Probabilistic approach for embedding arbitrary features of text

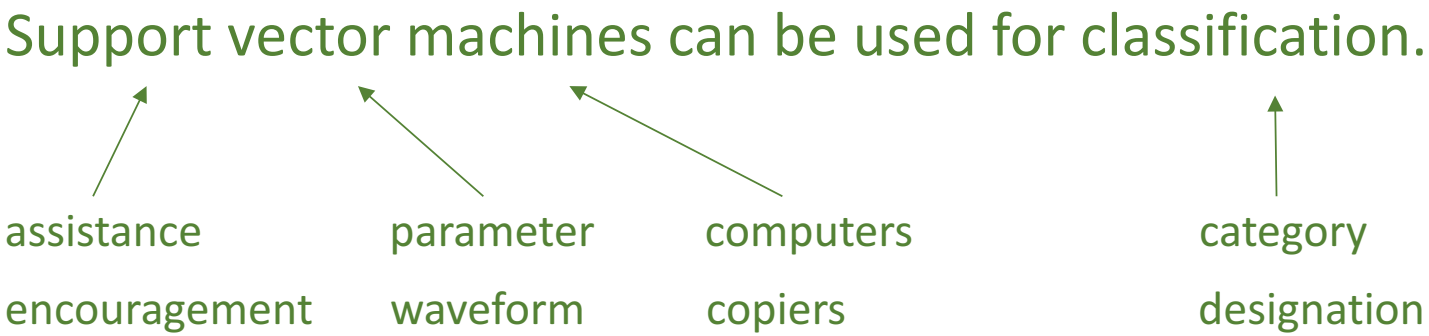
Anna Potapenko

National Research University Higher School of Economics

AIST 2018

Introduction

What are the smallest units of sense?



- Words
- Word senses
- N-grams
- Sentences
- Sub-word pieces

- Embedding components
- Topics from topic modeling
- Arora's atoms (dictionary learning)
- …

Sentence by averaging its words (n-grams)

- S. Arora et al. Simple but tough-to-beat baseline for sentence embeddings, ICLR 2017.
  - According to their experiments, beats Skip-thought and other LSTM-based models.

- M. Pagliardini et al. Unsupervised Learning of Sentence Embeddings using Compositional n-Gram Features. NAACL 2018.
  - Learns on word-in-sentence occurrences.

- L. Wu et al. StarSpace: Embed All The Things! AAAI 2018.
  - Learns on similar sentence pairs (there are also other modes).

Intersection-averaging procedure

Embeddings as soft topic assignments

Let us define embedding of an object as a vector of topic probabilities for this object.

E.g. consider a word embeddings:

$$\phi_w = [\, p(t_1|w), \ldots, p(t_k|w) \,]$$

Note, that the components fall in range [0, 1] and sum into 1.

We will also consider unconditional distribution of topics in the corpus:

$$\tau = [\, p(t_1), \ldots, p(t_k) \,]$$

PLSA topic model revisited

E-step: soft intersection of word and document embeddings:

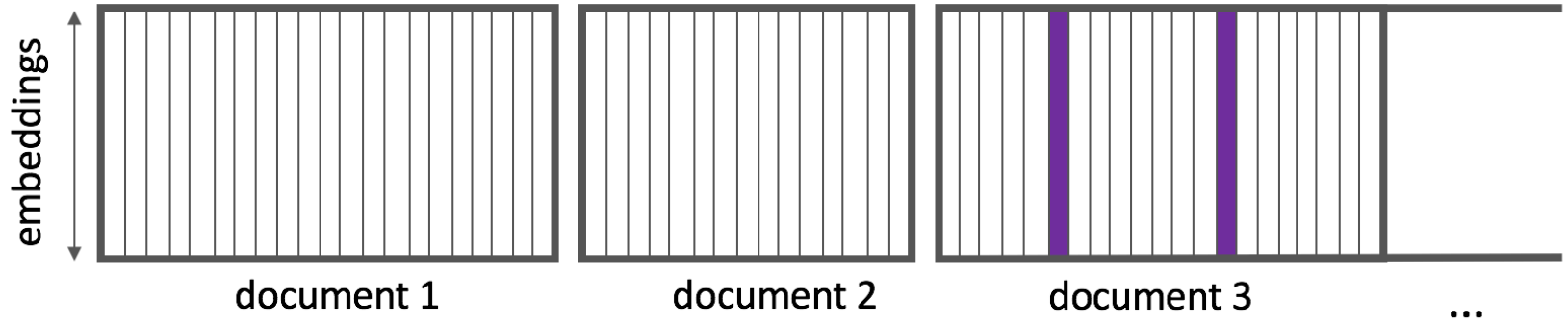
$$\psi_i = \frac{1}{Z_i}\, \tau^{-1} \circ \phi_{w_i} \circ \theta_{d_i}$$

