Modern word embedding methods

Väinö Yrjänäinen

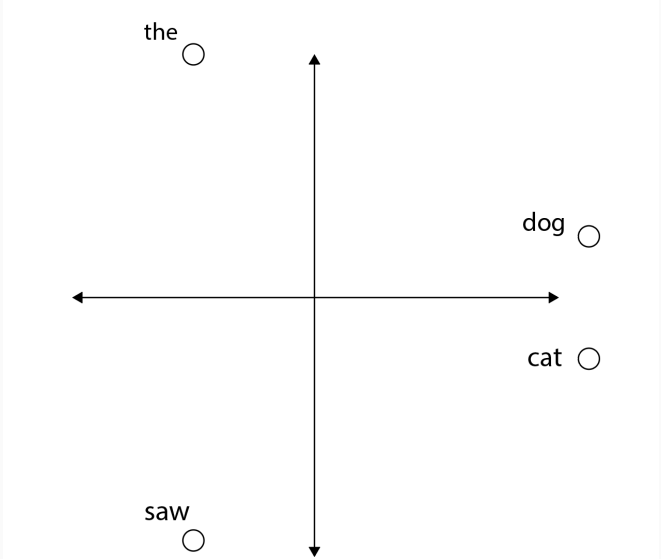
8.12.2022

Analysis of large corpora?

Simple methods

- Counting words
  - Conceptually straightforward and computationally fast
  - Not very flexible
  - All words are treated as independent categories
- Manual analysis by humans
  - Not applicable to big data

Word embeddings



Word embeddings

$$\text{dog} = \begin{bmatrix} 0.23 \\ -1.47 \\ 1.01 \\ 1.33 \\ \dots \\ -0.56 \end{bmatrix}, \text{cat} = \begin{bmatrix} 0.42 \\ -1.07 \\ 0.31 \\ 1.54 \\ \dots \\ 0.14 \end{bmatrix} \in \mathbb{R}^{100}$$

Word embeddings

- Distance
  - Cosine distance
  - How similar are words x, y
  - What is the most similar word to word x
- Analogies
  - king $-$ man $+$ woman $\approx$ queen
- And more

Analogies

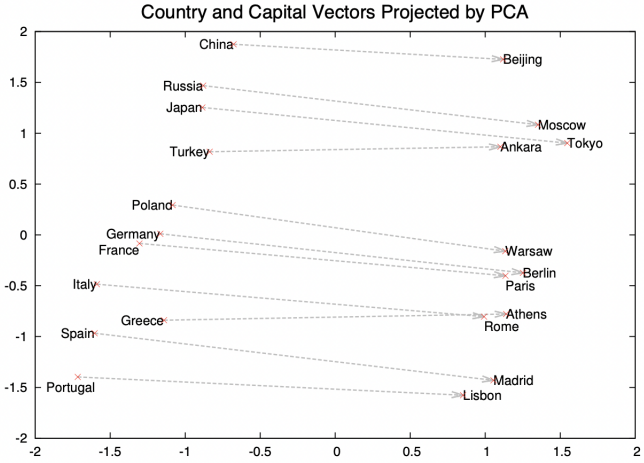


Figure 2: Two-dimensional PCA projection of the 1000-dimensional Skip-gram vectors of countries and their capital cities. The figure illustrates ability of the model to automatically organize concepts and learn implicitly the relationships between them, as during the training we did not provide any supervised information about what a capital city means.

Training word embeddings

[...] as the oldest domesticated species, dogs' minds inevitably have been shaped by millennia of contact with humans [...]

Context window based methods

[...] as the oldest domesticated species, dogs' minds inevitably have been shaped by millennia of contact with humans [...]

dog

{domesticated, species, minds, inevitably}

Word2Vec – concept

- Continuous bag-of-words CBOW
  Does the word x appear, given that its context window is $y_1, y_2, \ldots, y_{ws}$
- Skip-gram
  Do words x and y appear in the same word window?

Continuous bag-of-words

[...] as the oldest domesticated species, [???] minds inevitably have been shaped by millennia of contact with humans [...]

Skip-gram

[...] as the oldest domesticated species, [???] minds inevitably have been shaped by millennia of contact with humans [...]

Skip-gram

[...] as the oldest domesticated
species, [???] minds inevitably
have been shaped by millennia
of contact with humans [...]

Skip-gram

[...] as the oldest domesticated species, [???] minds inevitably have been shaped by millennia of contact with humans [...]

