INDIAN INSTITUTE OF INFORMATION TECHNOLOGY

Dharwad
ज्ञानेन विकासः

Bioinformatics and Systems Engineering

Dr. Animesh Chaturvedi

Assistant Professor: IIIT Dharwad

Young Researcher: Heidelberg Laureate Forum

Postdoc: King's College London & The Alan Turing Institute

PhD: IIT Indore MTech: IIITDM Jabalpur

HEIDELBERG LAUREATE FORUM

KING'S College LONDON

Indian Institute of Technology Indore
भारतीय प्रौद्योगिकी संस्थान इंदौर

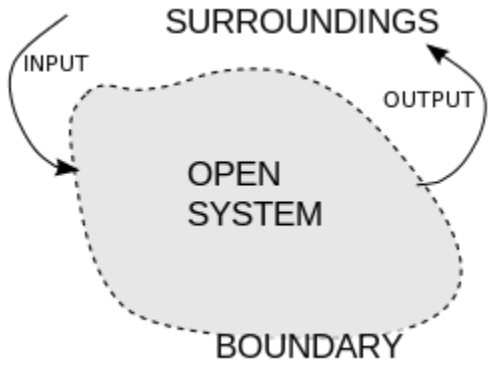
The Alan Turing Institute

PDPM
Indian Institute of Information Technology, Design and Manufacturing, Jabalpur

Systems Engineering

Systems Engineering

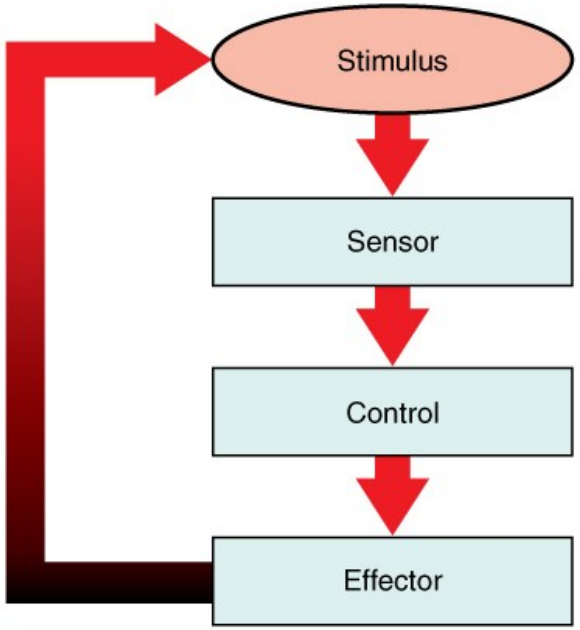
- An interdisciplinary domain
- Composed of many components (or entities) interacting with each other.
  - power grid, transportation or communication systems,
  - social and economic organizations (like cities),
  - organisms, a living cell, the human brain, and
  - an ecosystem, climate, entire universe.
- Behavior is hard to model with dependencies, relationships,
  - interactions between their components or
  - interactions between system and its environment

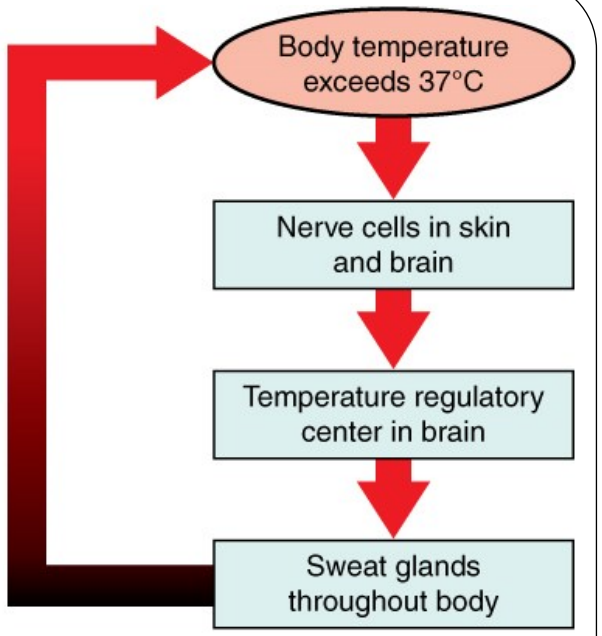


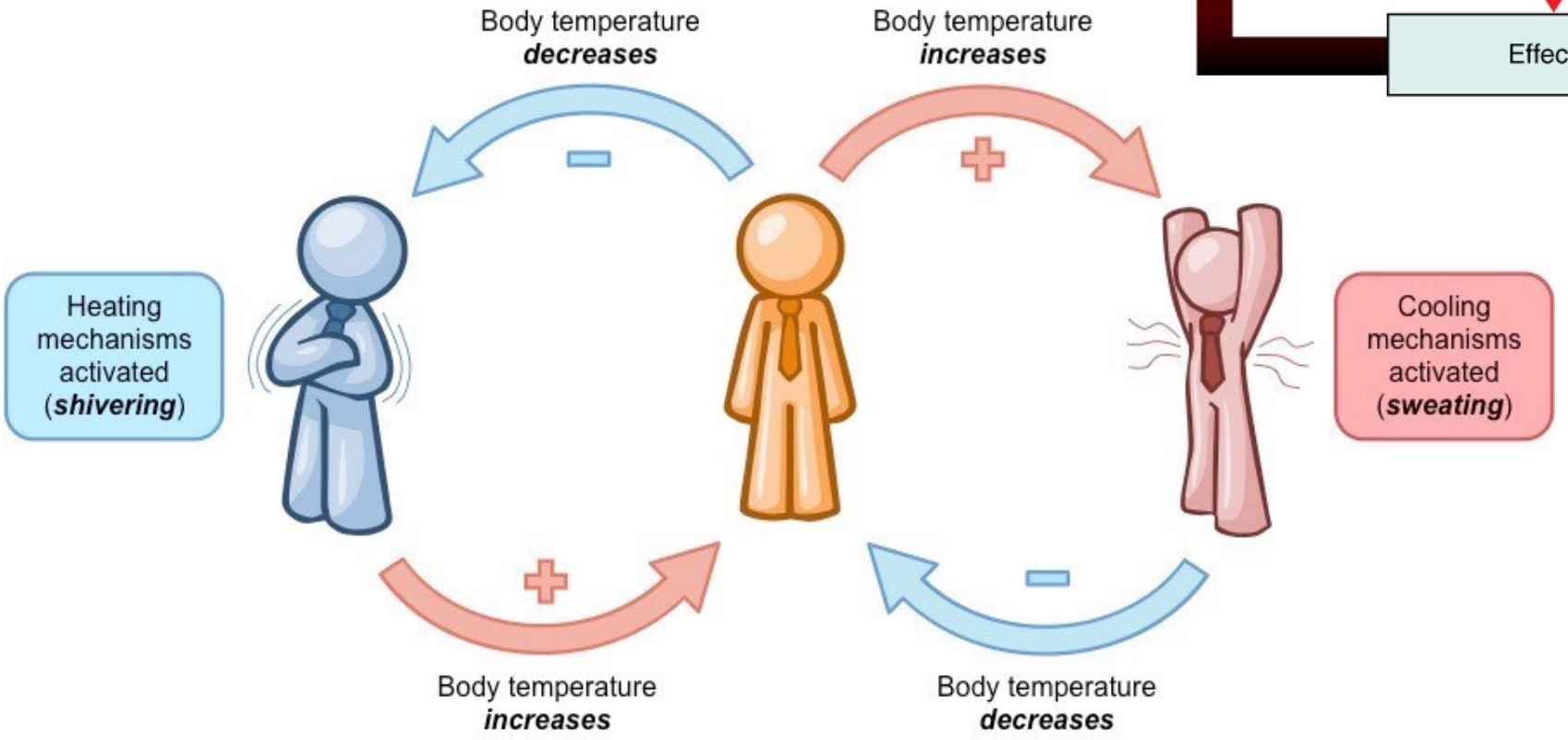
https://en.wikipedia.org/wiki/Complex_system

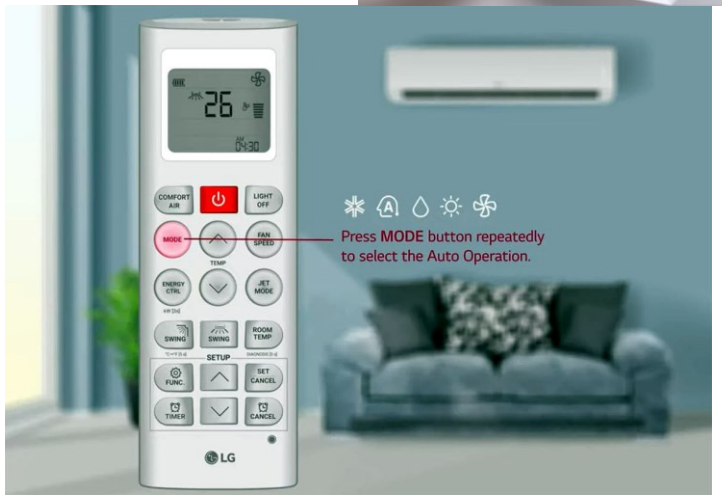
Fundamental Concepts

- **Adaptation:** Tendency of making internal changes to protect itself and to maintain functionalities.





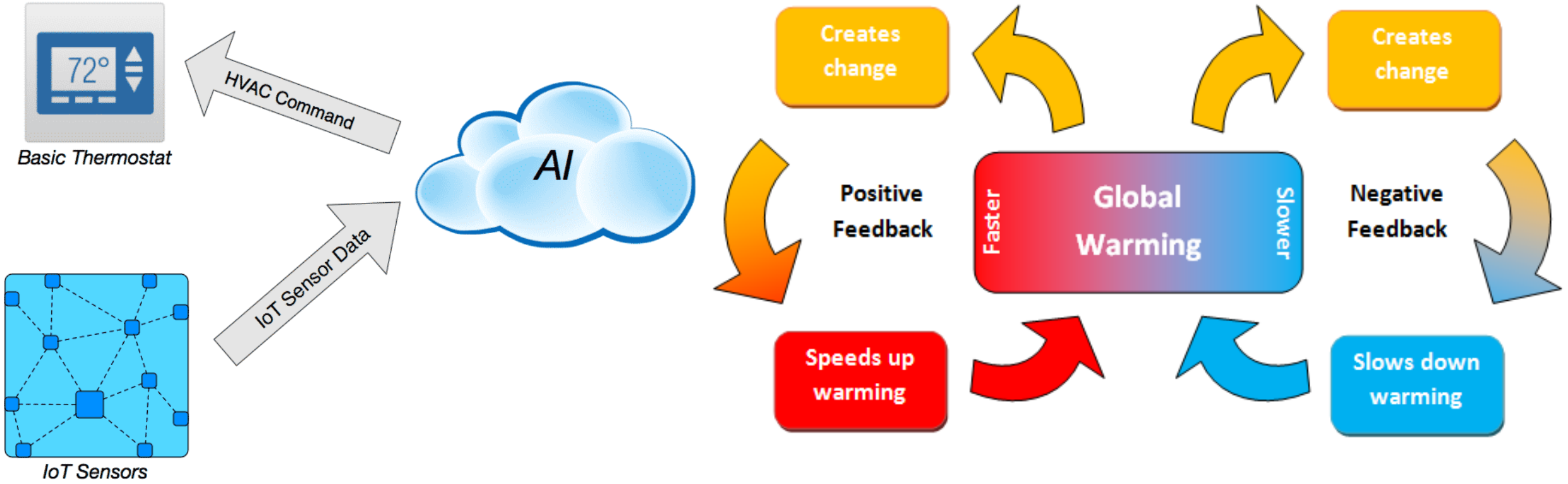




https://en.wikipedia.org/wiki/Systems_theory

Fundamental Concepts

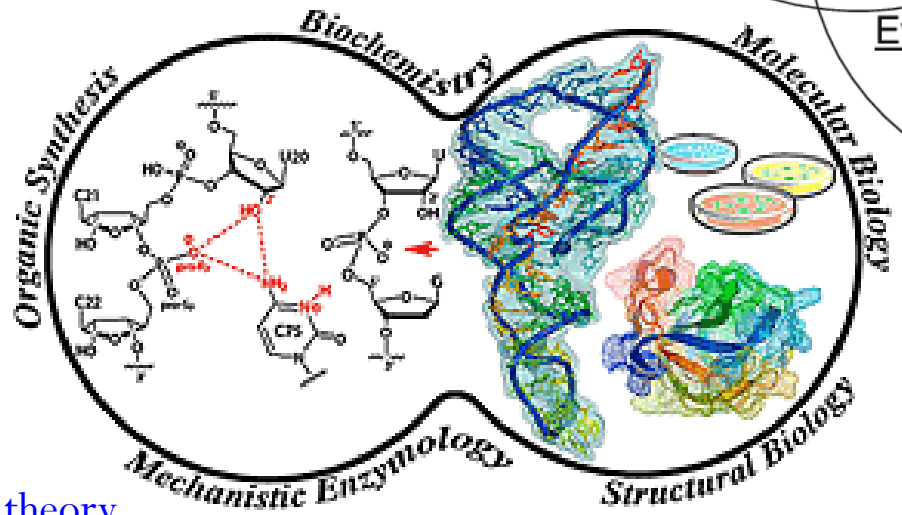
- **Homeostasis:** Tendency to be resilient w.r.t external disruption and to maintain functionalities.

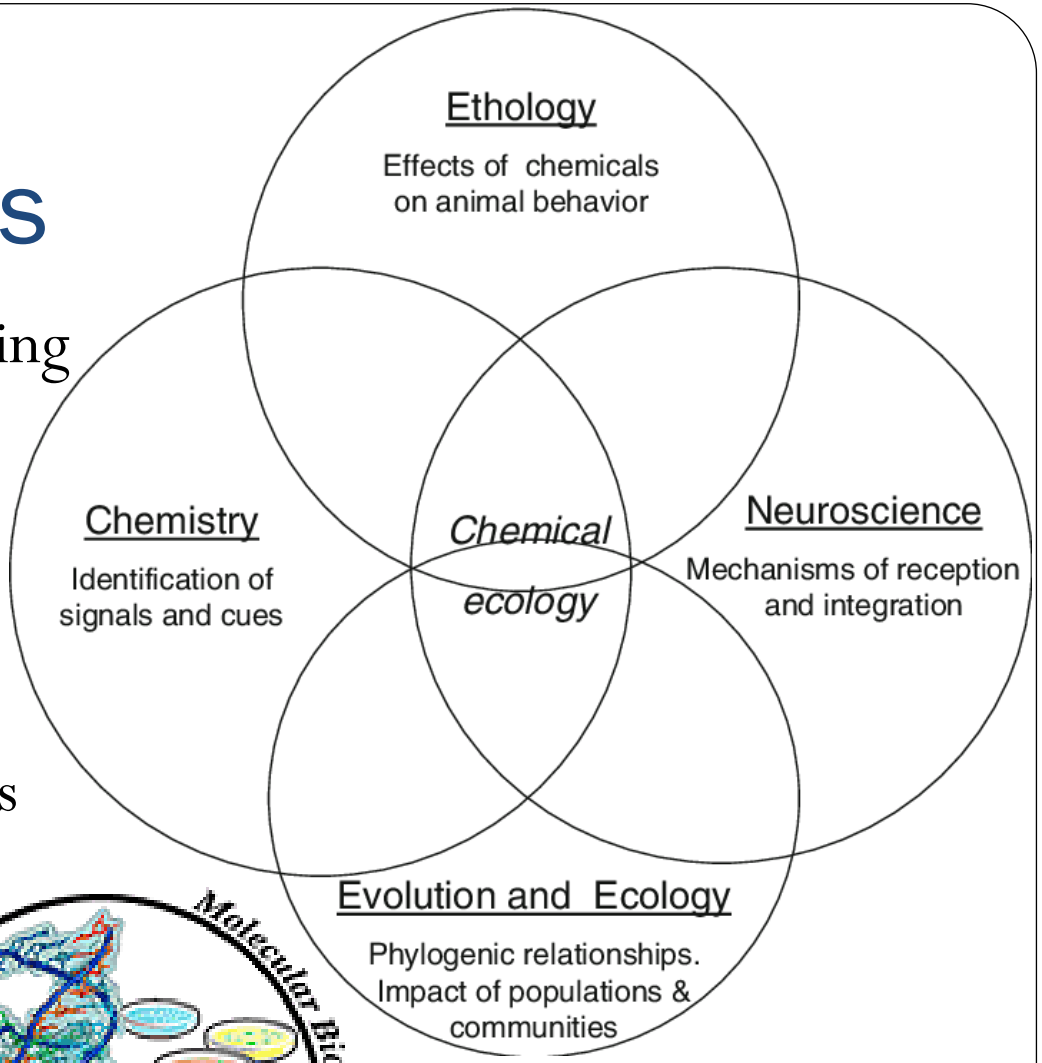


https://en.wikipedia.org/wiki/Systems_theory

Systems Theory Applications

- **Systems biology** is bioscience research focusing on complex interactions in biological systems.
- **Systems chemistry** studies networks of interacting molecules.
- **Systems ecology** is a field of ecology (subset Earth science) that studies of ecological systems i.e. ecosystems.





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

AI Application on Interdisciplinary Science and Engineering

Interdisciplinary

- An organizational unit involving two or more academic disciplines,
- Dedicated journals, conferences and university departments.
- Three levels of cross-disciplinary research:
  - **Multidisciplinarity:** Pluridisciplinary level draws knowledge from different disciplines but stays within their boundaries.
  - **Interdisciplinarity:** Cross-disciplinary level analyzes, synthesizes and harmonizes links between disciplines.
  - **Transdisciplinarity:** Discipline-forming level integrates and transcends traditional boundaries.

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

- Choi BC, Pak AW. Multidisciplinarity, interdisciplinarity and transdisciplinarity in health research, services, education and policy: 1. Definitions, objectives, and evidence of effectiveness. *Clin Invest Med.* 2006 Dec; 29(6):351-64. PMID: 17330451.

