Chapter 6. Examples for Advanced PGM Representation

PGM 2021

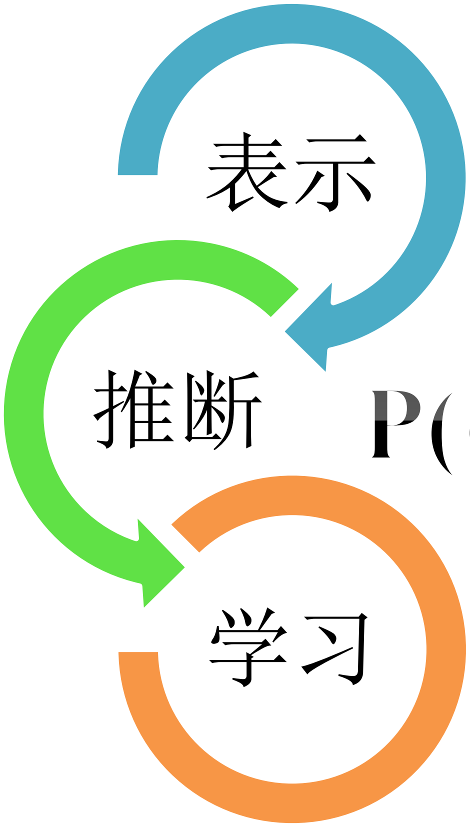
Jin Gu (古槿)

Outlines

- Short reviews on *Representation*
  - Equivalence between probability and graph
  - Comments on undirected models
- Model conditional information
- Model context and complex information
- Model high-order information
  - Disadvantages and advantages of deep structures
  - Representative deep structure models
- Model hierarchical information
  - Samples are independent given parameters
  - Latent *Dirichlet* allocation (LDA) models

课程内容体系

$$P \Leftrightarrow \{P, G/H\}$$



$$P(C|X=x,\theta)$$

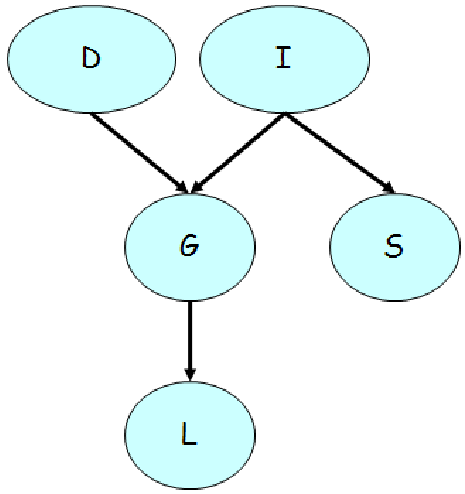
$$P(\theta\,|x^1, x^2,\ldots)$$

课程内容体系（表示部分）

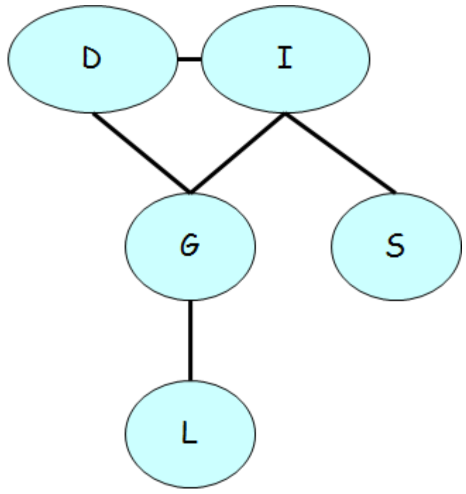
- 贝叶斯网络 有向无环图
- 马尔可夫网络 无向图
- 动态/序列模型 有环图
- 连续分布的概率图模型

课程内容体系（两大类模型比较）

- Bayesian Networks
  - Local structures
    - Parents→Node ⇔ local conditional probability
  - Joint probability P
    - Product of LCPs
  - Markov blanket
    - Parents, children and their parents



- Markov Networks
  - Local structures
    - Cliques ⇔ positive local factors (un-normalized)
  - Joint probability P
    - Gibbs distribution
    - Log-linear representations
  - Markov blanket
    - All neighbor nodes



课程内容体系（MN模型补充说明）

- Local probability on any maximal clique i cannot be calculated without knowing other cliques (global context-dependence)
- According to a constructive proof of Hammersley-Clifford Theorem, undirected graphical models are usually represented in log-linear format rather than factor product
- The exponential term is called as potential function: linear combination on all the cliques:
  - $P = \frac{1}{Z} e^{-U(x)}$ (Gibbs distribution)
  - $U(\boldsymbol{x}) = -\sum_{C_i} \left[\psi_i(C_i) \prod_{x_j \in C_i} x_j \right]$

课程内容体系（“表示”三步走）

1、定义随机变量

2、绘制图模型拓扑结构

3、确定局部概率模型

检查上述三步是否合理

Model Conditional Information

Only Model the Conditional Probability

An Example: Word POS Identification

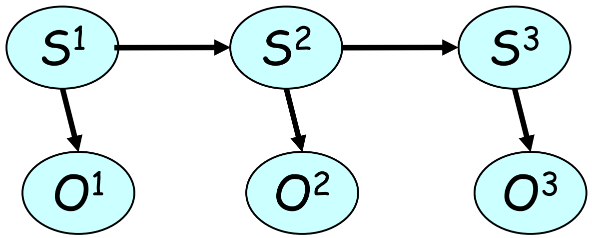
- Given a sentence, we need to infer the part of speech (POS, 词性) of each word in the sentence

- The robot wheels Fred round.
  - na. + n. + vt. + n. + adv.
- The robot wheels are round.
  - na. + n. + n. + aux.v. + adj.

- How about HMMs (model the POS as the hidden states and the words as the observations)?

An Example: Word POS Identification

- Given sentences, we need to infer the part of speech (词性) of each word in the sentence



- The disadvantages of HMMs
  - The dimensions of observations are too high (equal to the number of words)
  - Actually, we do not want to know the probability of the words given the POS

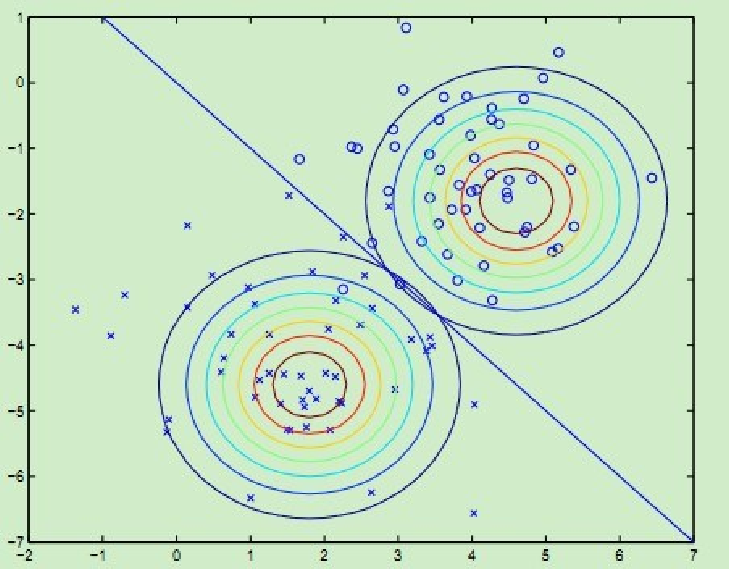
Generative / Discriminative Models

- **Generative model**: you need to model the joint distribution of all variables, including the observation variables ***X*** and the state variables ***Y***. HMM is a generative model.

- **Discriminative model**: To infer the state variables ***Y*** (sequence labels or sample class assignments), you only need to model the conditional distribution of the state variables given the configuration of the observation variables ***Y***|***X***.

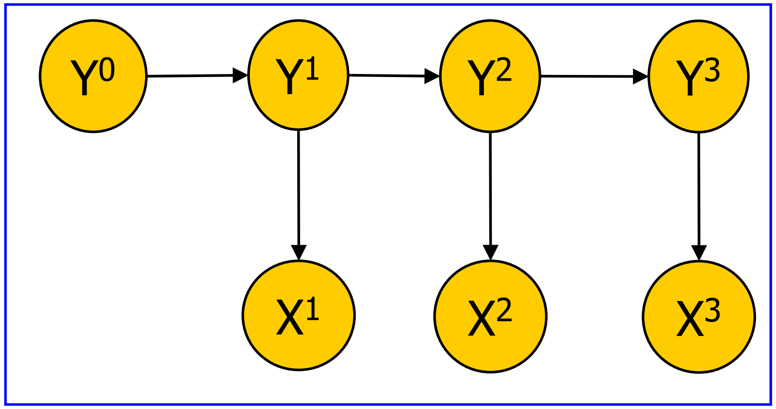
A Simple Example with Two Classes

- Generative models
  - Estimate the distribution of each class
  - Then, find the optimal cutoff for classifying
- Discriminative models
  - Directly find the boundaries of the two classes

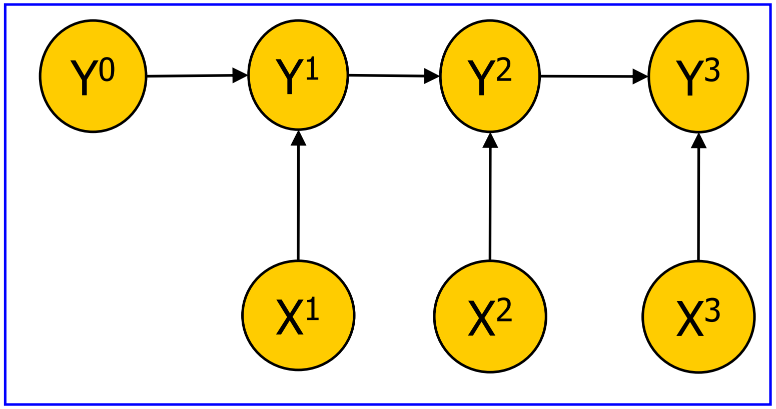


Max Entropy Markov Models

- In machine learning, a maximum-entropy Markov model (MEMM), or conditional Markov model (CMM), is a graphical model for sequence labeling that combines features of hidden Markov models (HMMs) and maximum entropy (MaxEnt) models. An **MEMM is a discriminative model** that extends a standard maximum entropy classifier by assuming that the unknown values to be learned are connected in a Markov chain rather than being conditionally independent of each other.



