Combining BERT and Multiple Embedding Methods with the Deep Neural Network for Humor Detection

Rida Miraj(✉) and Masaki Aono(✉)

Toyohashi University of Technology, Toyohashi, Japan
aono@tut.jp

Abstract. Humor detection from written sentences has been an interesting and challenging task in the last few years. Most of the prior studies have been explored the traditional approaches of embedding, e.g., Word2Vec or Glove. Recently Bidirectional Encoder Representations from Transformers (BERT) sentence embedding has also been used for this task. In this paper, we propose a framework for humor detection in short texts taken from news headlines. Our proposed framework attempts to extract information from written text via the use of different layers of BERT. After several trials, weights were assigned to different layers of the BERT model. The extracted information was then sent to a Bi-GRU neural network as an embedding matrix. We utilized the properties of some external embedding models. A multi-kernel convolution in our neural network was also employed to extract higher-level sentence representations. This framework performed very well on the task of humor detection.

Keywords: Humor detection · Embedding · BERT · Text · CNN · Bi-GRU

1 Introduction

Humor is a ubiquitous, elusive event that exists all around the world. In previous research and studies, mostly the problems related to humor were based on binary classification or based on the selection of linguistic features. Purandare and Litman analyzed humorous spoken conversations as data from a classic comedy television and used standard supervised learning classifiers to identify humorous speech in the conversation [21]. Taylor and Mazlack used the methodology that was based on the extraction of structural patterns and peculiar structure of jokes newcite [23]. Luke de Oliveira and Alfredo applied recurrent neural network (RNN) and convolutional neural networks (CNNs) to humor detection from reviews in Yelp dataset [10]. The detection of humor from a small and formal sentence is a unique challenge to the research community. To address the challenge of humor detection, Hossain et al. [12] presented a task that focuses on

H. Qiu et al. (Eds.): KSEM 2021, LNAI 12817, pp. 53–61, 2021.
https://doi.org/10.1007/978-3-030-82153-1_5

detecting humor in English news headlines with micro-edits. The edited headlines have one selected word or entity that is replaced by editors, which are then graded by the degree of funniness. Accurate scoring of the funniness from micro-edits can serve as a footstone of humorous text generation [12]. Figure 1 depicts how a single word is replaced with another word to make the sentence funny or humicroedit(dataset formed for this task is named as humicroedit in the article [12]).

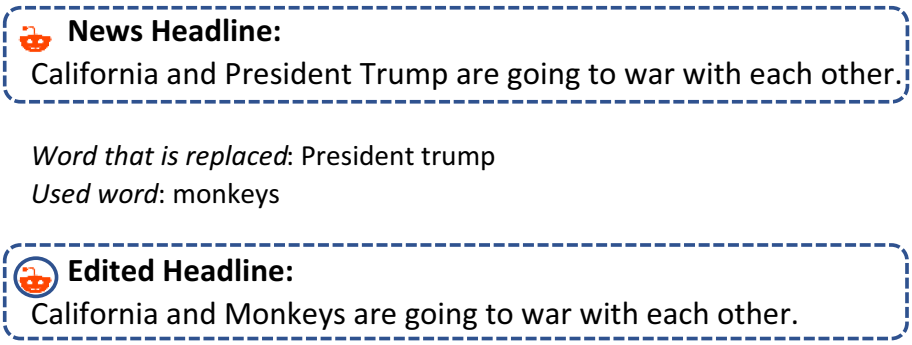

Fig. 1. Example of edited news headline.

However, most of the related work of humor detection explored the traditional way of embeddings in their methods. In this paper, we propose a framework that combines the inner layers information of BERT with Bi-GRU and uses the multiple word embeddings with the multi-kernel convolution and Bi-GRU in a unified architecture. Experimental results on edited news headlines demonstrate the efficacy of our framework.

The rest of the paper is structured as follows: Sect. 2 presents a summary of previous studies. In Sect. 3, we introduce our proposed humor detection framework. Section 4 includes experiments and evaluations. Some concluded remarks of our work are described in Sect. 5.

