MET CS688

WEB ANALYTICS AND MINING

ZLATKO VASILKOSKI

INTRODUCTION

BU Community COVID-19 Public Health Policies

- All students returning to campus will be required to be vaccinated against COVID-19, and upload information about their status (including applications for a medical or religious exemption or an extension) to the Patient Connect portal.

- In addition to the vaccine requirement, students must follow all other safety protocols, including the face covering policy, and screening, contact tracing, and testing requirements.

- At the beginning of each class you will be asked to show a green Healthway compliance badge on your mobile device to the instructor, and wear your face mask over your mouth and nose at all times.

General Course Information

- *Instructor*: Zlatko Vasilkoski (email: zlatko@bu.edu or zlatko.vasilkoski@gmail.com)
- *Office hours*: by appointment (you can email me with questions at any time)
- *Class Time*: Thursdays from 6:00pm to 8:45pm
- *Course Prerequisites*:
  - MET CS 544 - Foundations of Analytics or
  - MET CS 555 - Data Analysis and Visualization
- *Course Grading Policy*:
  - Quizzes/Lab Projects (10%)
  - Assignments (20%)
  - Midterm exam (35%)
  - Term project (35%)

100–93.00	A
92.99–90.00	A–
89.99–87.00	B+
86.99–83.00	B
82.99–80.00	B–
79.99–77.00	C+
76.99–73.00	C
72.99–70.00	C–
69.99–60.00	D
Below 60.00	F

- *Course Topics*:
  - Web Analytics and Web Analytics Tools
  - Text and Web Mining
  - Mining the Social Web, Twitter, Game Analytics
  - Data Visualization, Google visualization APIs illustrated on above mining examples
  - Basics of machine learning
  - No specific textbook (reference textbooks are listed in the syllabus)

My Background

- PhD. in physics from Tufts University working with David Weaver and Martin Karplus on computational implementation of the diffusion collision multi scale model of protein folding, to which the 2013 Nobel Prize for chemistry was awarded. The algorithms I designed as part of my thesis, greatly expanded the applicability of the the diffusion collision model.
- My college teaching experience includes BU, Tufts, MIT, Suffolk, Wentworth and Bentley.
- I have been developing the curriculum and working as a lecturer at the Metropolitan College Computer Science department since 2012.

- My work experience includes
  - ML Architect at EBSCO
  - Chief Data Scientist at FacilityConneX
  - Senior research scientist at Neurala Inc.
  - Postdoctoral research work at MIT and Northeastern
  - Worked in the area of Neural Network's learning laws at Department of Cognitive and Neural Systems, at BU.
  - Worked as data scientist at Harvard Medical School.
- My current research interests include algorithm development in computational physics, biomedical image processing, computer graphics, computer vision, machine learning, NLP and neural network systems.

Class overview

- This course covers the theoretical and practical aspects of
  - Web Analytics and Internet of Things (IoT)
  - Text Mining (including NLP, and Deep Learning Neural Nets such as Transformer Models)
  - Mining the Social Web, Twitter, Game Analytics
  - Game and Sports Analytics
  - Data Visualization, Google visualization APIs illustrated on above mining examples
  - Basics of machine learning

Class overview

Web Analytics and Internet of Things (IoT)

- The web analytics part of the course studies
  - the metrics of web sites
  - their content
  - user behavior during web site visit
  - reporting

- The use of Google Analytics, Google Trends and Google Correlate will be also illustrated.
- Through an IoT homework you will get familiarized with this different aspect of Web Analytics.

Class overview

- The text mining part covers the analysis of text and it includes
  - Preprocessing and content extraction from various file types
  - The mathematical (matrix) representation of the extracted text
  - String matching, fuzzy string matching, and their measures of closeness
  - Documents matching in the “concept space” and the simple math behind it
  - Aspects of supervised learning, Tagging, Classification, and Categorization
  - NLP and introduce relevant ML techniques such as Deep Learning, Transformers, Reformers etc.
- The web mining (structure & content) part covers aspects such as
  - Web crawling (gathering pages from the web)
  - Indexing (to support a search engine)
  - Understanding Search Performance and how to measure it
  - The graph representation of the web pages and ranking the web pages
  - Practical applications to the social web and online game data

- Illustrations of these concepts are given using R (and few examples in Python).

Please complete the Blackboard survey on your programing language (R, Python) preference.

Class overview

At the end of the semester, you will need to present a Term Project on which you would work for some time.

Here are some Term Project examples from the previous years:

- Search engine:

Popcorn DB

PopCorn DB http://popcorn-db.net is a personal project which aims at recreating **from scratch** an IMDB like website with a machine learning layer on top of it. **Therefore it includes the following features:**

A fast & scalable web crawler.

I used Apache Spark for parallel computing, and InfluxDB for logging in live its activity. To reuse the CPU idle time when waiting for network responses I configured Spark to create 8 times more executors than CPU cores for each machine of the cluster.

A blazingly fast custom built search-engine with fuzzy search and autocompletion.

The average query time for 100K movies is 0.03ms. The speed is obtained by indexing every possible ngram of each movie title. The fuzzy search is done by building & exploring Levenstein automata on the go.

A movie genre & nationality predictor

I used a naive bayes network approach as it seemed after experimentation to be the best Machine Learning model adapted to this case.

A web-server, socket-server and front-end

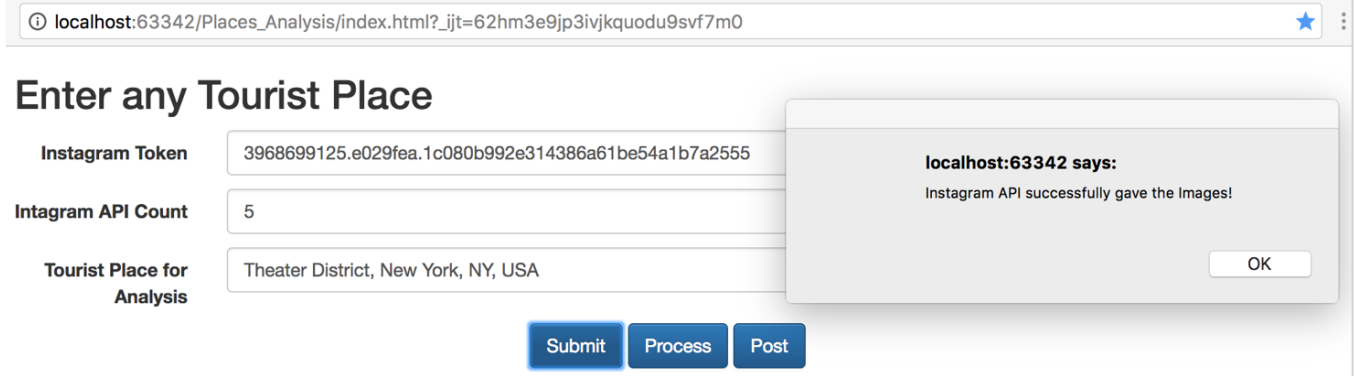
The search engine and machine learning layers are written in C++. So I decided to build a web-server also in my C++ program. No need of Apache or nginx, less overhead = more speed.

Class overview

Some other Term Projects Examples

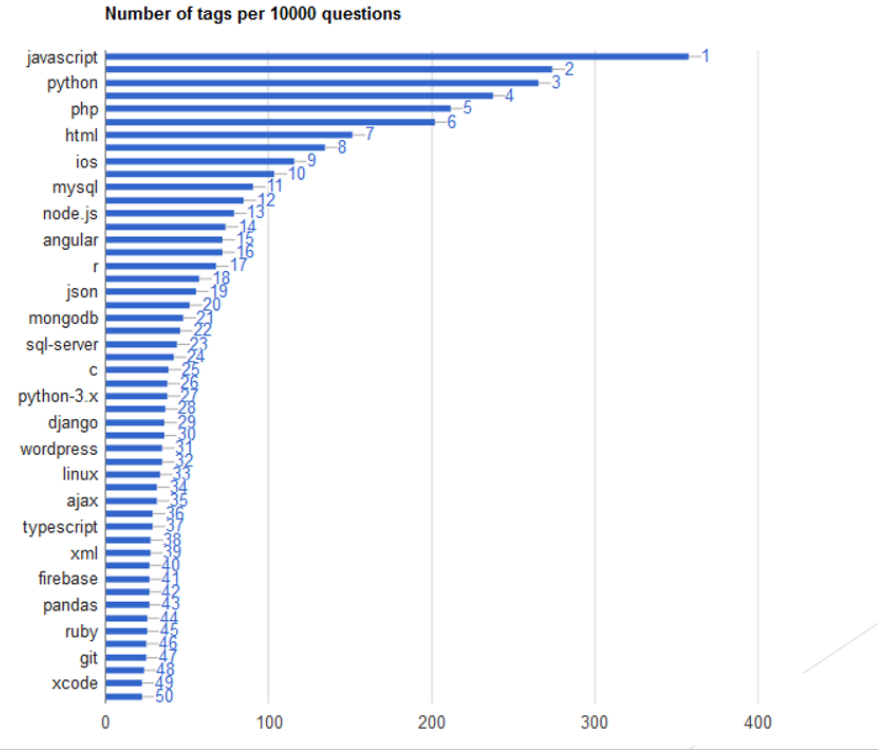
- Data mining Stackoverflow
- Places Analysis : Mining
  - Analysis on the basis of the place given by user
  - Shows the data related to the images on that location
  - Analyze the Images to get the attributes like age, gender, smile
  - Sentiment Analysis on Comments from Instagram
  - Visualization of data on Dashboard

Instagram API Call



Tags analytics

TOTAL: 36534



Please Introduce Yourself

- Introduce yourself to me and the other students
- Tell us about your background, your interests, hobbies, etc., so that we can get to know each other better.
- Please describe two or three objectives you hope to accomplish by the end of the course, e.g.
  - How does this course fit into your academic and professional objectives;
  - What do you hope to gain from the course.
- Please describe the type of data you work with and what pattern you typically look for in it.

Introduction

- Most of the information we use today is stored online. There are claims that the data generated over the last 2 years is few fold larger than the data generated previously in the history of mankind.

- Most of this newly generated data is text, images and videos in a form of email, Google, YouTube, Facebook, Twitter, blogs, and most of the other technologies that define our digital age. To this we should also add the new communication tools such as social networks, instant messaging, Yammer, Twitter, Facebook, LinkedIn etc. too.

- By some general estimates a **third** of our time is spent on searching for information and another **quarter** analyzing it. It is widely believed that more and more data will be generated in the near future (IoT) and the time managing this data must be as productive as possible.

- This is just one aspect of what this course is about! We have an exciting journey ahead as we acquire the skills regarding this subject step-by-step.

Examples of Web analytics use

Web analytics is commonly used to give:

- Real-time visibility into web site performance,
- Order status,
- Inventory levels,
- Warehouse management systems.

Defining IoT

- The Internet of things (IoT) is relatively recent term and refers to interconnecting of any physical devices (things) embedded with electronics which enables these devices to collect and exchange data across existing network infrastructure. Typically, IoT refers to gateways and protocols to talk to devices that might be in cars, supermarkets, university campuses, nuclear powerplants or anywhere else. IoT gateways and protocols enable connectivity of these devices to the cloud, from where the cloud applications can directly talk to these devices.

- In 2013 the United Nations specialized agency for information and communication technologies, the Global Standards Initiative on Internet of Things (IoT-GSI) defined the IoT as:

"A global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies." (http://www.itu.int/en/ITU-T/gsi/iot/Pages/default.aspx)

IIoT – Industrial Internet of Things

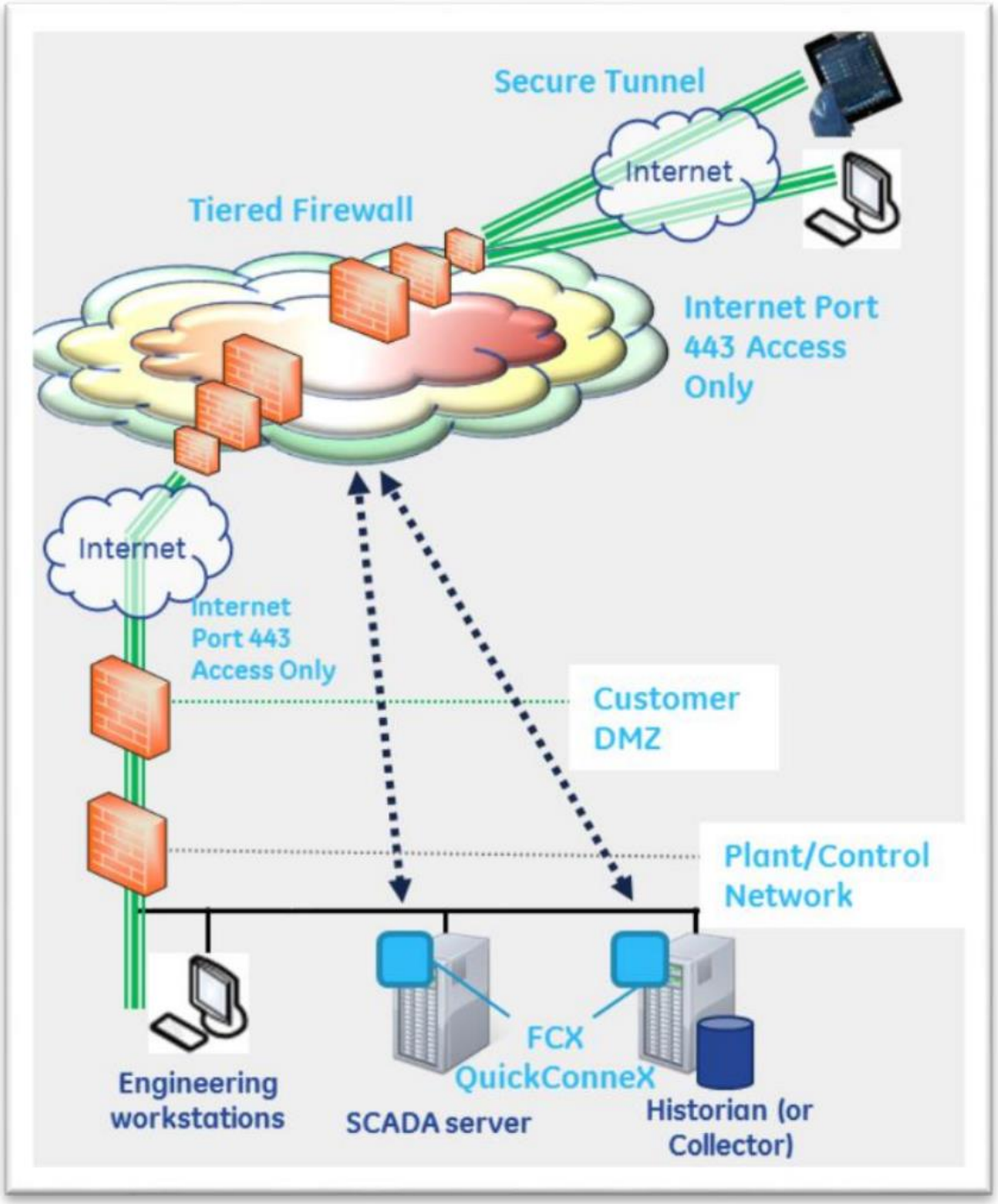
- The development of IIoT over the last few years was designed to collect and analyze equipment and environmental data to provide notification and operational recommendations about facility's optimal operation. With other words to provide **continuous commissioning**.

