Word Embeddings: Establishing A Coherence Between A Word and Its Context

Dr. Tiansi Dong, Prof. Dr. Christian Bauckhage

dongt@bit.uni-bonn.de, christian.bauckhage@iais.fraunhofer.de

March 2019, IPEC Winter School 2019
B-IT, University of Bonn
System 1 manages to construct a coherent story of data
System 1 manages to construct a coherent story of data

Using Neural-Networks To Simulate System 1
Introduction

- System 1 manages to construct a coherent story of data
- Using Neural-Networks To Simulate System 1
- Given texts, can we establish coherences between words and their contexts?
System 1 manages to construct a coherent story of data

Using Neural-Networks To Simulate System 1

Given texts, can we establish coherences between words and their contexts?

First is the representation problem: How shall we represent words, contexts?
Introduction

- System 1 manages to construct a coherent story of data
- Using Neural-Networks To Simulate System 1
- Given texts, can we establish coherences between words and their contexts?
- First is the representation problem: How shall we represent words, contexts?
- How shall we measure the coherence between words and contexts?
System 1 manages to construct a coherent story of data
Using Neural-Networks To Simulate System 1
Given texts, can we establish coherences between words and their contexts?
First is the representation problem: How shall we represent words, contexts?
How shall we measure the coherence between words and contexts?
Word Embeddings, [Mikolov et al., 2013, Levy and Goldberg, 2014, Pennington et al., 2014]
Word Embeddings

- Represent each word as a vector
Word Embeddings

- Represent each word as a vector
- A context is a set of words
Word Embeddings

- Represent each word as a vector
- A context is a set of words
- Represent a context as the sum of its word vectors
Word Embeddings

- Represent each word as a vector
- A context is a set of words
- Represent a context as the sum of its word vectors
- The coherence between a word and a context is measured by the cosine value of their vectors
Input?

Text. That is a sequence of words $[w_1, \ldots, w_M]$. Vector representation of word $w_i^k$ represents the $k$-th word in the text, meanwhile the $i$-th word in the vocabulary $1 \leq k \leq M, 1 \leq i \leq V$. The vector representation of $w_i^k$ is a vector of dimension $V$, whose $i$-th element is '1', other elements are '0'.
Learning Word Embeddings

- Input?
- Text. That is a sequence of words $[w_1, \ldots, w_M]$
Learning Word Embeddings

- Input?
- Text. That is a sequence of words \([w_1, \ldots, w_M]\)
- Vector representation of word?
Learning Word Embeddings

- Input?
- Text. That is a sequence of words \([w_1, \ldots, w_M]\)
- Vector representation of word?
  - Random, e.g., fix the dimension of vector, each element is randomly initialised
Learning Word Embeddings

- Input?
- Text. That is a sequence of words \([w_1, \ldots, w_M]\)
- Vector representation of word?
  - Random, e.g., fix the dimension of vector, each element is randomly initialised
  - One-shot. Vocabulary with size \(V\), each word has a fixed position in the vocabulary. \(w_k^i\) represents the \(k\)-th word in the text, meanwhile the \(i\)-th word in the vocabulary \(1 \leq k \leq M, 1 \leq i \leq V\). The vector representation of \(w_k^i\) is a vector of dimension \(V\), whose \(i\)-th element is ‘1’, other elements are ‘0’
A vocabulary only has 3 words: $A, B, C$, ordered as $[A, B, C]$. 

A text $CBAABBAACA$ is then represented as a matrix:

$$
\begin{pmatrix}
0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\end{pmatrix}
$$
A vocabulary only has 3 words: $A$, $B$, $C$, ordered as $[A, B, C]$.

The vector representation of $A$ is $[1, 0, 0]^\top$, the vector representation of $B$ is $[0, 1, 0]^\top$, the vector representation of $C$ is $[0, 0, 1]^\top$. 

A text sequence like $\text{CBAABBAACA}$ is then represented as a matrix:

$$
\begin{bmatrix}
0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\end{bmatrix}
$$
A vocabulary only has 3 words: $A, B, C$, ordered as $[A, B, C]$.

The vector representation of $A$ is $[1, 0, 0]^\top$, the vector representation of $B$ is $[0, 1, 0]^\top$, the vector representation of $C$ is $[0, 0, 1]^\top$.

A text $CBAABBAACA$ is then represented as a matrix

$$
\begin{bmatrix}
0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{bmatrix}
$$
Fix $W$ and $W'$ satisfying $Y \approx \text{softmax}(W' \cdot f(W \cdot X))$. 

$w \cdot i$ is the vector representation of the $i$-th word.
Predict The Next Word From The Current Word

- Fix $W$ and $W'$ satisfying $Y \approx \text{softmax}(W' \cdot f(W \cdot X))$

- $w_{i}$ is the vector representation of the $i$-th word
We understood it as a method of counting for the voting result.
The Softmax Method

- We understood it as a method of counting for the voting result.
- Given current word $w_c$, which word can be the next word?
The Softmax Method

- We understood it as a method of counting for the voting result
- Given current word $w_c$, which word can be the next word?
- A neural-network will produce a vector, e.g. $[0.1, -0.5, 0, 0.001, 1.2, 3.1]$
The Softmax Method

- We understood it as a method of counting for the voting result.
- Given current word $w_c$, which word can be the next word?
- A neural-network will produce a vector, e.g. 
  $[0.1, -0.5, 0, 0.001, 1.2, 3.1]$
- We believe
The Softmax Method