Skip-gram

[...] as the oldest domesticated species, [???] minds inevitably have been shaped by millennia of contact with humans [...]

Word2Vec - mathematical definition

- Each word is assigned a word vector w and a context vector c
  - $w, c \in \mathbb{R}^D$, where D is the *dimensionality* of the embedding
- Conditional probabilities are functions of these vectors
  - $P(x \mid y_1, \ldots, y_w) = f(w_x, c_{y_1}, \ldots, c_{y_w})$

Skip-gram

Do words x and y appear in the same word window?

$$p(\text{x and y co-occur}) = \sigma(w_x^T c_y)$$

where $\sigma(\cdot)$ is the logistic function.

Skip-gram numerical example

$$w_{\text{dog}} = \begin{bmatrix} 0.23 \\ -1.47 \\ 1.01 \\ 1.33 \\ \dots \\ -0.56 \end{bmatrix}, \; c_{\text{cat}} = \begin{bmatrix} 0.42 \\ -1.07 \\ 0.31 \\ 1.54 \\ \dots \\ 0.14 \end{bmatrix}$$

Skip-gram numerical example

Do the words *dog* and *cat* appear in the same word window?

$$p(\text{dog and cat co-occur}) = \sigma\left(\begin{bmatrix} 0.23 & -1.47 & 1.01 & 1.33 & \dots & -0.56 \end{bmatrix} \begin{bmatrix} 0.42 \\ -1.07 \\ 0.31 \\ 1.54 \\ \dots \\ 0.14 \end{bmatrix}\right)$$

$$= \sigma(6.2325) = 0.998$$

Skip-gram numerical example

$$w_{\text{dog}} = \begin{bmatrix} 0.23 \\ -1.47 \\ 1.01 \\ 1.33 \\ \dots \\ -0.56 \end{bmatrix}, \; c_{\text{university}} = \begin{bmatrix} -2.03 \\ 0.07 \\ 1.01 \\ -0.34 \\ \dots \\ 0.89 \end{bmatrix}$$

Skip-gram numerical example

Do the words *dog* and *university* appear in the same word window?

$$p(\text{dog and university co-occur}) = \sigma\left(\begin{bmatrix}0.23 & -1.47 & 1.01 & 1.33 & \dots & -0.56\end{bmatrix}\begin{bmatrix}-2.03\\0.07\\1.01\\-0.34\\\dots\\0.89\end{bmatrix}\right)$$

$$= \sigma(-0.13) = 0.4675$$

Continuous bag-of-words

Do words x and y appear in the same word window?

$$p(\text{x occurs given C} = y_1, \ldots, y_{ws}) = \sigma(w_x^T \sum_{y \in C} c_y)$$

where C is the context window, and $\sigma(\cdot)$ is the logistic function.

Negative sampling – idea

- In the data, we can easily find instances where words do co-occur
  - "the dog saw the cat"
  - Eg. (dog,saw), (dog,cat) and (the, cat) co-occur
- We do not have direct examples of words not co-occuring
- We need negative examples, otherwise we will just have all vectors be the same
- Solution: pick at random

Negative sampling

- For each word x in the data
  - Sample ns negative samples y'_1, y'_{ns} from the empirical distribution of word types
  - Add a term $1 - p(\text{x and } y' \text{ co-occur})$ in the likelihood

Negative sampling – putting it all together

Say we only have the data point *dog co-occurs with cat*. We draw a negative sample *university* from the empirical distribution of words in the data. The likelihood is

$$
\begin{aligned}
p(\text{data}) &= \underbrace{p(\text{dog and cat co-occur})}_{\text{positive sample}} \underbrace{(1 - p(\text{dog and university co-occur}))}_{\text{negative sample}} \\
&= 0.998 \cdot (1 - 0.4675) = 0.531435
\end{aligned}
$$

Negative sampling – putting it all together

For a real dataset, the log likelihood is

$$\log p(\mathcal{D} \mid w, c) = \sum_{i=1}^{N} \left(\underbrace{\sum_{y \in C_i} \log \sigma(w_{x_i}^T c_y)}_{\text{positive samples}} + \underbrace{\sum_{y \in C_{ns}} \log(1 - \sigma(w_{x_i}^T c_y))}_{\text{negative samples}} \right)$$

where $\mathcal{D} = (x_1, \ldots, x_N)$ is the dataset indexed by $i \in \{1, \ldots, N\}$, C_i is the context window at i and C_{ns} is a randomized context window generated from the empirical distribution of words.