- Just as an illustration, a large university campus such as BU typically spends $30 million per year on fuel. A continuous commissioning implemented over the IIoT typically can bring 3% to 5% savings in fuel consumption. This accumulates to yearly savings in millions of dollars on fuel only, which is not to be ignored. If not continuously commissioned, losses of a university campus for example can accumulate to hundreds of millions of dollars over a period of 10 years. On country level, the opportunities for savings are even larger. For example, US spends $29 billion each year on air-conditioning, consuming 6% of all the electricity produced in the United States. That is why great savings can be achieved by implementing an IIoT analytics.

- The power of continuous commissioning is the ease with which analytics can be implemented over IIoT to large set of facilities to achieve the desired improvements. Typically, the goal of the implemented continuous commissioning analytics is to
  - Detect abnormal condition scenario and report on it - fault detection and diagnostics (FDD).
  - Forecasting – predict certain scenarios based on collected data.

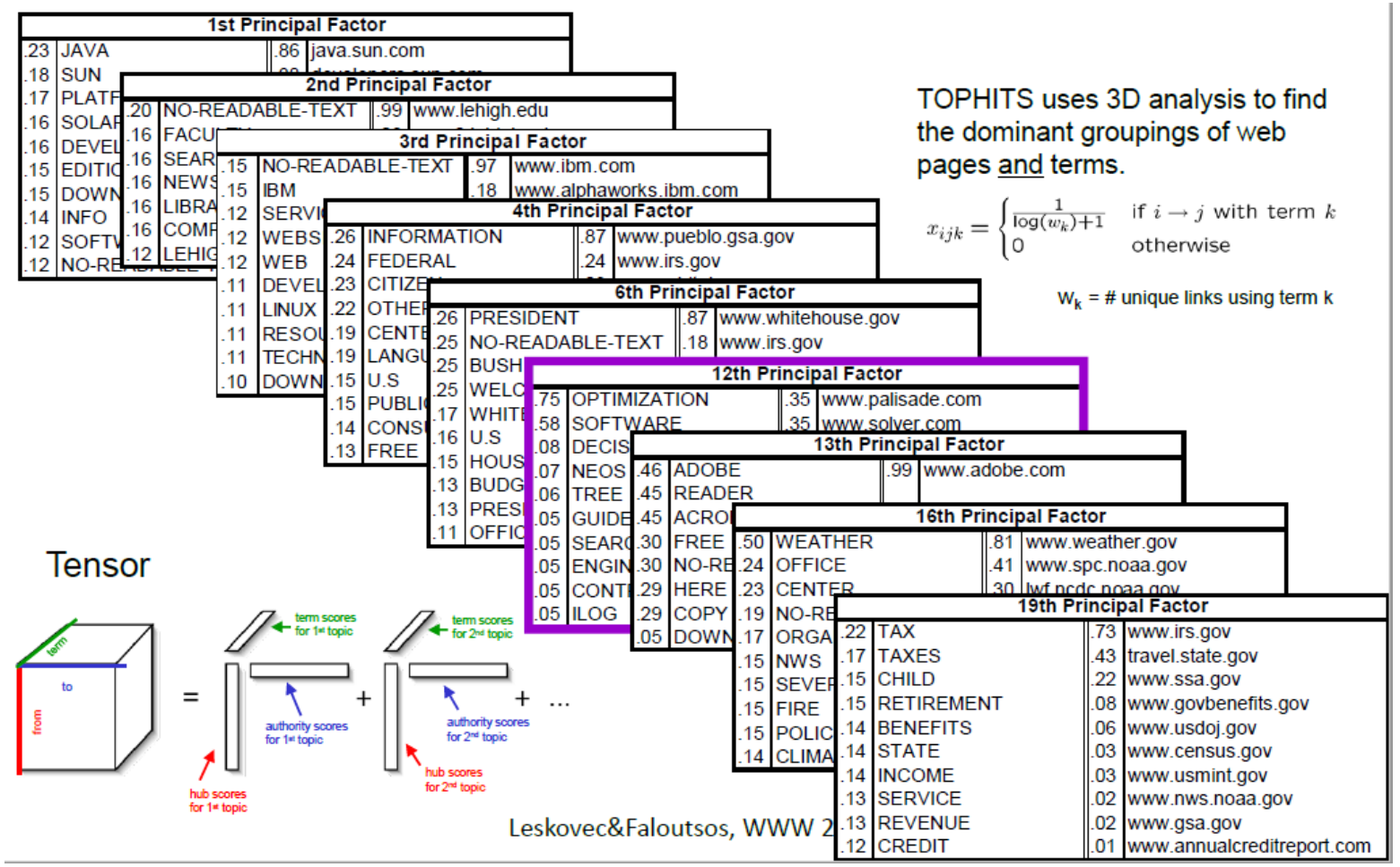
Connecting to the Edge with FCX

- Let us illustrate the IIoT's infrastructure on the example how FacilityConneX (FCX) platform is connecting to the Edge. FacilityConneX (www.facilityconnex.com) is a cloud-based monitoring and analytics service, leveraging aspects of General Electric (GE) platform for the Industrial Internet, Amazon Web Services (AWS) for the cloud services and EMC for security technology. The basic IIoT infrastructure used by FacilityConneX (as described in one of their white papers at https://www.facilityconnex.com/white-papers/) is shown to the right.



Data Analytics

- Real data are often in high dimensions with multiple aspects (modes)
- Matrices and tensors provide elegant theory and algorithms

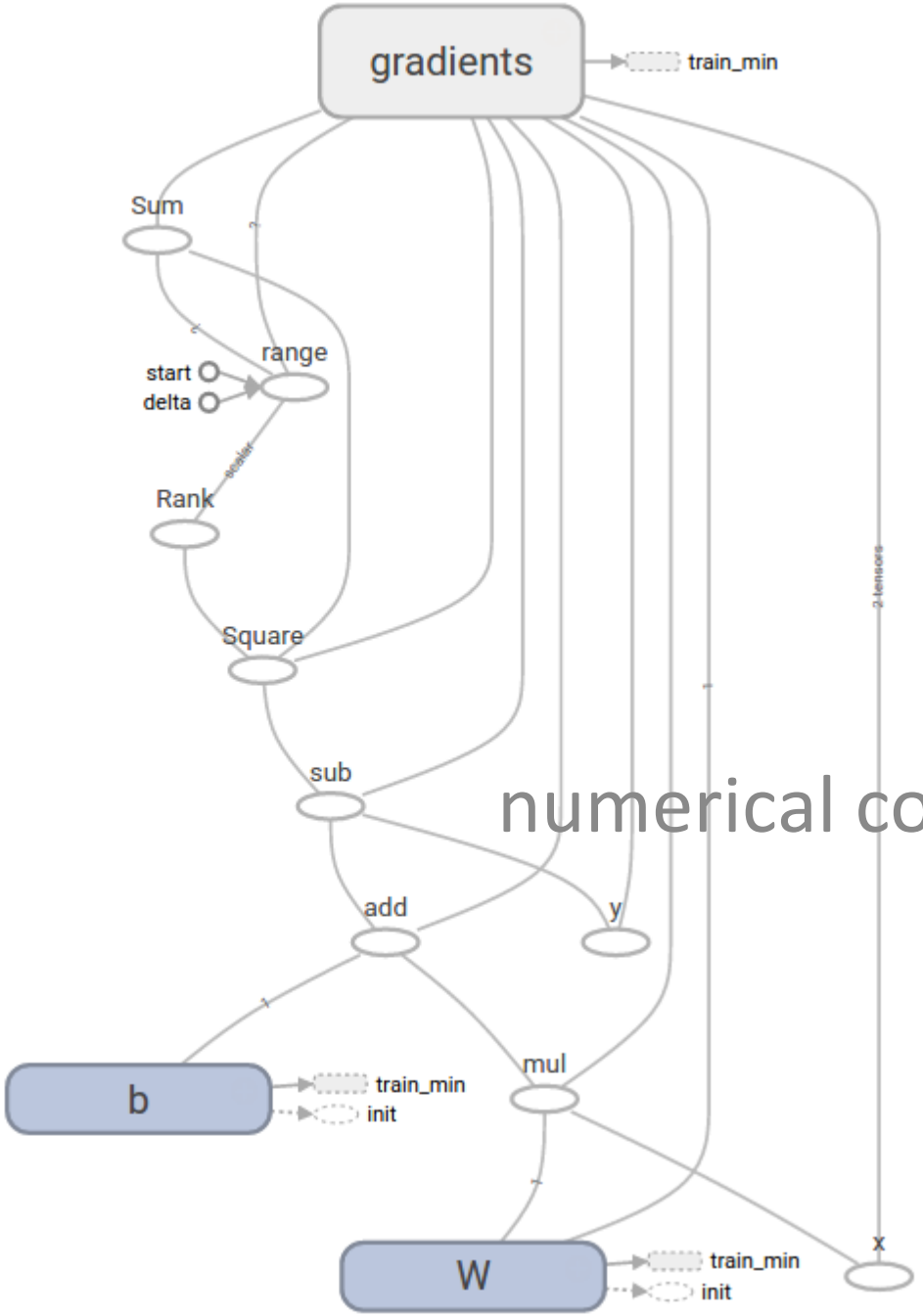


What is TensorFlow

- **TensorFlow** is a high-profile entrant into machine learning, developed by Google as an open-source successor to DistBelief, their previous framework for training neural networks. TensorFlow uses a system of multi-layered nodes that allow you to quickly set up, train, and deploy artificial neural networks with large datasets. It combines four key abilities:
  - Efficiently executing low-level tensor operations on CPU, GPU, or TPU.
  - Computing the gradient (ML & Backpropagation) of arbitrary differentiable expressions.
  - Scaling computation to many devices (e.g. the Summit supercomputer at Oak Ridge National Lab, which spans 27,000 GPUs).
  - Exporting programs ("graphs") to external runtimes such as servers, browsers, mobile and embedded devices.

TensorFlow & ML

- TensorFlow was originally developed by researchers and engineers working on the Google Brain Team within Google's Machine Intelligence research organization for the purposes of conducting machine learning and deep neural networks research, but the system is general enough to be applicable in a wide variety of other domains as well.
- That is how TensorFlow™ become an open-source software library for numerical computation using data flow graphs.
- Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) communicated between them.
- The flexible architecture allows you to deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device with a single API.
- Another very used framework is PyTorch.



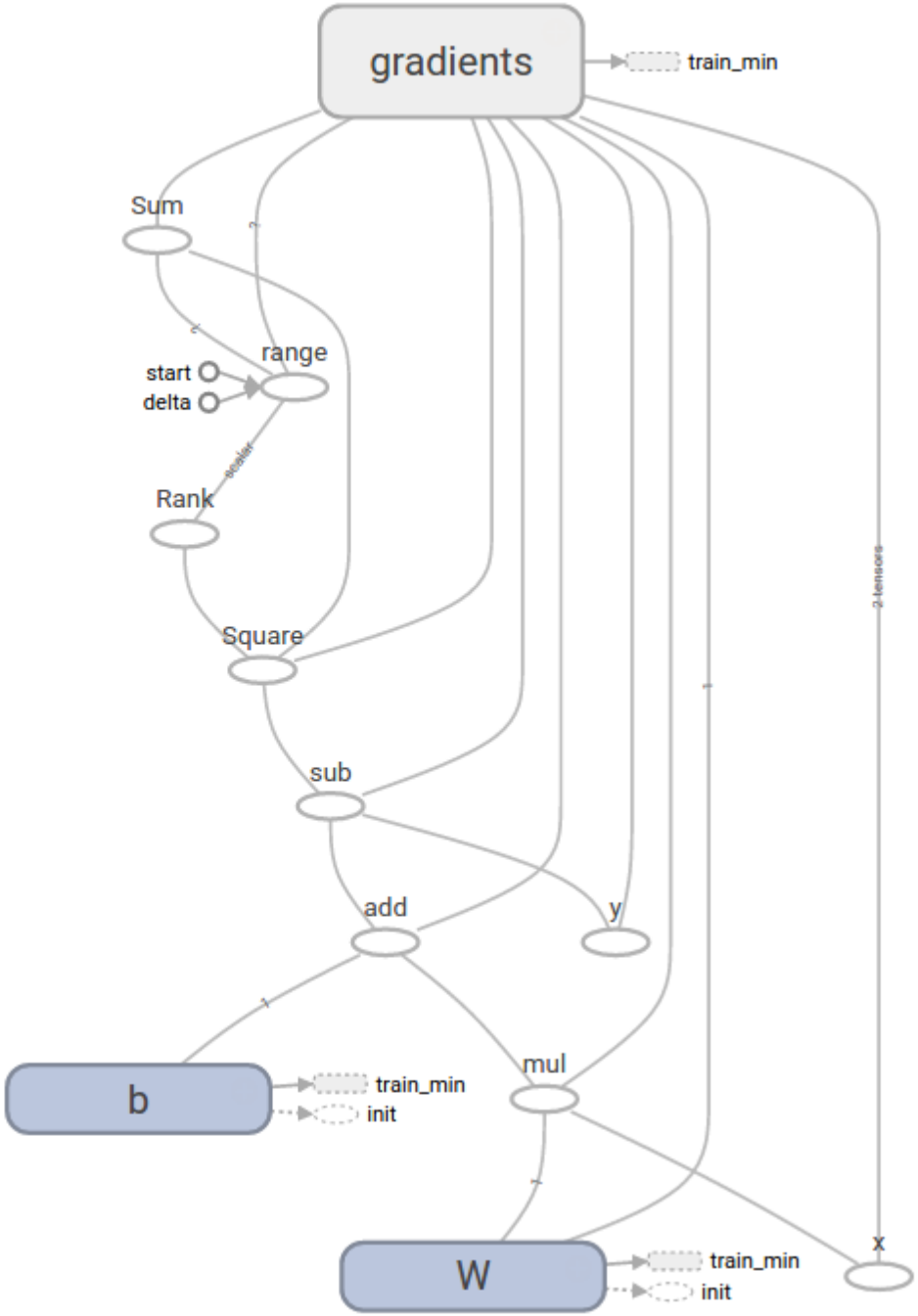
TensorFlow™

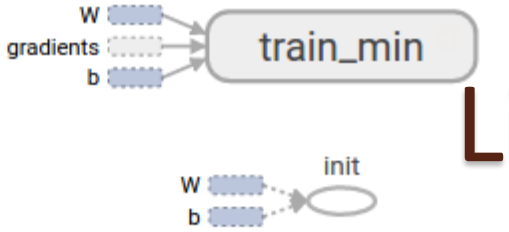
numerical computation using data flow graphs.

Basic Usage & How TensorFlow Works

- It is a programming system in which you represent computations as graphs.
  - Graph nodes are called *ops* (short for operations)
  - An op takes zero or more Tensors, performs some computation, and produces zero or more Tensors.
- Executes graphs in the context of Sessions.
  - A TF graph is a description of computations, and to compute anything, a graph must be launched in a Session.
  - A Session places the graph ops onto Devices, such as CPUs or GPUs, and provides methods to execute them.
  - These methods return tensors produced by ops as R vectors, matrices, and multi-dimensional arrays.
- Represents data as tensors. Maintains state with Variables.
- Uses feeds and fetches to get data into and out of arbitrary operations.
- TensorFlow computations define a computation graph that has no numerical value until evaluated!

