ECNU at SemEval-2020 Task 7: Assessing Humor in Edited News Headlines Using BiLSTM with Attention

Tiantian Zhang, Zhixuan Chen, Man Lan*
Department of Computer Science and Technology,
East China Normal University, Shanghai, P.R.China
51194506050,10165102127@stu.ecnu.edu.cn, mlan@cs.ecnu.edu.cn

Abstract

In this paper we describe our system submitted to SemEval 2020 Task 7: "Assessing Humor in Edited News Headlines". We participated in all subtasks, in which the main goal is to predict the mean funniness of the edited headline given the original and the edited headline. Our system involves two similar sub-networks, which generate vector representations for the original and edited headlines respectively. And then we do a subtract operation of the outputs from two sub-networks to predict the funniness of the edited headline.

1 Introduction

Humor can be defined as the aspiration of provoking laughter and provides amusement from expressions intended (Bertero and Fung, 2016). The task of humor recognition refers to determining whether a sentence in a given context contains some level of humorous content.

The Semeval 2020 Task 7 (Hossain et al., 2020a) aims to automatically computes the funniness of edited news headlines which are generated using an insertion of a single-word noun or verb to replace an existing entity or single-word noun or verb in original headline (Hossain et al., 2019). There are two sub-tasks in Task 7. The sub-task 1 is to predict the mean funniness of the edited headline given the original and edited headline. The sub-task 2 is based on sub-task 1, which aims to determine which version of edits makes the headline more humorous given the original headline and two edited versions.

In prior studies, humor recognition has been approached as a binary classification problem. Traditional classification algorithms like SVM and Naive Bayes (Mihalcea and Strapparava, 2005), and a deep learning CNN architecture (Chen and Lee, 2017) are adopted to distinguish between humorous and non-humorous texts. However, humor is not just a binary concept and it occurs in various intensities. In addition, in the past, the research objective for humor recognition is a sentence or text. However, it is interesting to study how short edits applied to a text can turn it from non-funny to funny, which can help us focus on the humorous effects of atomic changes and pointing out the key difference between non-humorous and humorous text.

In this paper, we present our funniness prediction system which is mainly based on bidirectional LSTM (Hochreiter and Schmidhuber, 1997) neural networks with attention mechanism (Bahdanau et al., 2014). Besides, we show some features related to humor and then analyze the effectiveness of different configurations of our system.

2 System Description

We develop a framework including two similar sub-networks whose inputs are the sequence of tokens in the original headline and edited headline respectively. Then two outputs of sub-networks and other text features are combined together to predict the mean funniness of the edited headline. Figure 1 represents the network structure of our overall model, with two similar sub-networks displayed in two sides. The highlighted word, in original headline is the replaced word, and in edited headline is the replacement word. Specifically, each sub-network contains the token representation layer using pre-trained word embeddings,

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 995–1000
Barcelona, Spain (Online), December 12, 2020.

BiLSTM layer and attention layer. We apply BiLSTM to obtain contextual token representations. We also use attention mechanism in order to get the headline representations.

2.1 Pre-trained Word Embeddings

The input to each sub-network is a news headline, treated as a sequence of tokens. We use a token representation layer to project the headline $W = (w_1, w_2, ..., w_t)$ to a low-dimensional vector space $\mathbb{R}^E$, where E is the size of the representation layer and t is the number of tokens in the headline. By projection, the sequence of tokens can be represented as $X = (x_1, x_2, ..., x_t)$. We obtain token representations using GloVe word vectors (Pennington et al., 2014) and BERT pre-trained word embeddings (Devlin et al., 2018). Here, we just use BERT as a feature extraction model to extract token features.