M-step: averaging of word-in-document embeddings:

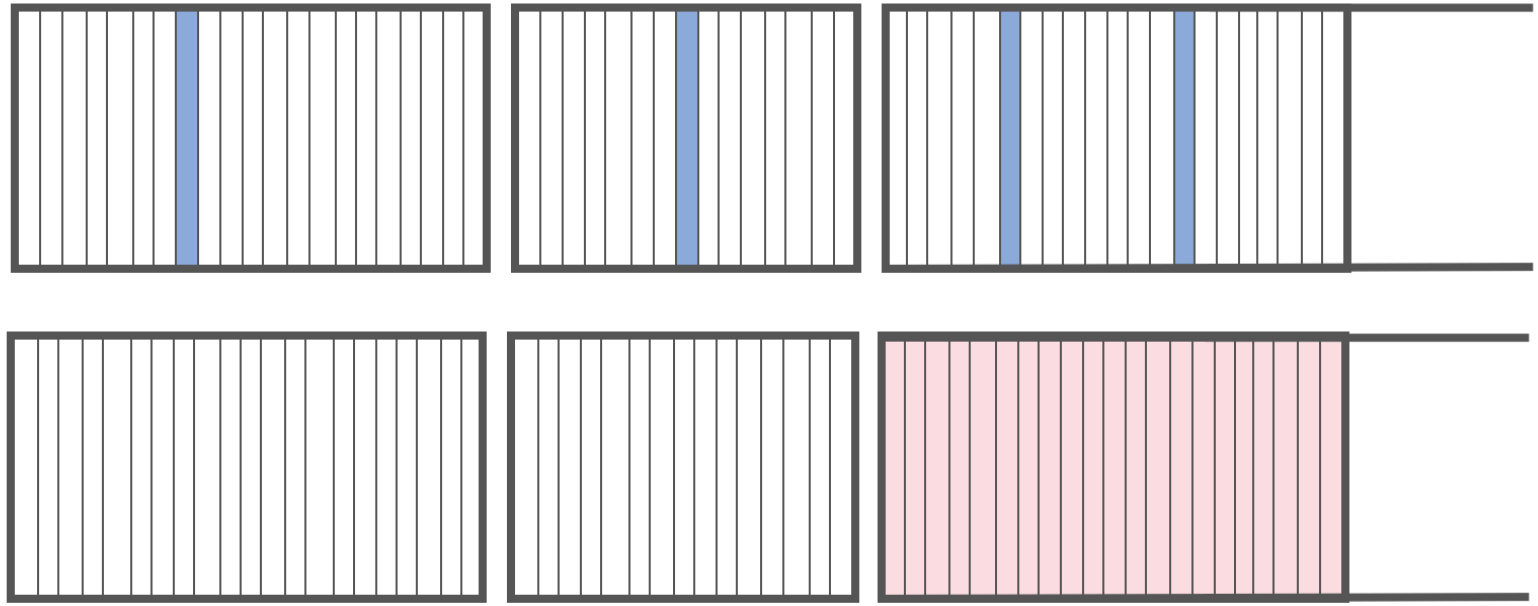
$$\phi_w = \underset{i:w_i=w}{mean}\, \psi_i; \qquad \theta_d = \underset{i:d_i=d}{mean}\, \psi_i.$$

where i runs through all positions in the corpus.

E-step:



M-step:



M-step features define the smallest embedded units. Let’s choose them smartly!

