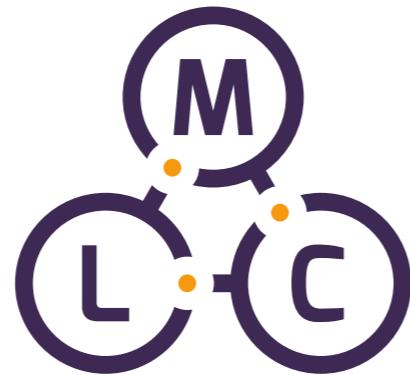


# Natural Language Processing for Raiffeisenbank International

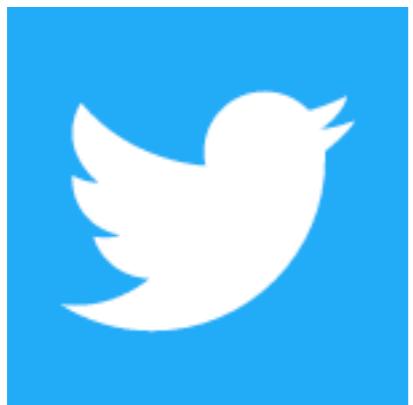
Jiří Materna



Machine  
Learning  
College



@mlcollegecom



@mlcollegecom



# About me

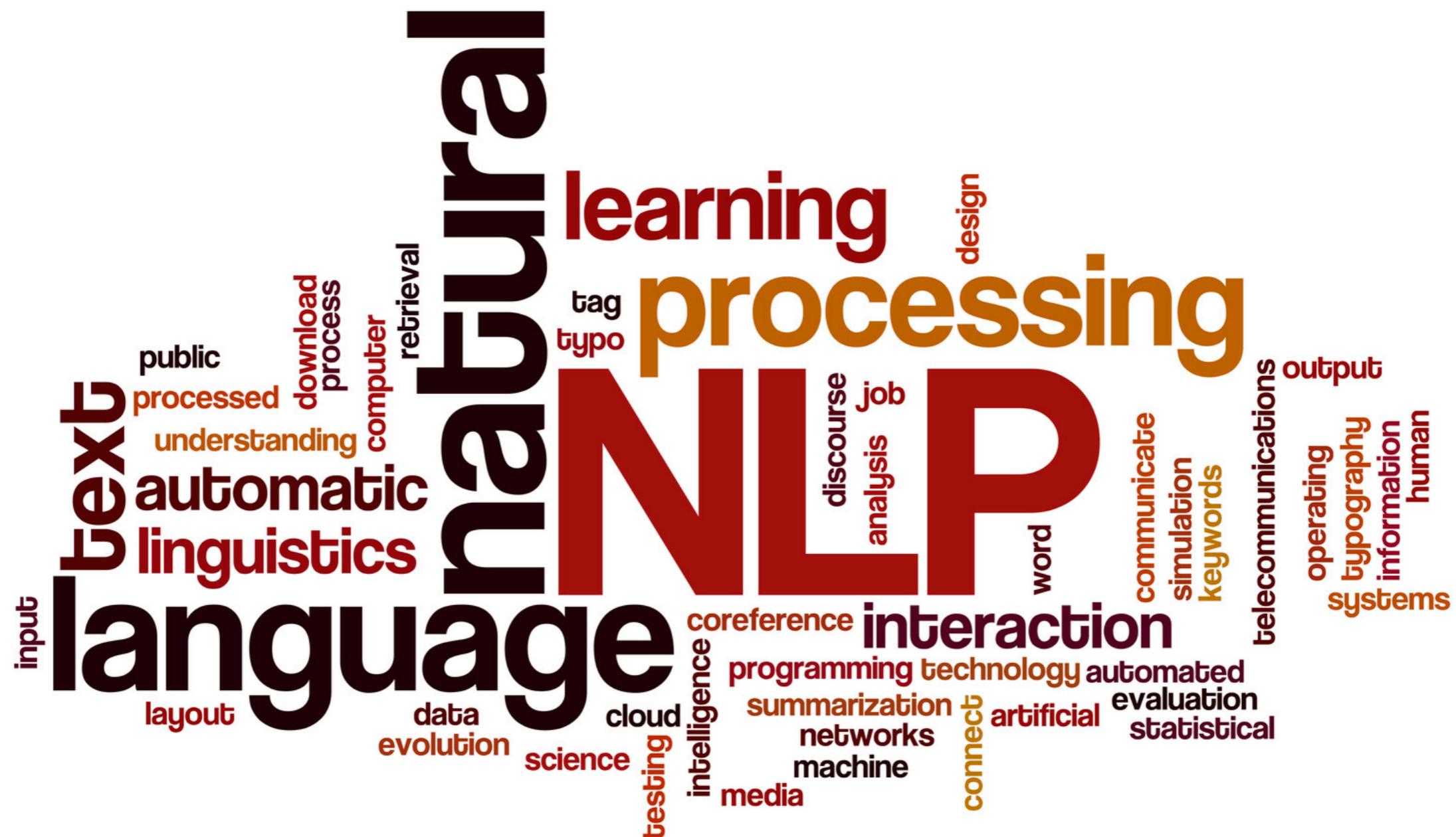
- Ph.D. in Natural Language Processing and Artificial Intelligence at Masaryk University
- 10 years at seznam.cz (last 8 years as Head Of Research)
- Founder and co-organizer of ML Prague
- Mentor at StartupYard
- ML Freelancer and consultant

# Outline

## Day 1

- Introduction to natural language processing
- Computational linguistics
- Text document vectorization
- Practical document classification task
- Language modeling
- Practical tasks on language modeling

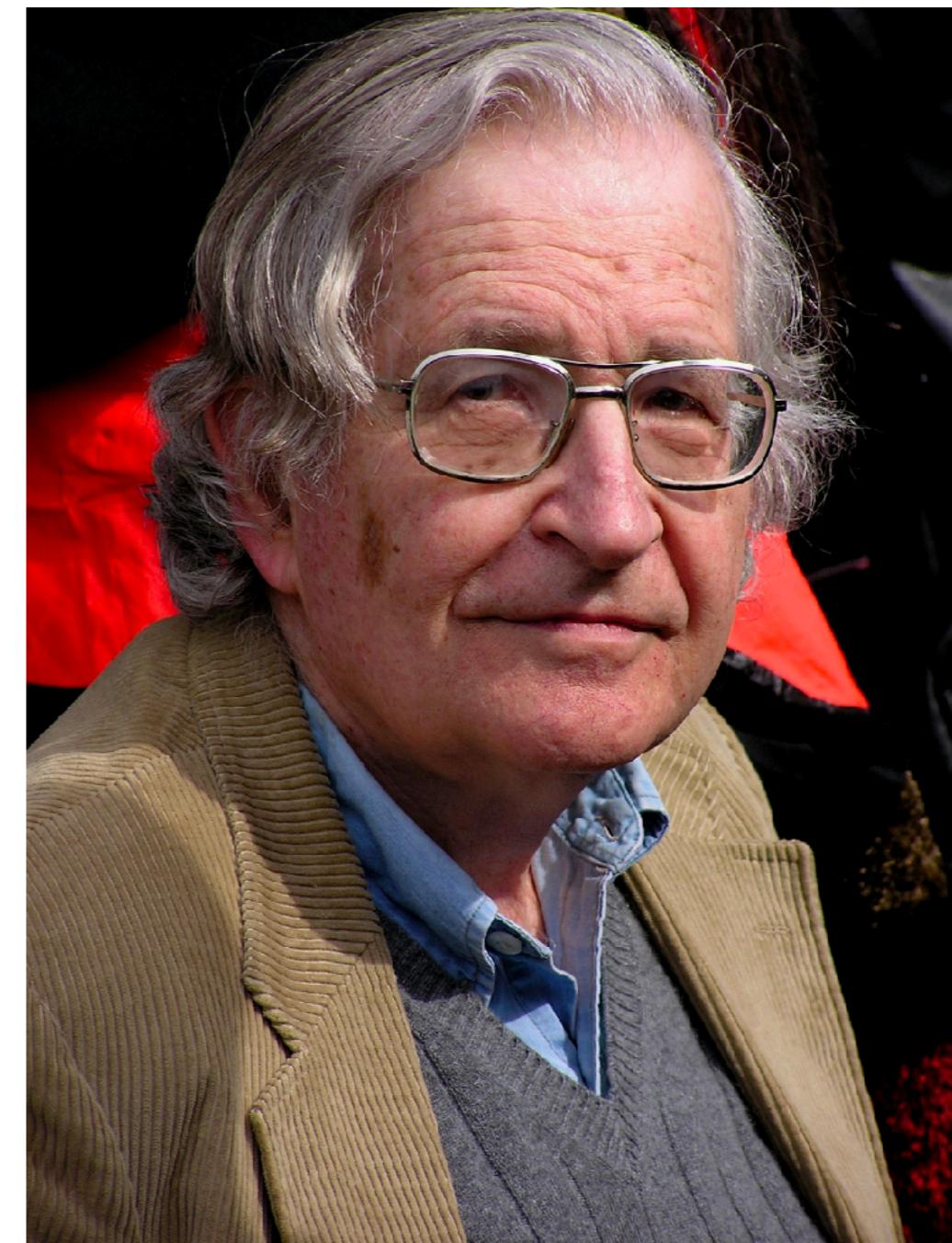
# What is Natural Language Processing?



# Norvig vs. Chomsky



source: <https://www.commarts.com>



source: <https://citaty.net>

# Text corpus



natural language   British National Corpus

Home

Search

Word list

Word sketch

Thesaurus

Sketch diff

Trends

Corpus info

My jobs

User guide ↗

Save

Make subcorpus

View options

KWIC

Sentence

Sort

Left

Right

Node

References

Shuffle

Sample

Query **natural, language** 255 (2.27 per million)

Page  of 13   |

J2K nature of deixis (see Chapter 2 below) in **natural languages**, for sentences like (II) are true or false only  
J2K of the simple but immensely important fact that **natural languages** are primarily designed, so to speak, for use in  
J2K . </p><p> The many facets of deixis are so pervasive in **natural languages**, and so deeply grammaticalized, that it is hard  
J2K the utterance, within the utterance itself. **Natural language** utterances are thus "anchored" directly to  
J2K semantics deals with certain **natural language** expressions. Suppose we identify the semantic  
J2K or self-referring expressions in **natural languages**, as in (12) and, arguably, in (13) (see Chapter 5  
J2K , is perhaps a philosophical red-herring. **Natural languages**, after all, just do have indexicals, and it is  
J2K . Semantics is then not concerned directly with **natural language** at all, but only with the abstract entities  
J2K to leave us with no term for all those aspects of **natural language** significance that are not in any way amenable to  
J2K of the deictic expressions that occur in **natural languages**, and we should now turn to consider linguistic  
J2K in familiar languages. </p><p> Deictic systems in **natural languages** are not arbitrarily organized around the  
J2K . But this has the consequence, as we noted, that **natural languages** will only have a syntax and a pragmatics, and no  
J2K more or less directly on fragments of **natural language** (as initiated by Montague, 1974) would make  
J2K The semanticist who takes the other tack, that **natural language** senses are protean, sloppy and variable, is  
J2K offers a way out, for it allows one to claim that **natural language** expressions do tend to have simple, stable and  
J2K radical differences between logic and **natural language** seem to fade away. We shall explore this below  
J2K on what can be a possible lexical item in **natural languages** . </p><p> Finally, the principles that generate  
J0V is meant any single document, or any stretch of **natural language** regarded as a self-contained unit for  
J53 recognition and those that can understand **natural languages**, such as English, are known by the collective  
HRK through a dialogue, which approaches a **natural language** dialogue, or via a menu. In figure 6.2, the users

Page  of 13   |

# Token & tokenization

This is a non-trivial English sentence: Ludolph's number is approx. 3.14.

**Python library:** <http://www.nltk.org/>

# Stemming & lemmatization

Original	Stemming	Lemmatization
compensation	compens	compensation
compensations	compens	compensation
mouse	mous	mouse
mice	mice	mouse

# Stemming & lemmatization

**English:**

<https://tartarus.org/martin/PorterStemmer/>

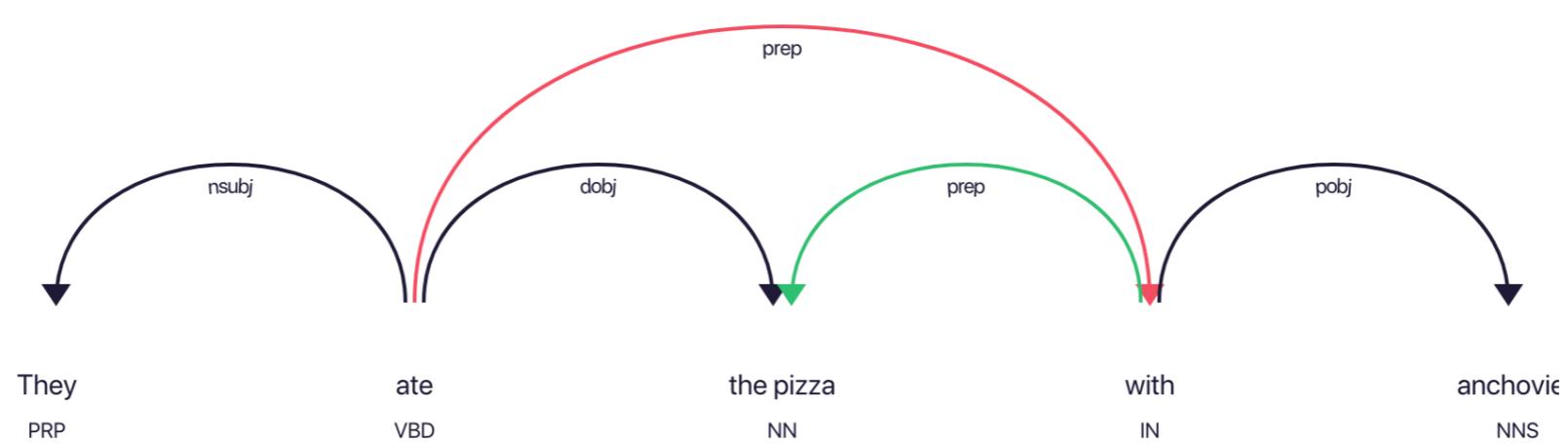
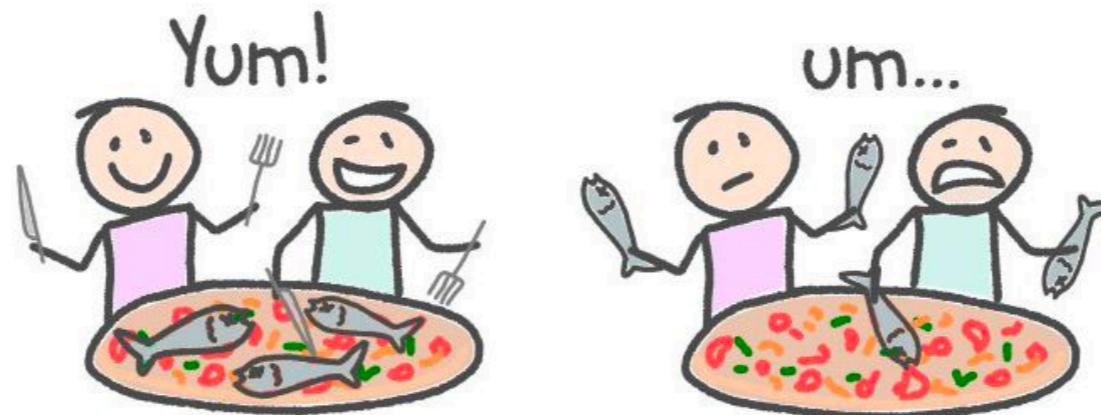
<http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/>

**Czech:**

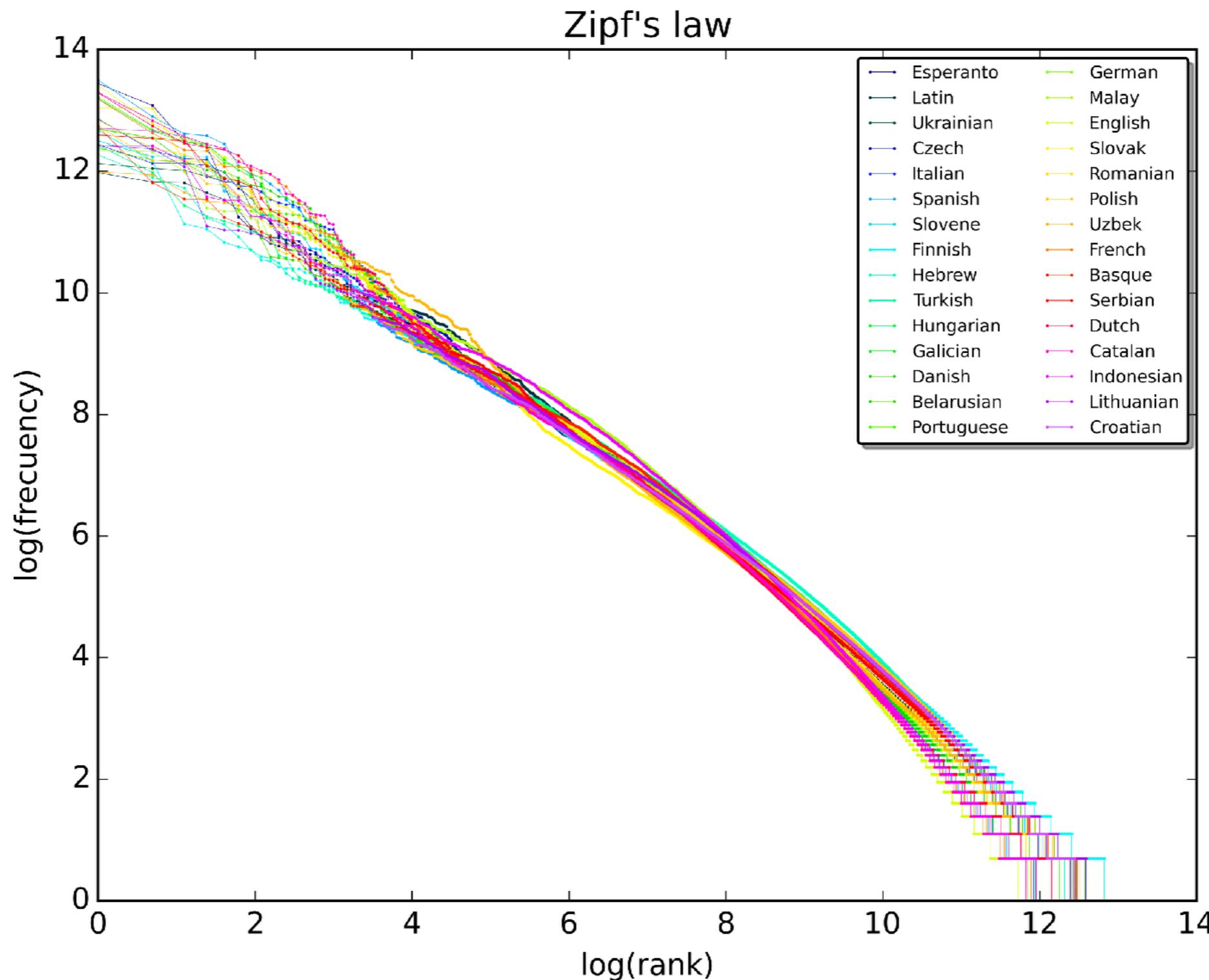
<http://ufal.mff.cuni.cz/morphodita>

# Parsing

They ate the pizza with anchovies



# Zipf's law & long tail



# Publicly available corpora

**British National Corpus:** <http://www.natcorp.ox.ac.uk/>

**Common Crawl:** <http://commoncrawl.org/the-data/get-started/>

**Wikipedia:** <https://dumps.wikimedia.org/>

# Feature extraction for NLP

1. *the man walked the dog*
2. *the man took the dog to the park*
3. *the dog went to the park*

[dog, man, park, the, to, took, walked, went]

1. [1, 1, 0, 1, 0, 0, 1, 0]
2. [1, 1, 1, 1, 1, 1, 0, 0]
3. [1, 0, 1, 1, 1, 0, 0, 1]

$$\text{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

1. [1, 1, 0, 2, 0, 0, 1, 0]
2. [1, 1, 1, 3, 1, 1, 0, 0]
3. [1, 0, 1, 2, 1, 0, 0, 1]

1. [0, 0.18, 0, 0, 0, 0, 0.48, 0]
2. [0, 0.18, 0.18, 0, 0.18, 0.48, 0, 0]
3. [0, 0, 0.18, 0, 0.18, 0, 0, 0.48]

— . . .

# NLP Introduction task

**01-text-classification-introduction.ipynb**

# Language models

- spell checking
- speech recognition
- machine translation
- ...

# n-gram models

$$P(w_1, w_2, \dots, w_n) = P(w_1)P(w_2|w_1)\dots P(w_n|w_1, \dots, w_{n-1})$$

$$= \prod_i P(w_i|w_1, w_2 \dots w_{i-1})$$

$$\approx \prod_i P(w_i|w_{i-k}, w_{i-k-1} \dots w_{i-1})$$

$$P(w_i|w_{i-k}, w_{i-k-1} \dots w_{i-1}) = \frac{\text{count}(w_{i-k}, w_{i-k-1} \dots w_{i-1}, w_i)}{\text{count}(w_{i-k}, w_{i-k-1} \dots w_{i-1})}$$

# Language model smoothing

- Laplace smoothing (plus one)

$$P(w_i|w_{i-1}) = \frac{\text{count}(w_{i-1}, w_i) + 1}{\text{count}(w_{i-1}) + V}$$

- interpolation
- Good-Turing
- Witten-Bell
- ...