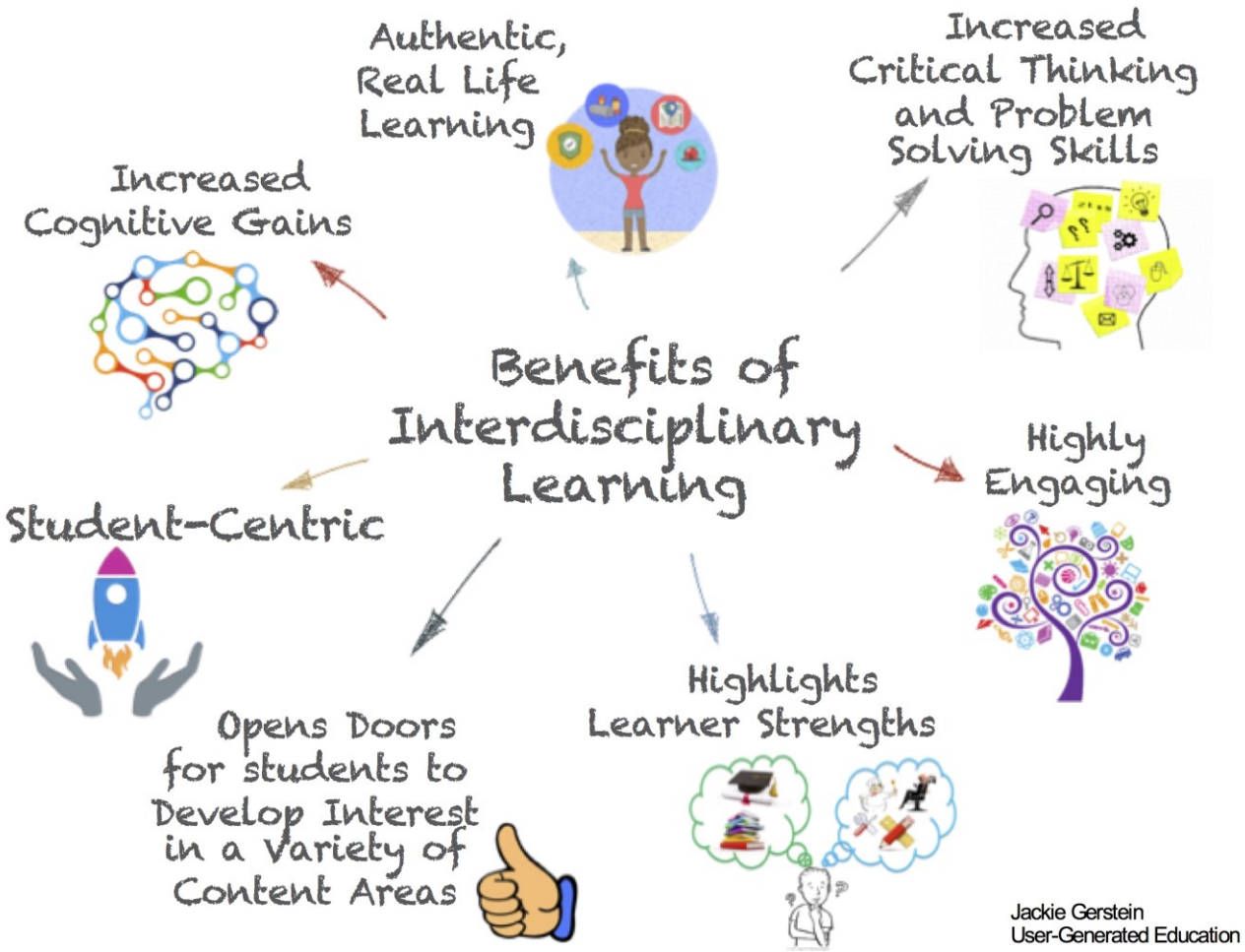
Interdisciplinary application of SysEng and AI

- aka. Interdisciplinary studies
- combination of two or more academic disciplines into one activity (e.g., a research project).
- Disciplines could be like
  - social science, biology, chemistry, mathematics
  - mechanical engineering,
  - electrical engineering,
  - computer science and engineering, etc.
- Inter-discipline examples Electromechanics, Mechatronics, Bioinformatics, Biomedical Engineering, Data Science/Analytics, Computational Social Systems etc.

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

Interdisciplinary Examples for AI applications

- Biomedical engineering (BME) is the application of engineering principles and design concepts to medicine and biology for healthcare purposes (e.g., diagnostic or therapeutic).
- Bioinformatics develops methods and software tools for understanding biological data, in particular when the data sets are large and complex.



https://en.wikipedia.org/wiki/Biomedical_engineering
https://en.wikipedia.org/wiki/Bioinformatics

Bio-Medical Engineering (BME) for applying AI

- Bioinformatics
- Biomechanics
- Biomaterials science or engineering
- Biomedical optics
- Tissue engineering
- Genetic engineering
- Neural engineering
- Pharmaceutical engineering
- Medical devices (Medical imaging, Implants, Bionics, and Biomedical sensors)
- Clinical engineering
- Rehabilitation engineering

https://en.wikipedia.org/wiki/Biomedical_engineering
https://en.wikipedia.org/wiki/Bioinformatics

Bioinformatics based on Data Science (DS)

- **Informatics:** set of digital codes and a language
- **Bioinformatics:** Study of biological (or life) information (digital code for studying properties of bio-systems)

Computer scientists, Mathematicians, Data Scientist etc.

Develop tools, software, algorithms

Store and analyze the data.

Biologists

collect molecular data: DNA & Protein sequences, gene expression, etc.

Bioinformaticians

Study biological questions by analyzing molecular data

Bioinformatics based on Data Science (DS)

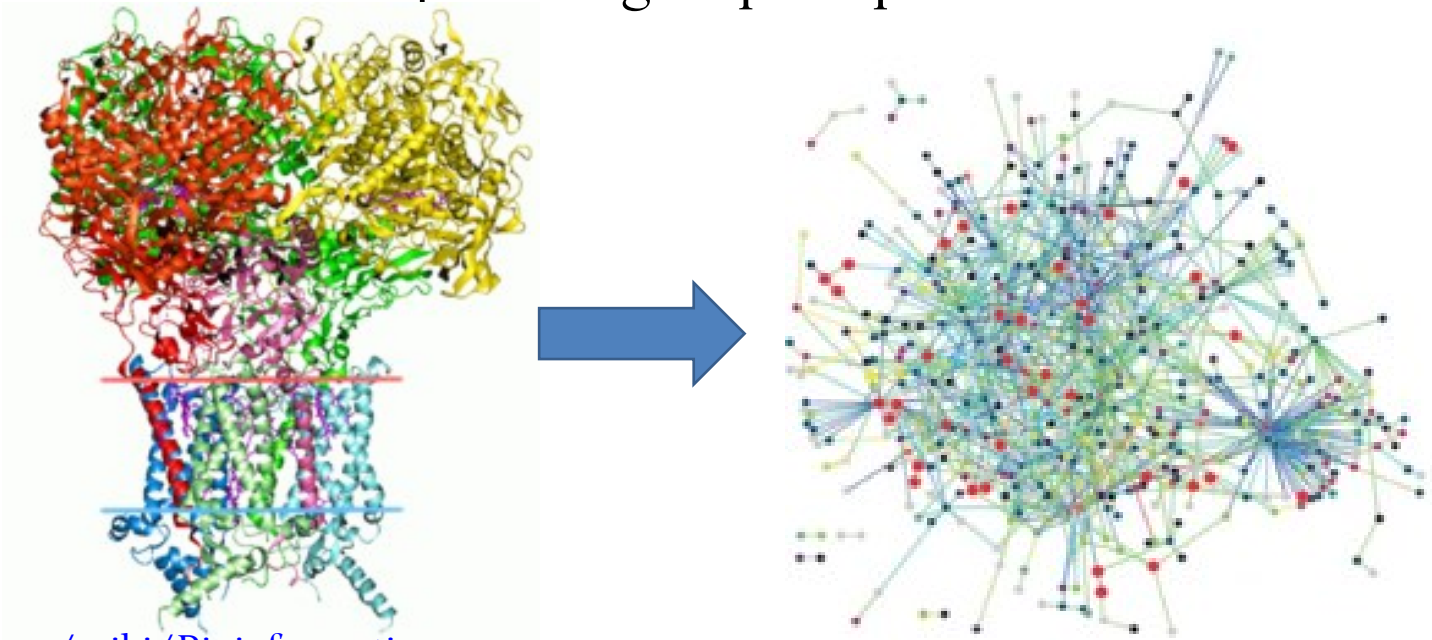
- Computers became essential in molecular biology when protein sequences, amino acid sequences, protein domains, protein structures etc.
- Sequences of genetic material are frequently used in bioinformatics and are easier to manage using computers than manually.
- DNA sequencing is still a non-trivial problem as the raw data may be noisy or afflicted by weak signals. Algorithms have been developed for base calling for the various experimental approaches to DNA sequencing.

```
5'ATGACGTGGGGA3'
3'TACTGCACCCCT5'
```

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

Molecular interaction networks based on DS

- BME: Tens of thousands of three-dimensional protein structures are determined by X-ray Crystallography and protein Nuclear Magnetic Resonance (NMR) spectroscopy.
- Bioinformatics: Protein–protein interaction identifies, predicts, and catalog physical interactions between pairs or groups of proteins.



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

Network analysis

- Study of relationships within biological networks
  - metabolic or
  - protein–protein interaction networks.
- Biological networks can be constructed from
  - a single type of molecule or entity (such as genes),
  - many different data types, such as proteins, small molecules, gene expression data.
- Abbreviation recognition – identify the long-form and abbreviation of biological terms
- Named entity recognition – recognizing biological terms such as gene names
- Protein–protein interaction – identify which proteins interact with which proteins from text

Systems biology

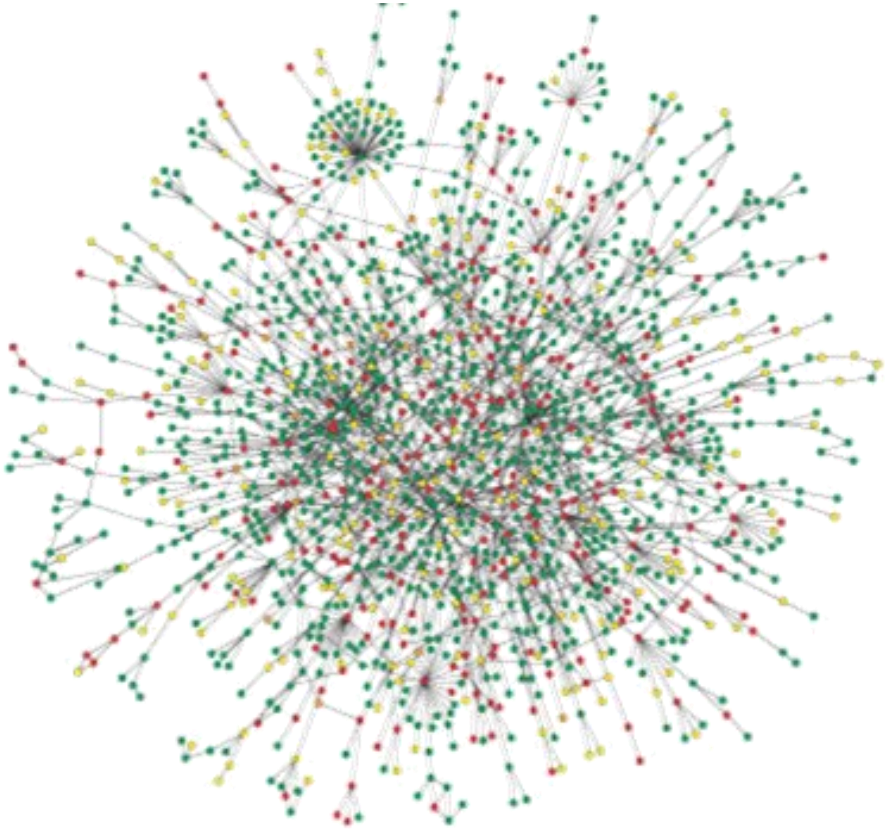
- It involves the use of computer simulations of cellular subsystems such as the networks of metabolites and enzymes that comprise metabolism.
- Signal Transduction Pathways and Gene Regulatory Networks to both analyze and visualize the complex connections of these cellular processes.
- **Artificial** life or virtual evolution attempts to understand evolutionary processes via the **computer simulation of simple (artificial) life forms**.

Bio Interactions

- Protein-Protein Interaction
- DNA-Protein interactions
- GeneNet (Gene networks)
- Biomolecular Interaction
- Molecular interactions
- Protein and Biochemical Interactions

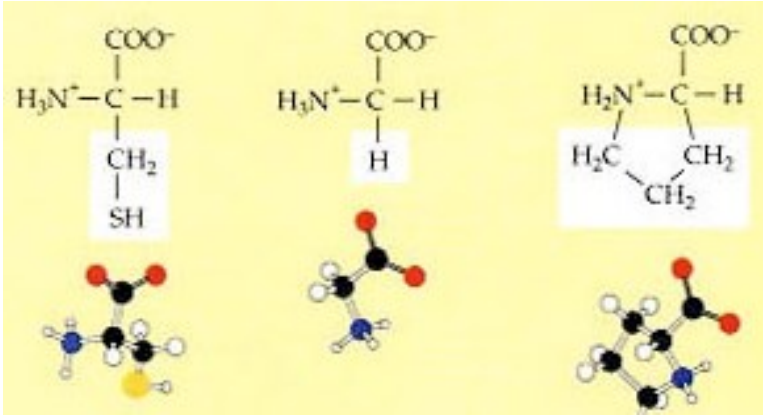
Nodes: proteins

Links: physical interactions (binding)



Bioinformatics Visualization

- Amino Acid to Graph



- Human Hemoglobin

>gi|14456711|ref|NM_000558.3| **Homo sapiens hemoglobin, alpha** 1 (HBA1), mRNA

ACTCTTCTGGTCCCCACAGACTCAGAGAGAACCCACCATGGTGCTGTCT
CCTGCCGACAAGACCAACGTCAAGGCCGCCTGGGGTAAGGTCGGCGCGC
ACGCTGGCGAGTATGGTGCGGAGGCCCTGGAGAGGATGTTCCTGTCCTT
CCCCACCACCAAGACCTACTTCCCGCACTTCGACCTGAGCCACGGCTCT
GCCCAGGTTAAGGGCCACGGCAAGAAGGTGGCCGACGCGCTGACCAACG
CCGTGGCGCACGTGGACGACATGCCCAACGCGCTGTCCGCCCTGAGCGA
CCTGCACGCGCACAAGCTTCGGGTGGACCCGGTCAACTTCAAGCTCCTA
AGCCACTGCCTGCTGGTGACCCTGGCCGCCCACCTCCCCGCCGAGTTCA
CCCCTGCGGTGCACGCCTCCCTGGACAAGTTCCTGGCTTCTGTGAGCAC
CGTGCTGACCTCCAAATACCGTTAAGCTGGAGCCTCGGTGGCCATGCTT
CTTGCCCCTTGGGCCTCCCCCCAGCCCCTCCTCCCCTTCCTGCACCCGT
ACCCCCGTGGTCTTTGAATAAAGTCTGAGTGGGCGGC

Bio-Informatics Databases

- KEGG (Kyoto Encyclopedia of Genes and Genomes)
  - http://www.genome.ad.jp/kegg/
  - Institute for Chemical Research, Kyoto University
- PathDB
  - http://www.ncgr.org/pathdb/index.html
  - National Center for Genomic Resources
- SPAD: Signaling PAthway Database
  - Graduate School of Genetic Resources Technology. Kyushu University.
- Cytokine Signaling Pathway DB.
  - Dept. of Biochemistry. Kumamoto Univ.
- EcoCyc and MetaCyc
  - Stanford Research Institute
- BIND (Biomolecular Interaction Network Database)
  - UBC, Univ. of Toronto

Biological Synergy Systems

Synergy

- Synergy is two or more things functioning together to produce a result not independently obtainable.
- This may give both positive and negative effects.
- In systems engineering, Synergy describes how system behavior emerges from the interaction between elements or components or entities.
- Synergy is closely related to Emergence.
- An interaction or cooperation giving rise to a whole that is greater than the simple sum of its parts.
- Term comes from the Attic Greek, meaning "working together".

https://www.sebokwiki.org/wiki/Synergy_(glossary)
https://en.wikipedia.org/wiki/Synergy

Synergy

- In an organization synergy is the ability of a group to outperform even its best individual member. (Buchanan and Huczynski,1997).
- A construct or collection of different elements working together to produce results not obtainable by any of the elements alone. The elements, or parts, can include people, hardware, software, facilities, policies, documents: all things required to produce system-level results. (Blanchard 2004).

https://www.sebokwiki.org/wiki/Synergy_(glossary)
https://en.wikipedia.org/wiki/Synergy

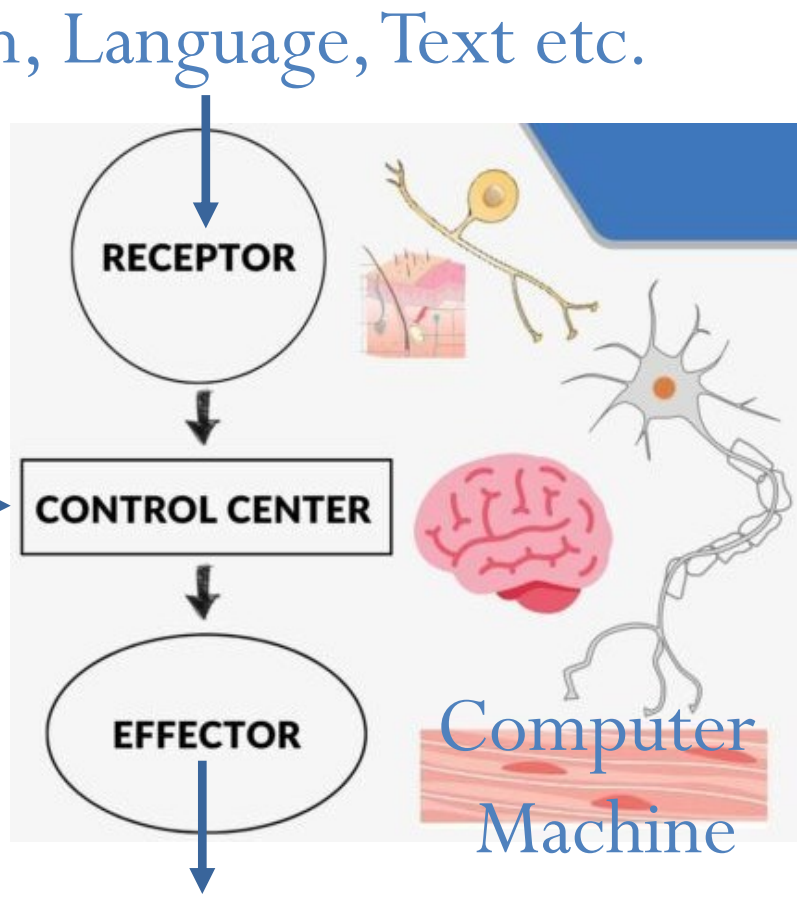
Biological and Artificial Neural Network synergy

- Biology and AI synergy: Artificial Neural Network
- In medicine synergy is used to describe combinations of drugs which interact in ways that enhance or magnify one or more effects, or side-effects, of those drugs.
- Pest synergy occur in a biological host organism population.
  - parasite A cause 10% fatalities, and parasite B also cause 10% loss. When both parasites are present, the parasites in combination have a synergistic effect.
- Drug synergy: involved in the development of synergistic effects of drugs
  - two different antibiotics can improve the effect

https://www.sebokwiki.org/wiki/Synergy_(glossary)
https://en.wikipedia.org/wiki/Synergy

Control Behaviour using AI

- **Cybernetics**, the science of control, defines two basic control mechanisms:
  - Negative feedback,
  - Positive feedback,
- **Control behavior** is a trade between:
  - Specialization, the focus of system behavior to exploit particular features of its environment, and
  - Flexibility, the ability of a system to adapt quickly to environmental change.