2 Related Research

In the related work of humor identification, there are a lot of work that is done over the year which includes statistical and N-gram analysis [23], Regression Trees [21], Word2Vec combined with K-NN Human Centric Features, and Convolutional Neural Networks [8]. When working with a limited number of characteristics, neural networks function exceptionally effectively. When dealing with changing length sequences, sequence variants to prior states, as in recurrent neural networks, can be introduced to the network. To identify jokes from non-jokes, several humor detection algorithms include hand-crafted (typically word-based)

characteristics [5,14,17,24]. Such word-based features work well when the non-joke dataset contains terms that are entirely distinct from the humor dataset. According to humor theory, the sequence of words matters, because announcing the punchline before the setup would merely lead to the discovery of the second interpretation of the joke, causing the joke to lose its humorous component [22]. Thus, using a big pre-trained model, such as the latest BERT-like models, is an intriguing fit for the humor detection problem. One potential disadvantage is that these models are not well adapted for comprehending complicated word-play since their tokens are ignorant of relative morphological similarities since the models are oblivious of the tokens' letters [6]. BERT-like models, on the other hand, have done well on English humor recognition datasets [4,26]. In this paper, we used BERT layers for extracting more information regarding words in a sentence and used it as an embedding matrix for our neural network.

3 Framework

In this section, we describe the details of our proposed framework for humor detection. Figure 2 depicts an overview of our proposed framework.

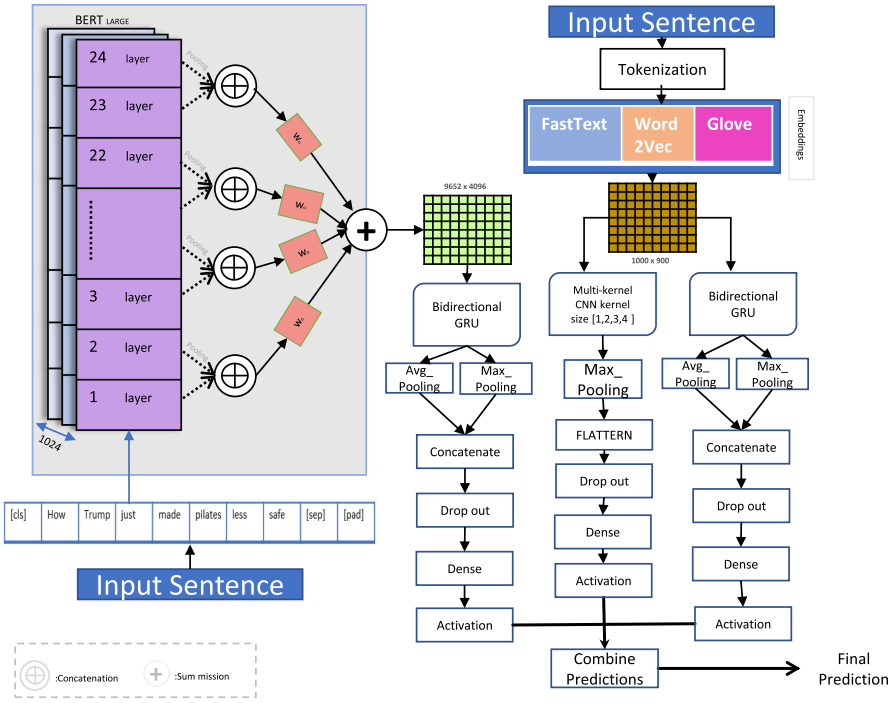

Fig. 2. Proposed framework.

On one side, we utilize the BERT layers for word embedding purposes. The embedding matrix is fed into the embedding layer of our neural network. On the other part, we utilize the multi-kernel convolution filters to extract higher-level feature sequences from the appended embeddings. After getting the predictions from these modules, results are blended and used to determine the degree of funniness. Next, we describe each component elaborately.