# Perplexity

$$PP(W) = P(w_1, w_2, \dots, w_N)^{-\frac{1}{N}}$$

$$= \sqrt[N]{\frac{1}{P(w_1, w_2, \dots, w_N)}}$$

$$= \sqrt[N]{\prod_{i=1}^N \frac{1}{P(w_i | w_1, \dots, w_{i-1})}}$$

$$= 2^{-\frac{1}{N} \sum_{i=1}^N \log_2 P(w_i | w_1, \dots, w_{i-1})}$$

# Language detection using language models

**02-Language-detection-assignment.ipynb**

# Travel agency review classification

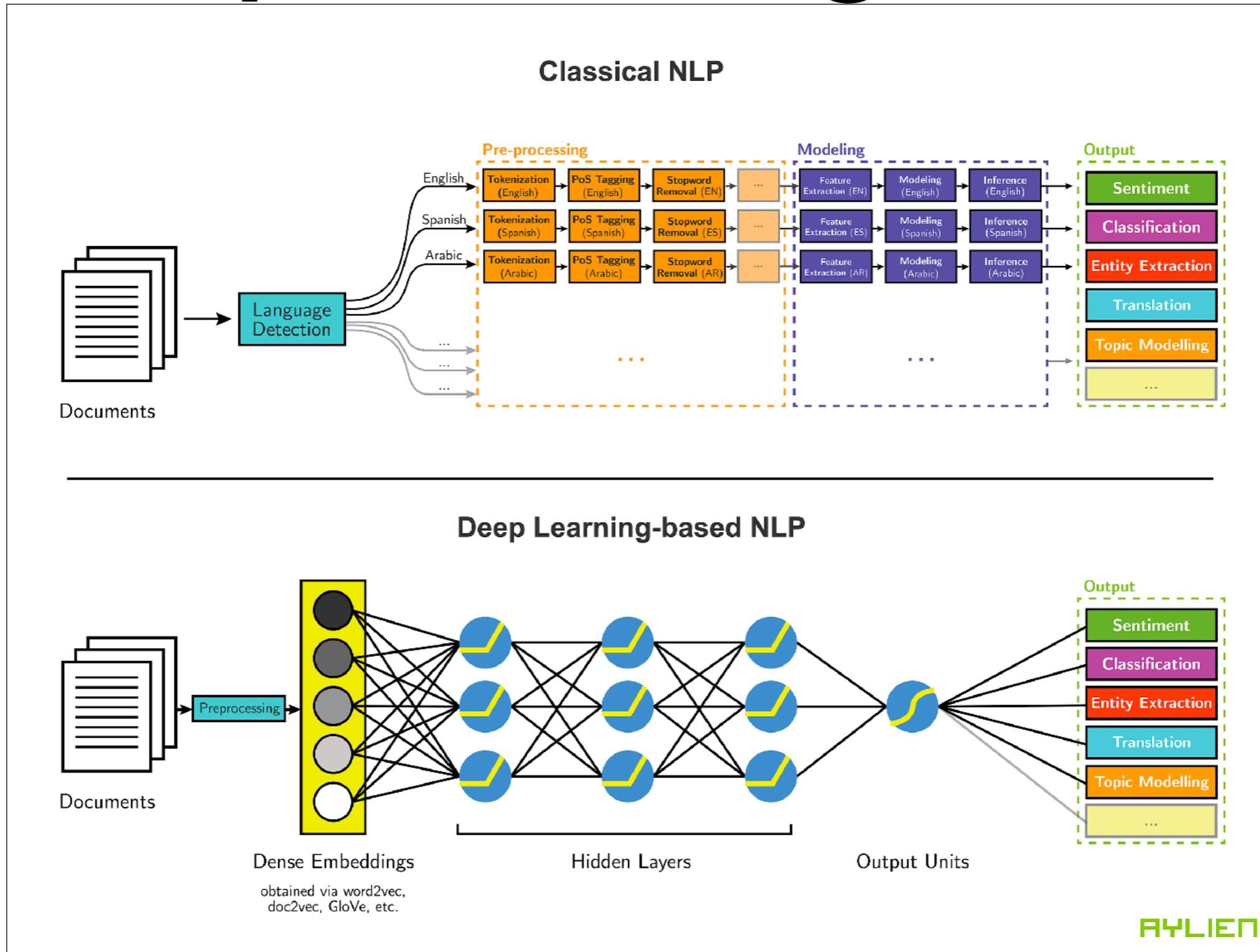
**03-Review-classification-assignment.ipynb**

# Outline

## Day 2

- Preprocessing for deep learning in NLP
- Recurrent neural networks
- Word embeddings and word2vec
- The Skip-gram model
- Experiments with word2vec
- Text classification with word embeddings

# Deep Learning in NLP



# Encoding and Unicode

## ASCII

H e l l o

48 65 6c 6c 6f

## Unicode

H e l l o ☺

00000048 00000065 0000006c 0000006c 0000006f 0000263a

# Encoding and Unicode

UTF-8

H e l l o ☺

48 65 6c 6c 6f e298ba

UTF-16

H e l l o ☺

0048 0065 006c 006c 006f 263a

# Unicode normalization

**NFD** (Normalization Form Canonical Decomposition)

**NFC** (Normalization Form Canonical Composition)

**NFKD** (Normalization Form Compatibility Decomposition)

**NFKC** (Normalization Form Compatibility Composition)

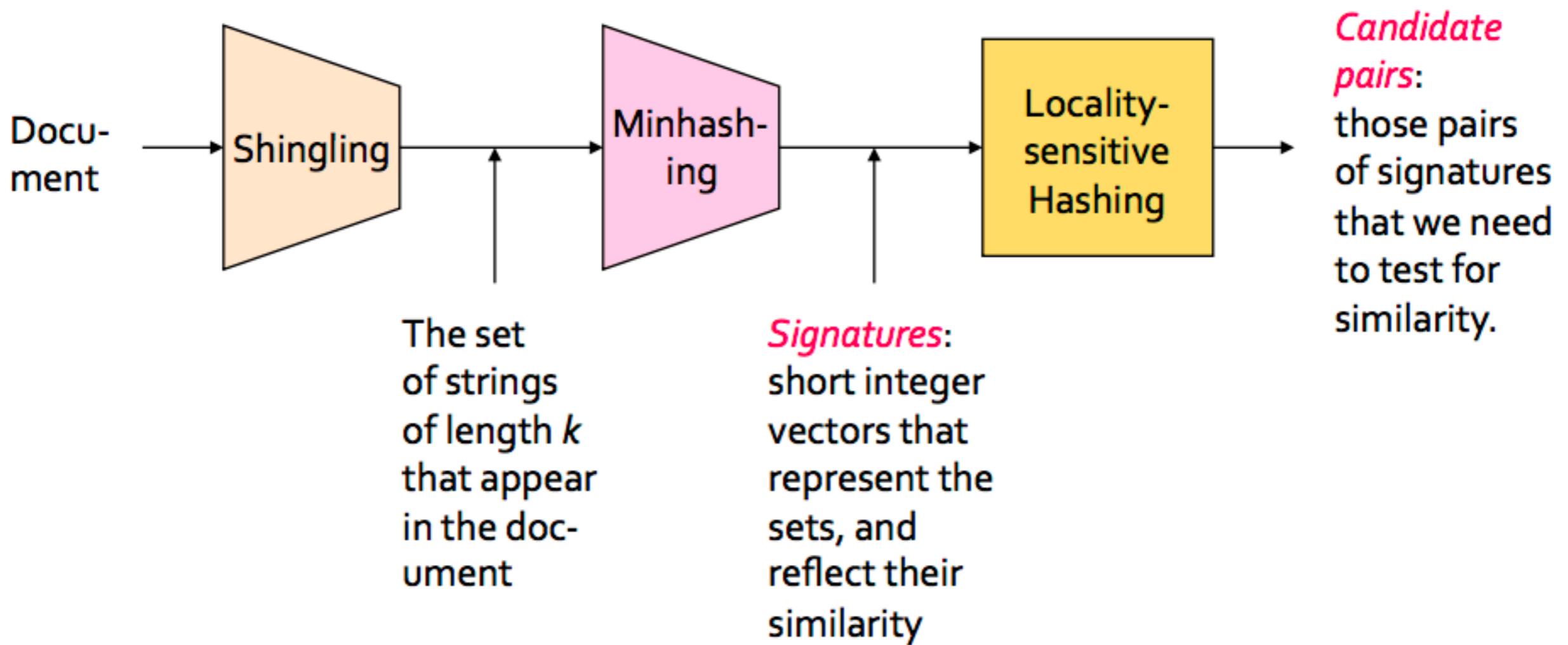
Source	NFD	NFC	NFKD	NFKC
<b>fi</b> FB01	<b>fi</b> FB01	<b>fi</b> FB01	<b>f i</b> 0066 0069	<b>f i</b> 0066 0069
<b>2<sup>5</sup></b> 0032 2075	<b>2<sup>5</sup></b> 0032 2075	<b>2<sup>5</sup></b> 0032 2075	<b>2<sup>5</sup></b> 0032 0035	<b>2<sup>5</sup></b> 0032 0035
<b>ſ</b> 1E9B 0323	<b>f ̧ ̧</b> 017F 0323 0307	<b>ſ ̧</b> 1E9B 0323	<b>s ̧ ̧</b> 0073 0323 0307	<b>§</b> 1E69

# Unicode normalization in Python 3

```
>>> aa = b'\xc4\x81'.decode('utf8')
>>> bb = b'a\xcc\x84'.decode('utf8')
>>> aa
'\u00e1'
>>> bb
'\u00e1'
>>> aa == bb
False
>>> import unicodedata as ud
>>> aa == ud.normalize('NFC',bb)
True
```

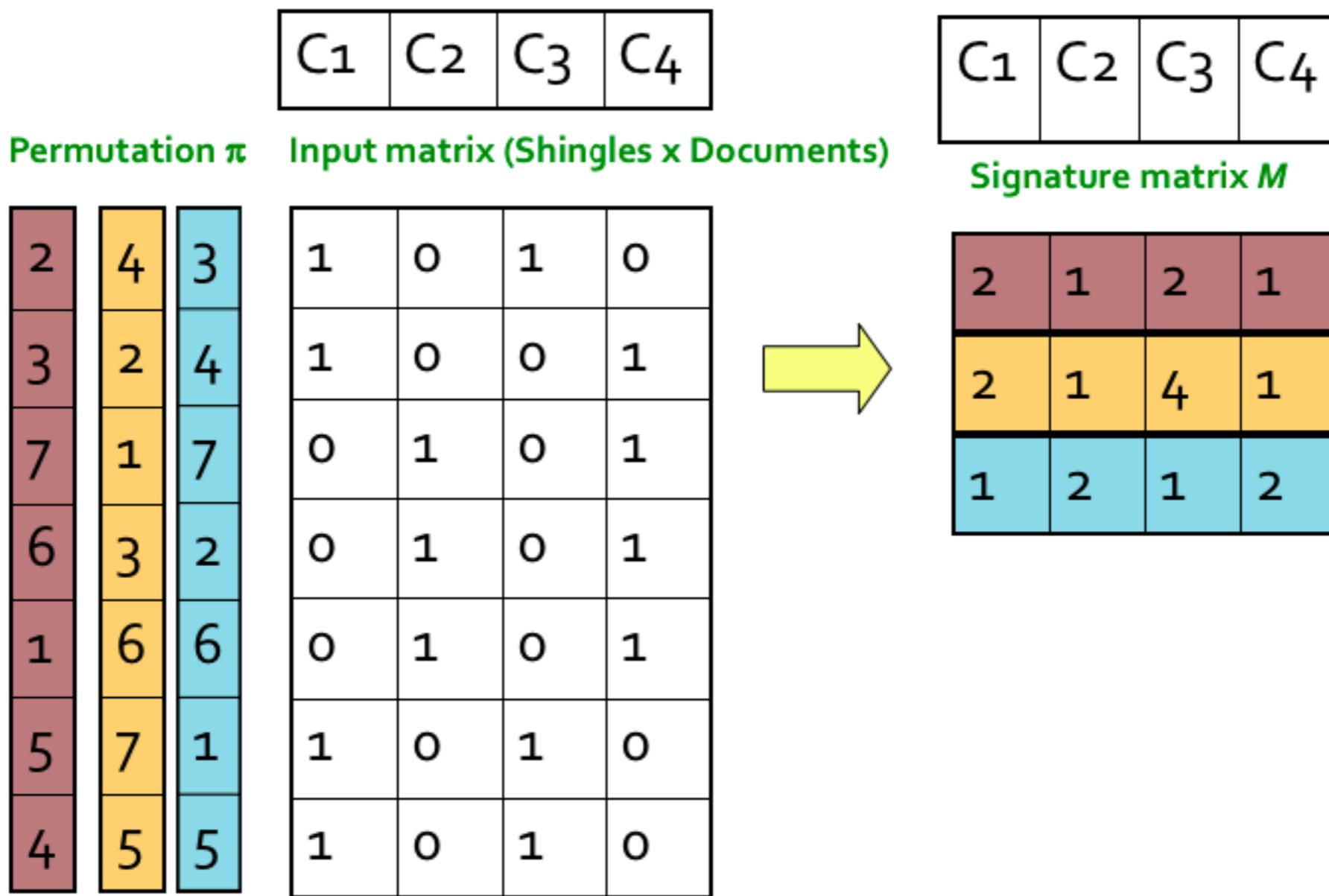
# Near deduplication

## Locality-sensitive hashing



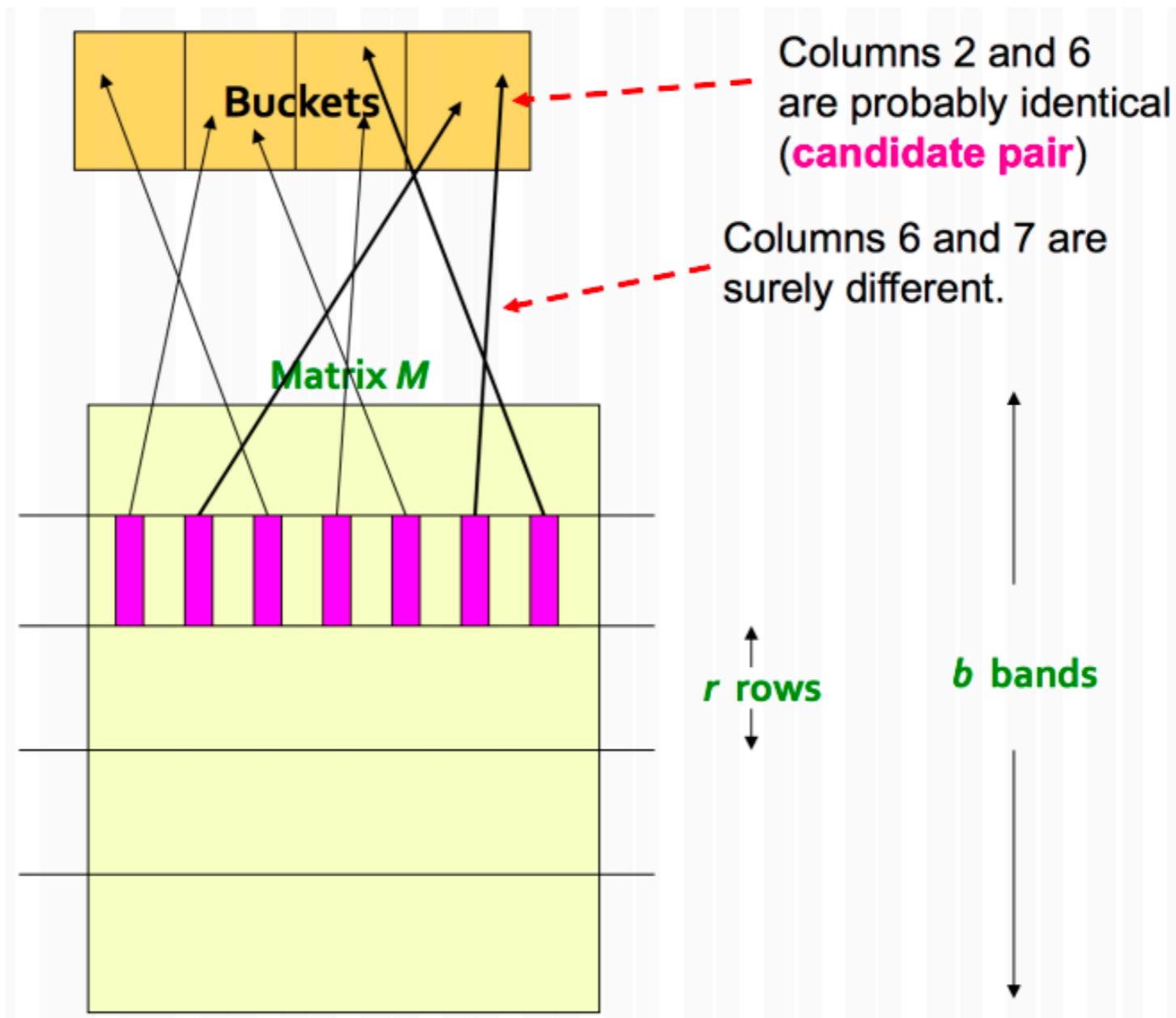
# Near deduplication

## MinHashing signatures

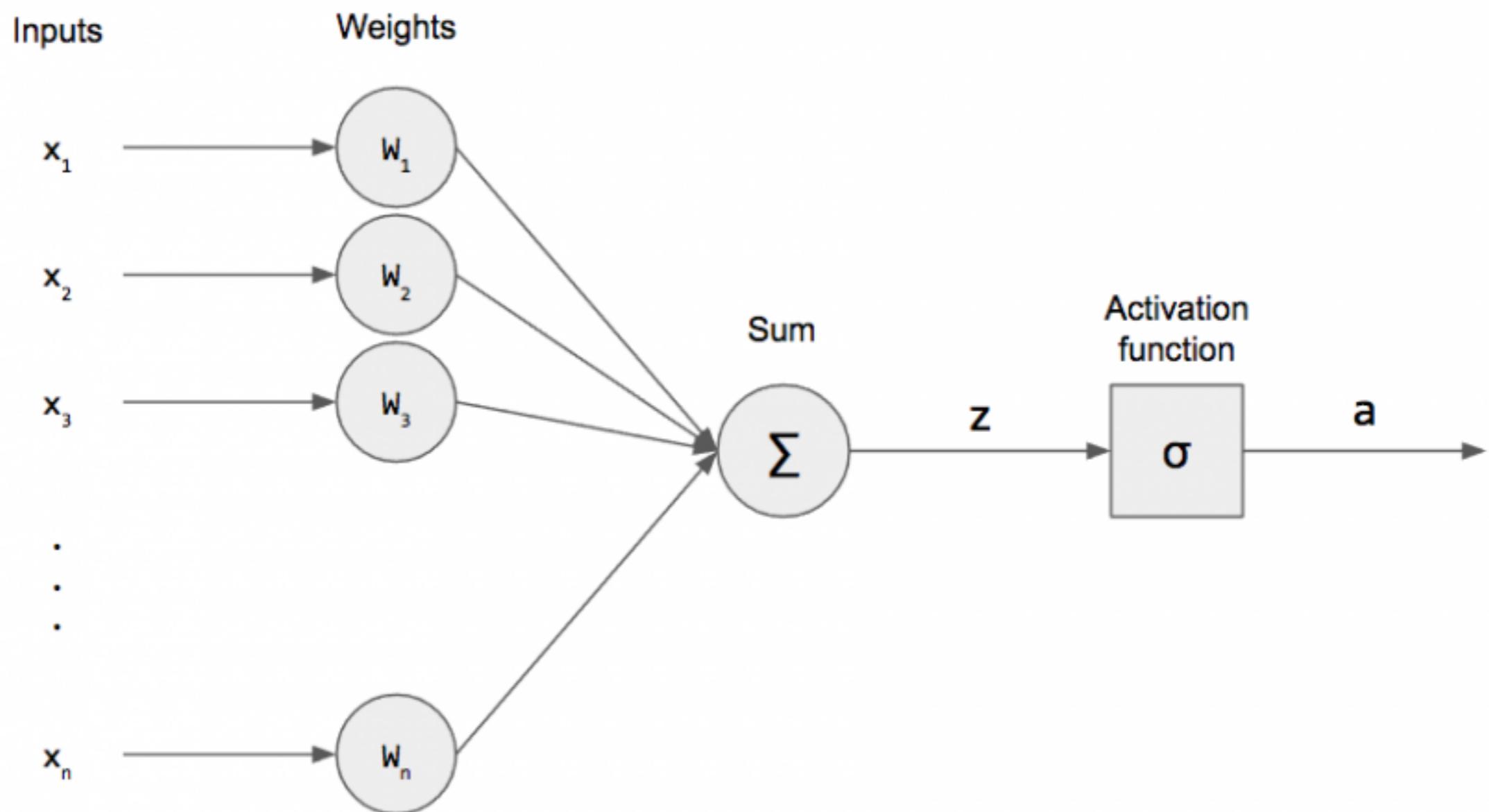


# Near deduplication

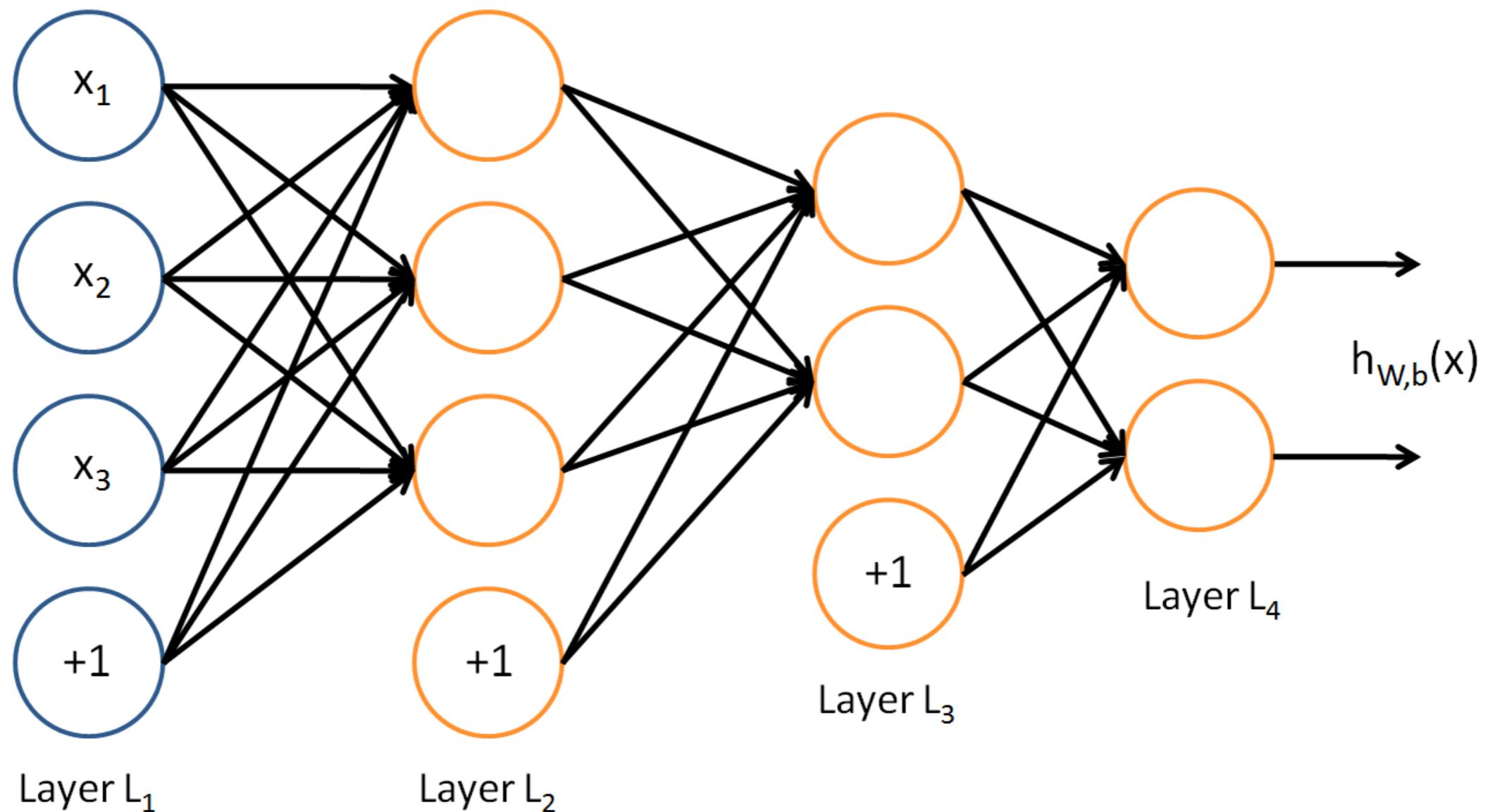
## Locality-sensitive hashing



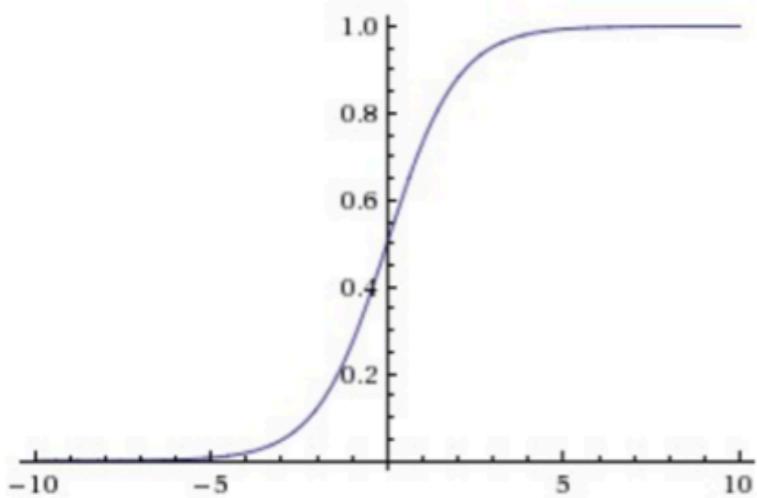
# Perceptron



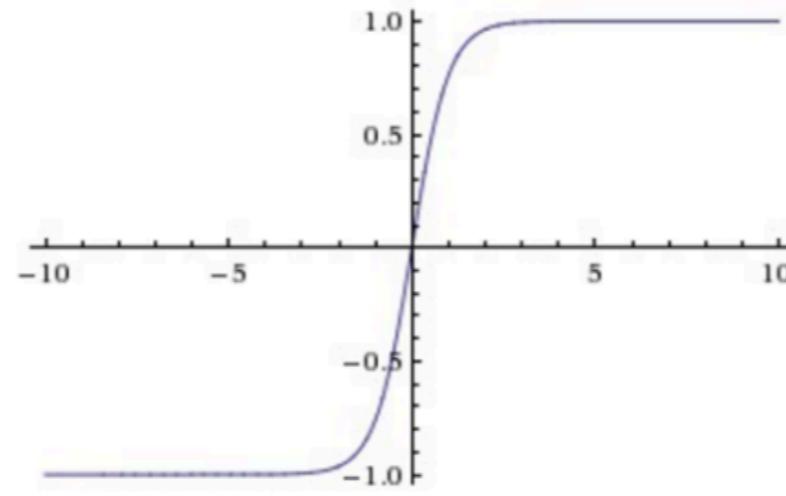
# Multilayer Neural Networks



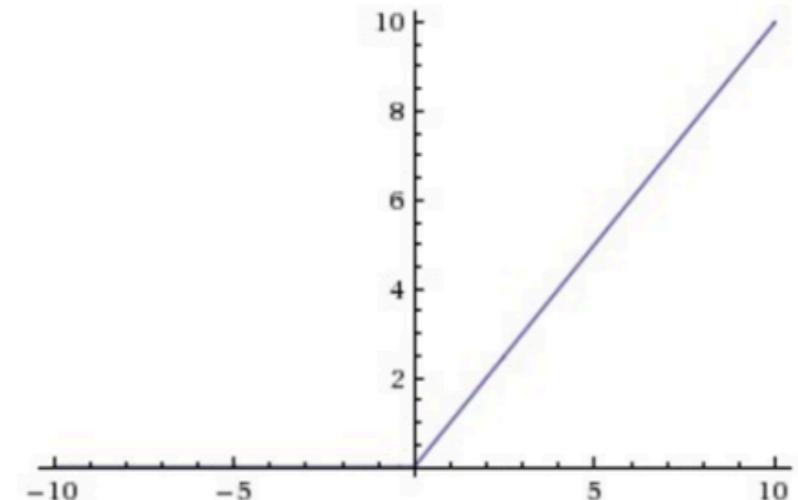
# Activation functions



Sigmoid



tanh

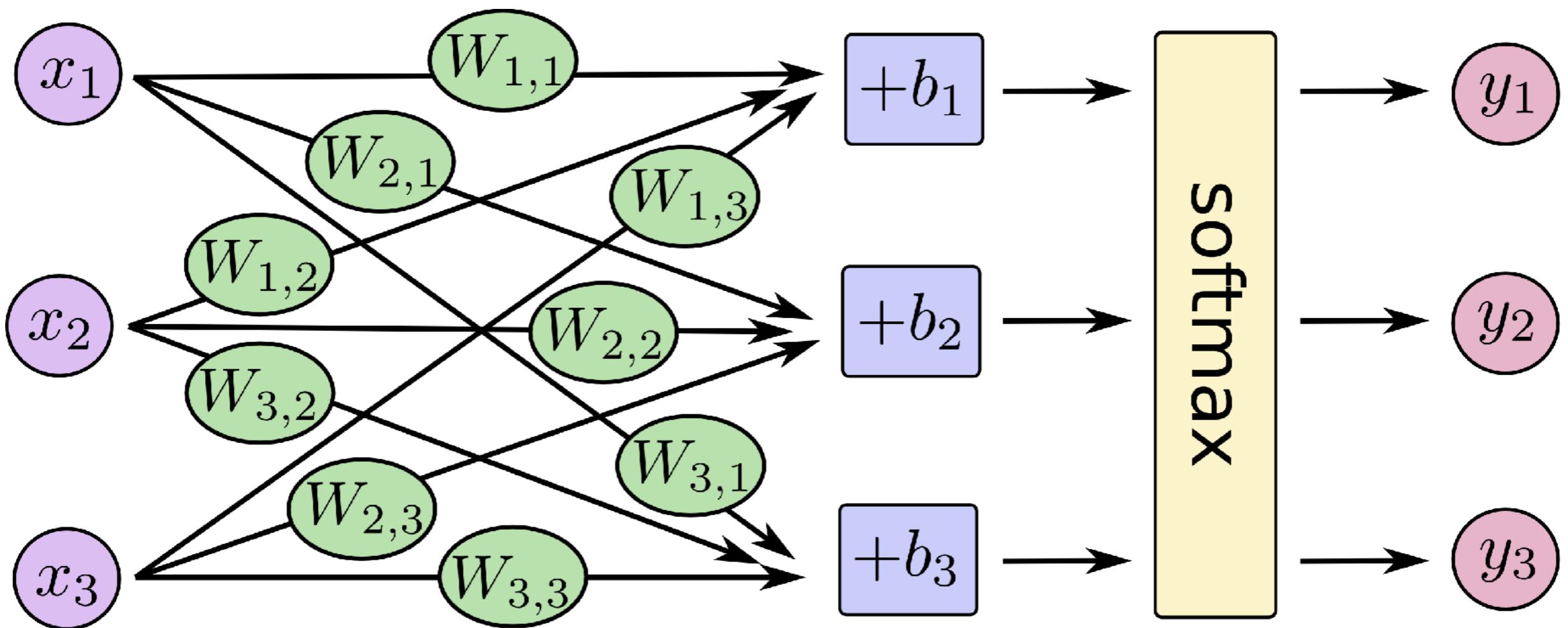


ReLU

Softmax:

$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$$

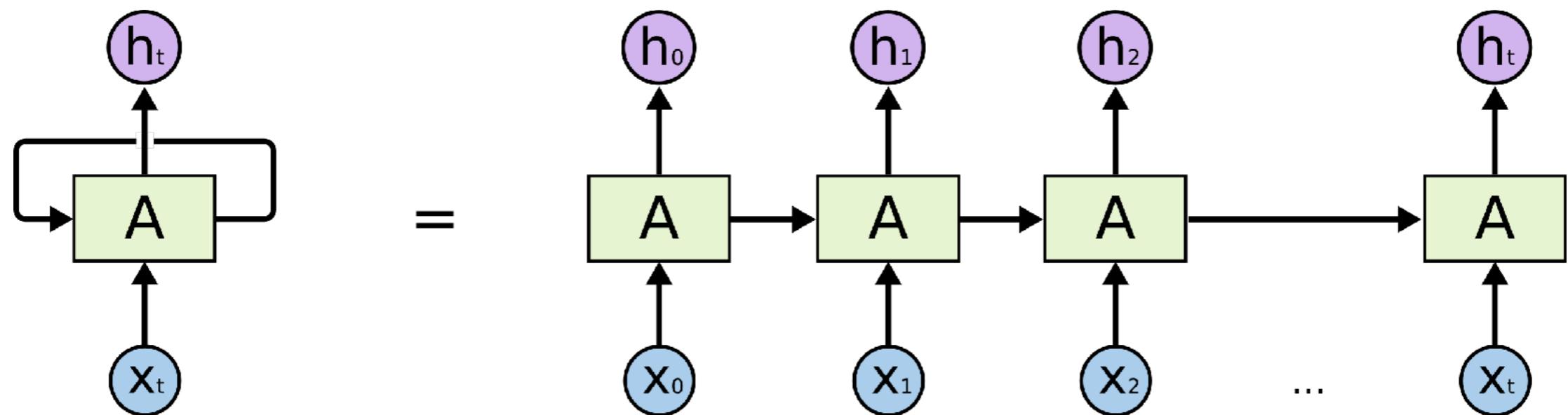
# Feed-Forward Neural Network



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

# Recurrent Neural networks

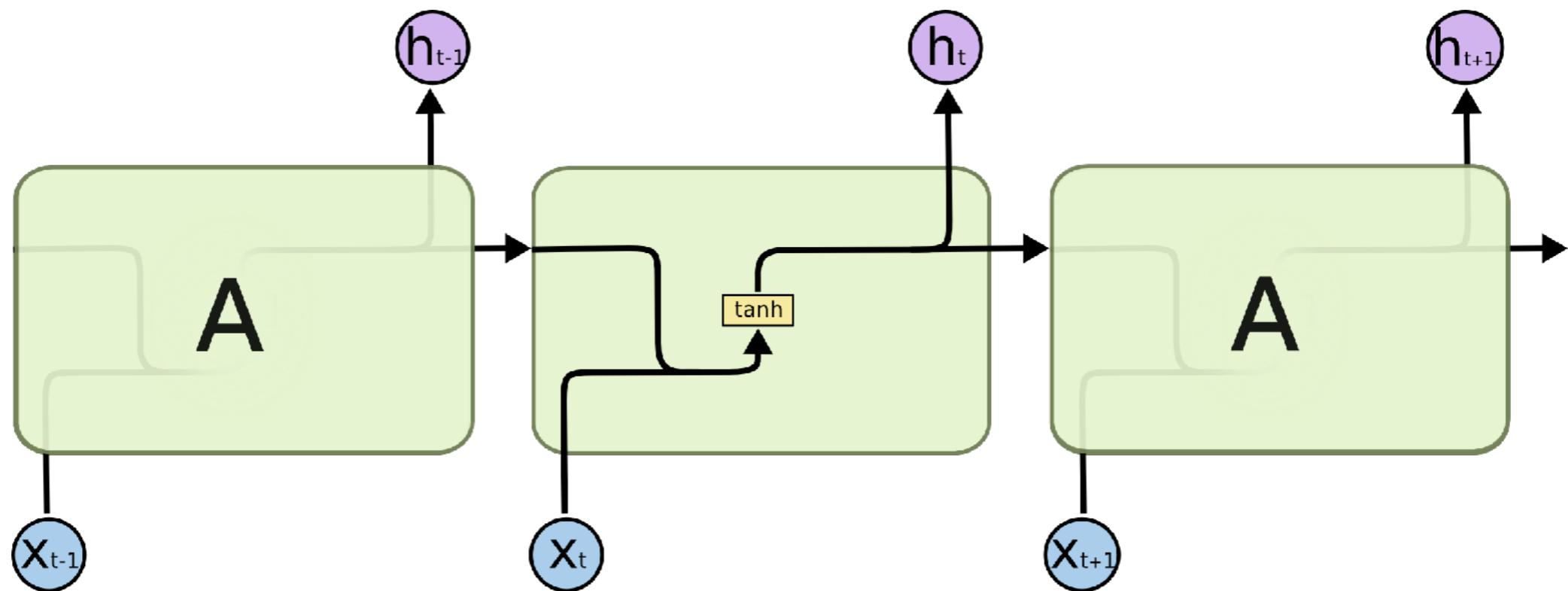
## 1/2



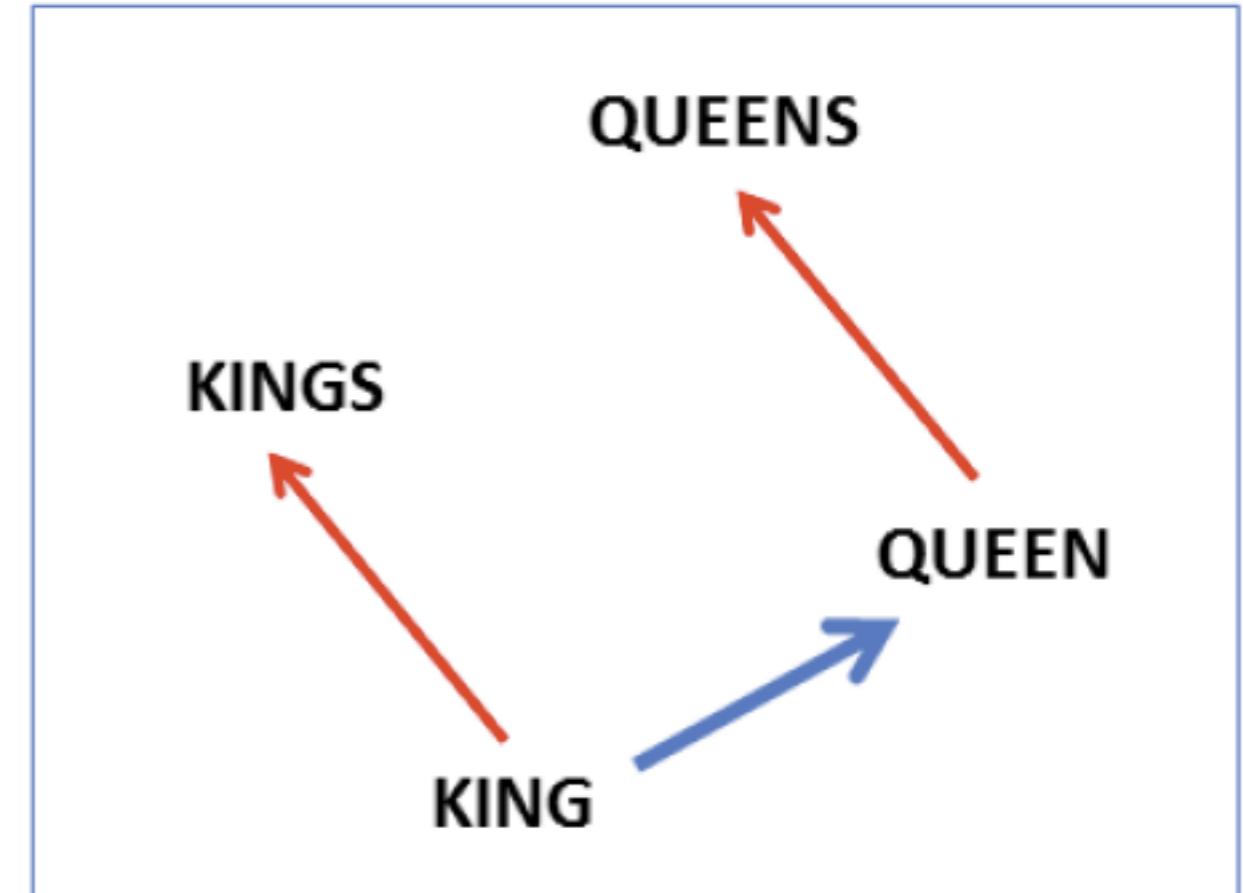
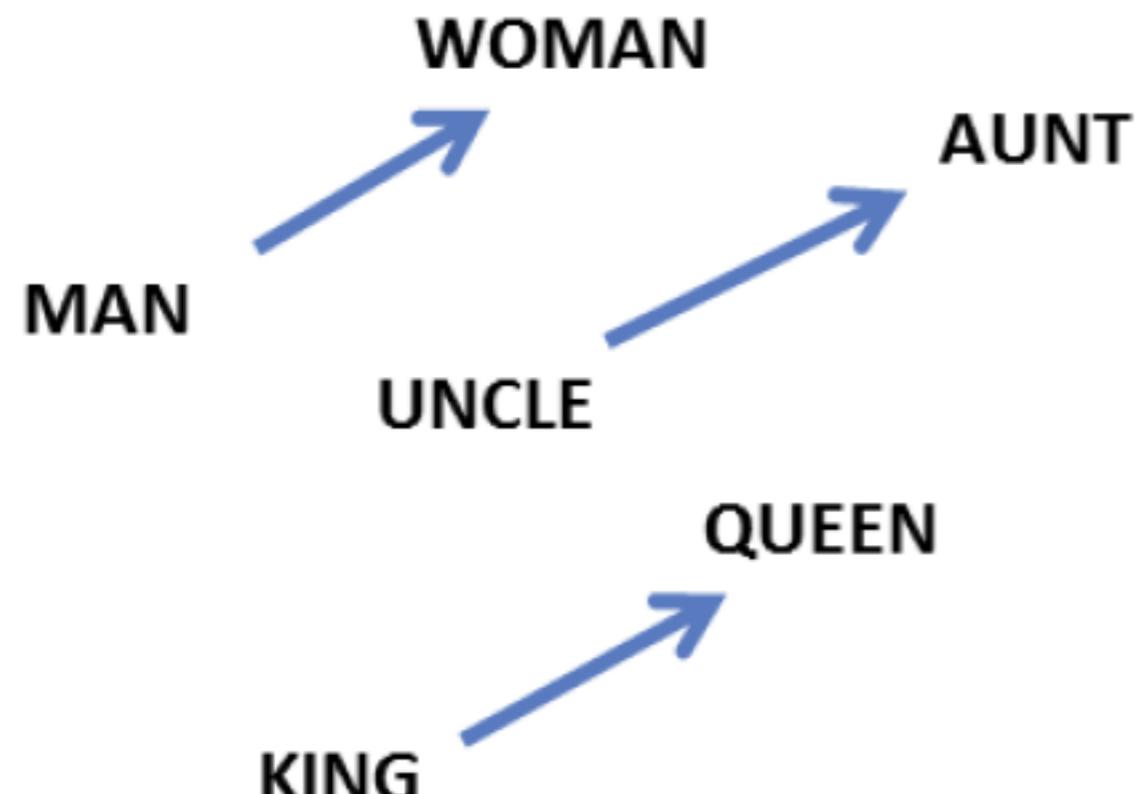
source: <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>

# Recurrent Neural Networks

2/2



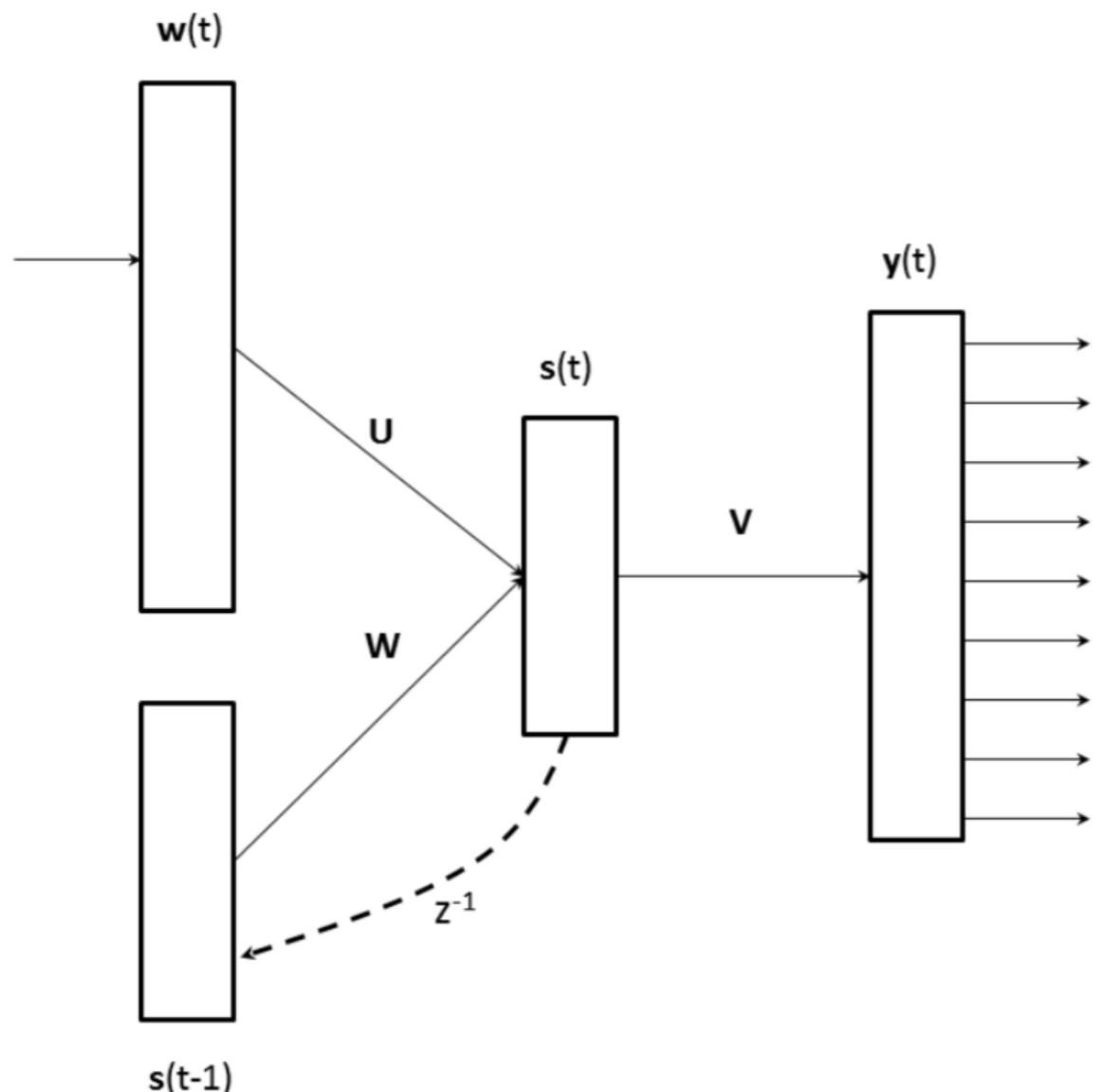
# word2vec



**king** is to **kings** as **queen** to ?.

$$v(\text{kings}) - v(\text{king}) = v(\text{queens}) - v(\text{queen})$$

# Recurrent Neural Network Language Modeling Toolkit

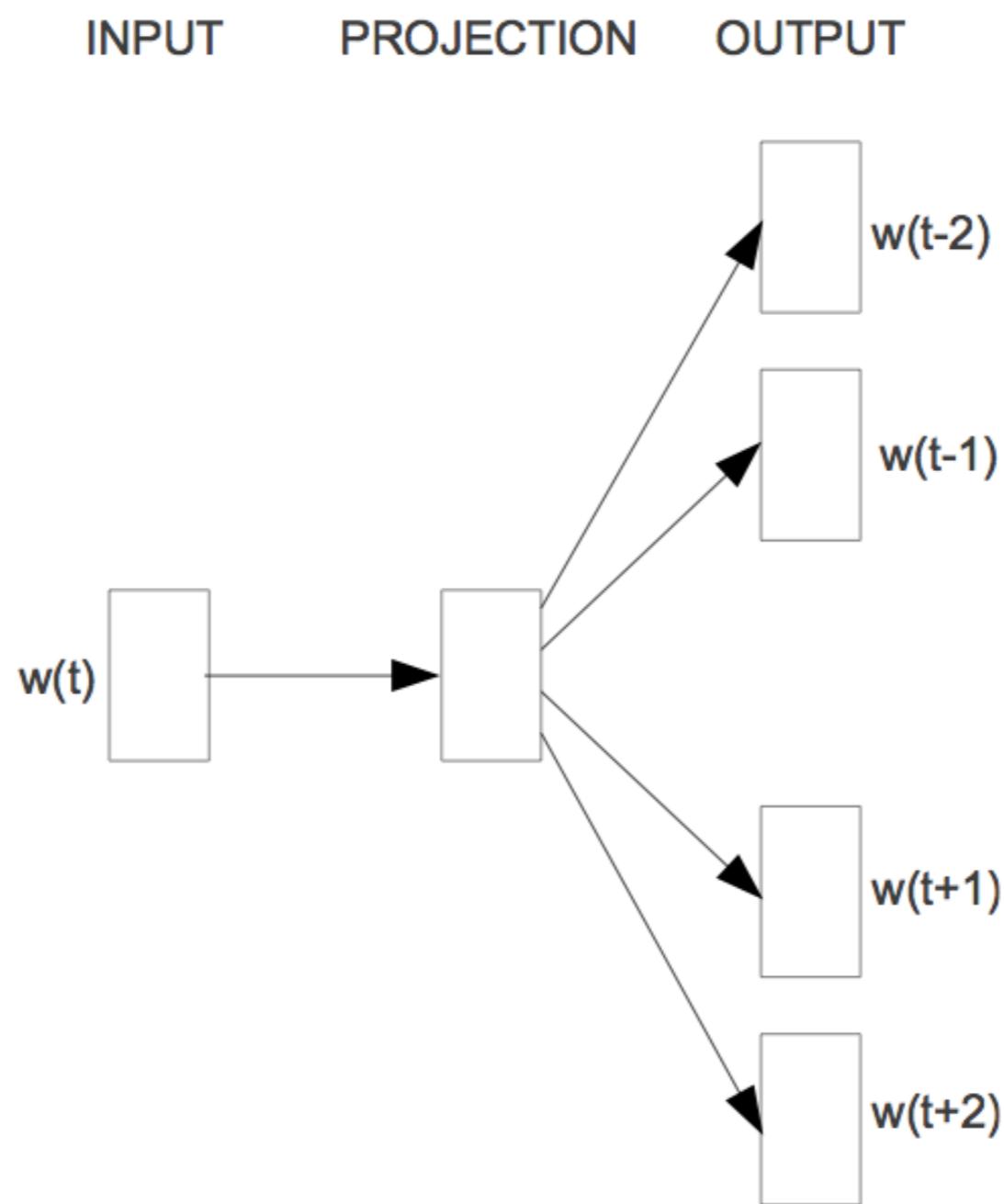


$$\mathbf{s}(t) = f(\mathbf{U}\mathbf{w}(t) + \mathbf{W}\mathbf{s}(t-1))$$

$$\mathbf{y}(t) = g(\mathbf{V}\mathbf{s}(t)),$$

$$f(z) = \frac{1}{1 + e^{-z}}, \quad g(z_m) = \frac{e^{z_m}}{\sum_k e^{z_k}}.$$

# The Skip-gram model



# Skip-gram improvements

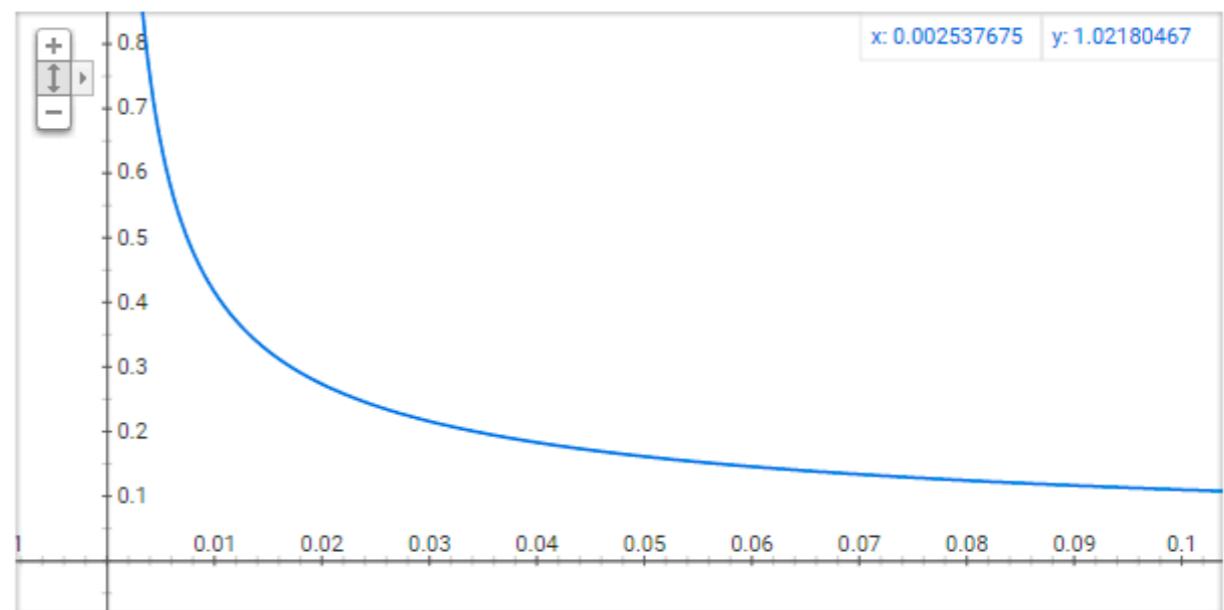
## Subsampling frequent inputs

$$P(w_i) = \left( \sqrt{\frac{z(w_i)}{0.001}} + 1 \right) \cdot \frac{0.001}{z(w_i)}$$

$z(w)$  Relative frequency of word  $w$

$P(w)$  Probability of keeping word  $w$

Graph for  $(\sqrt{x/0.001}+1)*0.001/x$



## Negative sampling

We select only 5-20 negative samples in the loss function.  
The probability of picking a word  $w$  is given by  $z(w)$ .