HMM

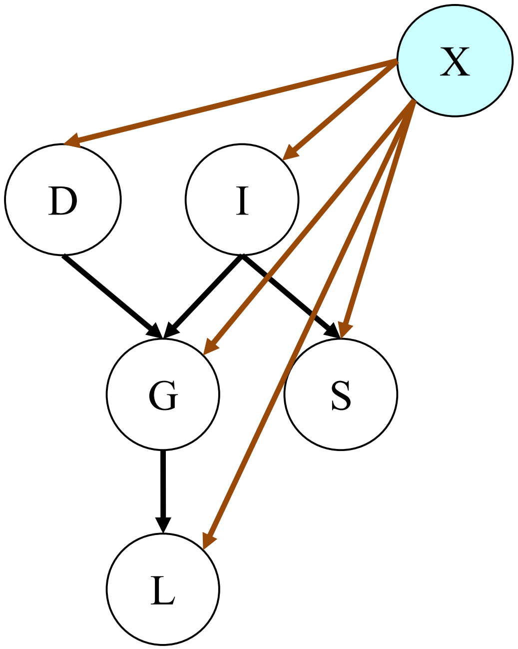


MEMM

Generalized Conditional BNs

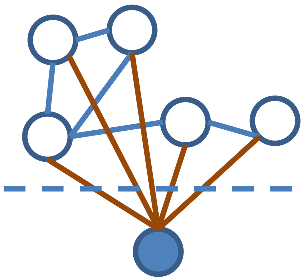
- The *conditional nodes* are always parents of other nodes
- The dependences and independences are preserved in the other parts

$$G \perp S \mid I, X$$
$$D \perp L \mid G, X$$



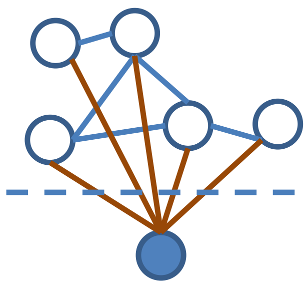
CRFs: Generalization to Markov networks

- Let $H = (V, E)$ be a graph such that variables $\boldsymbol{Y} = (Y_v)$ are indexed by the vertices of H.
- Then $(\boldsymbol{X}, \boldsymbol{Y})$ is a conditional random field in case, when conditioned on $\boldsymbol{X}$, the random variables Y_v obey the Markov property with respect to the graph:
- $p(Y_v|\boldsymbol{X}, \boldsymbol{Y}_{-v}) = p(Y_v|\boldsymbol{X}, \boldsymbol{Y}_w, w\sim v)$, $w\sim v$ means the neighbors of Y_v in H.



Gibbs Distribution of CRFs

- If omitting high-order interactions, other cliques are the edges and vertices (all linked to ***X***)
- By the fundamental theorem of undirected graphical models – **Hammersley & Clifford Theorem**, the joint distribution over ***Y*** given ***X*** has the form:



$$P(Y,X;\theta) \propto P(Y \mid X;\theta) \propto$$

$$\exp\left(\sum_{e \in E} \sum_{k} \lambda_k f_k \left(e, Y|_e, X\right) + \sum_{v=V} \sum_{l} \mu_l g_l \left(v, Y|_v, X\right) \right)$$

Drawbacks of Discriminative Models

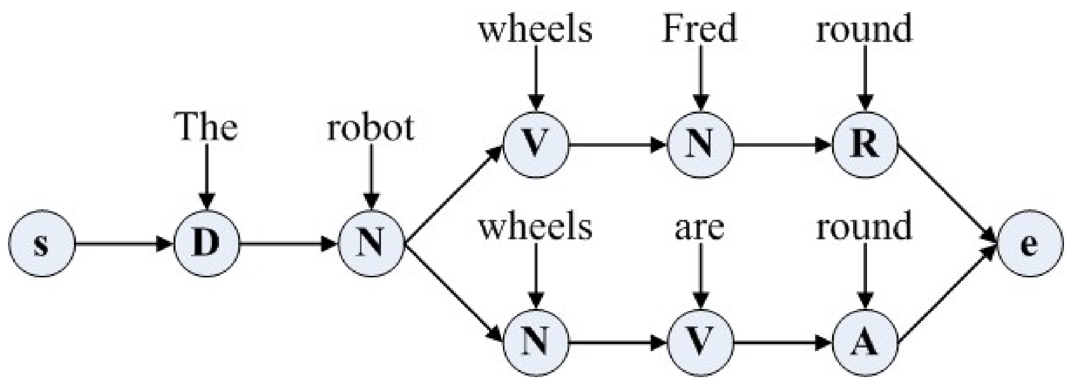
- You don't really know how the things work, unless you can make one
- You don't really know what you are working on, unless you can let anybody understand

- The outliers located around the boundary will cause misleading discriminative planes
- The models cannot generate new data. For examples, HCIs need the computer generates human-like voices

Model Context and Complex Information

Recall: Word POS Identification

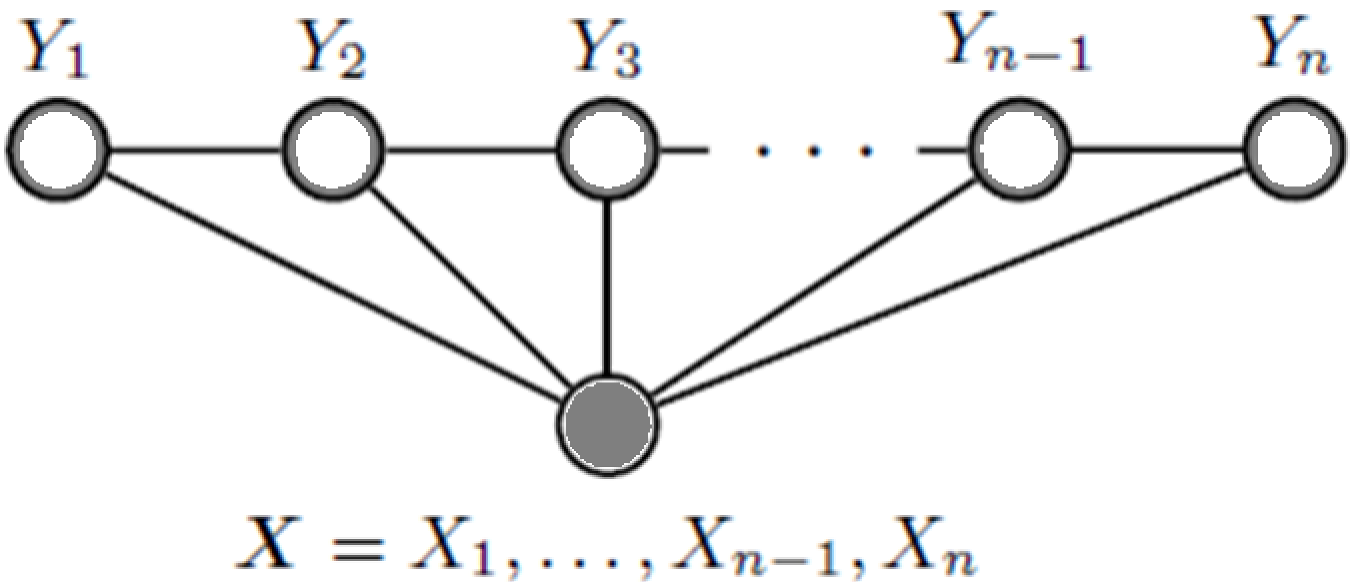
- Given a sentence, we need to infer the part of speech (POS, 词性) of each word in the sentence



- How to represent the following features?
  - The probability for a *noun* increases, if this word located at the beginning of the sentence
  - The probability for a *noun* increases, if the word is capitalized and not at the beginning of the sentence
  - The probability for a *nou*n increases, if the previous is a *vt*.

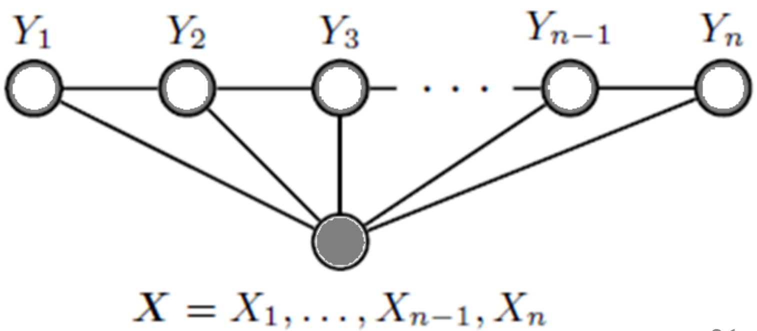
Linear CRFs

- Linear CRFs use a single log-linear model for the joint probability of the entire sequence of labels given the observation sequence.