TF programs structured into a construction phase, that assembles a graph, and an execution phase that uses a session to execute ops in the graph.





Linear Regression

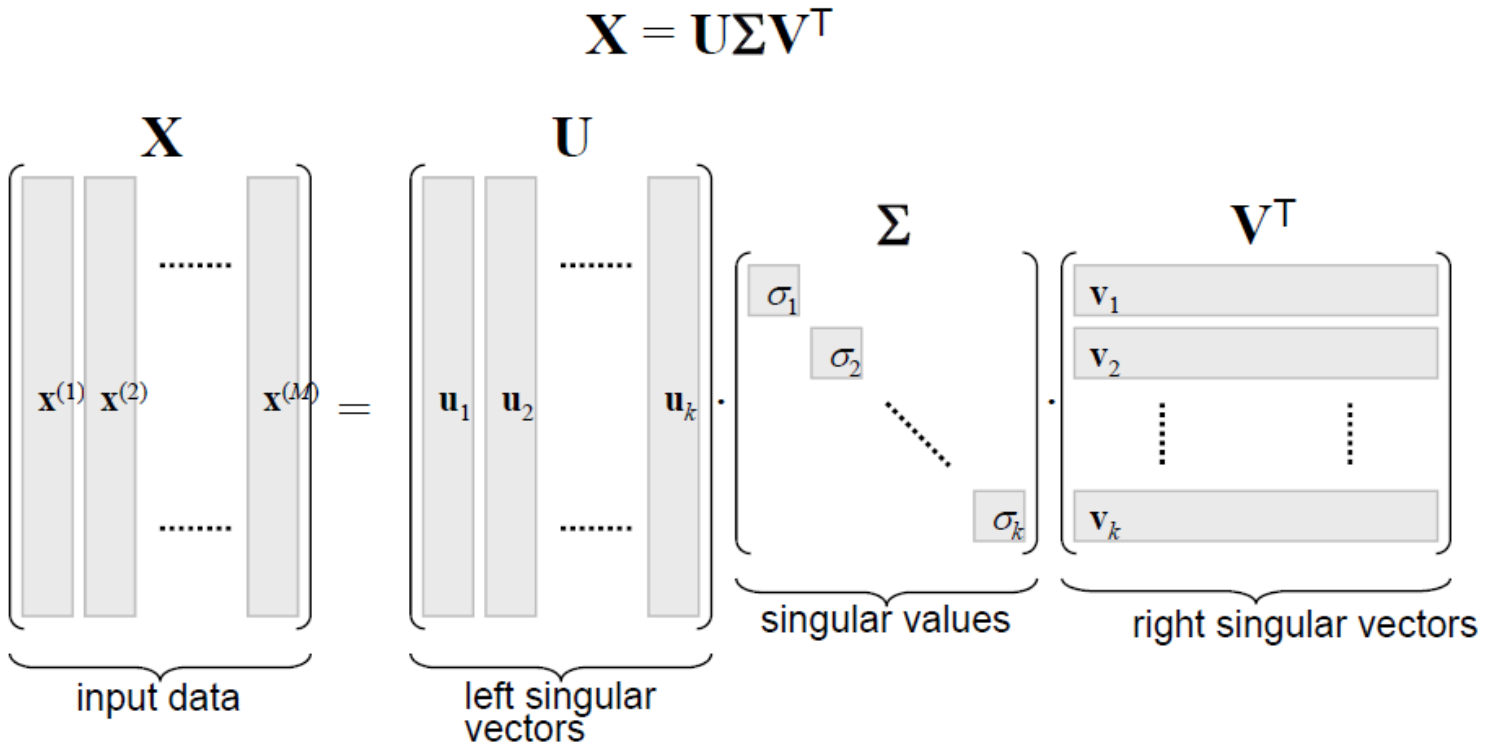
Basic TF Usage: programs structured into a

1. Construction phase, that assembles a graph, and
2. Execution phase that uses a session to execute the ops in the graph.

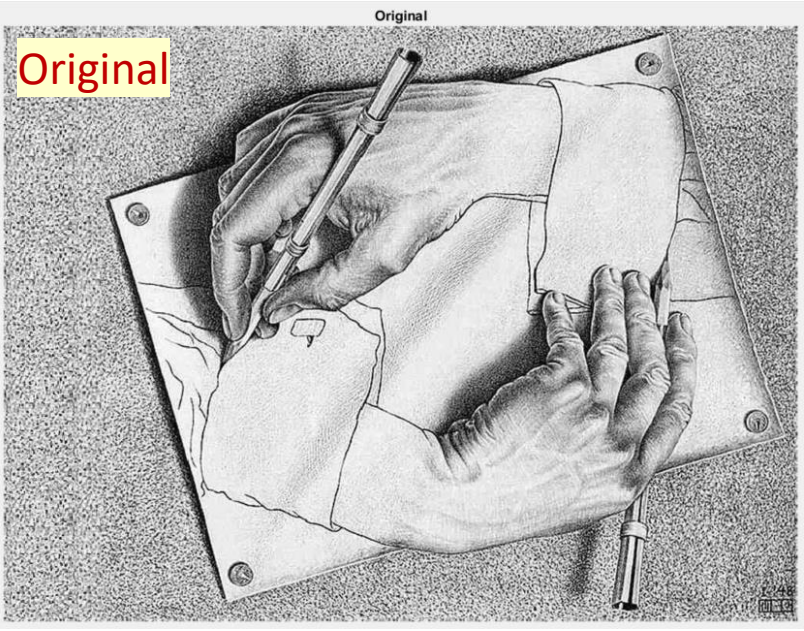
```
W <- tf$Variable(tf$random_uniform(shape(1L), -1.0, 1.0))
b <- tf$Variable(tf$zeros(shape(1L)))
y <- W * x_data + b

# Minimize the mean squared errors.
loss <- tf$reduce_mean((y - y_data) ^ 2) # Cost Function
optimizer <- tf$train$GradientDescentOptimizer(0.5) # Optimization using Gradient Descent
train <- optimizer$minimize(loss)

# Launch the graph and initialize the variables.
sess = tf$Session()
sess$run(tf$global_variables_initializer())
```

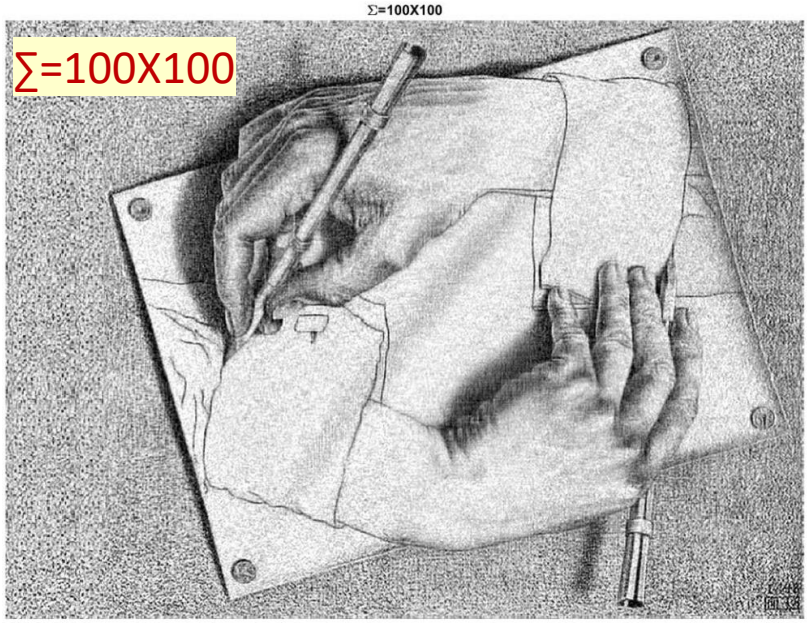
Singular Value Decomposition (SVD)

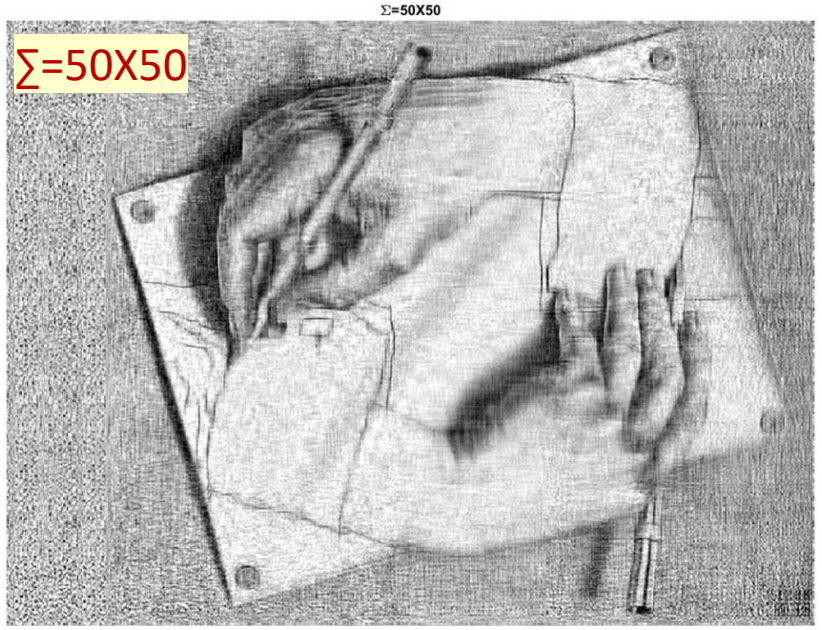
- https://en.wikipedia.org/wiki/Latent_semantic_analysis

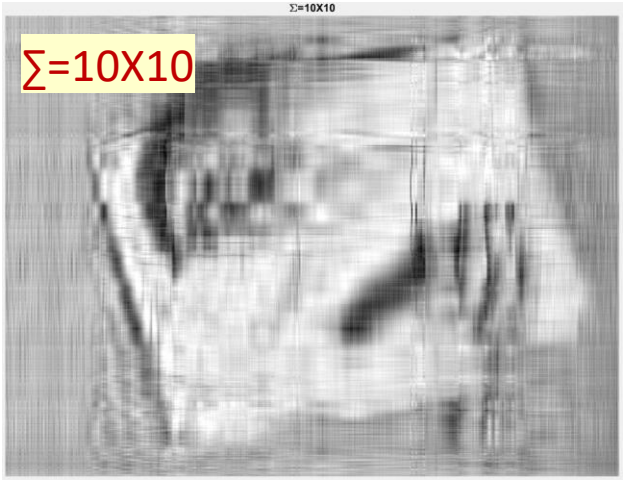


Singular Value Decomposition (SVD)







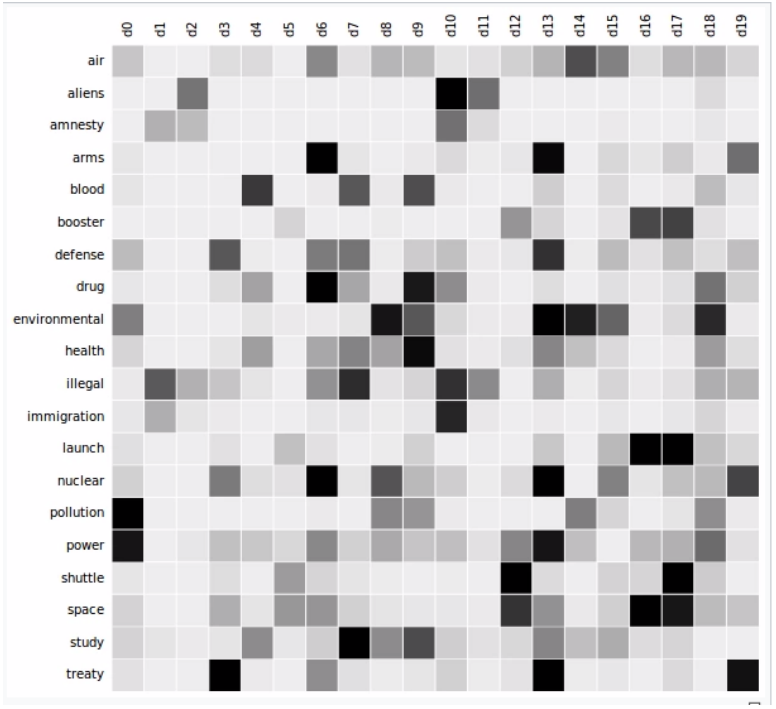


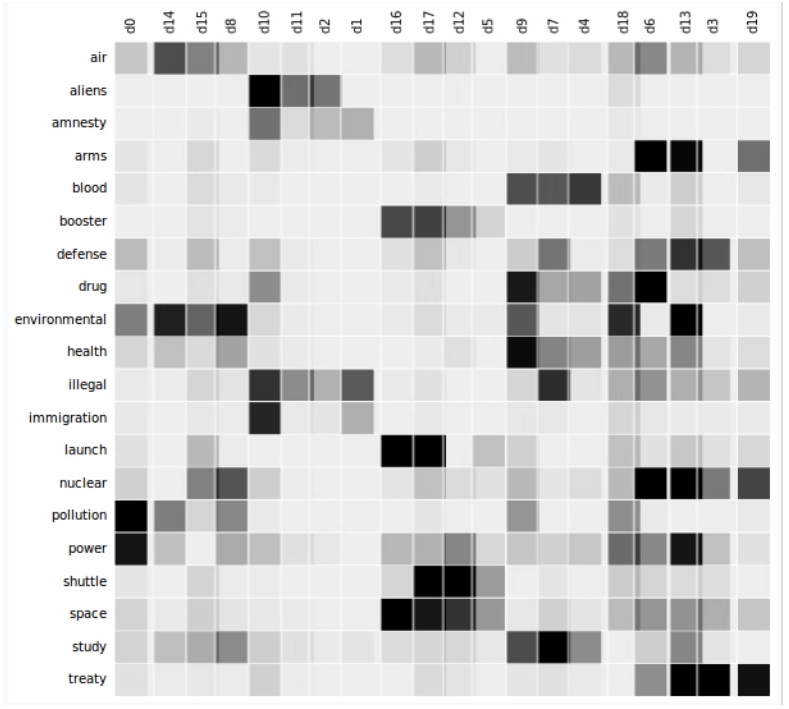
∑=10X10

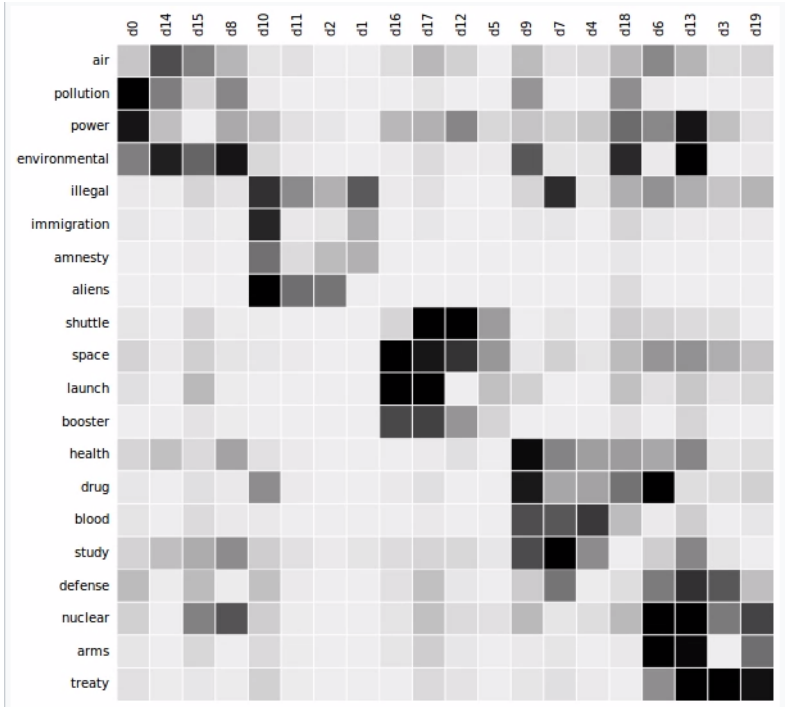
630.1644	0	0	0	0	0	0	0	0	0	0
0	84.7838	0	0	0	0	0	0	0	0	0
0	0	49.1495	0	0	0	0	0	0	0	0
0	0	0	41.1680	0	0	0	0	0	0	0
0	0	0	0	32.7525	0	0	0	0	0	0
0	0	0	0	0	30.8359	0	0	0	0	0
0	0	0	0	0	0	29.1636	0	0	0	0
0	0	0	0	0	0	0	28.1809	0	0	0
0	0	0	0	0	0	0	0	24.8132	0	0
0	0	0	0	0	0	0	0	0	23.7083	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0

