Text Embedding Techniques for Sentiment Analysis: A Empirical Review

Kh Mohmad Shafi
Assistant Professor Computer science
Direcotrate opf distance education,
University of Kashmir

Shakeel Ahmad Sheikh
Research Scholar,
Istanbul University, Turkey. Email: shakeelzmail608@gmail.com

Abstract: *The most prominent part of the information collection conduct is that what other people think. With the growth of opinion-rich resources like social media, blogs, review sites, various challenges and opportunities arise. People can now use this information to find out and understand the opinions and beliefs of others. Sentiment analysis and text classification is one of the fastest growing research areas in the field of machine learning. The vector representations are the real-valued representations of the words, able to capture the semantic meanings of the words, usually trained on large corpora. These vector representation of words are very helpful in solving many Natural Language Processing (NLP) tasks. So in this context, Word2Vec and GloVe are the two best known representation models so far. In this paper, these two models will be used to evaluate in order to know their effect on the performance metrics of Convolutional Neural Network (CNN) model for sentiment analysis of publicly available IMDB dataset.*

INTRODUCTION:

In recent years, deep learning techniques have attained phenomenal results in the field of computer vision [1] and speech recognition [2]. A lot of work with the deep learning methods has evolved within the scope of Natural Language Processing (NLP) like learning word vector representation of text using neural language models [3] and performing make-up over the learned word vectors for categorization or classification. In natural language processing, one of the interesting trends is the use of text embeddings. Embeddings are mappings of words or phrases from corpora to vectors of real numbers aiming to quantify and categorize semantic similarities [3] between them using the concept of 'the word is known by the company it keeps' (Firth, 1957). Embedding allows to have similar representation for words having similar meaning using the distributional properties learned from a large sample of language corpora [4] by building a low dimensional vectors. In this context, Word2vec [4] and GloVe are the best two well-known methods for producing word embeddings. Experimentally, these two models have been proved to be the most efficient ones in generating word embeddings and in word similarity tasks within the natural language processing. Word2Vec and GloVe have achieved remarkable results in sentiment analysis, opinion mining, text classifications. Howbeit, it is difficult to choose one among these two methods for text embedding generation. In Word2Vec, a distributed representation of a word is used with several hundred dimensions. Each element in the vector contributes to the definition of many words. GloVe is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space. GloVe word embedding is a global log-bilinear regression model and is based on co-occurrence and factorization of matrix in order to get vectors. The automatic sentiment classification and text categorization problems have been investigated by various researchers in recent years. The efficacy of applying deep learning approach in sentiment analysis of text is achieved using conventional techniques like representing movie reviews or any other text opinions using continuous bag-of-words (CBOW) and skip-gram models [4] and different other methods such as MEC (Maximum Entropy Classification), NV (Naïve Bayes) and SVM (Support Vector Machine) [5]–[7]. Convolutional Neural Networks (CNNs) utilize layers with convolving filters that are applied to local features [8]. CNNs were actually invented for computer vision but have subsequently been shown to be very efficient for many NLP tasks and have achieved immense results. According to [9]–[11] . Convolutional Neural Networks (CNNs) have attained prodigious results in the domain of search query retrieval [12], sentiment analysis [13] and text classification problems [14]. In this article, we conduct an experimental study of the two well-known models- Word2Vec and GloVe for sentiment analysis to evaluate their effect on the performance metrics of CNN model using IMDB dataset comprising of 50k user reviews about the movie ratings. In the present work, we train a simple CNN model with 8 layers of convolution on top of the embedding layer obtained through the embedding models- initially by GloVe and then by Word2Vec. For our experimental study, we use 100, 300-dimensional pre-trained embedding vectors of Word2Vec [4] and GloVe [15] .The Word2Vec were trained by [4] using CBOW architecture on 100 billion words of Google news with vocab size of 3M. The GloVe were trained on the combination of Gigaword5 + Wikipedia2014 containing 6 billion tokens with vocab size of 400K [15].

Recently text embeddings have been used for sentence classification using CNN architectures. CNNs have recently achieved remarkably strong performance on the task of sentence classification. A series of experiments using a simple one-layer convolutional neural network built on top of pre-trained word2vec models obtained from an unsupervised neural language model with little parameter tuning for text classification and sentiment analysis was put forward that performed remarkably [9]. [10] offer practical advice by exploring the study of the effect of architecture components of CNNs for sentence classification. [10] conducted a sensitivity analysis of one-layer CNNs to explore the effect of architecture components of CNNs for sentence classification on model performance having results that surpass the baseline methods such as Support Vector Machine (SVM) or logistic regression. [16] proposed a CNN architecture with multiple convolution layers, positing latent, dense and low-dimensional word vectors (initialized to random values) as inputs [11] proposed a model with high dimensional 'one-hot' vector representations of words as CNN inputs. The main focus of these models were on classifications of large texts rather on embeddings. In this work, our primary goal is to study and compare how the text embedding models- GloVe and Word2Vec would affect the two basic sources of error (bias and variance) in sentence classification on small corpora.