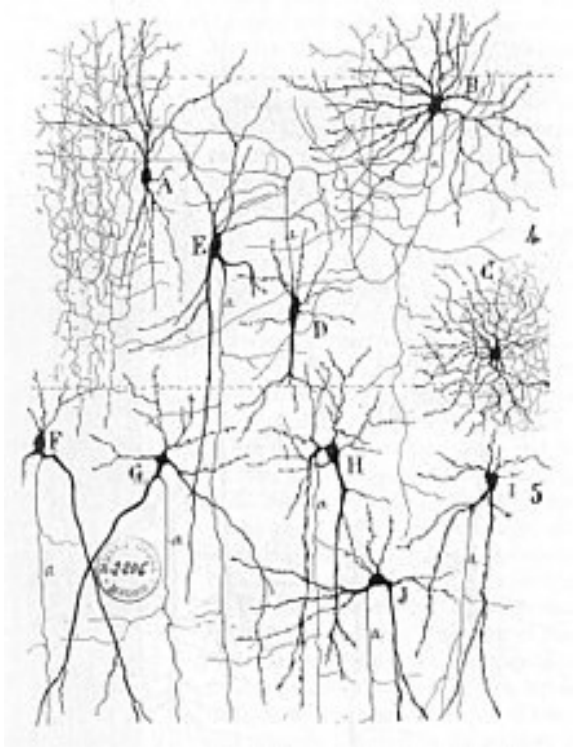


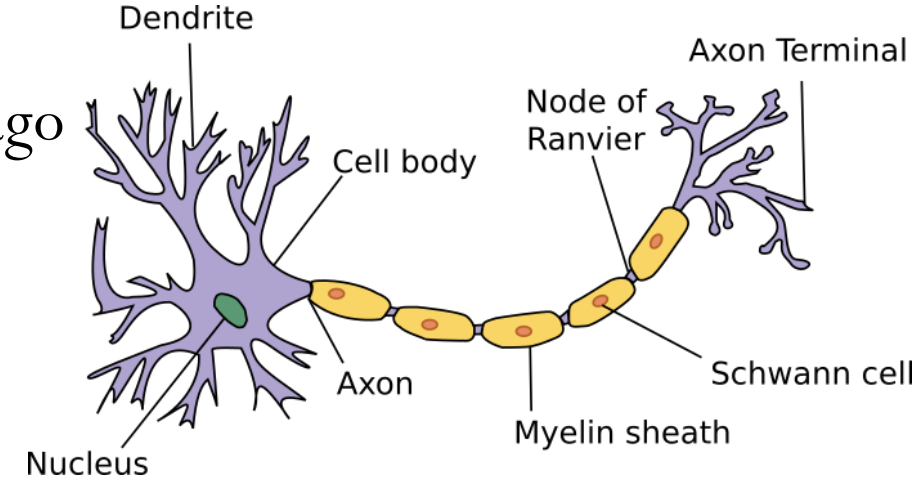
- Cloutier, R. J. "The Guide to the Systems Engineering Body of Knowledge (SEBoK); v. 2.2." *INCOSE and The Trustees of the Stevens Institute of Technology: Hoboken, NJ, USA* (2016).

History of Artificial Neural Network (ANN)

Neural Network (Anatomy)

- **1870s:** Reticular Theory "nervous system and brain is a single continuous network, all nerve cells in the nervous system constituted a continuous, interconnected network"
- **1888:** Neurons "fundamental units of the brain and nervous system, each nerve cell is an independent entity and nerve synapses transfer nerve impulses from one cell to another"
- **1906:** Nobel prize to both Camillo Golgi and Santiago Ramón y Cajal "in recognition of their work on the structure of the nervous system."

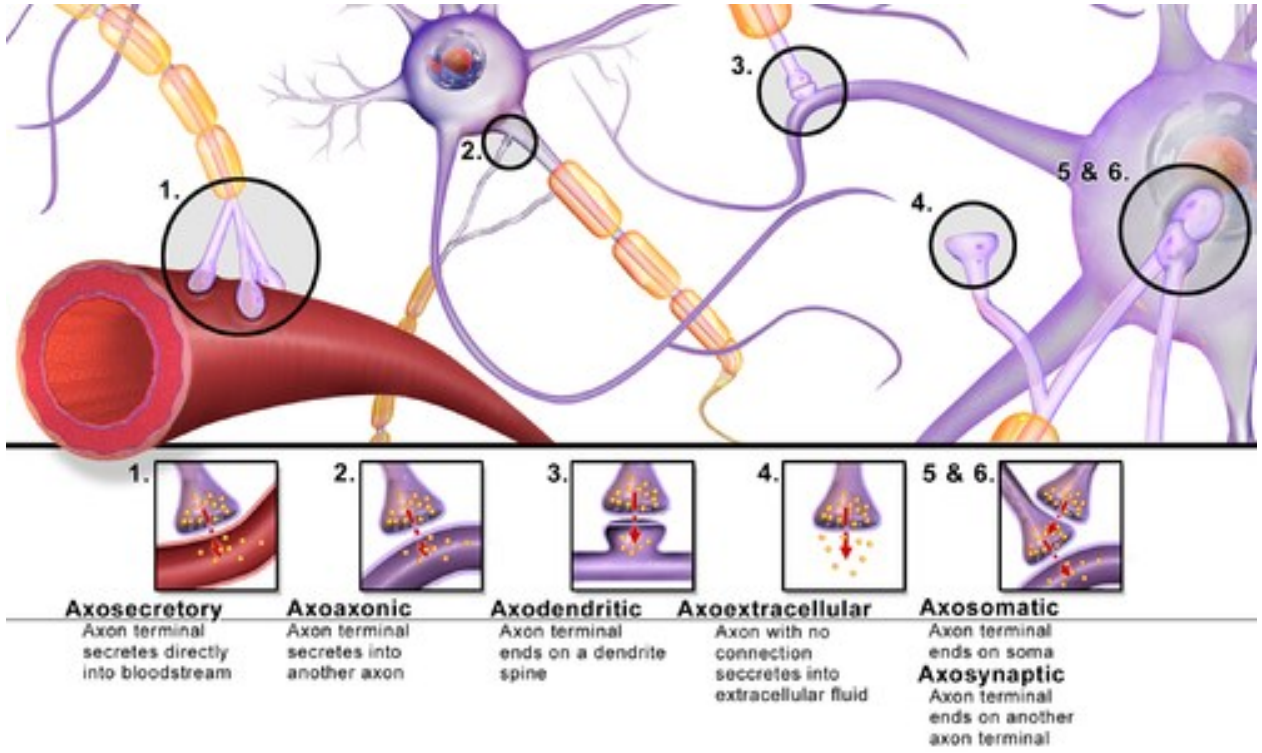




https://www.nobelprize.org/prizes/medicine/1906/summary/

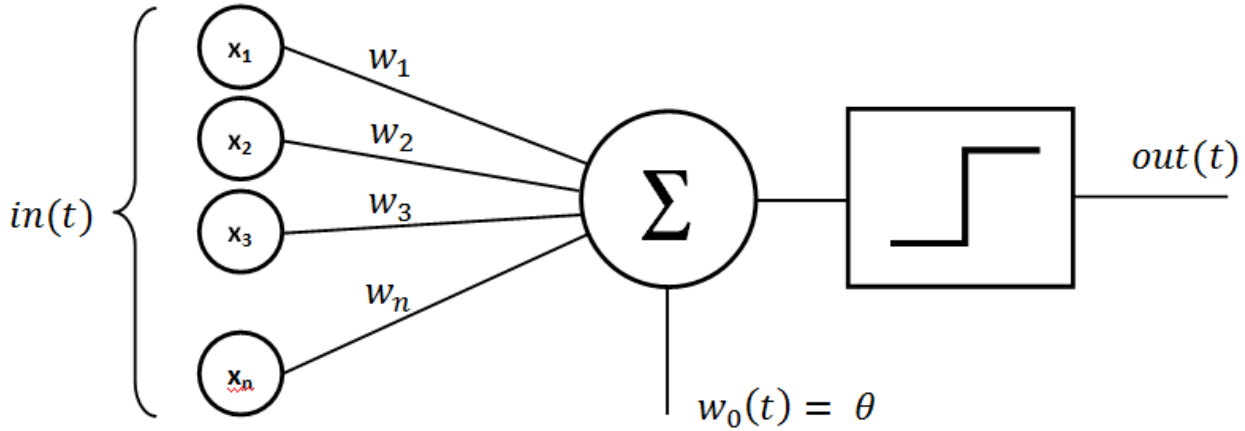
Neural Network (Anatomy)

- Synapse is a structure that permits a neuron (or nerve cell) to pass an electrical or chemical signal to another neuron or to the target effector cell.
- Synapses are essential to the transmission of nervous impulses from one neuron to another.
- Different types of synapses



Perceptron

- **1943 Perceptron** was invented by **McCulloch and Pitts**
- **1958 Mark I Perceptron** hardware developed and constructed by **Frank Rosenblatt**



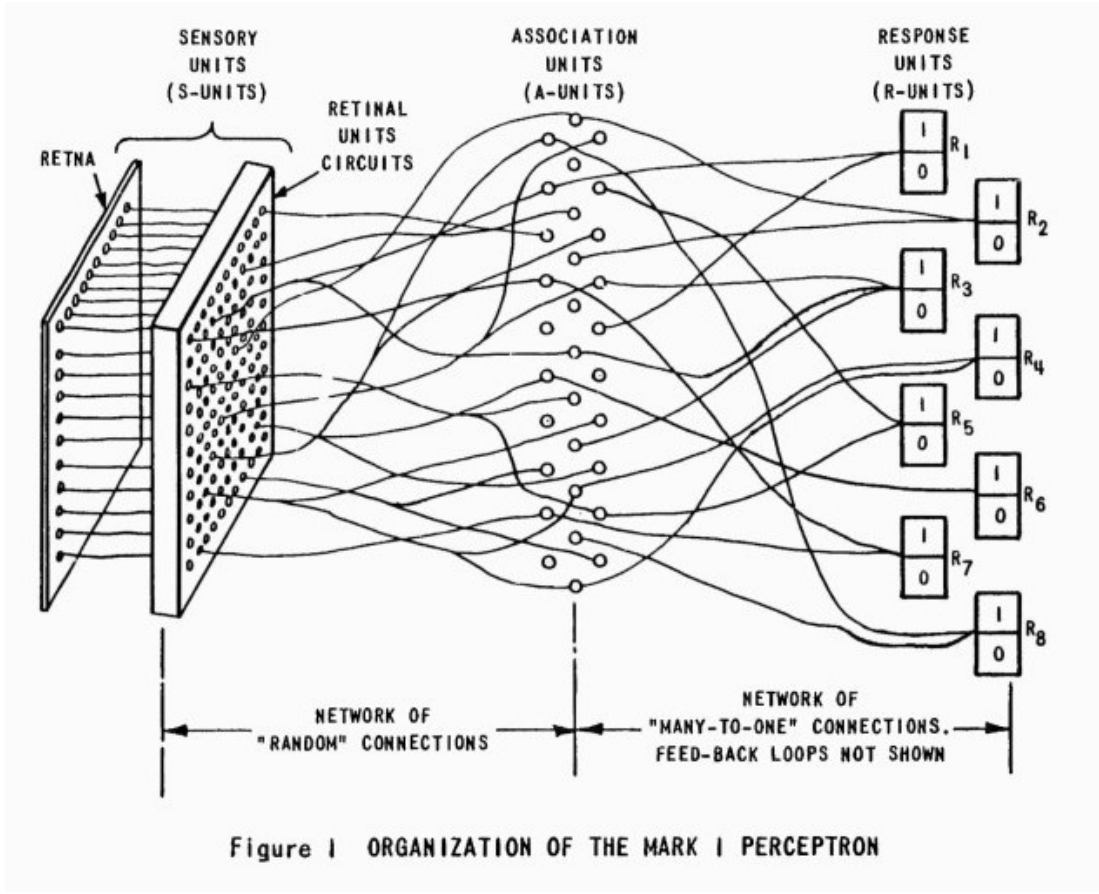
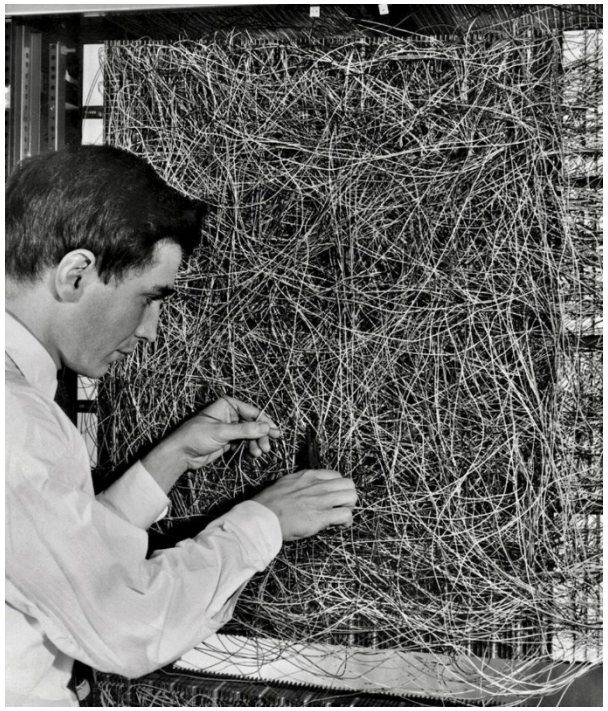
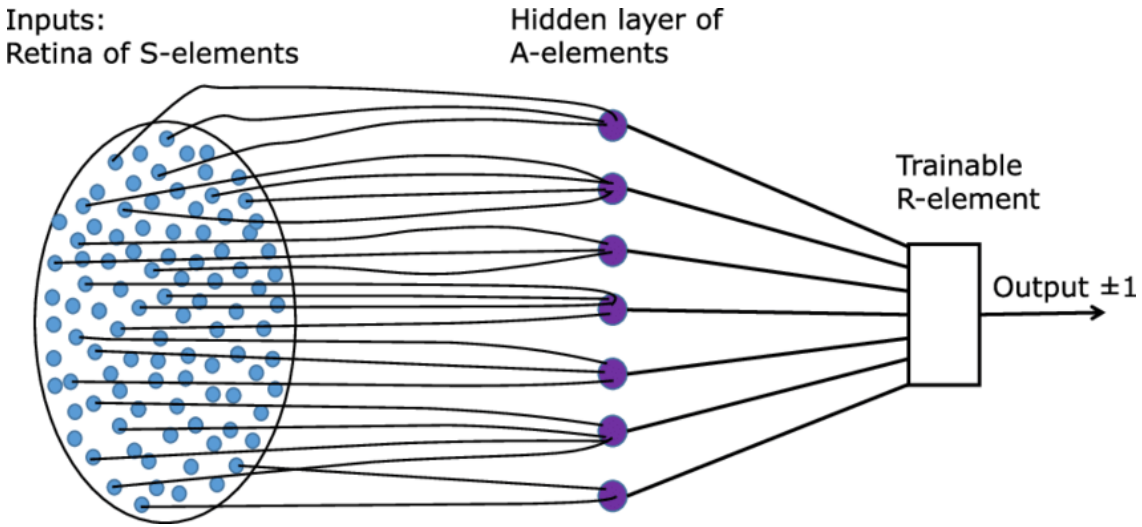


