CS 6501 Natural Language Processing

Word Embeddings

Yangfeng Ji

October 22, 2018

Department of Computer Science
University of Virginia

UNIVERSITY of VIRGINIA | ENGINEERING

Overview

1. Distributional Hypothesis
2. Latent Semantic Analysis
3. Word Embeddings

Distributional Hypothesis

Distributional Hypothesis

words that occur in the same contexts tend to have similar meanings

- `to have a splendid time in Rome`
- `to have a wonderful time in Rome`

Generalized Hypotheses

- **Statistical semantics hypothesis**: Statistical patterns of human word usage can be used to figure out what people mean.
- **Bag of words hypothesis**: The frequencies of words in a document tend to indicate the relevance of the document to a query

[Turney and Pantel, 2010]

Latent Semantic Analysis

Word-document Matrix

For a corpus of d documents over a vocabulary $\mathcal{V}$, the cooccurence matrix is defined as $\mathbf{C}$,

$$\mathbf{C} = [C_{ij}] \in \mathbb{R}^{v \times d}, \tag{1}$$

where $v = |\mathcal{V}|$ is the size of vocab, and C_{ij} is the count of word i in document j.

Word-document Matrix

Word	Documents							
	1	2	3	4	5	6	7	8
w_1	0	1	0	0	0	0	0	0
w_2	0	0	1	0	0	3	0	0
w_3	1	0	0	2	0	0	5	0
w_4	3	0	0	1	1	0	2	0
w_5	0	1	3	0	1	2	1	0
w_6	1	2	0	0	0	0	1	0
w_7	0	1	0	1	0	1	0	1
w_8	0	0	0	0	0	7	0	0

Similarity

If we consider a document as a context, we can use row vectors $\boldsymbol{c}_i$ to represent words and hence measure similarity between words as

$$\mathrm{sim}(\boldsymbol{c}_i, \boldsymbol{c}_j) = \frac{\langle \boldsymbol{c}_i, \boldsymbol{c}_j \rangle}{\|\boldsymbol{c}_i\| \|\boldsymbol{c}_j\|} \tag{2}$$

Data Sparsity

Word	Documents							
	1	2	3	4	5	6	7	8
w_1	0	1	0	0	0	0	0	0
w_2	0	0	1	0	0	3	0	0
w_3	1	0	0	2	0	0	5	0
w_4	3	0	0	1	1	0	2	0
w_5	0	1	3	0	1	2	1	0
w_6	1	2	0	0	0	0	1	0
w_7	0	1	0	1	0	1	0	1
w_8	0	0	0	0	0	7	0	0

- Vocab size: 50K, 100K, etc.

Context Window Size

Word	Documents							
	1	2	3	4	5	6	7	8
w_1	0	1	0	0	0	0	0	0
w_2	0	0	1	0	0	3	0	0
w_3	1	0	0	2	0	0	5	0
w_4	3	0	0	1	1	0	2	0
w_5	0	1	3	0	1	2	1	0
w_6	1	2	0	0	0	0	1	0
w_7	0	1	0	1	0	1	0	1
w_8	0	0	0	0	0	7	0	0

Are w_i and w_j similar to each other, when they appear in the same documents but far away from each other?

Solution to Data Sparsity

Using SVD, the matrix $\mathbf{C}$ is decomposed into a multiplication of three matrices

$$\mathbf{C} = \mathbf{U}_0 \mathbf{\Sigma}_0 \mathbf{V}_0^\top. \tag{3}$$

where $\mathbf{U}_0 \in \mathbb{R}^{v \times v}$, $\mathbf{\Sigma}_0 \in \mathbb{R}^{v \times d}$ is a diagonal matrix and $\mathbf{V}_0 \in \mathbb{R}^{d \times d}$.

The columns of $\mathbf{U}_0$ (and $\mathbf{V}_0$) are orthonormal.

SVD

the approximation can be written as

$$\mathbf{C} \approx \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top \tag{4}$$

where $\mathbf{U} \in \mathbb{R}^{v \times k}$, $\mathbf{V} \in \mathbb{R}^{k \times d}$ and $\boldsymbol{\Sigma} \in \mathbb{R}^{k \times k}$.

Optimization

Mathematically, if we are looking for a rank-k approximation of $\mathbf{C}$ as,

$$\arg\min_{\mathbf{M}} \|\mathbf{M} - \mathbf{C}\|_2 \tag{5}$$

where $\mathbf{M}$ is rank-k,

Based on 4, the word representation can be constructed as

$$\mathbf{W} = \mathbf{U}\mathbf{\Sigma} \tag{6}$$

and document representation as

$$\mathbf{D} = \mathbf{\Sigma}\mathbf{V} \tag{7}$$

Re-weighting

- TF-IDF
- Pointwise Mutual Information: the definition of $\mathrm{PMI}(w_i, w_j)$ is

$$\mathrm{PMI}(w_i, w_j) = \log \frac{P(w_i, w_j)}{P(w_i)P(w_j)} = \log \frac{P(w_j \mid w_i)}{P(w_j)} \quad (8)$$

Word Embeddings

Skip-gram

One way of finding a better word representation is to make sure it has the potential to predict its surrounding words

$$P(w_{t+i} \mid w_t; \boldsymbol{\theta}) = \frac{\exp(\boldsymbol{u}_{w_{t+i}}^\top \boldsymbol{v}_{w_t})}{\sum_{w' \in \mathcal{V}} \exp(\boldsymbol{u}_{w'}^\top \boldsymbol{v}_{w_t})} \tag{9}$$

where $i \in \{-c, \ldots, -1, 1, \ldots, c\}$ and c is the window size. Usually, larger window size c gives better quality of word representations, but it also causes large computational complexity.

