From Frequency to Meaning: Vector Space Models of Semantics

Yifu Huang

School of Computer Science, Fudan University
huangyifu@fudan.edu.cn

COMP620028 Information Retrieval Report, 2013
Outline

1. Matrices
   - The Word – Context Matrix
   - The Pair – Pattern Matrix

2. Weighting the Elements
   - Positive PMI

3. Smoothing the Matrices
   - Truncated SVD

4. Efficient Comparisons
   - LSH

5. Applications
   - The Word – Context Matrix
   - The Pair – Pattern Matrix
Outline

1. Matrices
   • The Word – Context Matrix
   • The Pair – Pattern Matrix

2. Weighting the Elements
   • Positive PMI

3. Smoothing the Matrices
   • Truncated SVD

4. Efficient Comparisons
   • LSH

5. Applications
   • The Word – Context Matrix
   • The Pair – Pattern Matrix
Distributional hypothesis [2]

- Words that occur in similar contexts tend to have similar meanings
- “vague”, “obscure”

The context is given by words, phrases, sentences, paragraphs, chapters, documents, or more exotic possibilities, such as sequences of characters or patterns [3]
Distributional hypothesis [2]

- Words that occur in similar contexts tend to have similar meanings
- “vague”, “obscure”

The context is given by words, phrases, sentences, paragraphs, chapters, documents, or more exotic possibilities, such as sequences of characters or patterns [3]
Example.

Whereof one cannot speak thereof one must be silent

<table>
<thead>
<tr>
<th>Word</th>
<th>Co-occurrences</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>thereof</td>
</tr>
<tr>
<td>thereof</td>
<td>0</td>
</tr>
<tr>
<td>one</td>
<td>1</td>
</tr>
<tr>
<td>cannot</td>
<td>0</td>
</tr>
<tr>
<td>speak</td>
<td>0</td>
</tr>
<tr>
<td>thereof</td>
<td>0</td>
</tr>
<tr>
<td>must</td>
<td>0</td>
</tr>
<tr>
<td>be</td>
<td>0</td>
</tr>
<tr>
<td>silent</td>
<td>0</td>
</tr>
</tbody>
</table>
Outline

1 Matrices
   - The Word – Context Matrix
   - The Pair – Pattern Matrix

2 Weighting the Elements
   - Positive PMI

3 Smoothing the Matrices
   - Truncated SVD

4 Efficient Comparisons
   - LSH

5 Applications
   - The Word – Context Matrix
   - The Pair – Pattern Matrix
The Pair – Pattern Matrix

- **Extended distributional hypothesis [4]**
  - Patterns that co-occur with similar pairs tend to have similar meanings
  - “X solves Y”, “Y is solved by X”

- **Latent relation hypothesis [5]**
  - Pairs of words that co-occur in similar patterns tend to have similar semantic relations
  - “committee:problem”, “congress:crisis”
The Pair – Pattern Matrix

- Extended distributional hypothesis [4]
  - Patterns that co-occur with similar pairs tend to have similar meanings
  - “X solves Y”, “Y is solved by X”

- Latent relation hypothesis [5]
  - Pairs of words that co-occur in similar patterns tend to have similar semantic relations
  - “committee:problem”, “congress:crisis”
Example.

Committee finds a solution to problem
Strike is solved by committee
Congress finds a solution to crisis
Congress solves crisis
Civil war is solved by committee

<table>
<thead>
<tr>
<th>Pair</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>X finds a solution to Y</td>
</tr>
<tr>
<td>committee:problem</td>
<td>1</td>
</tr>
<tr>
<td>strike:committee</td>
<td>0</td>
</tr>
<tr>
<td>congress:crisis</td>
<td>1</td>
</tr>
<tr>
<td>civil war:committee</td>
<td>0</td>
</tr>
</tbody>
</table>
Outline