EXPERIMENT SETUP:

Requirements	Parameters and Hyper-Parameters
Intel-Core i5 3rd Generation 2.60GHz	Dataset IMDB movie reviews (50k, 25k TrainSet, 25k DevSet)
RAM 8GB	Features = 109766 (Vocab Size)
Python 3.6 (Anaconda Distribution)	Review Max Length = 400 Tokens
Keras (TensorFlow)	Batch Size = 32
Pandas	Embedding Dimensions = 100, 300
	Filters = 250
	No. of epochs = 5

Table 1: Experimental Requirements

MODEL DESIGN

The deep model developed included one embedding layer, one 1D-convolutional layer, its corresponding polling layer, two fully connected layers, one ReLU layer and one sigmoid layer. Keras framework with TensorFlow as backend was used to encode the model. Model was fed pre-trained embeddings from GloVe and Word2Vec with 100 and 300 embedding dimension variants for each.

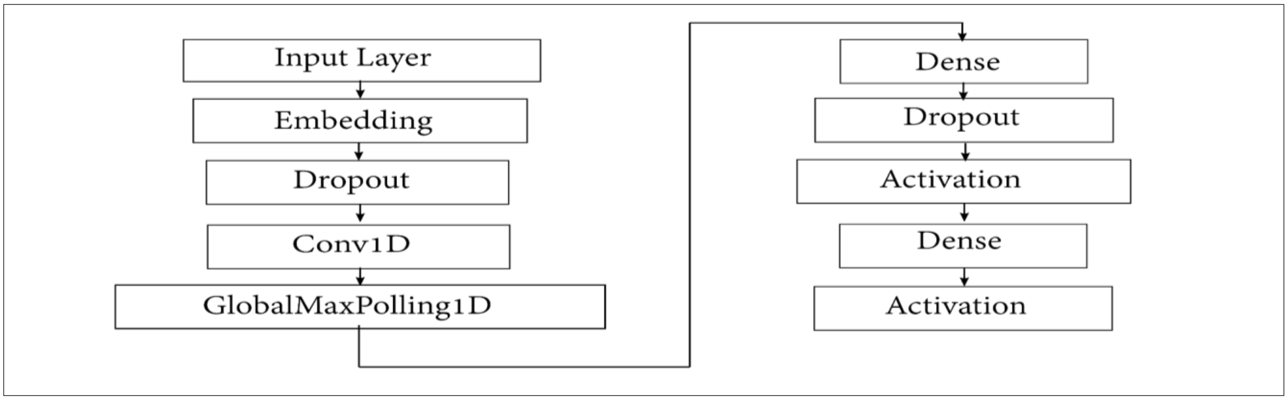

Figure 1: CNN Model

RESULTS

By running the model with different dimension variants of GloVe and Word2Vec, we calculated the training accuracy, development accuracy, avoidable bias-difference between the training error and the unavoidable bias, unavoidable bias also called optimal error rate-difference between the error rate of best possible system and the training error of the system in consideration, variance-the difference between the dev error and the training error. The ideal accuracy for this problem was assumed 100% (human accuracy).

The table 2 summarizes the results calculated from the different model setups

Model ↓	Embedding Dimension	Training Accuracy	Dev. Accuracy	Avoidable Bias *	Unavoidable Bias	Variance
Pre-trained GloVe	100	98.14	85.1	1.86	0	**13.04 (Over fit)**
Pre-trained Word2Vec	100	89.12	85.2	10.88	0	3.92
Pre-trained GloVe	300	99.34	85.5	0.66	0	**13.84 (Over fit)**
Pre-trained Word2Vec	300	93.96	86.5	6.04	0	7.46

Table 2: Results calculated from different model setups ** against 100% ideal accuracy*

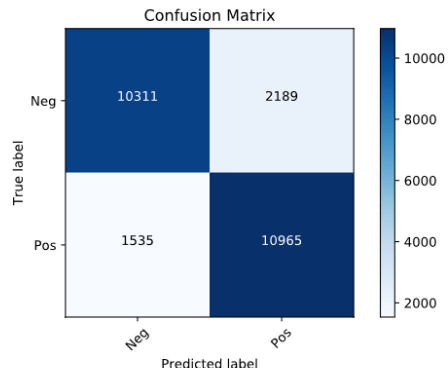

Figure 2: CM of Model using GloVe Embeddings (100 Dimensions)

