Tales Araujo Leonidas
Professor Patricia McManus
ITAI 2376: Deep Learning
March 21, 2024

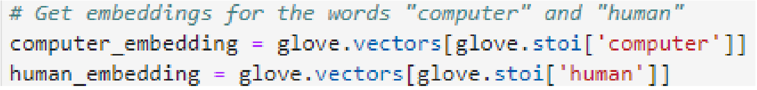
GloVe Word Vectors: Reflective Journal

Introduction

This reflective journal describes my experience, reflection, and considerations performing the AWS Academy Application of Deep Learning to Text and Image Data, focusing on Module 2, Lab 3. The Lab reviewed the concept of word embeddings, guided how to use Global Vectors for Word Representation (GloVe), “an unsupervised learning algorithm for obtaining vector representations for words” (Pennington et al.), and presented cosine similarity to compare words. This report aims to synthesize the key topics to enhance my skills and learning experience.

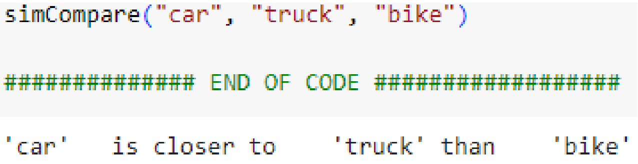
Experience Description

Loading GloVe, which is an enhanced variation of the Word2Vec model, was very simple through *torchtext*, “a companion package to PyTorch consisting of data processing utilities and popular datasets for natural language” (IBM). By using the simplest GloVe embedding model with six billion parameters and 50-dimensional vectors, I generated word embeddings for the words "computer" and "human" to map them into a vector space where words with similar meanings have their vectors close to each other.



```
# Get embeddings for the words "computer" and "human"
computer_embedding = glove.vectors[glove.stoi['computer']]
human_embedding = glove.vectors[glove.stoi['human']]
```

Following the Lab, I utilized *cosine_similarity()* from the *scikit-learn* library, which is “often used to measure document similarity in text analysis” (“Getting to Know Your Data” #39), to calculate the similarity between the word vectors. GloVe embeddings seemed to perform pretty well with little code, even in its simplest form.



```
simCompare("car", "truck", "bike")

############## END OF CODE ##################

'car'   is closer to    'truck' than    'bike'
```

Personal Reflection

The experience was clarifying, presenting a great technique of word embeddings in natural language processing (NLP) with its intuitive practical application. The Lab enhanced my understanding of text processing, especially when the semantic meaning and setting of the relationships between words are needed.

As someone working on a text classification model for detecting harmful content across online platforms, I better understand why my decision tree classifier, the first of the ensemble models in my project, was not capturing the human language complexities. It is because I used a technique effective for tasks where the frequency of word occurrence is paramount, *CountVectorizer*.

Improvements and Learning

Using a pre-trained GloVe embedding model from Hugging Face, which enabled the use of the model with no need to train it, was very interesting. However, my greatest learning during this lab was differentiating when to use GloVe instead of other embedding techniques. I asked ChatGPT4 to concisely differentiate them, and the result is:

Feature	**CountVectorizer**	**GloVe**
Context Sensitivity	Treats words independently.	Considers word context.
Handling of Semantics	Ignores semantics.	Captures semantics.
Applications	Good for spam detection, topic classification.	Suitable for sentiment analysis, machine translation.

Table 1. Short Comparison of CountVectorizer and GloVe.

Conclusion

In conclusion, my journey through this Lab has been crucial in deepening my understanding of word embeddings, specifically through the application of GloVe vectors. The practical experience of implementing this concept to real-world text data, and comparing different vectorization techniques, has significantly enhanced my appreciation for the nuances involved in NLP. It has clarified the importance of context sensitivity and semantic understanding in creating more effective NLP models.

Works Cited

“Getting to Know Your Data.” *Data Mining: Concepts and Techniques, 3rd edition*, by Jiawei Han, et al., Elsevier Science, 2011, p. 703. *ScienceDirect*, https://www.sciencedirect.com/topics/computer-science/cosine-similarity. Accessed 21 March 2024.

IBM. *IBM Watson Machine Learning Community Edition*, 04 March 2021, https://www.ibm.com/docs/sk/wmlce/1.6.1?topic=frameworks-getting-started-torchtext-pytext. Accessed 21 March 2024.

Pennington, Jeffrey, et al. “GloVe: Global Vectors for Word Representation.” *Stanford NLP Group*, August 2014, https://nlp.stanford.edu/projects/glove/. Accessed 21 March 2024.