3.1 BERT Feature Extraction

BERT [11] is a recent language representation model that has remarkably accomplished well in diverse language understanding benchmarks which indicates the possibility that BERT networks capture structural information about the language. BERT builds on Transformer networks [25] to pre-train bidirectional representations by conditioning on both left and right contexts jointly in all layers. The transformer network inside BERT uses the encoder which has a self-attention layer. The self-attention helps the current node not only focus on the current word but also obtain the semantics of the context. Different BERT layers capture different information. Our target is to extract those hidden information denoted as $\{h_L = h_1, h_2, ..., h_{24}\}$ where L is the no. of layers in BERT model. For extraction, we use two pooling strategies together. One is taking the average of the hidden state of the encoding layer. The second pooling technique is taking the maximum of the hidden state of the encoding layer. The reason to use these strategies is: In average pooling, the meaning of a sentence is represented by all words contained, and in max-pooling, the meaning is represented by a small number of keywords and their salient features only. Two special tokens [CLS] and [SEP] are padded to the beginning and the end of an input sequence, respectively. However, our BERT model is only pre-trained and not fine-tuned on our task, embeddings from those two symbols are meaningless here. After the pooling, the extracted information is concatenated. This extraction method is applied on several layers of the BERT model because every layer has something informative inside it. i.e., the last layer is closed to the training output, so it may give a biased representation towards training targets. The first layer is closed to the word embedding, may preserve the very original word information (with no fancy self-attention) [3]. However, this trade-off between different layers can be beneficial in feature extraction in our task. The rest of the layers are processed accordingly. We can define the above process as follows:

$$h_L^o = AVG(h_L) \odot MAX(h_L), \tag{1}$$

$$h_o^l = (h_L^o) \odot (h_{L-1}^o), \tag{2}$$

$$E_h^o = \sum_{l=1}^{l} \alpha[h_o^l] \tag{3}$$

In above equations, $\odot$ sign is concatenation operation. In next step, the summation E_h^o of concatenated layers are done after adding some weights to them as shown in Fig. 2.

3.2 Embedding

In prior work, target information for humor detection is gained from traditional methods of embedding. As shown in Fig. 2, we also use these embedding techniques in our proposed framework. To integrate the target information, we generate a unified word vector matrix by concatenating the vector representations of the news. The dimensionality of the matrix $E_{glove,fasttext,word2vec}$ will be $L \times D$, where length L is the target length, and D denotes the word-vector dimension. We utilize a pre-trained word embedding model for obtaining the vector representation of words.

3.3 Bi-GRU

Recurrent Neural Network is widely used in the NLP field, which can learn context information of one word. Long Short Term Memory is designed to solve the RNN gradient vanishing problem, especially learning long sentence [7]. Grate Recurrent Unit is a simplified LSTM cell structure [9]. It is a bidirectional recurrent neural network with only the input and forget gates as shown in Fig. 3.

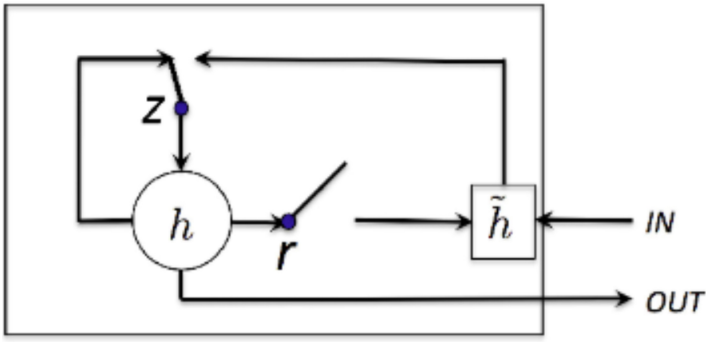

Fig. 3. Simple GRU unit

Taking advantage of its simple cell, GRU can get a close performance to LSTM with less time. With the emerging trend of deep learning, recurrent networks are the most popular for sequential tasks. A Bidirectional GRU, or BiGRU, is a sequence processing model that consists of two GRUs. One taking the input in a forward direction, and the other in a backward direction. It is a bidirectional recurrent neural network with only the input and forget gates. GRUs use less training parameters and therefore use less memory, execute faster and train faster than LSTM's. In our proposed framework, we utilize the Bi-GRU model. The embeddings E_h^o and $E_{glove,fasttext,word2vec}$ passes through the Bi-GRU layer separately. A max & avg pooling functions are then applied which are concatenated to form a feature vector.

3.4 Multi-kernel Convolution

In our multi-kernel convolution, we adopt the idea proposed by [15] to extract the higher-level features. The input of this module is the embedding matrix generated in the embedding layer. We then perform the convolution on it by using a filter. We apply multiple convolutions based on four different kernel sizes, i.e., the size of the convolution filters: 1, 2, 3, and 4. After performing convolutions, each filter generates the corresponding feature maps, and a max-pooling function is then applied to generate a univariate feature vector. Finally, the feature vectors generated from each kernel are concatenated to form a single high-level feature vector.