Figure 1 ORGANIZATION OF THE MARK I PERCEPTRON

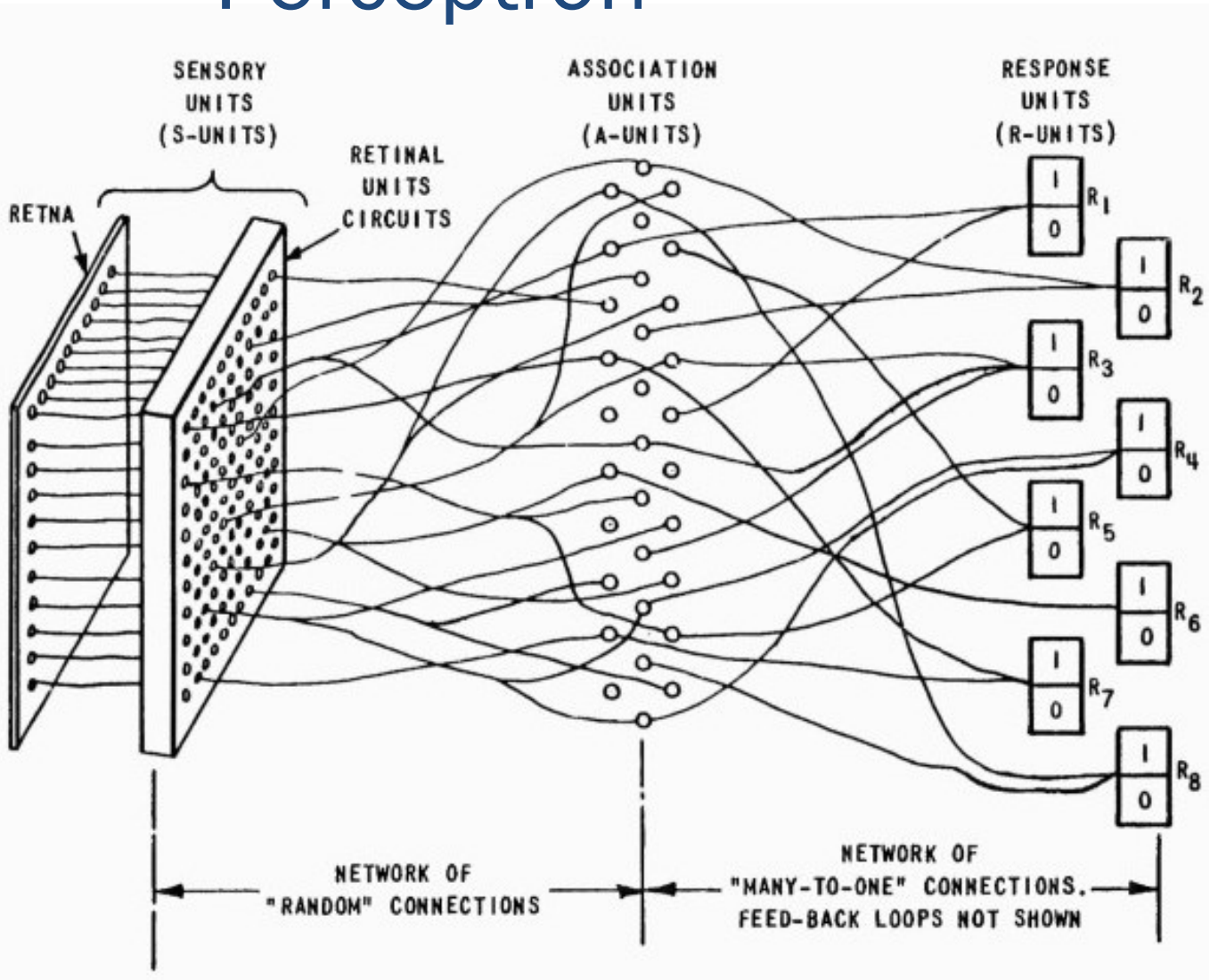
Perceptron

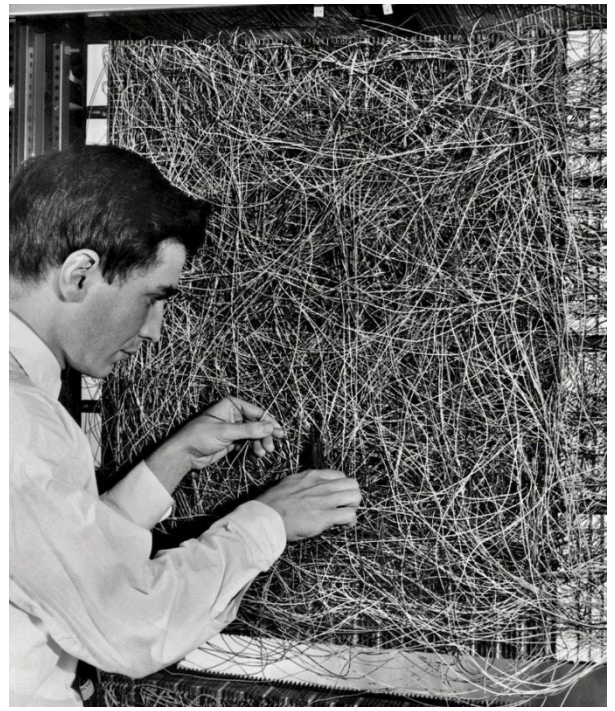
- **1958** Frank Rosenblatt development and hardware construction of the "**Mark I Perceptron**"
  - the first computer that could learn new skills by trial and error, using a type of neural network that simulates human thought processes.
  - at the Cornell Aeronautical Laboratory funded by the United States Office of Naval Research.

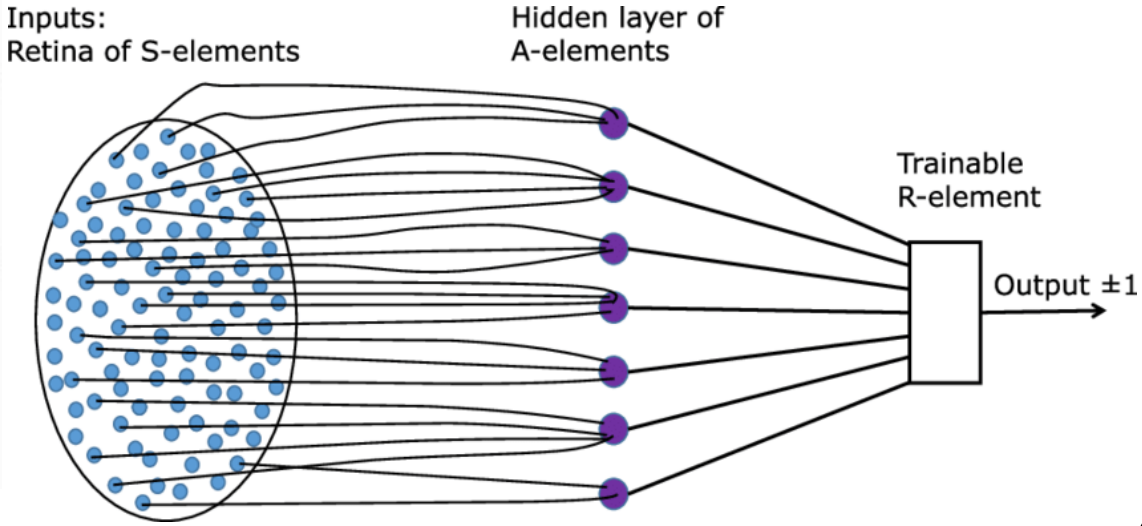




Perceptron

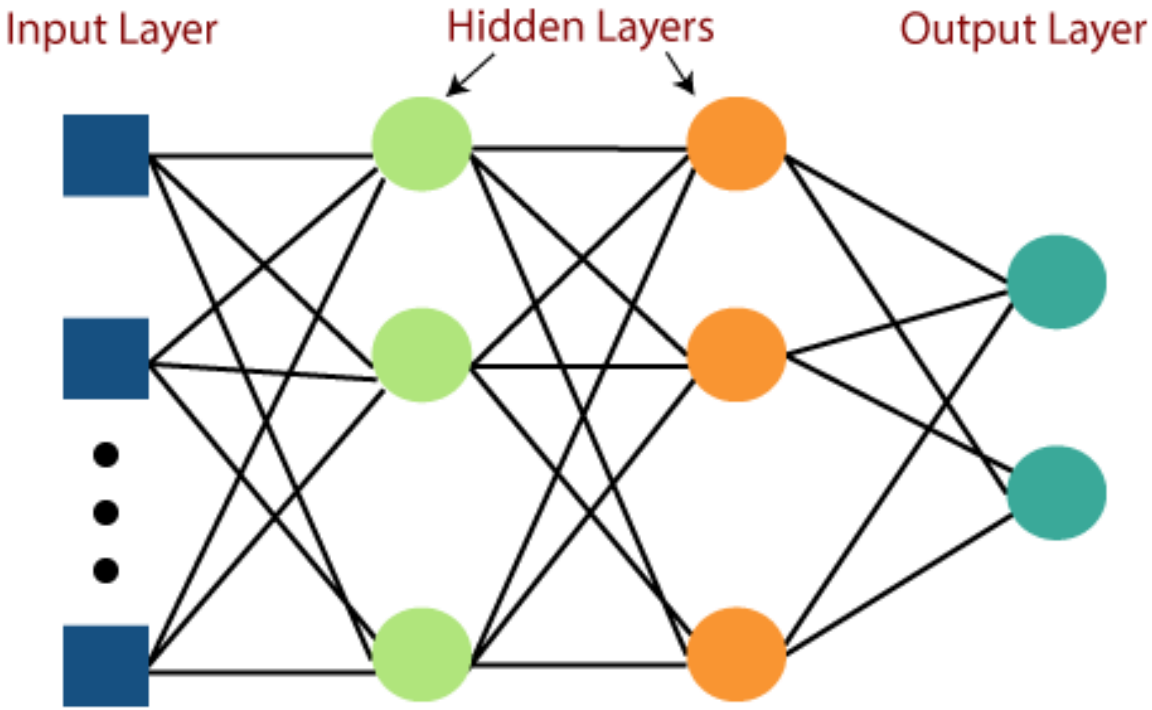

SENSORY UNITS (S-UNITS)
ASSOCIATION UNITS (A-UNITS)
RESPONSE UNITS (R-UNITS)
RETINAL UNITS CIRCUITS
RETNA
R1
R2
R3
R4
R5
R6
R7
R8
NETWORK OF "RANDOM" CONNECTIONS
NETWORK OF "MANY-TO-ONE" CONNECTIONS. FEED-BACK LOOPS NOT SHOWN

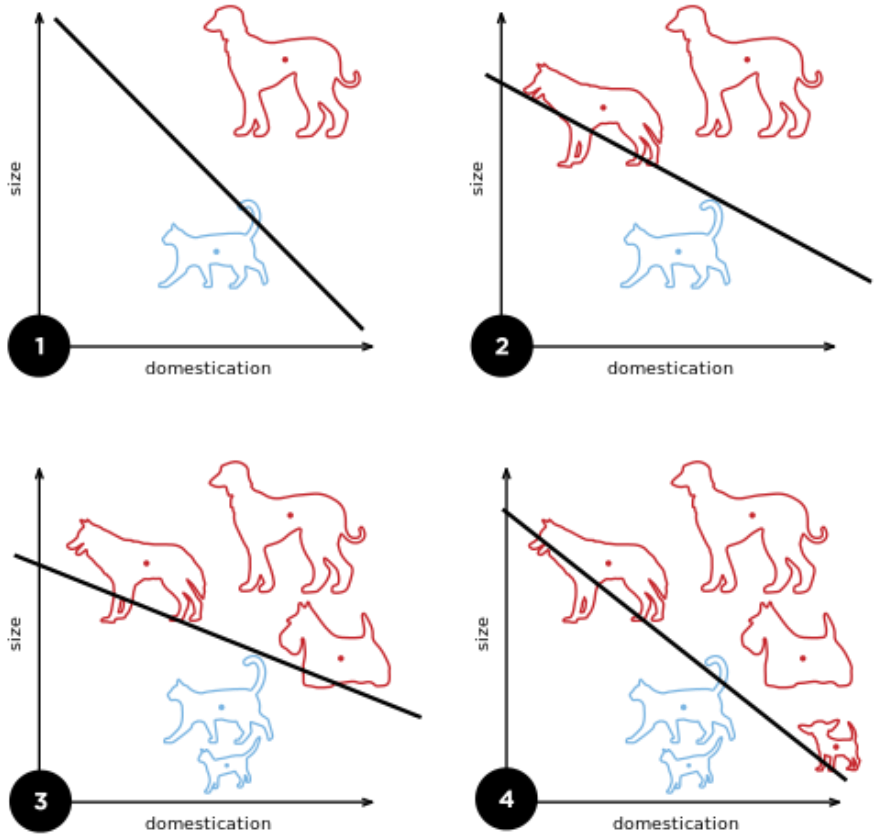



Inputs:
Retina of S-elements
Hidden layer of
A-elements
Trainable
R-element
Output ±1

Multi-layer Perceptron

- **1960s** Limitation of Perceptron: Failed to classify basic problems
- Ivakhnenko et. al. introduced Multilayer Perceptrons

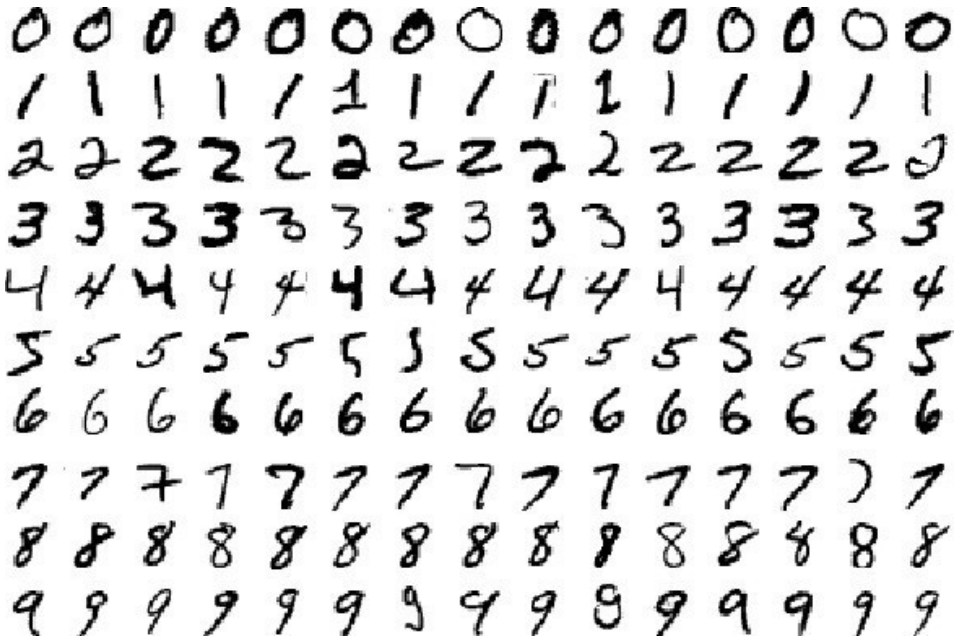




Artificial Neural Network and Backpropagation

- 1982: Werbos introduced Backpropagation first used it in the context of Artificial Neural Networks
- 1986: Rumelhart et. al. popularized the Backpropagation
- Use of Cauchy discovered theories of Gradient Descent on Convex function for Optimization
- 1989: Universal Approximation Theorem "A multilayered network of neurons with a single hidden layer can be used to approximate any continuous function to any desired precision"

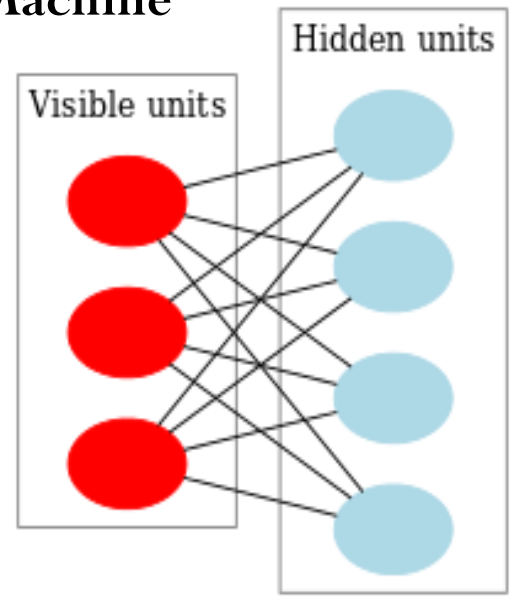
Deep Learning



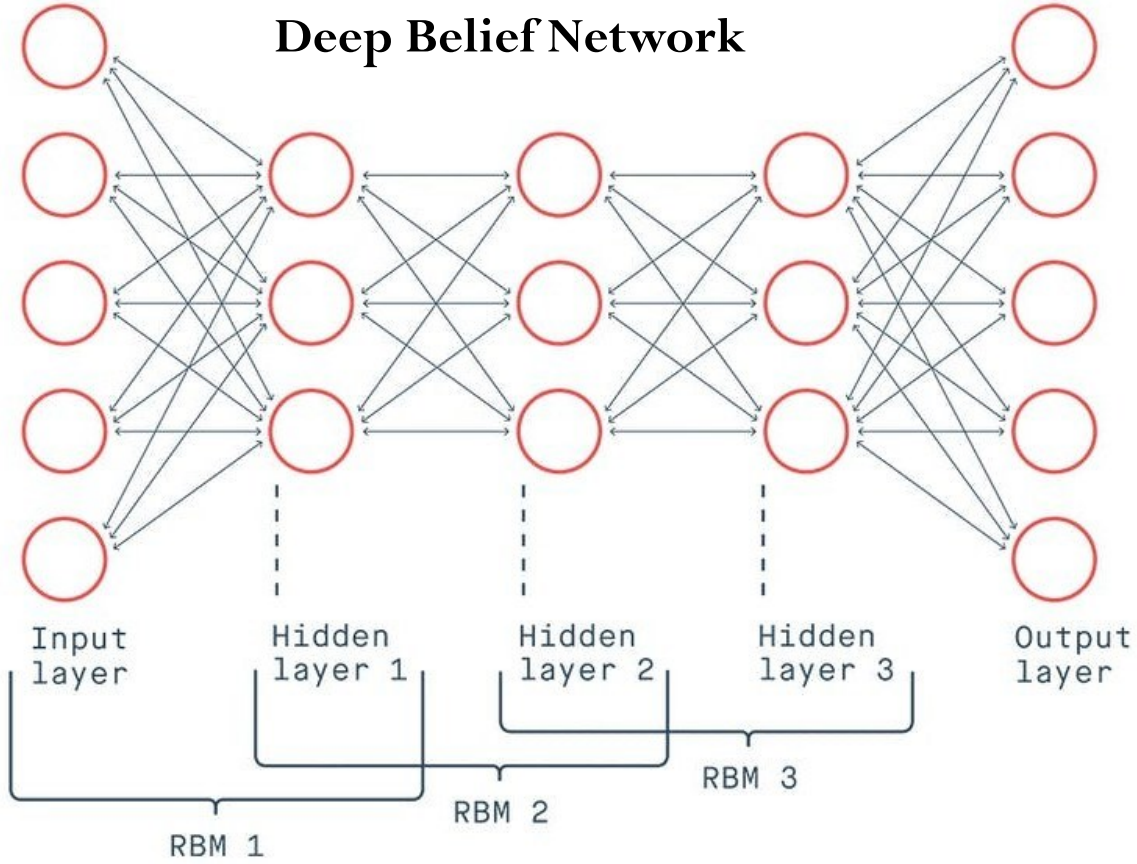
- Unsupervised Pre-Training
  - 1991-1993: J. Schmidhuber "Very Deep Learner"
  - 2006: Hinton and Salakhutdinov
- 2009: Handwriting Recognition
  - MNIST dataset
- 2010: Speech Recognition
  - Dahl et. al. achieved error reduction of 16.0% to 23.2% over previous works
- Traffic Sign Recognition Competition
  - D. C. Ciresan et. al. got only 0.56% error rate