- We understood it as a method of counting for the voting result.
- Given current word $w_c$, which word can be the next word?
- A neural-network will produce a vector, e.g. $[0.1, -0.5, 0, 0.001, 1.2, 3.1]$
- We believe
  - *every word is possible*
The Softmax Method

- We understood it as a method of counting for the voting result.
- Given current word $w_c$, which word can be the next word?
- A neural-network will produce a vector, e.g. $[0.1, -0.5, 0, 0.001, 1.2, 3.1]$.
- We believe
  - *every word is possible*
  - *every voting weight is greater than 0*
The Softmax Method

- We understood it as a method of counting for the voting result.
- Given current word $w_c$, which word can be the next word?
- A neural-network will produce a vector, e.g.
  \[ [0.1, -0.5, 0, 0.001, 1.2, 3.1] \]
- We believe:
  - every word is possible
  - every voting weight is greater than 0
  - the sum of all voting weights equals to 1
The Softmax Method

- We understood it as a method of counting for the voting result.
- Given current word $w_c$, which word can be the next word?
- A neural-network will produce a vector, e.g. $[0.1, -0.5, 0, 0.001, 1.2, 3.1]$
- We believe:
  - every word is possible
  - every voting weight is greater than 0
  - the sum of all voting weights equals to 1
- Transform raw voting weight $r_i$ into the power of $e^{r_i}$
The Softmax Method

- We understood it as a method of counting for the voting result.
- Given current word $w_c$, which word can be the next word?
- A neural-network will produce a vector, e.g. $[0.1, -0.5, 0, 0.001, 1.2, 3.1]$
- We believe
  - every word is possible
  - every voting weight is greater than 0
  - the sum of all voting weights equals to 1
- Transform raw voting weight $r_i$ into the power of $e^{r_i}$
- Normalise by dividing the sum of all $e^{r_i}$s.
The Softmax Method

A neural-network will produce a vector \([r_1, r_2, \ldots, r_V]\)
A neural-network will produce a vector \([r_1, r_2, \ldots, r_V]\)

The softmax value of \(r_i\) is

\[
\frac{e^{r_i}}{\sum_{k=1}^{V} e^{r_k}}
\]  \hspace{1cm} (1)
Predict The Next Word From The Current Word

- Train $\mathbf{W}$ and $\mathbf{W}'$ by utilising the Back-Propagation Algorithm [Rumelhart et al., 1986]
Train $\mathbf{W}$ and $\mathbf{W}'$ by utilising the Back-Propagation Algorithm [Rumelhart et al., 1986]

$w \cdot i$ in $\mathbf{W}$ is the word embedding of the $i$-th word.
Predict The Word in The Continuous Bag-of-Word Context

- given a sequence of words $w_{i-2}w_{i-1}w_iw_{i+1}w_{i+2}$, a context of $w_i$ is $w_{i-2}w_{i-1}w_{i+1}w_{i+2}$
Predict The Word in The Continuous Bag-of-Word Context

- given a sequence of words $w_{i-2} w_{i-1} w_i w_{i+1} w_{i+2}$, a context of $w_i$ is $w_{i-2} w_{i-1} w_{i+1} w_{i+2}$

CBOw: Input $X = w_{i-2} + w_{i-1} + w_{i+1} + w_{i+2}$. 
Reading and Practicing

- Word2Vec
  https://code.google.com/archive/p/word2vec/
  [Mikolov et al., 2013]
Reading and Practicing

- **Word2Vec**
  https://code.google.com/archive/p/word2vec/
  [Mikolov et al., 2013]

- **GloVe: Global Vectors for Word Representation**
  https://nlp.stanford.edu/projects/glove/
  [Pennington et al., 2014]
Reading and Practicing

- **Word2Vec**
  https://code.google.com/archive/p/word2vec/
  [Mikolov et al., 2013]

- **GloVe: Global Vectors for Word Representation**
  https://nlp.stanford.edu/projects/glove/
  [Pennington et al., 2014]

- **Dependency-Based Word Embeddings**
  https://levyomer.wordpress.com/2014/04/25/dependency-based-word-embeddings/
  [Levy and Goldberg, 2014]
Each word $w$ has a vector representation $\mathbf{w}$.
Evaluation fo Word Embeddings

- Each word $w$ has a vector representation $\vec{w}$
- Is it true that similar words have similar vectors?
Evaluation fo Word Embeddings

- Each word $w$ has a vector representation $\vec{w}$
- Is it true that similar words have similar vectors?
- Compute $\cos(\vec{bonn}, \vec{berlin})$, $\cos(\vec{bonn}, \vec{moon})$,
  $\cos(\vec{bonn}, \vec{wien})$, $\cos(\vec{bonn}, \vec{pinguin})$
Evaluation of Word Embeddings

- Each word $w$ has a vector representation $\vec{w}$
- Is it true that similar words have similar vectors?
- Compute $\cos(\vec{\text{bonn}}, \vec{\text{berlin}})$, $\cos(\vec{\text{bonn}}, \vec{\text{moon}})$, $\cos(\vec{\text{bonn}}, \vec{\text{wien}})$, $\cos(\vec{\text{bonn}}, \vec{\text{pinguin}})$
- Compute $\cos(\vec{\text{bonn}}, \vec{\text{city}})$, $\cos(\vec{\text{moon}}, \vec{\text{satellite}})$, $\cos(\vec{\text{wien}}, \vec{\text{capital}})$, $\cos(\vec{\text{pinguin}}, \vec{\text{animal}})$
Word Embeddings shall establish a coherence between a word and its context.
Test Word Embeddings and Report

- Word Embeddings shall establish a coherence between a word and its context
- What problems do you find?
References


