Unsupervised NLP

Vsevolod Dyomkin

prj-nlp-1, 2018-04-26

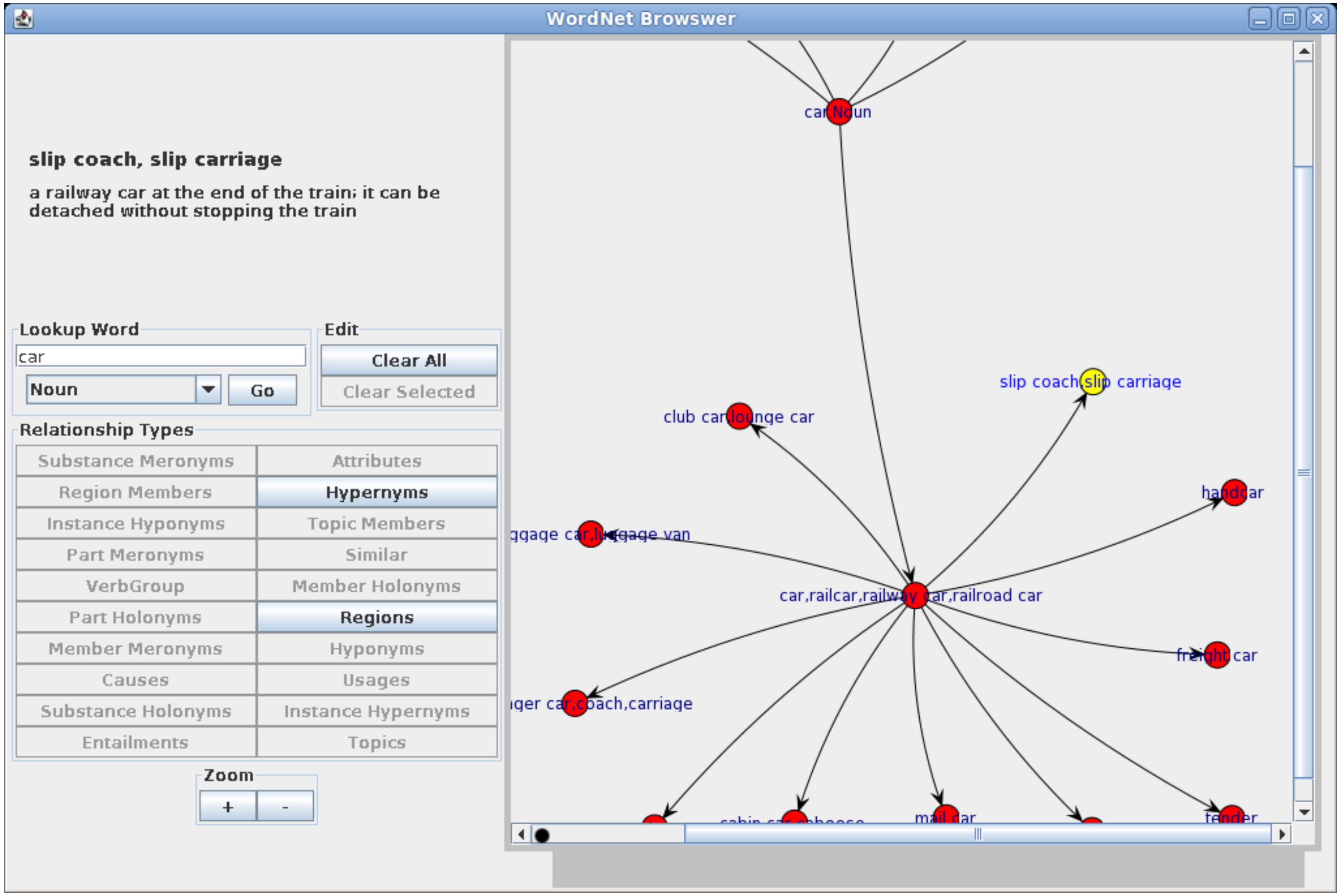
Approaches

* matrix factorization
* expectation maximization
* clustering

Graph-based Semantics

Question: how to model relationships between words?

