Detecting Hate Speech with Locality Sensitive Hashing and Word Embedding

Erica Wei, Chuhui Chen, Gary Liu, Samuel Weissmann
Semantic Representations for Natural Language Processing

Introduction

As the proliferation of online hate speech continues to grow, many people turn to natural language processing to help curb it's spread. However, online communities that routinely engage in hate speech often employ simple word substitutions with fictitious, rare, or out-of-context words to avoid detection. In our project, we hope develop and employ novel methods for detecting this style of encoded hate speech.

Data

We relied on two datasets, Ethos Hate Speech and HateXplain. Both datasets contain labelled examples of hate speech and non-hate speech taken primarily from social media.

Methods

Our approach aims to leverage the significant amount of data that can be gleaned from a word's contextual information. LSHWE uses a Nearest Neighbor (NN) Search to identify words that share high amounts of contextual similarities, while an autoencoder helps learn representations for rare or obfuscated words that share contexts with known words. Locality Sensitive Hashing (LSH), which is a clustering method that hashes similar input items into the same buckets, is used to generate our NN matrix and improve performance in word similarity tasks.

Analysis and Results

In Table 1 below, we give the results for select word pairs. Wordsim gives us a standard baseline that we hope to achieve, while we compare our results against those of Word2Vec. The LSHWE column shows the computed similarity using our locality sensitive hashing word embedding technique.

Word 1	Word 2	LSHWE Similarity	Word2vec	Wordsim Similarity
fuck	sex	9.9126	9.9948	9.44
life	death	9.8316	9.9901	7.88
type	kind	9.8914	9.9463	8.97
man	woman	9.7699	9.9977	8.3
experience	music	9.9334	9.9439	3.47

Table 1

In table 2, we compare the runtime and accuracy of LSHWE and Word2Vec classified using a Support-Vector Machine (SVM) learning model.

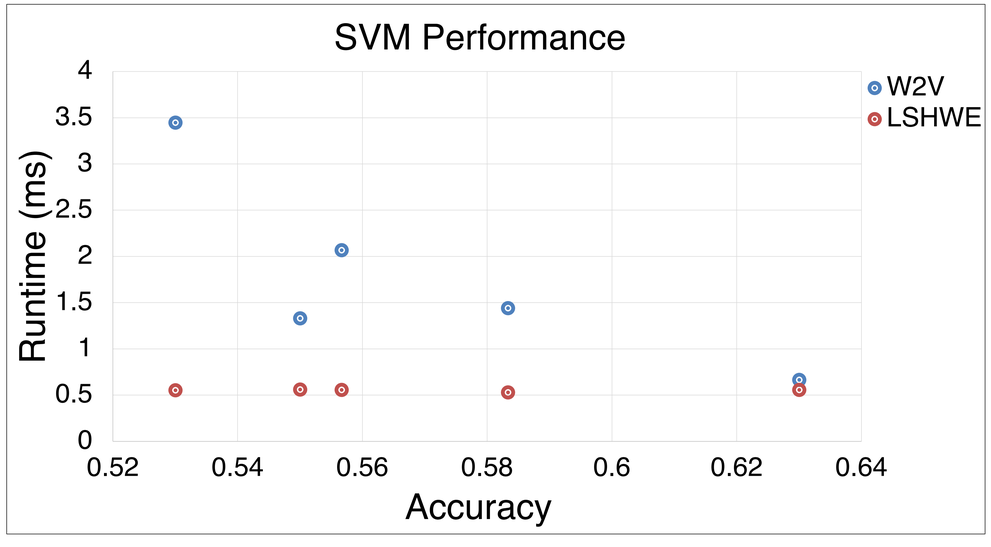

Table 2

Conclusions

While LSHWE was able to compute comparable similarity scores to Word2Vec, both ultimately failed to match their respective Wordsim similarity scores. We posit that this may be due to insufficient high-quality data. In future tests, using subword information may prove helpful as they have shown to improve performance in word similarity tasks where there is insufficient data, e.g. for rare words that do not appear frequently within a corpus.

However, LSHWE does post promising performance metrics, offering improved runtimes over Word2Vec without sacrificing accuracy. This may make it suitable for tasks with expansive datasets, where the performance differences will become even more pronounced

Related Works

1. "*LSHWE: Improving Similarity-Based Word Embedding with Locality Sensitive Hashing for Cyberbullying Detection*", Z. Zhao, M. Gao, F. Luo, Y. Zhang and Q. Xiong, doi: 10.1109/IJCNN48605.2020.9207640.