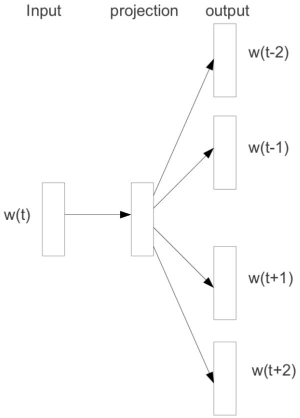


Figure: The skip-gram model.

Word Vectors

$$P(w_{t+i} \mid w_t; \boldsymbol{\theta}) = \frac{\exp(\boldsymbol{u}_{w_{t+i}}^\top \boldsymbol{v}_{w_t})}{\sum_{w' \in \mathcal{V}} \exp(\boldsymbol{u}_{w'}^\top \boldsymbol{v}_{w_t})} \tag{10}$$

- $\boldsymbol{v}_w$: word vector (as input)
- $\boldsymbol{u}_w$: context vector (as output)

Question

Why we need two vectors for a word?

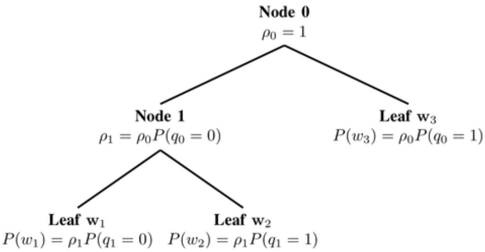
Objective Function

The objective function of a skip-gram model is defined as

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq i\leq c;i\neq 0}\log P(w_{t+i}\mid w_t) \tag{11}$$

In practice, it could be 10K, 50K or even bigger and the computation of $\sum_{w'\in\mathcal{V}}\exp(\boldsymbol{v}_{w'}^{\top}\boldsymbol{u}_{w_t})$ can be very expensive.

Hierarchical Softmax



- Huffman tree based word frequency

Negative Sampling

Negative sampling of the skip-gram model is defined as, for a specific w_{t+i} to predict, we have

$$\log \sigma(\boldsymbol{u}_{w_{t+i}}^\top \boldsymbol{v}_{w_t}) + \sum_{i=1}^{k} E_{w' \sim P_n(w)} \left[\log \sigma(-\boldsymbol{u}_{w'}^\top \boldsymbol{v}_{w_t}) \right] \qquad (12)$$

Two Factors in Negative Sampling

$$\log \sigma(\boldsymbol{u}_{w_{t+i}}^{\top} \boldsymbol{v}_{w_t}) + \sum_{i=1}^{k} E_{w' \sim P_n(w)} \left[\log \sigma(-\boldsymbol{u}_{w'}^{\top} \boldsymbol{v}_{w_t}) \right] \qquad (13)$$

Two factors [Mikolov et al., 2013]

- $k = ?$
  - $5 \leq k \leq 20$ works better for small datasets, while for large datasets, $2 \leq k \leq 5$ is enough
- noisy distribution $P(w)$
  - $P(w) \propto U(w)^{\frac{3}{4}}$

Glove

The motivation of GloVe [Pennington et al., 2014] is to find a balance between the methods based on

- global matrix factorization (e.g., LSA) and
- local context windows (e.g., Skip-gram).

Word-to-word Co-occurrence Matrix

- **X** with X_{ij} denotes the frequency of word j appears in the context of word i

Empirical probability estimation of w_j given w_i

$$Q(w_j \mid w_i) = \frac{X_{ij}}{X_i} \tag{14}$$

Probability Estimation via Word Embeddings

$$P(w_j \mid w_i) = \frac{\exp(\boldsymbol{u}_{w_j}^\top \boldsymbol{v}_{w_i})}{\sum_{w' \in \mathcal{V}} \exp(\boldsymbol{u}_{w'}^\top \boldsymbol{v}_{w_i})} \tag{15}$$

The basic idea is to learn $\boldsymbol{v}_w$ and $\boldsymbol{u}_w$, such that

$$Q(w_j \mid w_i) \approx P(w_j \mid w_i) \tag{16}$$

The basic idea is to learn $\boldsymbol{v}_w$ and $\boldsymbol{u}_w$, such that

$$Q(w_j \mid w_i) \approx P(w_j \mid w_i) \tag{16}$$

Or more specific

$$\log(X_{ij}) - \log(X_i) \approx \boldsymbol{u}_{w_j}^\top \boldsymbol{v}_{w_i} - \log \sum_{w' \in \mathcal{V}} \exp(\boldsymbol{u}_{w'}^\top \boldsymbol{v}_{w_i}) \tag{17}$$