3.5 Humor Prediction and Model Training

We concatenate the final results from the BERT based Bi-GRU model and Embedding based CNN & Bi-GRU after passing them to a fully connected linear layer for humor detection. We consider mean square error (mse) as the loss function and train the models by minimizing the error, which is defined as:

$$mse = \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{4}$$

where i is the sample index with its true label y_i. $\hat{y}_i$ is the estimated value. We use the stochastic gradient descent (SGD) to learn the model parameter and adopt the Adam optimizer [16].

Table 1. Comparative results with different experimental settings for original news headlines and for edited headlines. The best results are highlighted in boldface.

Embedding	Model	*RMSE*
Baseline		.5750
Our Framework		**.5516**
Edited headlines		
Glove	Bi-GRU	.6291
Glove+FastText	Bi-GRU	.6212
Glove+FastText+GoogleNews	CNN	.6057
BERT (using layers (1, 2, 23, 24))	Bi-GRU	.5879
BERT (using layers (1, 2, 3, 4, 24, 23, 22, 21))	Bi-GRU	.5701
Original headline		
Glove	Bi-GRU	.6311
Glove+FastText	Bi-GRU	.6232
Glove+FastText+GoogleNews	CNN	.6370
BERT (using layers (1, 2, 23, 24))	Bi-GRU	.6194
BERT (using layers (1, 2, 3, 4, 21, 22, 23, 24))	Bi-GRU	.6045

4 Evaluation

4.1 Dataset

To validate the effectiveness of our proposed framework for the humor detection, we made use of a dataset used in the SemEval-2020 Task 7 [13]. The training set consists of 9652 news headlines, the validation set and the test set consist of 2419 and 3024 news headlines respectively. The dataset collected from the Reddit website used for predicting the funniness score. The score ranges from **0** to **3**, where 0 means not funny and 3 means funny among all. Before training, the original headlines are changed with the given edit words. Next, we need to pad the input. The reason behind padding the input is that text sentences have varying length, however models used in our framework expects input instances with the same length. Therefore, we need to convert our sentences into fixed-length vectors. For padding, the maximum sentence length that is set in our framework is 40. For the evaluation measure, root mean square error (rmse) is employed.

$$rmse = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y_i})^2} \tag{5}$$

4.2 Model Configuration

In the following, we describe the set of parameters that we have used in our framework during experiments. We used three embedding models to initialize the word embeddings in the embedding layer. The embedding models are 300-dimensional **FastText** embedding model pre-trained on Wikipedia with 16B tokens [18], 300-dimensional **Glove** embedding model with 2.2M vocab and 840B tokens [20] and 300-dimensional **Word2Vec** embedding model pre-trained on part of Google News dataset [1].

For the multi-kernel convolution, we employed 4 kernel sizes (1, 2, 3, 4), and the number of filters was set to 36. We use Bert-as-Service [27] for extracting information from the BERT model. In our system, the layers of BERT-Large(uncased) are used in the following manner. $h_1^l \odot h_2^l, h_3^l \odot h_4^l, ..., h_{21}^l \odot h_{22}^l, h_{23}^l \odot h_{24}^l$. The framework which we used to design our model was based on TensorFlow [2] and training of our model is done on a GPU [19] to capture the benefit from the efficiency of parallel computation of tensors. We trained all models for a max of 25 epochs with a batch size of 16 and an initial learning rate of 0.001 by Adam optimizer. In this paper, we reported the results based on these settings. Unless otherwise stated, default settings were used for the other parameters.

4.3 Results and Analysis

Our target is to detect the level of funniness from the news headlines that are not supposed to be funny. Here, we used the dataset in two different manners to

show the efficacy of our framework. We showed some summarized experimental results of original and edited headlines both in Table 1. At first, we reported the results based on a naive baseline system. Next, we reported the results of our proposed framework. In order to estimate the effect of each component of our framework, we showed the performances of each component individually. From the results, it can be observed that the simple embedding technique on original and edited headlines gave almost the same RMSE error, which shows it cannot significantly distinguish between being funny or not. For the validation dataset, these techniques produced biased results for most of the cases. However, having in-depth knowledge of a sentence via BERT layers make it better regarding original and edited headlines. We did not perform multi-kernel convolution on BERT based Embedding due to large computational time.