1 Matrices
   - The Word – Context Matrix
   - The Pair – Pattern Matrix

2 Weighting the Elements
   - Positive PMI

3 Smoothing the Matrices
   - Truncated SVD

4 Efficient Comparisons
   - LSH

5 Applications
   - The Word – Context Matrix
   - The Pair – Pattern Matrix
Positive PMI

- An alternative to tf-idf which works well for both word-context matrices [6] and pair-pattern matrices [5]
- $F$ be a word-context frequency matrix with $n_r$ rows and $n_c$ columns
- $f_{ij}$ is the number of times that word $w_i$ occurs in the context $c_j$
- $X$ be the matrix that results when Positive PMI is applied to $F$
- $x_{ij}$ in $X$ is defined as follows
Positive PMI

- An alternative to tf-idf which works well for both word-context matrices [6] and pair-pattern matrices [5]
- Let $F$ be a word-context frequency matrix with $n_r$ rows and $n_c$ columns
- $f_{ij}$ is the number of times that word $w_i$ occurs in the context $c_j$
- Let $X$ be the matrix that results when Positive PMI is applied to $F$
- $x_{ij}$ in $X$ is defined as follows
Positive PMI

- An alternative to tf-idf which works well for both word-context matrices [6] and pair-pattern matrices [5]
- Let $F$ be a word-context frequency matrix with $n_r$ rows and $n_c$ columns
- $f_{ij}$ is the number of times that word $w_i$ occurs in the context $c_j$
- Let $X$ be the matrix that results when Positive PMI is applied to $F$
- $x_{ij}$ in $X$ is defined as follows
An alternative to tf-idf which works well for both word-context matrices [6] and pair-pattern matrices [5]

- F be a word-context frequency matrix with \( n_r \) rows and \( n_c \) columns
- \( f_{ij} \) is the number of times that word \( w_i \) occurs in the context \( c_j \)
- X be the matrix that results when Positive PMI is applied to F
  - \( x_{ij} \) in X is defined as follows
An alternative to tf-idf which works well for both word-context matrices [6] and pair-pattern matrices [5].

Let $F$ be a word-context frequency matrix with $n_r$ rows and $n_c$ columns.

- $f_{ij}$ is the number of times that word $w_i$ occurs in the context $c_j$.
- $X$ be the matrix that results when Positive PMI is applied to $F$.
- $x_{ij}$ in $X$ is defined as follows.
Positive PMI (cont.)

\[ p_{ij} = \frac{f_{ij}}{\sum_{i=1}^{n_r} \sum_{j=1}^{n_c} f_{ij}} \]

\[ p_{i*} = \frac{\sum_{j=1}^{n_c} f_{ij}}{\sum_{i=1}^{n_r} \sum_{j=1}^{n_c} f_{ij}} \]

\[ p_{*j} = \frac{\sum_{i=1}^{n_r} f_{ij}}{\sum_{i=1}^{n_r} \sum_{j=1}^{n_c} f_{ij}} \]

\[ pmi_{ij} = \log \left( \frac{p_{ij}}{p_{i*} p_{*j}} \right) \]

\[ x_{ij} = \begin{cases} 
  pmi_{ij} & \text{if } pmi_{ij} > 0 \\
  0 & \text{otherwise}
\end{cases} \]
Outline

1. Matrices
   - The Word – Context Matrix
   - The Pair – Pattern Matrix

2. Weighting the Elements
   - Positive PMI

3. Smoothing the Matrices
   - Truncated SVD

4. Efficient Comparisons
   - LSH

5. Applications
   - The Word – Context Matrix
   - The Pair – Pattern Matrix
An elegant way to improve similarity measurements can be applied to both documents [7] and words [8]

SVD decomposes $X$ into the product of three matrices $U \Sigma V^T$

$U$ and $V$ are in column orthonormal form, and $\Sigma$ is a diagonal matrix of singular values

If $X$ is of rank $r$, then $\Sigma$ is also of rank $r$, let $k < r$

$\Sigma_k$, the diagonal matrix formed from the top $k$ singular values

$U_k$ and $V_k$, the matrices produced by selecting the corresponding columns from $U$ and $V$

$\tilde{X} = U_k \Sigma_k V_k^T$ is the matrix of rank $k$ that best approximates the original matrix $X$
An elegant way to improve similarity measurements can be applied to both documents [7] and words [8].