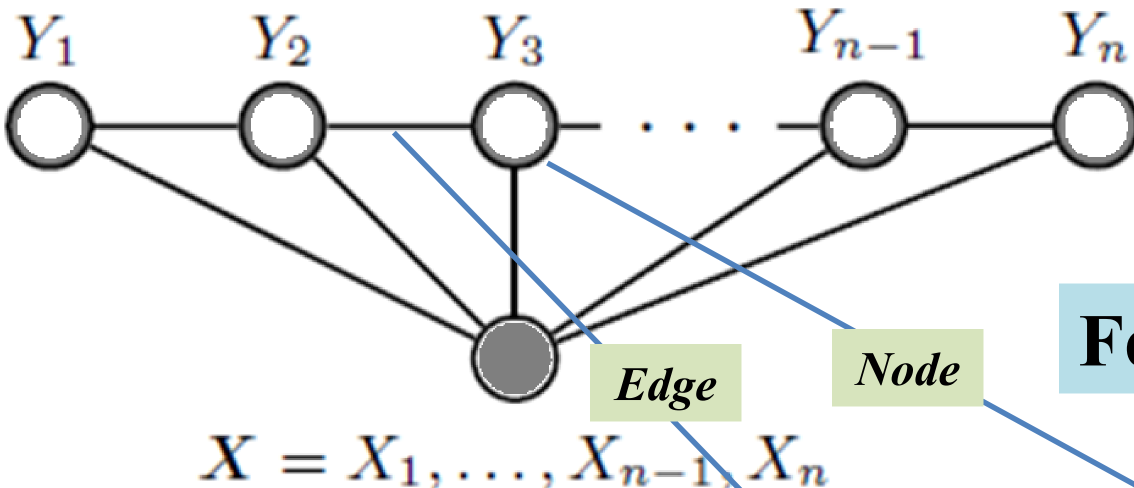


Feature Functions

- Feature functions in CRFs are defined as *indicator functions* of the state and a sub-sequence of observation, such as $f_k(X, Y_i) = 1$ if "Y_i has value *noun* and the current word is capitalized".
- Each feature function encodes one node or two neighbor nodes of the chain and a sub-sequence of *X*.
- The feature functions associated with the same nodes or edges can be grouped as the terms associated with the cliques in the graphical models



Linear-Chain CRFs: the Gibbs Distributions



Edge

Node

Feature functions

$$P(Y \mid X;\theta) \propto \exp\left(\sum_{i=1}^{n} \sum_{k} \lambda_k f_k\left(Y_{i-1}, Y_i, X, i\right) + \sum_{i=1}^{n} \sum_{l} \mu_l g_l\left(Y_i, X, i\right) \right)$$

$$\boxed{P(Y \mid X;\theta) \propto \exp\left(\sum_{i=1}^{n} \sum_{m} w_m f_m\left(Y_{i-1}, Y_i, X, i\right) \right)}$$

Another Example: Name Entity Labeling

- We need to extract the name entities in large-scale texts, such as gene names in abstracts, people names in newspaper and city names in magazines.
- Our training data are many texts labeled by some experts (*supervised learning*).
- All we need is to give the feature functions:
  - The first letter of current word is capitalized
  - The next word is a *verb*
  - The previous word is "Mr."
  - …..
- Learn the weights of all features

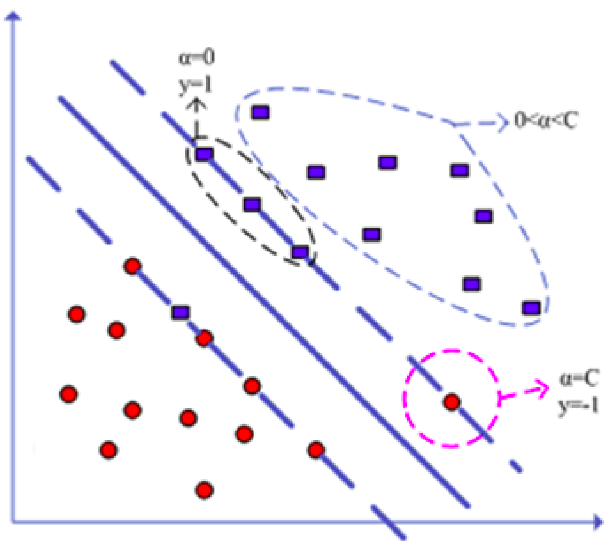
$$\boxed{P(Y \mid X;\theta) \propto \exp\left(\sum_{i=1}^{n}\sum_{m} w_m f_m\left(Y_{i-1}, Y_i, X, i\right)\right)}$$

Deep Structures

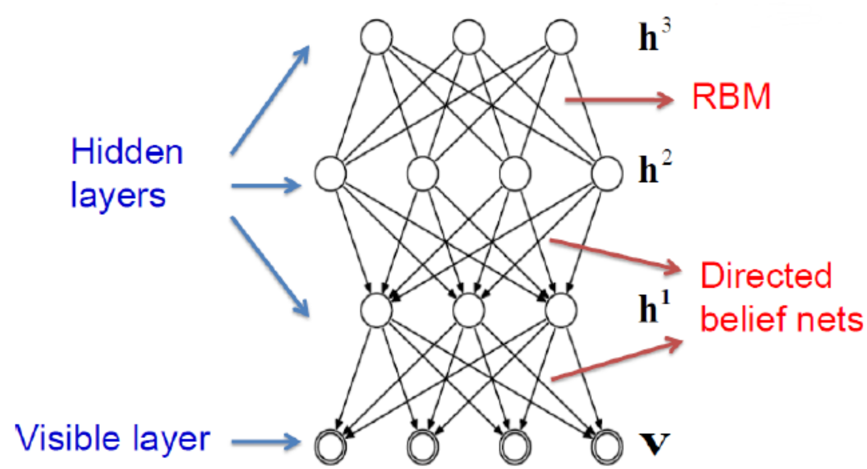
An Introduction to DL Representation

“Shallow” models vs “Deep” models

- Why shallow?
  - Reduce structure risk for overfitting
  - We only have “**small**” labeled data for **discriminative** models

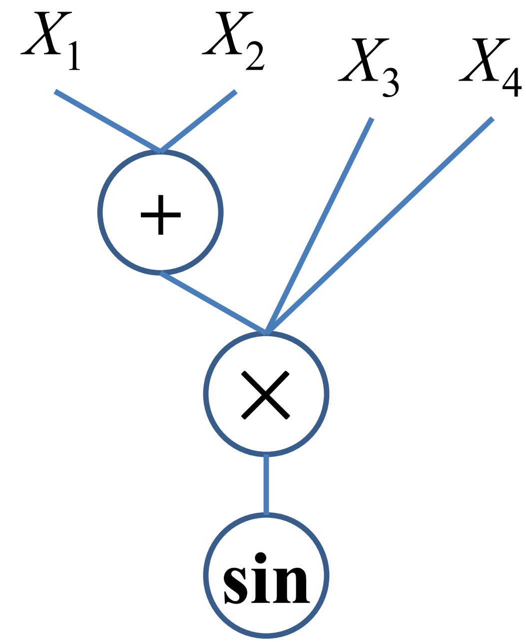


- Why deep?
  - Model complexity should be comparable with data complexity
  - We can obtain “**big**” unlabeled data for **generative** models



A Simple Example of "Deep" Structures

- $Y \sim \sin\big((X_1 + X_2)X_3X_4\big)$
- You cannot fully represent above functions if only two-layer models are used
- At least three layers are needed!



If the data are "*big enough*", we need to use *deep structures* to fully represent the *generative* processes!

Difficulties for Deep Neural Networks

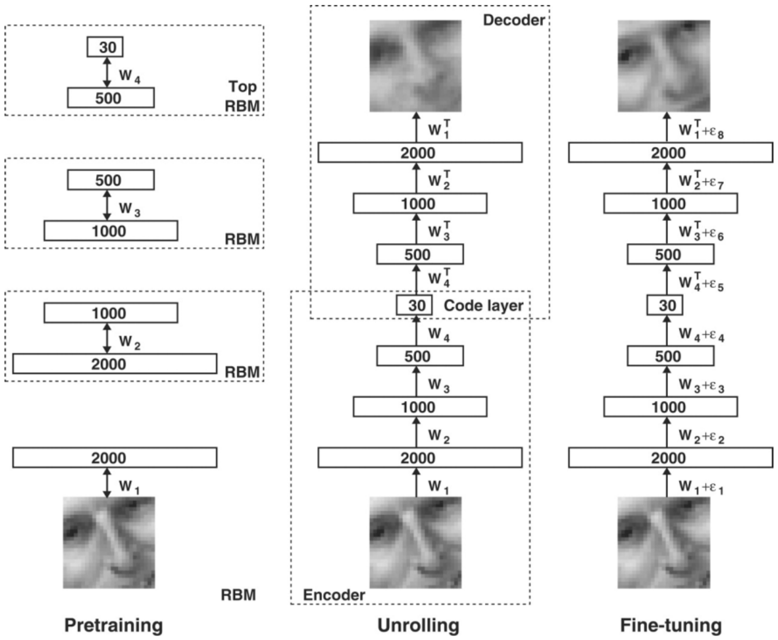
- Please refer to Geoffrey Hinton's tutorial
  - "UCL Tutorial on: Deep Belief Nets, 2009"

- Difficulties
  - Limited number of labeled data
  - Local optima
  - Explaining away
    - Even if two hidden causes are independent, they can become dependent when we observe an effect that they can both influence

Generative & Undirected Representation

- "Big" unlabeled data and "small" labeled data
  - Use generative representation on X using big unlabeled data $P(X)$, and then finalize the discriminative model based on small labeled data $P(Y|X)$
  - Use undirected restricted Boltzmann machines instead of directed neurons
  - Pre-train one layer by one layer

