Word Similarity via Symmetric Patterns

Roy Schwartz, NLP Lab, The Hebrew University
IBM ML Seminar, September 2015
Joint work with Roi Reichart and Ari Rappoport
Apples and Oranges
fruits  
sweet  
juicy  
round  
Apples and Oranges
fruits

juicy

Apples

and

Oranges

sweet

round
Overview

• Word Similarity
  – Main Approaches
  – Limitations of existing approaches

• Symmetric Patterns
  – Automatically acquired Symmetric Patterns
  – Word Similarity via Symmetric Patterns

• First order Symmetric Patterns
  – Schwartz, Reichart and Rappoport, Coling 2014

• Second + Third order Symmetric Patterns
  – Schwartz, Reichart and Rappoport, CoNLL 2015
Word Similarity

• Whether two words are **semantically** similar
  – cats are similar to dogs
Word Similarity

• Whether two words are **semantically** similar
  – **cats** are similar to **dogs**

• Definition is not entirely clear
  – Synonyms (i.e., share the same meaning)
  – Co-hyponyms (i.e., belong to the same category)
Word Similarity

• Whether two words are *semantically* similar
  – *cats* are similar to *dogs*

• Definition is not entirely clear
  – Synonyms (i.e., share the same meaning)
  – Co-hyponyms (i.e., belong to the same category)

• Human judgment evaluation
Vector Space Models

DS Hypothesis (Harris, 1954)

... tokens to date, friend lists and recent ...
... by my dear friend and companion, Fritz von ...
... even have a friend who never fails ...
... by my worthy friend Doctor Haygarth of ...
... and as a friend pointed out to ...
... partner, in-laws, relatives or friends speak a different ...
... petition to a friend Go to the ...
... otherwise, to a friend or family member ...
... images from my friend Rory though - ...
... great, and a friend as well as a colleague, who, ...
...

Examples taken from the ukwac corpus (Baroni et al., 2009)
Vector Space Models
DS Hypothesis (Harris, 1954)

... tokens to date, friend lists and recent ...
... by my dear friend and companion, Fritz von ...
... even have a friend who never fails ...
... by my worthy friend Doctor Haygarth of ...
... and as a friend pointed out to ...
... partner, in-laws, relatives or friends speak a different ...
... petition to a friend Go to the ...
... otherwise, to a friend or family member ...
... images from my friend Rory though - ...
... great, and a friend as well as a colleague, who, ...
...
Vector Space Models

\[
\begin{bmatrix}
0 \\
0.5 \\
0.76 \\
-0.12 \\
0.76 \\
0 \\
0 \\
0 \\
0 \\
-0.51 \\
. \\
. \\
. \\
. \\
\end{bmatrix}
\]
Vector Space Models

\[ \begin{pmatrix}
0 \\
0.5 \\
0.76 \\
-0.12 \\
0.76 \\
0 \\
0 \\
0 \\
-0.51 \\
\cdot \\
\cdot \\
\cdot \\
\end{pmatrix} \]

\[ \theta \]

friend 

\[ \text{colleague} \]
Vector Space Models
Baroni et al., 2014

• Count-based approaches
  – ‘France’ = { ‘Paris’: 125, ‘Baguette’: 18, ‘François Hollande’: 99, ... }
  – Many improvements: weighting schemes (e.g., PPMI), dimensionality reduction (SVD, PCA, etc.)
Vector Space Models
Baroni et al., 2014

• Count-based approaches
  – ‘France’ = { ‘Paris’: 125, ‘Baguette’: 18, ‘françois hollande’: 99, ... }
  – Many improvements: weighting schemes (e.g., PPMI), dimensionality reduction (SVD, PCA, etc.)

• Predict-based models
  – Often referred to as “word embeddings”
  – Embeddings are learnt as a by-product of a different task (most commonly a language model)
  – word2vec skip-gram (Mikolov et al., 2013)

\[
\max \sum_{t=1}^{T} \sum_{-c \leq j \leq c, j \neq 0} \log p(w_{t+j} | w_t)
\]
Similarity or Relatedness?
Hill et al., 2014

cup	coffee
Similarity or Relatedness?
Hill et al., 2014
Similarity or Dissimilarity?

tall    short
Similarity or Dissimilarity?