Humor is a difficult achievement for computational models. True humor comprehension would need extensive language expertise as well as common sense about the world to recognize that a first interpretation is being revealed to be incompatible with the second, concealed meaning that fits the entire joke rather than just the premise. Due to the cultural effect and short length of a sentence, this work becomes more challenging. We tries to contribute in this area by proposing the BERT and traditional embedding based neural network. Each component performs efficiently when combine with each other.

5 Conclusion

In this paper, we proposed a framework to detect the level of funniness. The integration of the BERT and external embeddings with Bi-GRU and CNN models provides the great understanding of sentence. The results show the performance of our framework.

In a nutshell, the main contribution of our unified framework is to learn the contextual information effectively which in turn improved the humor detection performance. In the future, we want to use external information to generalize our model for humor identification in the same area.

Acknowledgments. This research was supported by the Japan International Cooperation Agency – JICA under Innovative Asia program.

References

1. Google code archive - long-term storage for google code project hosting, July 2013. https://code.google.com/archive/p/word2vec/
2. Abadi, M., et al.: Tensorflow: a system for large-scale machine learning. In: 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16), pp. 265–283 (2016)
3. Alammar, J.: The illustrated BERT, ELMo, and co. (How NLP cracked transfer learning), December 2018. http://jalammar.github.io/illustrated-bert/
4. Annamoradnejad, I., Zoghi, G.: ColBERT: using BERT sentence embedding for humor detection. arXiv preprint arXiv:2004.12765 (2020)

5. van den Beukel, S., Aroyo, L.: Homonym detection for humor recognition in short text. In: Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 286–291 (2018)
6. Branwen, G.: GPT-3 creative fiction (2020)
7. Cascade-correlation, R., Chunking, N.S.: 2 Previous work **9**(8), 1–32 (1997)
8. Chen, P.Y., Soo, V.W.: Humor recognition using deep learning, pp. 113–117 (2018). https://doi.org/10.18653/v1/n18-2018
9. Chung, J.: Gated recurrent neural networks on sequence modeling. arXiv:1412.3555v1 [cs. NE], pp. 1–9, 11 December 2014
10. De Oliveira, L., Rodrigo, A.L.: Humor detection in yelp reviews (2015). Accessed 15 Dec 2019
11. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: NAACL HLT 2019 – 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference 1(Mlm), pp. 4171–4186 (2019)
12. Hossain, N., Krumm, J., Gamon, M.: "President vows to cut <Taxes> hair": dataset and analysis of creative text editing for humorous headlines (iv), pp. 133–142 (2019). https://doi.org/10.18653/v1/n19-1012
13. Hossain, N., Krumm, J., Gamon, M., Kautz, H., Corporation, M.: SemEval-2020 Task 7: assessing humor in edited news headlines (2019) (2020)
14. Kiddon, C., Brun, Y.: That's what she said: double entendre identification. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 89–94 (2011)
15. Kim, Y.: Convolutional neural networks for sentence classification, pp. 1746–1751 (2014)
16. Kingma, D.P., Ba, J.L.: A: a m s o, pp. 1–15 (2015)
17. Mihalcea, R., Strapparava, C.: Making computers laugh: investigations in automatic humor recognition. In: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, pp. 531–538 (2005)
18. Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., Joulin, A.: Advances in pre-training distributed word representations. In: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018) (2018)
19. Owens, J.D., Houston, M., Luebke, D., Green, S., Stone, J.E., Phillips, J.C.: GPU computing. Proc. IEEE **96**(5), 879–899 (2008)
20. Pennington, J.: Blue. https://nlp.stanford.edu/projects/glove/
21. Purandare, A., Litman, D.: Humor: prosody analysis and automatic recognition for f* r* i* e* n* d* s. In: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, pp. 208–215 (2006)
22. Ritchie, G.: Developing the incongruity-resolution theory. Technical report (1999)
23. Taylor, J.M., Mazlack, L.J.: Computationally recognizing wordplay in jokes theories of humor (1991) (2000)
24. Taylor, J.M., Mazlack, L.J.: Computationally recognizing wordplay in jokes. In: Proceedings of the Annual Meeting of the Cognitive Science Society, vol. 26 (2004)
25. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems (NIPS), pp. 5999–6009, December 2017
26. Weller, O., Seppi, K.: Humor detection: a transformer gets the last laugh. arXiv preprint arXiv:1909.00252 (2019)
27. Xiao, H.: BERT-as-service (2018). https://github.com/hanxiao/bert-as-service