# **Experiments with word2vec**

**04-Word2vec-in-gensim.ipynb**

**05-Review-classification-w2v-assignment.ipynb**

# Outline

## Day 3

- Subword tokenization
- LSTM and GRU
- Recurrent neural networks for text classification
- Text generation using RNN
- Practical task on text generation
- Attention is all you need
- Transformers (GPT3, BERT, XLNET)
- Practical task on classification using BERT

# Traditional tokenization

## NLTK tokenizers

```
>>> from nltk.tokenize import word_tokenize #simple
>>> from nltk.tokenize.moses import MosesTokenizer #enables detokenization
>>> from nltk.tokenize import ToktokTokenizer #fast
>>>
>>> moses = MosesTokenizer()
>>> toktok = ToktokTokenizer()
>>>
>>> text = "Welcome to Machine Learning College."
>>> print(word_tokenize(text))
>>> print(moses.tokenize(text))
>>> print(toktok.tokenize(text))
['Welcome', 'to', 'Machine', 'Learning', 'College', '.']
['Welcome', 'to', 'Machine', 'Learning', 'College', '.']
['Welcome', 'to', 'Machine', 'Learning', 'College', '.']
```

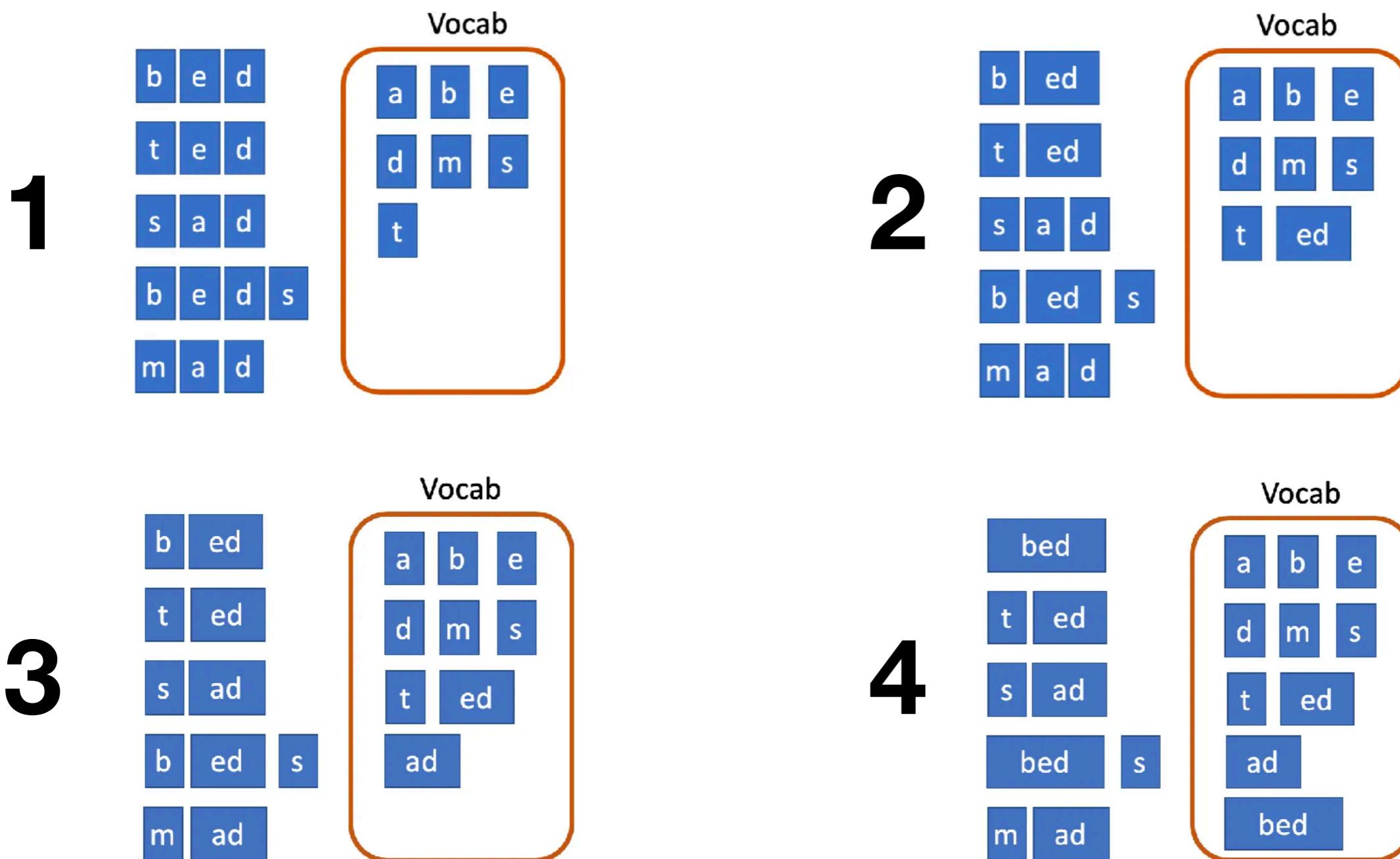
# Traditional tokenization

## SpaCy tokenizer

```
>>> import spacy
>>> sp = spacy.load('en_core_web_sm')
>>> tokens = sp("Welcome to Machine Learning College.")
>>>
>>> [word.text for word in tokens]
['Welcome', 'to', 'Machine', 'Learning', 'College', '.']
```

# Subword tokenization

## Byte-pair encoding



# Subword tokenization

## Wordpiece and sentencepiece tokenization

Merges bigrams with maximum mutual information instead of maximum frequency.

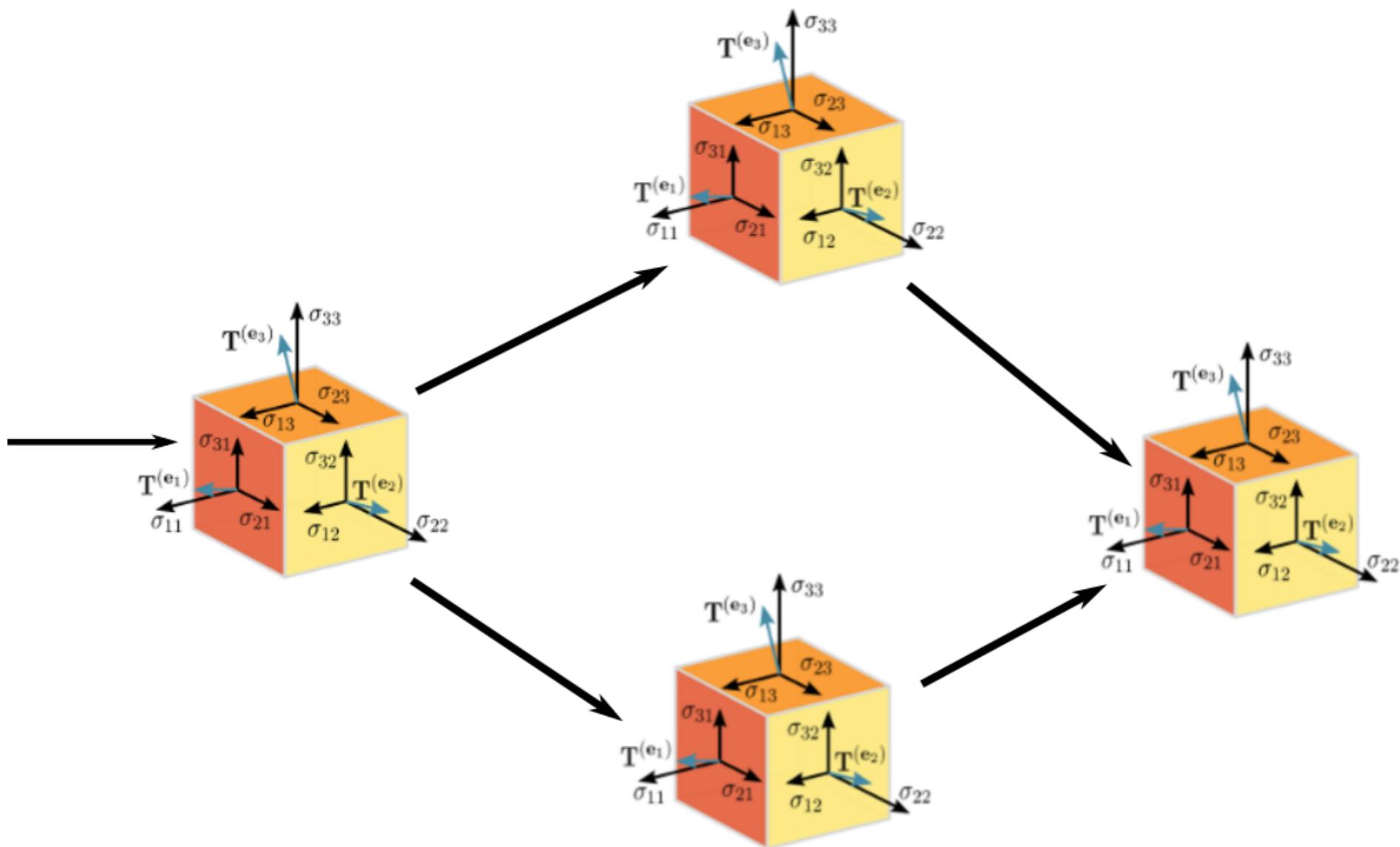
$$I(x, y) = \log \left( \frac{p(x, y)}{p(x) p(y)} \right)$$

playing      ->      play, ##ing

# What is not TensorFlow



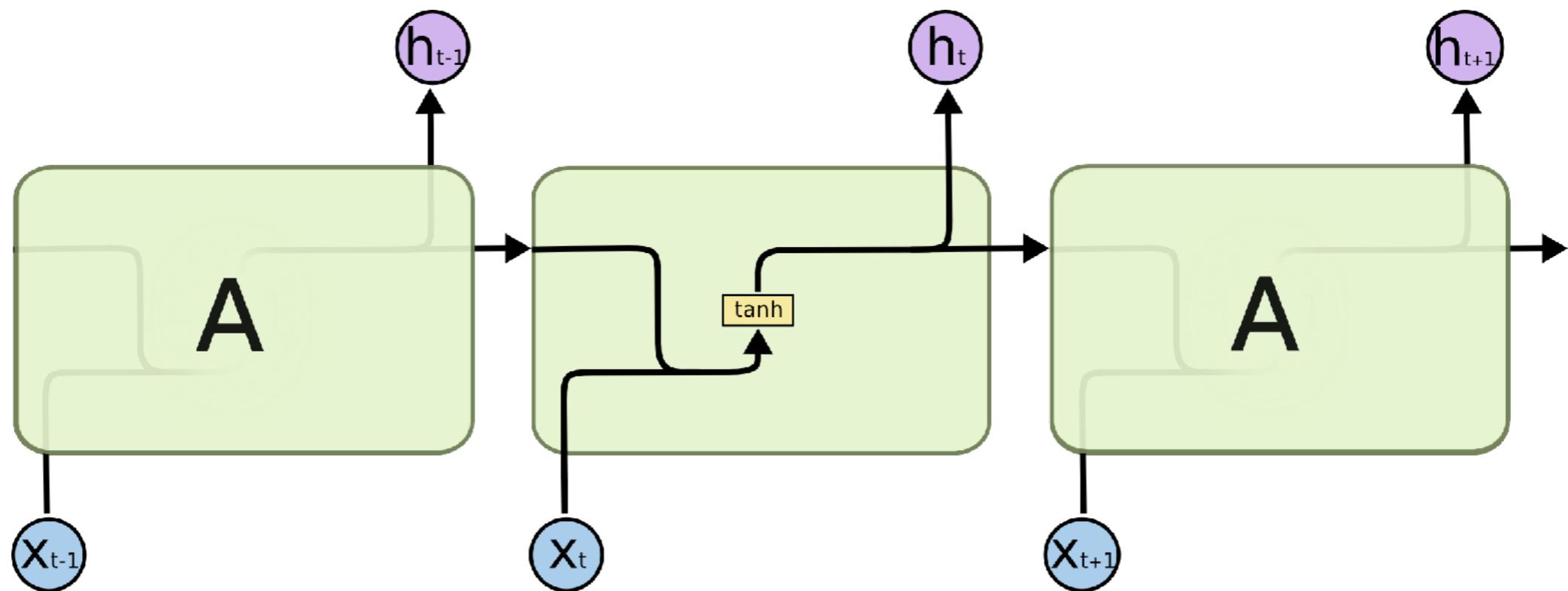
# What is TensorFlow?



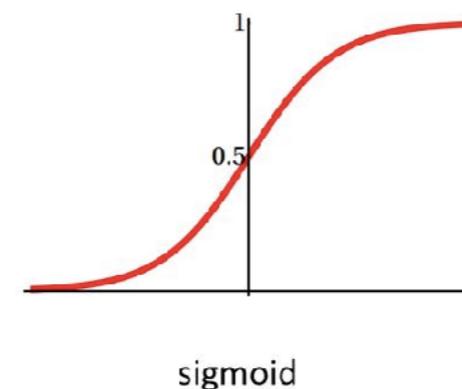
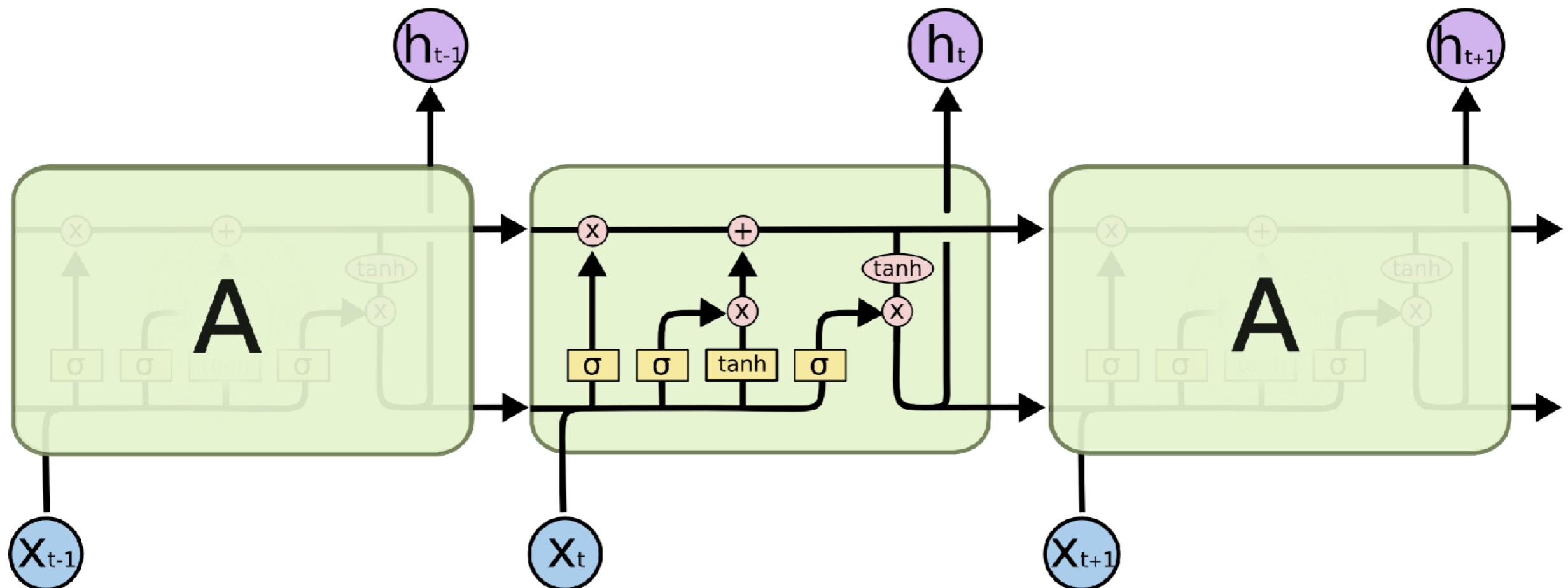
# Keras tutorial

**Keras-introduction.ipynb**

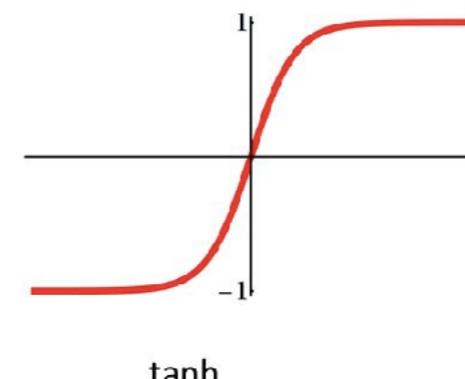
# Recurrent Neural Networks



# Long Short-Term Memory



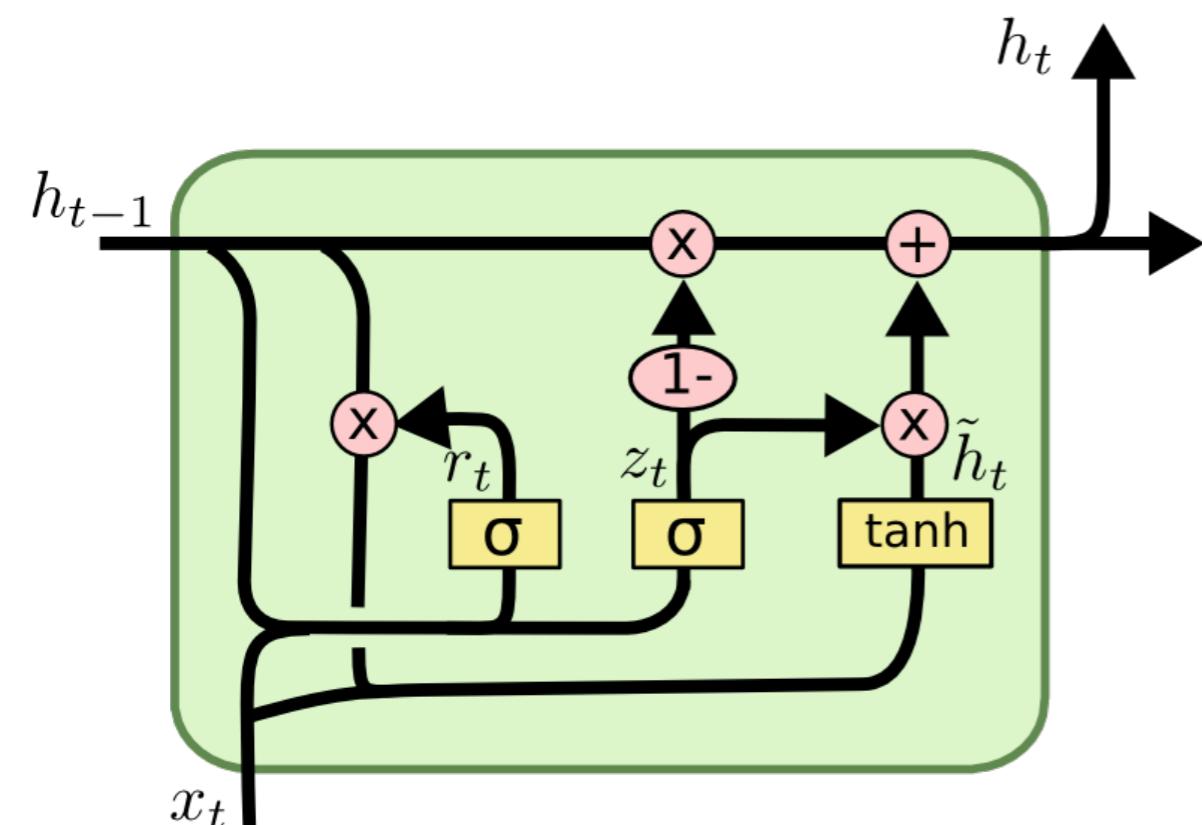
sigmoid



tanh

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

# Gated Recurrent Unit



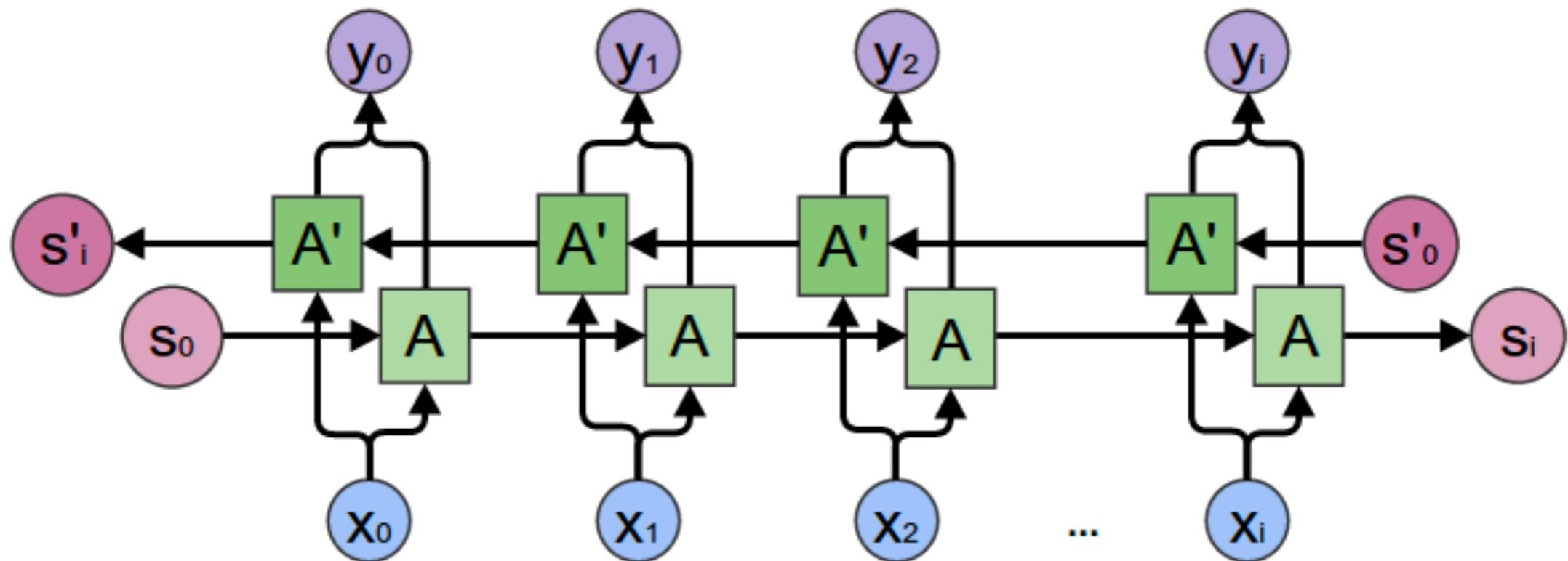
$$z_t = \sigma (W_z \cdot [h_{t-1}, x_t])$$

$$r_t = \sigma (W_r \cdot [h_{t-1}, x_t])$$

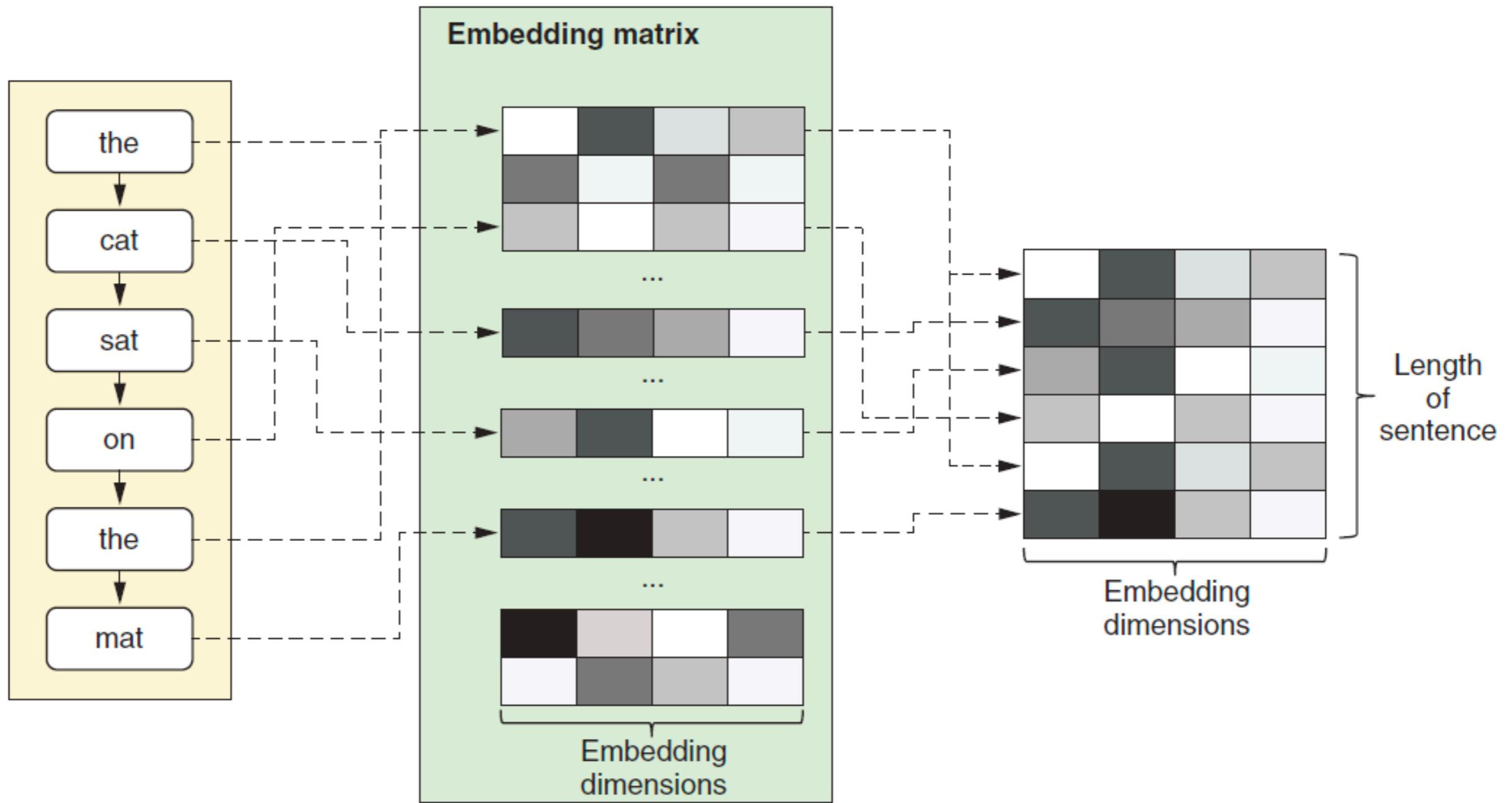
$$\tilde{h}_t = \tanh (W \cdot [r_t * h_{t-1}, x_t])$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