cm  
feet

tall  short

John  is  length
Current Vector Space Models do not Capture (pure) Word Similarity
Symmetric Patterns Contexts
Davidov and Rappoport, 2006

from X to Y

X or Y

neither X nor Y

X and Y

X as well as Y
Symmetric Patterns Contexts
Davidov and Rappoport, 2006

bright and shiny
shiny and bright
Symmetric Patterns (SPs)

- Words that co-occur in SPs tend to be semantically similar
  - Widdows and Dorow, 2002
  - Davidov and Rappoport, 2006
  - Kozareva et al., 2008
  - Feng et al., 2013
Symmetric Patterns (SPs)

- Words that co-occur in SPs tend to be semantically similar
  - Widdows and Dorow, 2002
  - Davidov and Rappoport, 2006
  - Kozareva et al., 2008
  - Feng et al., 2013

neither here nor there

bold and beautiful

John and Mike

Paris or Rome
Symmetric Patterns (SPs)

- Words that co-occur in SPs tend to be semantically similar
  - Widdows and Dorow, 2002
  - Davidov and Rappoport, 2006
  - Kozareva et al., 2008
  - Feng et al., 2013

neither here nor there  #car or wheel  John and Mike

#neither cup nor coffee  bold and beautiful  Paris or Rome

#dog and leash
Manually Defined SPs

from X to Y

X or Y

neither X nor Y

X and Y

X as well as Y

Word Similarity via Symmetric Patterns @ Roy Schwartz
Manually Defined SPs

from X to Y

X or Y

X and Y

neither X nor Y

X as well as Y
Automatically Extracted Symmetric Patterns

The (Davidov and Rappoport, 2006) algorithm

• A graph-based algorithm
  – Input: a corpus of plain text
  – Output: a set of SPs
Automatically Extracted Symmetric Patterns
The (Davidov and Rappoport, 2006) algorithm

• A graph-based algorithm
  – Input: a corpus of plain text
  – Output: a set of SPs

• The idea: search for patterns with interchangeable word pairs
  – For each pattern candidate, compute symmetry measure (M)
  – Select the patterns with the highest M values
Automatically Extracted Symmetric Patterns

The (Davidov and Rappoport, 2006) algorithm

• A graph-based algorithm
  – Input: a corpus of plain text
  – Output: a set of SPs

• The idea: search for patterns with interchangeable word pairs
  – For each pattern candidate, compute symmetry measure ($M$)
  – Select the patterns with the highest $M$ values

• The $M$ measure computes, for each pattern $p$ (e.g., “X and Y”),
  the proportion of instances of $p$ that occur in both directions
  (“cat and dog” + “dog and cat”)
  – High $M$ value $\Rightarrow$ A symmetric pattern
Dog

Cat

House

Computer

Camera

dog and house

cat and dog

dog and cat

rat and cat

cat and rat

house and computer

computer and camera

camera and computer

Example

X and Y

Word Similarity via Symmetric Patterns @ Roy Schwartz
DR06 Example

X and Y

Word Similarity via Symmetric Patterns @ Roy Schwartz
DR06 Example

X and Y

Asymmetric edges

Symmetric edges

Dog → Cat
Dog and house
Cat and dog
Dog and cat

Cat → Rat
Rat and cat
Cat and rat

House → Computer
House and computer

Computer → Camera
Camera and computer

Word Similarity via Symmetric Patterns @ Roy Schwartz
DR06 Example

X and Y

Asymmetric edges

Symmetric edges

Dog -> Cat

M = \#symmetric\_edges

\#all\_edges

Word Similarity via Symmetric Patterns @ Roy Schwartz
So Far

- Vector space models face an inherent challenge when used to tackle the task of word *similarity*
So Far

- Vector space models face an inherent challenge when used to tackle the task of word similarity

- Symmetric Patterns (SP) are useful for representing word similarity
So Far

• Vector space models face an inherent challenge when used to tackle the task of word *similarity*

• Symmetric Patterns (SP) are useful for representing word similarity

• The set of patterns can be extracted automatically from *plain text*
First Order Symmetric Patterns

*Minimally Supervised Classification to Semantic Categories using Automatically Acquired Symmetric Patterns*

**Schwartz**, Reichart and Rappoport, Coling 2014
The Task

• Binary Classification of Nouns into Semantic Categories
  – Is “dog” an animal?
  – Is “couch” a tool?

