Neural Word Embeddings as Matrix Factorization

Master's Thesis Mathematics

Presented by:
Theresa Klumpp

Supervisors:
Prof. P. Pfaffelhuber
Prof. H. Bast

January 15, 2020

Table of Contents

Problem

Goal: word vectors that reflect similarities and dissimilarities

Problem

Goal: word vectors that reflect similarities and dissimilarities

Underlying hypothesis: words in similar contexts have similar meanings

Problem

Goal: word vectors that reflect similarities and dissimilarities

Underlying hypothesis: words in similar contexts have similar meanings

- I get to work faster when I take the ***.

Problem

Goal: word vectors that reflect similarities and dissimilarities

Underlying hypothesis: words in similar contexts have similar meanings

- I get to work faster when I take the ***.
- This model has amazing acceleration for a *** of its size.

Problem

Goal: word vectors that reflect similarities and dissimilarities

Underlying hypothesis: words in similar contexts have similar meanings

- I get to work faster when I take the ***.
- This model has amazing acceleration for a *** of its size.
- I would never drive my *** into Paris if I could get there by train.

Problem

Goal: word vectors that reflect similarities and dissimilarities

Underlying hypothesis: words in similar contexts have similar meanings

- I get to work faster when I take the ***.
- This model has amazing acceleration for a *** of its size.
- I would never drive my *** into Paris if I could get there by train.

Demo

Contributions

- Gaining an understanding of the objective functions of skip-gram (with and without negative sampling) and the statistical models behind them.
- Finding a maximum for skip-gram's objective.
- Showing the connection between the neural networks and Singular Value Decomposition (SVD).
- Comparing different metrics on the sphere.
- Finding a formula for the expectation of the distance of the closest vector.
- An implementation of the SGNS neural network and the SVD variant for both skip-gram and SGNS.
- Evaluation of the models on word similarity and analogy tasks.

Contributions

- Gaining an understanding of the objective functions of skip-gram (with and without negative sampling) and the statistical models behind them.
- Finding a maximum for skip-gram's objective.
- Showing the connection between the neural networks and Singular Value Decomposition (SVD).
- Comparing different metrics on the sphere.
- Finding a formula for the expectation of the distance of the closest vector.
- An implementation of the SGNS neural network and the SVD variant for both skip-gram and SGNS.
- Evaluation of the models on word similarity and analogy tasks.

Questions?

Table of Contents

Definition: Context

Text		Samples
I get to work faster when I take the car.	⇒	(I, get)
		(I, to)
I get to work faster when I take the car.	⇒	(get, I)
		(get, to)
		(get, work)
I get to work faster when I take the car.	⇒	(to, I)
		(to, get)
		(to, work)
		(to, faster)
I get to work faster when I take the car.	⇒	(work, get)
		(work, to)
		(work, faster)
		(work, when)

...

Notation

- $\boldsymbol{V_W}$ and $\boldsymbol{V_C}$: word and context vocabulary (we have $V_W = V_C$)
- $\boldsymbol{D}$: observed word context pairs
- $\boldsymbol{\#(w,c)}$: number of times the pair (w, c) appears in D
- $\boldsymbol{\#(w)} = \sum_{c' \in V_C} \#(w, c')$ and $\boldsymbol{\#(c)} = \sum_{w' \in V_W} \#(w', c)$

Mathematical Goal

Find embeddings such that $\vec{w} \cdot \vec{c}$ is

- high for pairs with large $\#(w, c)$ and
- small for pairs with low $\#(w, c)$

Mathematical Goal

Find embeddings such that $\vec{w} \cdot \vec{c}$ is

- high for pairs with large $\#(w, c)$ and
- small for pairs with low $\#(w, c)$

Why does this yield good embeddings?