# Bidirectional recursive layer in Keras



# Embedding layer in Keras



# **Text classification with bidirectional LSTM**

**06-Review-classification-LSTM.ipynb**

# Language models for text generating

Nacházíte se: Úvod > Oddělení > Krásná literatura > Poezie > Česká a slovenská poezie > Elektronická kniha Poezie umělého světa



# Poezie umělého světa [E-kniha]

Jiří Materna



Hodnotilo 7 uživatelů, zatím žádné recenze, [napsat vlastní recenzi](#)

**Popis:** Elektronická kniha, 50 stran, bez zabezpečení DRM,  ePUB,  Mobi,  PDF, česky - [více](#)



## Stáhnout



**Zdarma**

K dispozici pro **okamžité** stáhnutí

Ke stažení

Anotace

Všechny básně v této knize byly automaticky vygenerovány počítačem za pomocí umělých neuronových sítí. Neuronová síť sama o sobě nic neumí a je třeba ji natrénovat pro činnost, kterou má vykonávat.



## LISTOPAD

usínám, pláču, umírám, přemýšlím  
co cítíš ty?  
cítim tvou slabost  
a whisky

## NOVEMBER

I am falling asleep, crying, dying, thinking  
what do you feel?  
I feel your weakness  
and whisky

## SPRAVEDLNOST

na tvou dekadentní duši  
ráno i v poledne  
bůh má připravenou kuší

## JUSTICE

for your decadent soul  
in the morning, in the evening  
the god has prepared a crossbow

# Metaphores

...tělo plné červánků...

...body full of blush of dawn...

...tak vzácný jako listí...

...as rare as leaves of trees...

# Language models for text generation

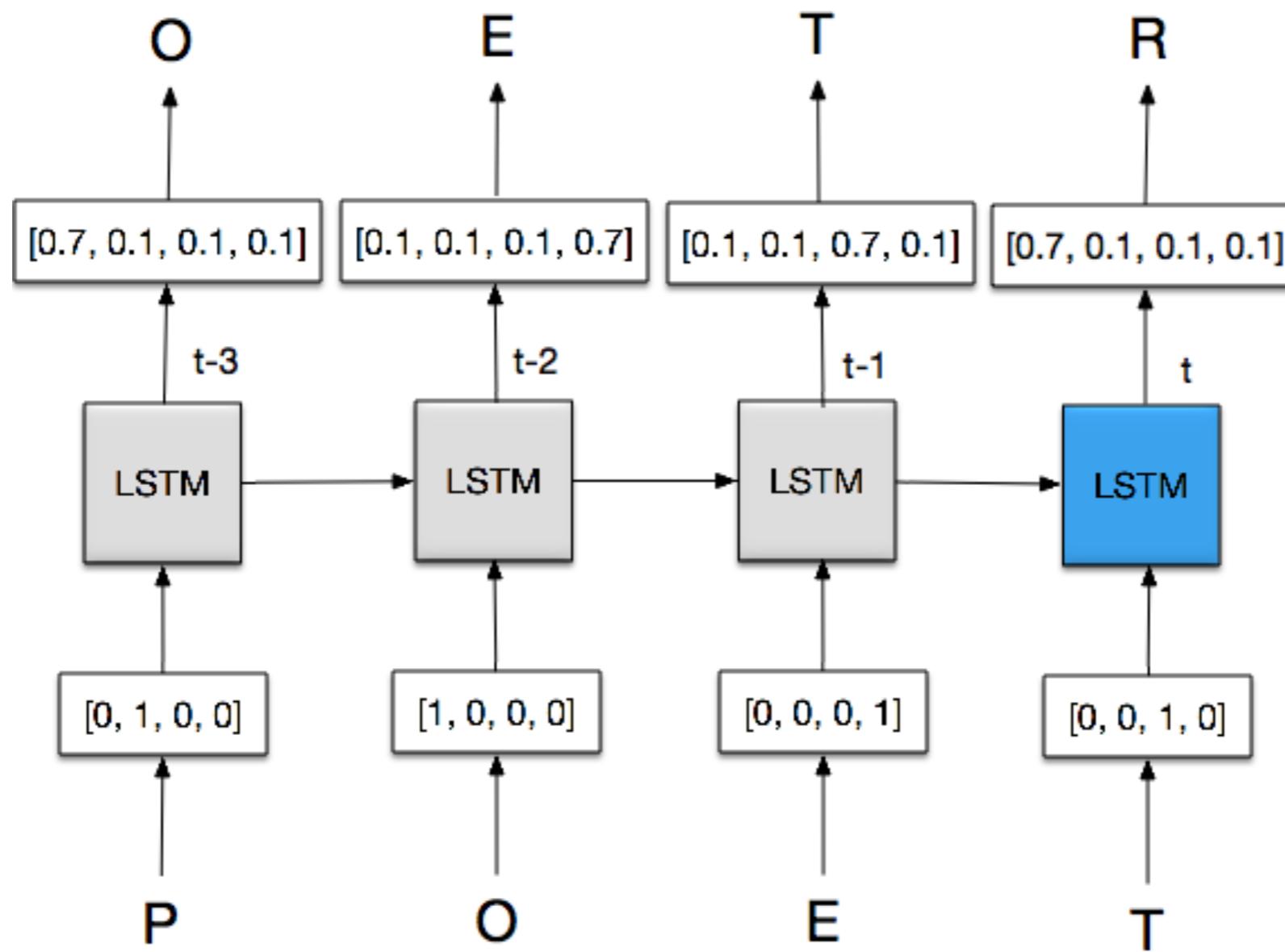
$$P(\text{college} \mid \text{machine, learning}) = 0.5$$

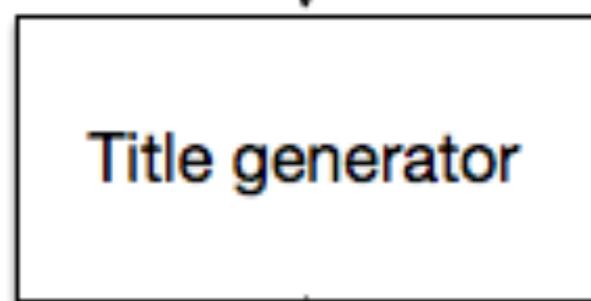
$$P(\text{tiger} \mid \text{machine, learning}) = 0.3$$

$$P(\text{yellow} \mid \text{machine, learning}) = 0.2$$

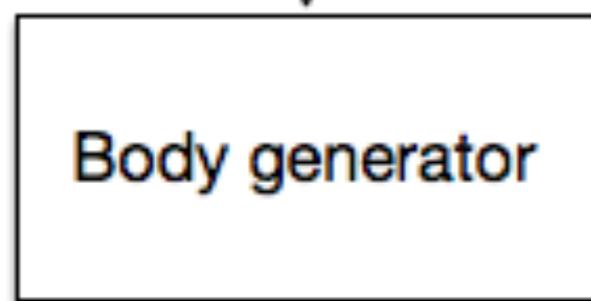
```
t ~ Uniform(0, 1)
s = 0
for v in Vocabulary:
    s += v.prob
    if t < s:
        return v.word
```

# LSTM language model





AUTUMN SONG



why don't you kill yourself?  
a phone call isn't hope  
this planet is still your home  
your time is still going on

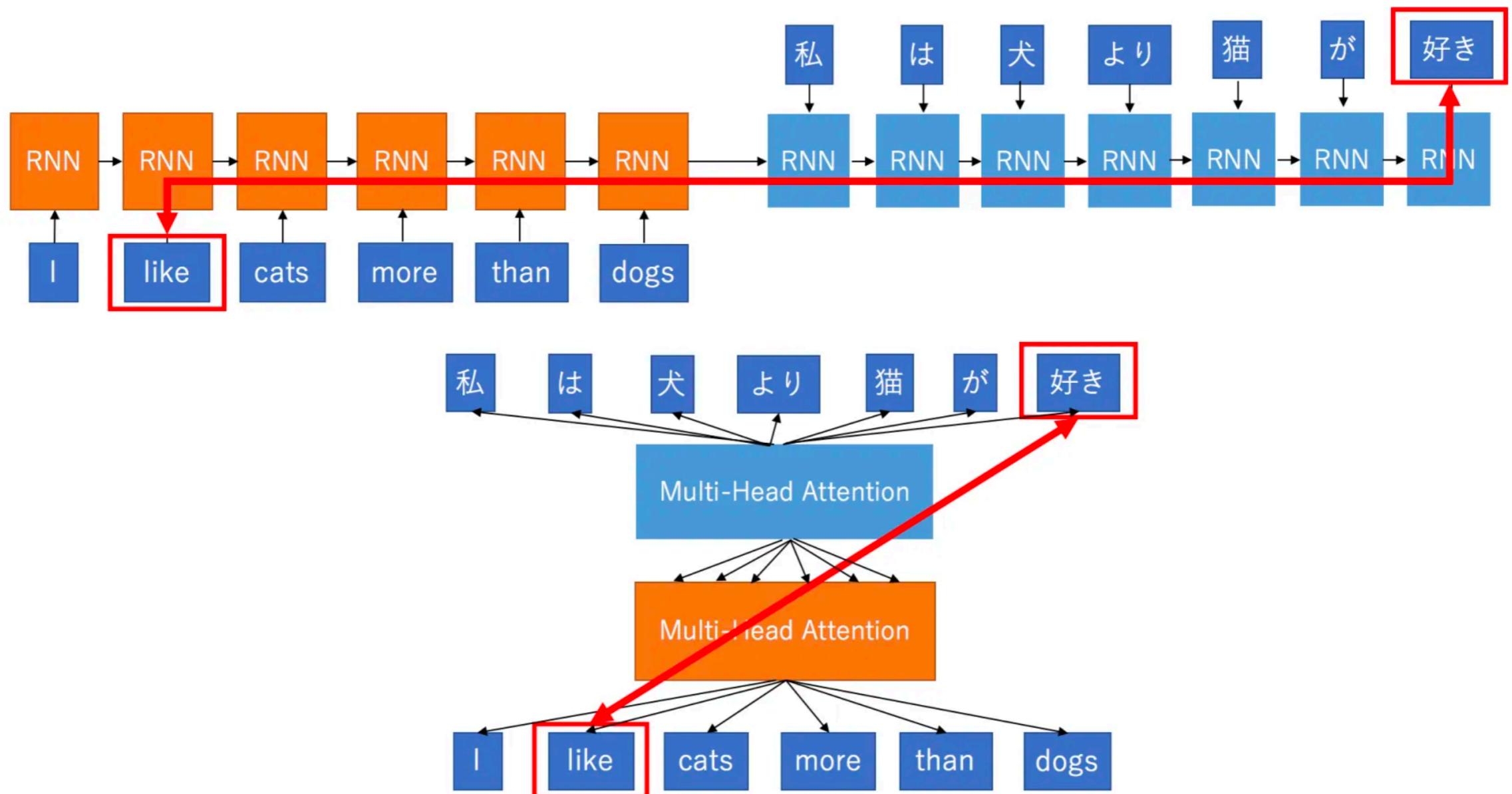
na na na...

I know, I'm revealing a book of dreams  
I'll find out I'm nothing more than this

# LSTM review generator

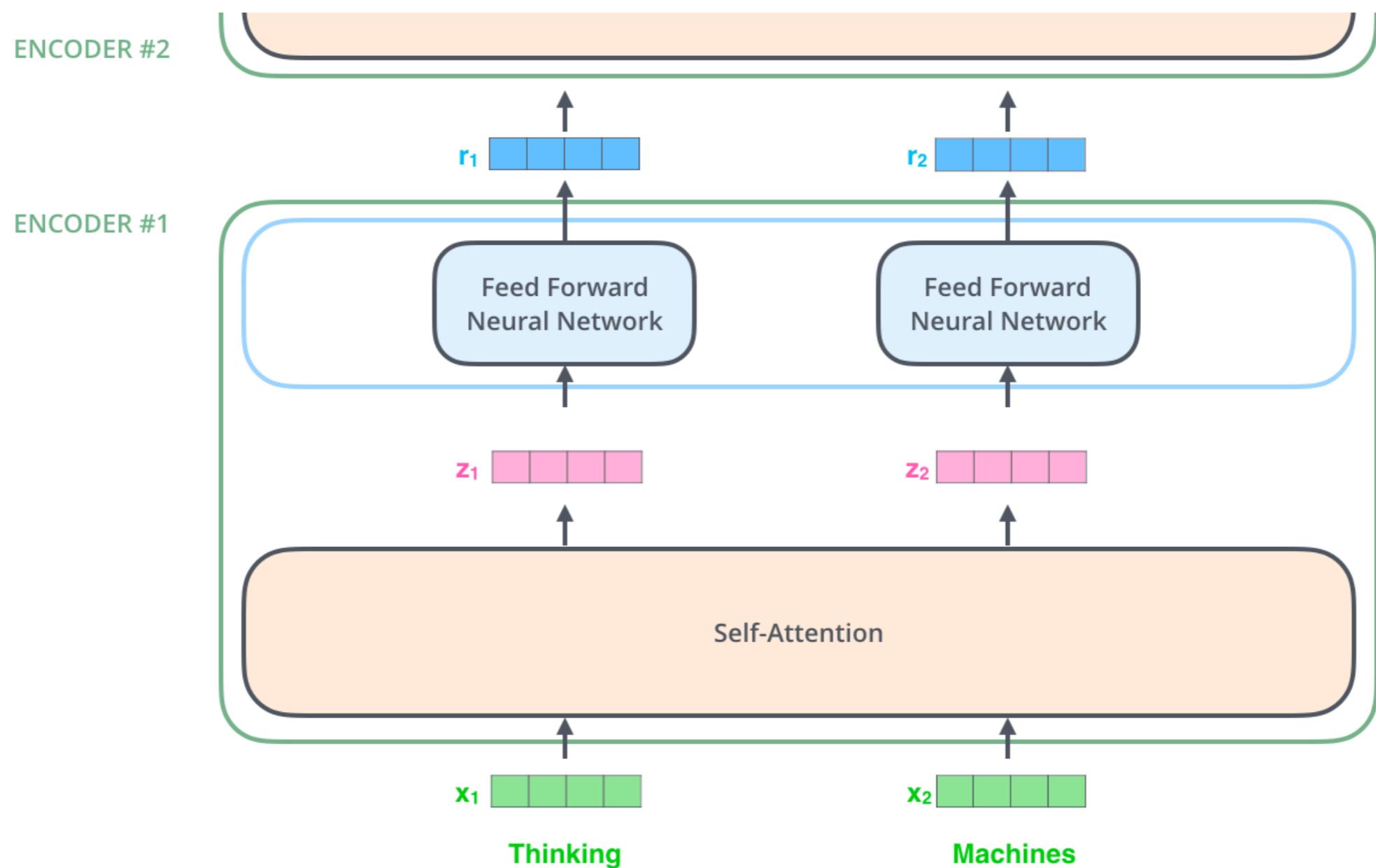
**07-Review-generator.ipynb**

# RNN vs. Transformer

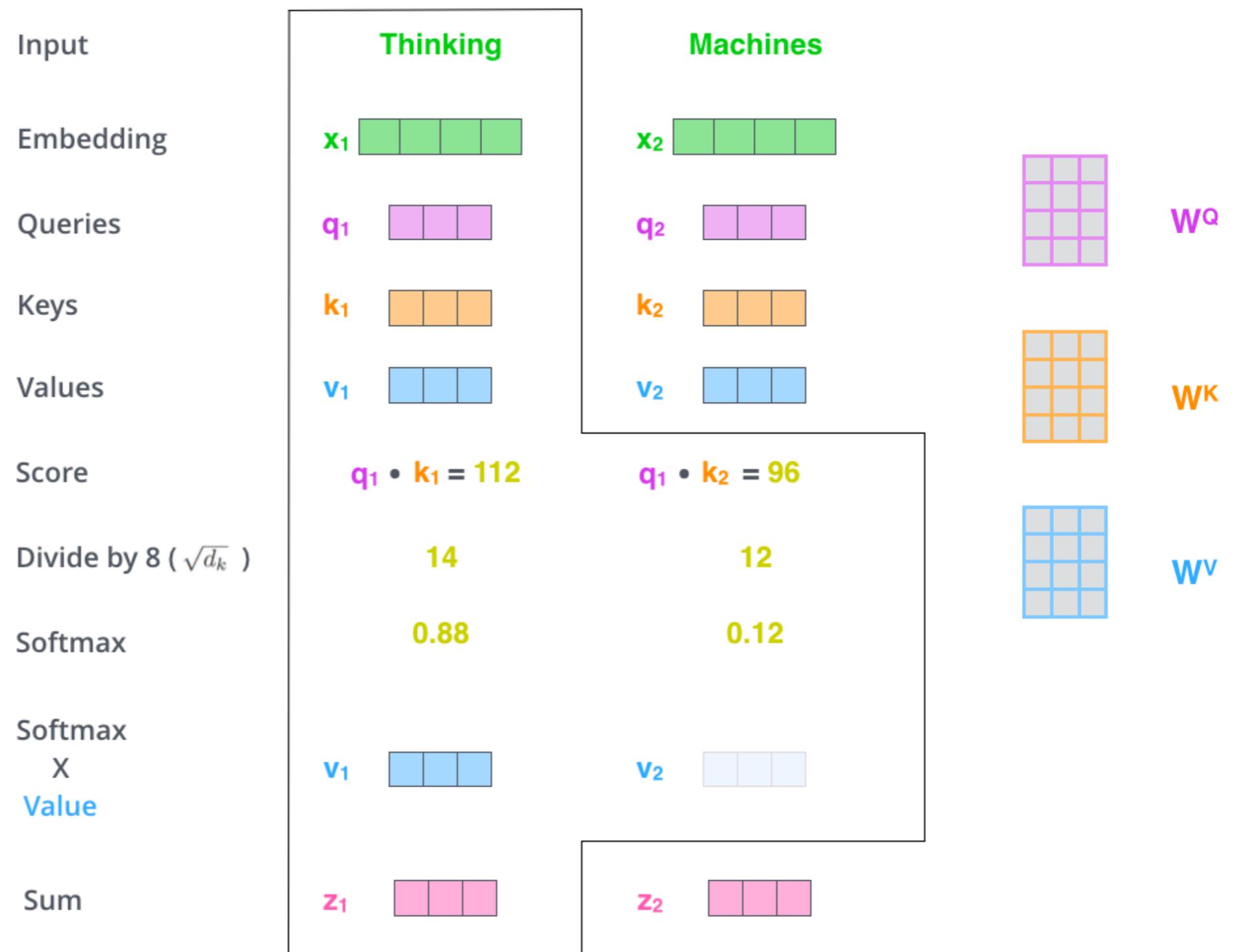


source: [www.mlexplained.com](http://www.mlexplained.com)