Our approach: how to embed multiple features

Topic-conditional independence assumption:

$$p(w,a,d) = \sum_t p(w,a,d|t)p(t) = \sum_t p(w|t)p(a|t)p(d|t)p(t)$$

E-step:

$$\psi_i = \frac{1}{Z}\,\tau^{-2} \circ \phi_w \circ \zeta_a \circ \theta_d$$

M-step:

$$\phi_w = \underset{i:w_i=w}{mean}\ \psi_i; \quad \zeta_a = \underset{i:a_i=a}{mean}\ \psi_i; \quad \theta_d = \underset{i:d_i=d}{mean}\ \psi_i.$$

NLP position-based features

- Token annotation: POS-tag, grammar role, reference to WordNet node, etc.
- Feature of a token: capitalized, ends with -ing, bold, etc.
- Aligned modality: e.g. author of each token in dialogues data.
- Context word: a word contained in the window of the position.
- …

Local contexts: a variety of options

Skip-Gram and CBOW

Skip-Gram:

$$p(c_{i-h}, \ldots, c_{i+h}|w_i) = \prod_{j \in H_i} p(c_j|w_i) = \prod_{j \in H_i} \frac{\exp \langle c_j, w_i \rangle}{\sum_c \exp \langle c, w_i \rangle} = \frac{1}{Z_i} \prod_{j \in H_i} \exp \langle c_j, w_i \rangle$$

CBOW:

$$p(w_i|c_{i-h}, \ldots, c_{i+h}) = \frac{\exp \left\langle w_i, \sum_{j \in H_i} c_j \right\rangle}{\sum_w \exp \left\langle w, \sum_{j \in H_i} c_j \right\rangle} = \frac{1}{Z'_i} \prod_{j \in H_i} \exp \langle w_i, c_j \rangle$$

What is different: partition functions; stochastic gradient descent samples.

WNTM and BTM

Again, assume independence of context words:

$$p(c_{i-h}, \ldots, c_{i+h}|w_i) = \prod_{j \in H_i} p(c_j|w_i) = \prod_{j \in H_i} \sum_t p(c_j|t)p(t|w_i)$$

Word Network Topic Model (WNTM):

$$p(c|w) = \sum_t \frac{\phi_{wt}\theta_{ct}}{\tau_t}$$

Biterm Topic Model (BTM):

$$p(c|w) = \sum_t \frac{\phi_{wt}\phi_{ct}}{\tau_t}$$

What will happen with the context embeddings? Averaging.

Our approach: conditional independence

Topic-conditional independence for context words:

$$p(c_{i-h}, \ldots, c_{i+h}, w_i) = \sum_t \prod_{j \in H_i} p(c_j|t)\, p(w_i|t)\, p(t)$$

What will happen with context embeddings? Multiplication:

$$\psi_i = \frac{1}{Z}\, \tau^{-2h} \circ \phi_{w_i} \circ \theta_{c_{i-h}} \circ \cdots \circ \theta_{c_{i+h}}$$

Note: a latent topic is assigned to each window (not to a pair of words!).

Why it's better: (transfer, river, money, bank) will not get high probability.

General case (theorem)

Task: given the corpus X of subsets of V, learn |V|×|T| embeddings matrix:

$$\mathcal{L} = \prod_{x \in X} p(x) \to \max_{\Theta} \qquad p(x) = \sum_t \prod_{v \in x} p(v|t)p(t)$$

E-step (Bayes' rule):

$$p(t|x) = \frac{p(t)p(x|t)}{p(x)} = \frac{\prod_{v \in x} p(t)p(v|t)}{p(x)} = \frac{p(t)}{p(x)} \prod_{v \in x} \frac{p(t|v)p(v)}{p(t)}, \; \forall x \in X$$

M-step (likelihood maximization):

$$p(t|v) = \frac{1}{\sum_x [v \in x]} \sum_{x : v \in x} p(t|x), \quad \forall v \in V$$

Experiments? In progress.