Linguist's answer: build a graph



Word Similarity

Next question: now, how do we measure the intensity of those relations?

$$Sim(C1, C2) = 2 * Max(C1, C2) - SP$$

$$Sim_{Rod}(C1^p, C2^q) = W_w S_w(C1^p, C2^q) + W_u S_u(C1^p, C2^q) + W_n S_n(C1^p, C2^q))$$

$$Sim_{Resnik}(C1, C2) = \frac{2 * \ln((p_{mis}(C1, C2))}{\ln(p(c1)) + \ln(p(c2))}$$

$$Sim_{Knappe}(C1, C2) = p * \frac{|Ans(C1) \cap Ans(C2)|}{|Ans(C1)|} + (1 - p) * \frac{|Ans(C1) \cap Ans(C2)|}{|Ans(C2)|}$$

$$Sim_{Zhou}(C1, C2) = 1 - k\left(\frac{\ln(len(C1, C2) + 1}{\ln(2 * (deep_{max} - 1))}\right) - (1 - k) * ((IC(C1) + IC(C2) - 2 * IC(lso(C1, C2))/2)$$

$$Sim_{Resnik}(C1, C2) = -ln(p_{mis}(C1, C2))$$

$$Sim_{tvsk}(C1, C2) = \frac{|C1 \cap C2|}{|C1 \cap C2| + \propto |C1 - C2| + (\propto - 1)|C2 - C1|}$$

$$Sim_{LC}(C1, C2) = -\log\left(\frac{length}{2.D}\right)$$

$$Sim_{HSO}(C1, C2) = C - SP - k * d$$

$$Sim_{wup}(C1, C2) = \frac{2 * N}{N1 + N2 + 2 * N}$$

https://arxiv.org/pdf/1310.8059.pdf

Many Faces of Similarity

- dog -- cat
- dog -- poodle
- dog -- animal
- dog -- bark
- dog -- leash

- dog -- chair same POS
- dog -- dig edit distance
- dog -- god same letters
- dog -- fog rhyme
- dog -- 6op shape

Distributional Semantics

Distributional hypothesis:
”You shall know a word by the company it keeps”
--John Rupert Firth

Explicit word graph representation
Number of nonzero dimensions:
max:474234, min:3, mean:1595, median:415

Co-occurance Matrix

- I like deep learning.
- I like NLP.
- I enjoy flying.

counts	I	like	enjoy	deep	learning	NLP	flying	.
I	0	2	1	0	0	0	0	0
like	2	0	0	1	0	1	0	0
enjoy	1	0	0	0	0	0	1	0
deep	0	1	0	0	1	0	0	0
learning	0	0	0	1	0	0	0	1
NLP	0	1	0	0	0	0	0	1
flying	0	0	1	0	0	0	0	1
.	0	0	0	0	1	1	1	0

PPMI Matrix

Dan Jurafsky

$$pmi_{ij} = \log_2 \frac{p_{ij}}{p_{i*}p_{*j}}$$

p(w,context)

	computer	data	pinch	result	sugar	**p(w)**
apricot	0.00	0.00	0.05	0.00	0.05	0.11
pineapple	0.00	0.00	0.05	0.00	0.05	0.11
digital	0.11	0.05	0.00	0.05	0.00	0.21
information	0.05	0.32	0.00	0.21	0.00	0.58
p(context)	0.16	0.37	0.11	0.26	0.11	