SVD and Text Mining

- https://en.wikipedia.org/wiki/Latent_semantic_analysis







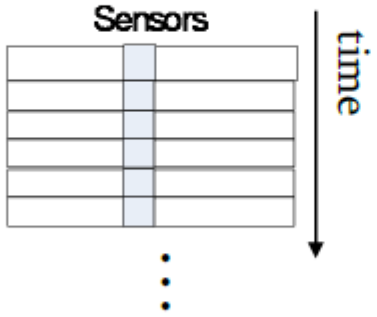
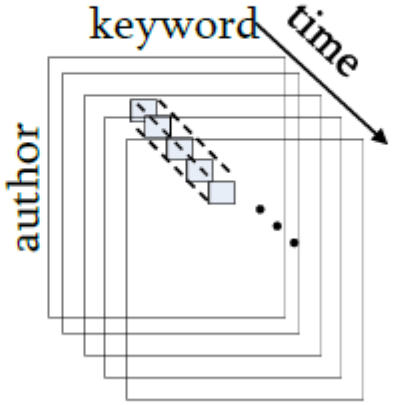
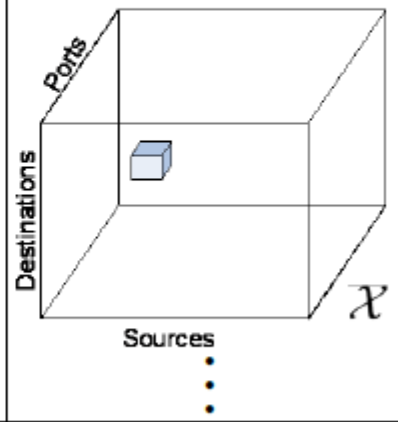
A is document term [mxn] matrix (m rows, n columns).

A^TA is term to term similarity [mxm] matrix.

AA^T is document to document similarity [nxn] matrix.

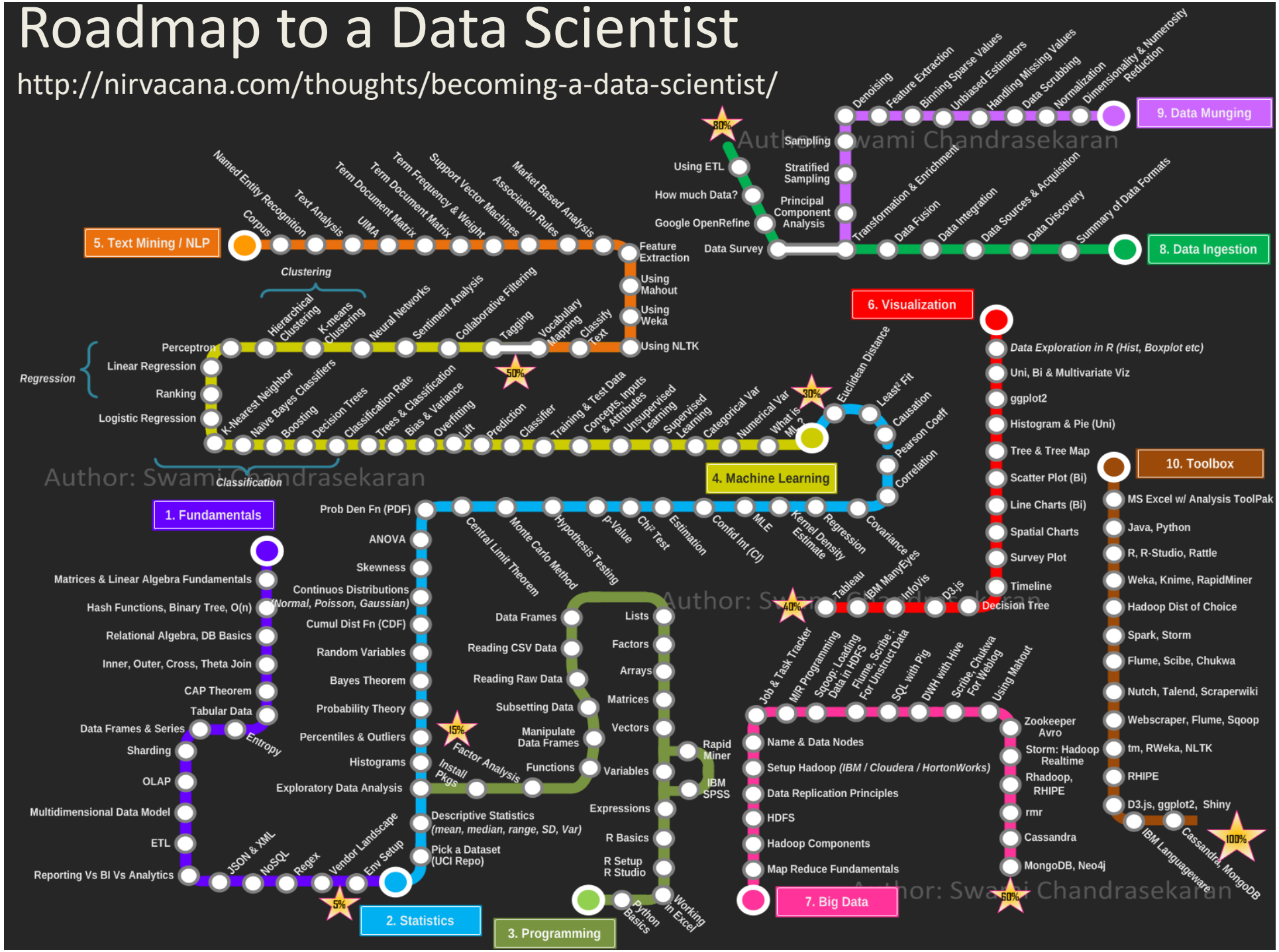
Time Series Data

- Dynamical (streaming) data such as in IoT
  - A sequence of M^{th} order tensors.

Order	1st	2nd	3rd
Correspondence	Multiple streams	Time evolving graphs	3D arrays
Example			

Roadmap to a Data Scientist

http://nirvacana.com/thoughts/becoming-a-data-scientist/



Introduction: ML & Statistics Components

Data Modeling

Supervised Learning
- Naive Bayes Classification
- Linear Regression
- Multinomial Logistic Regression
- Cross Validation
- Elastic Net Regularization
- Decision Tree
- Random Forest
- Linear Support Vector Machines
- Support Vector Machines
- Cox-Proportional Hazards Regression
- Conditional Random Field

Unsupervised Learning
- k-Means Clustering
- Latent Dirichlet Allocation
- SVD Matrix Factorization
- Association Rules
- Low-rank Matrix Factorization

Descriptive Statistics

Sketch-based Estimators
- CountMin (Cormode-Muthukrishnan)
- FM (Flajolet-Martin)
- MFV (Most Frequent Values)

Profile

Quantile

Pearson's Correlation

Inferential Statistics

Hypothesis Tests

Support

Array Operations

Sparse Vectors

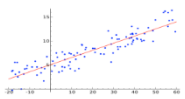
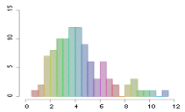
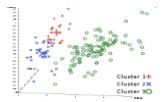
Conjugate Gradient

Probability Functions

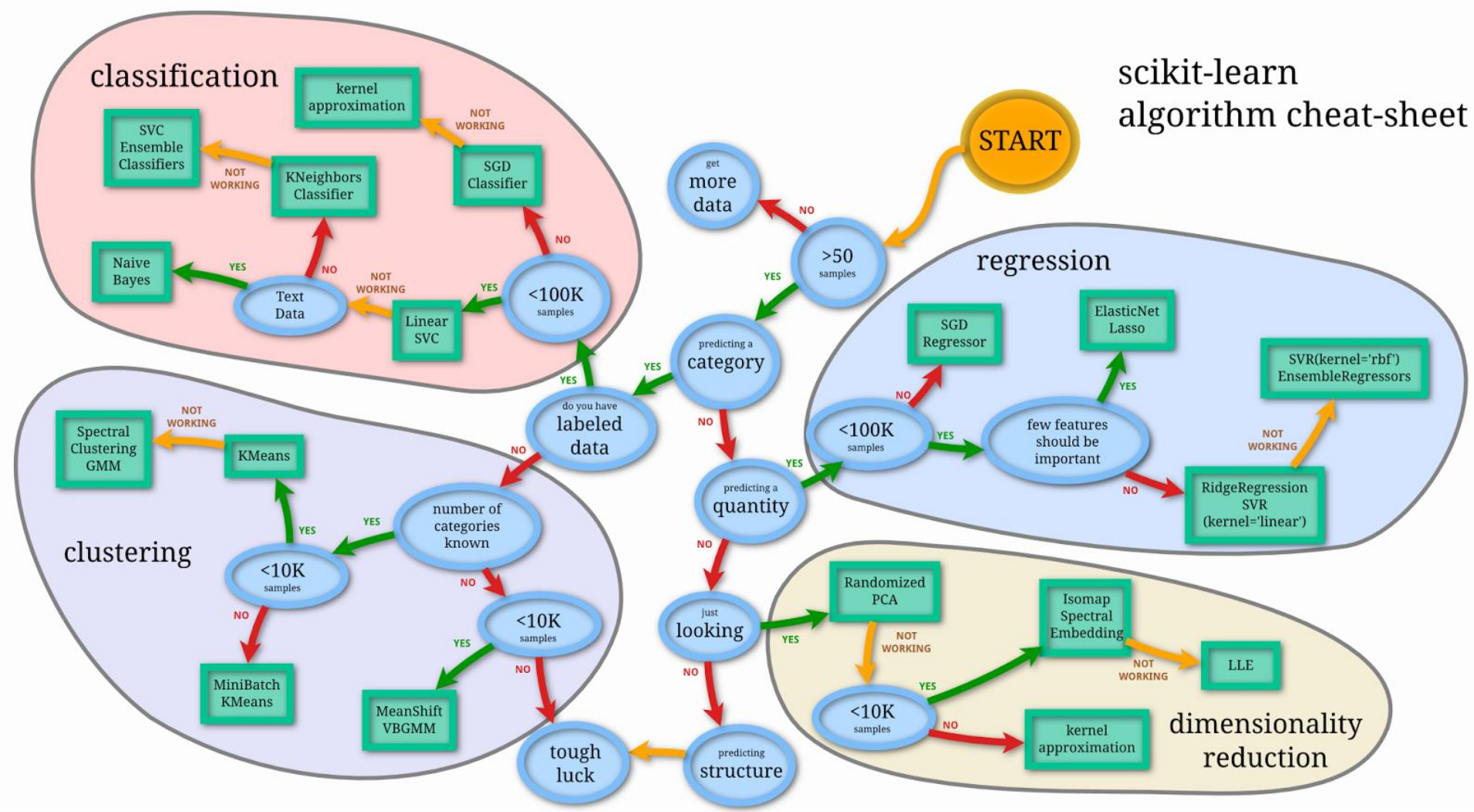
Random Sampling

Linear Algebra Operations

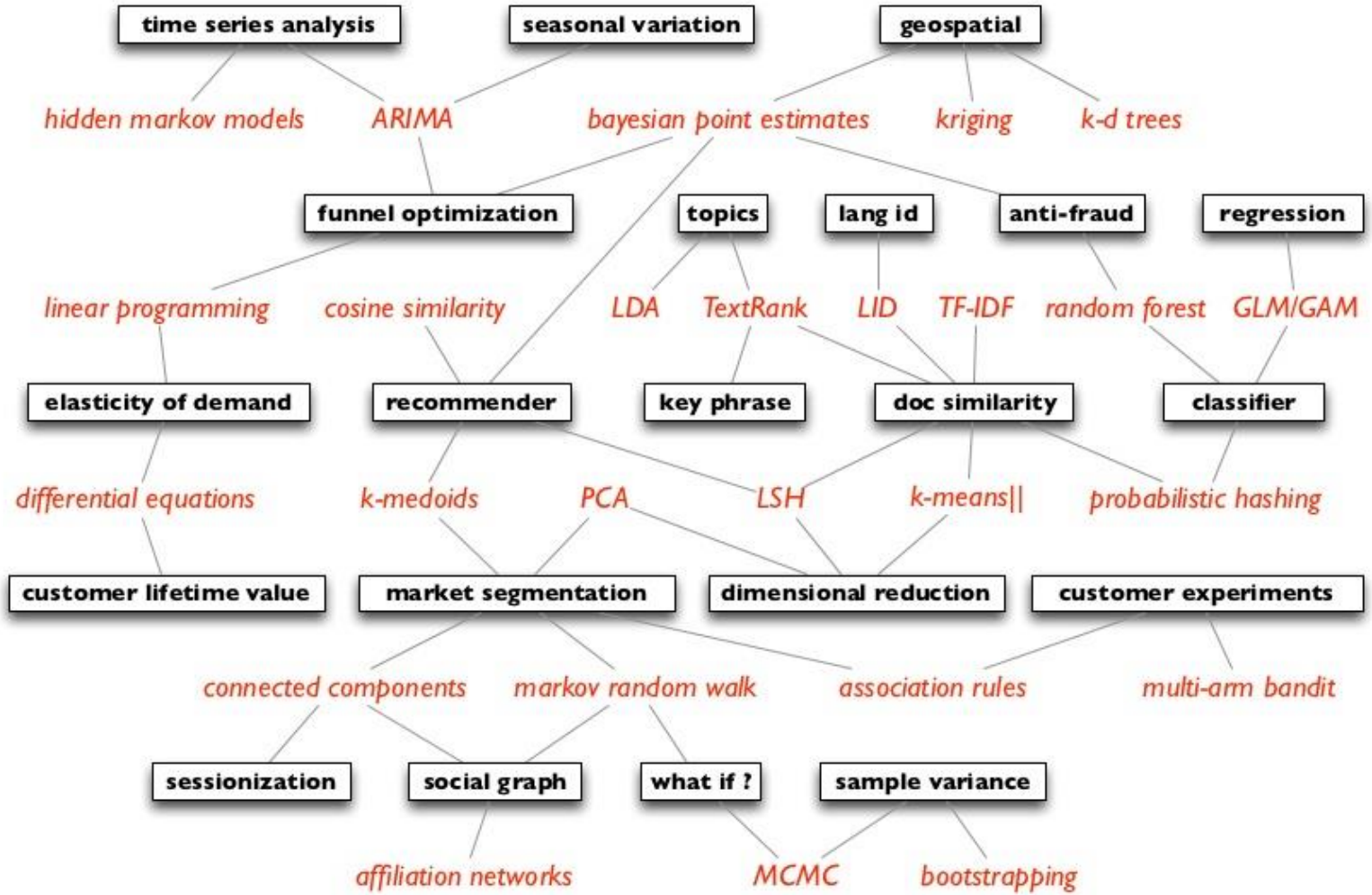
DB Administrator Utilities

Introduction: ML & Data Science Analysis & Techniques



a sample of great algorithms...