Experiment: WNTM and BTM

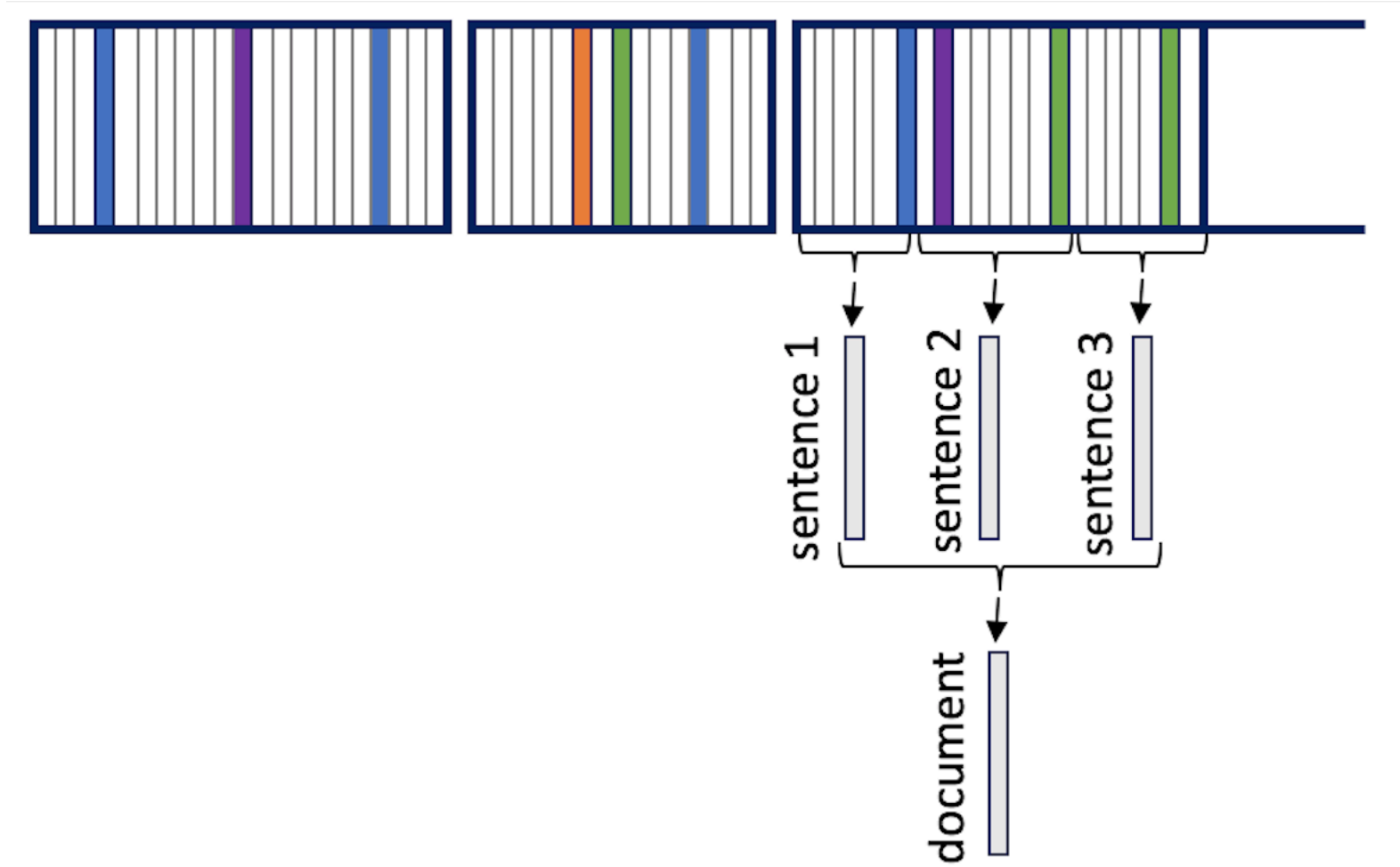
Spearman correlation on standard word similarity tasks.

All models trained on Wikipedia with BigARTM library.