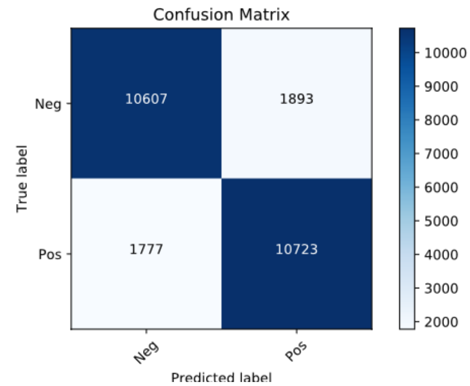

Figure 3: CM of Model using Word2Vec Embeddings (100 Dimensions)

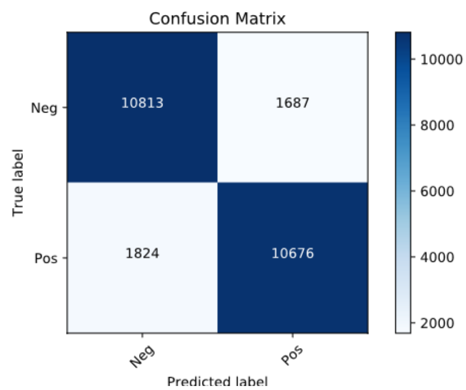

Figure 4: CM of Model using GloVe Embeddings (300 Dimensions)

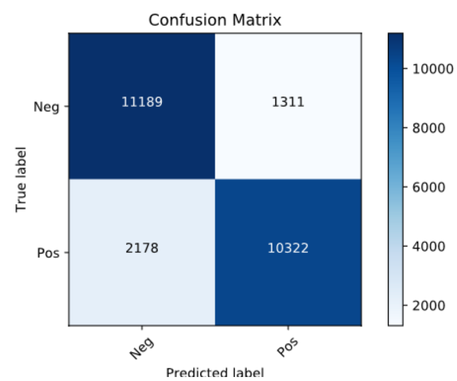

Figure 5: CM of Model using Word2Vec Embeddings (300 mensions)

OBSERVATIONS & CONCLUSION

In both the cases of GloVe embeddings with 100 and 300 dimensions, the models showed high variance and are therefore highly over fitted as compared to their Word2Vec contemporary models. Although, the Word2Vec embeddings with 100 dimensions showed comparatively least variance, but it showed higher avoidable bias and therefore under fitted the data. Word2Vec embeddings with 300 dimensions showed comparatively less avoidable bias but more variance as compared to its 100 dimension version. Also, the Word2Vec with 300 dimensions showed tilt towards negative reviews as evident from Figure 5. Considering the above observations, it can be concluded that Word2Vec embeddings are less prone to over fitting as compared to their GloVe counterparts.

REFERENCES

[1] A. Krizhevsky and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," pp. 1–9.

[2] A. Graves , A.Mohamed, G.Hinton, "Speech Recognition with Deep Recurrent Neural Networks", Department of Computer Science , University of Toronto, no. 6, pp. 6645–

6649, 2013.

[3] Yoshua Bengio, “A Neural Probabilistic Language Model,” vol. 3, pp. 1137–1155, 2003.

[4] T. Mikolov, G. Corrado, K. Chen, and J. Dean, “Vector Space,” pp. 1–12.

[5] B. Pang and L. Lee, “Opinion mining and sentiment analysis,” vol. 2, no. 1, 2008.

[6] B. Pang, *Opinion Mining and Sentiment Analysis*. .

[7] B. Pang, L. Lee, H. Rd, and S. Jose, “Thumbs up ? Sentiment Classification using Machine Learning Techniques,” no. July, pp. 79–86, 2002.

[8] Y. Lecun, L. Bottou, Y. Bengio, and P. Ha, “Gradient-Based Learning Applied to Document Recognition,” no. November, pp. 1–46, 1998.

[9] Y. Kim, “Convolutional Neural Networks for Sentence Classification,” pp. 1746–1751, 2014.

[10] B. C. Wallace, “A Sensitivity Analysis of (and Practitioners’ Guide to) Convolutional Neural Networks for Sentence Classification,” 2014.

[11] R. Johnson, “Effective Use of Word Order for Text Categorization with Convolutional Neural Networks,” no. 2011, 2014.

[12] Y. Shen, “A Latent Semantic Model with Convolutional-Pooling Structure for Information Retrieval,” 2014.

[13] A. Salinca, “Convolutional Neural Networks for Sentiment Classification on Business Reviews,” 2017.

[14] J. Weston and M. Karlen, “Natural Language Processing (Almost) from Scratch,” vol. 12, pp. 2493–2537, 2011.

[15] J. Pennington, R. Socher, and C. D. Manning, “GloVe : Global Vectors for Word Representation.”

[16] E. Grefenstette and P. Blunsom, “A Convolutional Neural Network for Modelling Sentences,” pp. 655–665, 2014.