CS11-747 Neural Networks for NLP

Models of Words

Graham Neubig

Carnegie Mellon University
Language Technologies Institute

Site
https://phontron.com/class/nn4nlp2017/
What do we want to know about words?

• Are they the same part of speech?

• Do they have the same conjugation?

• Do these two words mean the same thing?

• Do they have some semantic relation (is-a, part-of, went-to-school-at)?
A Manual Attempt: WordNet

- WordNet is a large database of words including parts of speech, semantic relations

- Major effort to develop, projects in many languages.

- But can we do something similar, more complete, and without the effort?
An Answer (?): Word Embeddings!

- A continuous vector representation of words

- Within the word embedding, these features of syntax and semantics may be included
  - Element 1 might be more positive for nouns
  - Element 2 might be positive for animate objects
  - Element 3 might have no intuitive meaning whatsoever
Word Embeddings are Cool!
(An Obligatory Slide)

• e.g. king-man+woman = queen (Mikolov et al. 2013)

• “What is the female equivalent of king?” is not easily accessible in many traditional resources
How to Train Word Embeddings?

• **Initialize randomly**, train jointly with the task

• Pre-train on a **supervised** task (e.g. POS tagging) and test on another, (e.g. parsing)

• Pre-train on an **unsupervised** task (e.g. word2vec)
Unsupervised Pre-training of Word Embeddings
(Summary of Goldberg 10.4)
Distributional vs. Distributed Representations

- **Distributional representations**
  - Words are similar if they appear in similar contexts (Harris 1954); distribution of words indicative of usage
  - In contrast: *non-distributional* representations created from lexical resources such as WordNet, etc.

- **Distributed representations**
  - Basically, something is represented by a vector of values, each representing activations
  - In contrast: *local* representations, where represented by a discrete symbol (one-hot vector)
Distributional Representations
(see Goldberg 10.4.1)

- Words appear in a context

(try it yourself w/ kwic.py)
Count-based Methods

• Create a word-context count matrix

  • **Count** the number of co-occurrences of word/context, with rows as word, columns as contexts

  • Maybe **weight** with pointwise mutual information

  • Maybe **reduce dimensions** using SVD

• **Measure their closeness** using cosine similarity (or generalized Jaccard similarity, others)
Prediction-basd Methods
(See Goldberg 10.4.2)

• Instead, try to **predict** the words within a neural network

• Word embeddings are the byproduct
Word Embeddings from Language Models

giving

lookup lookup

tanh(W₁ h + b₁)

W + bias scores = softmax probs
Context Window Methods

• If we don’t need to calculate the probability of the sentence, other methods possible!

• These can move closer to the contexts used in count-based methods

• These drive word2vec, etc.
CBOW
(Mikolov et al. 2013)

- Predict word based on sum of surrounding embeddings

\[
\text{giving} \quad a \quad *** \quad \text{at} \quad \text{the} \\
\text{lookup} \quad \text{lookup} \quad \text{lookup} \quad \text{lookup} \\
+ \quad + \quad + \quad + \\
= \text{W} \\
\text{scores} \xrightarrow{\text{softmax}} \text{probs} \\
\text{loss}
\]
Let’s Try it Out!

wordemb-cbow.py
Skip-gram
(Mikolov et al. 2013)

• Predict each word in the context given the word
Let’s Try it Out!

wordemb-skipgram.py
Other Notes

• Strong connection between count-based methods and prediction-based methods (Levy and Goldberg 2014)

• Skip-gram objective is equivalent to matrix factorization with PMI and discount for number of samples $k$ (sampling covered next time)

\[ M_{w,c} = \text{PMI}(w, c) - \log(k) \]

• Other estimation methods: GloVe (Pennington et al. 2014), etc.
What Contexts?

• Context has a large effect!