Mathematical Goal

Find embeddings such that $\vec{w} \cdot \vec{c}$ is

- high for pairs with large $\#(w, c)$ and
- small for pairs with low $\#(w, c)$

Why does this yield good embeddings?

	c_1 = drive	c_2 = road	c_3 = space	c_4 = bottle
w_1 = car	0.9	0.8	0.2	0.1
w_2 = truck	0.8	0.7	0.2	0.2

Mathematical Goal

Find embeddings such that $\vec{w} \cdot \vec{c}$ is

- high for pairs with large $\#(w, c)$ and
- small for pairs with low $\#(w, c)$

$$W = \begin{pmatrix} \vec{w}_1 \\ \vdots \\ \vec{w}_{|V_W|} \end{pmatrix} \text{ and } C = \begin{pmatrix} \vec{c}_1 \\ \vdots \\ \vec{c}_{|V_C|} \end{pmatrix}$$

Mathematical Goal

Find embeddings such that $\vec{w} \cdot \vec{c}$ is

- high for pairs with large $\#(w, c)$ and
- small for pairs with low $\#(w, c)$

$$W = \begin{pmatrix} \vec{w}_1 \\ \vdots \\ \vec{w}_{|V_W|} \end{pmatrix} \text{ and } C = \begin{pmatrix} \vec{c}_1 \\ \vdots \\ \vec{c}_{|V_C|} \end{pmatrix}$$

$\Rightarrow$ Find a function $\ell(W, C)$ that is maximized when the properties above hold.

Skip-Gram: Objective functions

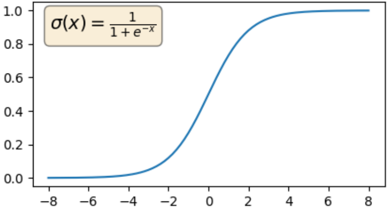
$$\ell_{SG}(W, C) = \sum_{(w,c) \in D} \left(\vec{w} \cdot \vec{c} - \log \left(\sum_{c' \in V_C} \exp \left(\vec{w} \cdot \vec{c'} \right) \right) \right)$$

More

Skip-Gram: Objective functions

$$\ell_{SG}(W, C) = \sum_{(w,c)\in D} \left(\vec{w} \cdot \vec{c} - \log \left(\sum_{c' \in V_C} \exp\left(\vec{w} \cdot \vec{c'}\right) \right) \right)$$

$$\ell_{SGNS}(W, C) = \sum_{(w,c)\in D} \left(\log \sigma\left(\vec{w} \cdot \vec{c}\right) + \sum_{j=1}^{k} \log \sigma\left(-\vec{w} \cdot \vec{c}_j\right) \right)$$



More

Optimal value for the dot products

- $\ell_{SGNS}(W, C)$ is maximized for

$$(\vec{w} \cdot \vec{c})^{\mathrm{OPT}} = \log\left(\frac{\#(w,c) \cdot |D|}{\#(w) \cdot \#(c)}\right) - \log k$$

Optimal value for the dot products

- $\ell_{SGNS}(W, C)$ is maximized for

$$(\vec{w} \cdot \vec{c})^{\mathrm{OPT}} = \log\left(\frac{\#(w,c) \cdot |D|}{\#(w) \cdot \#(c)}\right) - \log k$$

- Note that

$$\left(W \cdot C^{T}\right)_{ij} = \vec{w}_i \cdot \vec{c}_j$$

Optimal value for the dot products

- $\ell_{SGNS}(W, C)$ is maximized for

$$(\vec{w} \cdot \vec{c})^{\text{OPT}} = \log\left(\frac{\#(w,c) \cdot |D|}{\#(w) \cdot \#(c)}\right) - \log k$$

- Note that

$$\left(W \cdot C^T\right)_{ij} = \vec{w}_i \cdot \vec{c}_j$$

- Let M^{OPT} be the matrix containing the optimal dot products, that is

$$M_{ij}^{\text{OPT}} = (\vec{w}_i \cdot \vec{c}_j)^{\text{OPT}}$$

Singular Value Decomposition (SVD)

- $\left(W \cdot C^T\right)_{ij} = \vec{w}_i \cdot \vec{c}_j$ and $M_{ij}^{\mathrm{OPT}} = (\vec{w}_i \cdot \vec{c}_j)^{\mathrm{OPT}}$

Singular Value Decomposition (SVD)

- $\left(W \cdot C^T\right)_{ij} = \vec{w}_i \cdot \vec{c}_j$ and $M_{ij}^{\mathrm{OPT}} = (\vec{w}_i \cdot \vec{c}_j)^{\mathrm{OPT}}$

