Word & Sense Embedding and their Application to Word Sense Induction

Linfeng Song
Outline

- Word Embedding
- Sense Embedding
- Sense Embedding for Word Sense Induction
- Conclusion
Word Embedding

Word Embedding is a set of language techniques in which words from the vocabulary are mapped to vectors of real numbers in a low-dimensional space.
Word Embedding is a set of language techniques in which words from the vocabulary are mapped to vectors of real numbers in a low-dimensional space.

Previous Methods
- Build co-occurrence matrix from a corpus
- Perform dimension reduction with PCA
- Learn by counting
Word Embedding is a set of language techniques in which words from the vocabulary are mapped to vectors of real numbers in a low-dimensional space.

Current methods
- based on a neural network architecture
- Learn by predicting
Skip-Gram Model
\[ P(D = 1|w_i, w_j) = \frac{1}{1 + e^{v_i^T v_j}} \]
\[ P(D = 0|w_i, w_j) = 1 - P(D = 1|w_i, w_j) \]
Skip-Gram Model

Given a document formalized as a list of \((w_i, C_i)\)

\[
J = \sum_{i=1}^{T} \left[ \sum_{c \in C_i} P(D = 1|v_i, v_c) + \sum_{c' \in V - C_i} P(D = 0|v_i, v_{c'}) \right]
\]
Sense Embedding

- Ubiquitous polysemous words harm the performance for most NLP systems
- Solution: learn a embedding for each sense instead
Sense Embedding

- Clustering-based
- Nonparametric
- Ontology-based
Sense Embedding

- Clustering-based
- Nonparametric
- Ontology-based

How sense is defined?
Sense Embedding

- **Clustering-based** based on the distributional hypothesis of Harris, (1954):
  a word sense is reflected by a set of contexts where it appears

- **Nonparametric**

- **Ontology-based** based on the sense definition of a sense inventory
Reisinger and Mooney (2010)

- learn co-occurrence vector for each $w_i$
Reisinger and Mooney (2010)

- learn co-occurrence vector for each $w_i$
- cluster all tokens of $w_i$ into $K$ clusters
  - each token is represented by the context vector which is the average of word vectors in the context
Reisinger and Mooney (2010)

- learn co-occurrence vector for each $w_i$
- cluster all tokens of $w_i$ into K clusters
  - each token is represented by the context vector which is the average of word vectors in the context
- learn one vector for each centroid of $w_i$
  - averaging all belonging context vectors
The similarity functions

\[ \text{MaxSim}(u, v) = \max_{1 \leq i \leq K, 1 \leq j \leq K} d(\pi_i(u), \pi_j(v)) \]

\[ \text{AvgSim}(u, v) = \frac{1}{K^2} \sum_{i=1}^{K} \sum_{j=1}^{K} d(\pi_i(u), \pi_j(v)) \]

\[ \text{AvgSimC}(u, v) = \frac{1}{K^2} \sum_{i=1}^{K} \sum_{j=1}^{K} d(\text{vec}(c), \pi_i(u)) \times d(\text{vec}(c'), \pi_j(v)) \times d(\pi_i(u), \pi_j(v)) \]
Huang et al. (2012b)

clustering-based
non-parametric
ontology-based
Huang et al. (2012b)

- learn word vectors
- re-label the data by clustering
- learn sense vectors via the same neural network
Huang et al. (2012b)

- learn word vectors
- re-label the data becomes problematic!
The pipeline leads to error propagation!
- learn sense vectors via the same neural network
Neelakantan et al. (2014)

clustering-based
non-parametric
ontology-based
Use argmax to pick the cluster:

\[ s_i = \underset{k}{\text{argmax}} \cosine(\mu(w_i, k), \nu_{\text{context}}(c_i)) \quad k \in [1, 2, \ldots, K] \]
Probability for (not) observing words:

\[ p(D = 1|v_s(w_i, s_i), v_g(c)) = \frac{1}{1 + e^{-v_s(w_i, s_i)^T v_g(c)}} \]

\[ p(D = 0|v_s(w_i, s_i), v_g(c')) = 1 - p(D = 1|v_s(w_i, s_i), v_g(c')) \]
Neelakantan et al. (2014)

clustering-based
non-parametric
ontology-based

Update as Kmeans
Update as Skip-gram
Intuitively, different words should have different number of senses.
Li and Jurafsky (2015)

- Create a Chinese Restaurant Process for each word
  - a sense corresponds to a table
  - a data point is a customer
Li and Jurafsky (2015)

Create a Chinese Restaurant Process for each word
- a sense corresponds to a table
- a data point is a customer

