions.

Ask yourself, what is the real meaning of distribution?

why do we need to know different types of distribution?

What are the characteristics of any distribution?

What if pickup wrong distribution type?

What is the relationship between the probability function and the distribution plot?

What is the probability function? What does it do?

Continuous distribution is about continuous numbers like weight, height, width, volume, speed, radius, etc.

Discrete distribution is about integer numbers.

Uniform means equal probability.

Binomial means two probability both can be different but whatever is that it remain constant.

Gaussian distribution = Bernoulli distribution

What is a Random variable or stochastic variable  
Random Phenomenon

Toss a coin 3 times.  $X$  times head will come.  
 $X$  can be 0,1,2,3

$X$  is a random variable.

What is the probability 2 times head will come?  
What is the probability at least 2 times head will come?  
What is the probability more than 2 heads will come?  
What is the probability no more than 2 heads will come?

Throw 2 coins 3 times simultaneously.  $X$  times head, the head will come. 0,1,2,3,4,5,6.  
 $X$  is a random variable.

What is the probability at least one head will come?

Your university gives grades on a scale of 4. Random variable - student grade can be anything between 0 to 4. Including decimals.

You stop by a random person on your campus. The Height of that person can be 48 to 84 inches. Height is a random variable.

A mother delivers her first baby after  $X$  many months of marriage.  $X$  can be 9 months to 400 months.

You go to polling booth for casting votes. You need to wait there between 0 min to a number of minutes between 5 pm and the time of your arrival.

## **Future of AI**

automating data collection and experiment design; automating data cleanup and missing data imputation; automating feature selection and transformation; automating model discovery, criticism, and explanation; automating the allocation of computational resources; automating hyperparameter optimization; automating inference; and automating model monitoring and anomaly detection.



## what to do when I make a mistake



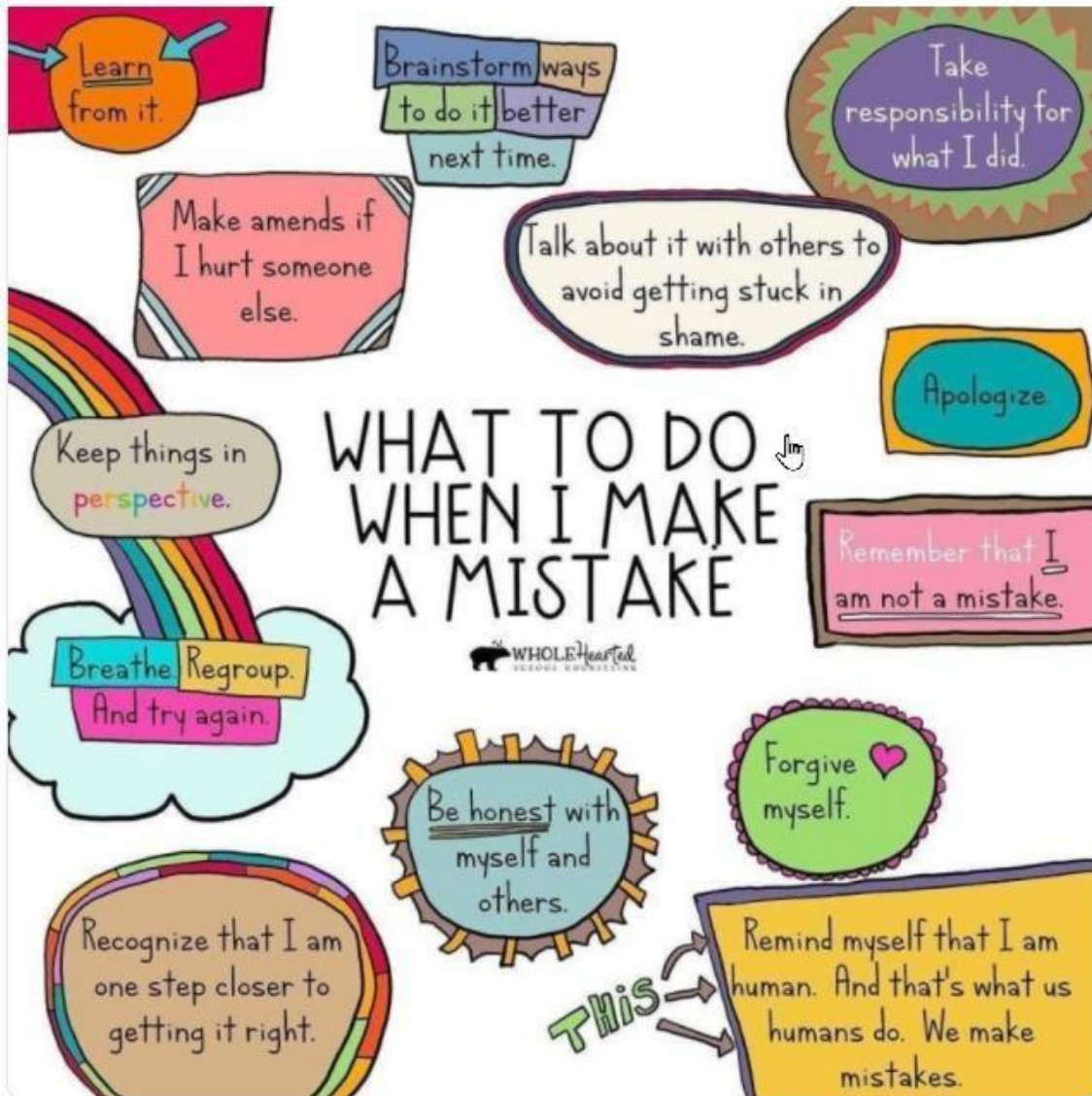
David McLean, MA (Leadership) CHRL • 2nd  
Human Resources Leader | Leadership Coach | Healthcare | Talen...  
2d • Edited •