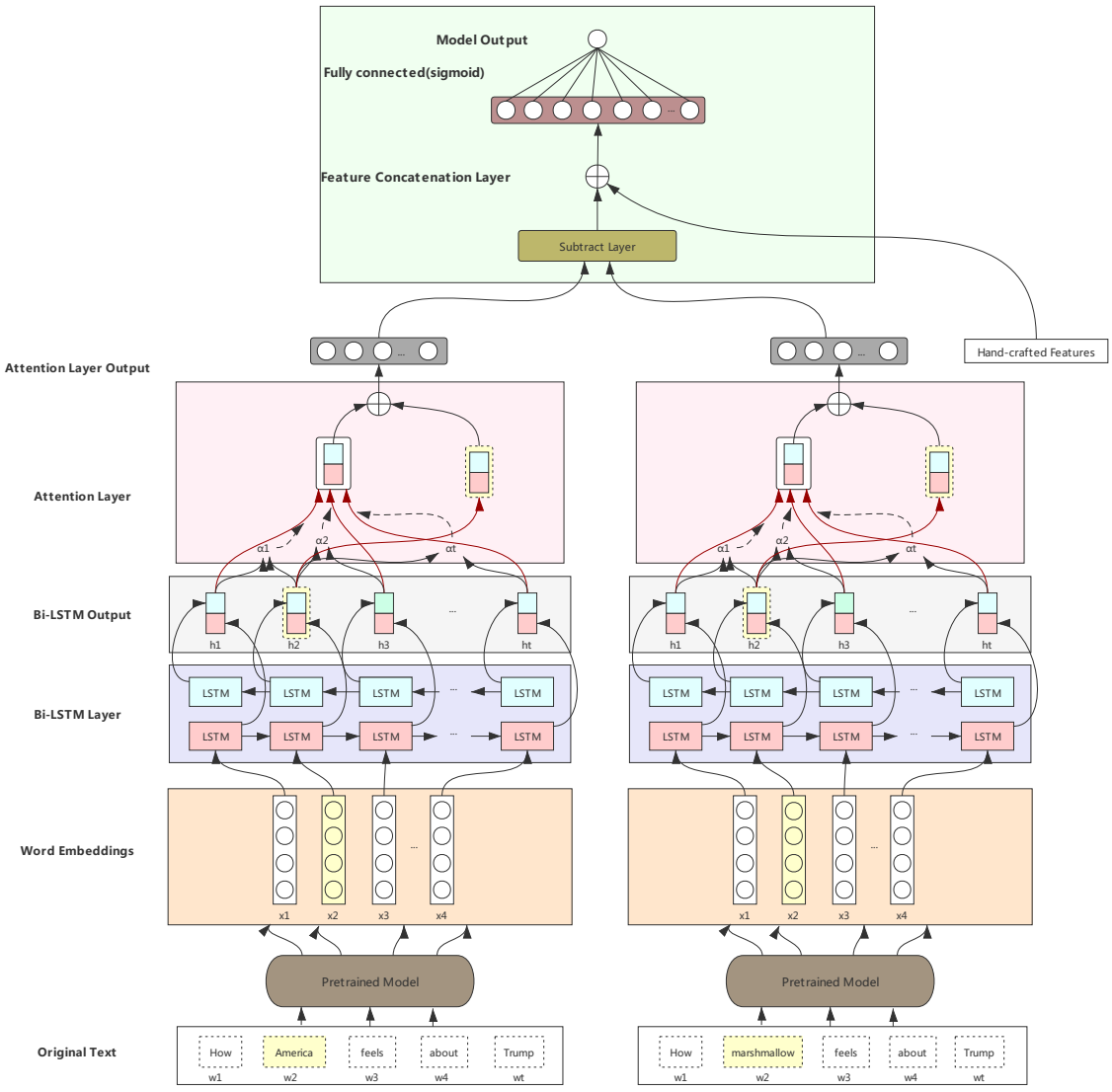

Figure 1: Overall Model Structure

2.2 BiLSTM Layer

We use a BiLSTM over a sequence of tokens to obtain token representations $H = (h_1, h_2, ..., h_t)$. As shown in figure 1, a BiLSTM encodes the sequence twice, once forward and once backward. A forward LSTM processes the sequence from x_1 to x_t, while a backward LSTM processes from x_t to x_1. For word x_i, a forward LSTM and backward LSTM produce the token representation as $\overrightarrow{h_i}$ and $\overleftarrow{h_i}$. Finally, the

overall output h_i is calculated as follows:

$$h_i = \overrightarrow{h_i} \oplus \overleftarrow{h_i} \tag{1}$$

where $\oplus$ denotes the concatenation operation. Particularly, $h_i \in \mathbb{R}^{2L}$, L is the size of LSTM.