• **Small context window:** more syntax-based embeddings

• **Large context window:** more semantics-based, topical embeddings

• **Context based on syntax:** more functional, w/ words with same inflection grouped
Evaluating Embeddings
Types of Evaluation

• Intrinsic vs. Extrinsic
  - **Intrinsic**: How good is it based on its features?
  - **Extrinsic**: How useful is it downstream?

• Qualitative vs. Quantitative
  - **Qualitative**: Examine the characteristics of examples.
  - **Quantitative**: Calculate statistics
Visualization of Embeddings

• Reduce high-dimensional embeddings into 2/3D for visualization (e.g. Mikolov et al. 2013)
Non-linear Projection

- Non-linear projections group things that are close in high-dimensional space

- e.g. SNE/t-SNE (van der Maaten and Hinton 2008) group things that give each other a high probability according to a Gaussian

(Image credit: Derksen 2016)
Let’s Try it Out!

wordemb-vis-tsne.py
t-SNE Visualization can be Misleading! (Wattenberg et al. 2016)

- Settings matter

- Linear correlations cannot be interpreted
Intrinsic Evaluation of Embeddings
(categorization from Schnabel et al 2015)

• **Relatedness:** The correlation btw. embedding cosine similarity and human eval of similarity?

• **Analogy:** Find x for “a is to b, as x is to y”.

• **Categorization:** Create clusters based on the embeddings, and measure purity of clusters.

• **Selectional Preference:** Determine whether a noun is a typical argument of a verb.
Extrinsic Evaluation: Using Word Embeddings in Systems

- **Initialize** w/ the embeddings
- **Concatenate** pre-trained embeddings with learned embeddings
- Latter has the potential to provide better generalization, but
How Do I Choose Embeddings?

- No one-size-fits-all embedding (Schnabel et al. 2015)

<table>
<thead>
<tr>
<th>relatedness</th>
<th>categorization</th>
<th>sel. prefs</th>
<th>analogy</th>
</tr>
</thead>
<tbody>
<tr>
<td>relatedness</td>
<td>relatedness</td>
<td>relatedness</td>
<td>relatedness</td>
</tr>
<tr>
<td>CBOB</td>
<td>74.0</td>
<td>64.0</td>
<td>71.5</td>
</tr>
<tr>
<td>GloVe</td>
<td>63.7</td>
<td>54.8</td>
<td>65.8</td>
</tr>
<tr>
<td>TSCCA</td>
<td>57.8</td>
<td>54.4</td>
<td>64.7</td>
</tr>
<tr>
<td>C&amp;W</td>
<td>48.1</td>
<td>49.8</td>
<td>60.7</td>
</tr>
<tr>
<td>H-PCA</td>
<td>19.8</td>
<td>32.9</td>
<td>43.6</td>
</tr>
<tr>
<td>Rand. Proj.</td>
<td>17.1</td>
<td>19.5</td>
<td>24.9</td>
</tr>
</tbody>
</table>

Table 1: Results on absolute intrinsic evaluation. The best result for each

The second row contains the names of the corresponding datasets.

- Be aware, and use the best one for the task
When are Pre-trained Embeddings Useful?

- Basically, when training data is insufficient
- **Very useful**: tagging, parsing, text classification
- **Less useful**: machine translation
- **Basically not useful**: language modeling
Improving Embeddings
Limitations of Embeddings

- Sensitive to **superficial differences** (dog/dogs)
- Insensitive to **context** (financial bank, bank of a river)
- Not necessarily **coordinated** with knowledge or across languages
- **Not interpretable**
- Can **encode bias** (encode stereotypical gender roles, racial biases)
Sub-word Embeddings (1)

- Can capture sub-word regularities
  - Morpheme-based (Luong et al. 2013)
  - Character-based (Ling et al. 2015)
Sub-word Embeddings (2)

- **Bag of character n-grams** used to represent word (Bojanowski et al. 2017)
  
  where

  $<$wh, whe, her, ere, re$>$

- Use n-grams from 3-6 plus word itself

- Used in the “fasttext” toolkit
Multi-prototype Embeddings

- Simple idea, words with multiple meanings should have different embeddings (Reisinger and Mooney 2010)

