

Distributional meaning representations Aka word embeddings

Pia Sommerauer & Antske Fokkens
Text mining @ CBS (July 1-2 2019)

Acknowledgments

- Antske Fokkens. LOT course ‘Distributional Semantics’
- Baroni & Boleda. Distributional Semantic Models
<https://www.cs.utexas.edu/~mooney/cs388/slides/dist-sem-intro-NLP-class-UT.pdf>

What we will cover

- Linguistic motivation
- What are distributional semantic models?
- How can we create distributional semantic models?
- What can they tell us about meaning?
- Evaluation & application
- Discussion & open questions

Motivation & intuition

Meaning representations (for NLP)

Traditional	Dictionary definitions
Hand-crafted and machine readable	WordNet and other computational lexica
Vector space for ML	<ul style="list-style-type: none">● One-hot encodings● Traditional co-occurrence vectors● Embeddings

Meaning representations (for NLP)

<i>Traditional</i>	Dictionary definitions
<i>Hand-crafted and machine readable</i>	WordNet and other computational lexica
<i>Vector space for ML</i>	<ul style="list-style-type: none">● One-hot encodings● Co-occurrence vectors● Embeddings

Meaning = Use

*“You shall know a word by the company
it keeps”*

(Firth, J. R. 1957:11)



What do we know about a word **X**?

*Whereas traditional politicians offer visitors **X**, the Reform of Heisei serves black coffee.*

*The river Neckinger, “the colour of strong **X**”, flowed round Jacob’s Island.
X comes from the leaves that have been withered and dried immediately after picking.*

*It is a large leaf **X** with a very delicate flavor.*

*It may be black or **X** flavored with jasmine flowers, is very fragrant and is always drunk without milk.*

(Examples taken from the BNC)

What do we know about a word **X**?

- a similar category as black coffee - a (hot) beverage?
- different degrees of strength:
mixed/drawn/brewed
- color can be used to describe a river:
transparent, blue, green, brown tone
- made from dried leaves
- can have delicate flavor; probably
variations in flavor exist
- there is something similar that is black

What do we know about a word **X**?

- a similar category as black coffee - a (hot) beverage?
- different degrees of strength: mixed/drawn/brewed
- color can be used to describe a river: transparent, blue, green, brown tone
- made from dried leaves
- can have delicate flavor; probably variations in flavor exist
- there is something similar that is black



A simple co-occurrence vector

*They **served** hot, steaming coffee.*

*We drink coffee in our **break**.*

*A **cup** of coffee, please!*

*Can I **offer** you some coffee or **tea**?*

*Would you like **milk** or **sugar** with your
coffee or do you **drink** it **black**?*

	Coffee
serve	1
hot	1
drink	2
tea	1
cup	1
sugar	1
milk	1
black	1

Similar words are used similarly

What else could **X** have referred to?

A simple co-occurrence matrix

	Coffee	Green tea	Black tea	dog
hot	15	17	18	4
drink	15	16	17	21
cup	11	12	13	2
sugar	20	10	15	3
milk	21	3	16	0
black	23	2	25	10
bark	0	0	0	26

A 2D plot with the x-axis labeled 'runs' ranging from 0 to 6. The y-axis ranges from 0 to 6. Three vectors originate from the point (0,0):

- The vector labeled 'cat (1,5)' points to the coordinates (1,5).
- The vector labeled 'dog (1,4)' points to the coordinates (1,4).
- The vector labeled 'car (4,0)' points to the coordinates (4,0).

Creating distributional vectors

Data	Select a corpus, preprocess
What is the context (or 'window')?	Document, paragraph, sentence, n words around target
Create matrix of word vectors	Count (count-based statistics), ML (learn to predict the representation)



Embeddings

What can distributional vectors tell us?

→ Meaning of a word derived from **actual usage**

→ Meaning of a word as the position in a **system of words and relations**

What is the meaning of X? It is more similar to x, y and z than to a, c and c

→ **High coverage** (compared to hand-crafted resources)

→ Very good **generalization**

I have never seen word x before, but it looks very similar to words y and z

→ Very compatible with **machine learning systems**

Distributional model creation & algorithms

Selecting the context

- document
- sentence
- n-nearest words
- syntactically related words

Preprocessing steps: frequency filter, stop-words, normalization
(lower-casing, numbers, punctuation)

Impact of window size

Nearest neighbors of the
target word 'dog'

(Baroni & Boleda)

2-word window

- ▶ cat
- ▶ horse
- ▶ fox
- ▶ pet
- ▶ rabbit
- ▶ pig
- ▶ animal
- ▶ mongrel
- ▶ sheep
- ▶ pigeon

30-word window

- ▶ kennel
- ▶ puppy
- ▶ pet
- ▶ bitch
- ▶ terrier
- ▶ rottweiler
- ▶ canine
- ▶ cat
- ▶ to bark
- ▶ Alsatian