Introduction: Working with data

- Lots of digital data generated over the last few years with expectations that more and more data will be generated in the near future.
- Questions arise how to optimally manage it in less time.
- Typically at some point all this unstructured data is reduced to digital text as most practical form.

- **Analytics** - discovery of relevant patterns in data.
- **Mining** - extracting useful information from data.
- Either way the goal is to learn from data.
- In the language of **machine learning** (to learn without explicitly being programmed) these learning categorizations can be subdivided as
  - **Supervised learning** (we know the categories into which we need to separate the data). Task: for new feature x, predict y)
    - Regression (continuous y data predictions from features x)
    - Classification (discrete y data predictions from features x)
  - **Unsupervised learning** (we don't have any knowledge into what categories the data can be subdivided) - find structure in large data set.
    - Clustering

What is an analytics?

- Methodology of revealing a meaningful pattern in recorded information (data) to quantify a performance.
- Relies on the simultaneous application of several steps and techniques including research and data storage managed by a computer system.
  - Database
    - Typically used for large, long-lived data.
  - The knowledge base data storage (called an ontology) is
    - A dynamic resource
    - An object model with classes, subclasses, and instances.
    - Benefits - being able to store, analyze, and reuse knowledge
    - Typically used to arrive at a specific answer to a problem.
- Use of computer systems to implement
  - Statistics techniques
  - Including machine learning
- Analytics is used to drive decision making
  - By identifying which data is useful and meaningful

Components of Analytics

- **Descriptive** analytics – describe what is happening in your system.
  - Simply describes past events and can allow for interpretation in preventing future negative impacts.
- **Predictive** analytics – predict what could happen.
  - Utilizes a variety of statistical, modeling, data mining and machine learning techniques to study historical and recent data.
  - Allows analysts to make predictions about future (positive or negative) events.
  - Currently being able to foresee positive and negative events is extremely powerful feature used to derive marginal advantages over competitors, if not to gain competitive advantage.
  - Predictive analytics takes into account all historical data, allowing for linking of key data points over time to provide predictive features related to operational cost effectiveness and possible downtrends.
- All this is used to prescribe solutions to help mitigate issues along the way.
- This new and growing area of predictive analytics gives the probability of an event and gives the data points needed to mitigate it.
- The combination of analytics and predictive analytics (coding) offers more advanced and cost-effective operation.
- In addition, machine based predictive analytics capabilities can help find the meaning and subset useful data based on input from the user.
- An intuitive predictive coding workflow can validate performance and allow weighing the costs and benefits based on specific measure thresholds (such as precision and recall).
- It can recommend one or more courses of action – and show the likely outcome of each scenario.

Introduction: ML & AI

- A form of artificial intelligence, Machine Learning is revolutionizing the world of computing as well as all people's digital interactions. Machine Learning powers such innovative automated technologies as recommendation engines, facial recognition, fraud protection and even self-driving cars.

- Machine learning is a core sub-area of artificial intelligence; it enables computers to get into a mode of self-learning without being explicitly programmed. When exposed to new data, these computer programs are enabled to learn, grow, change, and develop by themselves. So, to put simply, the iterative aspect of machine learning is the ability to adapt to new data independently.

Introduction: Why ML?

- Machine Learning is taking over the world and with that, there is a growing need among companies for professionals to know the ins and outs of Machine Learning.

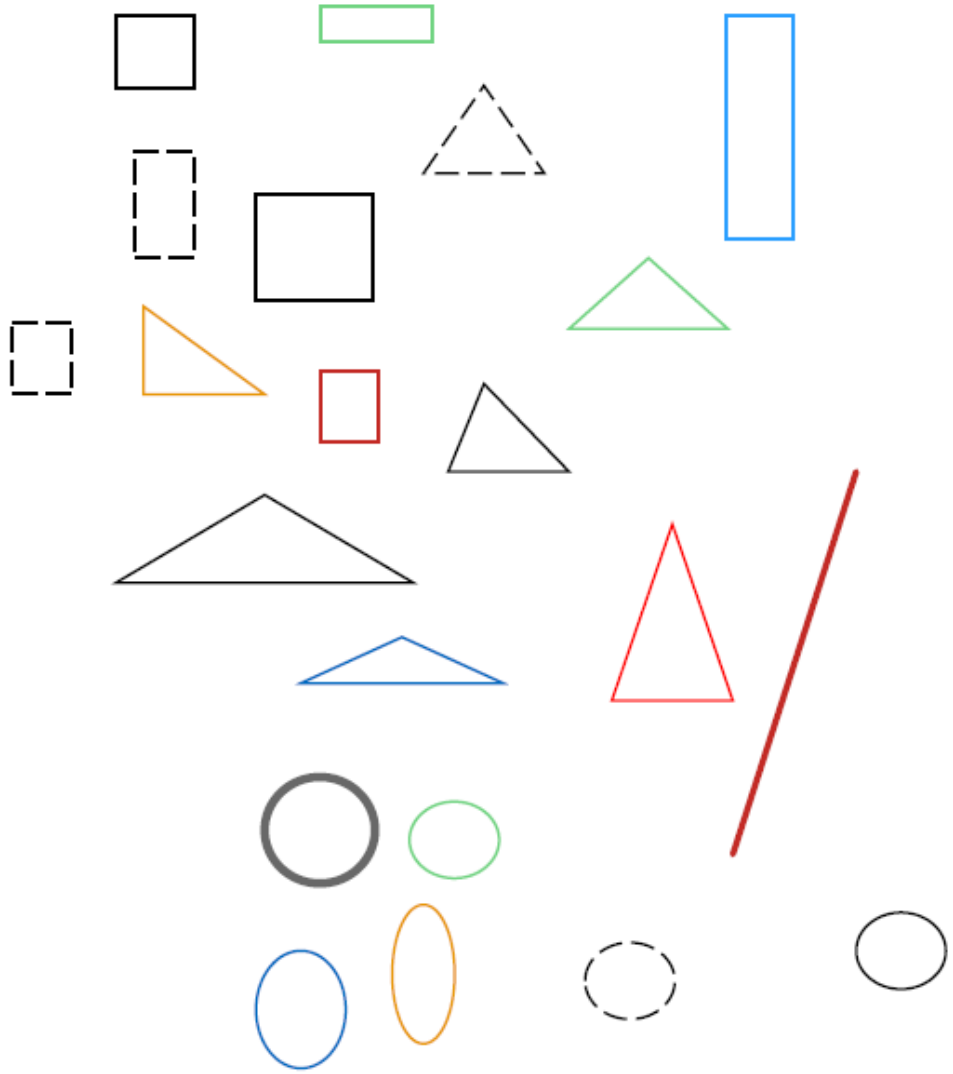
- The Machine Learning market size is expected to grow from USD 1.03 Billion in 2016 to USD 8.81 Billion by 2022, at a Compound Annual Growth Rate (CAGR) of 44.1% during the forecast period.

Introduction: Basic ML Example

OBJECTIVE: - Learn about types of Geometric objects

- Have ability to Machine-Identify Shape Type (Classify)
- What “features” should we use?

- Correlation, causation.
  - Ice cream, Montreal…



Introduction: ML & AI

Artificial Intelligence

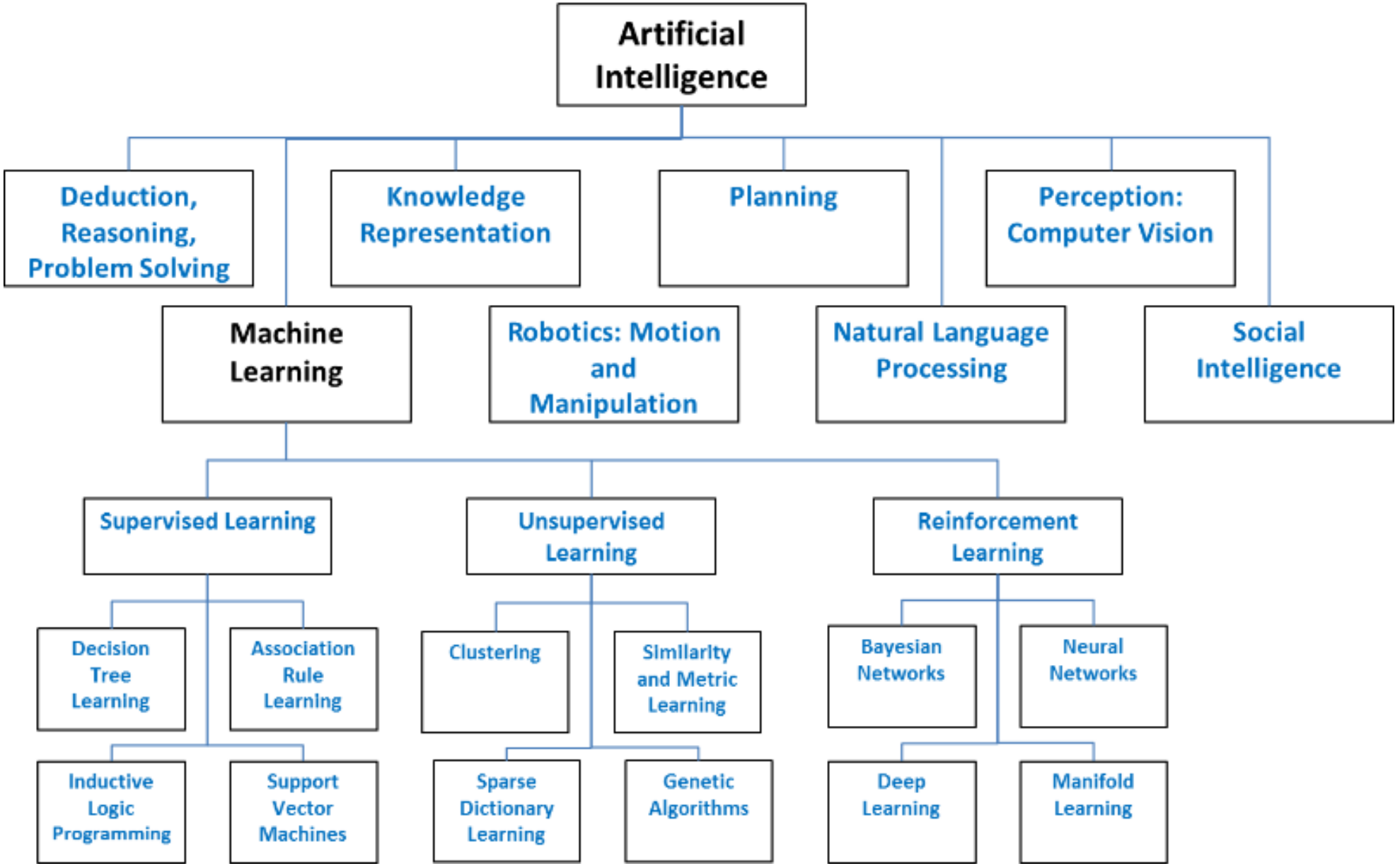
- Expert Systems/Symbolic AI

Machine Learning

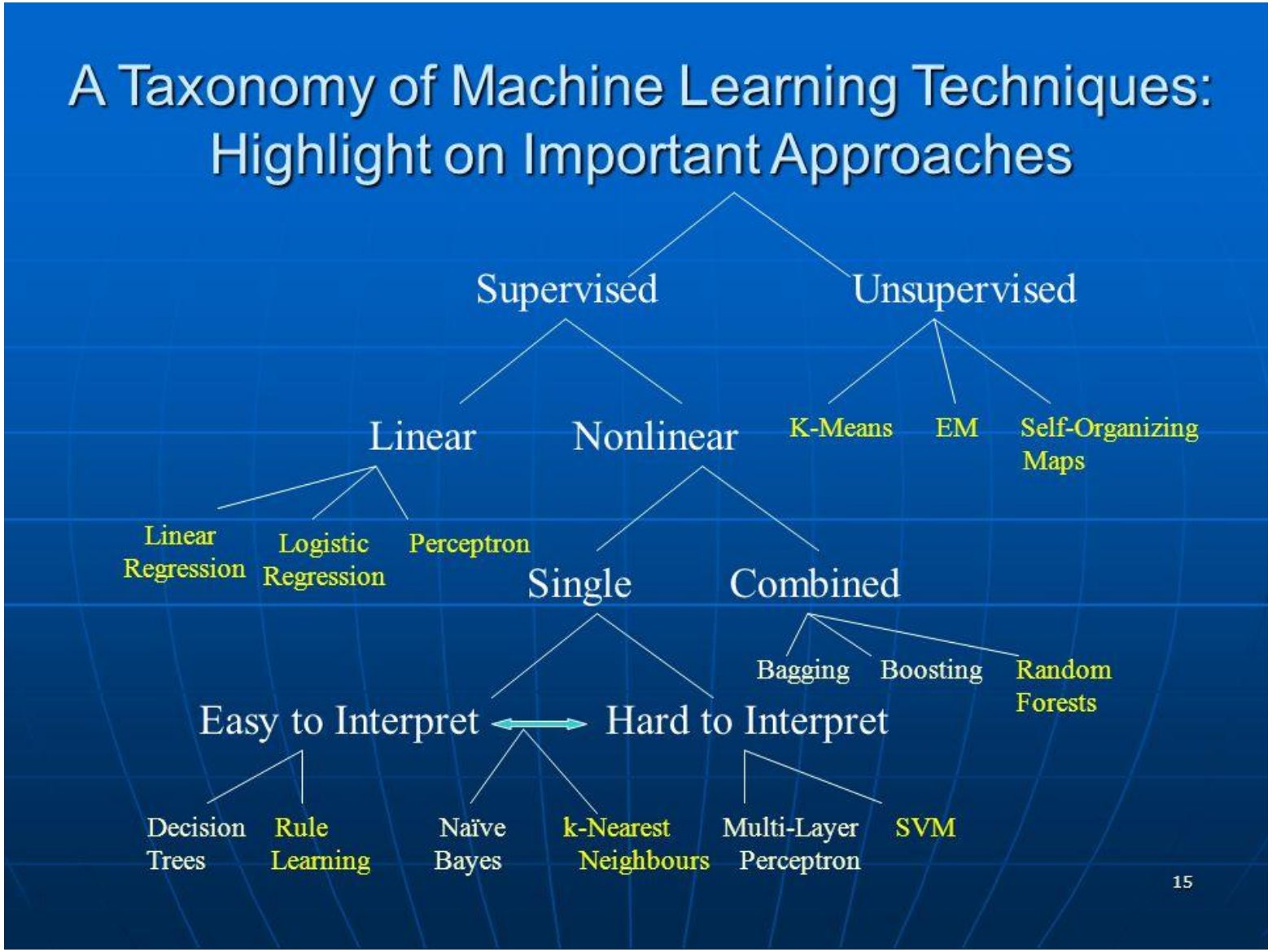
- Genetic Algorithms
- Statistical/Probabilistic
- Analogical/Clustering
- Artificial Neural Networks
  - Shallow Neural Network
  - Deep Neural Network/“Deep Learning”