- Non-parametric estimation (Neelakantan et al. 2014) also possible
Multilingual Coordination of Embeddings (Faruqui et al. 2014)

• We have word embeddings in two languages, and want them to match
Retrofitting of Embeddings to Existing Lexicons

- We have an existing lexicon like WordNet, and would like our vectors to match (Faruqui et al. 2015)
Sparse Embeddings

- Each dimension of a word embedding is not interpretable

- Solution: add a sparsity constraint to increase the information content of non-zero dimensions for each word (e.g. Murphy et al. 2012)

<table>
<thead>
<tr>
<th>Model</th>
<th>Top 5 Words (per dimension)</th>
</tr>
</thead>
</table>
| SVD$_{300}$ | well, long, if, year, watch  
                 plan, engine, e, rock, very  
                 get, no, features, music, via  
                 features, by, links, free, down  
                 works, sound, video, building, section |
| NNSE$_{1000}$ | inhibitor, inhibitors, antagonists, receptors, inhibition  
                   bristol, thames, southampton, brighton, poole  
                   delhi, india, bombay, chennai, madras  
                   pundits, forecasters, proponents, commentators, observers  
                   nosy, averse, leery, unsympathetic, snotty |
De-biasing Word Embeddings (Bolukbasi et al. 2016)

- Word embeddings reflect bias in statistics

<table>
<thead>
<tr>
<th>Extreme she</th>
<th>Extreme he</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. homemaker</td>
<td>1. maestro</td>
</tr>
<tr>
<td>2. nurse</td>
<td>2. skipper</td>
</tr>
<tr>
<td>3. receptionist</td>
<td>3. protege</td>
</tr>
<tr>
<td>4. librarian</td>
<td>4. philosopher</td>
</tr>
<tr>
<td>5. socialite</td>
<td>5. captain</td>
</tr>
<tr>
<td>6. hairdresser</td>
<td>6. architect</td>
</tr>
<tr>
<td>7. nanny</td>
<td>7. financier</td>
</tr>
<tr>
<td>8. bookkeeper</td>
<td>8. warrior</td>
</tr>
<tr>
<td>9. stylist</td>
<td>9. broadcaster</td>
</tr>
<tr>
<td>10. housekeeper</td>
<td>10. magician</td>
</tr>
</tbody>
</table>

- Identify pairs to “neutralize”, find the direction of the trait to neutralize, and ensure that they are neutral in that direction

<table>
<thead>
<tr>
<th>Gender stereotype she-he analogies</th>
</tr>
</thead>
<tbody>
<tr>
<td>sewing-carpentry</td>
</tr>
<tr>
<td>nurse-surgeon</td>
</tr>
<tr>
<td>blond-burly</td>
</tr>
<tr>
<td>giggle-chuckle</td>
</tr>
<tr>
<td>sassy-snappy</td>
</tr>
<tr>
<td>volleyball-football</td>
</tr>
<tr>
<td>registered nurse-physician</td>
</tr>
<tr>
<td>interior designer-architect</td>
</tr>
<tr>
<td>feminism-conservatism</td>
</tr>
<tr>
<td>vocalist-guitarist</td>
</tr>
<tr>
<td>diva-superstar</td>
</tr>
<tr>
<td>charcoal</td>
</tr>
<tr>
<td>charm</td>
</tr>
<tr>
<td>lovely-brilliant</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Gender appropriate she-he analogies</th>
</tr>
</thead>
<tbody>
<tr>
<td>queen-king</td>
</tr>
<tr>
<td>waitess-waiter</td>
</tr>
<tr>
<td>sister-brother</td>
</tr>
<tr>
<td>mother-father</td>
</tr>
<tr>
<td>ovarian cancer-prostate cancer</td>
</tr>
<tr>
<td>convent-monastery</td>
</tr>
</tbody>
</table>
Questions?