2.3 Attention Layer

Because of the incongruity theory of humor (Morreall, 2016), we suppose that the distance between the new replacement word and other words in the edited headline is further than that between the replaced word or entity and other words in the original headline. Attention mechanism (Bahdanau et al., 2014) can capture the relationship between two texts, including words, sentences, etc. Therefore we use attention in our model. The attention mechanism assigns a weight w_i to each output h_i of the BiLSTM layer except for the replaced word and replacement word. The hidden states are finally calculated to produce a hidden sentence feature vector r by a weighted sum function, as indicated in figure 1 by arrows. Formally:

$$e_i = \tanh(W_h h_i + b_h) \tag{2}$$

$$w_i = \frac{\exp(e_i)}{\sum_{j=1}^{t} \exp(e_j)} \tag{3}$$

$$r = \sum_{i=1}^{t} w_i h_i, r \in \mathbb{R}^{2L} \tag{4}$$

The parameters W_h and b_h above are the weight and bias from the attention layer.

2.4 Features

In this paper, we also design some hand-crafted features and they are directly added in output layer.

Statistical features: By doing statistical analysis of replaced words and replacement words that generate humor on news headlines, we counted the occurrence of each replaced word and replacement word. We used the humor grade of an edited headline as its replaced word and replacement word's humor grade. And then we calculated the *average*, *minimum* and *maximum* humor grades of all replaced words and replacement words respectively. These three statistical data for the replaced words and replacement words respectively are used as our hand-crafted features, 6 feature numbers in total. As for a new replaced or replacement word in test dataset, we use features of the word most similar to current word by computing word similarity instead.

Sentiment lexicon features: SenticNet 5 (Cambria et al., 2018) is used to calculate the sentiment polarity of words in headlines. The sum of the words' polarity in original headline and edited headline separately are used as two of our hand-crafted features.

Humor lexicon features: Humor Norm Lexion (Engelthaler and Hills, 2017) is used to calculate the humor grades of words in headlines. The sum of the words' humor grades in original headline and edited headline separately are used as two of our hand-crafted features.

2.5 Output Layer

For both the original headline and edited headline, their representation can be obtained by concatenating the output of attention layer and replaced word representation or the replacement word representation separately. Considering that the incongruity of the replaced word and the replacement word, we make a subtract operation for the representation of the original headline and the edited headline. Then the vector concatenating the output and extracted features above is fed to final fully-connected sigmoid layer which outputs a humor grade.

3 Experimental Setup

3.1 Dataset

Experiments are conducted on Humicroedit dataset (Hossain et al., 2019) and additional training data collected from the FunLines competition (Hossain et al., 2020b). We follow the standard data partition of Semeval 2020 Task 7.

3.2 Evaluation metrics

Following the official evaluation criteria given in the competition, the Root Mean Squared Error (RMSE) is adopted for sub-task 1 to measure the predicted values and the ground truth mean funniness. The classification accuracy is adopted for sub-task 2. The definitions are as follow:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{N}} \tag{5}$$

where $\hat{y}_i$ and y_i represent the predicted outputs and gold labels, respectively.

$$Accuracy = \frac{T}{N} \tag{6}$$

where N is the number of overall samples, and T denotes the correct number of predicted samples.

3.3 Training

To predict the funniness of the edited headlines, we trained our model to minimize MSE loss with rmsprop optimizer. We applied BERT and GloVe pre-trained embeddings, two-layers BiLSTM with 128 hidden units and a dropout of 0.5 to the all BiLSTM layers. At the output layer, we tried two ways to predict the final funniness. One is to predict humor grade using sigmoid function and the other is to predict 4 values which represent the percentage of grade 0, 1, 2 and 3 scored by judges using softmax function.

4 Results

Since sub-task 2 is based on the predictions of sub-task 1 to a great extent, we only report the results of sub-task 1. As described in section 2.1, we trained the model based on GloVe embeddings and BERT pre-trained embeddings. Table 1 shows that the model based on BERT outperforms GloVe embeddings in dev set. This means that humor recognition benefits from token representations based on context, which accords with our cognition.

Models	**Sub-Task 1**
Baseline	0.57840
GloVe-based	0.54980
BERT-based	**0.53929**

Table 1: Performance of baseline and our models on dev set. `BERT-based` means that our model is based on BERT pre-trained embeddings and a single neuron layer in output layer. `GloVe-based` means the model based on GloVe embeddings.