Introduction: ML & AI

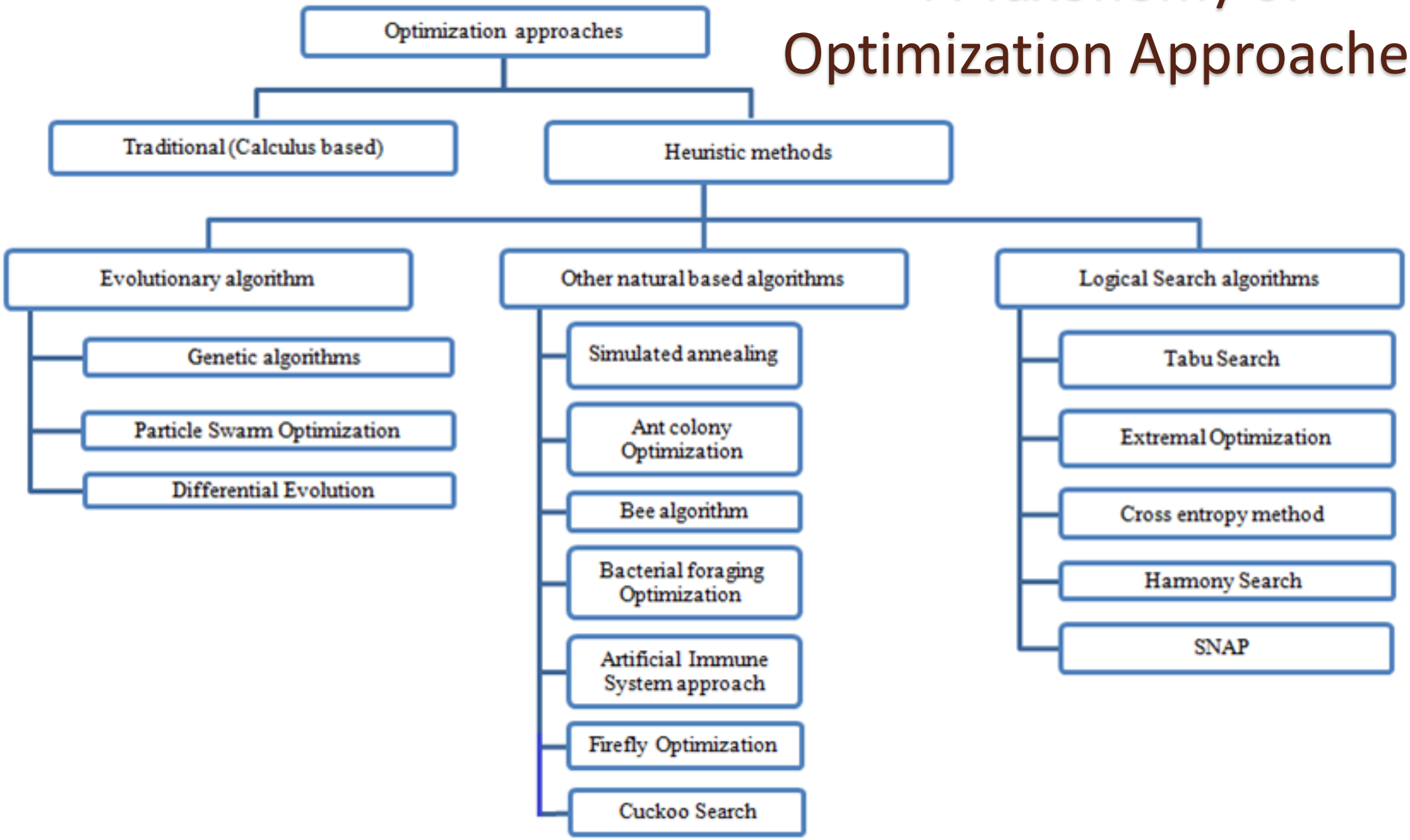
- **Artificial Intelligence** – Intelligence demonstrated by machines as opposed to natural intelligence.
  - **Expert Systems/Symbolic AI** – Domain specific set of **hand-crafted rules** chosen from human experts in the domain.
  - **Machine Learning** – **learns** to execute a task by examples, as opposed to operating via handcrafted rules. Mainly used for **classification, value estimation, clustering, and skill acquisition**. The learning can be performed in a **supervised, unsupervised, or reinforcement** manner.
    - **Genetic Algorithms** – Learns via a process of artificial "evolution".
    - **Statistical/Probabilistic** – Bayesian Learning tunes a prior hypothesis based on sample evidence.
    - **Analogical/Clustering** – Groups similar data together by smart feature selection.
    - **Artificial Neural Networks** – Modeled on the brain, a parametric model whose weights/neural pathways are strengthened/weakened by "training" on data examples.
      - **Shallow Neural Network** – A neural network with two or three "**layers**" of neurons. It learns to identify the most salient immediate "features" of the input data.
      - **Deep Neural Network** – Achieves "Deep Learning" by **stacking many layers (called encoders)** of neurons, which learn increasingly sophisticated abstractions of the input data. The most advanced form of artificial intelligence at the moment.


Artificial Intelligence
Deduction, Reasoning, Problem Solving
Knowledge Representation
Planning
Perception: Computer Vision
Machine Learning
Robotics: Motion and Manipulation
Natural Language Processing
Social Intelligence
Supervised Learning
Unsupervised Learning
Reinforcement Learning
Decision Tree Learning
Association Rule Learning
Clustering
Similarity and Metric Learning
Bayesian Networks
Neural Networks
Inductive Logic Programming
Support Vector Machines
Sparse Dictionary Learning
Genetic Algorithms
Deep Learning
Manifold Learning

Machine Learning Taxonomy



A Taxonomy of Optimization Approaches



Ideas from 1998 Work

- John Holland's Work
  - Genetic Algorithms
  - Classifier Systems
  - Bucket Brigade Algorithm (Credit Assignment)
- Reinforcement Learning
- Intelligent Agents
- Belief-Desire-Intention (BDI) Model of Agency
- Tuple Spaces
  - LINDA – David Gelernter, Yale
  - JINI, JavaSpaces
- Parallel Algorithms
- 'Blackboard' Architecture
  - *Pattern-Oriented Software Architecture*, Buschmann et. al.

Ideas from Today

- Private/Public Cloud
  - Containers, Clustering

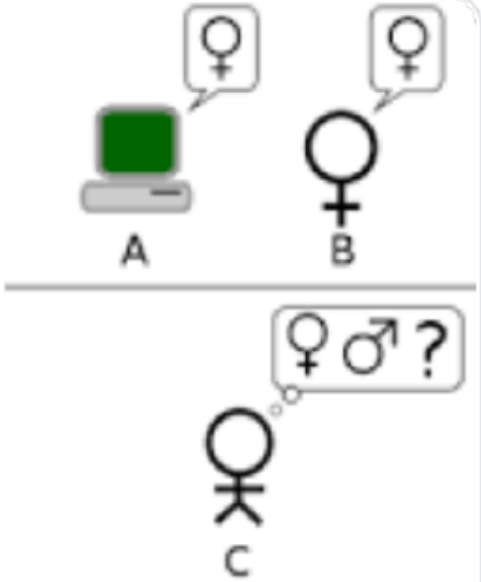
Inexpensive Computing, Storage

- SPARK, Spark Streaming
- Stream Processing
  - Kafka, Storm, Heron, Flink, etc.
- In-Memory Data Grids
  - Apache Ignite, Geode, Hazelcast, etc.
- Actor Systems
  - Akka, Akka .NET
- Machine Learning Libraries, Reactive ML
  - MLLib, Mahout, H2O
- Mulit-objective Optimization Algorithms

+ Realization – Answers don't have to be 'perfect' – just good enough, delivered quickly (and through ML they will improve over time…)

NLU & Turing Test

The **Turing test**, developed by Alan **Turing** in 1950, is a **test** of a machine's ability to exhibit intelligent behavior equivalent to, or indistinguishable from, that of a human.

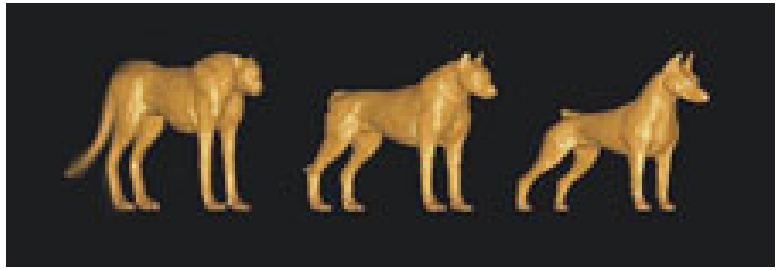


Turing test - Wikipedia

https://en.wikipedia.org/wiki/Turing_test

Prize for Cat-Dog image recognition algorithm

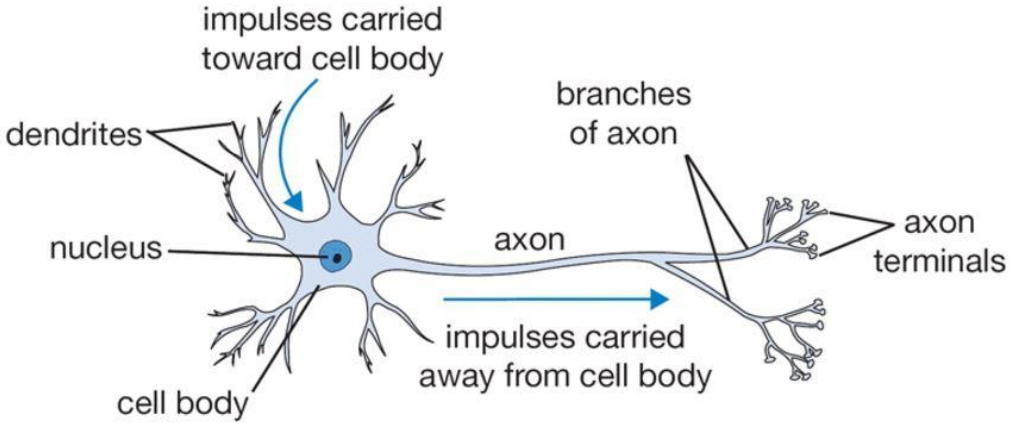
- Any child can tell a cat from a dog.
- David J. Freedman explored this question by training monkeys (Nature 409, 300 (18 January 2001))
- Even when the image was 60% cat and 40% dog, the monkeys reliably reported that it was like a cat.
- Surprisingly, they found category information represented in the lateral prefrontal cortex at the level of single neurons.
- Individual neurons responded similarly regardless of the image being 60%, 80% or 100% dog, but they responded differently for 60%, 80% or 100% cat.
- Abstract category representations in the brain can rapidly be changed – this is what we call learning.
- By 2012, the ML technique called Deep Learning (DL) was able to over 90% of accuracy in making a distinction between cat and dog images.

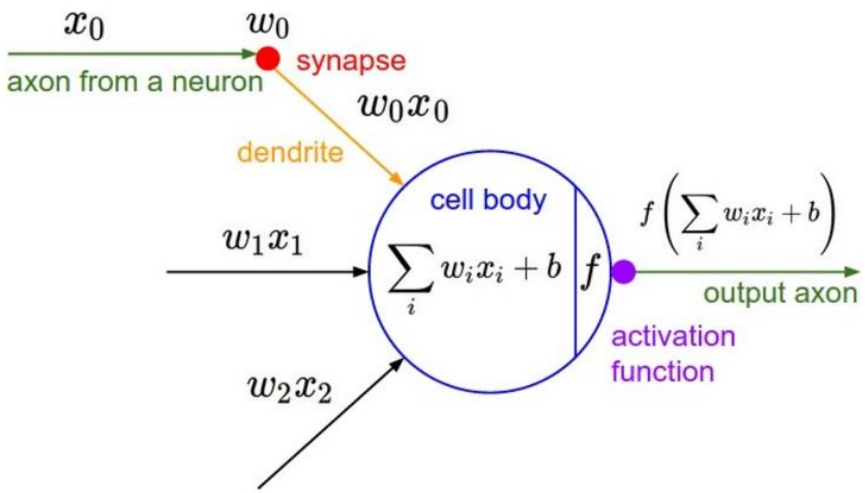


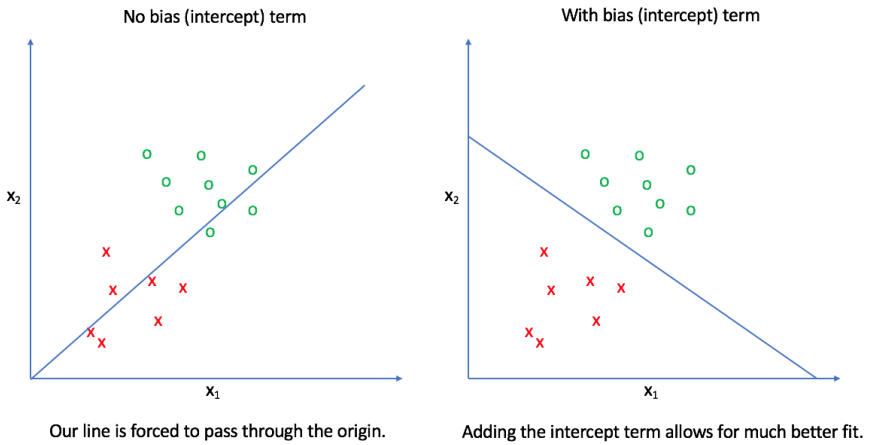
Inspiration

Neural networks are a biologically-inspired

- They attempt to mimic the functions of neurons in the brain.
- Each neuron acts as a computational unit, accepting input from the **dendrites** and outputting signal through the **axon** terminals. Actions are triggered when a specific combination of neurons are **activated**.
- One way to think of the neuron is as a linear function $\boldsymbol{y = kx + b}$ where $\boldsymbol{b}$ is bias.