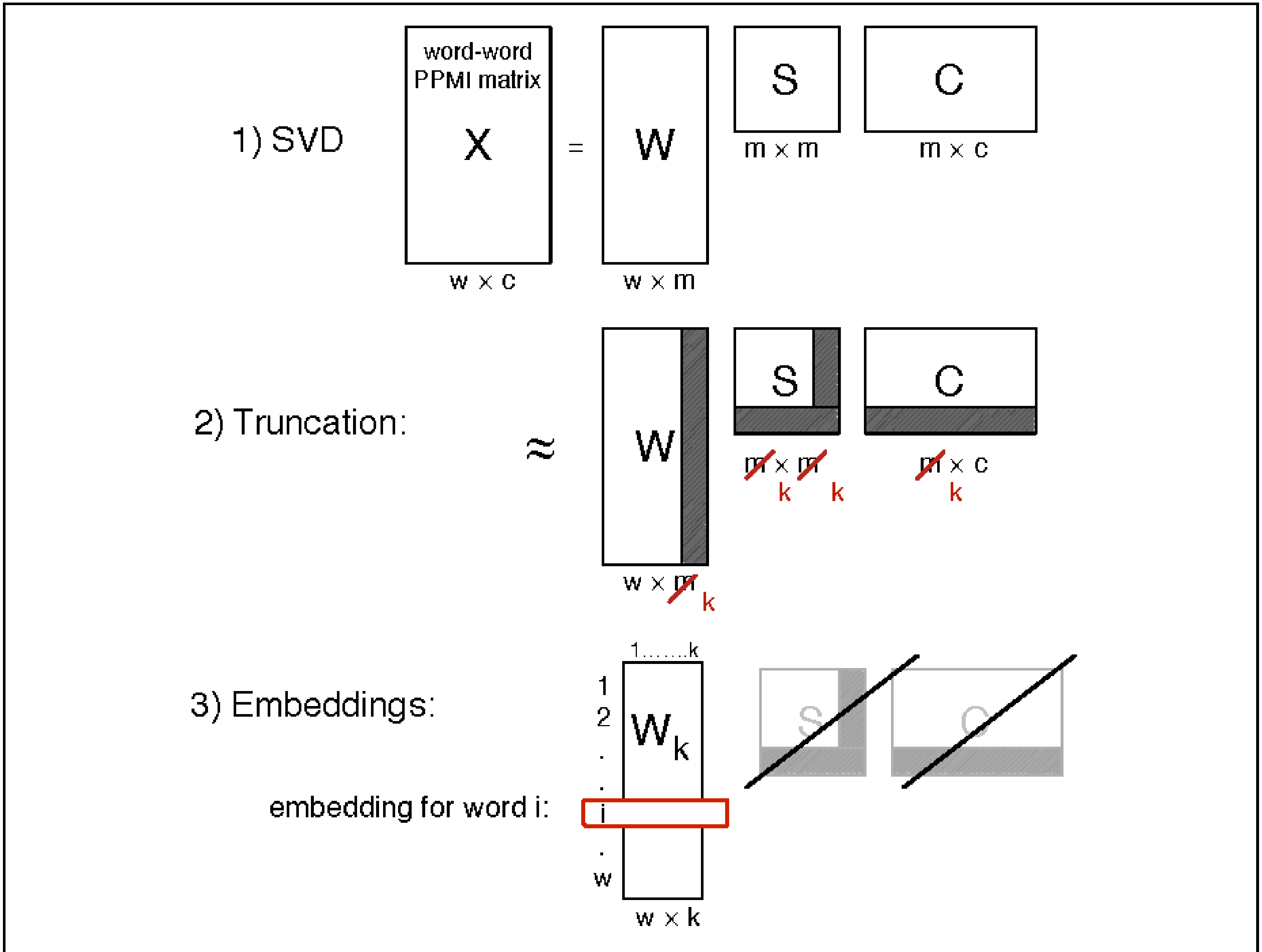
- pmi(information,data) = $\log_2$ (.32 / (.37*.58)) = .58

(.57 using full precision)

PPMI(w,context)

	computer	data	pinch	result	sugar
apricot	-	-	2.25	-	2.25
pineapple	-	-	2.25	-	2.25
digital	1.66	0.00	-	0.00	-
information	0.00	0.57	-	0.47	-