SVD decomposes $X$ into the product of three matrices $U \Sigma V^T$.

- $U$ and $V$ are in column orthonormal form, and $\Sigma$ is a diagonal matrix of singular values.
- If $X$ is of rank $r$, then $\Sigma$ is also of rank $r$, let $k < r$.
- $\Sigma_k$, the diagonal matrix formed from the top $k$ singular values.
- $U_k$ and $V_k$, the matrices produced by selecting the corresponding columns from $U$ and $V$.
- $\tilde{X} = U_k \Sigma_k V_k^T$ is the matrix of rank $k$ that best approximates the original matrix $X$. 

Yifu Huang

VSM review
An elegant way to improve similarity measurements can be applied to both documents [7] and words [8].

SVD decomposes $X$ into the product of three matrices $U \Sigma V^T$.

$U$ and $V$ are in column orthonormal form, and $\Sigma$ is a diagonal matrix of singular values.

If $X$ is of rank $r$, then $\Sigma$ is also of rank $r$, let $k < r$.

$\Sigma_k$, the diagonal matrix formed from the top $k$ singular values.

$U_k$ and $V_k$, the matrices produced by selecting the corresponding columns from $U$ and $V$.

$\tilde{X} = U_k \Sigma_k V_k^T$ is the matrix of rank $k$ that best approximates the original matrix $X$. 
Truncated SVD

- An elegant way to improve similarity measurements can be applied to both documents [7] and words [8].
- SVD decomposes $X$ into the product of three matrices $U\Sigma V^T$.
- $U$ and $V$ are in column orthonormal form, and $\Sigma$ is a diagonal matrix of singular values.
- If $X$ is of rank $r$, then $\Sigma$ is also of rank $r$, let $k < r$.
  - $\Sigma_k$, the diagonal matrix formed from the top $k$ singular values.
  - $U_k$ and $V_k$, the matrices produced by selecting the corresponding columns from $U$ and $V$.
  - $\tilde{X} = U_k\Sigma_k V_k^T$ is the matrix of rank $k$ that best approximates the original matrix $X$. 
An elegant way to improve similarity measurements can be applied to both documents [7] and words [8].

SVD decomposes $X$ into the product of three matrices $U \Sigma V^T$.

$U$ and $V$ are in column orthonormal form, and $\Sigma$ is a diagonal matrix of singular values.

If $X$ is of rank $r$, then $\Sigma$ is also of rank $r$, let $k < r$.

$\Sigma_k$, the diagonal matrix formed from the top $k$ singular values.

$U_k$ and $V_k$, the matrices produced by selecting the corresponding columns from $U$ and $V$.

$\tilde{X} = U_k \Sigma_k V_k^T$ is the matrix of rank $k$ that best approximates the original matrix $X$. 
Truncated SVD

- An elegant way to improve similarity measurements can be applied to both documents [7] and words [8].
- SVD decomposes $X$ into the product of three matrices $U \Sigma V^T$.
- $U$ and $V$ are in column orthonormal form, and $\Sigma$ is a diagonal matrix of singular values.
- If $X$ is of rank $r$, then $\Sigma$ is also of rank $r$, let $k < r$.
- $\Sigma_k$, the diagonal matrix formed from the top $k$ singular values.
- $U_k$ and $V_k$, the matrices produced by selecting the corresponding columns from $U$ and $V$.
- $\tilde{X} = U_k \Sigma_k V_k^T$ is the matrix of rank $k$ that best approximates the original matrix $X$. 