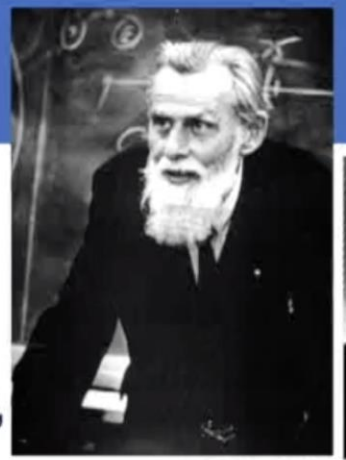


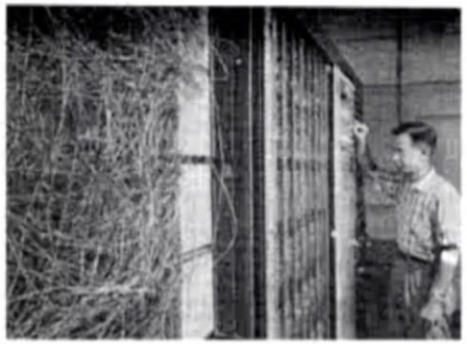
The History of Neural Networks

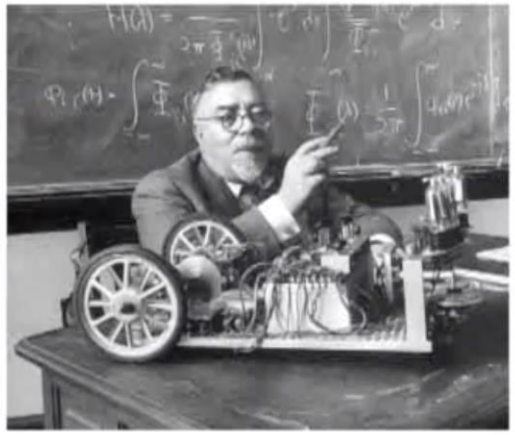
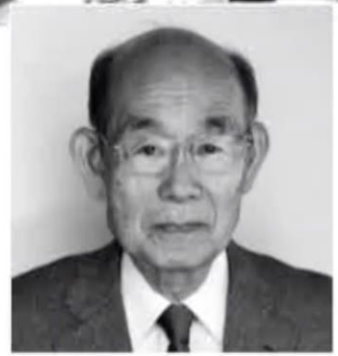
Inspiration for DL: The Brain!

Y. LeCun

- McCulloch & Pitts (1943): networks of binary neurons can do logic
- Donald Hebb (1947): Hebbian synaptic plasticity
- Norbert Wiener (1948): cybernetics, optimal filter, feedback, autopoïesis, auto-organization.
- Frank Rosenblatt (1957): Perceptron



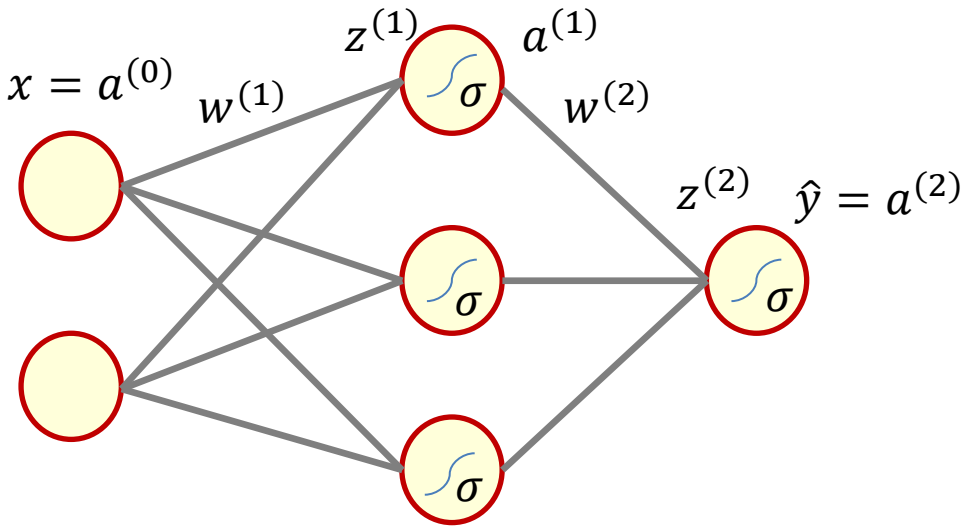



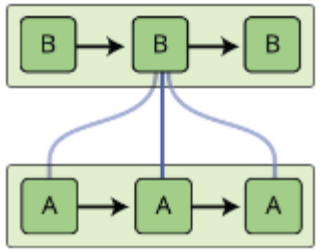
Historical Perspective of Neural Nets & DL

- Deep Learning (DL) Neural Net is not a copy of the brain, but it was certainly inspired by Neuroscience, that goes back to the works in 1940's.
- The main idea is the relationship between the computation a neuron (1943) can do and doing logic with neurons.
  - This idea that might not have been right, captured the imagination of many people.
  - Psychologists like Donald Hebb (1947) hypothesized how synaptic efficacy can change the brain and cause learning.
  - Methods invented by Norbert Wiener, the father of cybernetics (1948), inspired a whole branch of AI (adaptive machine learning) are used in Neural Nets today.
- The first concept - Perception (in 1957 by Rosenblatt)
  - This was a program on an analog computer (electronics).
  - Each module on the slide is a tunable weight of the Neural Net. The weight was a tunable potentiometer changed by a motor by a learning algorithm.
  - These few thousands weights were the first layer of the Neural Net in which the nodes were randomly (spaghetti like) connected.
  - Today this code can be essentially replaced by 3 lines of python

The concept of adjusting the weights in a Neural Net

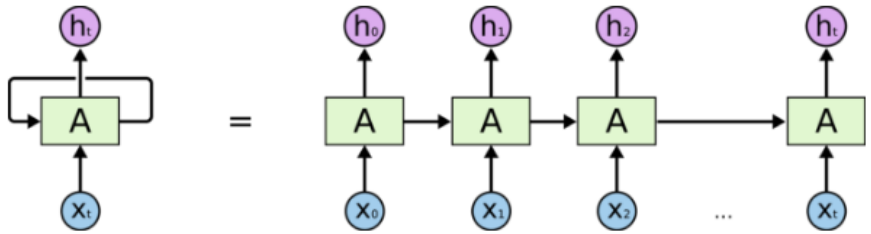
- The goal in supervised learning is how do you adjust the weights from a number of examples so that the system can learn how to classify the correct category.
- Back in the 1943 McCulloch–Pitts (MCP) introduce the first artificial binary neuron.
  - They called it “A logical calculus of the ideas immanent in nervous activity”.
- In the 1950's & 1960’s there were several modals that were based on adjusting the weights of a Neural Net.
  - **Perceptron**, developed in 1958, was a linear, binary classifier algorithm for supervised learning.
    - Intended to be a machine, rather than a program, first implemented as a software for the IBM 704, later as a hardware as "Mark 1 perceptron".
    - Initially designed for image recognition - it had an array of 400 photocells, randomly connected to the "neurons".
    - Weights were encoded in potentiometers, and weight updates during learning were performed by electric motors.
  - **ADALINE** (Adaptive Linear Neuron), developed in 1960, is a single-layer model of a Neural Net that uses memristors.
    - Single layer neural network with multiple nodes where each node accepts multiple inputs and generates one output ($y = \sum_{i=1}^{m} x_i \cdot w_i + \theta$)
    - Unlike the transistor, memristor’s conductance between two of the terminals is controlled by the time integral of the current in the third terminal, rather than its instantaneous value as in the transistor.
- The idea of supervised learning behind the Perception is the following:
  - Learning achieved by a two valued function (1 or 0) defined as
    - $f(x) = \begin{cases} 1 & if\ \boldsymbol{W} \cdot \boldsymbol{x} + b > 0 \\ 0 & otherwise \end{cases}$ where b is a bias and $\boldsymbol{W} \cdot \boldsymbol{x} = \sum_{i=1}^{m} x_i \cdot w_i$ is a dot product of the weights and the m inputs x_i.
  - We have some representation of the input feature vector $\boldsymbol{x}$, which is obtained through some hard-wired extractor.
  - We compute $\boldsymbol{W}$ - the weighted sum of these features, and we compare this weighted sum to some threshold.
    - If the sum $\boldsymbol{W}$ is above the threshold we have category A.
    - If the sum $\boldsymbol{W}$ is below the threshold we have category B.



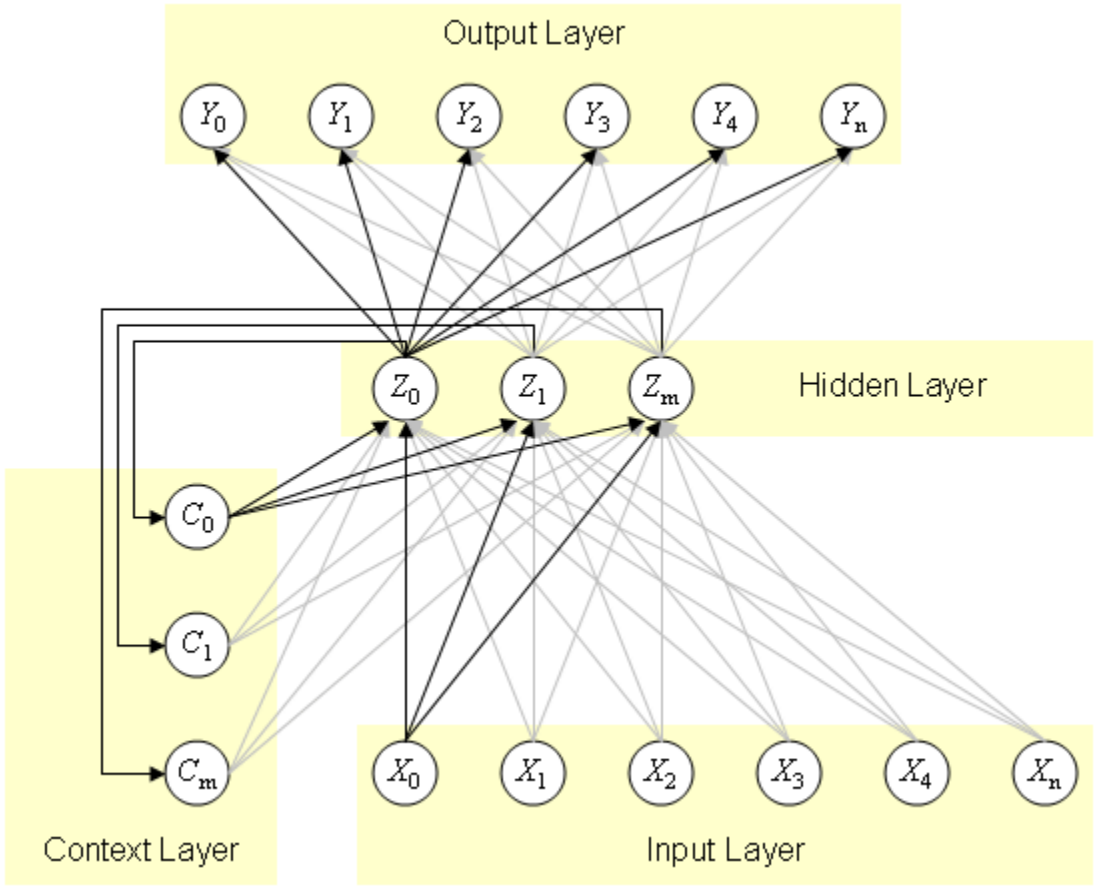


Recurrent Neural Networks

- Attention and Augmented Recurrent Neural Networks work with sequences of data like text, audio and video.
- Contain Context layer
- A special variant – “long short-term memory” LSTM networks
- Very powerful, remarkable results in many tasks including
  - translation,
  - voice recognition, and
  - image captioning.



An unrolled recurrent neural network.



Elman's RNN

COGNITIVE SCIENCE **14,** 179–211 (1990)

Finding Structure in Time

JEFFREY L. ELMAN
University of California, San Diego

- Introduced by Elman in 1990 with intention to analyze language.

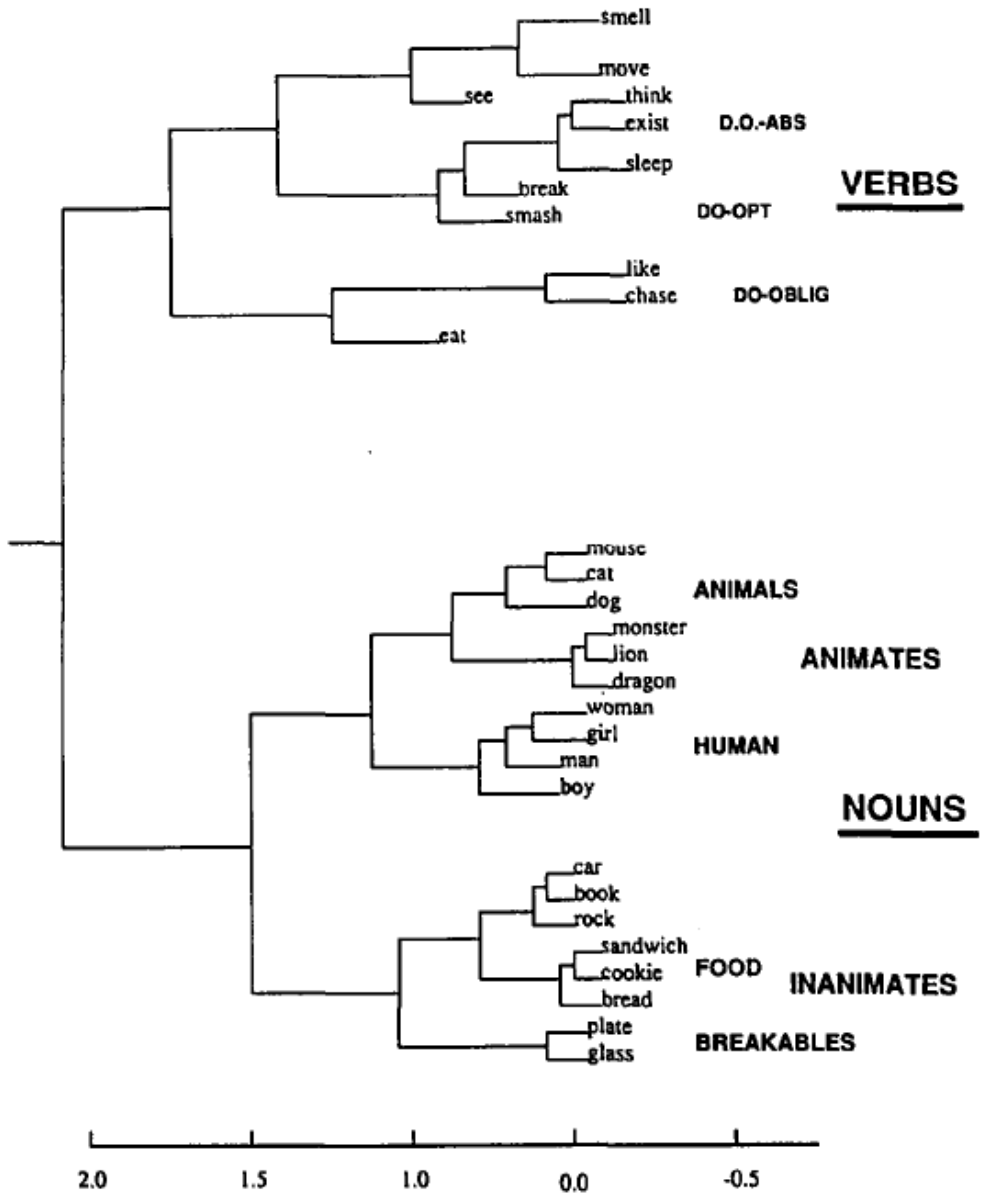


Figure 7. Hierarchical cluster diagram of hidden unit activation vectors in simple sentence prediction task. Labels indicate the inputs which produced the hidden unit vectors; inputs were presented in context, and the hidden unit vectors averaged across multiple contexts.