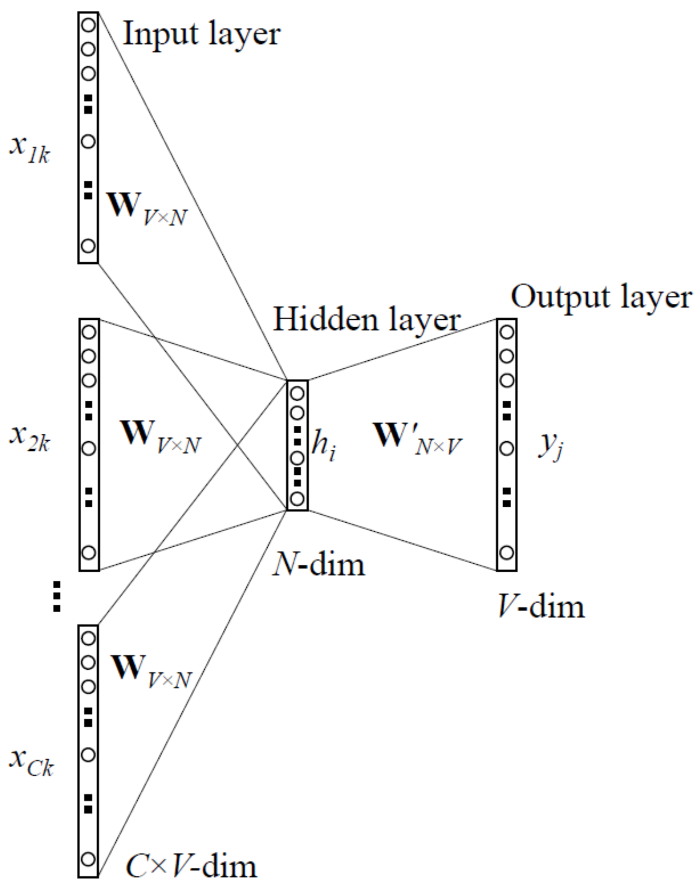
SVD



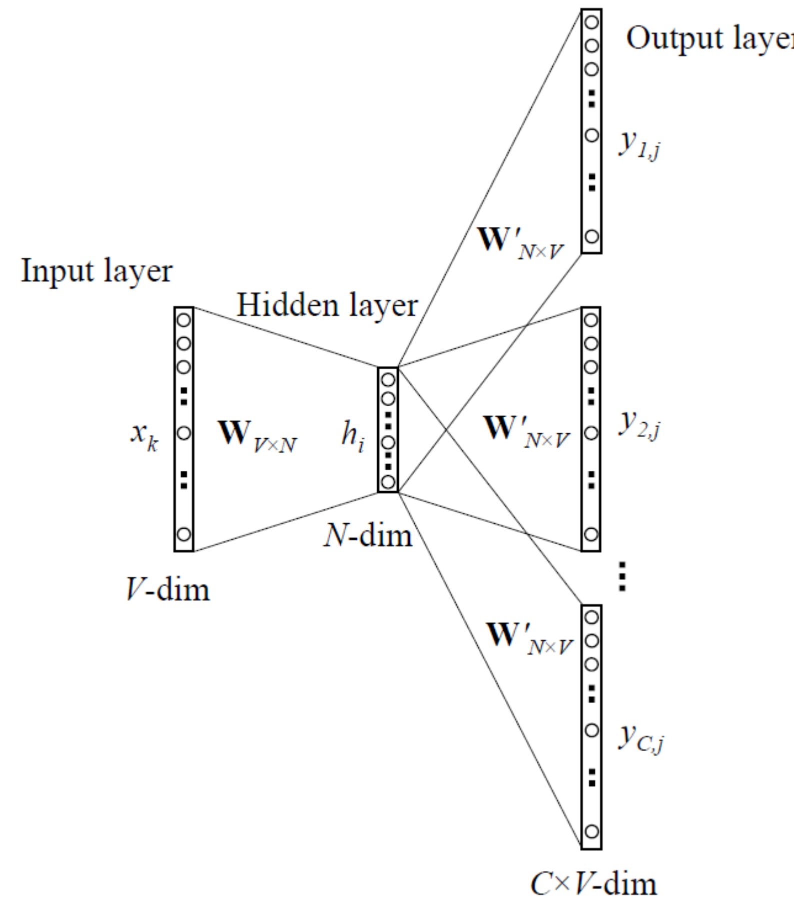
1) SVD
word-word
PPMI matrix
X
w × c
=
W
w × m
S
m × m
C
m × c
2) Truncation:
≈
W
w × m k
S
m × m
k k
C
m × c
k
3) Embeddings:
1.......k
1
2
.
.
i
.
w
W_k
w × k
embedding for word i:
S
C

word2vec

CBOW



SGNS



http://u.cs.biu.ac.il/~yogo/cvsc2015.pdf

Word Vectors Evaluation

* extrinsic
* intrinsic
  - relatedness
  - analogy
  - categorization
  - selectional preference

https://aclanthology.info/pdf/D/D15/D15-1036.pdf

fasttext

Extension to SGNS to take into account subword information (character ngrams):

The word “where” is reprsented as a sum of representations of “<where>”, “<wh”, “whe”, “her”, “ere”, “re>”

https://arxiv.org/pdf/1607.04606.pdf

ConceptNet Numberbatch

~~Wordnet~~ConceptNet strikes back

The current SOTA vectors due to
* vector ensemble using ConceptNet to merge vectors
* OOV handling

https://blog.conceptnet.io/2016/05/25/conceptnet-numberbatch-a-new-name-for-the-best-word-embeddings-you-can-download/

https://blog.conceptnet.io/2017/03/02/how-luminoso-made-conceptnet-into-the-best-word-vectors-and-won-at-semeval/