Restricted Boltzmann Machine and Deep Belief Network

Restricted Boltzmann Machine

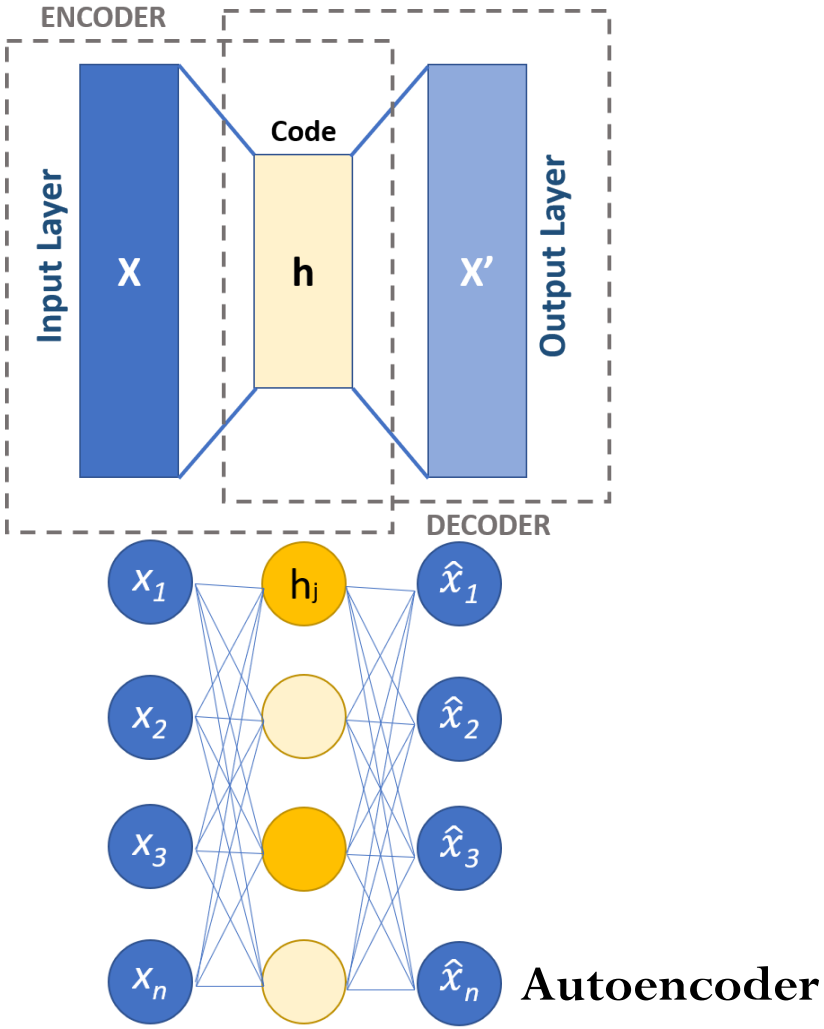


Deep Belief Network

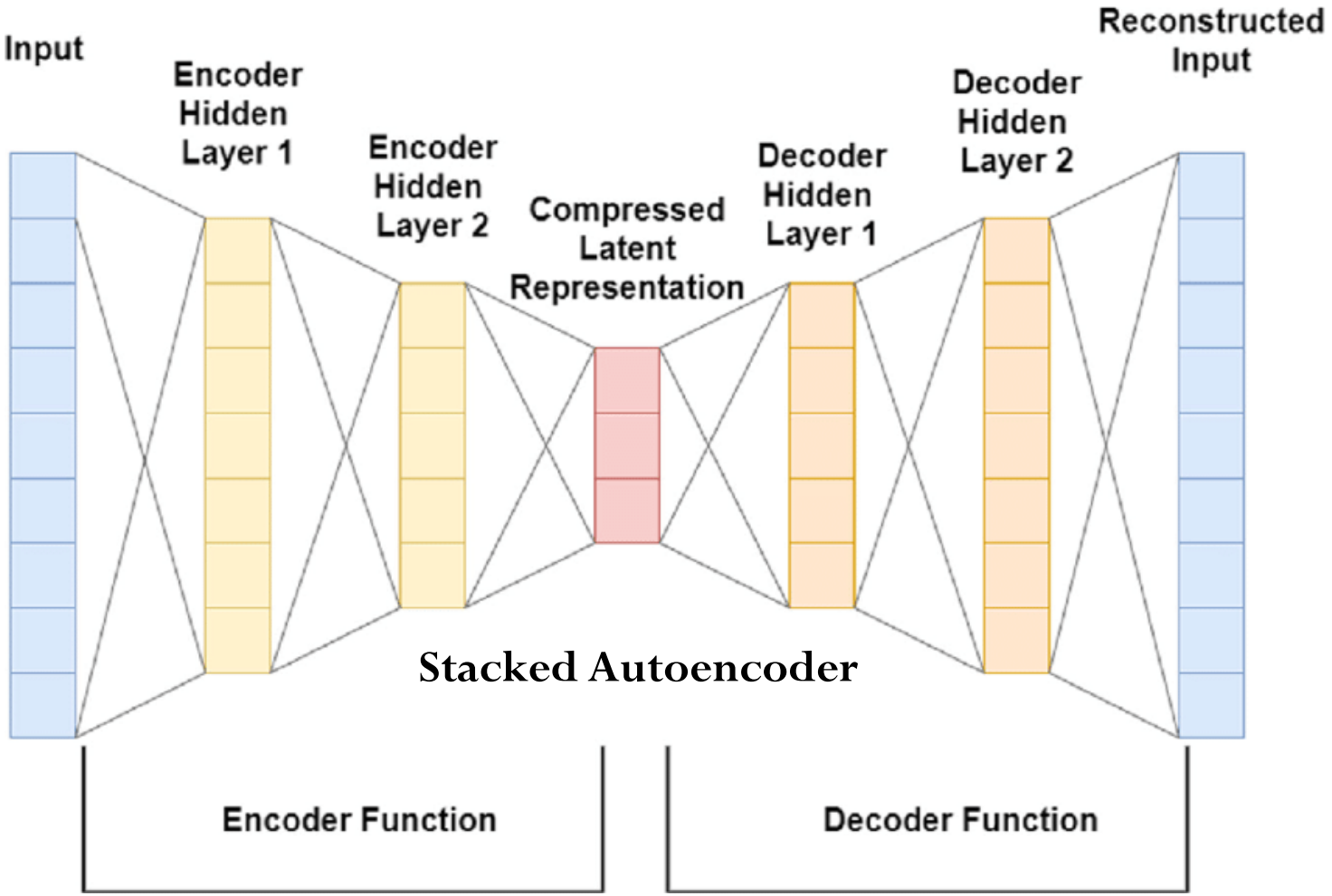


- Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. "Restricted Boltzmann machines for collaborative filtering." *Proceedings of the 24th International Conference on Machine learning*. 2007.
- Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh. "A fast learning algorithm for deep belief nets." *Neural computation* 18.7 (2006): 1527-1554.

Autoencoder and Stacked Autoencoder



Autoencoder

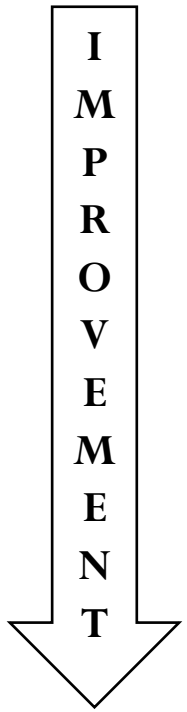


- Pascal Vincent, et al. "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion." *Journal of Machine Learning Research* 11.12 (2010).

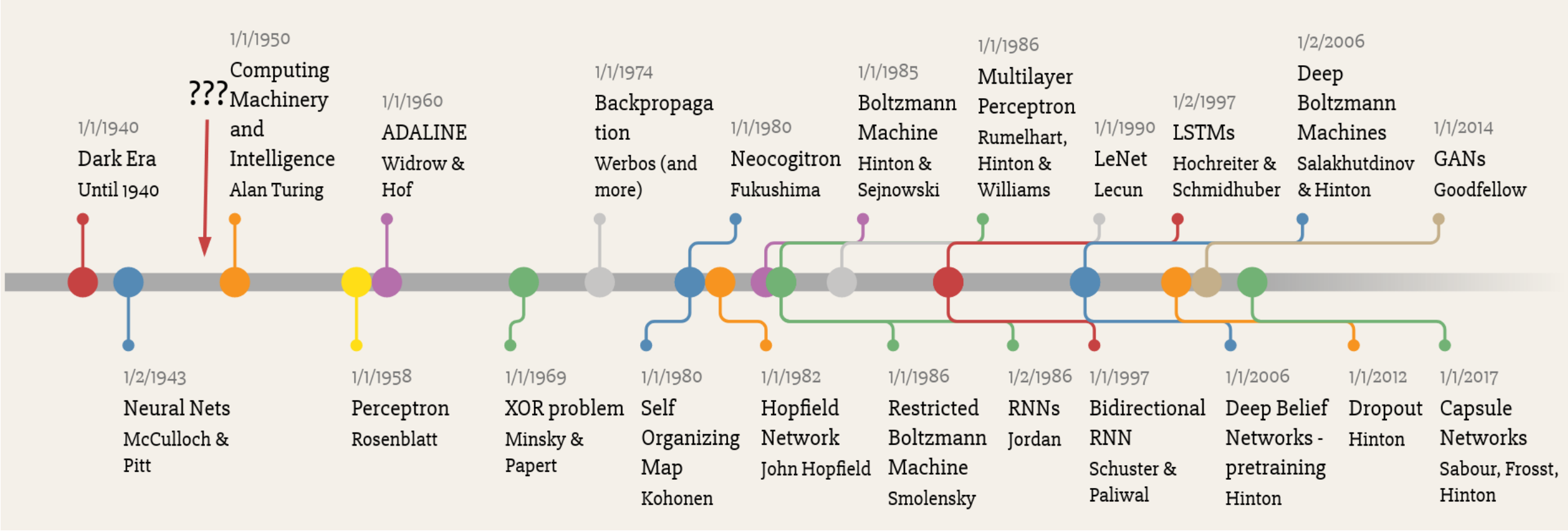
Visual Recognition Challenges

- ImageNet challenge successful 2012-2016

Network	Error	Layers
AlexNet	16.0%	8
ZFNet	11.2%	8
VGGNet	7.3%	19
GoogLeNet	6.7%	22
MS ResNet	3.6%	152!!



Artificial Neural Network (ANN) Timeline



- Deep Learning for the Masses (… and The Semantic Layer), Deep Learning timeline by Favio Vazquez
https://www.kdnuggets.com/2018/11/deep-learning-masses-semantic-layer.html

Subgraphs, Network Motifs, and Graphlets

Apriori algorithm: Association Rule Mining

- Let I = $\{i_1, i_2, \ldots . i_m\}$ be a set of literals, called items.
- *Support* of a rule X →Y is the percentage of transactions that contain both X and Y.
- *Confidence* of a rule is percentage the if-then statements (X →Y) are found true
- Find all rules that satisfy a user-specified *minimum support* and *minimum confidence*

TID	Transaction Items
1	Bread, Jelly, PeanutButter
2	Bread, PeanutButter
3	Bread, Milk, PeanutButter
4	Beer, Bread
5	Beer, Milk

{Bread} → {PeanutButter} (Sup = 60%, Conf = 75%)

{PeanutButter} → {Bread} (Sup = 60%, Conf = 100%)

{Beer} → {Bread} (Sup = 20%, Conf = 50%)

{PeanutButter} → {Jelly} (Sup = 20%, Conf = 33.33%)

{Jelly} → {PeanutButter} (Sup = 20%, Conf = 100%)

{Jelly} → {Milk} (Sup = 0%, Conf = 0%)

- Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. "Mining association rules between sets of items in large databases." *SIG-MOD*. 1993.
- Ramakrishnan Srikant, and Rakesh Agrawal. "Mining Generalized Association Rules." *VLDB* 1995.

Apriori algorithm: Association Rule Mining

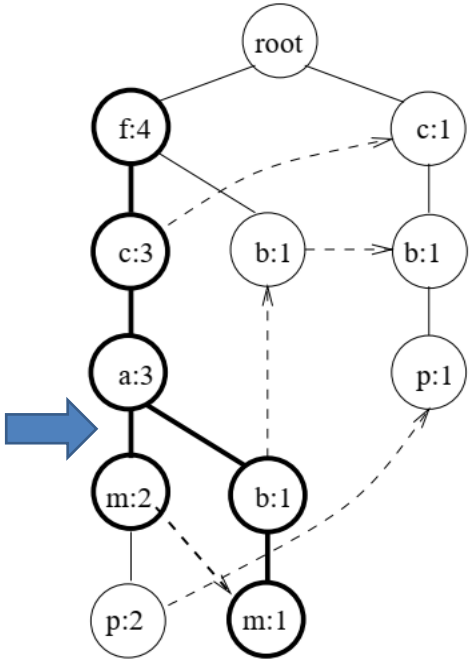
- Let I = $\{i_1, i_2, \ldots . i_m\}$ be a set of literals, called items.
- *Support* of a rule X $\rightarrow$Y is the percentage of transactions that contain both X and Y.
- *Confidence* of a rule is percentage the if-then statements (X $\rightarrow$Y) are found true
- Find all rules that satisfy a user-specified *minimum support* and *minimum confidence*
  - 75% of transactions that purchase *PeanutButter* (antecedent) also purchase *Bread* (consequent). The number 75% is the confidence factor of the rule
    - [PeanutButter] $\rightarrow$ [Bread] (Sup = 60%, Conf = 75%)
  - 98% of customers who purchase *Tires* and *Auto accessories* also buy some *Automotive services*; here 98% is called the confidence of the rule.
    - [*Auto Accessories*], [*Tires*] $\rightarrow$ [*Automotive Services*] 98%

- Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. "Mining association rules between sets of items in large databases." *SIG-MOD*. 1993.
- Ramakrishnan Srikant, and Rakesh Agrawal. "Mining Generalized Association Rules." *VLDB* 1995.

FP-Growth for recommendation

- "FP" stands for Frequent Pattern in a Dataset of transactions
  1. calculate item frequencies and identify frequent items,
  2. a suffix tree (FP-tree) structure to encode transactions, and
  3. frequent itemsets can be extracted from the FP-tree.
- Input: Transaction database
- Intermediate Output: FP-Tree
- Output: {f, c, a → a, m p}, {f, c, a → b, m}

TID	Items Bought	(Ordered) Frequent Items
100	f, a, c, d, g, i, m, p	f, c, a, m, p
200	a, b, c, f, l, m, o	f, c, a, b, m
300	b, f, h, j, o	f, b
400	b, c, k, s, p	c, b, p
500	a, f, c, e, l, p, m, n	f, c, a, m, p



- Han Jiawei, Jian Pei, and Yiwen Yin. "Mining frequent patterns without candidate generation." *ACM SIGMOD Record* 29.2 (2000): 1-12.

An Apriori-based Algorithm for Mining Frequent Substructures from Graph Data

- Association rules among the frequently appearing substructures in a given graph data set
- The frequent patterns appearing in the matrices are mined through the extended algorithm of the basket analysis.

$$V(G_s) \subset V(G), E(G_s) \subset E(G),$$

$$\forall u, v \in V(G_s), (u, v) \in E(G_s) \Leftrightarrow (u, v) \in E(G).$$

$$When\ G_s\ is\ an\ induced\ subgraph\ of\ G,\ it\ is\ denoted\ as\ G_s \subset G.$$

$$sup(G_s) = \frac{number\ of\ graph\ transactions\ G\ where\ G_s \subset G \in GD}{total\ number\ of\ graph\ transactions\ G \in GD}$$

$$conf(G_b \Rightarrow G_h) = \frac{number\ of\ graphs\ G\ where\ G_b \cup G_h \subset G \in GD}{number\ of\ graphs\ G\ where\ G_b \subset G \in GD}$$

- Inokuchi, Akihiro, Takashi Washio, and Hiroshi Motoda. "An apriori-based algorithm for mining frequent substructures from graph data." *4th European Conference Principles of Data Mining and Knowledge Discovery, PKDD* 2000 Lyon, France, September 13–16, 2000 Proceedings 4. Springer Berlin Heidelberg, 2000.

Frequent Subgraph Discovery

- The problem of finding frequent patterns becomes that of discovering subgraphs that occur frequently over the entire set of graphs.
- A computationally efficient algorithm for finding all frequent subgraphs in large graph databases.
- Each vertex of the graph will correspond to an entity and
- Each edge will correspond to a relation between two entities.
- The resulting frequent subgraphs will be encapsulating relations (or edges) between some of entities (or vertices) of various objects.

- Kuramochi, Michihiro, and George Karypis. "Frequent subgraph discovery." *Proceedings 2001 IEEE International Conference on Data Mining*. IEEE, 2001.