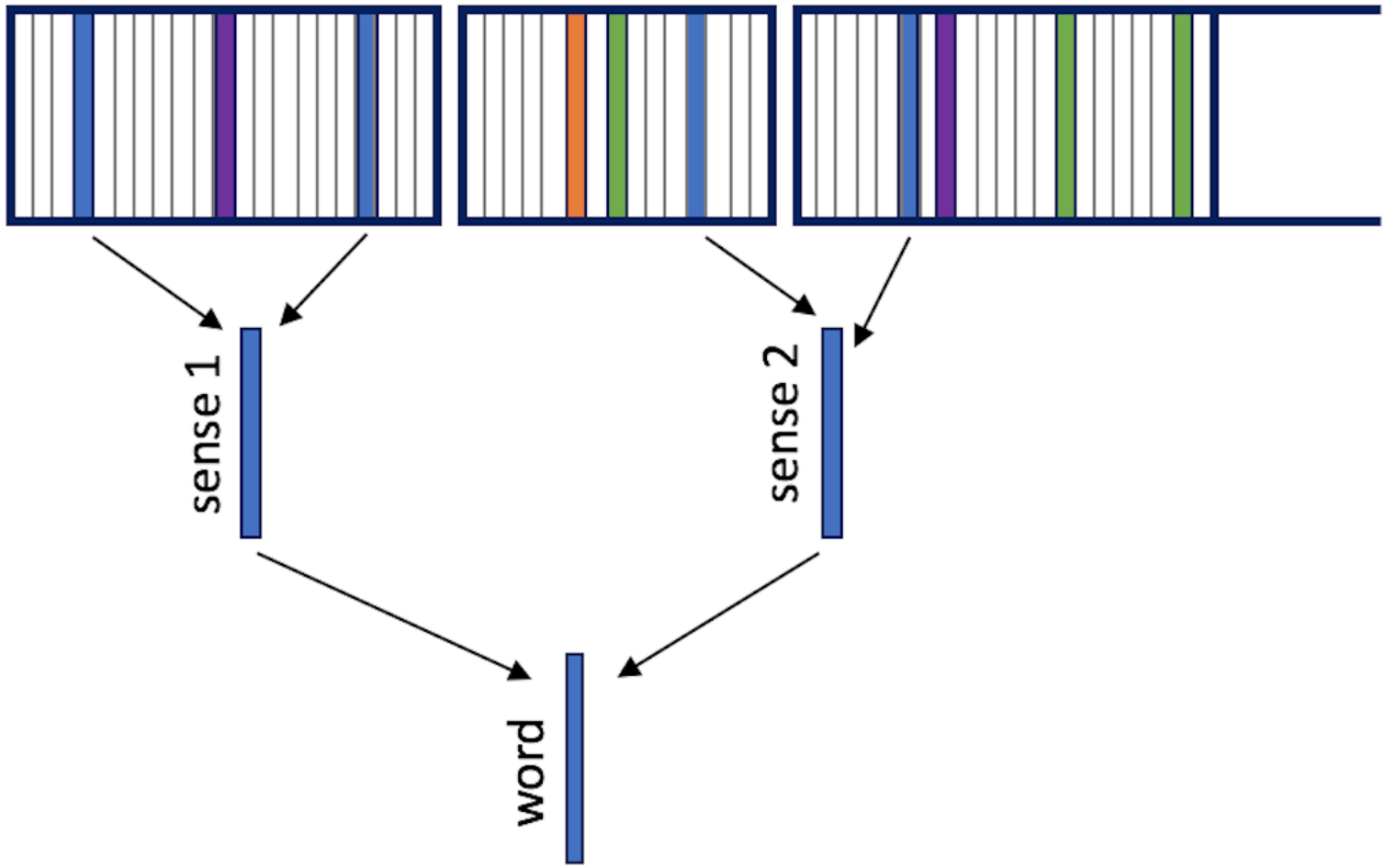
Model	WS-353 Sim	WS-353 Rel	WS-353 All	SimLex Hill et al.	MEN Bruni et. al	RareWords Luong et al.	Radinsky M. Turk
BTM	**0.68**	**0.59**	**0.61**	0.24	0.65	0.32	0.54
WNTM	0.67	0.58	0.60	0.24	**0.66**	**0.33**	**0.55**

BTM and WNTM produce embeddings of the same quality, while the number of parameters in BTM is two times smaller.

Averaging E-step embeddings



Averaging E-step embeddings



Experiment: sentence embeddings

Sentence embeddings performance for unsupervised tasks:

- Person/Spearman correlations for SICK relatedness and STS-2014
- Accuracy for SICK entailment

Model	STS-2014						SICK	
	Forum	News	Headlines	Images	Tweets	Average	Rel	Ent
BOW (ours)	0.41/ 0.42	0.70/ 0.62	0.60/ 0.53	0.76/ 0.71	0.68/ 0.63	0.64/ 0.60	0.77/ 0.70	76.27
Fitted (ours)	**0.45/ 0.46**	**0.70**/ 0.62	0.61/ 0.55	**0.76**/ 0.71	0.68/ 0.62	**0.65**/ 0.62	0.78/ **0.71**	**76.96**
BOW (w2v)	0.39/ 0.46	0.67/ **0.66**	**0.64/ 0.60**	0.76/ **0.72**	**0.70/ 0.69**	0.65/ **0.65**	**0.79**/ 0.69	75.62

Averaging E-step embeddings outperforms averaging word embeddings.

Summary:

- Topic modeling EM is an iterative procedure of learning embeddings
  - for the most fine-grained units of sense (E-step)
  - for all entities in the provided data (M-step)
- Choice of the smallest embedded units is important, we covered:
  - Intersection of equivalence relations with respect to a number of text features
  - A group of words in a sliding window (or in a sentence)
- Those units can be used to build a consistent hierarchy of embeddings
  - Sentence embeddings is one example