NNSE

Non-Negative Sparse Embedding
- using non-negative matrix factorization
- and sparse coding

http://talukdar.net/papers/nnse_coling12.pdf

CNNSE (Compositional):
- add composition constraint to training

http://www.aclweb.org/anthology/N15-1004

word2gauss

Each word is represented as a multivariate Gaussian: a probability P[i] — a K-dimensional Gaussian parameterized by mean mu and co-variance matrix Sigma:

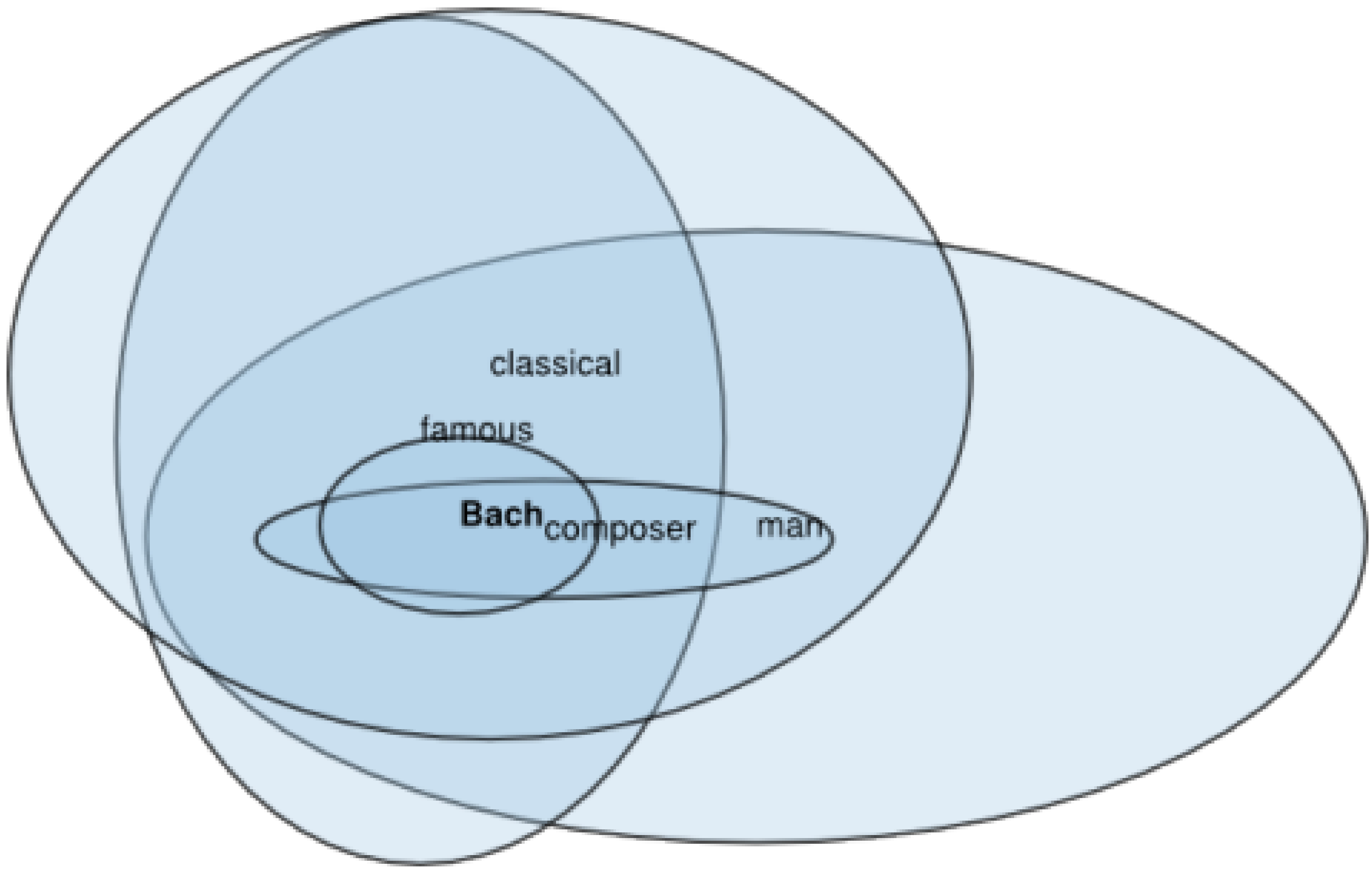
```
P[i] ~ N(x; mu[i], Sigma[i])
```

The mean is a vector of length K and in the most general case Sigma[i] is a (K, K) matrix. 2 approximations to simplify Sigma:

- diagonal - a vector length K
- spherical - a float

https://arxiv.org/pdf/1412.6623.pdf
https://github.com/seomoz/word2gauss

word2gauss

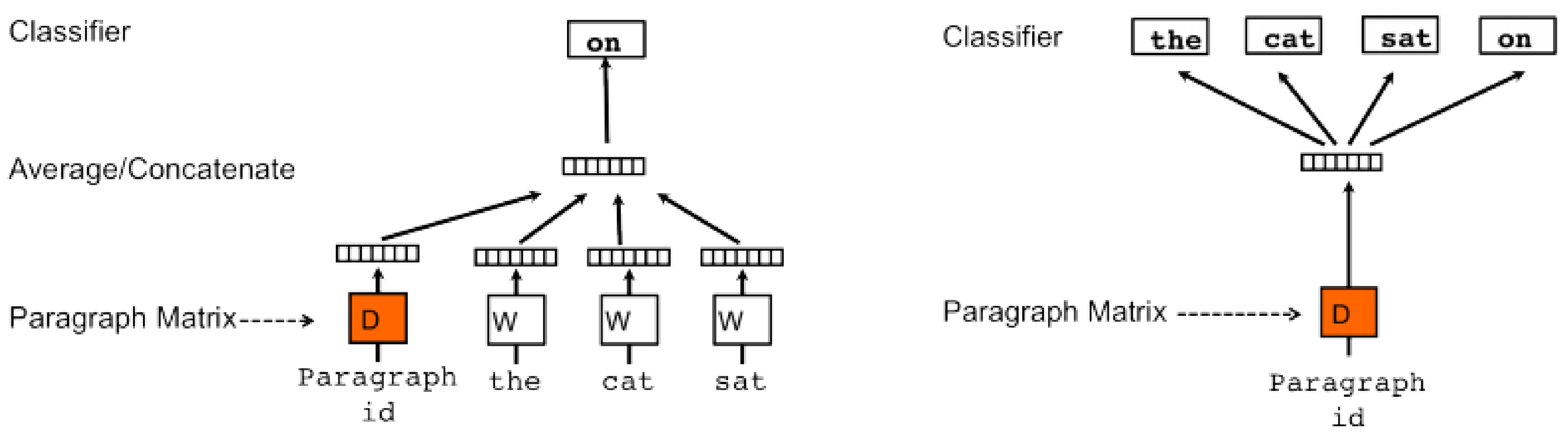


doc2vec

Question: how to represent phrases/sentences/paragraps/documents with dense vectors?

Default answer: average the word vectors
Alternative: “paragraph vectors”

PV-DM PV-DBOW



https://cs.stanford.edu/~quocle/paragraph_vector.pdf

Skip-thoughts

Train an encoder-decoder model where the encoder maps the input sentence to a sentence vector and the decoder generates the sentences surrounding the original sentence. Similar to the skip-gram model in the sense that surrounding sentences are used to learn sentence vectors.

https://arxiv.org/abs/1506.06726

https://www.intelnervana.com/building-skip-thought-vectors-document-understanding/

Universal Sentence Encoder

Specifically targeted at transfer learning tasks

https://arxiv.org/pdf/1803.11175.pdf

Dense Representations Recap

Key idea: transition from sparse (BoW) to dense vectors and maximize the vectors' affinity to some relation in the process.