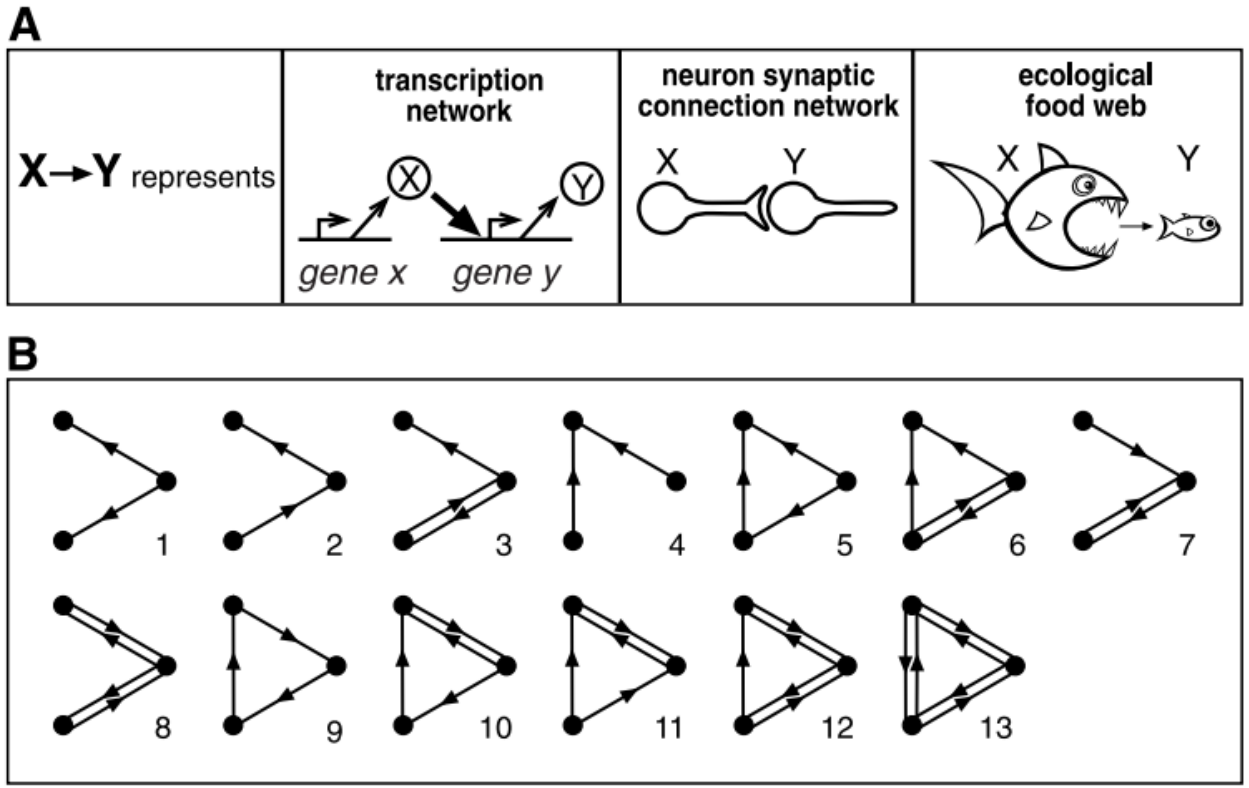
Network Motifs

- To uncover their structural design principles, we defined “network motifs,” patterns of interconnections occurring in complex networks at numbers that are significantly higher than those in randomized networks.
- Motifs may thus define universal classes of networks.
- This uncover the basic building blocks of most networks.
- It would be fascinating to see what types of motifs occur in other networks and to understand the processes that yield given motifs during network evolution.

- Milo, Ron, et al. "Network motifs: simple building blocks of complex networks." *Science* 298.5594 (2002): 824-827.

Network Motifs

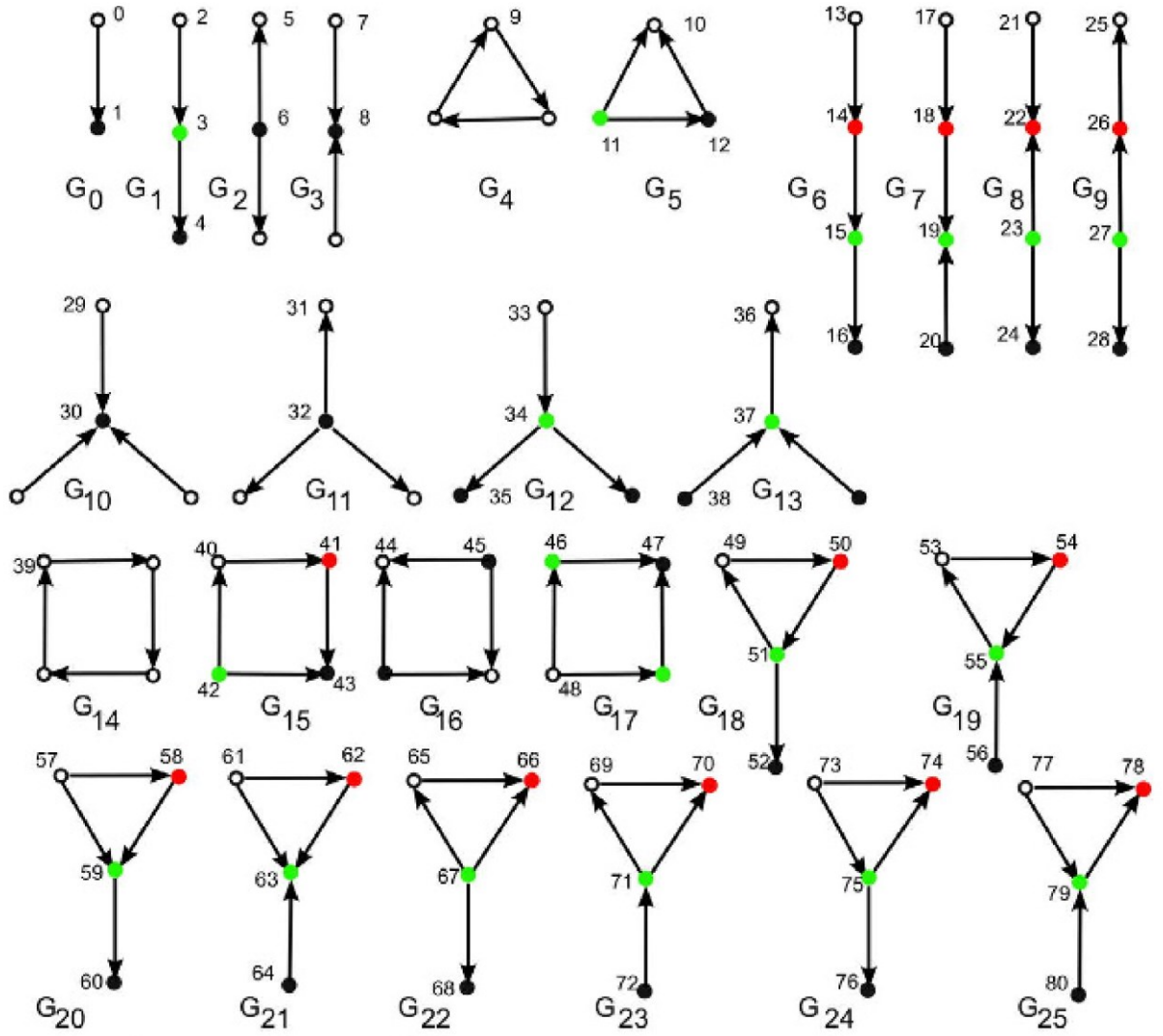
- A) Examples of interactions represented by directed edges between nodes in some of the networks used for the present study. These networks go from the scale of biomolecules (transcription factor protein X binds regulatory DNA regions of a gene to regulate the production rate of protein Y), through cells (neuron X is synaptically connected to neuron Y), to organisms (X feeds on Y).
- B) All 13 types of three-node connected subgraphs.



- Milo, Ron, et al. "Network motifs: simple building blocks of complex networks." *Science* 298.5594 (2002): 824-827.

Graphlets

- Graphlets are defined as small induced subgraphs of a large network that appear at any frequency;
- An induced sub-graph means that once you pick the nodes in the large network, you must pick all the edges between them to form the sub-graph.
- Graphlets must be induced subgraphs, while non-induced network motifs exist



- Pržulj N, Corneil DG, Jurisica I: Modeling Interactome, Scale-Free or Geometric?, *Bioinformatics* 2004, 20(18):3508-3515.
- Sarajlić, Anida, et al. "Graphlet-based characterization of directed networks." *Scientific reports* 6.1 (2016): 1-14.

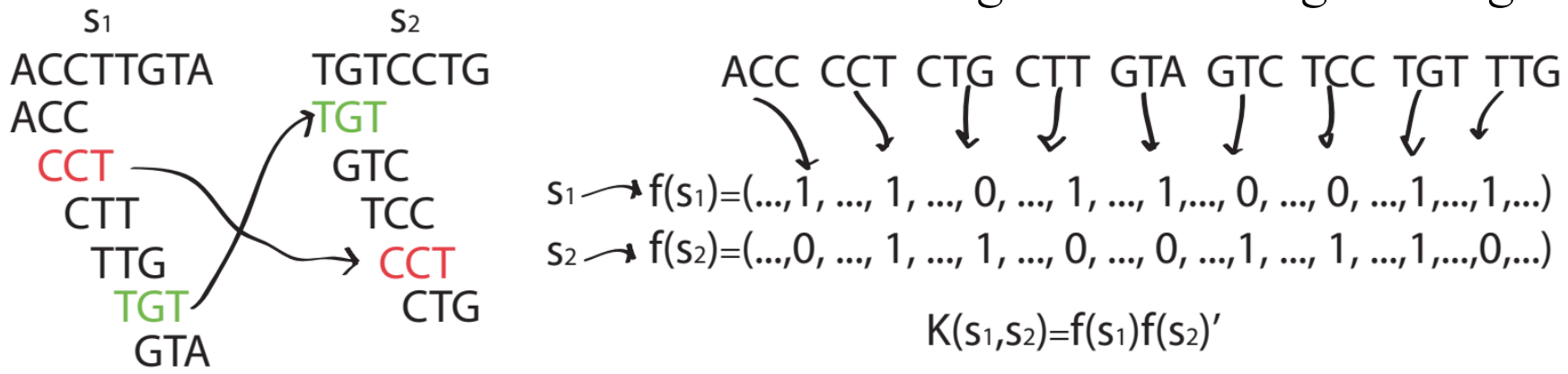
Accelerated Network Motif Detection

- Motifs have been mainly applied in Bioinformatics, regarding gene regulation networks.
- Motif detection is based on induced subgraph counting.
- An algorithm to count subgraphs of size k + 2 based on the set of induced subgraphs of size k.
- The general technique was applied to detect 3, 4 and 5-sized motifs in directed graphs.
- Problem 1 (Motif-k). Given a directed graph G(V; E), the Motif-k problem consists of counting the number of connected induced subgraphs of G of size k grouped by isomorphic distinct graphs.
- Other such tools are: NetMODE, FANMOD and Kavosh

- Meira, Luis AA, et al. "Acc-motif: accelerated network motif detection." *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 11.5 (2014): 853-862.

Biomedical Applications

- Graphlet Based Metrics for the Comparison of Gene Regulatory Networks
- Biological Network Comparison Using Graphlet Degree Distribution
- Uncover biological network function via graphlet degree signatures
- Basic idea: count the number of common contiguous substrings of length k



- Martin, Alberto JM, et al. "Graphlet based metrics for the comparison of gene regulatory networks." *PloS one* 11.10 (2016): e0163497.

- Pržulj N, Biological Network Comparison Using Graphlet Degree Distribution, *Bioinformatics* 2007, 23:e177-e183.

- T. Milenkovic and N. Przulj. (2008, Jan.). "Uncovering biological network function via graphlet degree signatures.," *arXiv, q-bio. MN*. [Online]. Available: http://arxiv.org/abs/0802.0556v1

Network Evolution Subgraph Mining for System Stability, Changeability, and Complexity

1. Animesh Chaturvedi and Aruna Tiwari. "System Network Complexity: Network Evolution Subgraphs of System State Series." *IEEE Transactions on Emerging Topics in Computational Intelligence*, Vol 4.2 (2018): 130-139. DOI: 10.1109/TETCI.2018.2848293. (IEEE Computer Society and IEEE Computational Intelligence Society)
2. Animesh Chaturvedi and Aruna Tiwari. "System Evolution Analytics: Evolution and Change Pattern Mining of Inter-Connected Entities". *48th 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 3075-3080). IEEE.

Proposed Definitions

- A *network evolution subgraph* is a subgraph with changing frequencies in the state series SS
  - *network evolution frequent subgraph*: frequently occurring in a network;
  - *network evolution motif*: subgraph is statistically recurrent in a network; and
  - *network evolution graphlet*: subgraph is induced subgraph in a network.
- The $Aggregate\_freq_j$ denotes *aggregate frequency* of a NEG G_j.
- The arithmetic mean of frequency over N states for a NEG G_j is given by

$$Aggregate\_freq_j = \frac{\sum_{i=1}^{N} freq_{ji}}{N}$$

where, $freq_{ji}$ is the *frequency* of NEG G_j in state S_i with i varying from integer 1 to N and j is constant.

Proposed Definitions

- *Network Evolution Graphlets information* is a doubleton set of retrieved NEGs (as a subgraph) and their *aggregate frequencies* $Aggregate\_freq_j$.

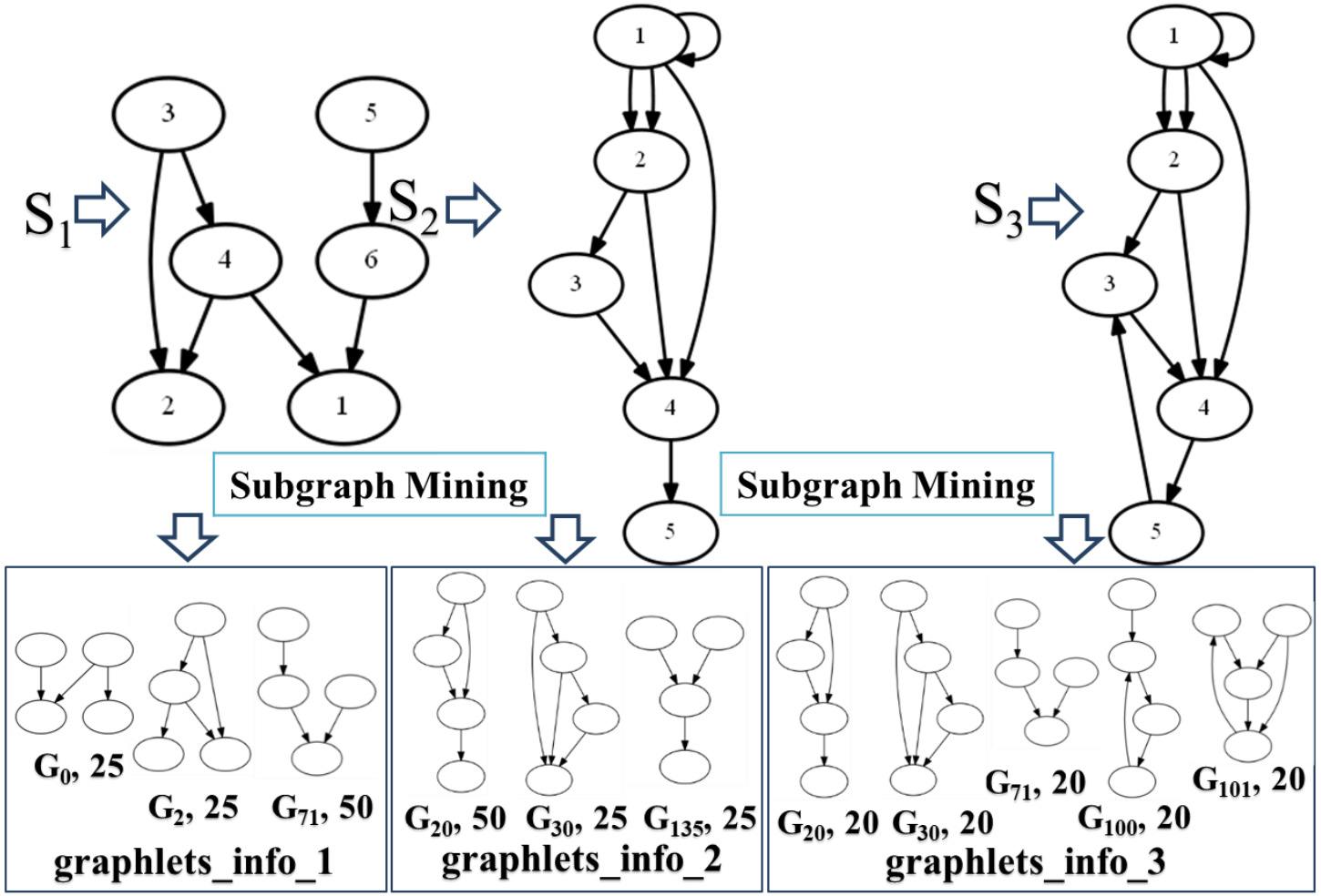
$$NEGs\_info = < G_j, Aggregate\_freq_j > \mid 0 \leq j, m' \leq M$$

where,

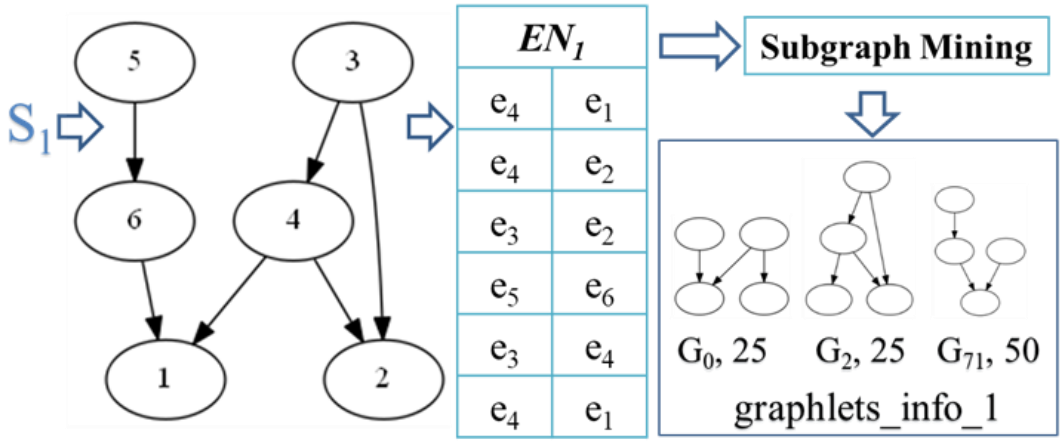
- G_j is j^{th} NEG with aggregate frequency as $Aggregate\_freq_j$ and j is the enumeration of the NEG;
- m' is the number of retrieved NEGs (distinctly non-redundant graphlets) over all the states in SS.