Different word embeddings

Probabilistic word embeddings

- Word2Vec can be formulated as a probabilistic language model
  - Same likelihood
  - Adds a prior on the parameters, such as $w_x \sim \mathcal{N}(0, \lambda_0 I)$ for all word types $x \in W$
  - Different priors are essential
- Estimated via MAP or variational inference

Bernoulli embeddings (Rudolph et al 2016)

$$\begin{aligned}\log p(w, c \mid \mathcal{D}) = \log p(\mathcal{D} \mid w, c) \\ + \sum_{x \in W} \lambda_1 \|w_{x,t}\|^2 \\ + \sum_{x \in W} \lambda_0 \|c_x\|^2\end{aligned}$$

Probabilistic word embeddings – advantages

- Straightforward way of including prior knowledge
  - Regularize models for data with natural divisions to subsets, for example regularize models that model different time periods
  - Incorporate additional information about the words, such as dictionary information
  - Enforce certain properties of the embedding, such as sentiment aspects
- Uncertainty estimation

Dynamic probabilistic word embeddings

- Different set of word vectors for each time period $t \in \{1, \ldots, T\}$
- Random walk prior over the word vectors

$$w_{x,1} \sim \mathcal{N}(0, \lambda_0 I)$$
$$w_{x,t} \sim \mathcal{N}(w_{x,t-1}, \lambda_1 I) \text{ for } t \in \{2, \ldots, T\}$$

- Spherical prior on context vectors

$$c_x \sim \mathcal{N}(0, \lambda_0 I)$$

Dynamic probabilistic word embeddings

$$\log p(w, c \mid \mathcal{D}) = \log p(\mathcal{D} \mid w, c) + \sum_{x \in W} \sum_{t=1}^{T} \lambda_1 \| w_{x,t+1} - w_{x,t} \|^2 + \sum_{x \in W} \sum_{t=1}^{T} \lambda_0 \| w_{x,t} \|^2 + \sum_{x \in W} \lambda_0 \| c_x \|^2$$

Applications

words with largest drift (Senate)

IRAQ	3.09	coin	2.39
tax cuts	2.84	social security	2.38
health care	2.62	FINE	2.38
energy	2.55	signal	2.38
medicare	2.55	program	2.36
DISCIPLINE	2.44	moves	2.35
text	2.41	credit	2.34
VALUES	2.40	UNEMPLOYMENT	2.34

Figure 10: Detecting words whose usage has changed a lot over time. Rudolph et al (2018).

PELP
democrat, abortion, announce, maine, republican, gun, illegal, republicans, breaks, stimulus, taxes, immigration, kentucky, accounting, wyoming
Bernoulli
hubbert, islanders, rickover, pottawatomi, gaspee, mastercard, morgenthau, compean, 205106150, fairtax, vertical, peaking, follette, isna, hubberts

Table 2: Top 15 words w with the largest distance between $\rho_{w,R}$ and $\rho_{w,D}$. PELP model above, reference model Bernoulli Embeddings below.

Polarizing words over time

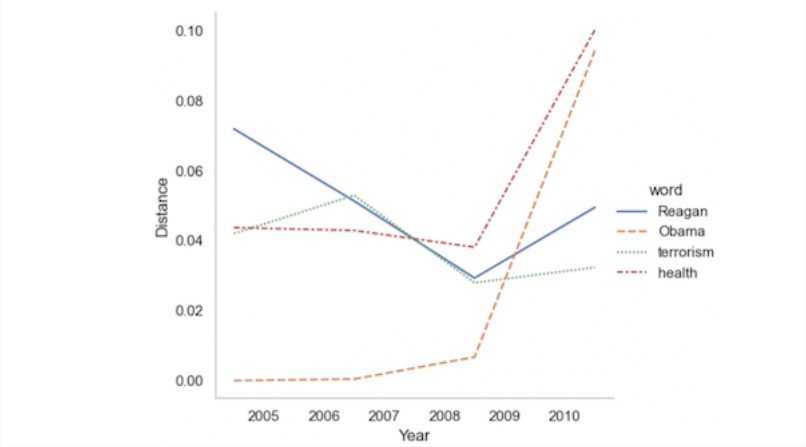


Figure 12: Partisan differences over time in the US Congress dataset.