Introduction: DEEP LEARNING

- Much of Machine Learning can be seen as the process of discovering a simpler form underlying the complex surface form of raw data. Deep learning in particular can be interpreted as a mathematical means of successively determining a hierarchy of simpler forms (manifolds) within the data.

- A useful visual analogy: imagine a few pieces of paper with writing on them stacked together and then crinkled up into a ball. Deep learning is able to rediscover the original shapes of the papers and thereby gain access to the information written on them.

NLP, NLU, NLG

NLP (Natural Language Processing), NLU (Natural Language Understanding), NLG (Natural Language Generation)

Amazon Rekognition - API to automatically identify objects, people, text, scenes, and activities, as well as detect any inappropriate content. Analyze facial attributes, identify objects, scenes & activities, detect & extract text in images.

NLP & Text Mining

- **Knowledge Graphs** and underlying graph databases - We use them every day for example
  - As voice assistants (such as Alexa, Siri or Google Assistant)
  - Intuitive search results
  - Personalized shopping experiences through online store recommenders
  - KG got popularized by Google's introduction of their KG in 2012 to augment their search results.
  - A model of a knowledge domain created by an expert with the help of intelligent machine learning algorithms.
  - It resides on top of existing databases.
  - Provides data structure enabling the creation of smart multilateral relations throughout a database.
    - KG can combine disparate Data Silos
    - Bring together structured and unstructured data
    - Improve decisions by faster search

NLP History

NLP – Natural Language Processing, rule based, heuristic techniques up until 2013.

Word2Vec – Introduced in 2013, changed how people thought about NLP.

- Literarily turns words into numerical representation (vectors) that allows to apply math operations to words (vectors).
- Similar words cluster together so we can find cosine similarity between their vector representations.
- For example: King – Man + Woman = Queen
- Learning representations rather than applying rules to raw text

FastText – Developed in 2015, extended W2V model by treating each word vector as a sum of series of character level vectors.

- 3-gram representation of “going” as go-goi-oin-ing-ng
- Allowed more robust representation of rare words (not in training data)

Transformers – Google released this new architecture in 2017, using a mechanism of attention.

- The slogan for this new architecture was: “Attention is all you need.”
- State of the art in automatic language translation
- Needed lots of computing power and lots of data to train on.

NLP History

BERT – Google released in 2019. Learns the full context of the word into account (from both ends of a sentence).

- Similar to how CNN changed computer vision field in 2012, BERT was the most important change in NLP history.
- Trained on massive amount of data, so it can be used for huge NLP tasks without requiring training on huge datasets.
  - Automatic (Machine) Translation
  - Question and Answering Systems
  - Text Summarization and Categorization
  - Sentiment Analysis

In 2019 variety of alternative BERT-like models were released from other companies.

- Some using Sesame street names such as: Erni, Elmo
- RoBERTa, XLNet,
- GPT (produces human like text)
- BioBERT a BERT model re-trained on medical domain knowledge from experts and includes
  - Based on symptoms provided, gives automatic diagnostics and possible treatment options.

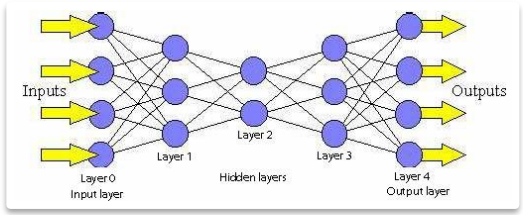
Reformer - Google released in 2020. Re-engineered Transformer architecture to be much more efficient in terms of speed & storage. They used 2 techniques:

- **Locality Sensitive Hashing** (**LSH**) attention.
- **Reversible layers**.

NLP & Text Mining

Standard NLP tools

- Deep Learning technologies
- Knowledge Graph (extend text corpora)
- Indexing
- Taxonomy for classification from corpora
- Single- and multi-document summarization



Deep Learning Neural Networks – Various Network Architectures

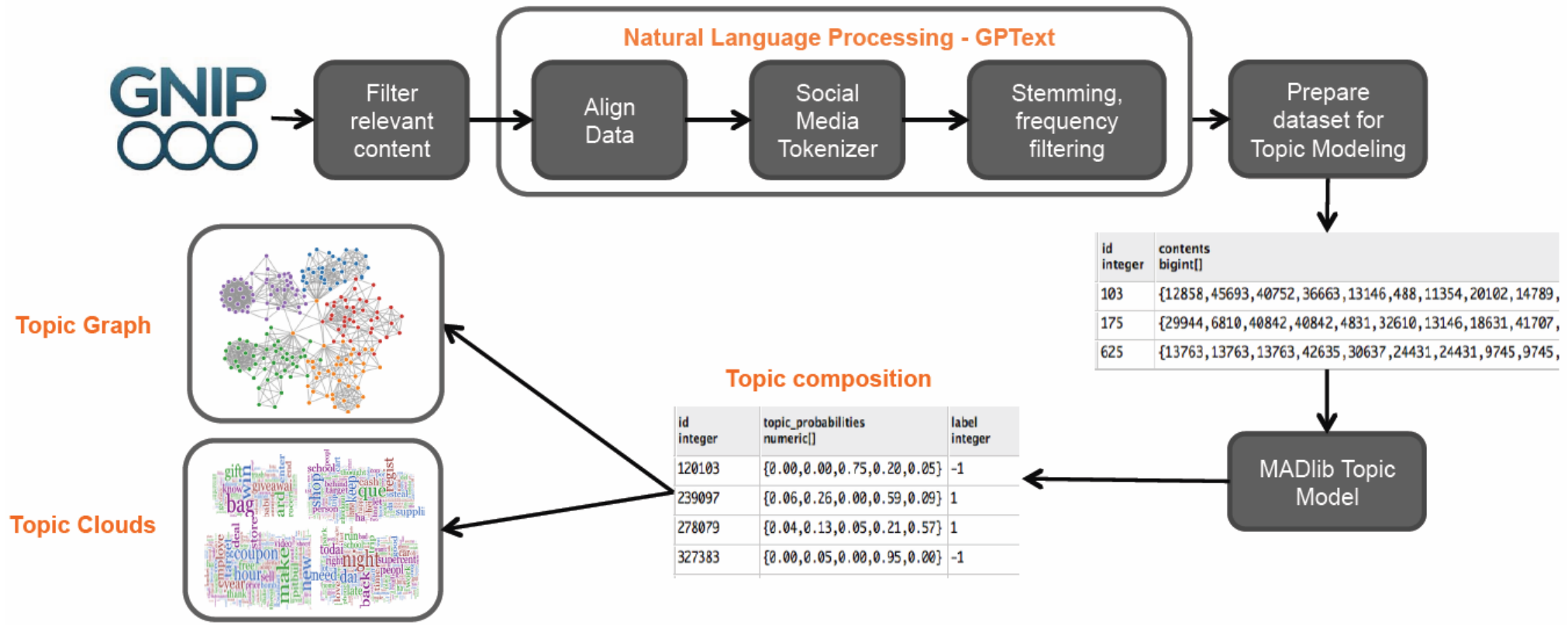
Step 1 – 'Standard' NLP

- Sentence Identification
- Named Entity Recognition (1st Pass)
- Tokenization
- Part of Speech Tagging

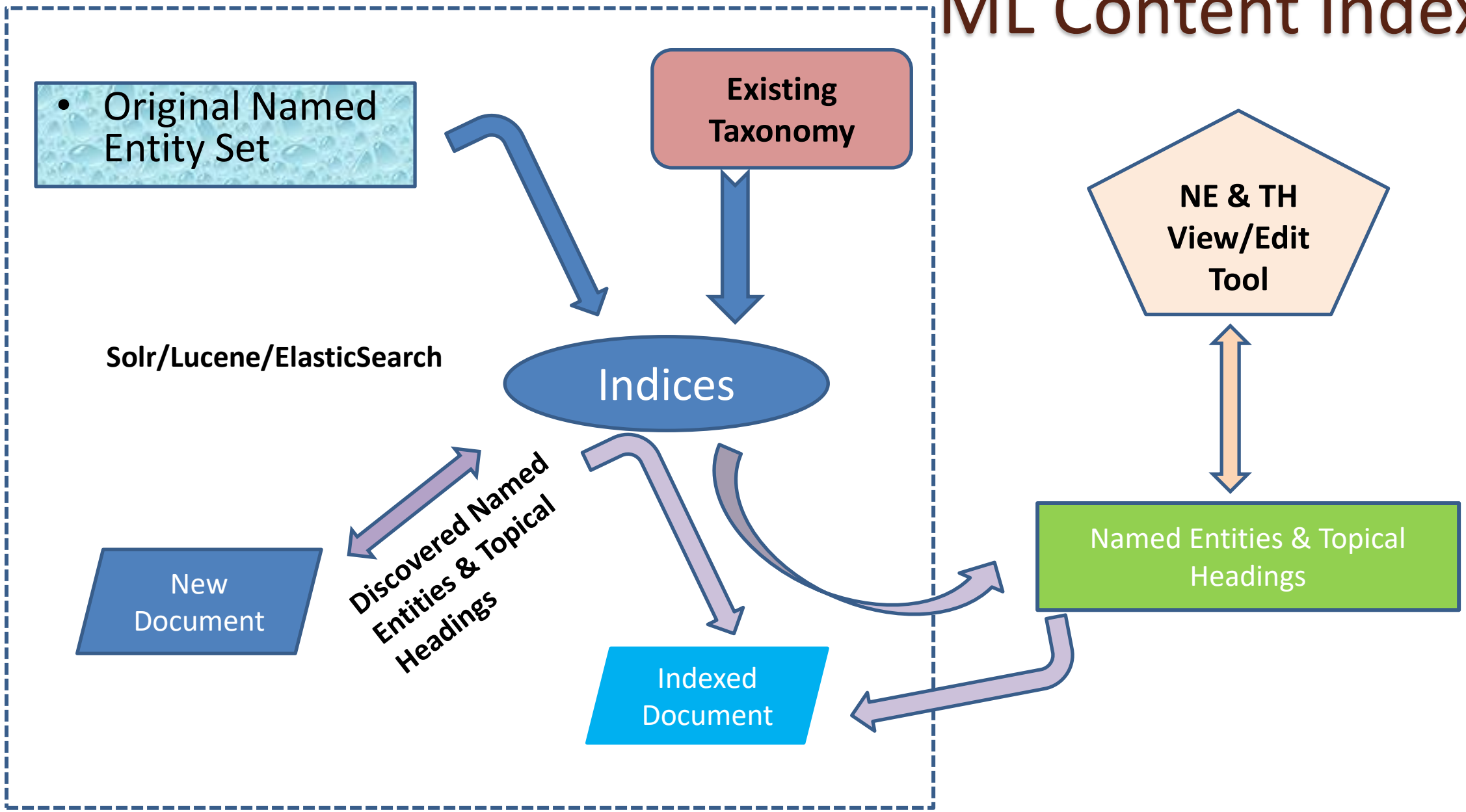
Step 2 – Deep Learning NLP

- Word/Sentence Vectorization
- Named Entity Recognition (2nd Pass)
- Entity Extraction
- Keyword/Keyphrase Extraction
- Relation Extraction

Topic Analysis



ML Content Indexing

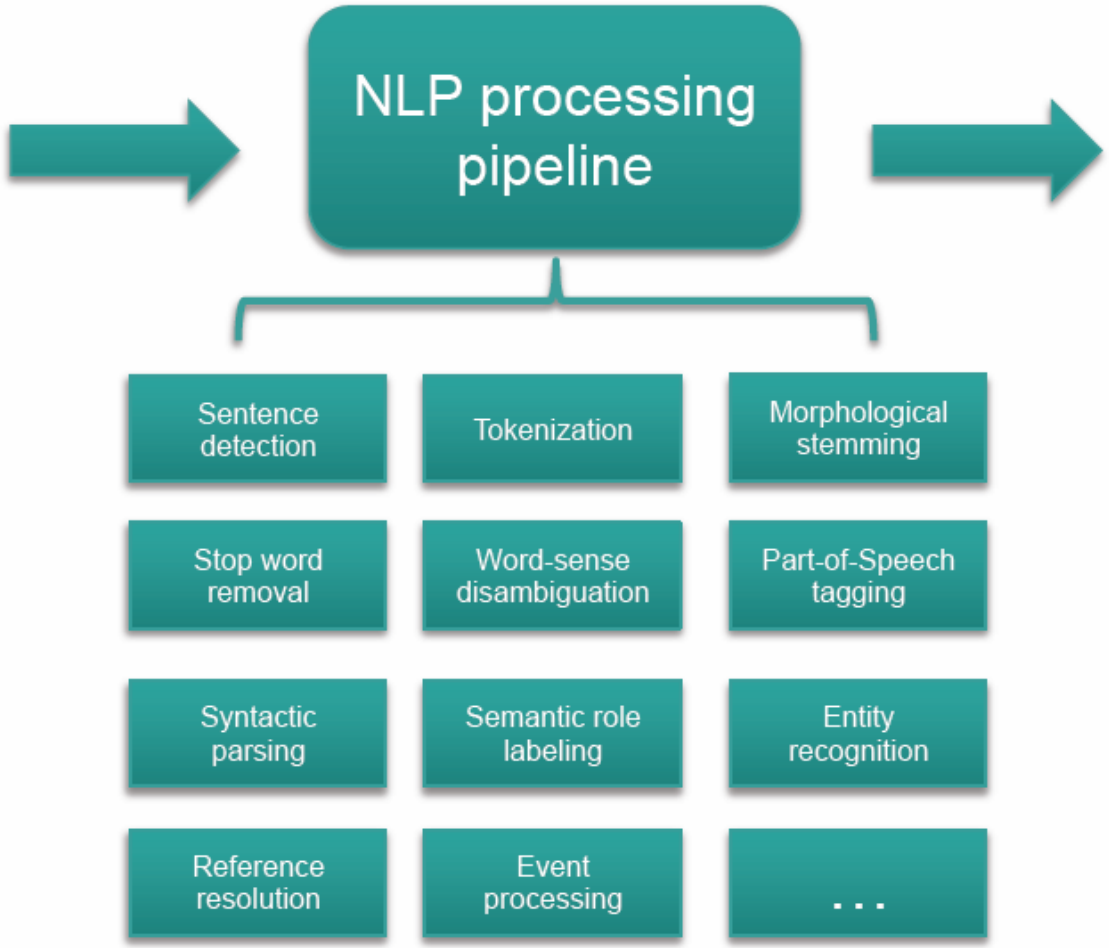


Natural Language Processing

Data sources

Text sources
Documents, books, emails

Speech
Phone logs, conversations



Common tasks/tools in NLP

Applications

Word clouds
Topic modeling
Sentiment analysis
Machine translation
Document classification
Document summarization
Language generation
Search
Question answering
Information Extraction

Open-Source Tools for NLP