Yifu Huang  
VSM review
Truncated SVD

- An elegant way to improve similarity measurements can be applied to both documents [7] and words [8]
- SVD decomposes $X$ into the product of three matrices $U \Sigma V^T$
- $U$ and $V$ are in column orthonormal form, and $\Sigma$ is a diagonal matrix of singular values
- If $X$ is of rank $r$, then $\Sigma$ is also of rank $r$, let $k < r$
- $\Sigma_k$, the diagonal matrix formed from the top $k$ singular values
- $U_k$ and $V_k$, the matrices produced by selecting the corresponding columns from $U$ and $V$
- $\tilde{X} = U_k \Sigma_k V_k^T$ is the matrix of rank $k$ that best approximates the original matrix $X$
Outline

1 Matrices
   - The Word – Context Matrix
   - The Pair – Pattern Matrix

2 Weighting the Elements
   - Positive PMI

3 Smoothing the Matrices
   - Truncated SVD

4 Efficient Comparisons
   - LSH

5 Applications
   - The Word – Context Matrix
   - The Pair – Pattern Matrix
One general approach to LSH [9] is to “hash” items several times, in such a way that similar items are more likely to be hashed to the same bucket than dissimilar items are.

- Denitions of LSH functions include the Min-wise independent function, such as Min-Hashing, that map vectors into short signatures or fingerprints.

- After LSH, remained candidate pairs those pairs of signatures that we need to test for similarity.
One general approach to LSH [9] is to “hash” items several times, in such a way that similar items are more likely to be hashed to the same bucket than dissimilar items are.

Definitions of LSH functions include the Min-wise independent function, such as Min-Hashing, that map vectors into short signatures or fingerprints.

After LSH, remained candidate pairs those pairs of signatures that we need to test for similarity.
One general approach to LSH [9] is to “hash” items several times, in such a way that similar items are more likely to be hashed to the same bucket than dissimilar items are.

Definitions of LSH functions include the Min-wise independent function, such as Min-Hashing, that map vectors into short signatures or fingerprints.

After LSH, remained candidate pairs those pairs of signatures that we need to test for similarity.
Outline

1. Matrices
   - The Word – Context Matrix
   - The Pair – Pattern Matrix

2. Weighting the Elements
   - Positive PMI

3. Smoothing the Matrices
   - Truncated SVD

4. Efficient Comparisons
   - LSH

5. Applications
   - The Word – Context Matrix
   - The Pair – Pattern Matrix
Open Source VSM System: Semantic Vectors [10]

Implementing the random projection approach to measuring word similarity

Word similarity

Landauer and Dumais evaluated this approach with 80 multiple-choice synonym questions from the Test of English as a Foreign Language (TOEFL), achieving human-level performance [8]

Word clustering

These algorithms are able to discover different senses of polysemous words, generating different clusters for each sense [6]
Open Source VSM System: Semantic Vectors [10]

- Implementing the random projection approach to measuring word similarity

Word similarity

- Landauer and Dumais evaluated this approach with 80 multiple-choice synonym questions from the Test of English as a Foreign Language (TOEFL), achieving human-level performance [8]

Word clustering

- These algorithms are able to discover different senses of polysemous words, generating different clusters for each sense [6]
Open Source VSM System: Semantic Vectors [10]

- Implementing the random projection approach to measuring word similarity

Word similarity

- Landauer and Dumais evaluated this approach with 80 multiple-choice synonym questions from the Test of English as a Foreign Language (TOEFL), achieving human-level performance [8]

Word clustering

- These algorithms are able to discover different senses of polysemous words, generating different clusters for each sense [6]
Automatic thesaurus generation

Creating and maintaining such lexical resources is labour intensive, so it is natural to wonder whether the process can be automated to some degree [11]

Context-sensitive spelling correction

These confusions cannot be detected by a simple dictionary-based spelling checker; they require context-sensitive spelling correction [12]

Semantic role labeling

Word-context matrices can reliably predict the semantic frame to which an unknown lexical unit refers, with good levels of accuracy [13]
Automatic thesaurus generation

Creating and maintaining such lexical resources is labour intensive, so it is natural to wonder whether the process can be automated to some degree [11]

Context-sensitive spelling correction

These confusions cannot be detected by a simple dictionary-based spelling checker; they require context-sensitive spelling correction [12]