+ Follow

What is your orientation to mistakes?

What is your "compass" or "map" for navigating mistakes?

...see more



## A new model that automatically generates movie trailers

<https://techxplore.com/news/2021-11-automatically-movie-trailers.html>

## Software Professions



**Backend**



**Frontend**



**APIs**



**Full-stack**



## Password Breaking Time

# Time it takes a Hacker to Brute Force your password

@coders.bro

| Numbers of Character | Numbers Only | Lowercase Letters | Upper and Lowercase Letters | Numbers, Upper and Lowercase Letters | Numbers, Upper and Lowercase Letters, Symbols |
|----------------------|--------------|-------------------|-----------------------------|--------------------------------------|-----------------------------------------------|
| 4                    | Instantly    | Instantly         | Instantly                   | Instantly                            | Instantly                                     |
| 5                    | Instantly    | Instantly         | Instantly                   | Instantly                            | Instantly                                     |
| 6                    | Instantly    | Instantly         | Instantly                   | 1 sec                                | 5 secs                                        |
| 7                    | Instantly    | Instantly         | 25 secs                     | 1 min                                | 6 mins                                        |
| 8                    | Instantly    | 5 Secs            | 22 mins                     | 1 hour                               | 8 hours                                       |
| 9                    | Instantly    | 2 mins            | 19 hours                    | 3 days                               | 3 weeks                                       |
| 10                   | Instantly    | 58 mins           | 1 month                     | 7 months                             | 5 years                                       |
| 11                   | 2 secs       | 1 day             | 5 years                     | 41 years                             | 400 years                                     |
| 12                   | 25 secs      | 3 weeks           | 300 years                   | 2k years                             | 34k years                                     |
| 13                   | 4 mins       | 1 year            | 16k years                   | 100k years                           | 2m years                                      |
| 14                   | 41 mins      | 51 years          | 800k years                  | 9m years                             | 200m years                                    |
| 15                   | 6 hours      | 1k years          | 43m years                   | 600m years                           | 15bn years                                    |
| 16                   | 2 days       | 34k years         | 2bn years                   | 37bn years                           | 1tn years                                     |
| 17                   | 4 weeks      | 800k years        | 100bn years                 | 2tn years                            | 93tn years                                    |
| 18                   | 9 months     | 23m years         | 6tn years                   | 100tn years                          | 7qd years                                     |

Are you in green zone?

