1.6 Million Tweets Sentimental Analysis

Jinye Cai*
Department of Computer Science
Boston University
Boston, MA 02215
jycai@bu.edu

Xichao Geng
Department of Computer Science
Boston University
Boston, MA 02215
gengxc@bu.edu

Tianmu Hu
Interdisciplinary Program in Bioinformatics
Boston University
Boston, MA 02215
timohu@bu.edu

Sang Yu
Department of Statistics
Boston University
Boston, MA 02215
yusang@bu.edu

Abstract

We present an innovative structure of neural-network model for sentiment analysis. The proposed model is able to predict the sentiment polarity of texts. To extract semantic features of different sizes, a block of Inception CNN (Convolutional Neural Network) with multiple kernel sizes is adopted. To handle sequence pattern of semantic features, bidirectional GRUs (Gated Recurrent Units) in addition with Attention Mechanism are also used. We trained and validated our model on the Kaggle's Sentiment140 Dataset, which contained 1.6 million raw texts of twitter, and improved accuracy from 79.12% to 83.66% according to a baseline of LSTM (Long Short-Term Memory) model. We then demonstrate the effectiveness of our model in extracting insights from tweets of specific identity.

1 Introduction

Twitter [1] is one of the most popular online social network in the world with more than 300 million users, it brings the world's real-time events and the topics that people are currently discussing together, including breaking news, entertainment, sports, politics, and everyday interests. The format of tweets is short messages that users post to share their sentiment and attitude regarding specific topic that they are interested in. Although each tweet can have as many as 280 characters, it commonly does not provide direct sentiment. It's always of great interests for the service providers to perform sentiment analysis via machine learning algorithms to explore underlying information of users' opinions.

Sentiment analysis, also known as opinion mining, is a method to detect the general opinion in the society of a specific topic. For example, "Make America great again!" is a tweet posted by President Trump, which can be defined as strongly positive. Sentiment analysis is widely applied to customer feedbacks such as reviews and survey responses, online and social media, and healthcare materials. Using the knowledge of natural language processing, it is possible to create a model to predict polarity of sentiment, namely positive or negative, for each tweet.

In this study, we designed a model of neural networks to perform two-class categorization for each tweet. In the preprocessing, we tokenized words of all tweets and then present each tweet as a fix-length vector with method of Word Embedding. To imitate human's recognition of semantic features, a combination of CNNs and GRUs is used. A layer of Attention at the bottom of GRUs

*Name of authors are listed in alphabetical order.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

helps memorize patterns of long sentences. The experimental results show this method achieves better accuracy than a baseline of one-layer LSTM model.

The paper is organized as follows. Section 2 gives the design considerations, including preprocessing, word embedding, and size and order of semantic features. The structure and training of Neural Networks is introduced in Section 3 and 4, followed by Section 5 that presents experiment setting and results. Section 6 concludes this study.

2 Design Considerations

2.1 Preprocessing

We removed the punctuations and some stop words, including a, as, the, t, s, at, just, such, an and so. Tokenization [2] is a way to split text into unique tokens. These tokens could be paragraphs, sentences, or individual words. In this paper, we used word tokenization, which generated a word dictionary after processing the whole dataset. The dictionary can be used to covert new input tweets to a sequence of tokens.

2.2 Word Embedding

Word embedding stands for a set of language modeling and feature learning techniques in natural language processing (NLP) where words from the vocabulary are mapped to vectors of real numbers. Global Vectors for Word Representation (GLoVe) [3]. The concept for GLoVe is that based on Word-word co-occurrence probabilities, we have the potential to infer the encoded semantic similarity between words. The ratio of the co-occurrence probabilities of two words contain some information, and GloVe aims at encoding this information to the vector differences. GLoVe is chosen because a count based word embedding method will be computationally costly.

2.3 Size and Order of Semantic Features

Our brain is able to recognize the latent semantic features of different sizes. It's easy for us to extract features of word level as "perfect" and "nice", phrase level as "do not like", and sentence level as "although something negative, I feel positive." For the sentence level case, we not only recognize the words "although", "negative", and "positive", but also realize the meaning of this sequence pattern. Intuitively, the desired model need to extract semantic feature of different sizes, as well as to understand and memorize the sequence of semantic features.