Semantic role labeling

Word-context matrices can reliably predict the semantic frame to which an unknown lexical unit refers, with good levels of accuracy [13]
Automatic thesaurus generation

Creating and maintaining such lexical resources is labour intensive, so it is natural to wonder whether the process can be automated to some degree [11]

Context-sensitive spelling correction

These confusions cannot be detected by a simple dictionary-based spelling checker; they require context-sensitive spelling correction [12]

Semantic role labeling

Word-context matrices can reliably predict the semantic frame to which an unknown lexical unit refers, with good levels of accuracy [13]
Outline

1. Matrices
   - The Word – Context Matrix
   - The Pair – Pattern Matrix

2. Weighting the Elements
   - Positive PMI

3. Smoothing the Matrices
   - Truncated SVD

4. Efficient Comparisons
   - LSH

5. Applications
   - The Word – Context Matrix
   - The Pair – Pattern Matrix
The Pair – Pattern Matrix

- Open Source VSM System: Latent Relational Analysis in S-Space [14]
  - Pattern frequencies are counted and then smoothed using SVD

- Relational similarity
  - Turney evaluated this approach to relational similarity with 374 multiple-choice analogy questions from the SAT college entrance test, achieving human-level performance [15]

- Relational clustering
  - The representative pairs to automatically generate multiple-choice analogy questions, in the style of SAT analogy questions [16]
The Pair – Pattern Matrix

- Open Source VSM System: Latent Relational Analysis in S-Space [14]
  - Pattern frequencies are counted and then smoothed using SVD
- Relational similarity
  - Turney evaluated this approach to relational similarity with 374 multiple-choice analogy questions from the SAT college entrance test, achieving human-level performance [15]
- Relational clustering
  - The representative pairs to automatically generate multiple-choice analogy questions, in the style of SAT analogy questions [16]
Open Source VSM System: Latent Relational Analysis in S-Space [14]
- Pattern frequencies are counted and then smoothed using SVD

Relational similarity
- Turney evaluated this approach to relational similarity with 374 multiple-choice analogy questions from the SAT college entrance test, achieving human-level performance [15]

Relational clustering
- The representative pairs to automatically generate multiple-choice analogy questions, in the style of SAT analogy questions [16]
The Pair – Pattern Matrix (cont.)

- **Relational classification**
  - Taint: poison is classified as strength (poisoning is stronger than tainting) and assess: review is classified as enablement (assessing is enabled by reviewing) [17]

- **Relational search**
  - A query for a relational search engine is "list all X such that X causes cancer". In this example, the relation, cause, and one of the terms in the relation, cancer, are given by the user, and the task of the search engine is to find terms that satisfy the user’s query [18]

- **Analogical mapping**
  - With a pair-pattern matrix, we can solve proportional analogies by selecting the choice that maximizes relational similarity [5]
Relational classification

- Taint: poison is classified as strength (poisoning is stronger than tainting) and assess: review is classified as enablement (assessing is enabled by reviewing) [17]

Relational search

- A query for a relational search engine is “list all X such that X causes cancer”. In this example, the relation, cause, and one of the terms in the relation, cancer, are given by the user, and the task of the search engine is to find terms that satisfy the user’s query [18]

Analogical mapping

- With a pair-pattern matrix, we can solve proportional analogies by selecting the choice that maximizes relational similarity [5]
The Pair – Pattern Matrix (cont.)

- **Relational classification**
  - Taint: poison is classified as strength (poisoning is stronger than tainting) and assess: review is classified as enablement (assessing is enabled by reviewing) [17]

- **Relational search**
  - A query for a relational search engine is “list all X such that X causes cancer”. In this example, the relation, cause, and one of the terms in the relation, cancer, are given by the user, and the task of the search engine is to find terms that satisfy the user’s query [18]

- **Analogical mapping**
  - With a pair-pattern matrix, we can solve proportional analogies by selecting the choice that maximizes relational similarity [5]
References