Model algorithms

Traditional

- Counts
- PPMI (positive pointwise mutual information)
- PPMI + SVD (singular value decomposition)

ML (prediction-based)

- Word2vec
- Glove
- ELMO

Count-based models

Just counting is not ideal

- High numbers for frequent words
- Frequency is overemphasized (e.g. the-dog vs leash-dog)

Alternative: Mutual information

- How likely are two words to occur together (compared to them occurring separately)?
- More weight to informative co-occurrences (dog-leash > the-dog)

Pointwise Mutual Information (PMI)

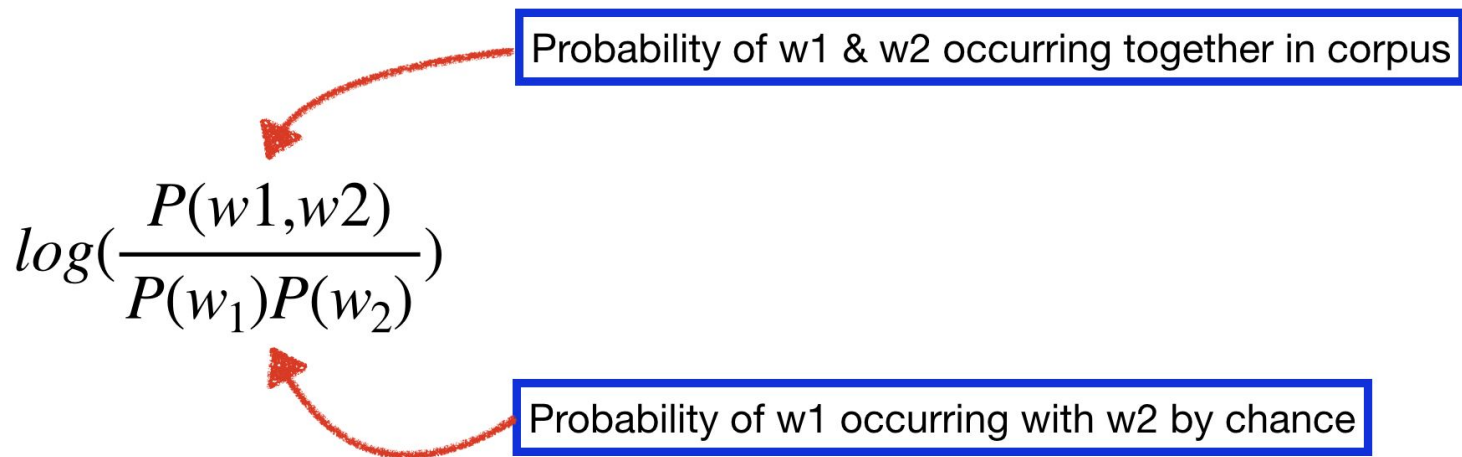


Diagram illustrating the components of the Pointwise Mutual Information (PMI) formula:

$$\log\left(\frac{P(w_1, w_2)}{P(w_1)P(w_2)}\right)$$

The formula is annotated with two explanatory boxes:

- Top Box:** Probability of w_1 & w_2 occurring together in corpus (points to the numerator $P(w_1, w_2)$)
- Bottom Box:** Probability of w_1 occurring with w_2 by chance (points to the denominator $P(w_1)P(w_2)$)

Positive Pointwise Mutual Information (PPMI)

Negative values (minus infinity)

$$\text{PPMI} = \text{argmax}(0, \log(\frac{P(w1, w2)}{P(w1)P(w2)}))$$

Positive Pointwise Mutual Information (PPMI)

Consequences (for applications)?

Positive Pointwise Mutual Information (PPMI)

Consequences (for applications)?

→ high dimensional

→ sparse

→ overemphasis on rare words

PPMI with Singular Value Decomposition (SVD)

- Reduce number of dimensions using linear algebra: preserve most of the variance in the original matrix
- Higher generalization
- Lower-dimensional vectors (predefined)
- Dense vectors

Prediction-based models

Inspired by language models

Given a sequence of words, predict the next word in the corpus → adjust representations accordingly

Prediction-based models

Prediction-based word embeddings (e.g. Word2vec models - Mikolov et al. 2013a,b)

- Given a (bag of) context words, predict the target word (CBOW)
- Given a target word, predict the context words (Skip-gram)

More explanation: <http://ruder.io/word-embeddings-1/>

Word2vec models

Mikolov et al. (2013a,b):

- Start with random initializations
- Word and context matrix
- Several model architectures:
 - CBOW & Skipgram
 - Training: hierarchical softmax & negative sampling
 - Preprocessing: dynamic context windows, subsampling, delete rare words
- Resulting distributional model: use the trained word matrix