3 Model

3.1 Proposed Structure

We proposed the following structure of neural networks to both recognize semantic features of different sizes and handle sequence patterns.

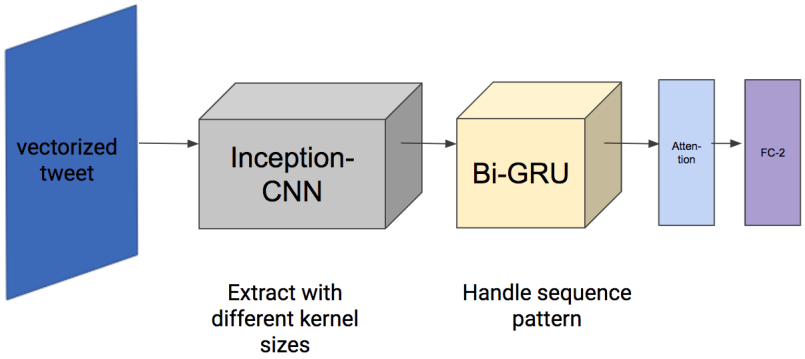

Figure 1: Proposed Model

3.2 Inception CNN

The Inception CNN [4] is an architecture of Convolutional Neural Network that combines different sizes of convolution kernels and concatenates the features together. With this architecture, our model will be able to extract semantic features of different lengths.

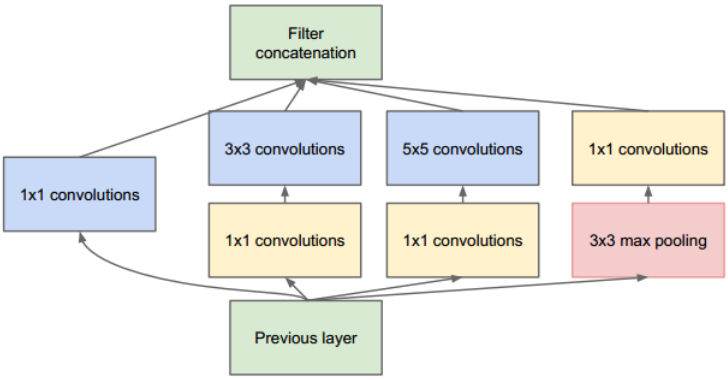

Figure 2: Architecture of Inception CNN

3.3 Gated Recurrent Uunits

Gated recurrent units (GRUs)[5] are a gating mechanism in recurrent neural networks. Similar to long short-term memory (LSTM)[6], GRU is with forget gate but has fewer parameters than LSTM, as it lacks an output gate. With architecture of bidirectional GRUs, our model is able to handle sequence patterns of semantic features.

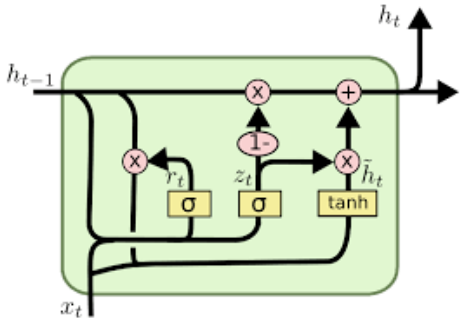

Figure 3: Architecture of GRU

3.4 Attention Mechanism

Attention [7] is proposed as a solution to the limitation of the Encoder-Decoder model encoding the input sequence to one fixed length vector from which to decode each output time step. This issue is believed to be more of a problem when decoding long sequences.

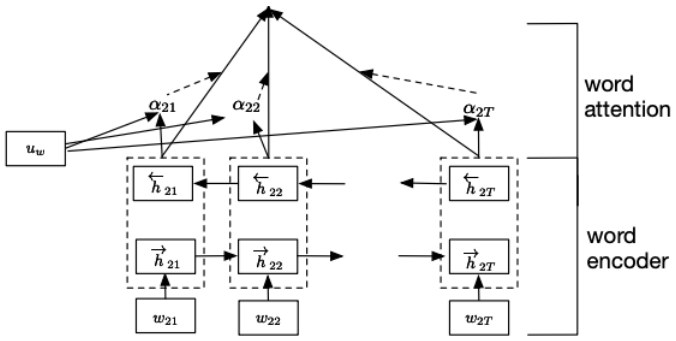

Figure 4: Hierarchical Attention Network

4 Training

The Kaggle's Sentiment140 dataset [8] was used to train and validate the model in this study. The Sentiment140 contained 1.6 million tweets texts extracted using the twitter API. The sentiment label of each tweet has been annotated 0 for negative or 4 for positive respectively. The two-class label were even distributed in this dataset.