Probability for choosing a sense is defined as:

\[ p(s_i = k_t) \propto \begin{cases} N_t p(k_t | c_i), & \text{if } k_t \text{ already exists} \\ \gamma, & \text{if } k_t \text{ is new} \end{cases} \]
Rothe and Schutze (2015)

clustering-based
non-parametric
ontology-based

Each dimension is an AutoEncoder

Independent between different dimensions

Best Student Paper of ACL 2015
<table>
<thead>
<tr>
<th></th>
<th>Synset₁</th>
<th>...</th>
<th>Synsetᵢ</th>
<th>...</th>
<th>Synsetₙ</th>
</tr>
</thead>
<tbody>
<tr>
<td>people</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dog</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>hound</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Rothe and Schutze (2015)**

clustering-based
non-parametric
ontology-based
Objectives

\[
\text{argmin}_{D(d), E(d)} \| D^{(d)} E^{(d)} w^{(d)} - w^{(d)} \| \quad \forall d
\]

\[
\text{argmin}_{D(d), E(d)} \| E^{(d)} \text{diag}(w^{(d)}) - D^{(d)} \text{diag}(s^{(d)}) \| \quad \forall d
\]

\[
\text{argmin}_{E(d)} \| RE^{(d)} w^{(d)} \| \quad \forall d
\]
### Evaluating on Word Similarity task

<table>
<thead>
<tr>
<th>Model</th>
<th>MaxSim</th>
<th>AvgSim</th>
<th>AvgSimC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Huang</td>
<td>26.1</td>
<td>62.8</td>
<td>65.7</td>
</tr>
<tr>
<td>MSSG</td>
<td>57.26</td>
<td>67.2</td>
<td>69.3</td>
</tr>
<tr>
<td>MSSG-NP</td>
<td>59.80</td>
<td>67.3</td>
<td>69.1</td>
</tr>
<tr>
<td>CRP</td>
<td>66.4</td>
<td>-</td>
<td>67.0</td>
</tr>
<tr>
<td>Retro</td>
<td>-</td>
<td>-</td>
<td>41.7</td>
</tr>
<tr>
<td>EM</td>
<td>-</td>
<td>-</td>
<td>61.3</td>
</tr>
<tr>
<td>Retro+EM</td>
<td>-</td>
<td>-</td>
<td>58.7</td>
</tr>
<tr>
<td>AutoExtend</td>
<td>-</td>
<td>68.9</td>
<td><strong>69.8</strong></td>
</tr>
</tbody>
</table>
## Evaluating on Word Similarity task

<table>
<thead>
<tr>
<th>Model</th>
<th>MaxSim</th>
<th>AvgSim</th>
<th>AvgSimC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Huang</td>
<td>26.1</td>
<td>62.8</td>
<td>65.7</td>
</tr>
<tr>
<td>MSSG</td>
<td>57.26</td>
<td>67.2</td>
<td>69.3</td>
</tr>
<tr>
<td>MSSG-NP</td>
<td>59.80</td>
<td>67.3</td>
<td>69.1</td>
</tr>
<tr>
<td>CRP</td>
<td>66.4</td>
<td>-</td>
<td>67.0</td>
</tr>
<tr>
<td>Retro</td>
<td>-</td>
<td>-</td>
<td>41.7</td>
</tr>
<tr>
<td>EM</td>
<td>-</td>
<td>-</td>
<td>61.3</td>
</tr>
<tr>
<td>Retro+EM</td>
<td>-</td>
<td>-</td>
<td>58.7</td>
</tr>
<tr>
<td>AutoExtend</td>
<td>-</td>
<td>68.9</td>
<td>69.8</td>
</tr>
</tbody>
</table>
Evaluating on Word Similarity task

<table>
<thead>
<tr>
<th>Model</th>
<th>MaxSim</th>
<th>AvgSim</th>
<th>AvgSimC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Huang</td>
<td>26.1</td>
<td>62.8</td>
<td>65.7</td>
</tr>
<tr>
<td>MSSG</td>
<td>57.26</td>
<td>67.2</td>
<td>69.3</td>
</tr>
<tr>
<td>MSSG-NP</td>
<td>59.80</td>
<td>67.3</td>
<td>69.1</td>
</tr>
<tr>
<td>CRP</td>
<td>66.4</td>
<td>-</td>
<td>67.0</td>
</tr>
<tr>
<td>Retro</td>
<td>-</td>
<td>-</td>
<td>41.7</td>
</tr>
<tr>
<td>EM</td>
<td>-</td>
<td>-</td>
<td>61.3</td>
</tr>
<tr>
<td>Retro+EM</td>
<td>-</td>
<td>-</td>
<td>58.7</td>
</tr>
<tr>
<td>AutoExtend</td>
<td>-</td>
<td>68.9</td>
<td>69.8</td>
</tr>
</tbody>
</table>
Word Sense Induction (WSI)
- automatically discover senses from unlabeled data without referring to any sense inventory
Sense Embedding for Word Sense Induction