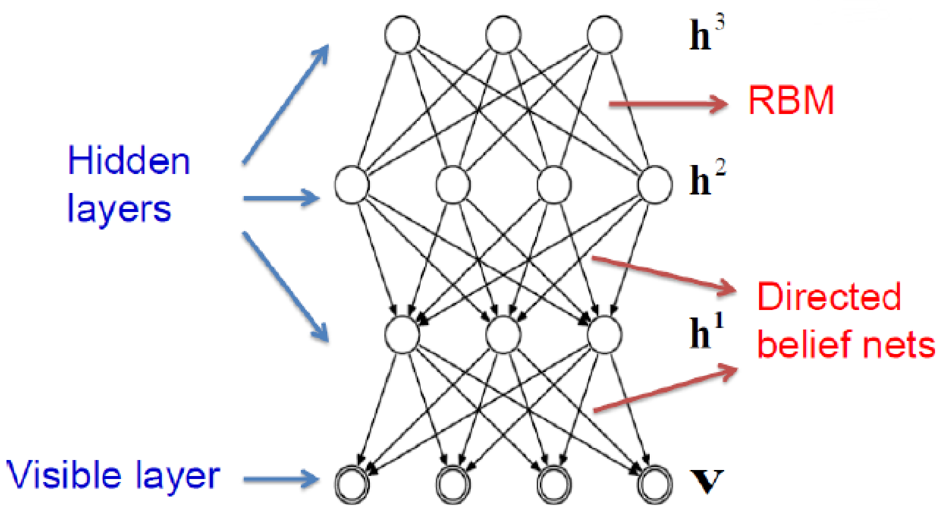
Comments: deep latent structures can be treated as complex nonlinear PCAs of unlabeled data

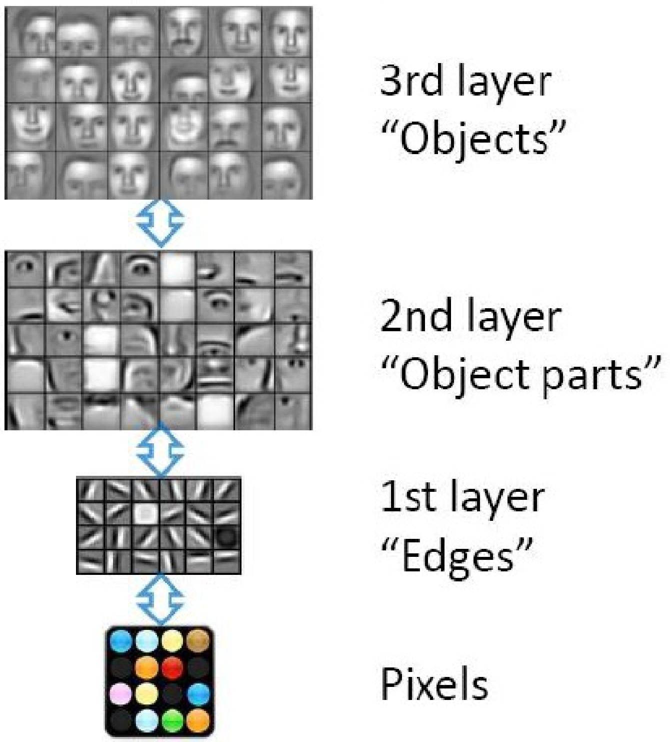


Hinton & Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science* 2006.

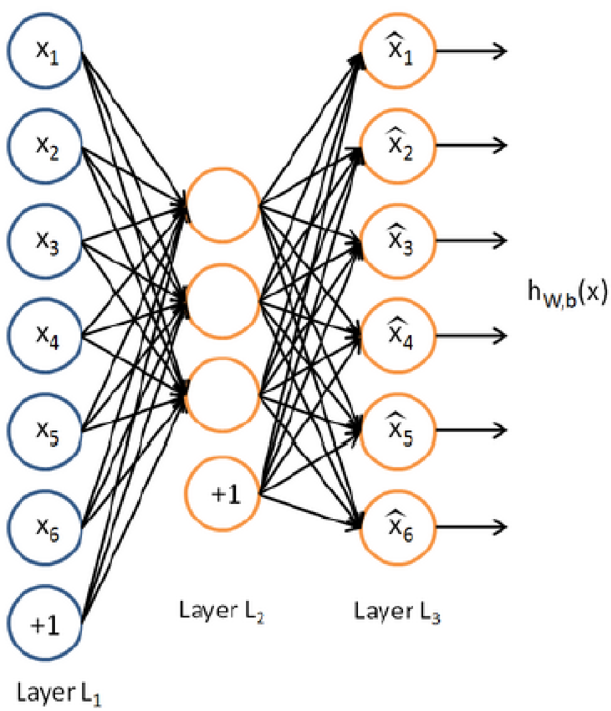
Representative Deep Structure Models

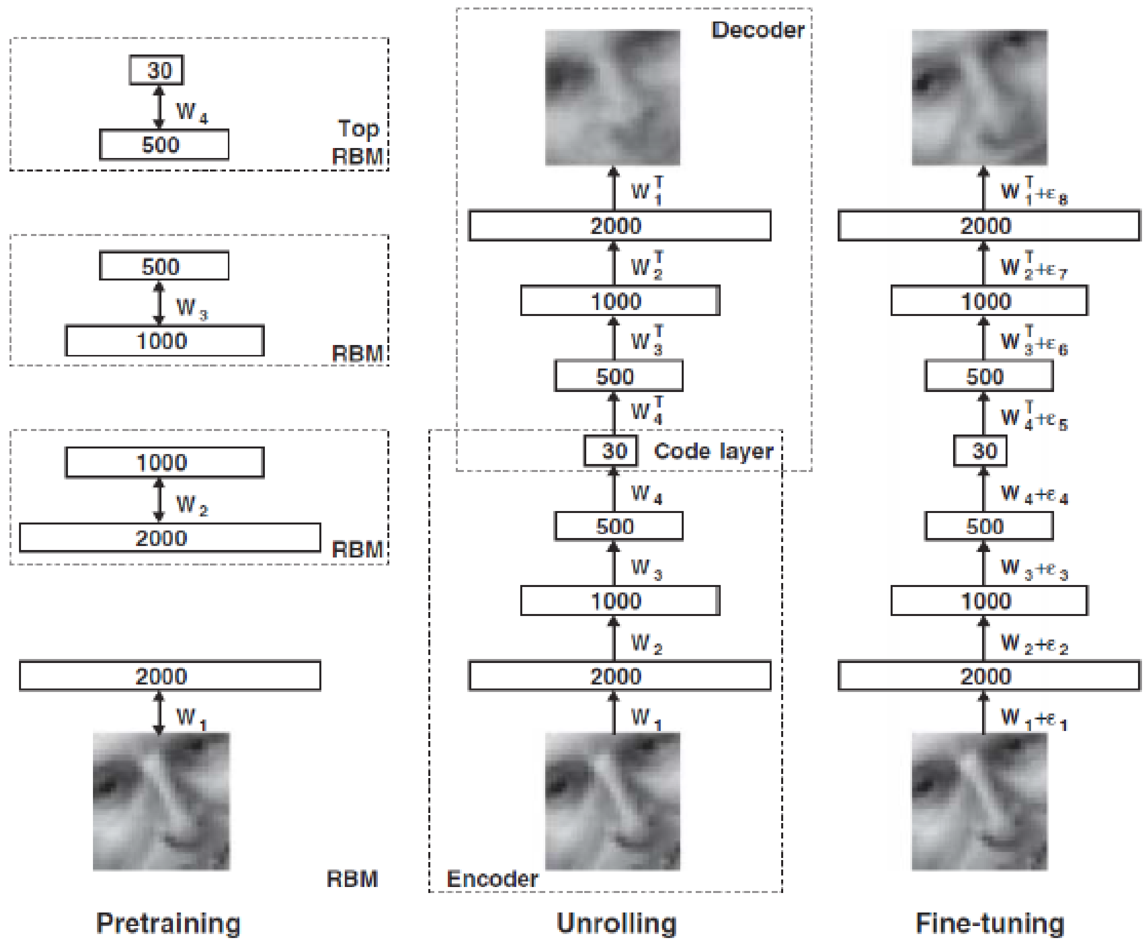
- Stacked auto-encoder
- Deep belief networks & Boltzmann machines
- Deep conventional neural networks