Incorporating additional information

- Probabilistic word embeddings can be prior knowledge about the embeddings
- For example, if have a list of synonym pairs, we can inject this prior knowledge into the embeddings
  - This will make embeddings more closely resemble human evaluation of similarity
- As the dataset grows larger, there will be more word types
  - There will always be words that benefit from regularization

Incorporating additional information – better correspondence with human evaluation

Model	50M	200M
PELP	.509	**.515**
Dict2Vec	**.512**	.513
Bernoulli	.444	.465
word2vec	.445	.461

Table 5: Word similarity performance with the 50M and the 200M token partitions of the English Wikipedia, D=100, $\lambda_0, \lambda_1 = 1.0, 7.0$. Averages over all evaluation sets.

Uncertainty estimation

- Many applications want to draw conclusions about the data
- Random noise can be mistaken for results
- Methods to estimate embedding uncertainty
  - Probabilistic word embeddings
  - **Bootstrap**

Bootstrapping word embeddings (Antoniak and Mimno 2018)

- Divide corpus into documents that can be resampled
- Generate N boostrap resamples
- Train the word embedding separately on all N resamples
- Calculate quantity of interest for all N word embeddings

Bootstrapping word embeddings

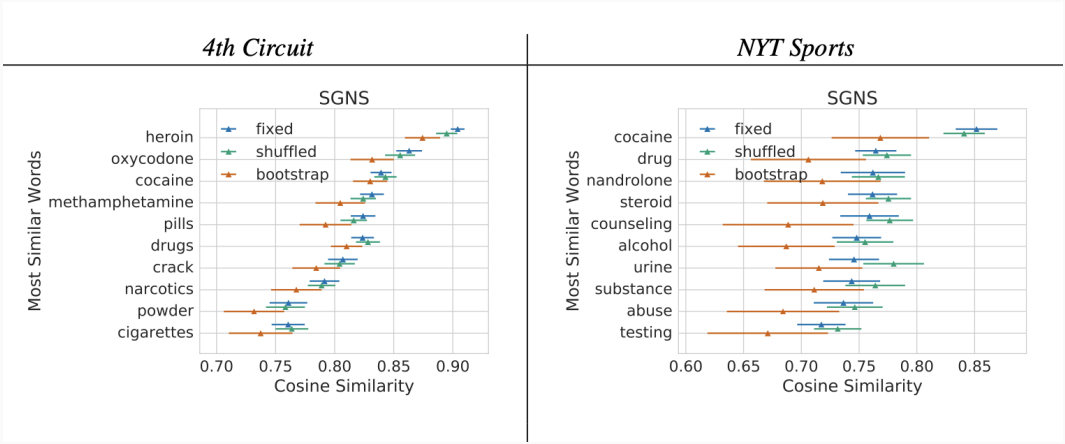


Figure 13: Most similar words to 'marijuana' in two corpora (Antoniak and Mimno 2018)

Bootstrapping word embeddings application (Garg et al 2018)

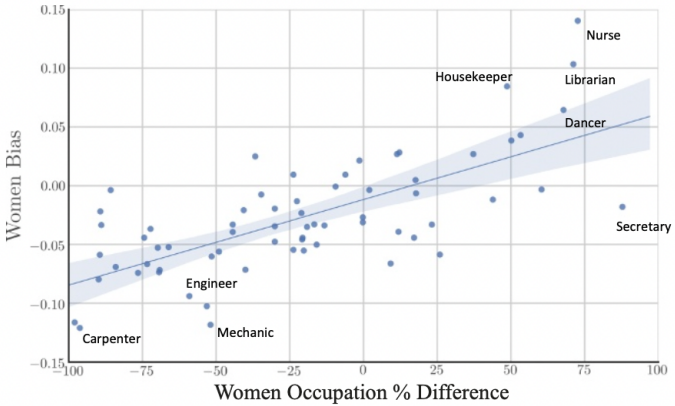


Fig. 1. Women's occupation relative percentage vs. embedding bias in Google News vectors. More positive indicates more associated with women on both axes. $P < 10^{-10}$, $r^2 = 0.499$. The shaded region is the 95% bootstrapped confidence interval of the regression line. In this single embedding, then, the association in the embedding effectively captures the percentage of women in an occupation.

Summary

- Word2vec was seminal work and its variants are still used to this day
- Probabilsitic embeddings enable the incorporation of prior knowledge and uncertainty estimation
- Boostrap can be used to estimate embedding uncertainty if the data is small