Pros:

- capture those relations
- easier to compute with (possible to use as input for neural nets)

Cons:

- expensive to compute the vectors themselves

Topic Modelling

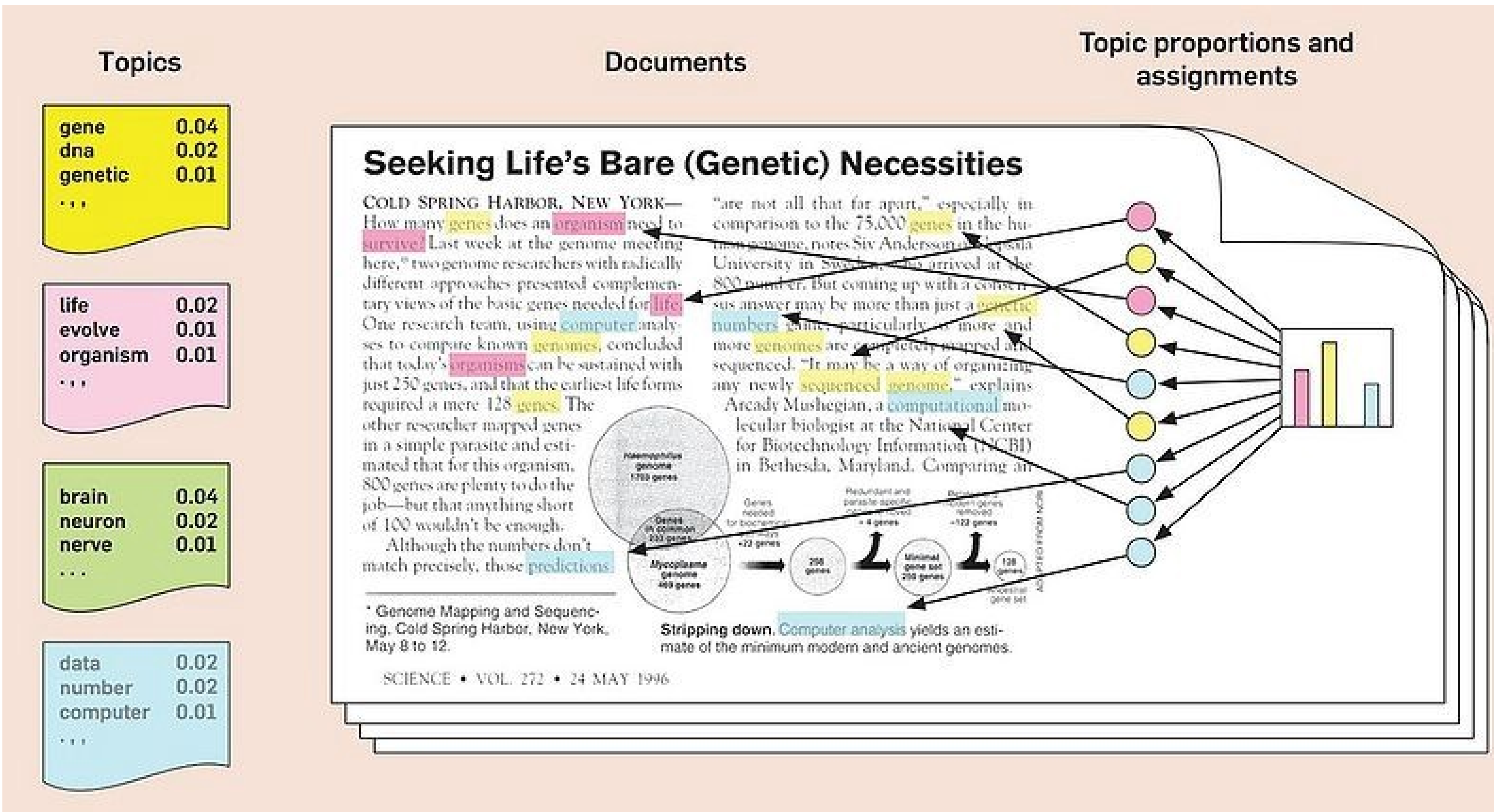
A multi-class whole-text classification/ranking problem.

A mostly **unsupervised** problem.