In order to find the best approximation, we could formulate this as a optimization problem

$$(\log(X_{ij}) - \log(X_i) - \boldsymbol{u}_{w_j}^\top \boldsymbol{v}_{w_i} + \log \sum_{w' \in \mathcal{V}} \exp(\boldsymbol{u}_{w'}^\top \boldsymbol{v}_{w_i})))^2 \quad (18)$$

In order to find the best approximation, we could formulate this as a optimization problem

$$(\log(X_{ij}) - \log(X_i) - \boldsymbol{u}_{w_j}^\top \boldsymbol{v}_{w_i} + \log \sum_{w' \in \mathcal{V}} \exp(\boldsymbol{u}_{w'}^\top \boldsymbol{v}_{w_i})))^2 \quad (18)$$

It can be further simplified as (Eq. 16 in [Pennington et al., 2014])

$$(\log(X_{ij}) - \boldsymbol{u}_{w_j}^\top \boldsymbol{v}_{w_i})^2 \quad (19)$$

if we only consider the *unnormalized* version of P and Q.

Objective Function

The overall objective function is defined as

$$\sum_{w_i} \sum_{w_j} (\log(X_{ij}) - \boldsymbol{u}_{w_j}^{\top} \boldsymbol{v}_{w_i})^2 \tag{20}$$

Objective Function

The overall objective function is defined as

$$\sum_{w_i} \sum_{w_j} (\log(X_{ij}) - \boldsymbol{u}_{w_j}^{\top} \boldsymbol{v}_{w_i})^2 \tag{20}$$

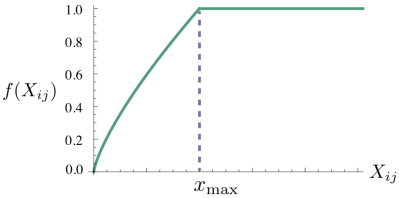
The objective function is further refined by discouraging high-frequency words as

$$\sum_{w_i} \sum_{w_j} f(X_{ij})(\log(X_{ij}) - \boldsymbol{u}_{w_j}^{\top} \boldsymbol{v}_{w_i})^2 \tag{21}$$

Down-weighting

$$f(x) = \begin{cases} (\frac{x}{x_{\max}})^{a} & \text{if } x < x_{\max} \\ 1 & \text{otherwise} \end{cases} \tag{22}$$

where $a = 3/4$.



Skip-gram as Implicit Matrix Factorization

[Levy and Goldberg, 2014] shows that skip-gram with negative sampling can be viewed as an implicit matrix factorization over a word-word co-occurrence matrix weighted by pointwise mutual information (PMI).

$$\boldsymbol{u}_{w_j}^{\top}\boldsymbol{v}_{w_i} \approx \mathrm{PMI}(w_i, w_j) - \log k \tag{23}$$

where $\mathrm{PMI}(w_i, w_j)$ is the mutual information of $P(w_i)$ and $P(w_j)$ with *a given window size* and k is the number of negative samples.

Skip-gram as Implicit Matrix Factorization (II)

The definition of $\mathrm{PMI}(w_i, w_j)$ is

$$\mathrm{PMI}(w_i, w_j) = \log \frac{P(w_i, w_j)}{P(w_i)P(w_j)} = \log P(w_j \mid w_i) - \log P(w_j) \tag{24}$$

Combine 23 and 24, we have

$$\begin{aligned} \boldsymbol{u}_{w_j}^{\top} \boldsymbol{v}_{w_i} &\approx \log \frac{P(w_i, w_j)}{P(w_i)P(w_j)} - \log k \\ &= \log P(w_j \mid w_i) - \log P(w_j) - \log k \\ &= \log(X_{ij}) - \log(X_i) - \log(X_j) + \log D - \log k \end{aligned} \tag{25}$$

Very similar to Eq. 8 in [Pennington et al., 2014].

Essentially,

$$\boldsymbol{u}_{w_j}^{\top}\boldsymbol{v}_{w_i} \approx \log(X_{ij}) + g(\mathbf{X}) \tag{26}$$

Essentially,

$$\boldsymbol{u}_{w_j}^{\top}\boldsymbol{v}_{w_i} \approx \log(X_{ij}) + g(\mathbf{X}) \tag{26}$$

Which one matters?

- $g(\mathbf{X})$, or
- Implicit/explicit optimization, or
- Other tricks (down-sampling, hyper-parameters, etc.)

Summary

1. Distributional Hypothesis
2. Latent Semantic Analysis
3. Word Embeddings

Reference

Levy, O. and Goldberg, Y. (2014).
Neural word embedding as implicit matrix factorization.
In *Advances in neural information processing systems*, pages 2177–2185.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013).
Distributed representations of words and phrases and their compositionality.
In *Advances in neural information processing systems*, pages 3111–3119.

Pennington, J., Socher, R., and Manning, C. (2014).
Glove: Global vectors for word representation.
In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Turney, P. D. and Pantel, P. (2010).
From frequency to meaning: Vector space models of semantics.
Journal of artificial intelligence research, 37:141–188.