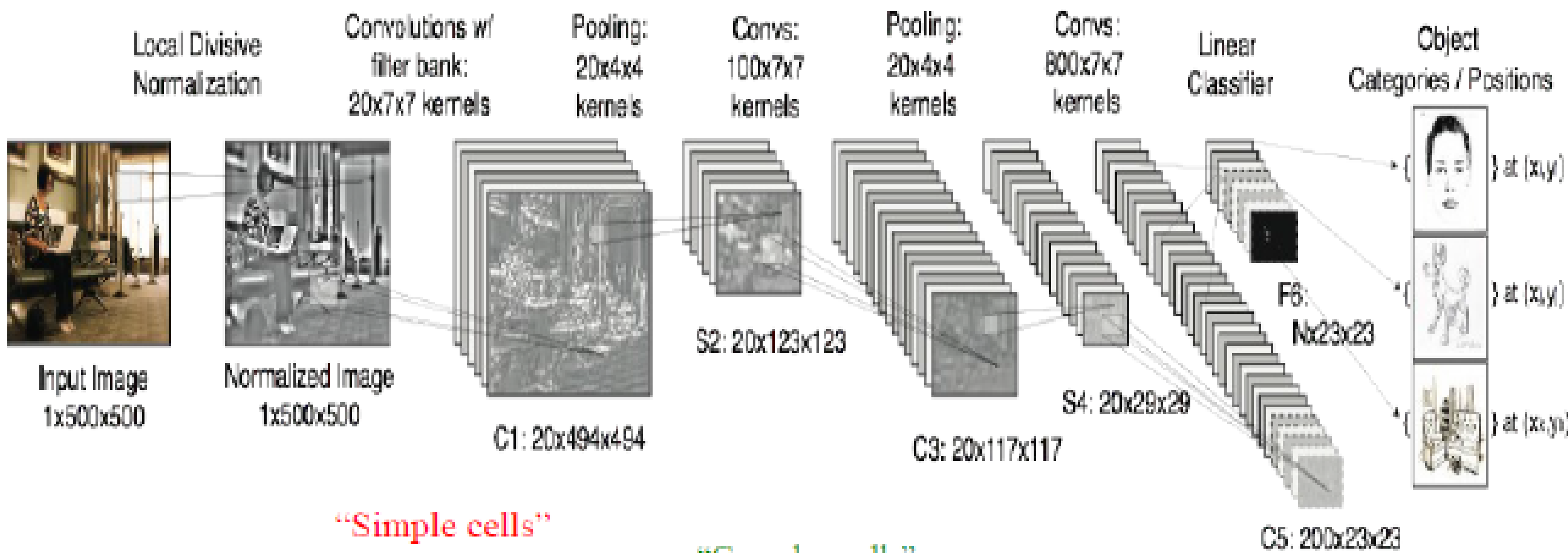
Stacked Auto-Encoders

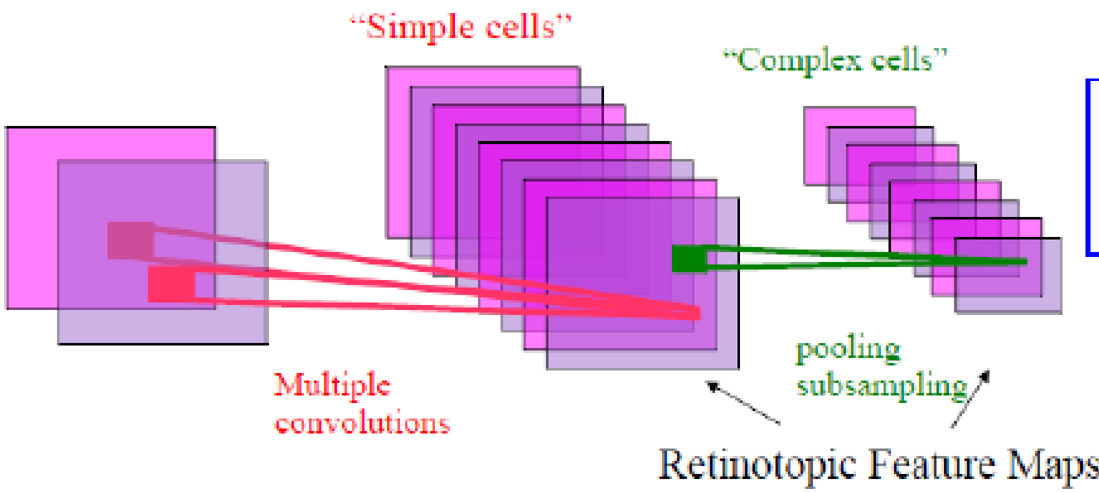




Hinton & Salakhutdinov. *Science* 2006.

Convolutional Neural Networks





- Training is supervised
- With stochastic gradient descent

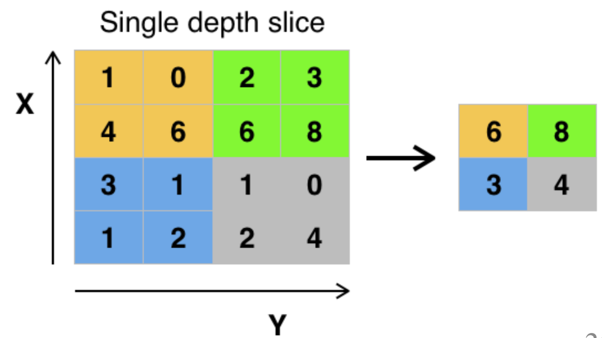
[LeCun et al. 89]
[LeCun et al. 98]

Convolutional Neural Networks

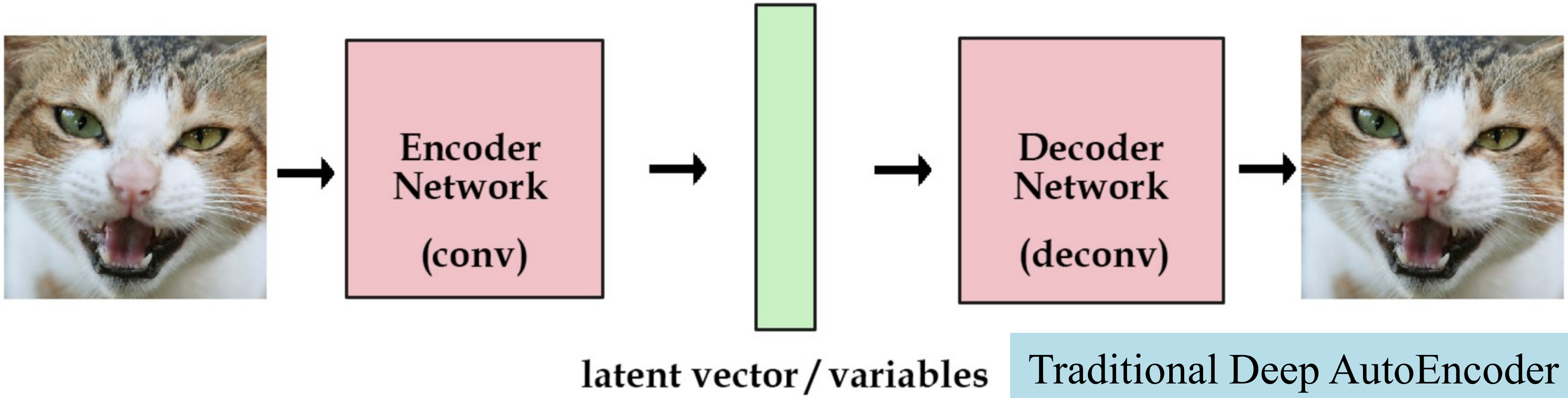
- What is convolution (卷积)?
  - In signaling processing, convolution is equal to multiplying in frequency domain
  - Different convolution matrices can be regarded as different signal filters, such as low-pass filter