## Adjacent Numbers in a Perfect Square 1-32

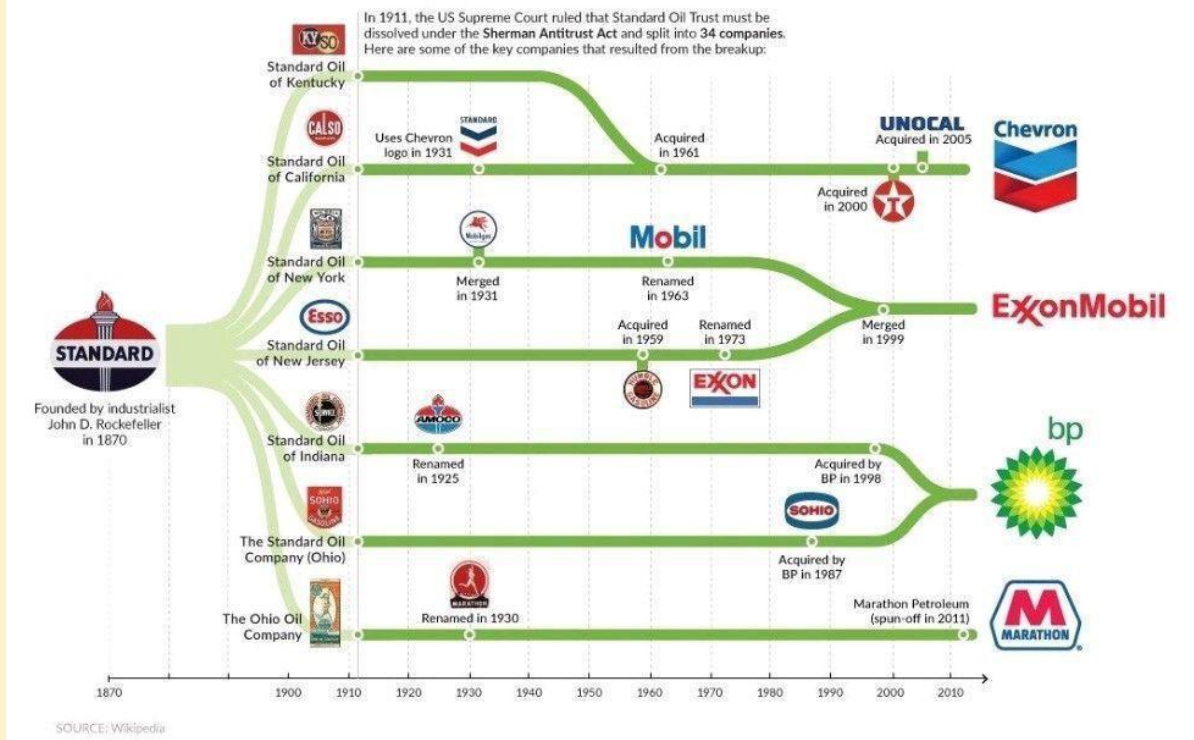
It is an amazing circle of numbers 1 to 32 without a repeat. The beauty is that the sum of any two adjacent numbers is a perfect square.

Beauty of mathematics

## Evolution of Standard Oil

### THE EVOLUTION OF STANDARD OIL

Following the remnants of John D. Rockefeller's oil juggernaut



## Watch Me Build an AI Startup

<https://youtu.be/NzmoPqte4V4>

Create this app on my own.

## ML Frameworks/Tools



## Data Science Project Stages

Data acquisition  
Data transformation  
Data security  
Data architecture  
Data analysis  
Data mining

Data modelling  
Feature engineering.  
Model evaluation  
Feature selection  
Model deployment  
Model monitoring



## Machine Learning Resources

# Machine Learning

1. Book, Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn and TensorFlow"
2. Foundations of Machine Learning
3. Best Resources to Learn Machine Learning
4. Curated List of Libraries For a Faster Machine Learning Workflow
5. Awesome TensorFlow
6. "Papers with Code" - Browse the State-of-the-Art in Machine Learning
7. Zhi-Hua Zhou. 2012. Ensemble Methods: Foundations and Algorithms. Chapman & Hall/CRC.
8. Feature Engineering for Machine Learning. Principles and Techniques for Data Scientists. By Alice Zheng, Amanda Casari
9. Google Research: Looking Back at 2019, and Forward to 2020 and Beyond
10. O'Reilly: The road to Software 2.0
11. Machine Learning and Data Science Applications in Industry
12. Deep Learning for Anomaly Detection
13. Federated Learning for Mobile Keyboard Prediction
14. Federated Learning. Building better products with on-device data and privacy on default
15. Federated Learning: Collaborative Machine Learning without Centralized Training Data
10. Book: Molnar, Christoph. "Interpretable machine learning. A Guide for Making Black Box Models Explainable", 2019
11. Book: Hutter, Frank, Lars Kotthoff, and Joaquin Vanschoren. "Automated Machine Learning". Springer, 2019.
16. ML resources by topic, curated by the community.
17. An Introduction to Machine Learning Interpretability, by Patrick Hall, Navdeep Gill, 2nd Edition. O'Reilly 2019
18. Examples of techniques for training interpretable machine learning (ML) models, explaining ML models, and debugging ML models for accuracy, discrimination, and security.

## Understanding Skewed Distribution

If a factory has 100 employees

They need to reach the factory by 9 am. This is called on-time arrival

$X$  = 100 employees reaching on time (random variable)

You observe this variable for 365 days

Day 1 60

Day 2 90

Day 3 25

Day 4 99

Day 5 100

.

.