We obtained 335,650 unique words from Sentiment140. After preprocessing, each tweet had in average 60 characters, or 11 words. We randomly selected 128 million tweets as training data, and the rest 32 million tweets as validation data. The word embedding we used had 400,000 words with 200 dimensions.

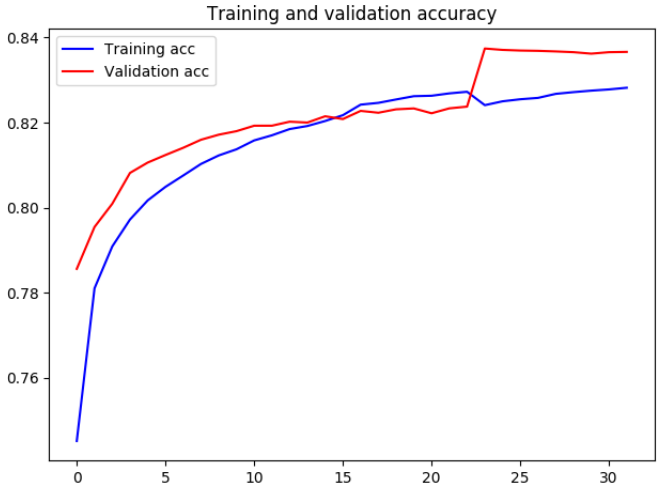

Figure 5: Plot of accuracy after each epoch of model training. X axis indicate epochs of the training process. Y axis shows accuracy. The blue line represents performance of training set and the red line represents that of validation set.

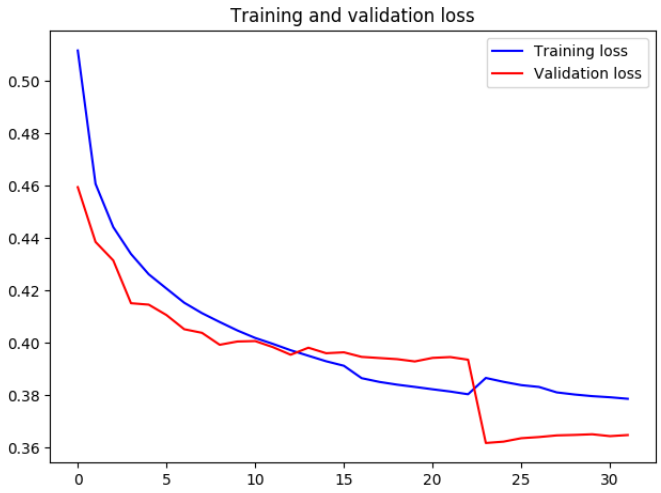

Figure 6: Plot of loss after each epoch of model training. X axis indicate epochs of the training process. Y axis represents loss. The blue line represents performance of training set and the red line represents that of validation set.

After 25 epochs, the model converged and we stopped training. The training accuracy is 0.8282046875 and validation accuracy is 0.8366375. A baseline model, a one-layer LSTM achieved accuracy of 0.7912. Our result improved by about 4% compared to the performance of the baseline model.

	Training Set	Validation Set
Accuracy (%)	82.82	83.66
Loss	0.3768	0.3648

Figure 7: Accuracy and loss value for training and validation

	Our Model	Baseline
Accuracy (%)	83.66	79.12
Loss	0.3648	0.4442

Figure 8: Accuracy and loss value for our model versus baseline

5 Experiments

In the experiment, we labeled a tweet as a neutral one if the absolute value of difference between two predicted probabilities was less than 20%. It was reasonable in real world to have three-class labels, like negative, positive, and neutral.