- Why pooling?
  - Invariants in images
  - Reduce noises

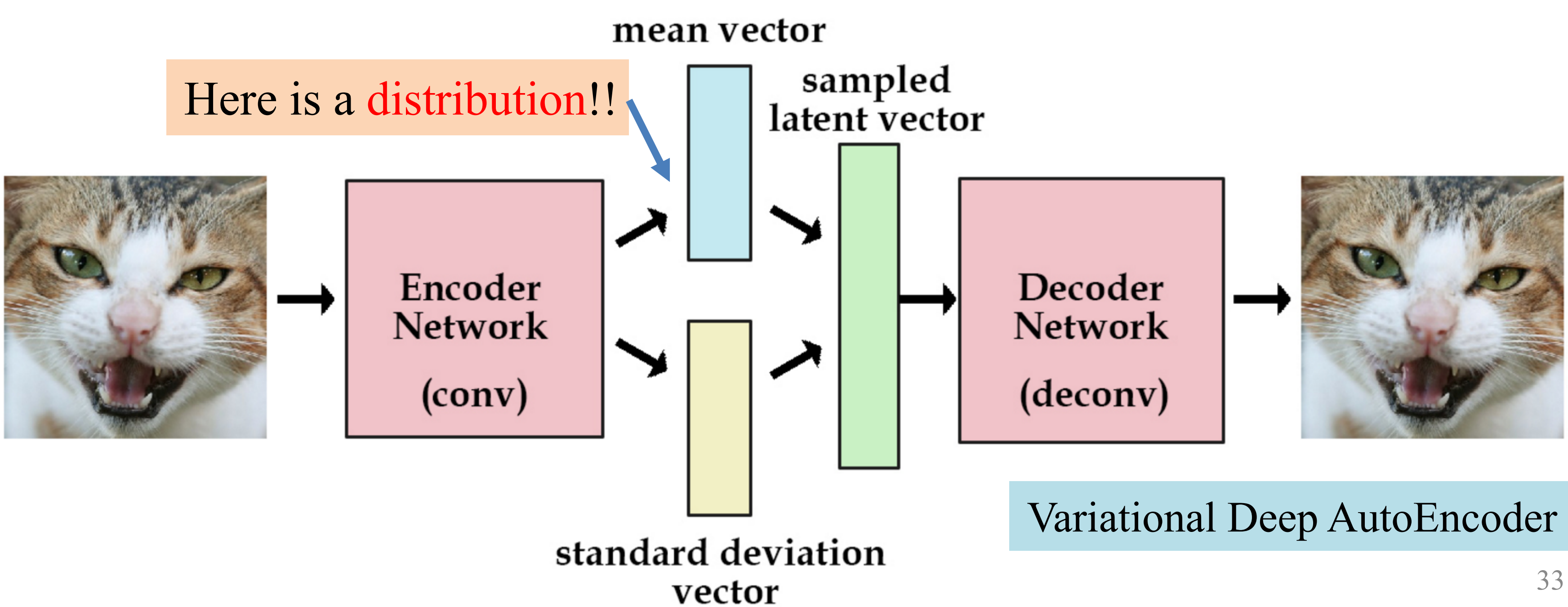
Max pooling



Probabilistic Deep Learning

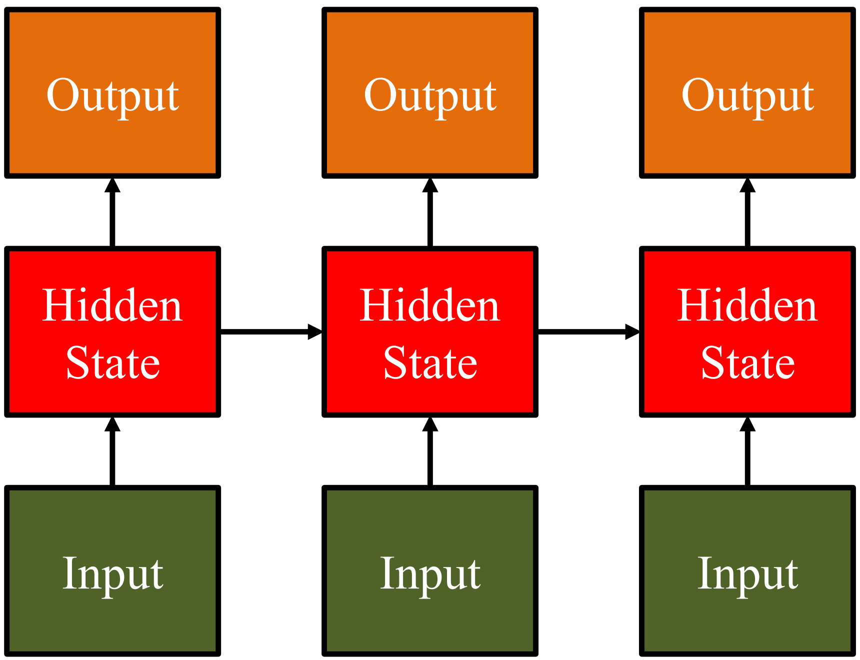


Traditional Deep AutoEncoder



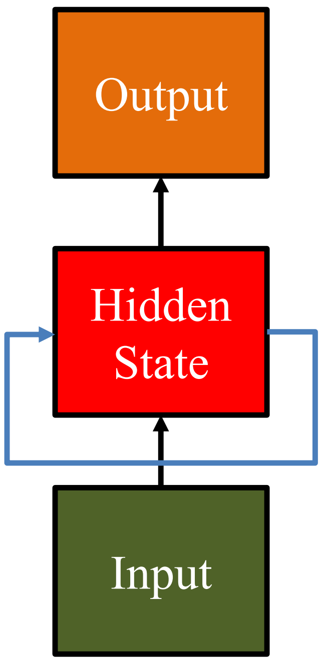
Variational Deep AutoEncoder

Recurrent Neural Networks

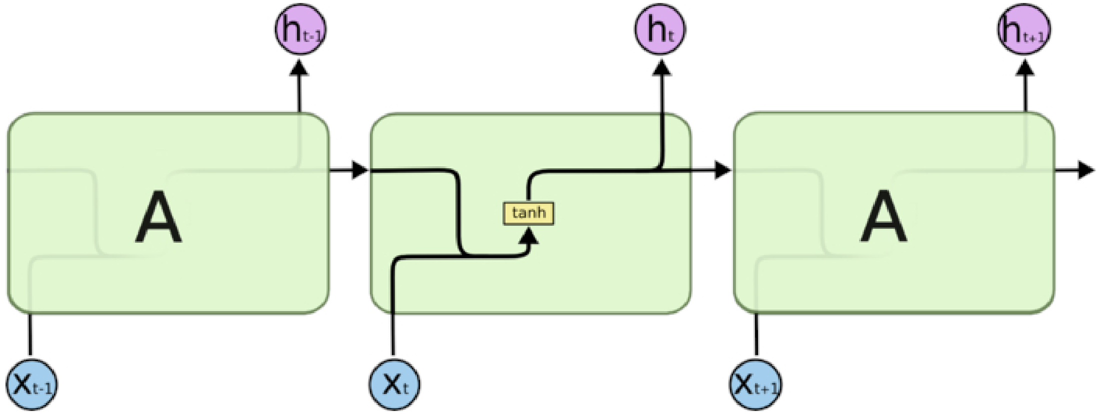


$$y^t = f\left(h^t\right)$$

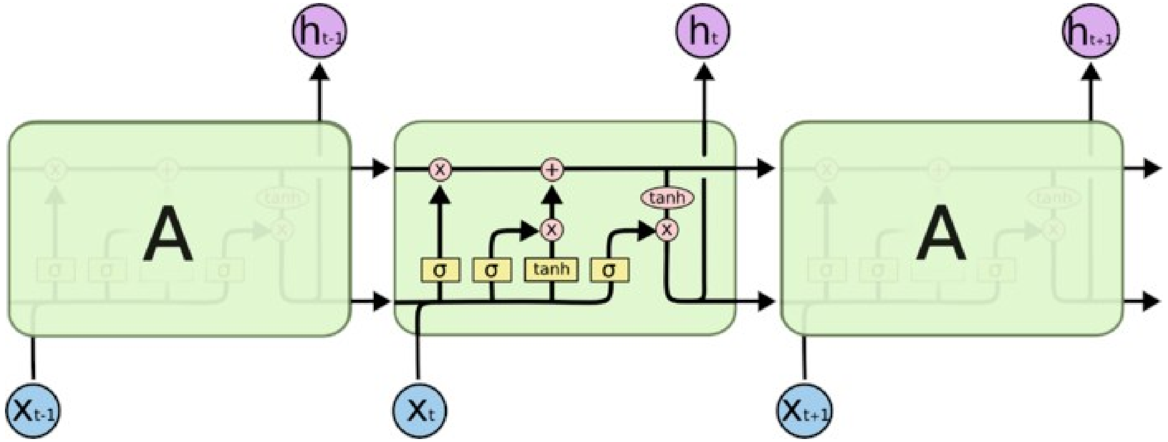
$$h^t = g\left(x^t, h^{t-1}\right)$$



From RNN to LSTM

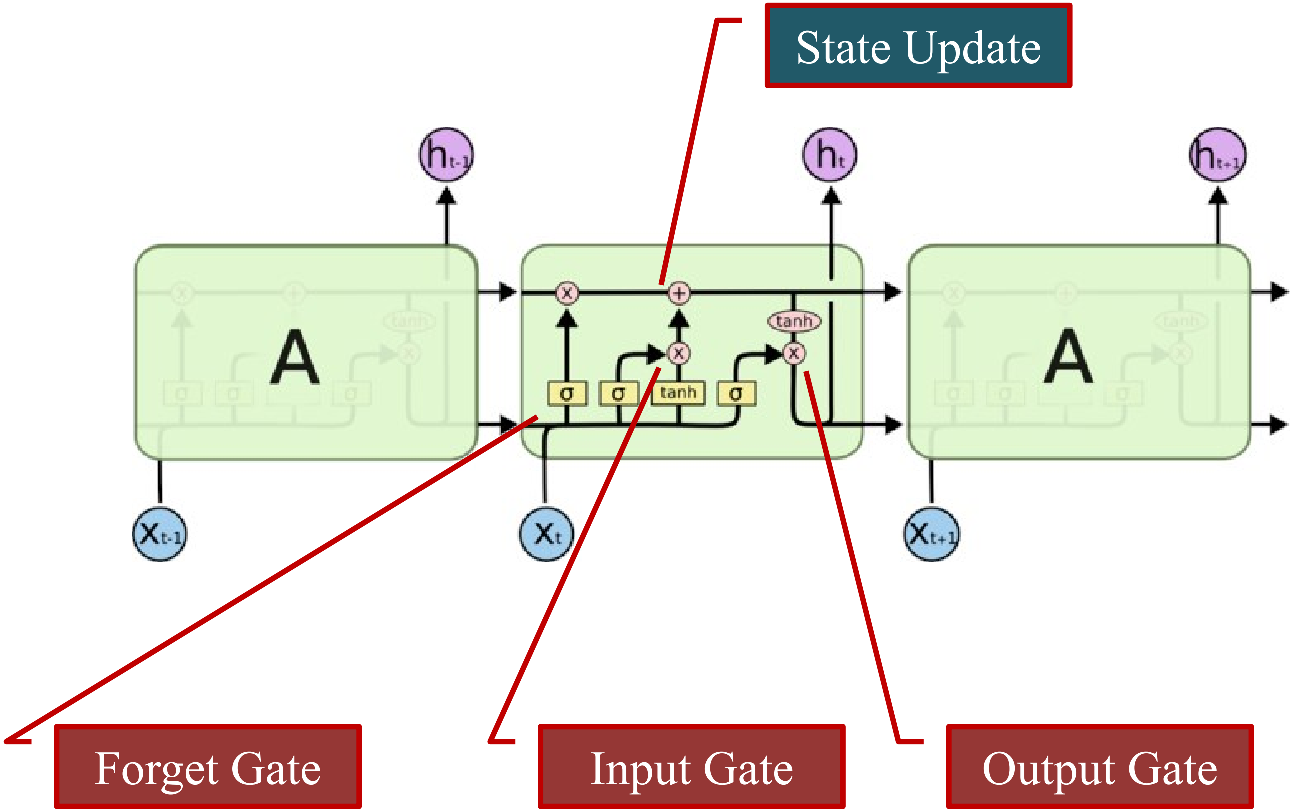


Disadvantages: strong Markov assumption



Sepp Hochreiter & Jürgen Schmidhuber. *Neural Computation* 1997, 9(8):1735-1780.