Word2vec training

Hierarchical softmax:

efficient way to determine most probable context given a word (or vice versa) over the whole model

Negative sampling:

distinguish the actual context words from k other words (randomly chosen)

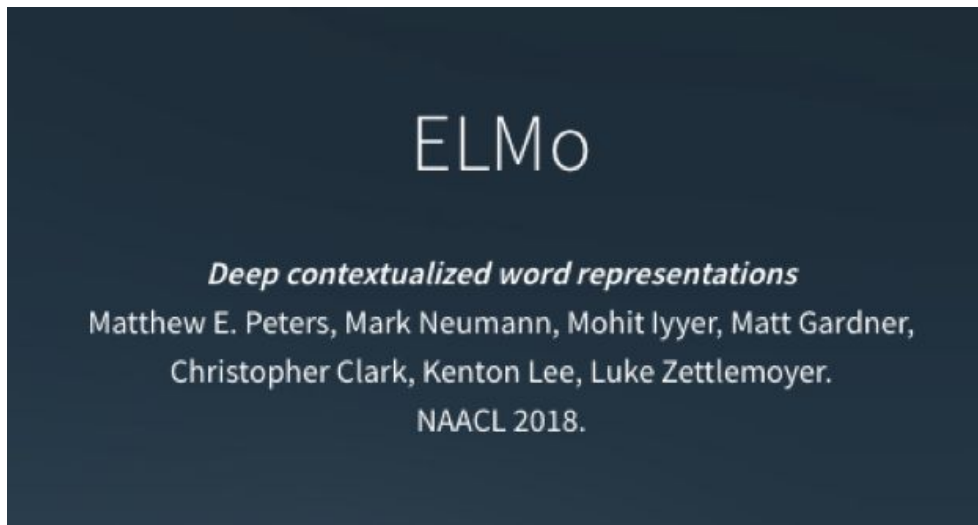
Properties of prediction-based models

- Low dimensionality (predefined, usually 100 - 500 dimensions)
- Dense
- Stability (attention when using small data):
 - The model starts with random vectors which are updated.
 - The model usually processes the data in random order.
 - Usually, models do not converge → the results on the same corpus may vary
- Not transparent: dimensions do not correspond to particular context words

Comparison

	dimensions	density	transparence	stability
Count and PPMI	High (vocabulary)	Low (most words do not co-occur)	1 d corresponds to 1 context word	stable
PPMI-SVD	Low	dense	1 d does not correspond to a context word	stable
Predict models	low	dense	1 d does not correspond to a context word	Not stable

From word to context embeddings (ELMO)



- Easy to use (with advanced context selection & neural network for learning)
- State-of-the-art for many NLP tasks
- <https://allennlp.org/elmo>

Data & commonly used models

Commonly used corpora (for English)

Goal: “general purpose embeddings”

Wikipedia dumps

Google n-grams

BNC (British National Corpus)

A note on corpus sizes

- Embeddings depend on patterns found in many word-context instances
- There is research on small data embeddings, but the general tendency is bigger corpus → better embeddings
- Stability becomes an issue with small data

	Corpus	Words
small	A section of a historical corpus for English (COHA)	22 million words
large	Wikipedia dump 2014	1.9 billion words

Pretrained models

- Google News model (biggest model available)
- Several pretrained wikipedia models (for various languages)
- ELMO models
- fastText models (Facebook)
- ...

Pretrained models are easy to use, but it is very hard (often impossible) to gain insights into the underlying corpus and its possible biases/artifacts/irregularities.

Creating your own models

Gensim

Hyperwords

Introduction in the afternoon

Evaluation

Types of evaluation

Intrinsic:

How well do the meaning representations correspond to human judgments?

Extrinsic:

How much do the meaning representations 'boost' a downstream NLP system?

Intrinsic evaluation

How can we compare vector representations to human judgements about meaning?

→ Compare relations

- Human judgements on word similarity
- Human judgements on word relatedness
- Relational analogy: A is to B as C is to X → predict X
(e.g man is to woman as king is to X)

Semantic similarity and relatedness

	Similar	Related
<i>coffee-tea</i>	yes	yes
<i>coffee-cup</i>	no	yes
<i>gasoline-cup</i>	no	yes

Semantic similarity and relatedness judgements

- Humans rank word pairs according to similarity and relatedness
- Average over multiple annotators
- Rankings produced by the distributional model should correlate with the human judgements (Spearman Rho correlation)

Similarity in the distributional model is expressed as the cosine between two vectors

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|}$$

Analogy task

given

bangkok

bangkok

bangkok

bangkok

bangkok

beijing

beijing

given

thailand

thailand

thailand

thailand

thailand

china

china

given

paris

rome

stockholm

tehran

tokyo

berlin

bern

predict

france

italy

sweden

iran

japan

germany

switzerland

Completing analogies

Are individual semantic properties encoded in (patterns of) dimensions?