We collected the last 3200 tweets of President Trump, including 1400 tweets during November of 2018, when was the midterm election of US. For last 3200 tweets, the proportions of negative, positive and neutral labeled by our model were 29%, 56.2%, and 14.8% respectively. For those in November of 2018, the proportion of negative, positive and neutral tweets were 22.5%, 66.9%, 10.6% respectively. The two images clearly showed that President Trump significantly increased his positive tweets, while reduced the negative and neutral tweets during the midterm election of 2018.

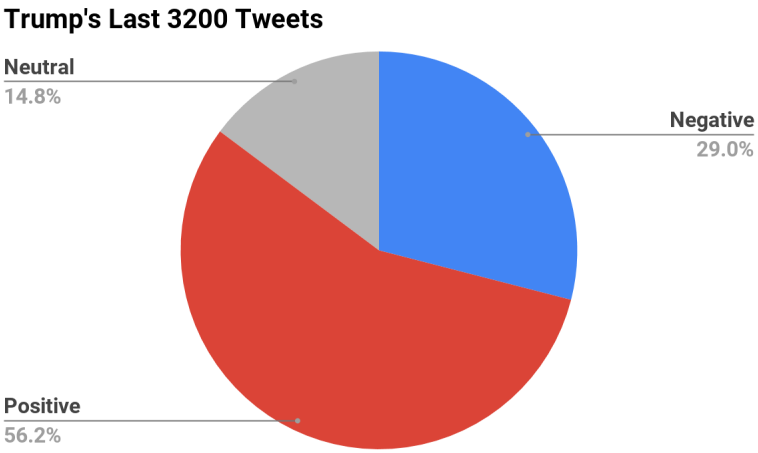

Figure 9: Accuracy and Loss for training and validation

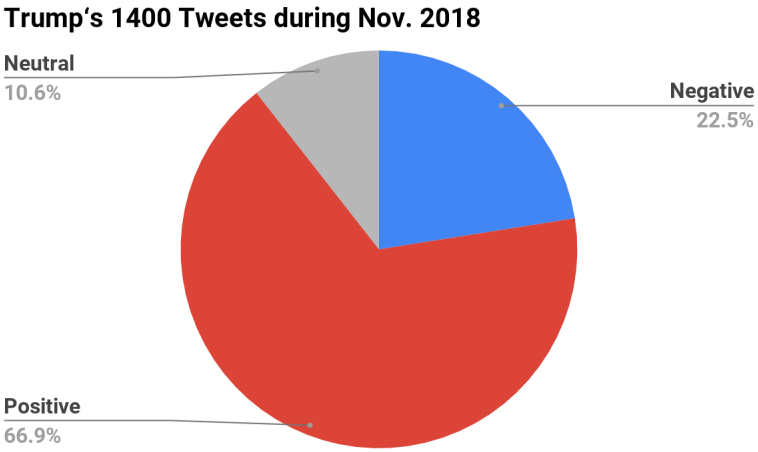

Figure 10: Accuracy and Loss for our model versus baseline

6 Conclusion

The proposed model in this study achieves 83.66% accuracy, which performs 4% better than the baseline LSTM model. The model's complexity is sufficient to learn the knowledge from data set of 1.6 million tweets as it converged after training. The inception block of CNN is an effective way to imitate human understanding of sentence, and Bidirectional GRU is useful to handle sequence patterns. The Attention mechanism is important to memorize features of long sentence. In practical usage, our model is able to monitor the sentiment of tweets of either individual one or a group of identities.

References

[1] Twitter.com. (2019). Twitter. [online] Available at: https://twitter.com/ [Accessed 30 Apr. 2019].

[2] Nltk.org. (2019). Natural Language Toolkit — NLTK 3.4.1 documentation. [online] Available at: https://www.nltk.org/ [Accessed 30 Apr. 2019].

[3] Pennington, J., Socher, R., Manning, C. (2014). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).

[4] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2818-2826).

[5] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

[6] Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

[7] Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E. (2016). Hierarchical attention networks for document classification. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 1480-1489).

[8] Kaggle.com. (2019). Sentiment140 dataset with 1.6 million tweets. [online] Available at: https://www.kaggle.com/kazanova/sentiment140 [Accessed 30 Apr. 2019].