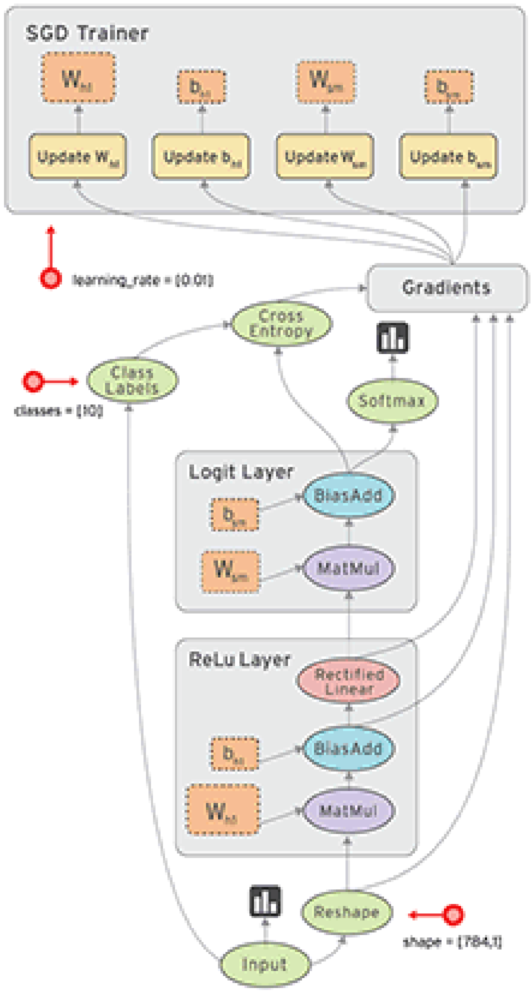
From RNN to LSTM



Resources

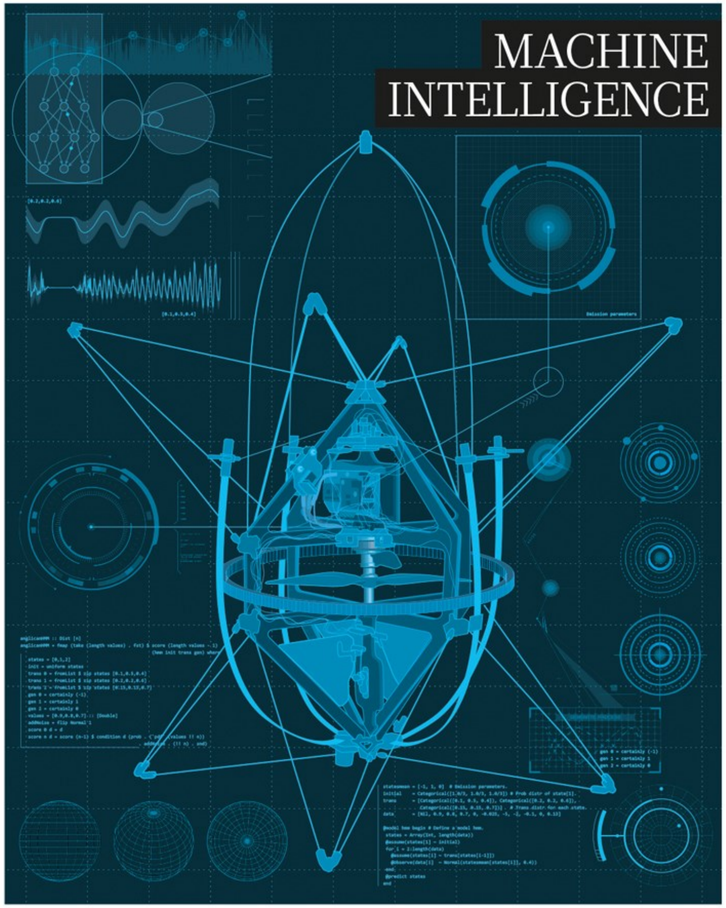
- Literature Survey Reference List
- Geoffrey Hinton’s tutorial
  - “UCL Tutorial on: Deep Belief Nets, 2009”
- Web resources:
  - **Machine intelligence (Nature 521:7553, 435)**
  - TensoFlow: https://www.tensorflow.org/

TensorFlow



- Nodes: mathematical operations
- Edges: data flow
  - The tensors connecting the different operations
- Tensor (张量)
  - In mathematics, tensors are geometric objects that describe linear relations between geometric vectors, scalars, and other tensors.
  - (多重线性映射)

nature INSIGHT



REVIEWS

Model Hierarchical Information

The Parameters are Random Variables

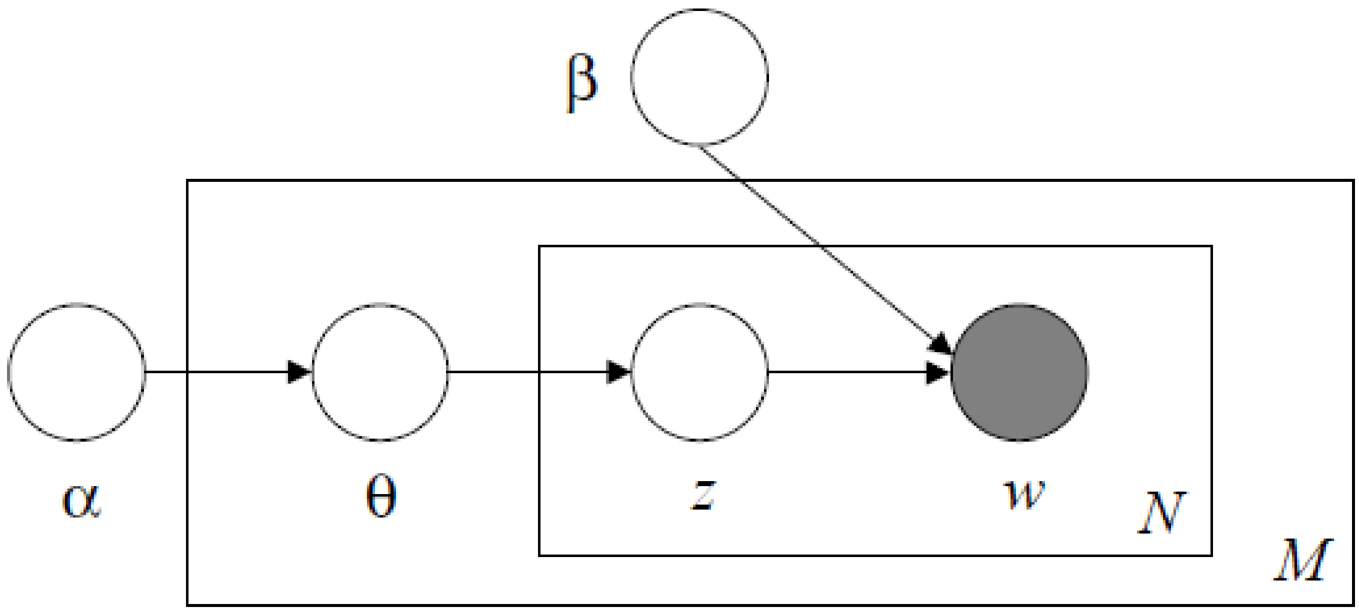
An Example: Topic Models

- We have many ***documents*** downloaded from newspaper. We want to know what are the ***topics*** talked by these documents.

- Each document has one or more ***topics***. One topic has distinct usages of ***words***.

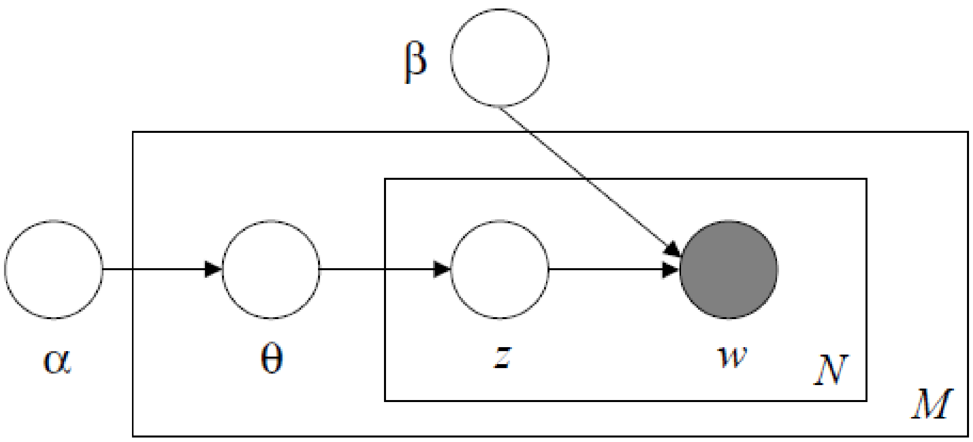
Latent *Dirichlet* Allocation

- Please refer to
  - Blei MN, Ng AY and Jordan MI. Latent *Dirichlet* Allocation. *Journal of Machine Learning Research* 2003, 3:993-1022.



Latent *Dirichlet* Allocation

- Word (w): indicated by a 0/1 vector
- Document: a series of N words
- Corpus: a collection of M documents



$$N \sim Poisson(\lambda)$$

$$\theta \sim Dirichlet(\alpha)$$

for each of the N word

$$z_n \sim Multinomial(\theta)$$

$$w_n \sim P(w_k \mid z = z_n, \beta)$$

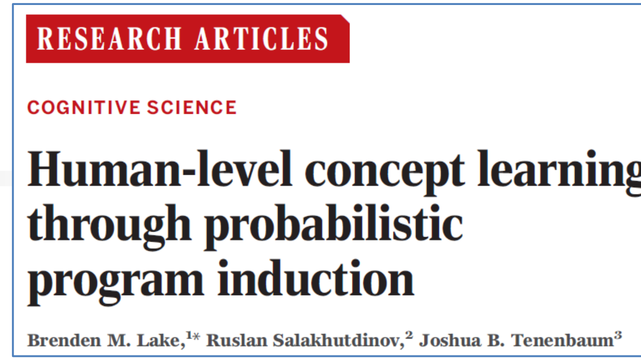
How about transcriptional programs?