- Skip-gram with negative sampling is trying to find W and C such that

$$W \cdot C^T = M^{\mathrm{OPT}}$$

Singular Value Decomposition (SVD)

- $\left(W \cdot C^T\right)_{ij} = \vec{w}_i \cdot \vec{c}_j$ and $M_{ij}^{\mathrm{OPT}} = (\vec{w}_i \cdot \vec{c}_j\,)^{\mathrm{OPT}}$
- Skip-gram with negative sampling is trying to find W and C such that

$$W \cdot C^T = M^{\mathrm{OPT}}$$

- Truncated SVD gives us a factorization of the best rank d approximation of M^{OPT}:

$$W_{\mathrm{SVD}} \cdot C_{\mathrm{SVD}}^T = \underset{M|\mathrm{rk}(M)=d}{\arg\min} ||M - M^{\mathrm{OPT}}||_{\mathrm{F}}$$

Skip-Gram (without negative sampling)

Recall from previous slide:

$$\ell_{SG}(W, C) = \sum_{(w,c)\in D} \left(\vec{w} \cdot \vec{c} - \log \left(\sum_{c' \in V_C} \exp\left(\vec{w} \cdot \vec{c'} \right) \right) \right)$$

Computations for the skip-gram model (without negative sampling) yield a maximum for

$$(\vec{w} \cdot \vec{c})^{\text{OPT}} = \log \#(w, c)$$

Problems with SVD

$$M_{ij}^{\mathrm{OPT}} = \log\left(\frac{\#\left(w_i, c_j\right) \cdot |D|}{\#\left(w_i\right) \cdot \#\left(c_j\right)}\right) - \log k$$

Problems with SVD

$$M_{ij}^{\text{OPT}} = \log\left(\frac{\#(w_i, c_j)\cdot|D|}{\#(w_i)\cdot\#(c_j)}\right) - \log k$$

1. What about pairs with $\#(w_i, c_j) = 0$?
   (This is the case for more than 98% of our pairs!)
2. M^{OPT} is dense.

Problems with SVD

$$M_{ij}^{\mathrm{OPT}} = \log\left(\frac{\#(w_i, c_j)\cdot|D|}{\#(w_i)\cdot\#(c_j)}\right) - \log k$$

1. What about pairs with $\#(w_i, c_j) = 0$?
   (This is the case for more than 98% of our pairs!)
2. M^{OPT} is dense.

Solution: Factorize

$$M_{ij}^{+} = \max\left(\log\left(\frac{\#(w_i, c_j)\cdot|D|}{\#(w_i)\cdot\#(c_j)}\right) - \log k, 0\right)$$

Questions?

Table of Contents

Experiment Setup

data:	$\sim$ 4.6 million English Wikipedia articles
vocabulary size:	$\sim 160,000$
	(words that appeared at least 300 times)
window size:	2
word-context samples:	$\sim$ 9.7 billion
embedding dimension:	200

Table of Contents

Optimizing the Objective

The following table shows the percentage of deviation from the optimal value, that is

$$\frac{\ell - \ell^{\mathrm{OPT}}}{\ell^{\mathrm{OPT}}}.$$

k	ℓ^{OPT}	ℓ^{+}	SVD	NN
0	0%	5.7%	25.1%	-
1	0%	29.3%	38.8%	22.7%
5	0%	120.9%	124.7%	9.5%
15	0%	309.0%	310.4%	8.9%

Table: Percentage of deviation from the optimal objective value.

Table of Contents

Word Similarity Tasks

Models were tested to two datasets:

- WordSim353: 353 word pairs
- MEN: 3000 word pairs

word pairs		human assigned similarity scores
stock	market	8.08
physics	chemistry	7.35
game	round	5.97
experience	music	3.47
stock	jaguar	0.92

Table: Examples from the WordSim353 dataset

Word Similarity Tasks

	WordSim353		MEN	
k	NN	SVD	NN	SVD
0	-	0.601	-	0.655
1	0.524	0.613	0.588	0.700
5	0.658	0.536	0.712	0.669
15	0.644	0.400	0.681	0.606

Table: Spearman's correlation between dataset similarity scores and similarity scores that different the models returned.