- Word Sense Induction (WSI)
  - automatically discover senses from unlabeled data without referring to any sense inventory

- Previous methods on WSI
  - learn co-occurrence vectors by counting
  - learn centroids by clustering
Word Sense Induction (WSI)
- automatically discover senses from unlabeled data without referring to any sense inventory

Previous methods on WSI
- learn co-occurrence vectors by counting
- learn centroids by clustering
- problematic: have to learn a model for each word impractical for real applications
Sense Embedding for Word Sense Induction

- Compare with existing methods, Sense Embedding:
  - perform joint learning for multiple words
  - learn by predicting
    - learn by predicting >> learn by counting
Sense Embedding for Word Sense Induction

- Compare with existing methods, Sense Embedding:
  - perform joint learning for multiple words
  - learn by predicting
  - Promising for this task!
### Sense Embedding for Word Sense Induction

<table>
<thead>
<tr>
<th>System</th>
<th>SemEval-2010 WSI</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>UoY (2010)</td>
<td>62.4</td>
<td>Best result of the task</td>
</tr>
<tr>
<td>NMF&lt;sub&gt;lib&lt;/sub&gt; (2011)</td>
<td>62.6</td>
<td>By Charniak @Brown U</td>
</tr>
<tr>
<td>NB (2013)</td>
<td>65.4</td>
<td></td>
</tr>
<tr>
<td>Spectral (2014)</td>
<td>60.7</td>
<td>By CMU</td>
</tr>
<tr>
<td>SE-WSI-fix</td>
<td>66.3</td>
<td></td>
</tr>
<tr>
<td>SE-WSI-CRP</td>
<td>61.2</td>
<td></td>
</tr>
<tr>
<td>CRP-PPMI</td>
<td>59.2</td>
<td></td>
</tr>
<tr>
<td>WE-Kmeans</td>
<td>58.6</td>
<td></td>
</tr>
</tbody>
</table>
## Sense Embedding for Word Sense Induction

<table>
<thead>
<tr>
<th>System</th>
<th>SemEval-2010 WSI</th>
</tr>
</thead>
<tbody>
<tr>
<td>UoY (2010)</td>
<td>62.4</td>
</tr>
<tr>
<td>NMF_{lib} (2011)</td>
<td>62.6</td>
</tr>
<tr>
<td>NB (2013)</td>
<td>65.4</td>
</tr>
<tr>
<td>Spectral (2014)</td>
<td>60.7</td>
</tr>
<tr>
<td>SE-WSI-fix</td>
<td><strong>66.3</strong></td>
</tr>
<tr>
<td>SE-WSI-CRP</td>
<td>61.2</td>
</tr>
<tr>
<td>CRP-PPMI</td>
<td>59.2</td>
</tr>
<tr>
<td>WE-Kmeans</td>
<td>58.6</td>
</tr>
</tbody>
</table>

Joint learning is better!

Neelakantan et al. (2014)

word2vec + Kmeans
<table>
<thead>
<tr>
<th>System</th>
<th>SemEval-2010 WSI</th>
</tr>
</thead>
<tbody>
<tr>
<td>UoY (2010)</td>
<td>62.4</td>
</tr>
<tr>
<td>NMF_{lib} (2011)</td>
<td>62.6</td>
</tr>
<tr>
<td>NB (2013)</td>
<td>65.4</td>
</tr>
<tr>
<td>Spectral (2014)</td>
<td>60.7</td>
</tr>
<tr>
<td>SE-WSI-fix</td>
<td>66.3</td>
</tr>
<tr>
<td>SE-WSI-CRP</td>
<td>61.2</td>
</tr>
<tr>
<td>CRP-PPMI</td>
<td>59.2</td>
</tr>
<tr>
<td>WE-Kmeans</td>
<td>58.6</td>
</tr>
</tbody>
</table>

Learn by predicting is better!

Li and Jurafsky (2015)
Co-occur+ CRP
Conclusion

- Introduced previous and current techniques for Word Embedding
  - Skip-gram

- Describe 3 directions for Sense Embedding
  - Clustering-based
  - Nonparametric
  - Ontology-based

- Sense Embedding for Word Sense Induction
  - Best performance right now!
Recent Publications


Thank you for listening

Questions?