- For the assigned project task #1?

- The ***gene expressions*** of tumors are detected by high-throughput techniques. You can infer the ***major transcriptional programs*** and then link them with patients' survivals.

- Each tumor (document) has one or more activated transcriptional programs (topics). Each program has distinct up- or down-regulated ***genes*** (words).

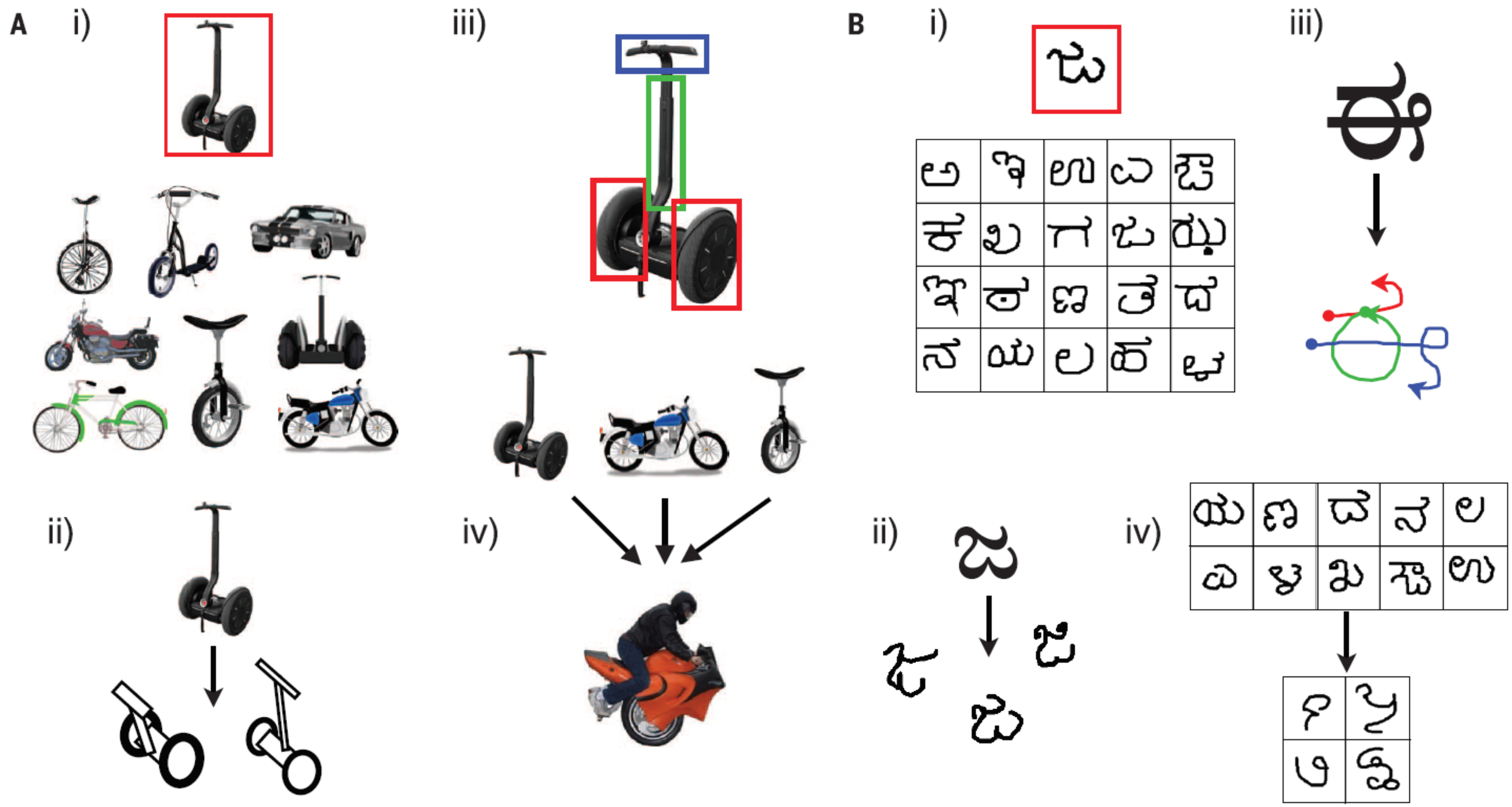
An Example for Writing


RESEARCH ARTICLES

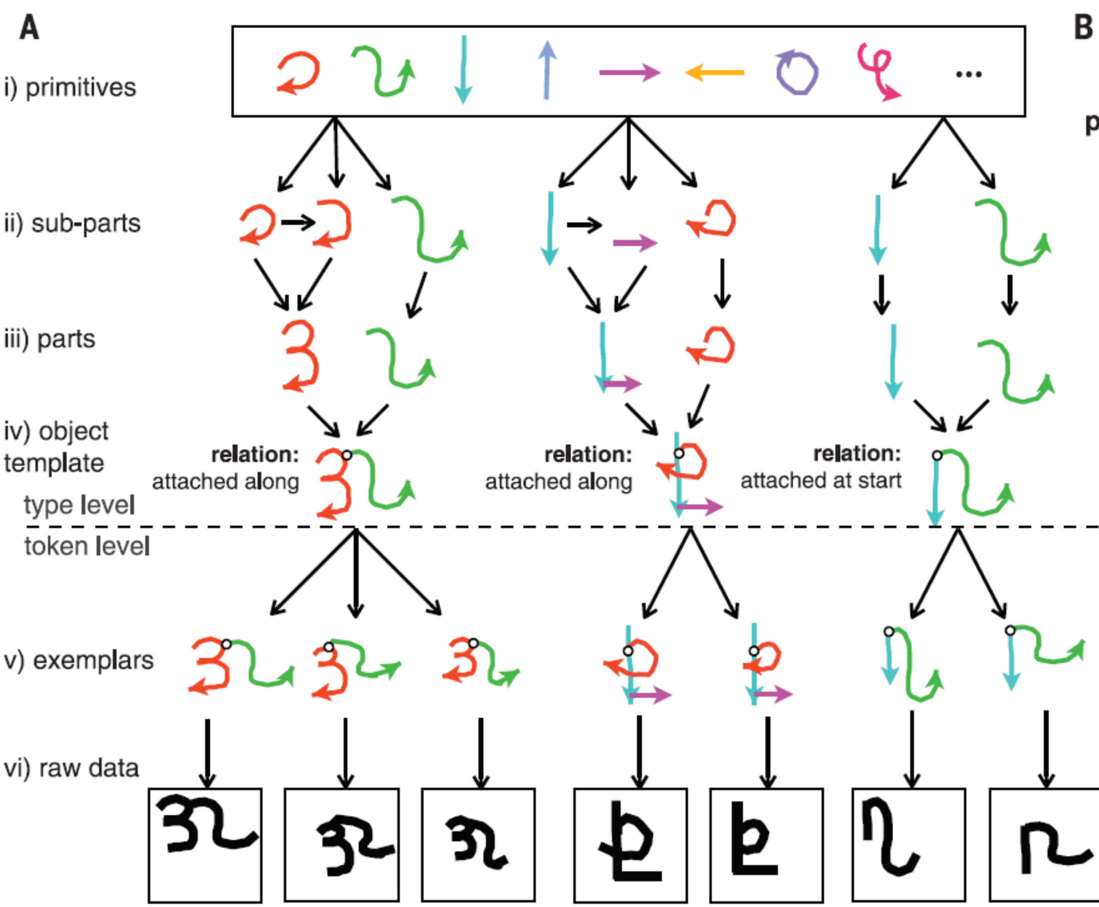
COGNITIVE SCIENCE

Human-level concept learning through probabilistic program induction

Brenden M. Lake,[1*] Ruslan Salakhutdinov,[2] Joshua B. Tenenbaum[3]



An Example for Writing



B

procedure GENERATETYPE
- $\kappa \leftarrow P(\kappa)$ ▷ Sample number of parts
- **for** i = 1 ... κ **do**
  - $n_i \leftarrow P(n_i|\kappa)$ ▷ Sample number of sub-parts
  - **for** j = 1 ... n_i **do**
    - $s_{ij} \leftarrow P(s_{ij}|s_{i(j-1)})$ ▷ Sample sub-part sequence
  - **end for**
  - $R_i \leftarrow P(R_i|S_1, ..., S_{i-1})$ ▷ Sample relation
- **end for**
- $\psi \leftarrow \{\kappa, R, S\}$
- **return** @GENERATETOKEN(ψ) ▷ Return program

procedure GENERATETOKEN(ψ)
- **for** $i = 1 ... \kappa$ **do**
  - $S_i^{(m)} \leftarrow P(S_i^{(m)}|S_i)$ ▷ Add motor variance
  - $L_i^{(m)} \leftarrow P(L_i^{(m)}|R_i, T_1^{(m)}, ..., T_{i-1}^{(m)})$ ▷ Sample part's start location
  - $T_i^{(m)} \leftarrow f(L_i^{(m)}, S_i^{(m)})$ ▷ Compose a part's trajectory
- **end for**
- $A^{(m)} \leftarrow P(A^{(m)})$ ▷ Sample affine transform
- $I^{(m)} \leftarrow P(I^{(m)}|T^{(m)}, A^{(m)})$ ▷ Sample image
- **return** $I^{(m)}$

Representation is an ART!

You need both theories & experiences