Table 2 lists the effectiveness of attention mechanism and statistical features. The system performance drops by 0.015 when ablating attention mechnism in BERT-based model. This indicates that attention mechanism is very important to capture the relation between the replaced words, replacement words and headlines.

Likewise, statistical features make contribution, but we find that this feature type is limited to models and has little potential in performance improvements. Apart from this feature, we also try sentiment lexicon features and humor lexicon features. However, adding these features to the model has no real

performance increase, so we don't report these results here. This is probably because the inappropriate representations of features and rich semantics and all sorts of ironies in news headlines.

Besides, if we use the percentage of grade 0, 1, 2 and 3 as our outputs, the performance is slightly better than predictions using one single output.

According to experimental results, we take the combination of the GloVe-based, BERT-based and BERT-based (4 output) model with statistical features as the ensemble model. The RMSE result in dev set of sub-task 1 is 0.52205 and the accuracy in sub-task 2 is 0.63078, which show that these three models are complementary in predicting funniness. And this ensemble model is used to predict the humor score in test set.

Models	**Sub-Task 1**
BERT-based	0.53929
- Attention	0.55451
+ Statistical features	**0.53036**
BERT-based (4 output)	0.53329
- Attention	0.57446
+ Statistical features	**0.53324**

Table 2: Performance of our models on dev set. -Attention ablates the attention mechnism. + Statistical features adds statistical features to the model. 4 output means 4-neurons layer in prediction layer.

	Our system	**Rank 1**	**Rank 2**
Sub-Task 1	0.52187 (6)	0.49725 (1)	0.50726 (2)
Sub-Task 2	0.64384 (6)	0.67428 (1)	0.66058 (2)

Table 3: Performance of our system, top-ranked systems for sub-task 1, 2. The numbers in the brackets are the official rankings.

Table 3 shows the official evaluation results of rank 1 and rank 2. Compared to other systems, our system has a lot of room for improvement, especially in identifying sarcasm.

5 Conclusion

In this paper, we present a deep learning model which contains two similar sub-networks. We design a two-layers BiLSTM with attention mechanism model, whose inputs are the original headlines and edited headlines. And then we predict the funniness of the edited headline by a subtract operation between the outputs of the two sub-networks mentioned above and concatenating the hand-crafted features. The experimental results show this method can assess the intensity of humor to some extent.

News headlines are short and concise, and humor often comes from phrases and common sense. In Semeval 2020 Task 7, there are all sorts of ironies in edited headlines, but no effective NLP tools can recognize them so far. In the future, we consider to introduce external knowledge to model headlines and improve the humor recognition performance.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate.

Dario Bertero and Pascale Fung. 2016. A long short-term memory framework for predicting humor in dialogues. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 130–135, San Diego, California, June. Association for Computational Linguistics.

Erik Cambria, Soujanya Poria, Devamanyu Hazarika, and Kenneth Kwok. 2018. Senticnet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings. In *AAAI*.

Lei Chen and Chong Min Lee. 2017. Convolutional neural network for humor recognition. *ArXiv*, abs/1702.02584.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. cite arxiv:1810.04805.

Tomas Engelthaler and Thomas T. Hills. 2017. Humor norms for 4,997 english words. *Behavior Research Methods*, 50:1116 – 1124.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November.

Nabil Hossain, John Krumm, and Michael Gamon. 2019. "president vows to cut <taxes> hair": Dataset and analysis of creative text editing for humorous headlines. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 133–142, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Nabil Hossain, John Krumm, Michael Gamon, and Henry Kautz. 2020a. Semeval-2020 Task 7: Assessing humor in edited news headlines. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2020)*, Barcelona, Spain.

Nabil Hossain, John Krumm, Tanvir Sajed, and Henry Kautz. 2020b. Stimulating creativity with FunLines: A case study of humor generation in headlines. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 256–262, Online, July. Association for Computational Linguistics.

Rada Mihalcea and Carlo Strapparava. 2005. Making computers laugh: Investigations in automatic humor recognition. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, page 531–538, USA. Association for Computational Linguistics.

John Morreall. 2016. Philosophy of humor. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, winter 2016 edition.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.