Exploring Word Embeddings

Davis Busteed
LING 581

Problem

“[Vector Semantics is] the standard way to represent meaning in NLP”

- Class slides from 9/20

- Used in
  - Machine learning (neural networks, etc)
  - Sentiment analysis
  - Question answering
  - Conversational agents
  - Lots more

But the process of creating word embeddings is quite abstract, making one of the most important topics in NLP out-of-reach for many people

Goal

Create a tool that allows users to observe how word embeddings are formed

How?

- Create multiple embedding models from a single source
- Compare the changes across the different models

Tools Used

- PyQt
  - Didn't know what I was doing
- NLTK
- Gensim's Word2Vec
- Scikit-learn
- Matplotlib

```
try:
    something_that_might_break()
except:
    pass
```

NLP Steps

1. Add text (manual input, corpus, etc)
2. Clean text w/ NLTK
3. Create model snapshots `[a,b,c,d] => [(a), (a,b), (a,b,c), (a,b,c,d)]`
4. Reduce embedding features with PCA
   a. Allows points to be plotted in 2D space

```
sentences = [i for x in [x.split('\n') for x in sent_tokenize(file_text)] for i in x]
sentences = [
    [x for x in wordpunct_tokenize(' '.join([(contractions[s.lower()]
    if s.lower() in contractions else s.lower()) for s in sent.split()])) if str.isalnum(x)]
    for sent in sentences]
```

Demo

Evaluation

- Inspected vocab output to ensure that data cleaning was correct
- Found some similarities between my tool and WordVis (wordvis.com)
- Some user-testing showed that the tool can be useful in exposing the concepts behind vector semantics

Questions?