• Use minimal supervision
The Task

Example

• Animals

Dog, Cat, House, Couch, Purse, Rat, Car, Mole, Chair, Computer, Owl, Apple, Whale
The Task

Goal

• Animals

- Dog
- Cat
- House
- Couch
- Purse
- Rat
- Car
- Mole
- Computer
- Chair
- Whale
- Owl
- Apple
- Chair

Word Similarity via Symmetric Patterns @ Roy Schwartz
Symmetric Patterns to Word Similarity

• Input: a large corpus C
Symmetric Patterns to Word Similarity

- Input: a large corpus \( C \)
- Extract a set of SPs \( P \) using the DR06 algorithm
Symmetric Patterns to Word Similarity

• Input: a large corpus C
• Extract a set of SPs $P$ using the DR06 algorithm
• Traverse C, extract all instances of all $p$ in $P$
  – cats and dogs
    • House and the rooms
  – from France to England
  – ...
First Order Symmetric Patterns

- $S_{XY} \rightarrow$ the number of times $X,Y$ appeared in the same symmetric pattern
First Order Symmetric Patterns

• $S_{XY} \rightarrow$ the number of times $X,Y$ appeared in the same symmetric pattern

• **orange $\leftrightarrow$ apple**
  1. ... *apples* and *oranges* ...
  2. ... *oranges* as well as *apples* ...
  ...
  K. ... neither *apple* nor *orange* ...

  \[ \Rightarrow \text{orange} \leftrightarrow \text{apple} = \frac{K}{Z} \]

  – Z: a normalization factor
First Order Symmetric Patterns

- $S_{XY} \rightarrow$ the number of times $X,Y$ appeared in the same symmetric pattern

- **orange $\leftrightarrow$ apple**
  1. ... apples and oranges ...
  2. ... oranges as well as apples ...
     ...
  K. ... neither apple nor orange ...

  $\Rightarrow$ orange $\leftrightarrow$ apple $= \frac{K}{Z}$

- **France $\leftrightarrow$ England**
  1. ... England or France ...
  2. ... from France to England ...
     ...
  M. ... England and France ...

  $\Rightarrow$ France $\leftrightarrow$ England $= \frac{M}{Z}$

- $Z$: a normalization factor
Word Similarity Measures

\[ S_{XY} \rightarrow \text{Similarity Between Words X and Y} \]

- **Symmetric patterns**
  - Extract a set of symmetric patterns from plain text
  - \( S_{XY} \rightarrow \) the number of times X and Y participate in the same symmetric pattern
Word Similarity Measures

$S_{XY} \rightarrow \text{Similarity Between Words X and Y}$

- **Symmetric patterns**
  - Extract a set of symmetric patterns from plain text
  - $S_{XY} \rightarrow$ the number of time X and Y participate in the same symmetric pattern

- **Baselines:**
  - **Senna** word embeddings (Collobert et al., 2011):
    - $S_{XY} \rightarrow$ cosine similarity between the word embeddings of X and Y
  - **Brown** Clusters (Brown et al., 1992):
    - $S_{XY} \rightarrow 1 - \text{tree distance between X and Y clusters}$
Label Propagation Algorithms

• Iterative variant of k-Nearest Neighbors

• Baselines
  – Normalized graph cut algorithm (Yu and Shi, 2003)
  – Modified Adsorption (MAD) algorithm (Talukdar and Crammer, 2009)
Experimental Setup

Experiments

• A subset of the CSLB property norms dataset (Devereux et al., 2013)
  – 450 concrete nouns
  – Thirty human annotators assigned each noun with semantic categories
  – *animals, tools, food, clothes*

• Symmetric pattern based scores computed using the google books n-gram corpus

• Number of labeled seed words
  – 4, 10, 20, 40
Results

Word Similarity Measures
Results
Word Similarity Measures

best *symmetric patterns* model >> any other model
12.5% accuracy, 0.13 F1 points difference
More Results

• When using as few as four labeled seed words
  – Accuracy results are 82-94%
  – F1 scores are 0.64-0.86

• Symmetric patterns are superior compared to the other word similarity measures across
  – semantic categories
  – label propagation algorithms
  – labeled seed set sizes
  – evaluation measures
Second Order Symmetric Patterns

Symmetric Pattern Based Word Embeddings for Improved Word Similarity Prediction

Schwartz, Reichart and Rappoport, CoNLL 2015
The goal:

A vector space model based on symmetric pattern contexts
Second Order Symmetric Patterns

• For each word $w$ in the lexicon, build a count vector ($V_w$) of all other words that co-occur with $w$ in SPs
Second Order Symmetric Patterns

- For each word $w$ in the lexicon, build a count vector ($V_w$) of all other words that co-occur with $w$ in SPs

- **orange**
  1. ... *apples* and *oranges* ...
  2. ... *oranges* as well as *grapes* ...
  K. ... neither *banana* nor *orange*

- **China**
  1. ... *Japan* or *China* ...
  2. ... *China* rather than *Korea* ...
  M. ... *Vietnam* and *China* ...
Second Order Symmetric Patterns (2)

• Compute the Positive Pointwise Mutual Information (PPMI) between each pair of words

\[
PMI(w_i, w_j) = \log \left( \frac{p(w_i, w_j)}{p(w_i) p(w_j)} \right)
\]

\[
PPMI(w_i, w_j) = \begin{cases} 
PMI(w_i, w_j) < 0 : 0 \\
otherwise : PMI(w_i, w_j)
\end{cases}
\]
The Result: Word Embeddings based on **Second-order** Symmetric Patterns

\[ V^{sp}_{dog} = \{ \text{PPMI}(\text{dog, house}), \text{PPMI}(\text{dog, mouse}), \text{PPMI}(\text{dog, zebra}), \text{PPMI}(\text{dog, wine}), \text{PPMI}(\text{dog, cat}), \text{PPMI}(\text{dog, dolphin}), \text{PPMI}(\text{dog, bottle}), \text{PPMI}(\text{dog, pen}) \} \]
The Result: Word Embeddings based on **Second-order** Symmetric Patterns

\[ V^{sp}_{\text{dog}} = \{ \text{PPMI(dog,house)}, \text{PPMI(dog,mouse)}, \text{PPMI(dog,zebra)}, \text{PPMI(dog,wine)}, \text{PPMI(dog,cat)}, \text{PPMI(dog,dolphin)}, \text{PPMI(dog,bottle)}, \text{PPMI(dog,pen)} \} \]

\[ |V^{SP}_w| = \sim 200K \]

\[ E_w(|\text{nonzero}(V^{SP}_w)|) = \sim 50 \]
The Result: Word Embeddings based on **Second-order** Symmetric Patterns

\[ V_{\text{dog}}^{\text{sp}} = \{ \text{PPMI(dog, house)}, \text{PPMI(dog, mouse)}, \text{PPMI(dog, zebra)}, \text{PPMI(dog, wine)}, \text{PPMI(dog, cat)}, \text{PPMI(dog, dolphin)}, \text{PPMI(dog, bottle)}, \text{PPMI(dog, pen)} \} \]

\[ |V_w^{SP}| = \sim 200K \]

\[ E_w(|\text{nonzero}(V_w^{SP})|) = \sim 50 \]

- similarity rather than relatedness
Antonyms

big / small
Antonyms

big / small

• Antonyms occur in similar contexts
  • Here is a X car
  • I live in a X house
Antonyms

big / small

• Antonyms occur in similar contexts
  • Here is a X car
  • I live in a X house

⇒ In typical word embeddings, $\cos(V_{\text{big}}, V_{\text{small}})$ is high
Antonyms

big / small

Some symmetric patterns are indicative of antonymy*

“either $X$ or $Y$” (either big or small), “from $X$ to $Y$” (from poverty to richness)

* Lin et al. (2003)
Antonyms

• A variant of our model that assigns dissimilar vectors to antonym pairs
Antonyms

- A variant of our model that assigns dissimilar vectors to antonym pairs

For each word \( w \), compute \( V_w^{AP} \) similarly to \( V_w^{SP} \), but using the set of antonym patterns

\[
V_w^{AP'} = V_w^{SP} - \beta \cdot V_w^{AP}
\]

- \( \beta \) is tuned using a development set
Experiments

• Word similarity task
  – Experiments with the SimLex999 dataset (Hill et al., 2014)
  – 999 word pairs, each assigned a similarity score by human annotators
  – \( f_{<\text{model}>}(w_i, w_j) = \cos(V_{<\text{model}>}w_i, V_{<\text{model}>}w_j) \)
  – Evaluation results is the Spearman’s \( \rho \) score between model and human judgments
  – Numbers are average scores of 10 folds of 25% (dev) / 75% (test) partitions
  – Baselines: 6 state-of-the-art models
Experiments