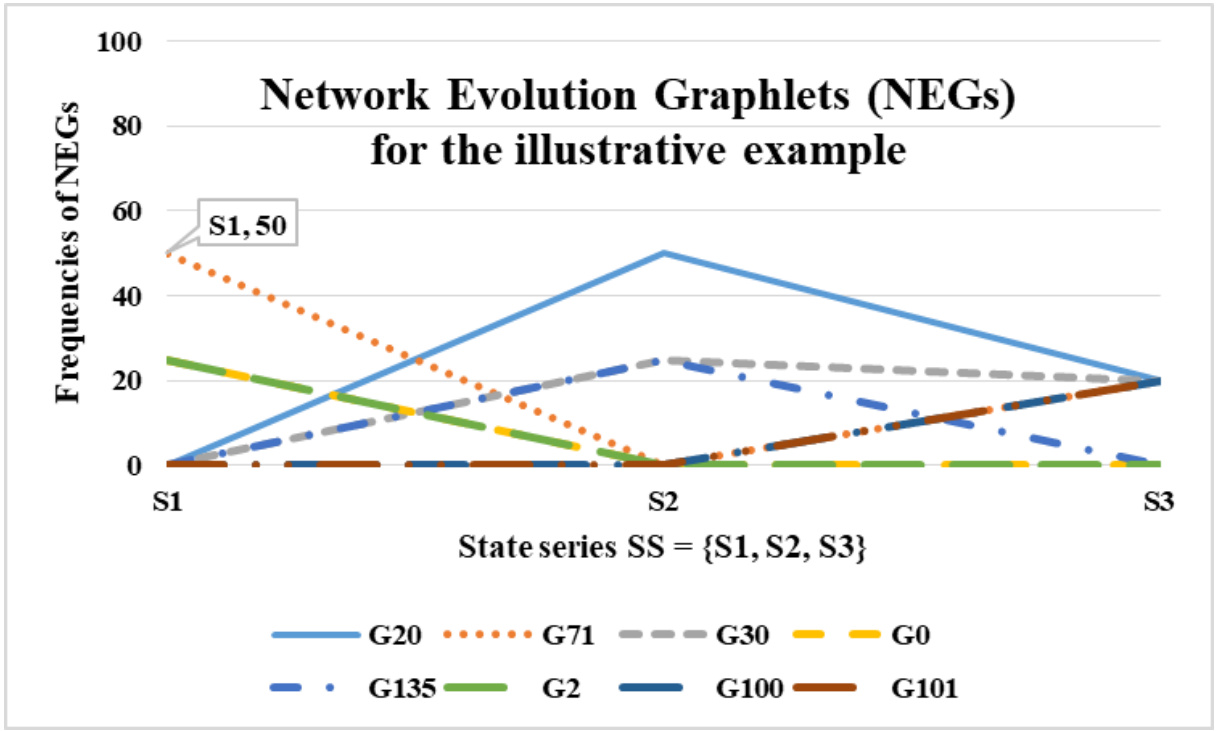
Illustrative example

Subgraph mining for a state series SS = $\{S_1, S_2, S_3\}$ represented as three evolving networks.



Illustrative example





- The graphlets information for state
  S_2 [<G_{20}, 50>, <G_{30}, 25>, <G_{135}, 25>]
  S_3 [<G_{20}, 20>, <G_{71}, 20>, <G_{30}, 20>, <G_{100}, 20>, <G_{101}, 20>].
- *NEGs_info* is
  [<G_{20}, 23.33>, <G_{71}, 23.33>, <G_{30}, 15>, <G_0, 8.33>, <G_{135}, 8.33>, <G_2, 8.33>, <G_{100}, 6.66>, <G_{101}, 6.66>]
- The NEMs are [<G_{20}, 23.33>, <G_{71}, 23.33>] if the threshold frequency is 20%.
- The changeability and stability are 320 and 0.00312 respectively.

System State Complexity (SSC) & Evolving System Complexity (ESC)

- System State Complexity of S_i (SSC_i) $= \frac{\sum_{j=0}^{m}(freq_{ji} \times C_j)}{\sum_{j=0}^{m} freq_{ji}}$

where C_j denotes cyclomatic complexity and $freq_{ji}$ represents frequency for graphlet G_j of state S_i and m is count of retrieved graphlets.

- The SSCs information $SSCs\_info = < S_i, SSC_i > \mid 0 \leq i \leq N$

- Evolving System Complexity $= \frac{\sum_{j=0}^{m'}(Aggregate\_freq_j \times C_j)}{\sum_{j=0}^{m'} Aggregate\_freq_j}$

where, $Aggregate\_freq_j$ denotes the *aggregate frequency* of a graphlet G_j over time, C_j denotes the *cyclomatic complexity* of the graphlet G_j and m' is count of retrieved NEGs.

Illustrative example

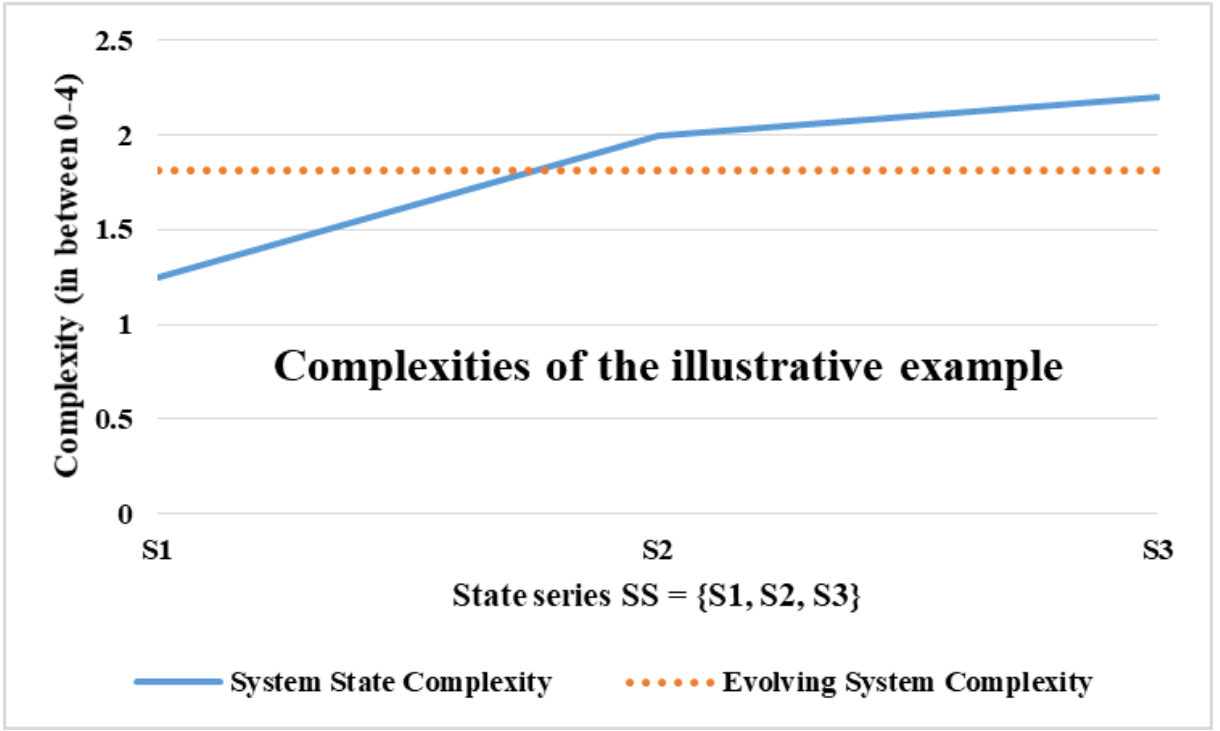
- Complexity calculation for three states of an evolving system
- For state S_1 graphlets (G_0, G_2, and G_{71}) has cyclomatic complexity
  - ($C_0 = 1$, $C_2 = 2$, $C_{71} = 1$) and frequency ($freq_{0,1} = 25$, $freq_{2,1} = 25$, $freq_{71,1} = 50$)

- SSC for S_1 is calculated as

$$= \frac{(25 \times 1) + (25 \times 2) + (50 \times 1)}{25 + 25 + 50}$$

$$= 1.25$$

- ESC for state series SS = $\{S_1, S_2, S_3\}$ is
  - 1.816



System Network Complexity

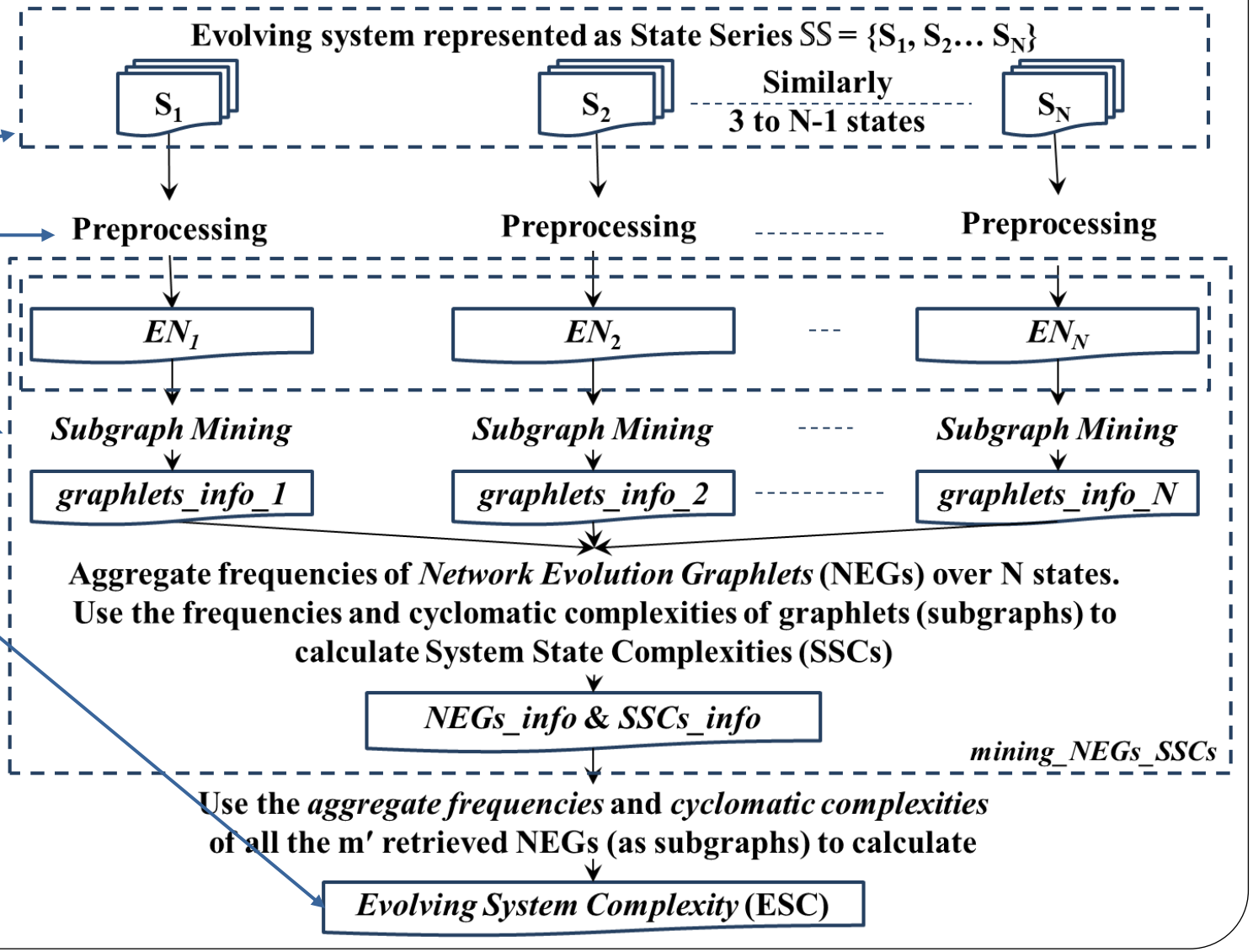
An overview of flow-chart

Algorithm SNC(*repository*)

Initialize $i \in$ integer 1 to N

Retrieve N states of a *state series* SS $= \{S_1, S_2 \ldots S_N\}$ stored in *repository*

1. *evolvingNetworks* = ***Preprocess***(*repository*)
2. *NEGs_info* & *SSCs_info* = ***mining_NEGs_SSCs***(*evolvingNetworks*)
3. *ESC* = ***Calculate_ESC***(*NEGs_info*)



Algorithm 2: mining_NEGs_SSCs(*evolvingNetworks*)

Initialize $j \in$ integer for graphlet G_j such that $0 \leq j \leq M$
Initialize $i \in$ integer varying from 1 to N states
Initialize HashMap *graphlets_info*$< G_j, freq_{ji} >$
Initialize HashMap *NEGs_info*$< G_j, Aggregate\_freq_j >$
Initialize HashMap *SSCs_info*$< S_i, SSC_i >$

Where, G_j is j^{th} *graphlet*, $freq_{ji}$ is *frequency* of G_j in state S_i, and $Aggregate\_freq_j$ is aggregate frequency of NEG G_j over a state series.
For each EN_i in *evolvingNetworks* where *i* varies from 1 to N
graphlets_info_i $< G_j, freq_{ji} >$ = ***subgraphMining***(EN_i)
End For
Let m' is count of retrieved graphlets over all states
For each G_j in m' graphlets, where $1 \leq j, m' \leq M$
Initialize float $frequencySum = 0$
Initialize integer $Aggregate\_freq_j = 0$
  For each *graphlets_info_i* where *i* varies from 1 to N
  $frequencySum = frequencySum + freq_{ji}$
  End For
  $Aggregate\_freq_j = frequencySum \div$ N
  Add tuple $< G_j, Aggregate\_freq_j >$ to *NEGs_info*
End For
Initialize cyclomatic complexity array C[M] for all graphlets
For each *graphlets_info_i* of S_i where *i* varies from 1 to N
Initialize float $frequencySum = 0$
Initialize float $sumOfProducts = 0$
  For each G_j in *graphlets_info_i*
  $frequencySum = frequencySum + freq_{ji}$
  $sumOfProducts = sumOfProducts + \{freq_{ji} \times C_j\}$
  //where *Cyclomatic complexity* C_j for graphlet G_j at C[j]
  End For
  Float $SSC_i = sumOfProducts \div frequencySum$
  Add tuple $< S_i, SSC_i >$ to *SSCs_info*
End For
Return *NEGs_info* & *SSCs_info*

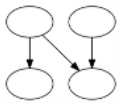
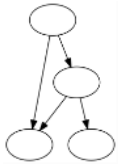
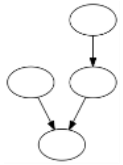
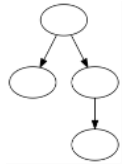
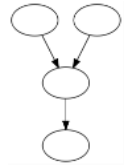
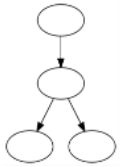
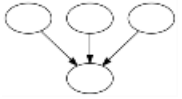
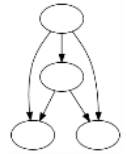
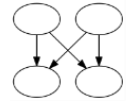
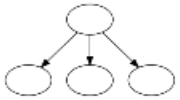
Algorithm 3: Calculate_ESC(*NEGs_info*)

Let m' is the number of retrieved graphlets in *NEGs_info*
Initialize $j \in$ integer for graphlets such that $0 \leq j \leq M$
Initialize float $sumOfProducts = 0$
Initialize cyclomatic complexity array C[M] for all graphlets
$G_j \in j^{th}$ NEG in *NEGs_info*
Initialize $Aggregate\_freq_j = 0$
$Aggregate\_freq_j \in$ *aggregate frequency* of G_j in *NEGs_info*
For each NEG G_j in *NEGs_info*
  $sumOfProducts = sumOfProducts + \{Aggregate\_freq_j \times C_j\}$
  //where *Cyclomatic complexity* C_j for graphlet G_j at C[j]
  $frequencySum = frequencySum + Aggregate\_freq_j$
End For
Float $ESC = sumOfProducts \div frequencySum$
Return *ESC*

