Journal Report 2

9/9/19 – 9/15/19

Valerie Nayak
Computer Systems Research Lab
Period 2, White

Daily Log

Monday September 9

I am working through the class assignment from Week 1, "exploring word vectors". I am using the Reuters corpus as an example and creating the co-occurrence matrix of words. To create the co-occurrence matrix, I created a window and count how many times certain words appear together.

Tuesday September 10

Today, I am continuing class assignment 1. I have been able to read in the Reuters corpus and create the co-occurrence matrix of words. I wrote a method to create a word embedding, which produces a new matrix. The new matrix was fairly easy to create using sklearn, but since I've never done linear algebra the concept of an SVD (singular value decomposition) was pretty new to me and I had to do some research on it to get a brief overview of what it is.

I am also able to plot the word vectors in a 2D space using matplotlib by creating a 2D word embedding.

I'm implementing a method to use word2vec vector embeddings so I can compare them to the outputs I got from my co-occurrence matrices. Word2vec is allows us to use preceding words to use back propagation on our matrices.

Using cosine similarity, my code is able to solve analogies using vector arithmatic. For example, man : king :: woman : ? returns queen. To do this, you add the vectors for woman and king and subtract man. The resulting vector is closest to the vector for queen. Please see the Reflection for more examples I tried.

Thursday September 12

Today was spent watching my next lecture video. This lecture continues discussion of word vectors and word embeddings, then it gets into the GloVe model of word vectors.

The lecture discusses stochastic gradient descent, which samples just a portion of the training data before making updates instead of looping through the entire training set. Stochastic gradient descent speeds up the learning process.

The end of the lecture discussed words that have multiple meanings and introduced the idea of distinguishing these words.

Timeline

Date	Goal	Met
Sunday Sep 8	Finalize new project proposal and find a database and online introductory tutorials to follow.	Yes. Finalized new project and found a fully recorded online class.
Sunday Sep 15	Be able to create vector representations of words using Word2Vec and finish lectures/exercises for week 1 of Stanford course.	Yes. Completed the first two lecture video and finished the first week's assignment.
Sunday Sep 22	Implement methods with GloVe (global vectors for word representation). Lectures/exercises for week 2 of online course.	
Sunday Sep 29	Week 3 of online course. Be able to perform basic sentence classification using neural networks and word windows.	

Reflection

Here is an analogy I tried on my own (not from a class example) using word2vec. If you take the vector for "car" and add "fly" and subtract the vector for "drive", here are the first four results:

```
    [('airplane', 0.524011492729187),
     ('plane', 0.5212047100067139),
     ('aircraft', 0.5073351860046387),
     ('planes', 0.48218047618865967)]
```

Another example I tried was "sun : summer :: snow : ?", and using word2vec I got the expected result. The word with the highest probability prediction was "winter".

One of the exercises I did for my class was finding examples of inherent biases in word vectors. I was tasked with finding an analogy that would showcase some kind of bias (gender, race, sexual orientation, etc.) in the word embeddings. In the results below, I compare the results of two analogies that differ based on gender.

```
man : intelligent :: women : ?
    [('smart', 0.5175650715827942),
     ('wheelchair_TAO', 0.48157620429992676),
     ('Telkonet_SmartEnergy_TSE', 0.4630429446697235),
     ('creationism_repackaged', 0.4610934555530548),
     ('intuitive', 0.45407718420028687),
     ('perceptive', 0.44969040155410767),
     ('Rufus_Johnstone', 0.4471580982208252),
     ('thoughtful', 0.44639939069747925),
     ('articulate', 0.43655407428741455),
     ('endpoint_devices_SIED', 0.4330403804779053)]
```

```
woman : intelligent :: man : ?
    [('smart', 0.5542781352996826),
     ('wheelchair_TAO', 0.5381926894187927),
     ('fraudulant_stuff', 0.4719175696372986),
     ('clever', 0.469494104385376),
     ('astute', 0.4674438238143921),
     ('skillfull', 0.46629250049591064),
     ('erudite', 0.46462637186050415),
     ('thoughtful', 0.4552404284477234),
     ('charismatic_manipulator', 0.45140716433525085),
     ('technically_adept', 0.448112428188324)]
```

Clearly, not all of the above results make sense – the functions being used don't always perfectly solve an analogy. However, it does illustrate some inherent bias in the top 10 probability outputs. In the first analogy, women are associated with being "intuitive, perceptive, thoughtful, and articulate". Using the same analogy but switching the gender produces some of the same results, but some are different, too. According to the function, men are "clever, astute, skillful, and technically adept". None of those words appeared in the female analogy.

The vectors and the functions only reflect the biases from the text on which they were based. Any bias in the input data will result in the same biases being ampliftied in the outputs. This exercise was interesting in understanding how this might play out and gave me something to think about when continuing my NLP studies and explorations.

This example reminded me of Amazon's experiment which tried to train a natural language model to identify good job applicant candidates based on their resumes. The model ended up learning to penalize women and Amazon abandoned the project. After just a week of exploring NLP and word vectors, I'm able to show biases in our use of language. As Amazon and I discovered, plenty of these hidden biases exist and they can be quite difficult to remove from our algorithms.