Note: Spearman's correlation $\rho_S \in [-1, 1]$, where negative (positive) numbers indicate negative (positive) correlation and zero indicates no correlation.

More about Spearman's correlation

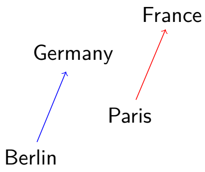
Table of Contents

Analogy Tasks

Berlin is to **Germany** as **Paris** is to **France**.

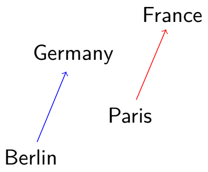
Analogy Tasks

Berlin is to **Germany** as **Paris** is to **France**.



Analogy Tasks

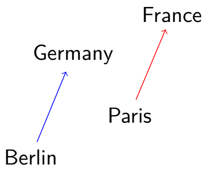
Berlin is to **Germany** as **Paris** is to **France**.



$$\Rightarrow \mathrm{vec}\left(\mathbf{Germany}\right) - \mathrm{vec}\left(\mathbf{Berlin}\right) = \mathrm{vec}\left(\mathbf{France}\right) - \mathrm{vec}\left(\mathbf{Paris}\right)$$

Analogy Tasks

Berlin is to **Germany** as **Paris** is to **France**.



$$\Rightarrow \text{vec}\left(\textbf{Germany}\right) - \text{vec}\left(\textbf{Berlin}\right) = \text{vec}\left(\textbf{France}\right) - \text{vec}\left(\textbf{Paris}\right)$$

in other words:

$$\text{vec}\left(\textbf{France}\right) = \text{vec}\left(\textbf{Germany}\right) - \text{vec}\left(\textbf{Berlin}\right) + \text{vec}\left(\textbf{Paris}\right)$$

Analogy Tasks

	Mixed dataset 19.500 analogies		**Syntactic dataset** 8.000 analogies	
k	NN	SVD	NN	SVD
0	-	26.8%	-	28.7%
1	27.3%	30.6%	32.3%	19.6%
5	51.0%	12.0%	51.0%	5.7%
15	53.2%	5.9%	47.9%	1.4%

Table: Percentage of correct answers on two word analogy datasets.

More examples

Questions?

Expectation of the closest vector

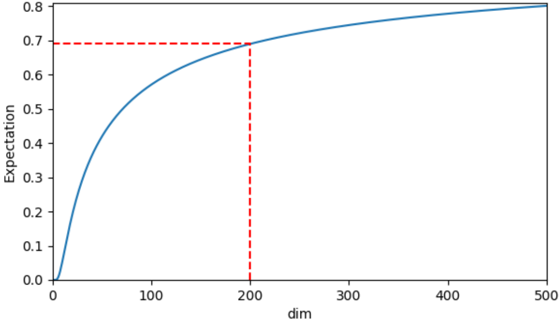


Figure: Expectation of the cosine distance to the nearest vector for $159,862$ vectors depending on the embedding dimension.

Back

Expectation of the closest vector

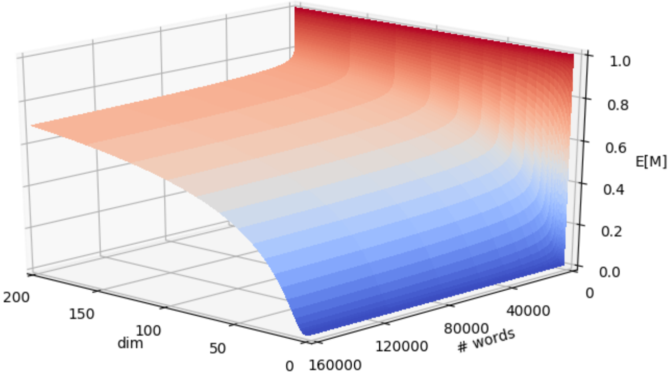
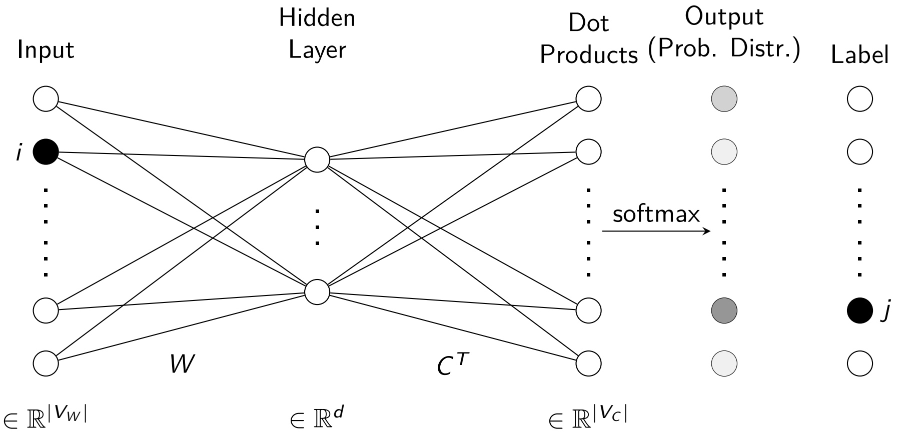