- Word similarity task
  - Experiments with the SimLex999 dataset (Hill et al., 2014)
  - 999 word pairs, each assigned a similarity score by human annotators
  - $f_{\text{model}}(w_i, w_j) = \cos(V_{\text{model}}^{w_i}, V_{\text{model}}^{w_j})$
  - Evaluation results is the Spearman’s $\rho$ score between model and human judgments
  - Numbers are average scores of 10 folds of 25% (dev) / 75% (test) partitions
  - Baselines: 6 state-of-the-art models

<table>
<thead>
<tr>
<th>Model</th>
<th>Spearman’s $\rho$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glove (Pennington et al., 2014)</td>
<td>0.35</td>
</tr>
<tr>
<td>PPMI-Bag-of-words</td>
<td>0.423</td>
</tr>
<tr>
<td>word2vec CBOV (Mikolov et al., 2013)</td>
<td>0.43</td>
</tr>
<tr>
<td>Dep (Levy and Goldberg, 2014)</td>
<td>0.436</td>
</tr>
<tr>
<td>NNSE (Murphy et al., 2012)</td>
<td>0.455</td>
</tr>
<tr>
<td>word2vec skip-gram (Mikolov et al., 2013)</td>
<td>0.462</td>
</tr>
<tr>
<td>2nd-order SP(+)</td>
<td>0.449</td>
</tr>
</tbody>
</table>
Third Order Symmetric Patterns

- For each word \(w\), \(V^N_w\) denotes the vectors for the top \(N\) first-order SP neighboring words with \(w\)

\[
V_{SP}'_w = V_{SP}^w + \alpha \sum_{v \in V^N_w} v
\]

- Using \(N=50\): \(E_w(|\text{nonzero}(V_{SP}'_w)|) \approx 8K\)

- \(\alpha\) and \(N\) are tuned using a development set
# Third Order SP results

<table>
<thead>
<tr>
<th>Model</th>
<th>Spearman’s ρ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glove (Pennington et al., 2014)</td>
<td>0.35</td>
</tr>
<tr>
<td>PPMI-Bag-of-words</td>
<td>0.423</td>
</tr>
<tr>
<td>word2vec CBOV (Mikolov et al., 2013)</td>
<td>0.43</td>
</tr>
<tr>
<td>Dep (Levy and Goldberg, 2014)</td>
<td>0.436</td>
</tr>
<tr>
<td>NNSE (Murphy et al., 2012)</td>
<td>0.455</td>
</tr>
<tr>
<td>word2vec skip-gram (Mikolov et al., 2013)</td>
<td>0.462</td>
</tr>
<tr>
<td>3rd-order SP(−)</td>
<td>0.434</td>
</tr>
<tr>
<td>2nd-order SP(+)</td>
<td>0.449</td>
</tr>
<tr>
<td>3rd-order SP(+)</td>
<td><strong>0.517</strong></td>
</tr>
</tbody>
</table>
Joint Model

\[ f_{\text{joint}}(w_i, w_j) = \gamma \cdot f_{\text{SP}}(w_i, w_j) + (1 - \gamma) \cdot f_{\text{skip-gram}}(w_i, w_j) \]

<table>
<thead>
<tr>
<th>Model</th>
<th>Spearman’s ρ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glove (Pennington et al., 2014)</td>
<td>0.35</td>
</tr>
<tr>
<td>PPMI-Bag-of-words</td>
<td>0.423</td>
</tr>
<tr>
<td>word2vec CBO (Mikolov et al., 2013)</td>
<td>0.43</td>
</tr>
<tr>
<td>Dep (Levy and Goldberg, 2014)</td>
<td>0.436</td>
</tr>
<tr>
<td>NNSE (Murphy et al., 2012)</td>
<td>0.455</td>
</tr>
<tr>
<td>word2vec skip-gram (Mikolov et al., 2013)</td>
<td>0.462</td>
</tr>
<tr>
<td>3rd-order SP(-)</td>
<td>0.434</td>
</tr>
<tr>
<td>2nd-order SP(+)</td>
<td>0.449</td>
</tr>
<tr>
<td>3rd-order SP(+)</td>
<td>0.517</td>
</tr>
<tr>
<td>Joint (3rd-order SP(+), skip-gram)</td>
<td>0.563</td>
</tr>
<tr>
<td>Average Human Score</td>
<td>0.651</td>
</tr>
</tbody>
</table>