Day 365 97

You plot a distribution of these 100 observations

Employees

1 0

2 0

3 1

...

80 20

...

95 125

100 1

This will be left-skewed distribution

## Google Search Tech

Use these words while searching

Define

Double quote

Site

Intitle

Allintitle

Inurl

Allinurl

Before

After

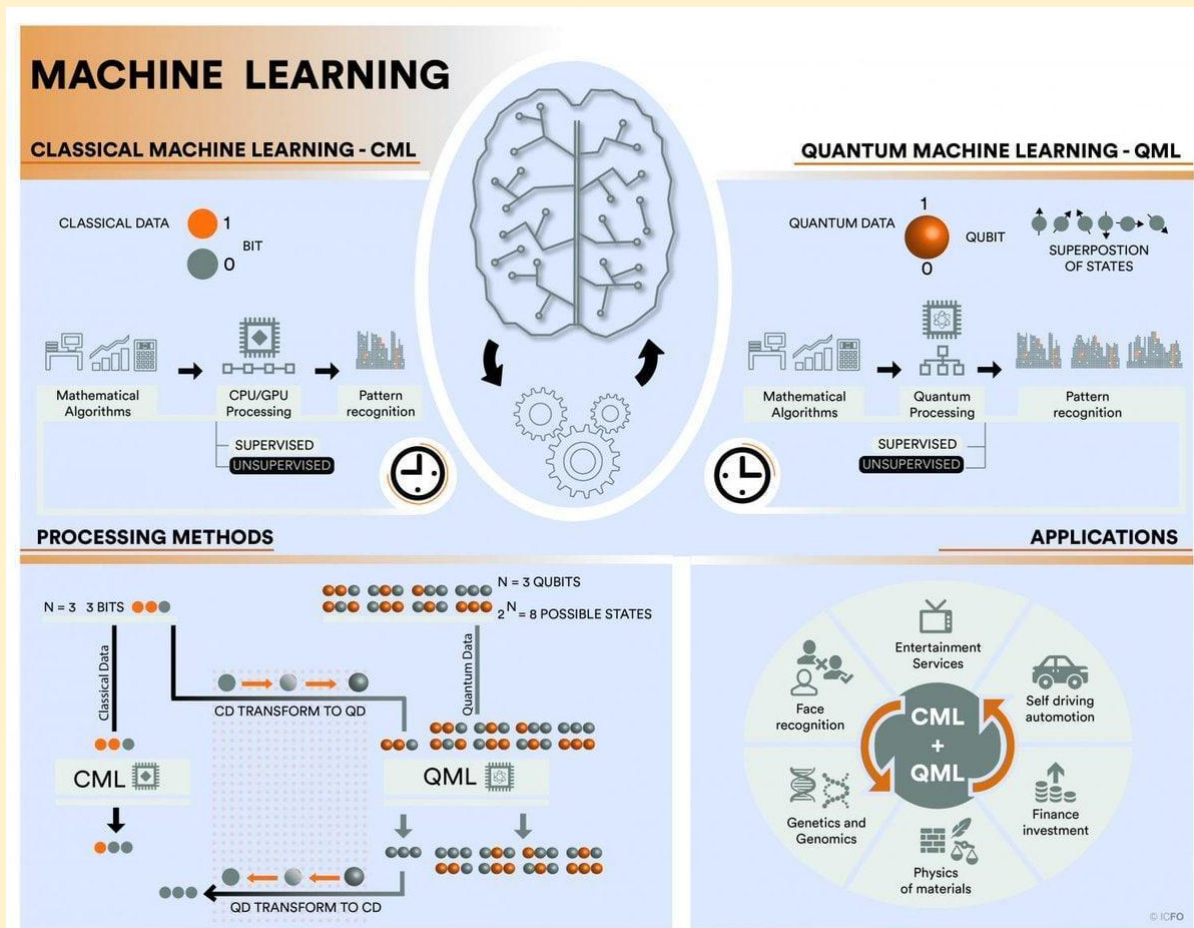
.. / \* / -

AND

OR

Alert

# Machine Learning Classical vs Quantum



## New England Complex Systems Institute

<https://necsi.edu/>

## Google Cloud Next

[https://twitter.com/i/events/1445131126134116353?t=YepEP4dXVVOI55ehX\\_U1qA&s=08](https://twitter.com/i/events/1445131126134116353?t=YepEP4dXVVOI55ehX_U1qA&s=08)

## Why You Should Always Switch: The Monty Hall Problem

Explained - By Steven Pinker - Behavioral Scientist

<https://behavioralscientist.org/steven-pinker-rationality-why-you-should-always-switch-the-monty-hall-problem-finally-explained/>