Figure: The expectation of the distance to the closest word depending on the embedding dimension and the number of words.

Back

Skip-Gram



Back

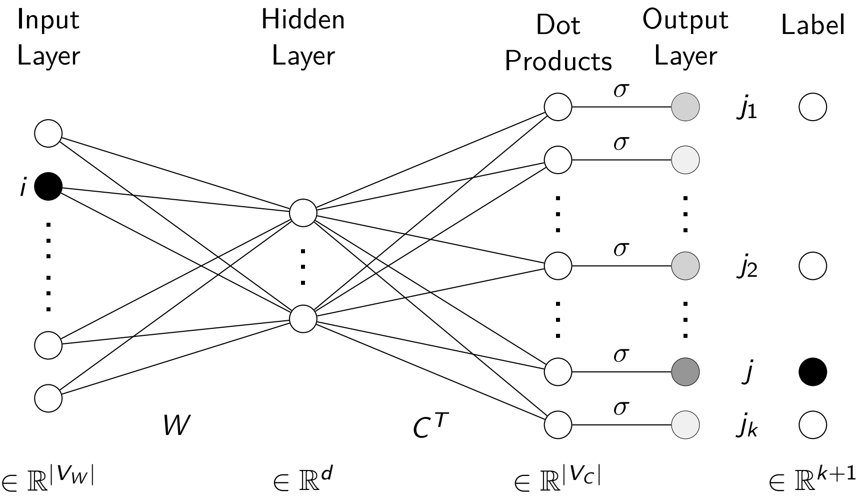
Objective function SG

$$\ell_{SG}(W, C) = \sum_{(w,c)\in D} \log \frac{\exp(\vec{w}\cdot\vec{c})}{\sum_{c'\in V_C} \exp\left(\vec{w}\cdot\vec{c'}\right)}$$

$$= \sum_{(w,c)\in D} \left(\vec{w}\cdot\vec{c} - \log\left(\sum_{c'\in V_C} \exp\left(\vec{w}\cdot\vec{c'}\right)\right)\right)$$

Back

Skip-Gram with negative sampling



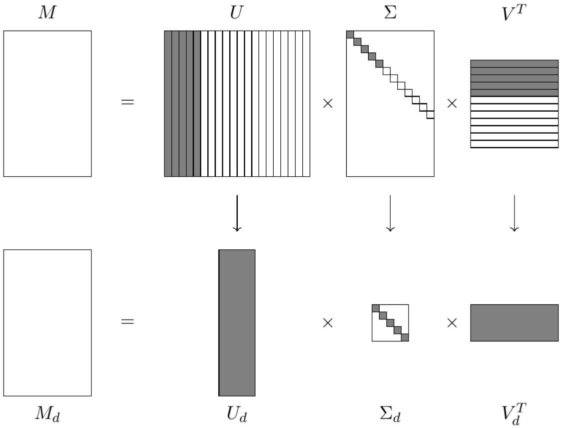
Back

Objective function SGNS

$$\ell_{SGNS}(W,C) = \sum_{(w_i,c_j)\in D} \left(\log\sigma(\vec{w}_i \cdot \vec{c}_j) + \sum_{l=1}^{k} \log(1-\sigma(\vec{w}_i \cdot \vec{c}_{j_l})) \right)$$
$$= \sum_{(w_i,c_j)\in D} \left(\log\sigma(\vec{w}_i \cdot \vec{c}_j) + \sum_{l=1}^{k} \log\sigma(-\vec{w}_i \cdot \vec{c}_{j_l}) \right)$$
$$\approx \sum_{(w,c)\in D} \left(\log\sigma(\vec{w} \cdot \vec{c}) + k \cdot \mathbb{E}_{c_N \sim \mathrm{P}_D} [\log\sigma(-\vec{w} \cdot \vec{c}_N)] \right)$$