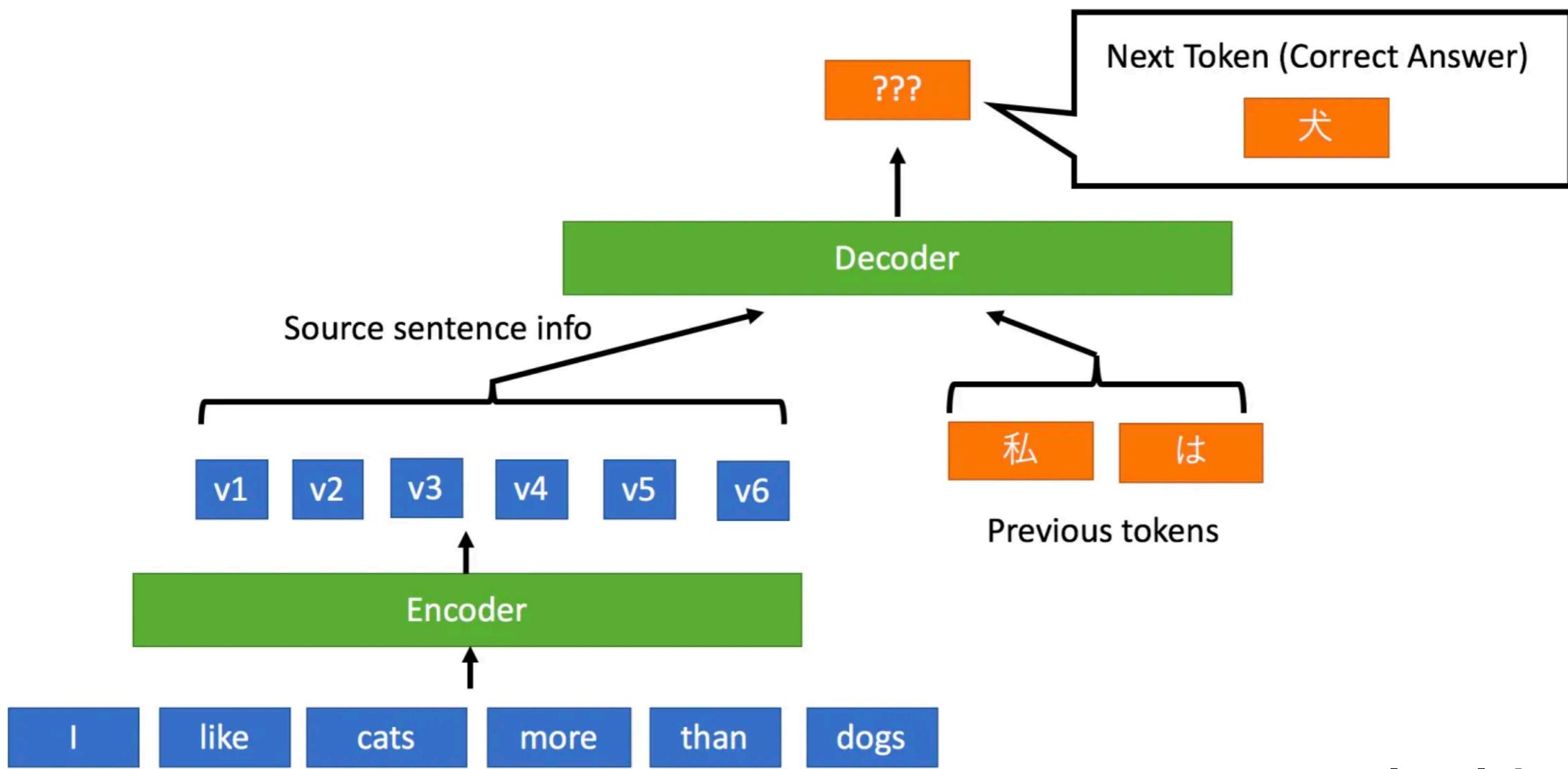
# Attention is all you need



# Self-attention



# Translation with Transformers



# GPT-2 Language model

**Donald Trump told...**

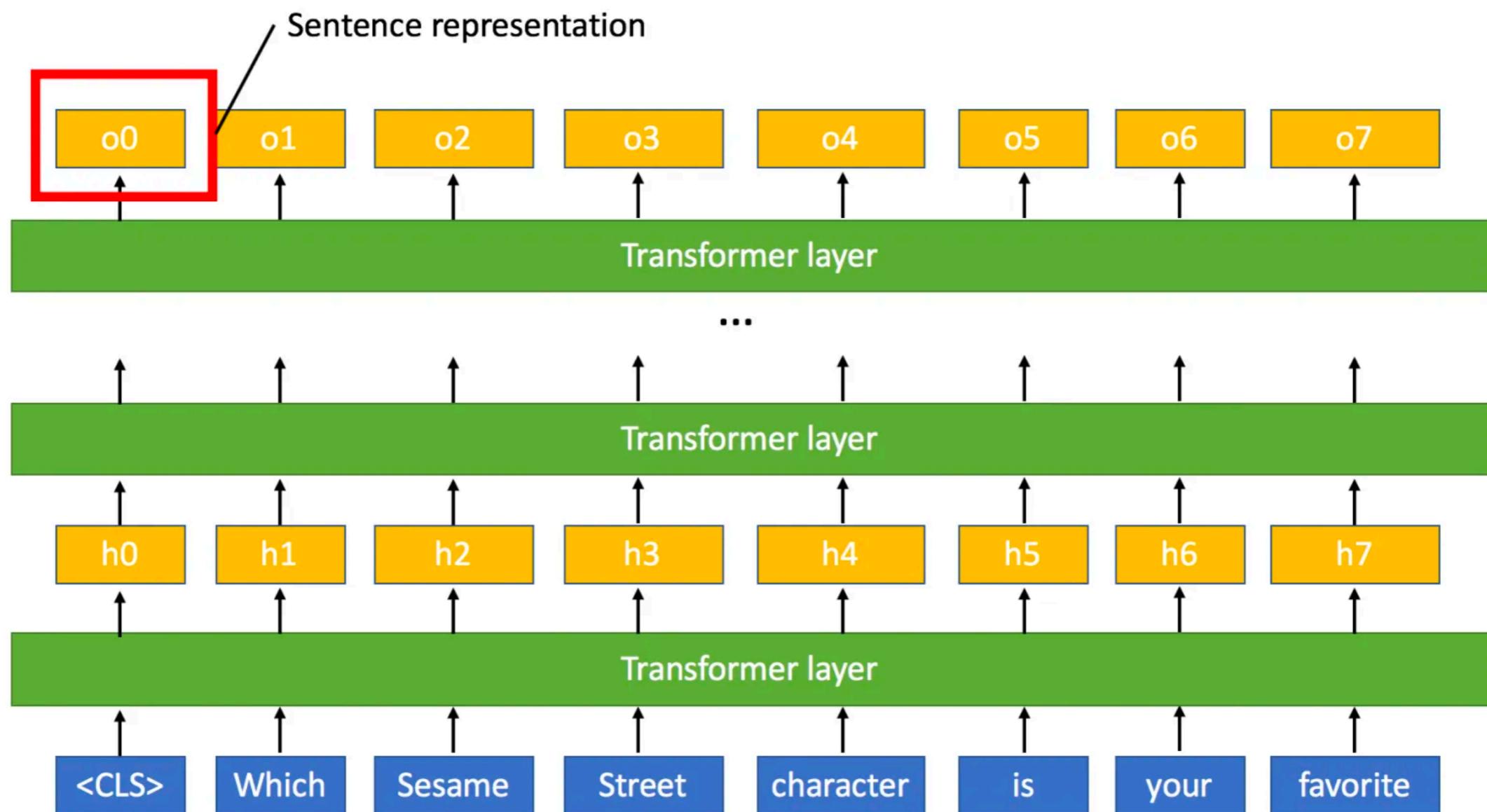
# GPT-2 Language model

**Donald Trump told** the Times he is preparing a "major speech" on his economic plans, but did not provide details on what it will entail.

"I'm getting ready for the speech. And I will have a major speech on Tuesday." Trump said during an interview in the White House residence.

**GPT-3:** <https://openai.com/blog/openai-api/>

# BERT (classification)



# Text classification using BERT

**08-Review-classification-BERT.ipynb**

# Outline

## Day 4

- Topic modeling
- Basics of the Probability theory
- Probabilistic Graphical Models
- Inference in Bayesian Networks
- Gaussian Linear Regression
- Gaussian Mixtures for clustering
- (Probabilistic) Latent Semantic Analysis
- Latent Dirichlet Allocation

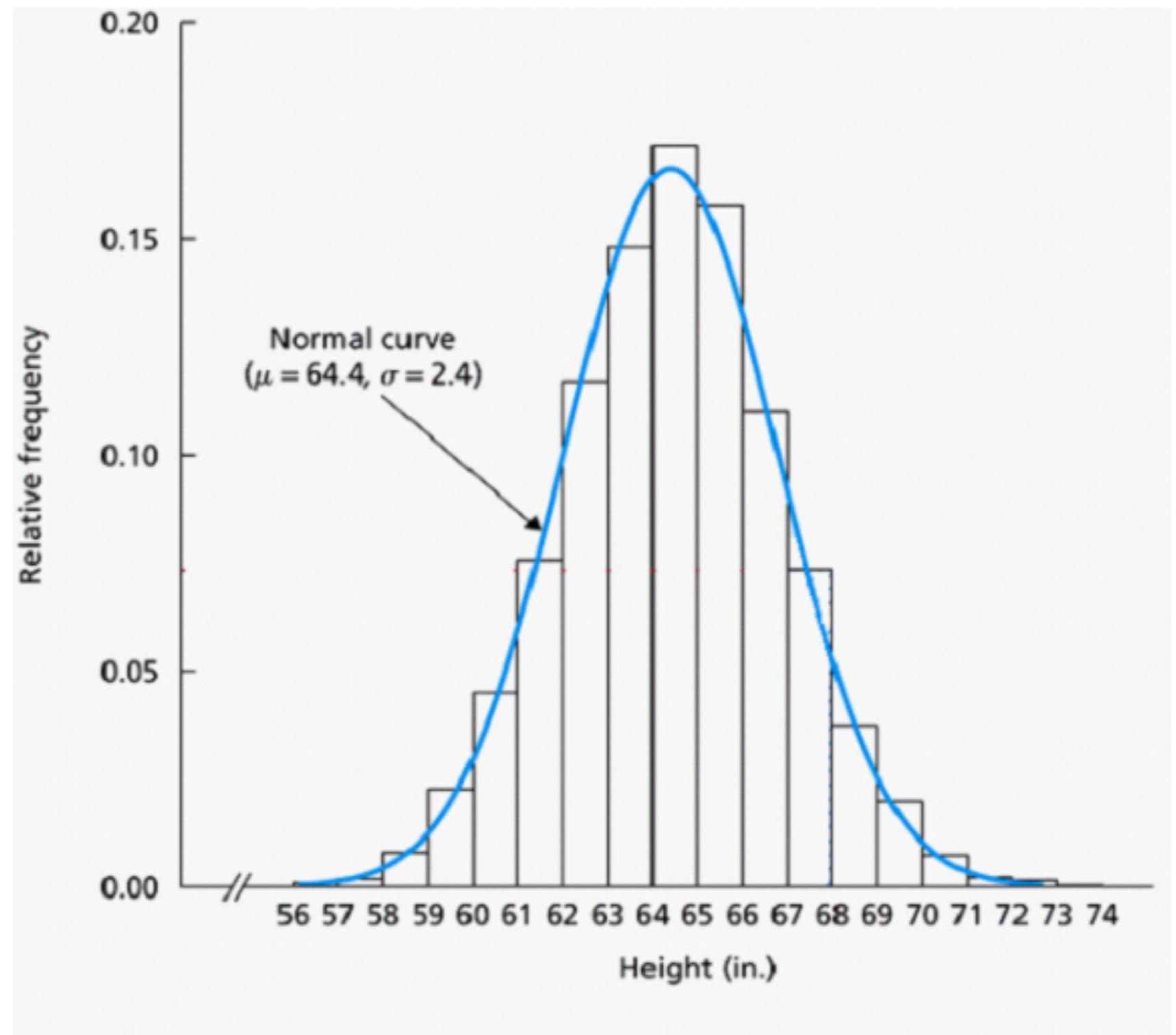
# Conditional probability and independence

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A)P(B|A)}{P(B)}$$

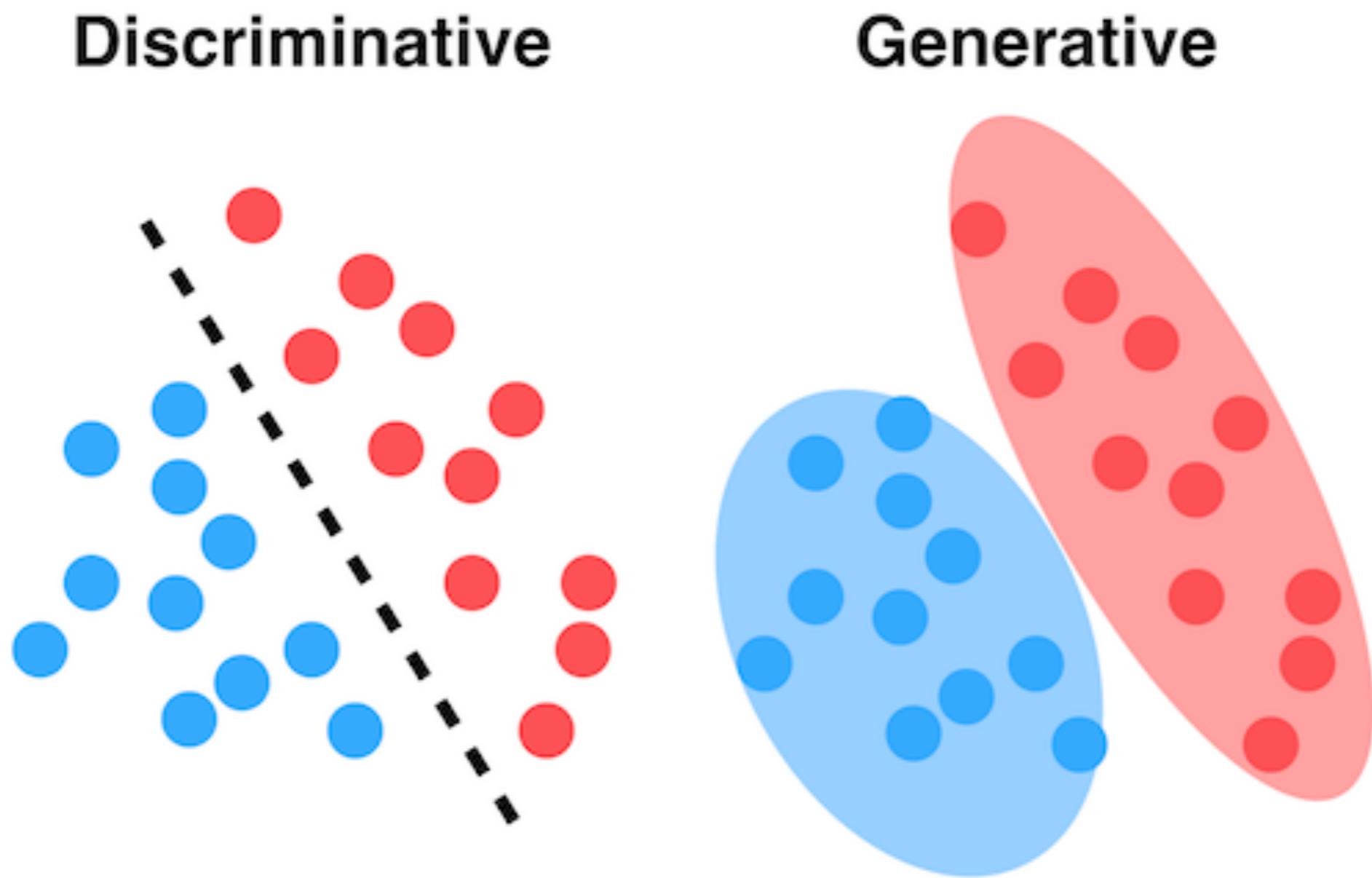
$$A \perp B \iff P(A \cap B) = P(A)P(B)$$

# Probability distribution

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$



# Discriminative vs. generative models



# Topic Modeling

## Topics

gene 0.04  
dna 0.02  
genetic 0.01  
...

life 0.02  
evolve 0.01  
organism 0.01  
...

brain 0.04  
neuron 0.02  
nerve 0.01  
...

data 0.02  
number 0.02  
computer 0.01  
...

## Documents

### Seeking Life's Bare (Genetic) Necessities

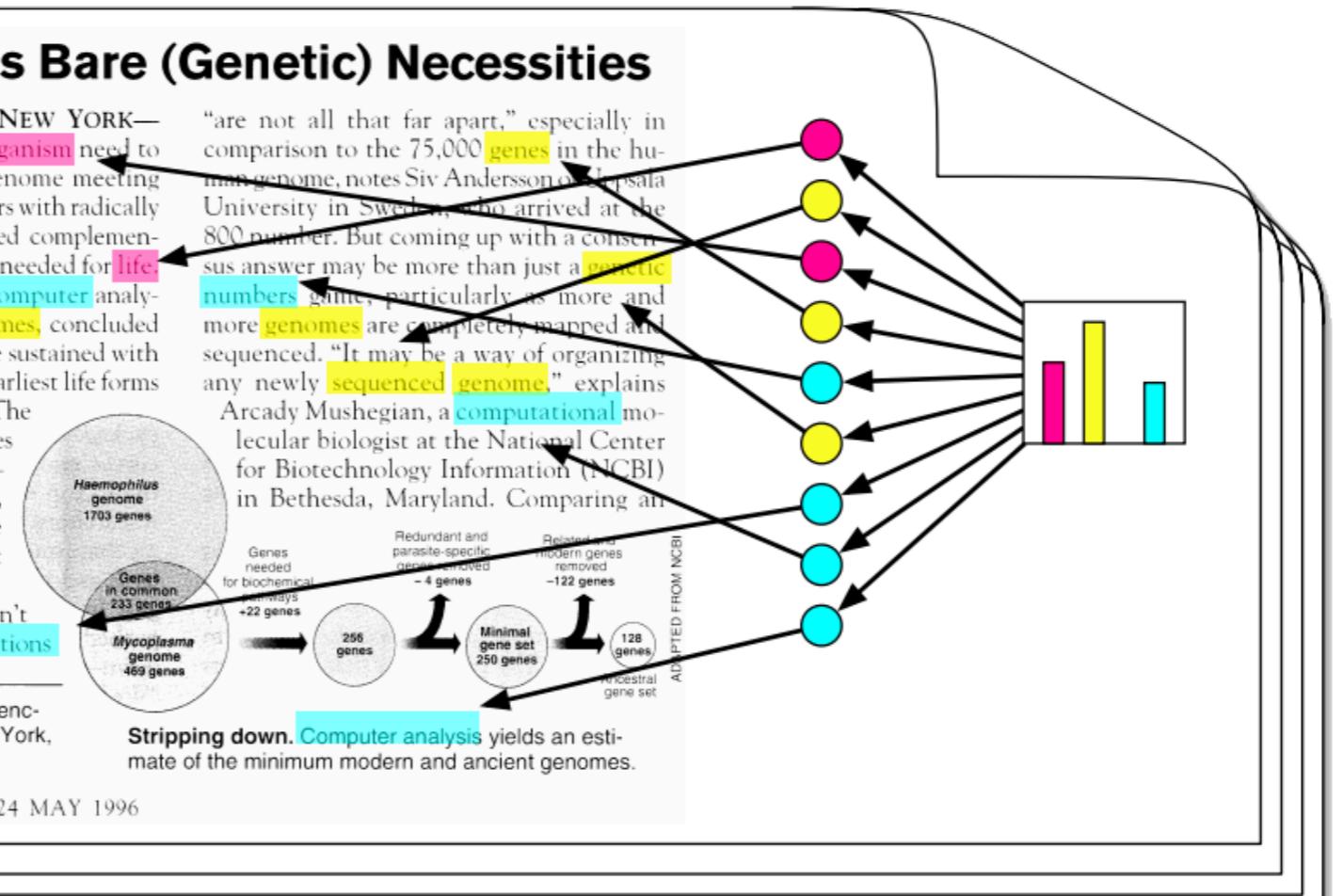
COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,\* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions

\* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

SCIENCE • VOL. 272 • 24 MAY 1996

## Topic proportions & assignments



# Generative model of people's heights

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

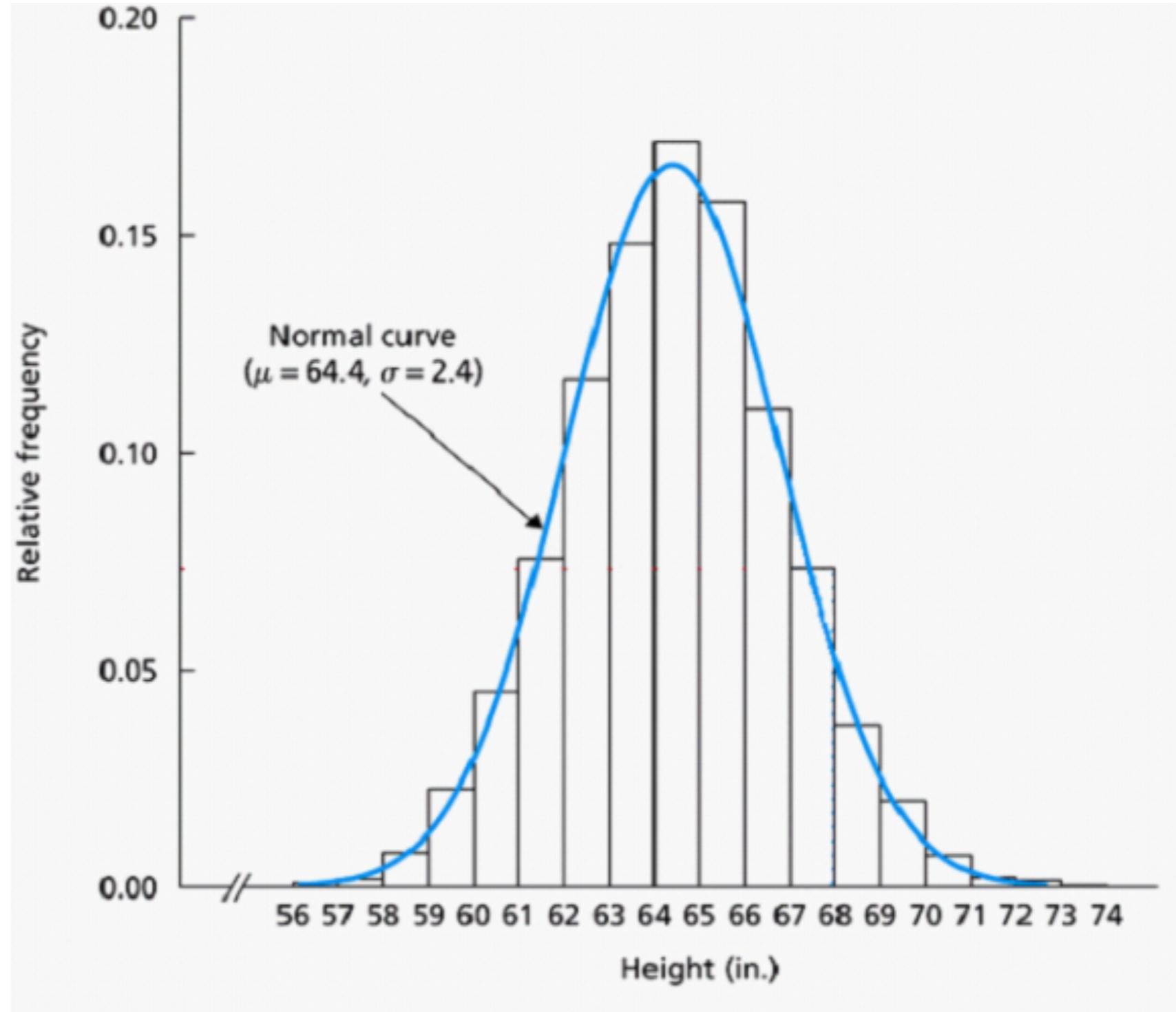
$$X = \{x_1, x_2 \dots x_n\}$$

$$X \sim N(\mu, \sigma^2), \alpha = (\mu, \sigma^2)$$

$$\bar{\alpha} = \arg \max_{\alpha} P(\alpha|X)$$

$$P(\alpha|X) = \frac{P(X|\alpha) \cdot P(\alpha)}{P(X)}$$

posterior  
likelihood  
prior

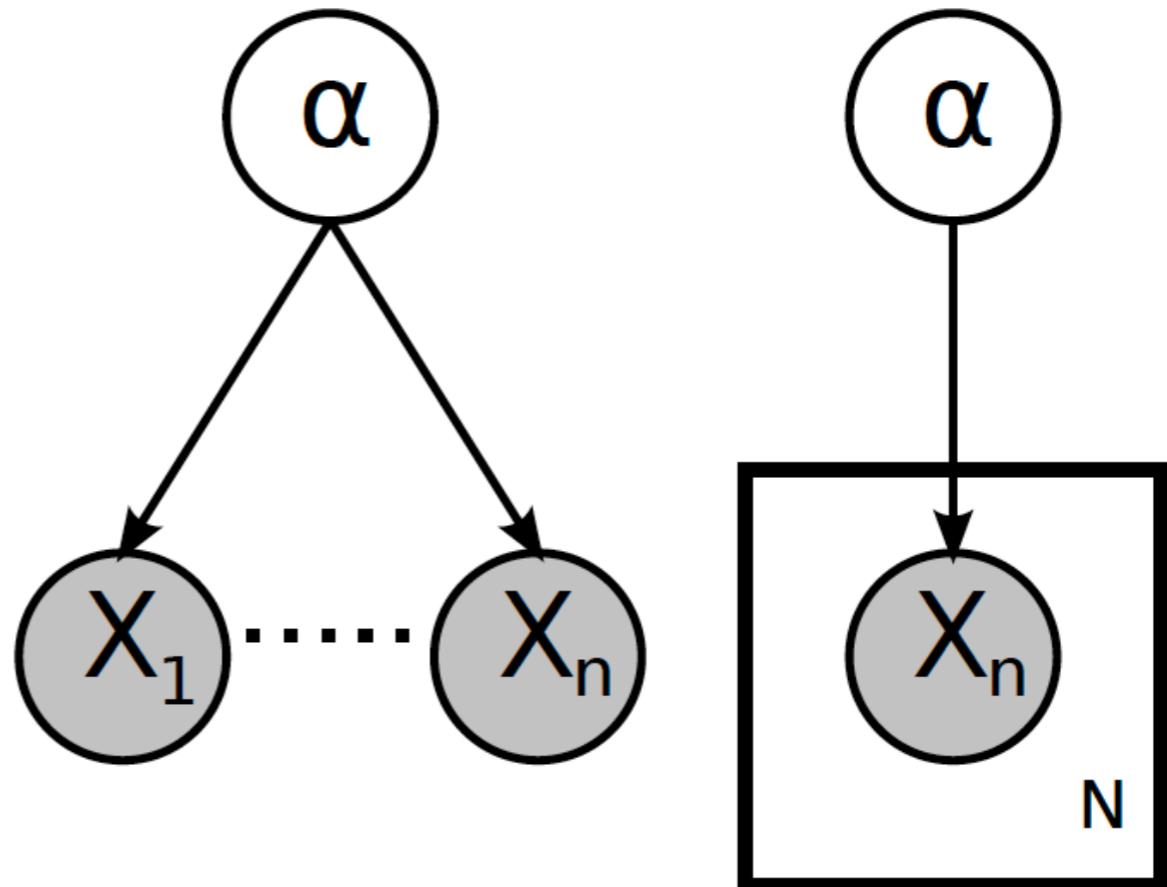


# Probabilistic graphical models

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

$$X = \{x_1, x_2 \dots x_n\}$$

$$X \sim N(\mu, \sigma^2), \alpha = (\mu, \sigma^2)$$



# Inference in graphical models

$$P(\alpha|X) = \frac{P(X|\alpha).P(\alpha)}{P(X)} \propto P(X|\alpha).P(\alpha) = \prod_{i=1}^n P(x_i|\alpha).P(\alpha)$$