Man:woman \approx king:queen (Mikolov et al. 2013)

King - man + woman \approx queen

0	0	1	1	female
1	1	0	0	male
1	0	0	1	royal

Heavily criticized
(e.g. Linzen 2016)

Analogy task criticism

given	given	given	predict
bangkok	thailand	paris	france
bangkok	thailand	rome	italy
bangkok	thailand	stockholm	sweden
bangkok	thailand	tehran	iran
bangkok	thailand	tokyo	japan
beijing	china	berlin	germany
beijing	china	bern	switzerland

Excluded from possible answers

Commonly used evaluation sets for English

Similarity and relatedness

- **WS-353** (Finkelstein et al. 2001): 353 pairs ranked for similarity & relatedness on a scale
 - WS-353-sim: subsection with just similarity or low score
 - WS-353—rel: subsection capturing other forms of relatedness
- **MEN** (Bruni et al. 2012): 3,000 pairs ranked for similarity & relatedness by having humans select the more related pair out of two pairs
- **SimLex-999** (Hill et al. 2015): 999 pairs annotated for similarity only: rated on a scale of 0-6 looking at 7 pairs simultaneously.
- **Radinsky** (Radinsky et al. 2011): 280 pairs of words occurring in the New York times and DBpedia with varying PMI scores. The general approach follows WS-353.
- **Luong** rare words (Luong et al. 2013): at least one of the two words in the pair is rare (5-10, 10-100, 100-1,000, 1,000-10,000 occurrences in wikipedia), filtered using WordNet.

Analogy sets (for English)

Mikolov et al. (2013a):

- Google semantic analogies
- Google morphological analogies

Alternative set with more options for analysis:

- BATS (Gladkova et al. 2016)

Which models perform best?

Count, predict, which corpus?

Intrinsic evaluation

Extensive comparisons using PPMI, PPMI-SVD, Word2vec models (trained on Wikipedia) lead to contradictory results (Baroni et al. 2014, Levy et al. 2015).

Results vary, hyperparameters have an impact, scores are sometimes very close

Intrinsic and extrinsic evaluation

What do we learn from evaluations?

Are models that perform well on a similarity task always helpful for, let's say, sentiment classification?

Intrinsic and extrinsic evaluation

“Performance on **downstream tasks** is **not consistent across tasks**, and may **not be consistent with intrinsic evaluations**. Comparing performance across tasks may provide insight into the information encoded by an embedding, but **we should not expect any specific task to act as a proxy for abstract quality**. “

From: [Evaluation methods for unsupervised word embeddings](#) (Schnabel et al. 2015, p. 304)

Thank you :-)

Questions?

pia.sommerauer@vu.nl

References 1

- Linzen, T. (2016) "[Issues in evaluating semantic spaces using word analogies](#)." *arXiv preprint arXiv:1606.07736*
- Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuoka. 2016. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of NAACL-HLT 2016*, pages 47–54. Association for Computational Linguistics.
- Schnabel, Tobias, Igor Labutov, David Mimno, and Thorsten Joachims. "Evaluation methods for unsupervised word embeddings." In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 298-307. 2015.
- Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. "Efficient Estimation of Word Representations in Vector Space." *arXiv preprint arXiv:1301.3781* (2013a).<https://arxiv.org/pdf/1301.3781.pdf>
- Mikolov, Tomas, Wen-tau Yih, and Geoffrey Zweig. "Linguistic regularities in continuous space word representations." In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 746-751. 2013b.
- Bruni, E., Boleda, G., Baroni, M., & Tran, N. K. (2012, July). Distributional semantics in technicolor. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1* (pp. 136-145). Association for Computational Linguistics.
- Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G. and Ruppin, E., 2002. Placing search in context: The concept revisited. *ACM Transactions on information systems*, 20(1), pp.116-131.

References 2

- Hill, F., Reichart, R. and Korhonen, A., 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4), pp.665-695.
- Luong, Minh-Thang, Richard Socher, and Christopher D. Manning. 2013. Better word representations with recursive neural networks for morphology. CoNLL-2013, page 104, Sofia.
- Radinsky, K., Agichtein, E., Gabrilovich, E. and Markovitch, S., 2011, March. A word at a time: computing word relatedness using temporal semantic analysis. In *Proceedings of the 20th international conference on World wide web* (pp. 337-346). ACM.
- Levy, Omer, Yoav Goldberg, and Ido Dagan. "Improving distributional similarity with lessons learned from word embeddings." *Transactions of the Association for Computational Linguistics* 3 (2015): 211-225.
- Baroni, Marco, Georgiana Dinu, and Germán Kruszewski. "Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors." In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, pp. 238-247. 2014.