\(\gamma\) determined using a development set
## POS Analysis

<table>
<thead>
<tr>
<th>Model</th>
<th>Adj.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glove (Pennington et al., 2014)</td>
<td>0.571</td>
</tr>
<tr>
<td>PPMI-Bag-of-words</td>
<td>0.548</td>
</tr>
<tr>
<td>word2vec CBOB (Mikolov et al., 2013)</td>
<td>0.579</td>
</tr>
<tr>
<td>Dep (Levy and Goldberg, 2014)</td>
<td>0.54</td>
</tr>
<tr>
<td>NNSE (Murphy et al., 2012)</td>
<td>0.594</td>
</tr>
<tr>
<td>word2vec skip-gram (Mikolov et al., 2013)</td>
<td>0.604</td>
</tr>
<tr>
<td>3rd-order SP(+)</td>
<td>0.663</td>
</tr>
</tbody>
</table>
### POS Analysis

<table>
<thead>
<tr>
<th>Model</th>
<th>Adj.</th>
<th>Nouns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glove (Pennington et al., 2014)</td>
<td>0.571</td>
<td>0.377</td>
</tr>
<tr>
<td>PPMI-Bag-of-words</td>
<td>0.548</td>
<td>0.451</td>
</tr>
<tr>
<td>word2vec CBOB (Mikolov et al., 2013)</td>
<td>0.579</td>
<td>0.48</td>
</tr>
<tr>
<td>Dep (Levy and Goldberg, 2014)</td>
<td>0.54</td>
<td>0.449</td>
</tr>
<tr>
<td>NNSE (Murphy et al., 2012)</td>
<td>0.594</td>
<td>0.487</td>
</tr>
<tr>
<td>word2vec skip-gram (Mikolov et al., 2013)</td>
<td>0.604</td>
<td>0.501</td>
</tr>
<tr>
<td><strong>3rd-order SP</strong></td>
<td><strong>0.663</strong></td>
<td>0.497</td>
</tr>
</tbody>
</table>
## POS Analysis

<table>
<thead>
<tr>
<th>Model</th>
<th>Adj.</th>
<th>Nouns</th>
<th>Verbs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glove (Pennington et al., 2014)</td>
<td>0.571</td>
<td>0.377</td>
<td>0.163</td>
</tr>
<tr>
<td>PPMI-Bag-of-words</td>
<td>0.548</td>
<td>0.451</td>
<td>0.276</td>
</tr>
<tr>
<td>word2vec CBO (Mikolov et al., 2013)</td>
<td>0.579</td>
<td>0.48</td>
<td>0.252</td>
</tr>
<tr>
<td>Dep (Levy and Goldberg, 2014)</td>
<td>0.54</td>
<td>0.449</td>
<td>0.376</td>
</tr>
<tr>
<td>NNSE (Murphy et al., 2012)</td>
<td>0.594</td>
<td>0.487</td>
<td>0.318</td>
</tr>
<tr>
<td>word2vec skip-gram (Mikolov et al., 2013)</td>
<td>0.604</td>
<td>0.501</td>
<td>0.307</td>
</tr>
<tr>
<td>3rd-order SP (+)</td>
<td>0.663</td>
<td>0.497</td>
<td>0.578</td>
</tr>
</tbody>
</table>
## POS Analysis

<table>
<thead>
<tr>
<th>Model</th>
<th>Adj.</th>
<th>Nouns</th>
<th>Verbs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glove (Pennington et al., 2014)</td>
<td>0.571</td>
<td>0.377</td>
<td>0.163</td>
</tr>
<tr>
<td>PPMI-Bag-of-words</td>
<td>0.548</td>
<td>0.451</td>
<td>0.276</td>
</tr>
<tr>
<td>word2vec CBOV (Mikolov et al., 2013)</td>
<td>0.579</td>
<td>0.48</td>
<td>0.252</td>
</tr>
<tr>
<td>Dep (Levy and Goldberg, 2014)</td>
<td>0.54</td>
<td>0.449</td>
<td><strong>0.376</strong></td>
</tr>
<tr>
<td>NNSE (Murphy et al., 2012)</td>
<td>0.594</td>
<td>0.487</td>
<td>0.318</td>
</tr>
<tr>
<td>word2vec skip-gram (Mikolov et al., 2013)</td>
<td>0.604</td>
<td><strong>0.501</strong></td>
<td>0.307</td>
</tr>
<tr>
<td>3rd-order SP(^{(+)})</td>
<td><strong>0.663</strong></td>
<td>0.497</td>
<td><strong>0.578</strong></td>
</tr>
</tbody>
</table>
**POS Analysis**