Experiments on Evolving Systems

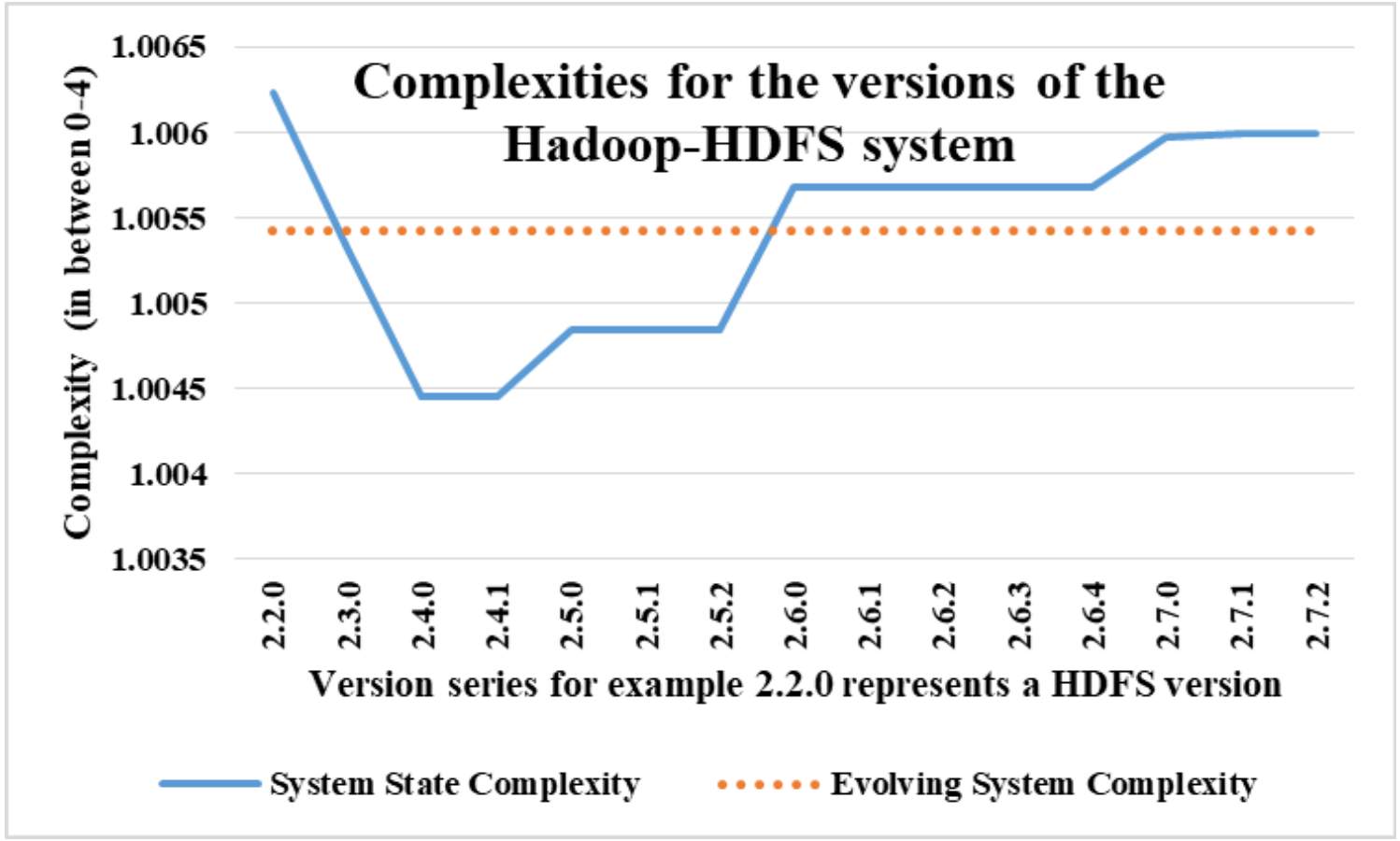
Evolving Systems	N	# entities	Average # neighbour	# NEGs	ESC*
HDFS-Core	15	3129	2.166	24	1.0054
Bible Translation	13	246	1.456	17	1.456
Multi-sport Events	13	141	1.786	10	1.00487
Frequent Market Basket[4]	13	118	8.002	34	1.33888
Positive sentiment of movie genres	16	284	2.661	4	1.03050
Negative sentiment of movie genres	16	510	3.303	20	1.03683

Network Evolution Motifs (NEMs) and
their complexities according to their subgraph patterns

$M_0, C_0 = 1$	$M_2, C_2 = 2$	$M_{71}, C_{71} = 1$	$M_{87}, C_{87} = 1$	$M_{135}, C_{135} = 1$
				
$M_{156}, C_{156} = 1$	$M_{178}, C_{178} = 1$	$M_{185}, C_{185} = 3$	$M_{192}, C_{192} = 2$	$M_{198}, C_{198} = 1$
				

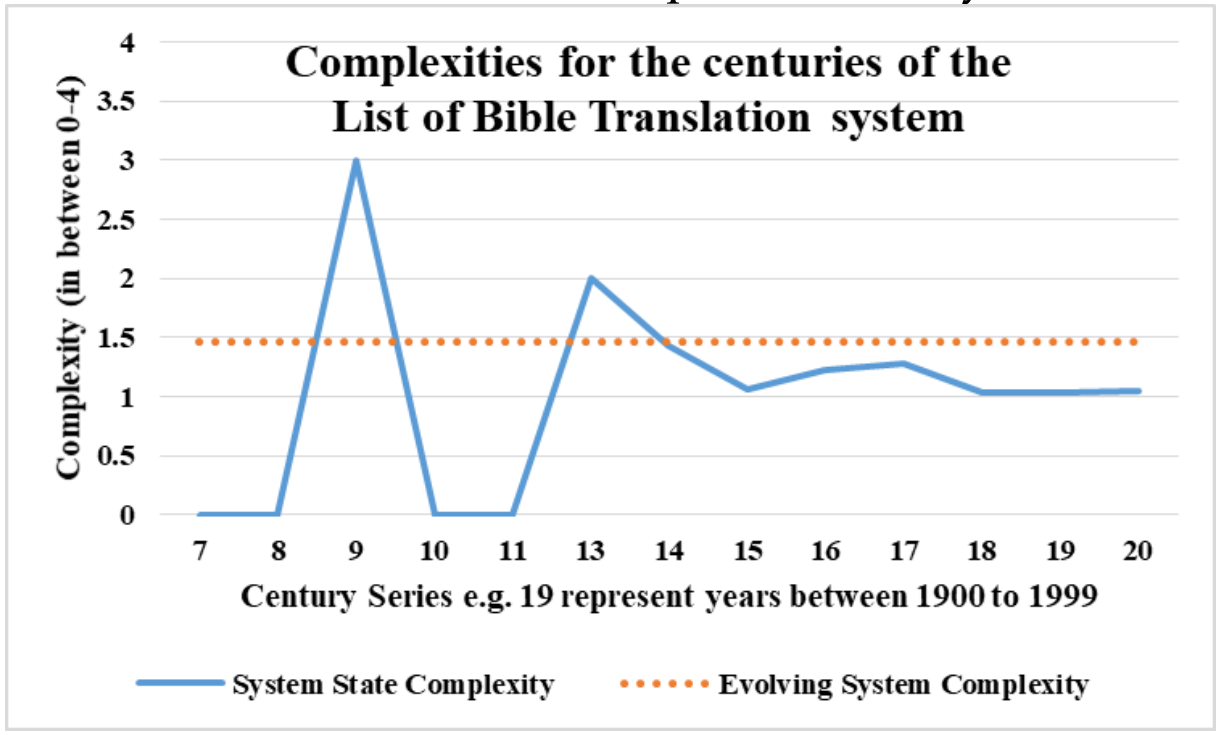
Software Evolution Analytics

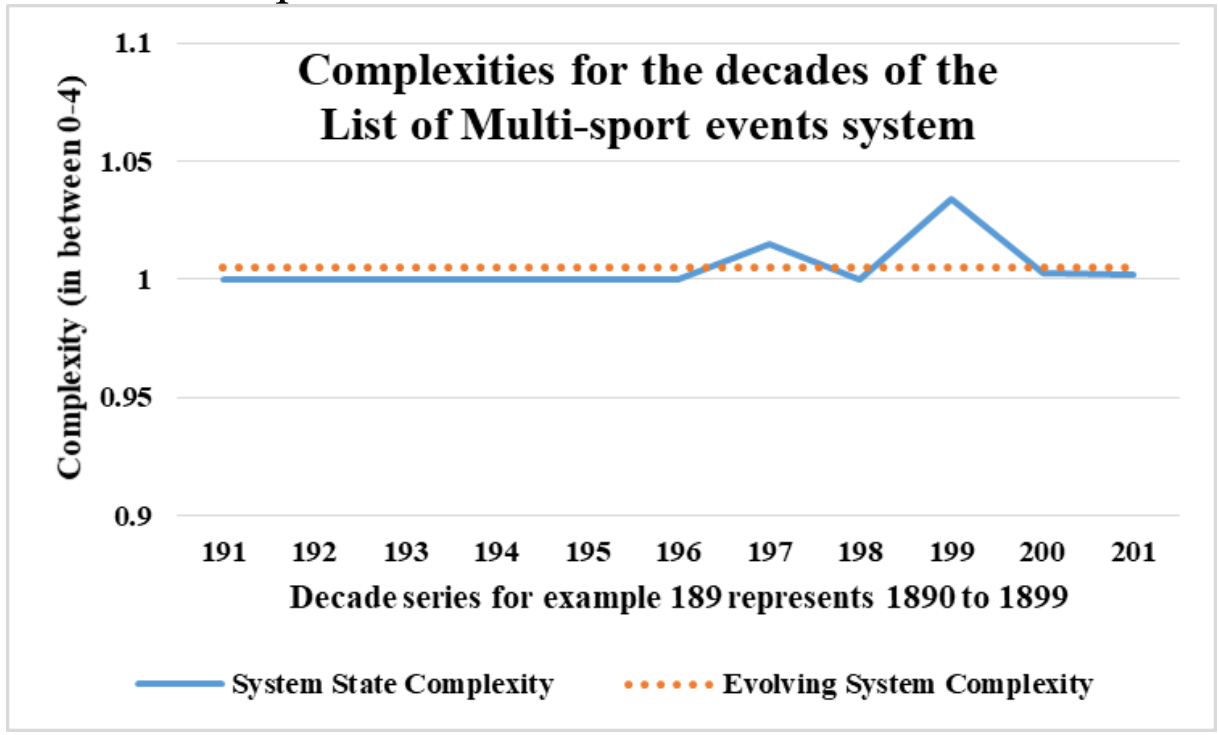
- Evolving Software Systems:
  - Hadoop – HDFS available on Software Repository



Natural-Language Evolution Analytics

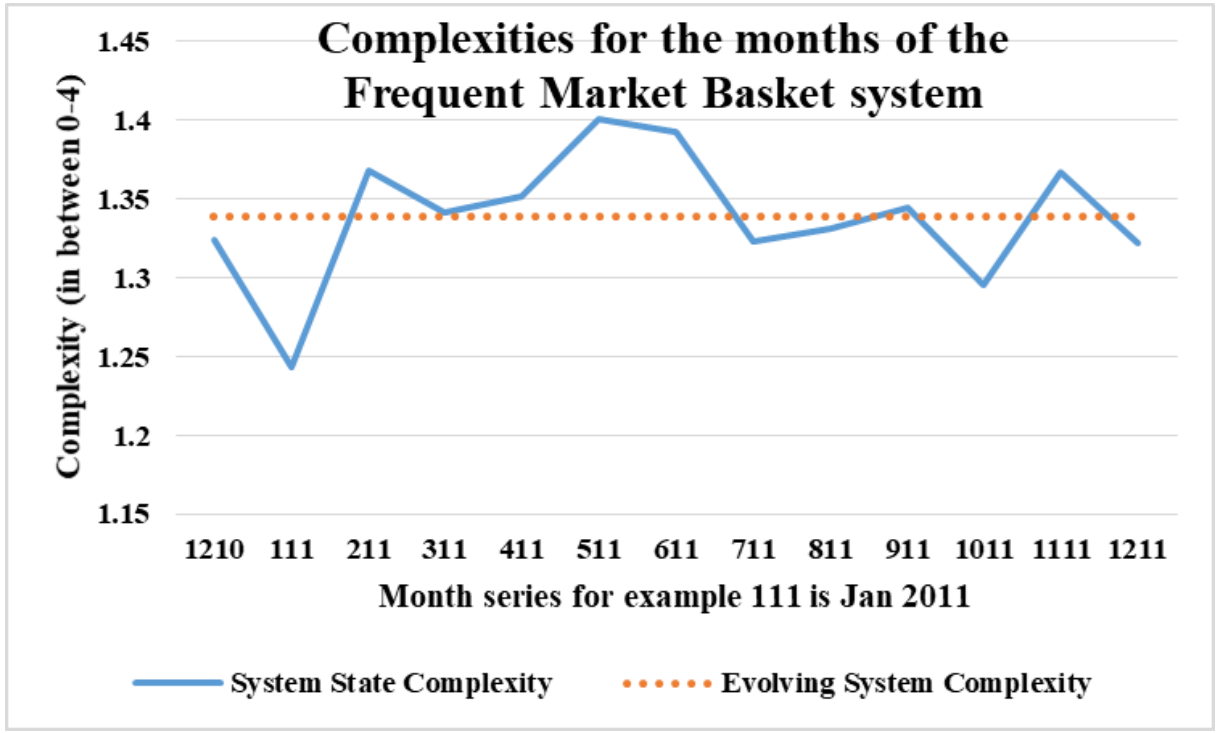
- Two Evolving Natural-Language systems:
  - List of Bible Translation system available on Wikipedia
  - List of Multi-sport events system available on Wikipedia





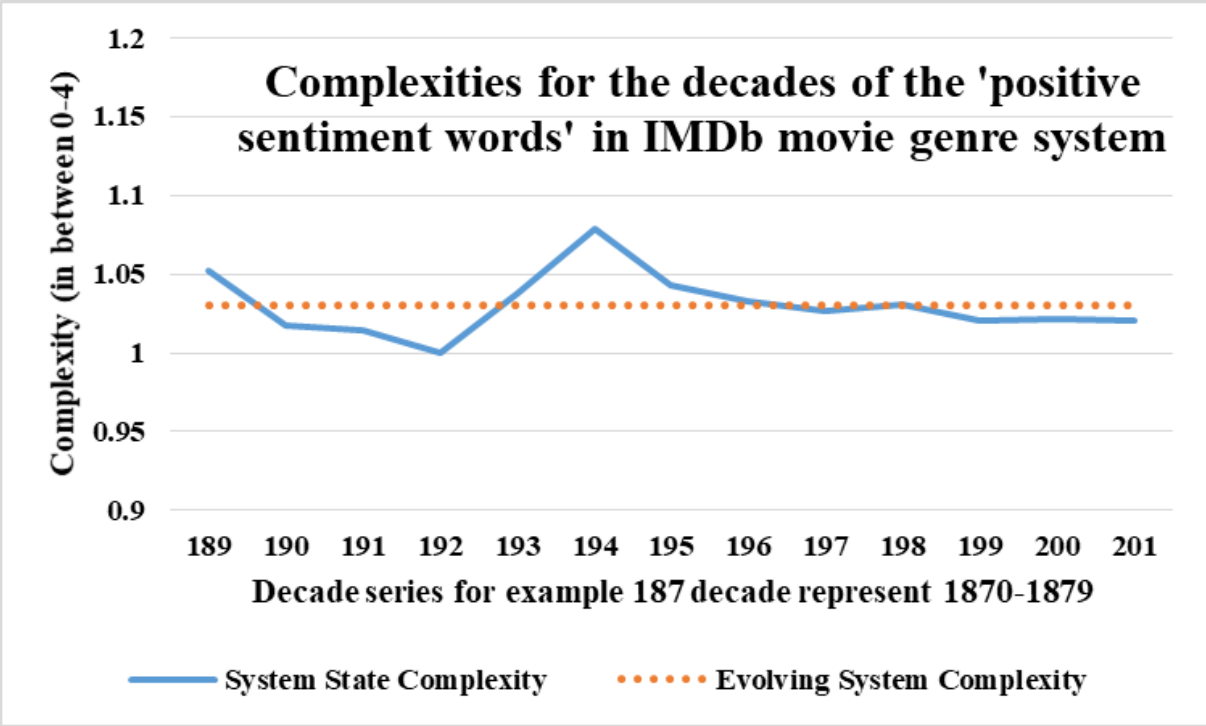
Market Evolution Analytics

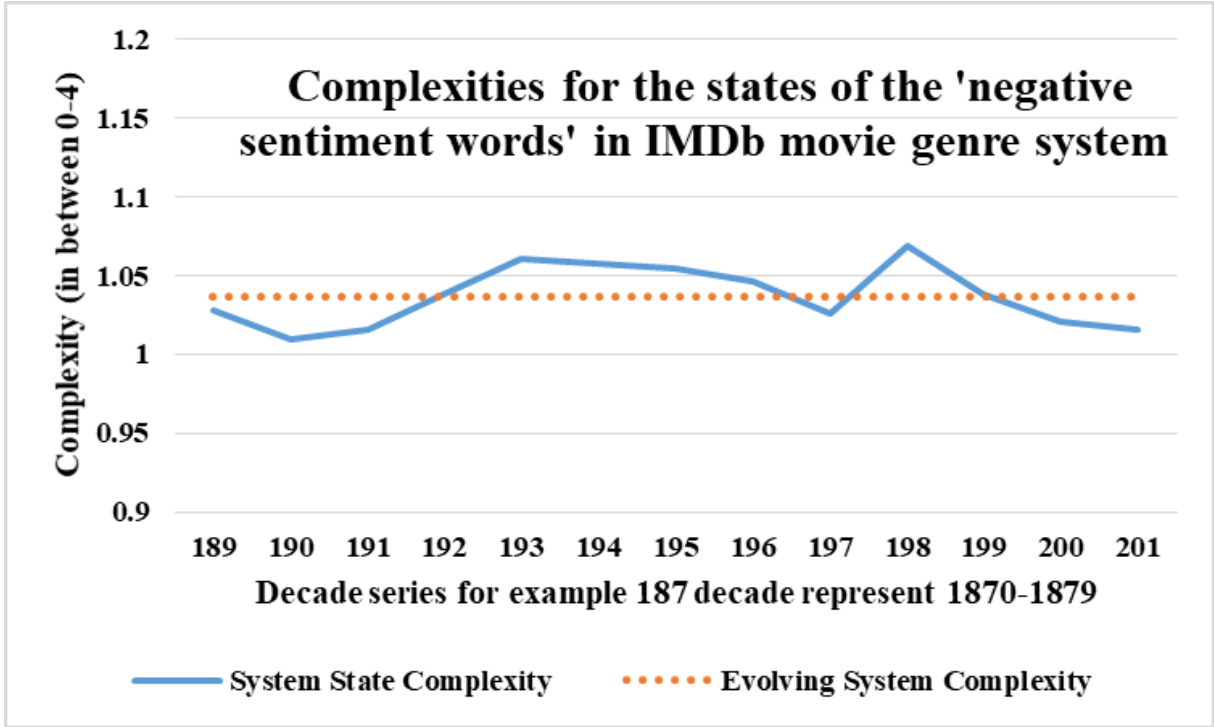
- Evolving Retail Market system:
  - Frequent Market Basket system made from retail-market baskets available on UCI Repository



Movie Evolution Analytics

- Two Evolving IMDb movie genre systems made from IMDb Repository
  - Positive sentiment system
  - Negative sentiment system





ขอบคุณ
Thai

Grazie
Italian

תודה רבה
Hebrew

धन्यवादः
Sanskrit

ಧನ್ಯವಾದಗಳು
Kannada

Ευχαριστώ
Greek

Thank You
English

Gracias
Spanish

Спасибо
Russian

Obrigado
Portuguese

شكراً
Arabic

https://sites.google.com/site/animeshchaturvedi07

Merci
French

多謝
Traditional
Chinese

धन्यवाद
Hindi

Danke
German

多谢
Simplified
Chinese

நன்றி
Tamil
Tamil

ありがとうございました
Japanese

감사합니다
Korean