Latent Semantic Indexing

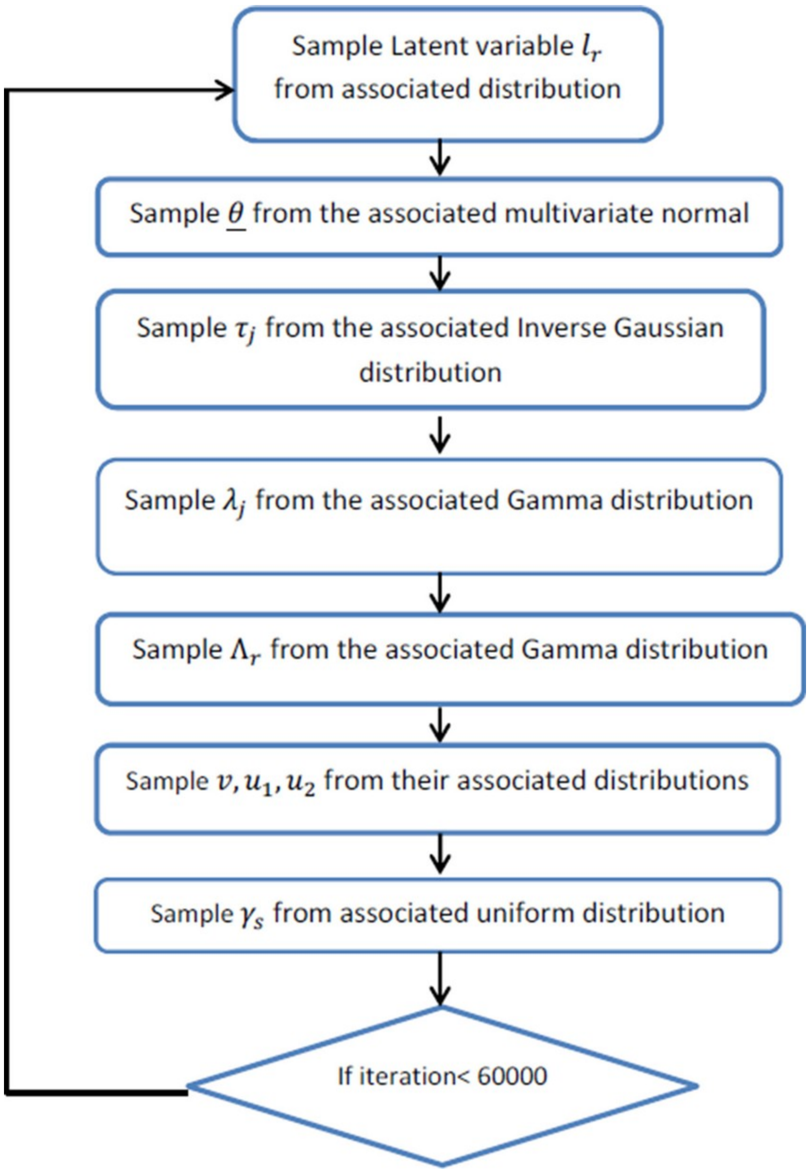
Factorization of the word-document matrix using SVD and leaving the top-N eigen values.

Latent Dirichlet Allocation



http://www.cl.cam.ac.uk/teaching/1213/L101/clark_lectures/lect7.pdf

Gibbs Sampling



https://stats.stackexchange.com/questions/10213/can-someone-explain-gibbs-sampling-in-very-simple-words

Kullback—Leibler & Jensen-Shannon Divergencies

KL-divergence:

$$D_{\mathrm{KL}}(P\|Q) = \sum_i P(i)\,\log\frac{P(i)}{Q(i)}.$$

JS-divergence:

$$\mathrm{JSD}(P \parallel Q) = \frac{1}{2}D(P \parallel M) + \frac{1}{2}D(Q \parallel M)$$

Anchor Words

Problem of LSI/LDA: hard to interpret topics.

Alternative factorization to SVD: Non-negative matrix factorization (NMF).

https://cs.stanford.edu/~rishig/courses/ref/l9b.pdf

Read More

word2vec parameter learning explained:
https://arxiv.org/pdf/1411.2738v3.pdf
https://blog.acolyer.org/2016/06/01/distributed-representations-of-sentences-and-documents/
https://github.com/RaRe-Technologies/gensim/blob/develop/docs/notebooks/doc2vec-IMDB.ipynb

doc2vec empirical evaluation:
https://arxiv.org/pdf/1607.05368.pdf

Word vectors & semantic lexicons:
https://arxiv.org/pdf/1411.4166.pdf

LDA:
http://pages.cs.wisc.edu/~jerryzhu/cs769/latent.pdf
https://www.youtube.com/watch?v=3mHy4OSyRf0
https://www.quora.com/What-is-an-intuitive-explanation-of-the-Dirichlet-distribution