<table>
<thead>
<tr>
<th>Model</th>
<th>Adj.</th>
<th>Nouns</th>
<th>Verbs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glove (Pennington et al., 2014)</td>
<td>0.571</td>
<td>0.377</td>
<td>0.163</td>
</tr>
<tr>
<td>PPMI-Bag-of-words</td>
<td>0.548</td>
<td>0.451</td>
<td>0.276</td>
</tr>
<tr>
<td>word2vec CBOV (Mikolov et al., 2013)</td>
<td>0.579</td>
<td>0.48</td>
<td>0.252</td>
</tr>
<tr>
<td>Dep (Levy and Goldberg, 2014)</td>
<td>0.54</td>
<td>0.449</td>
<td>0.376</td>
</tr>
<tr>
<td>NNSE (Murphy et al., 2012)</td>
<td>0.594</td>
<td>0.487</td>
<td>0.318</td>
</tr>
<tr>
<td>word2vec skip-gram (Mikolov et al., 2013)</td>
<td>0.604</td>
<td><strong>0.501</strong></td>
<td>0.307</td>
</tr>
<tr>
<td>3&lt;sup&gt;rd&lt;/sup&gt;-order SP&lt;sup&gt;(+)&lt;/sup&gt;</td>
<td><strong>0.663</strong></td>
<td>0.497</td>
<td><strong>0.578</strong></td>
</tr>
</tbody>
</table>
## POS Analysis

<table>
<thead>
<tr>
<th>Model</th>
<th>Adj.</th>
<th>Nouns</th>
<th>Verbs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glove (Pennington et al., 2014)</td>
<td>0.571</td>
<td>0.377</td>
<td>0.163</td>
</tr>
<tr>
<td>PPMI-Bag-of-words</td>
<td>0.548</td>
<td>0.451</td>
<td>0.276</td>
</tr>
<tr>
<td>word2vec CBOB (Mikolov et al., 2013)</td>
<td>0.579</td>
<td>0.48</td>
<td>0.252</td>
</tr>
<tr>
<td>Dep (Levy and Goldberg, 2014)</td>
<td>0.54</td>
<td>0.449</td>
<td>0.376</td>
</tr>
<tr>
<td>NNSE (Murphy et al., 2012)</td>
<td>0.594</td>
<td>0.487</td>
<td>0.318</td>
</tr>
<tr>
<td>word2vec skip-gram (Mikolov et al., 2013)</td>
<td>0.604</td>
<td>0.501</td>
<td>0.307</td>
</tr>
<tr>
<td>3rd-order SP(+)</td>
<td>0.663</td>
<td>0.497</td>
<td>0.578</td>
</tr>
</tbody>
</table>
# Antonyms

<table>
<thead>
<tr>
<th>Word Pair</th>
<th>$SP$</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>+AN</td>
<td>-AN</td>
<td>SG</td>
<td></td>
</tr>
<tr>
<td>new - old</td>
<td>1</td>
<td>6</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>narrow - wide</td>
<td>1</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>necessary - unnecessary</td>
<td>2</td>
<td>2</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>bottom - top</td>
<td>3</td>
<td>8</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>absence - presence</td>
<td>4</td>
<td>7</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>receive - send</td>
<td>1</td>
<td>9</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>fail - succeed</td>
<td>1</td>
<td>8</td>
<td>6</td>
<td></td>
</tr>
</tbody>
</table>
Summary

• Symmetric patterns are useful for representing word similarity
  – They capture **similarity** and not **relatedness**
  – They are able to mark antonym pairs as dissimilar

• First-, second- and third-order SPs are useful
  – 5.5% improvement over six state-of-the-art models
  – 10% improvement with a **joint** model
  – 20% improvement on **verbs**
Future Work

• Enhancing bag-of-words models with symmetric patterns information

• Does order count? asymmetric symmetric patterns