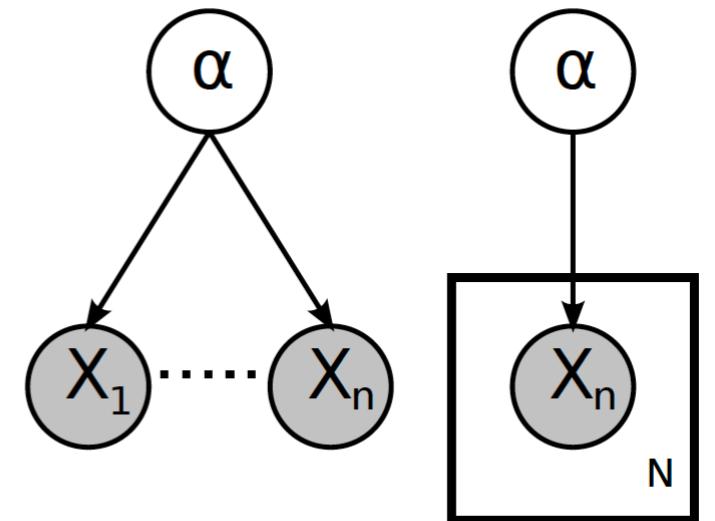
$$\bar{\alpha} = \arg \max_{\alpha} P(\alpha|X)$$

## Variational inference

1. Approximate the posterior function with a simpler one
2. Compute the hidden variables by minimization of KL Divergence of the true and simpler distributions

## Sampling (e.g. Gibbs sampling)

1. Draw samples from the true posterior
2. Compute mean of the samples



# Generative model for linear regression

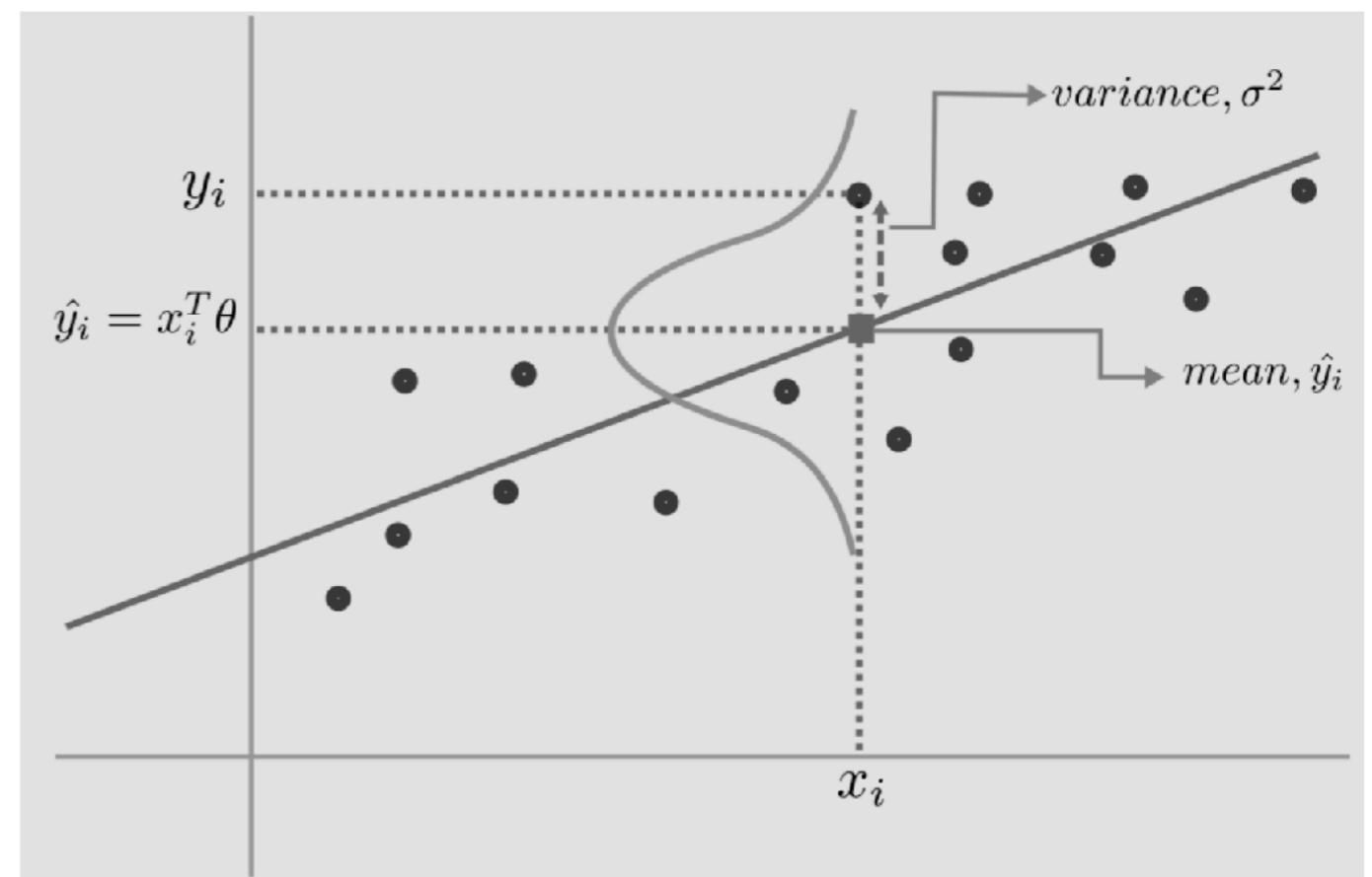
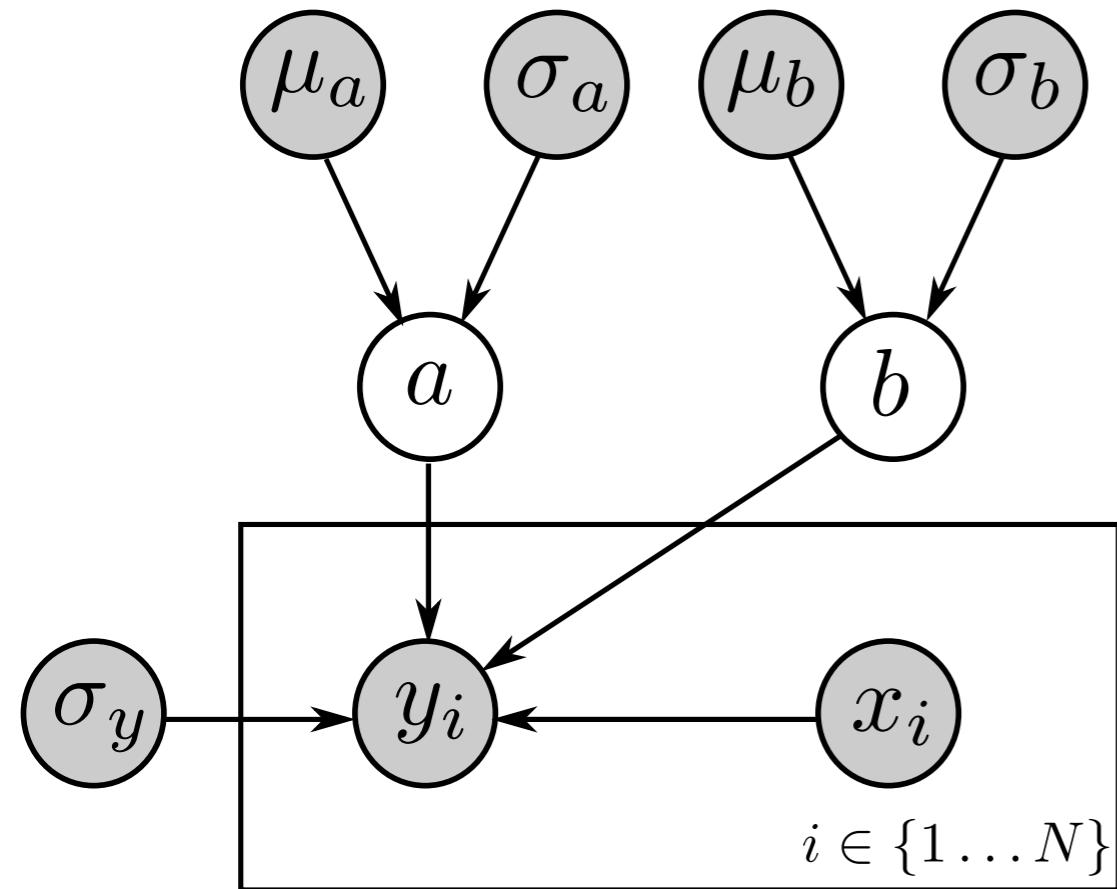
$$\mathbf{x} = \{x_1, x_2, \dots, x_N\}$$

$$a \sim \mathcal{N}(\mu_a, \sigma_a)$$

$$f(\mathbf{x}) = a\mathbf{x} + b$$

$$b \sim \mathcal{N}(\mu_b, \sigma_b)$$

$$\mathbf{y} \sim \mathcal{N}(a\mathbf{x} + b, \sigma_y)$$



# Generative model for linear regression in Edward

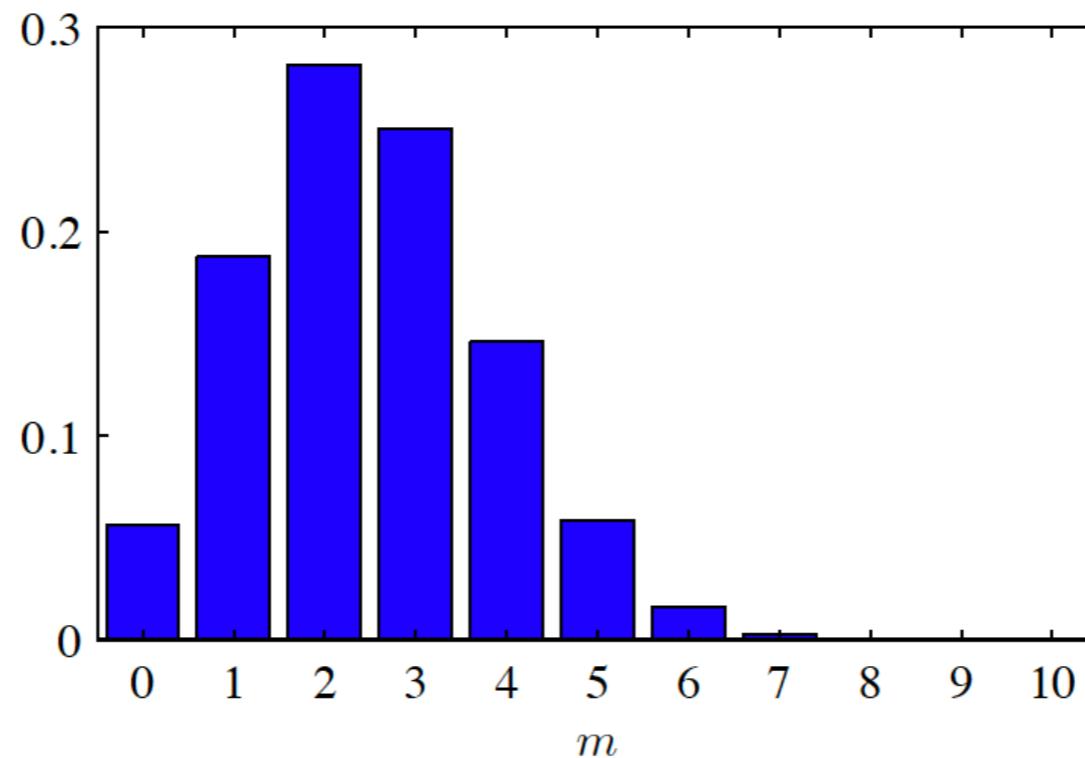
**09-Generative-linear-regression-edward.ipynb**

# Binomial distribution

$$\text{Bin}(k|n, p) = \binom{n}{k} p^k \cdot (1 - p)^{n-k} =$$

$$\text{Bin}(x_1, x_2 | p_1, p_2) = \frac{(x_1 + x_2)!}{x_1! x_2!} p_1^{x_1} \cdot p_2^{x_2}$$

$$p_1 + p_2 = 1$$



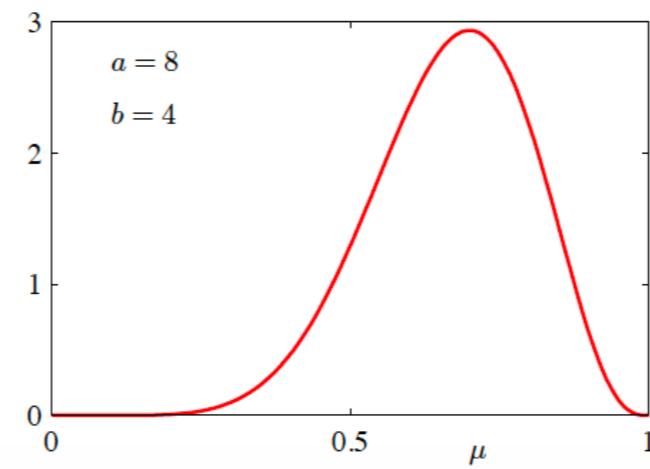
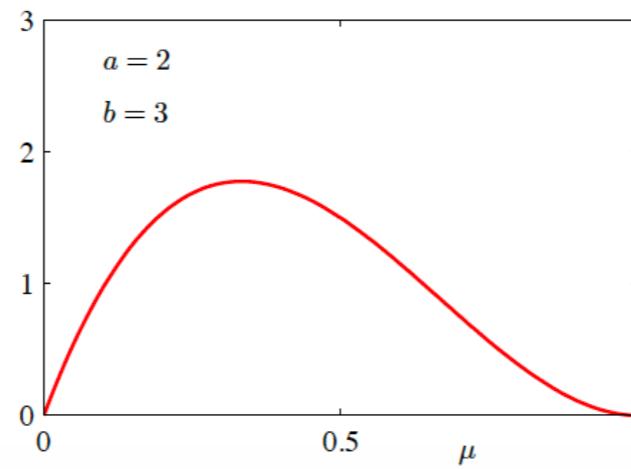
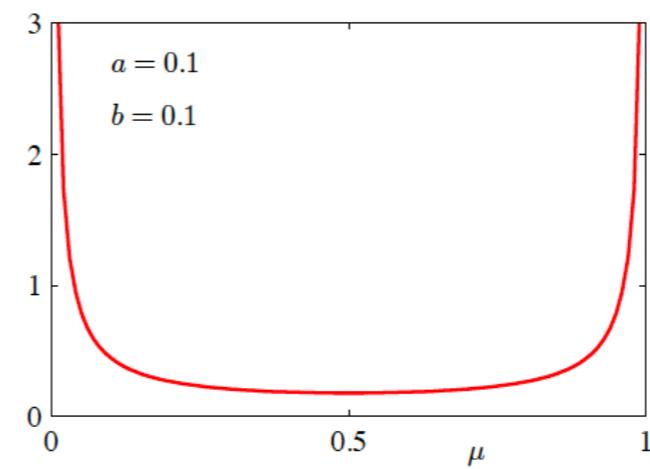
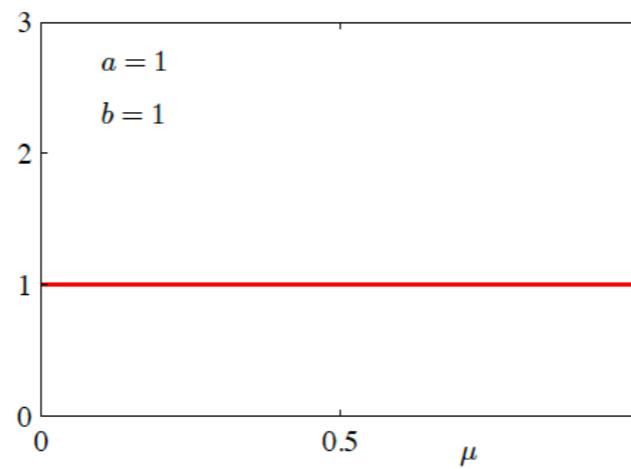
Example:  $n = 10, p = 0.25$

# Beta distribution

$$\text{Beta}(p_1, p_2 | \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} p_1^{\alpha-1} \cdot p_2^{\beta-1}$$

$$p_1 + p_2 = 1$$

$$\Gamma(x) = (x - 1)!$$



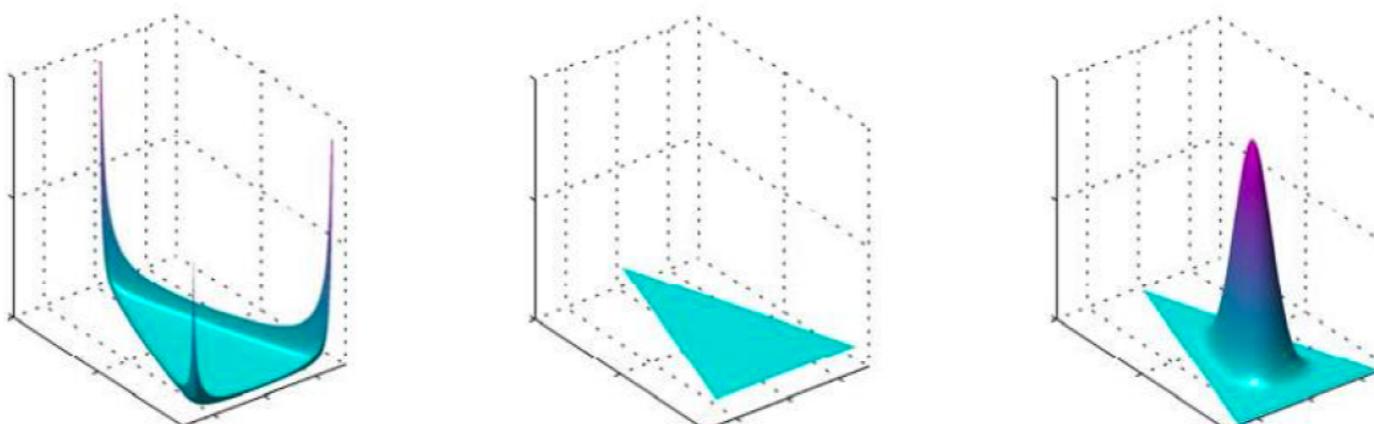
# Multinomial and Dirichlet distributions

Multinomial

$$\text{Mult}(x_1 \dots x_n | p_1 \dots p_n) = \frac{(\sum x_i)!}{\prod x_i!} \prod_{i=1}^n p_i^{x_i}$$

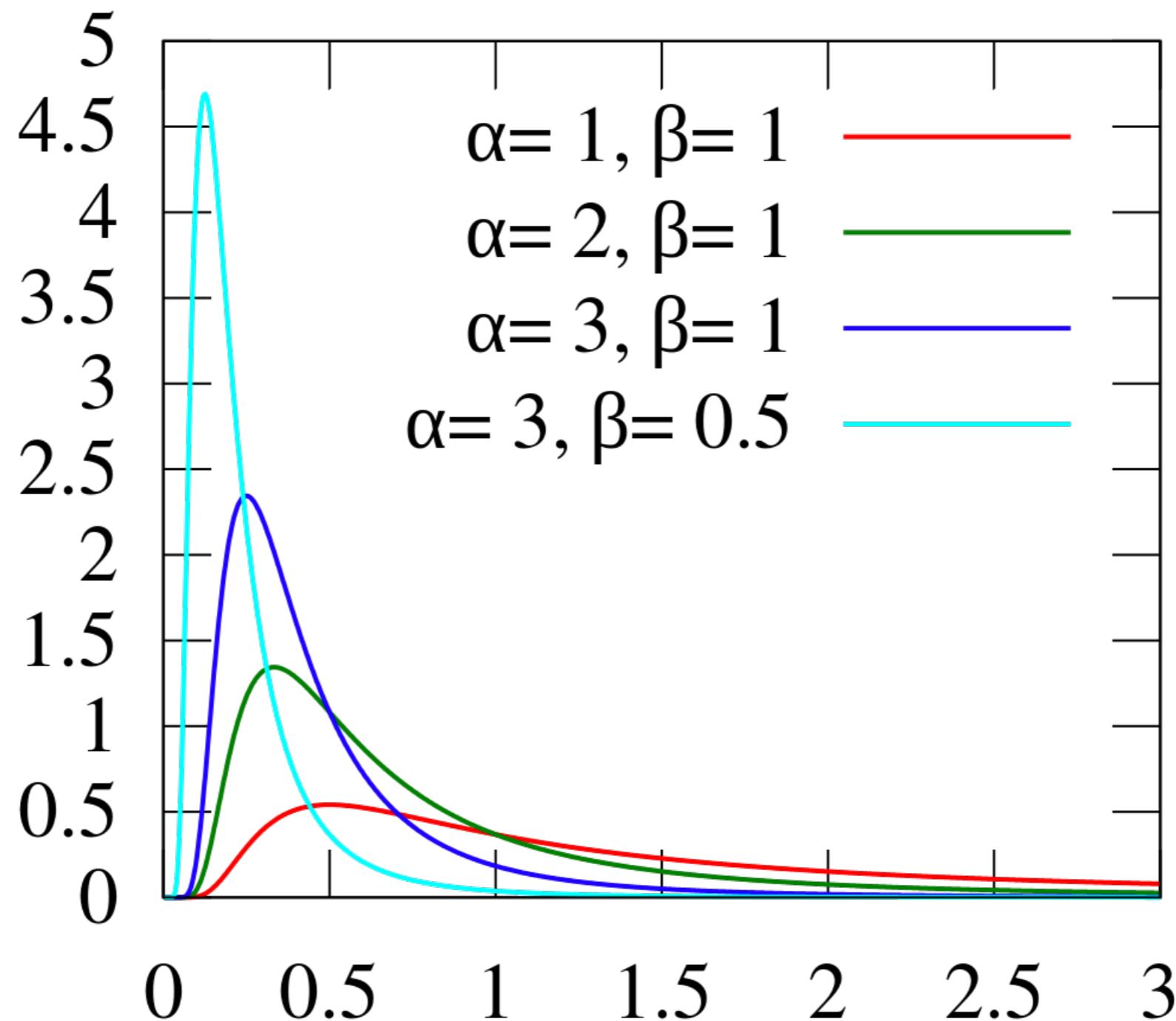
Dirichlet

$$\text{Dir}(p_1 \dots p_n | \alpha_1 \dots \alpha_n) = \frac{\Gamma(\sum \alpha_i)}{\prod \Gamma(\alpha_i)} \prod_{i=1}^n p_i^{\alpha_i - 1}$$

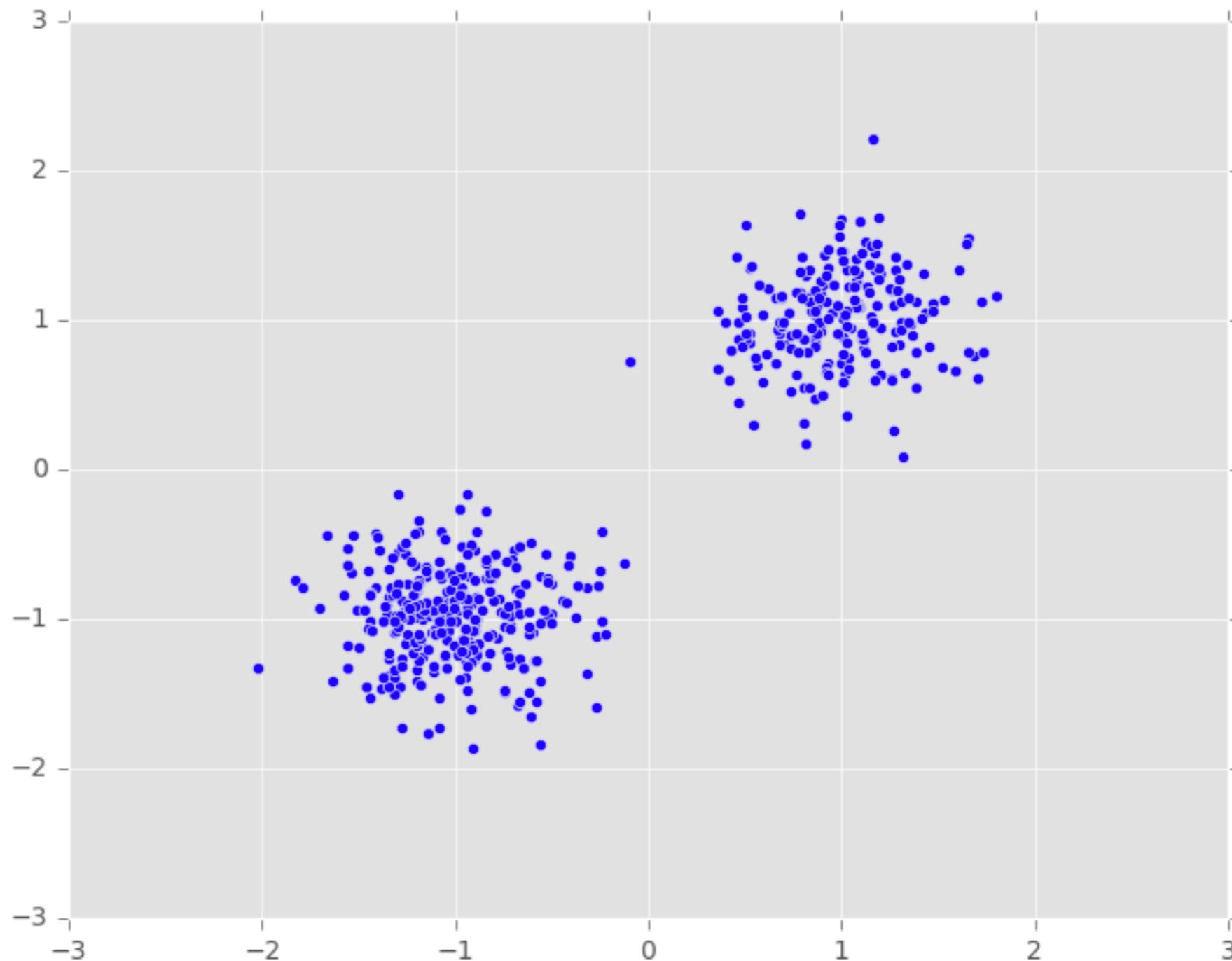


**Figure 2.5** Plots of the Dirichlet distribution over three variables, where the two horizontal axes are coordinates in the plane of the simplex and the vertical axis corresponds to the value of the density. Here  $\{\alpha_k\} = 0.1$  on the left plot,  $\{\alpha_k\} = 1$  in the centre plot, and  $\{\alpha_k\} = 10$  in the right plot.