Back

Truncated SVD

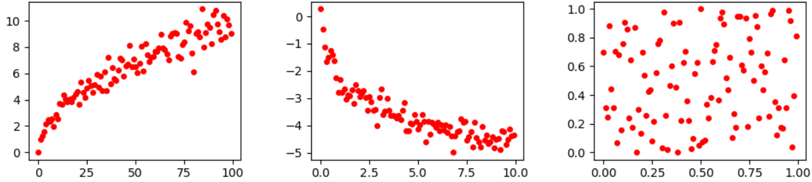


Back

Spearman correlation

Let X_i be the human-assigned scores and Y_i be the cosine similarity of the vectors. Then, the Spearman correlation is defined as

$$\rho_S = \frac{\operatorname{cov}\left(\operatorname{rg}\left(X\right), \operatorname{rg}\left(Y\right)\right)}{\sigma\left(\operatorname{rg}\left(X\right)\right)\sigma\left(\operatorname{rg}\left(Y\right)\right)} \in [-1, 1].$$



(a) positive (b) negative (c) around zero

Figure: Datasets with different Spearman correlation

Analogy Tasks

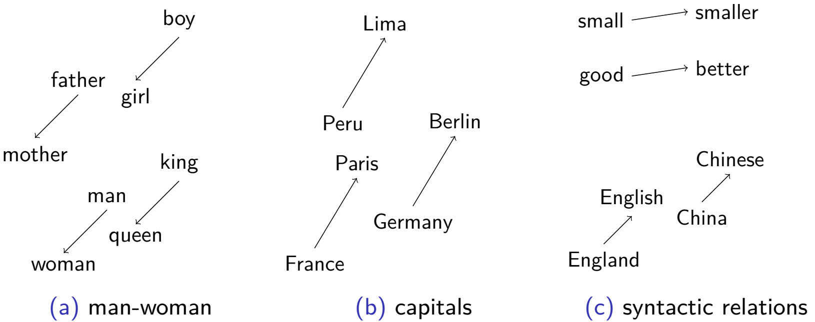
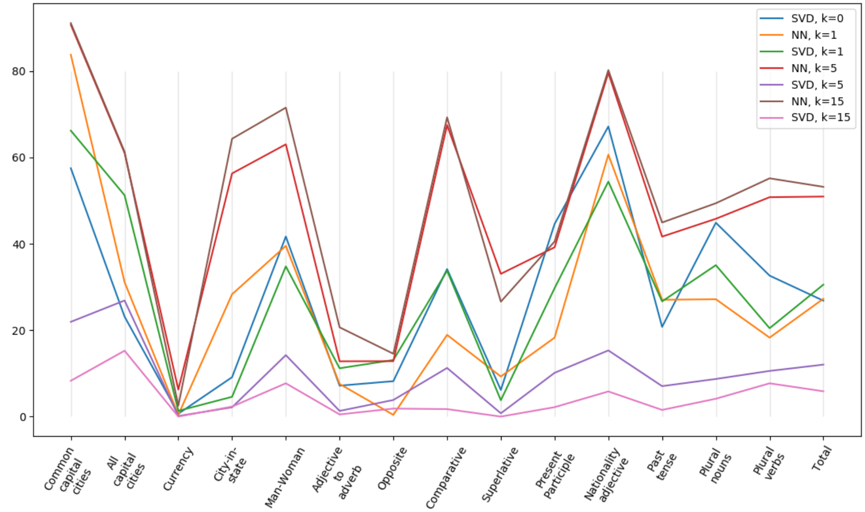


Figure: Examples of various relations between words

Back

Analogy Tasks



Back