# Inverse-gamma Distribution



# Clustering as gaussian mixtures

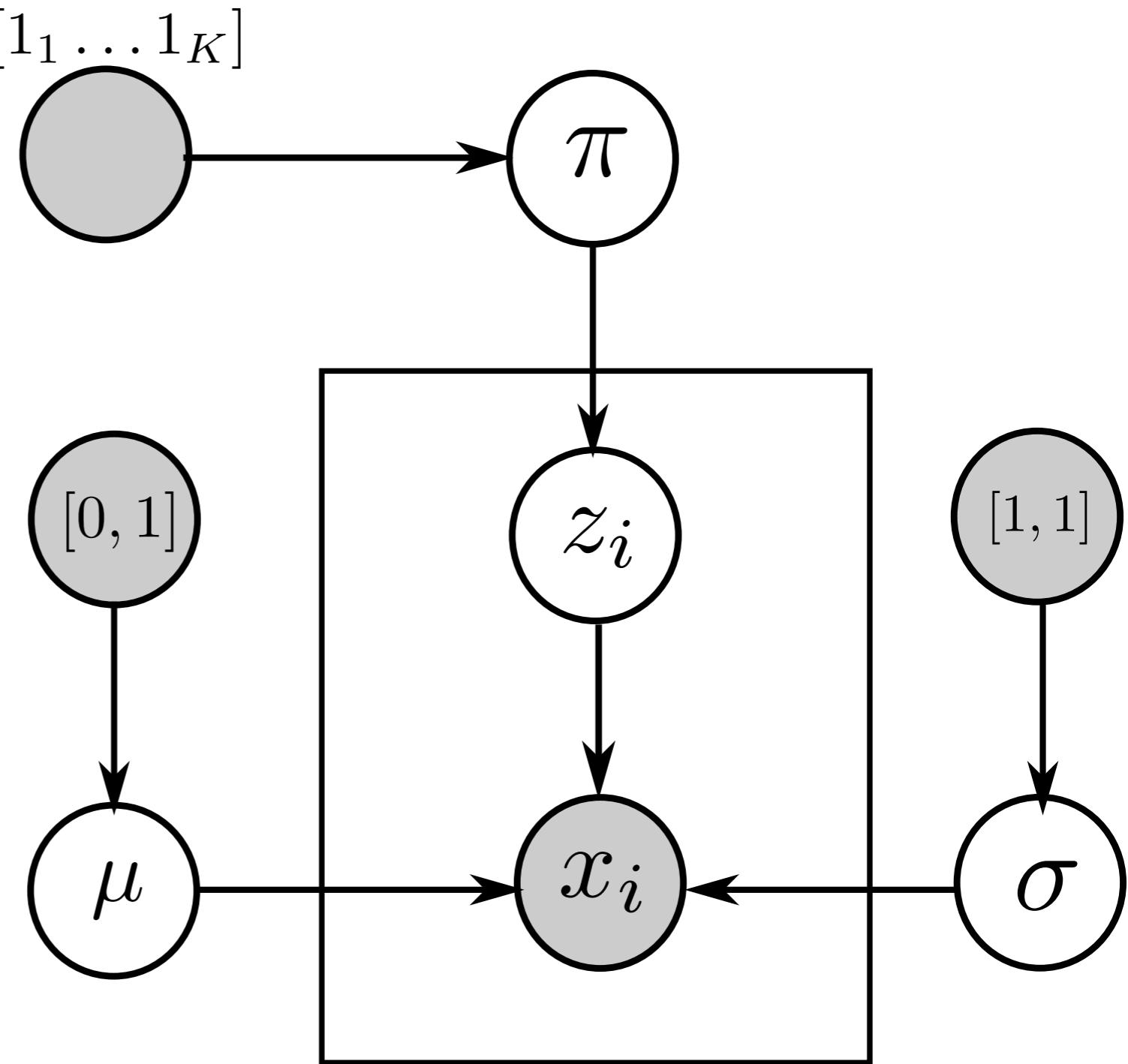


# Clustering as gaussian mixtures

$\pi \sim \text{Dirichlet}(1_1 \dots 1_K)$

$\mu_k \sim \mathcal{N}(0, 1)$

$\sigma_k^2 \sim \text{InverseGamma}(1, 1)$



# Gibbs sampling

- ① Initialize  $z_i : i \in 1, \dots, M$
- ② For  $\tau \in 1, \dots, T$ :
  - Sample  $z_1^{(\tau+1)} \sim P(z_1 | z_2^{(\tau)}, z_3^{(\tau)}, \dots, z_M^{(\tau)})$
  - Sample  $z_2^{(\tau+1)} \sim P(z_2 | z_1^{(\tau+1)}, z_3^{(\tau)}, \dots, z_M^{(\tau)})$
  - Sample  $z_3^{(\tau+1)} \sim P(z_3 | z_1^{(\tau+1)}, z_2^{(\tau+1)}, \dots, z_M^{(\tau)})$
  - ...
  - Sample  $z_M^{(\tau+1)} \sim P(z_M | z_1^{(\tau+1)}, z_2^{(\tau+1)}, \dots, z_{M-1}^{(\tau+1)})$

# Clustering as gaussian mixtures

**10-clustering-edward.ipynb**

# Topic Modeling

## Topics

gene 0.04  
dna 0.02  
genetic 0.01  
...

life 0.02  
evolve 0.01  
organism 0.01  
...

brain 0.04  
neuron 0.02  
nerve 0.01  
...

data 0.02  
number 0.02  
computer 0.01  
...

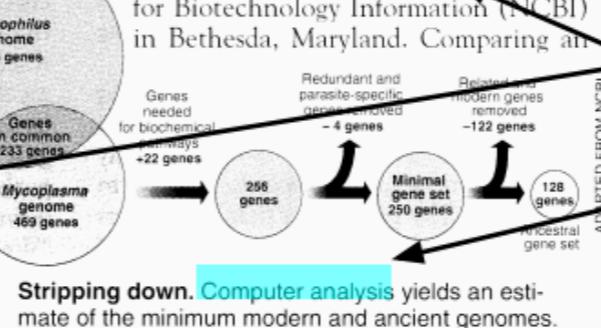
## Documents

### Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,\* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions

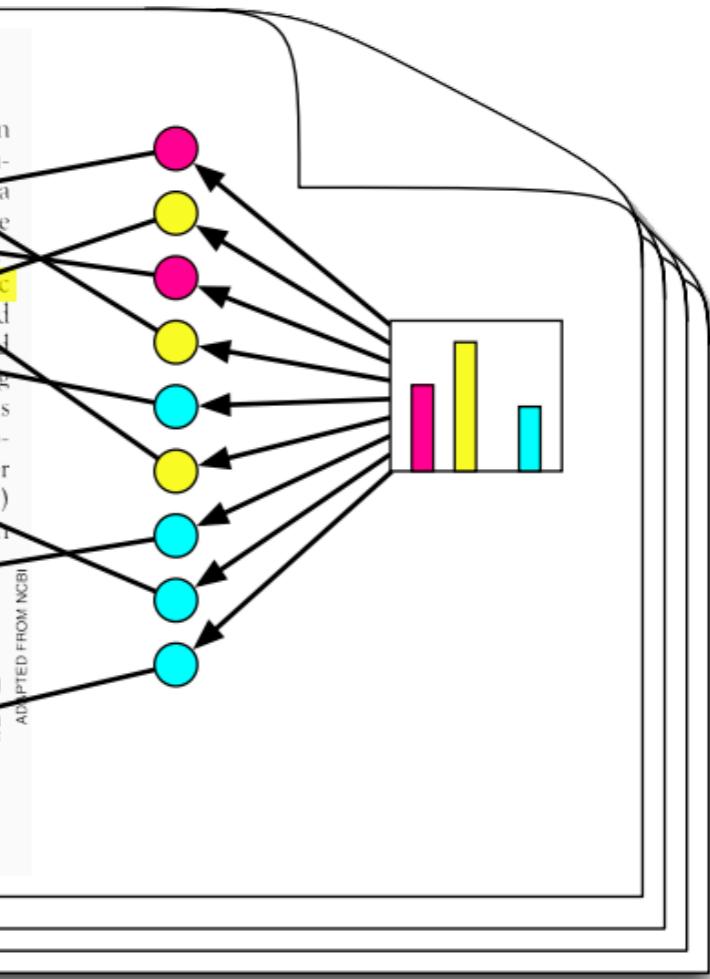
"are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an



\* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

SCIENCE • VOL. 272 • 24 MAY 1996

## Topic proportions & assignments



# Latent Semantic Analysis

**Topic modeling using LSA example:**

$$soccer = 1.8 * 'soccer' + 0.4 * 'ball' + 0.2 * 'FIFA' - 0.4 * 'tennis'$$

$$\text{doc} = 2.3 * soccer + 1.8 * sport + 0.9 * Europe + 0.8 * news$$

# Latent Semantic Analysis

**Doc<sub>1</sub>:** Machine learning helps people to understand data.

**Doc<sub>2</sub>:** Data can be understood using machine learning.

**Doc<sub>3</sub>:** *People can use machine learning for data understanding.*

	$\text{Doc}_1$	$\text{Doc}_2$	$\text{Doc}_3$
be	0	1	0
can	0	1	1
data	1	1	1
for	0	0	1
helps	1	0	0
learning	1	1	1
machine	1	1	1
people	1	0	1
to	1	0	0
understand	1	0	0
understanding	0	0	1
understood	0	1	0
use	0	0	1
using	0	1	0

1

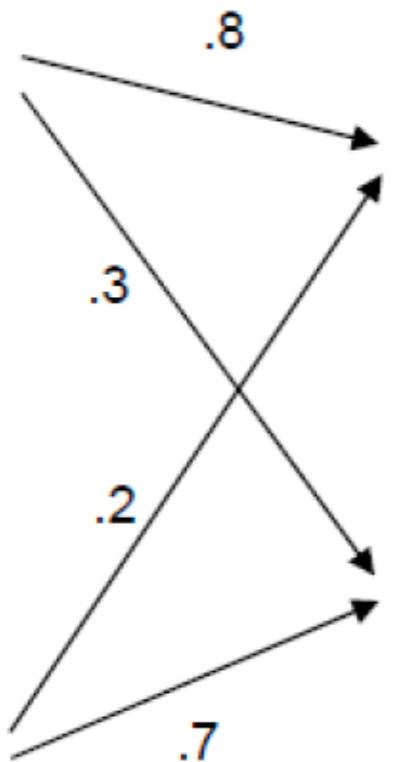
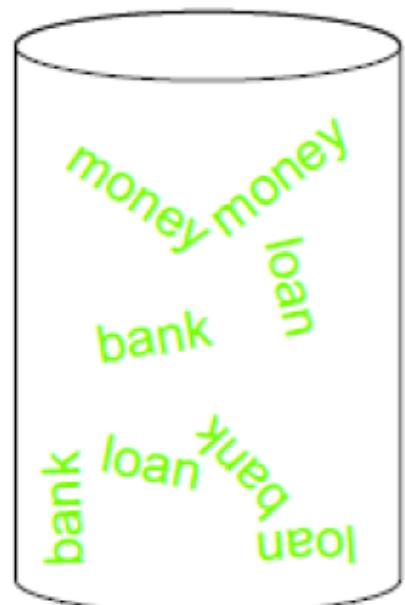
X

<b>TOP</b>	0	0	0
0	<b>IC</b>	0	0
0	0	<b>IMPO</b>	0
0	0	0	<b>RTAN CE</b>

x

4x4	DOCUMENTS		
T O P I C S			

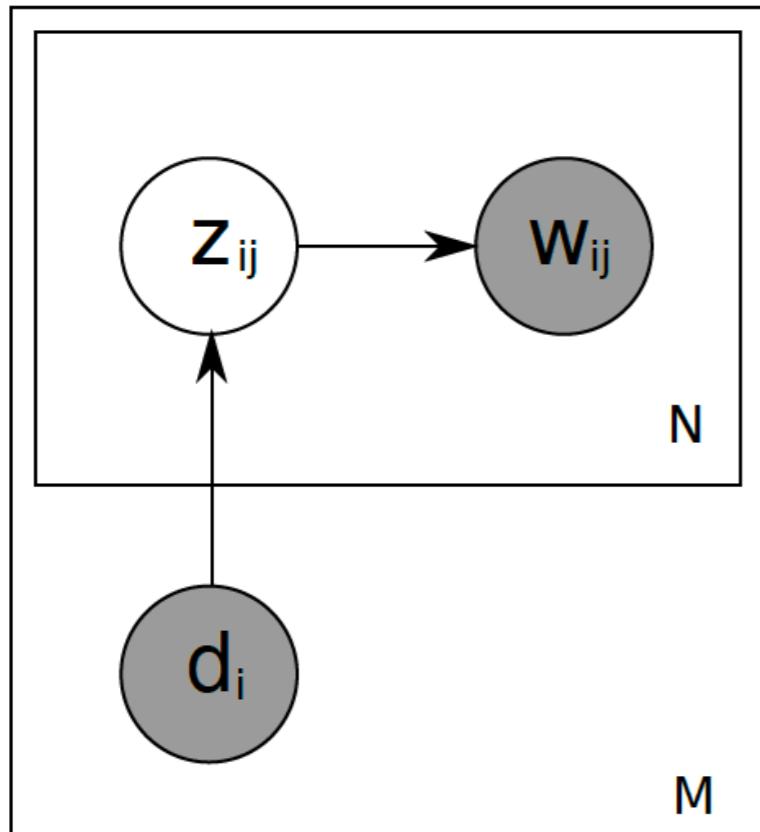
# Probabilistic Latent Semantic Analysis



DOCUMENT 1: money<sup>1</sup> bank<sup>1</sup> bank<sup>1</sup> loan<sup>1</sup> river<sup>2</sup> stream<sup>2</sup> bank<sup>1</sup> money<sup>1</sup> river<sup>2</sup> bank<sup>1</sup> money<sup>1</sup> bank<sup>1</sup> loan<sup>1</sup> money<sup>1</sup> stream<sup>2</sup> bank<sup>1</sup> money<sup>1</sup> bank<sup>1</sup> bank<sup>1</sup> loan<sup>1</sup> river<sup>2</sup> stream<sup>2</sup> bank<sup>1</sup> money<sup>1</sup> river<sup>2</sup> bank<sup>1</sup> money<sup>1</sup> bank<sup>1</sup> loan<sup>1</sup> bank<sup>1</sup> money<sup>1</sup> stream<sup>2</sup>

DOCUMENT 2: river<sup>2</sup> stream<sup>2</sup> bank<sup>2</sup> stream<sup>2</sup> bank<sup>2</sup> money<sup>1</sup> loan<sup>1</sup> river<sup>2</sup> stream<sup>2</sup> loan<sup>1</sup> bank<sup>2</sup> river<sup>2</sup> bank<sup>2</sup> bank<sup>1</sup> stream<sup>2</sup> river<sup>2</sup> loan<sup>1</sup> bank<sup>2</sup> stream<sup>2</sup> bank<sup>2</sup> money<sup>1</sup> loan<sup>1</sup> river<sup>2</sup> stream<sup>2</sup> bank<sup>2</sup> stream<sup>2</sup> bank<sup>2</sup> money<sup>1</sup> river<sup>2</sup> stream<sup>2</sup> loan<sup>1</sup> bank<sup>2</sup> river<sup>2</sup> bank<sup>2</sup> money<sup>1</sup> bank<sup>1</sup> stream<sup>2</sup> river<sup>2</sup> bank<sup>2</sup> stream<sup>2</sup> bank<sup>2</sup> money<sup>1</sup>

# Model of Probabilistic Latent Semantic Analysis



```
1: for  $i \in \{1, 2, \dots, N\}$  do
2:   for  $j \in \{1, 2, \dots, M\}$  do
3:     Choose a latent topic  $z_{ij}$  with probability  $P(z_{ij}|d_i)$ 
4:     Choose a word  $w_{ij}$  with probability  $P(w_{ij}|z_{ij})$ 
5:   end for
6: end for
```

Probabilities are computed from frequency analysis of words

Not a generative model (works for training data only)

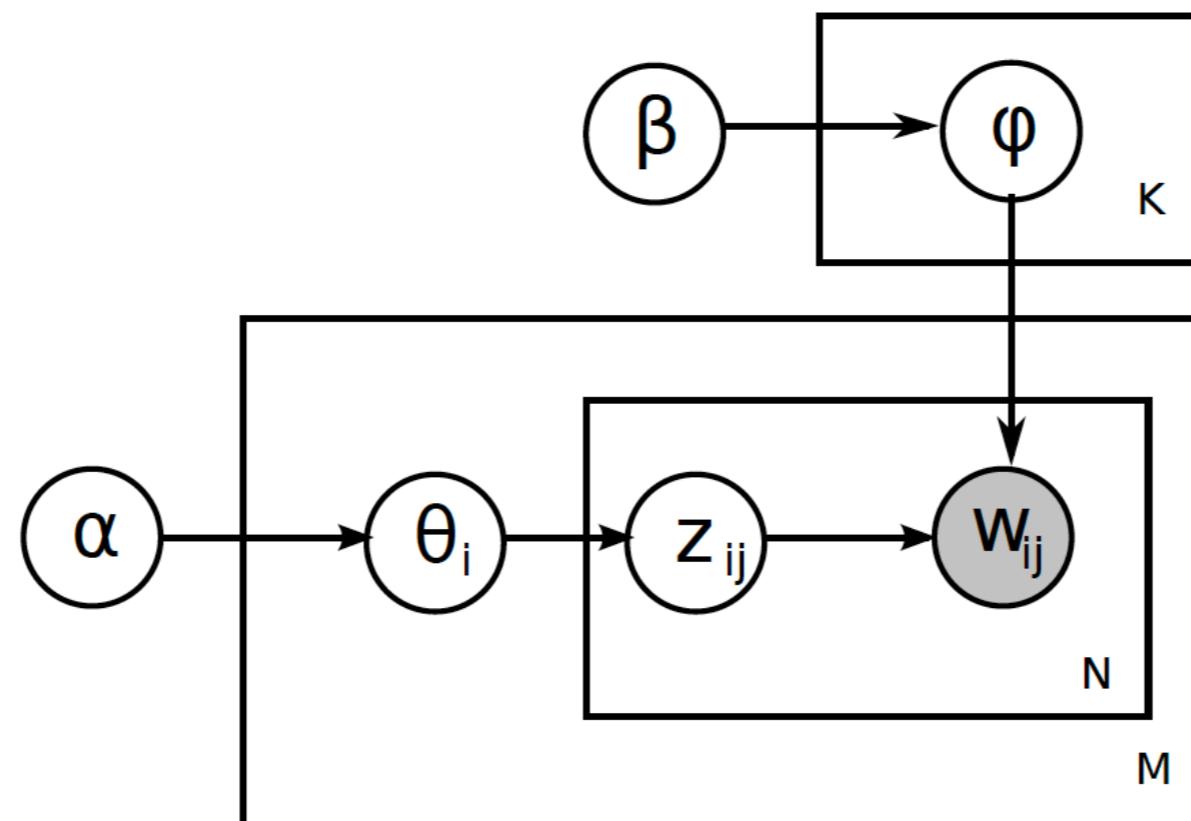
# Latent Dirichlet Allocation

For each document  $i \in 1 \dots M$  choose  $\theta_i \sim \text{Dir}(\alpha)$

For each word position  $j \in \dots N_i$  choose topic  $z_{i,j} \in 1 \dots K$ ,

$$z_{i,j} \sim \text{Mult}(\theta_i)$$

For each word position  $j$  choose word  $w_{i,j} \sim \text{Mult}(\varphi_{z_{i,j}})$



# Latent Dirichlet Allocation

## Topics

gene 0.04  
dna 0.02  
genetic 0.01  
...

life 0.02  
evolve 0.01  
organism 0.01  
...

brain 0.04  
neuron 0.02  
nerve 0.01  
...

data 0.02  
number 0.02  
computer 0.01  
...

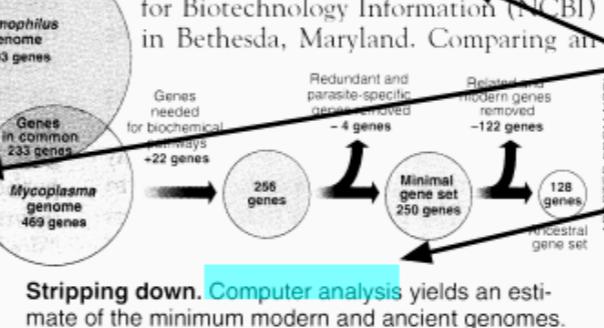
## Documents

### Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,\* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions

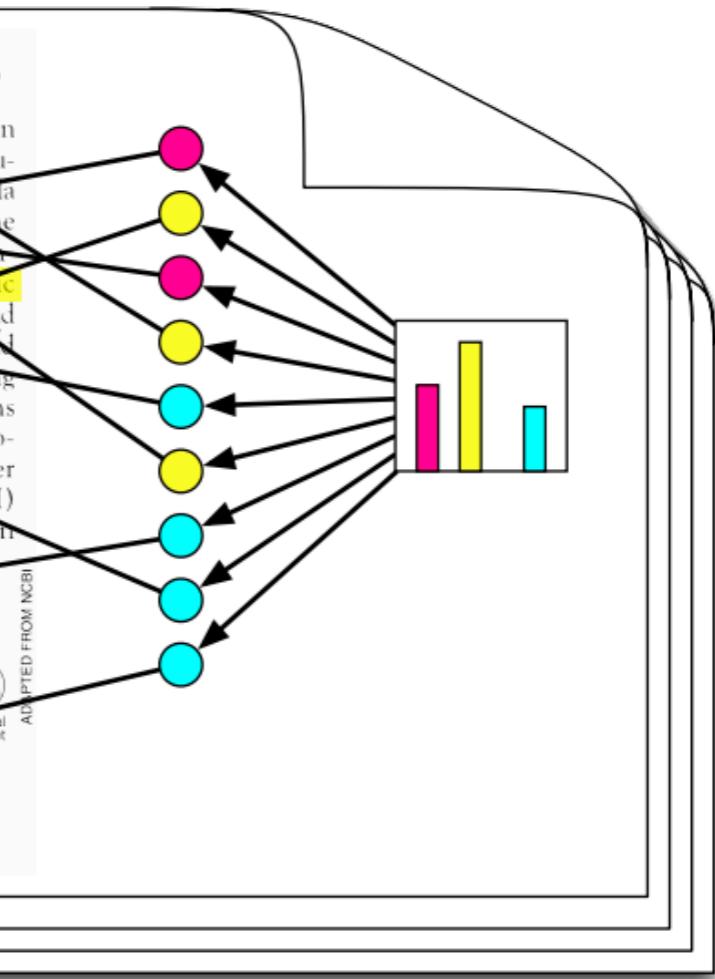
"are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing all



\* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

SCIENCE • VOL. 272 • 24 MAY 1996

## Topic proportions & assignments



# Topic modeling

**11\_Topic\_modeling.ipynb**

# Hate speech classification task

**12-Hate-speech-assignment.ipynb**

# Thank you for your attention

**e-mail:** jiri@mlcollege.com

**Web:** www.mlcollege.com

**Twitter:** @JiriMaterna

**Facebook:** <https://www.facebook.com/maternajiri>

**LinkedIn:** <https://www.linkedin.com/in/jirimaterna/>