HIMA-Net: Humor prediction by self-attention based on key information related to humor

Hang Qin[1, *], Mengnan He[2, †], Hanmin Jia[3, †]
[1] Department of Electronic and Information Engineering, South China University of Technology, Guangzhou, Guangdong, 510000, China
[2] Department of Physics, Tsinghua University, Beijing, 100000, China
[3] Department of Computer science, The University of Sheffield, Sheffield, S10 2TN, UK
[*]Corresponding author: eesimonqin@mail.scut.edu.cn
[†]These authors contributed equally

ABSTRACT

Humor is a high-level semantic emotion that can only be understood at a stage when the human mind has developed. Humor detection is a challenging task in the field of natural language processing. In this paper, we focus on the characteristics of humor from the way it is generated and propose the Humor Important Message Attention Net (HIMA-Net): a self-attention network based on the key messages related to humor. Results show that HIMA-Net outperforms the traditional models on three datasets (Headline, Pun, Short Jokes), and further analysis demonstrates the effectiveness of the proposed model.

Keywords: Humor detection; Natural language processing; Lightweight network; Self-Attention; Sequence representation model

1. INTRODUCTION

Humor detection is a challenging yet highly meaningful and valuable task. In the field of natural language processing (NLP), humor detection plays an important role. Therefore, many methods based on deep learning for humor detection have emerged in recent years, and they have achieved good results on different datasets.

As a classical sequence representation model, the long short-term memory framework [1] is widely used in humor recognition tasks, including predicting humor in dialogues [2] and computational modelling of conversational humor in psychotherapy [3]. Apart from LSTM, as an effective feature extraction model, CNN has also been applied to humor detection tasks [4-5]. Considering the correlation between emotion and humor [6-7], emotion-related features have also been added to the humor detection model [8]. These studies have demonstrated the effectiveness of emotion-related features in enhancing the humor detection ability of the model. In recent years, with the introduction of transformer structures and BERT [9], they have been used for humor detection, and these methods have achieved better results than ever before [10-11].

Although the transformer structure and BERT have made considerable progress on various tasks, the transformer structure also has limitations. First, the transformer structure only consists of self-attention and feed forward neural network. Hence, the transformer structure perhaps tends to exploit global context information in a short text, which leads to the possibility of ignoring local contextual information. In addition, due to the transformer structure, the models using the transformer structure are usually more computationally expensive. They cannot make good use of the position information of the words in the text.

We propose a humor detection model called Humor Important Message Attention Net (HIMA-Net) for the above considerations. It uses a BERT-based pre-trained text encoder with 1d-CNN and bi-directional LSTM with attention mechanism. It aims to improve the performance of traditional networks with the outstanding text-encoding ability of BERT, while using CNN and LSTM in combination enable the model to focus on not only global information but the local contextual information of the text and improve the representation ability of the model. In our model, CNN can extract local information, which is the key humorous fragment in the sentence. At the same time, LSTM with the attention layer can sufficiently consider the position information of words in the text and make full use of this key humor-related information.

2021 International Conference on Neural Networks, Information and Communication Engineering, edited by Zhiyong Zhang, Proc. of SPIE Vol. 11933, 119330Q · © 2021 SPIE · 0277-786X · doi: 10.1117/12.2615166

Hence, their combination precisely addresses the potential issues in the transformer structure and traditional networks such as CNN and LSTM. Experiments on three real-world datasets demonstrate the effectiveness of the proposed model.

2. METHOD

In this section, we will introduce an overview of the proposed HIMA-Net and then present the details of each module.

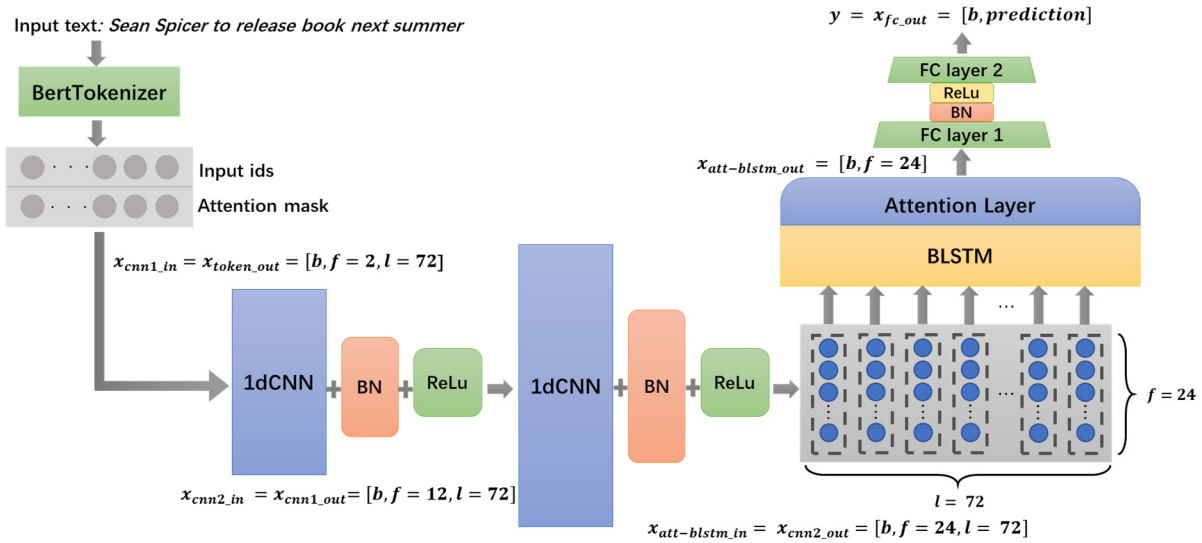

Figure 1. An overview of the HIMA-Net

The framework of the HIMA-Net is shown in figure 1. The proposed model contains a pre-trained Bert-based text encoder known as the BertTokenizer, two layers of one-dimensional Convolution Neural Network, a Bi-directional Long-Short Term Memory network with an attention layer, and two fully connected layers. CNNs and LSTM are connected in a pipeline manner.

As shown in Figure 1, firstly, the model gets the encoded representation through the pre-trained BertTokenizer. Then, the CNN module extracts the key information related to humor from the encoded text and increases the feature dimensions. These features of the key information related to humor will be fed into the Att-BLSTM, which is the BLSTM with the self-attention mechanism. Att-BLSTM module takes full use of these humor-related features, extracting the text's global information, and gets a whole-text representation related to humor. Finally, the fully connected layers reduce the dimensions of the features, and the model outputs the humor prediction result for the input text.

In Figure 1, we denote the input and output of each module by x and denote the final output of the whole network by y. We mark the forms of the data processed by each module as $[b, f, l]$, where b represents the batch-size, f represents the number of feature dimensions, and l represents the sequence length.

2.1BertTokenizer

The BertTokenizer aims to extract the basic features of the input text and generates an encoded representation of the text. When the text is inputted into the BertTokenizer, the text is firstly divided into tokens, and each token is converted into a token id by using a lookup table. Then, the BertTokenizer extracts features from text data, which are the embedding vectors of words and sentences. These vectors are used as high-quality features representing the original text.

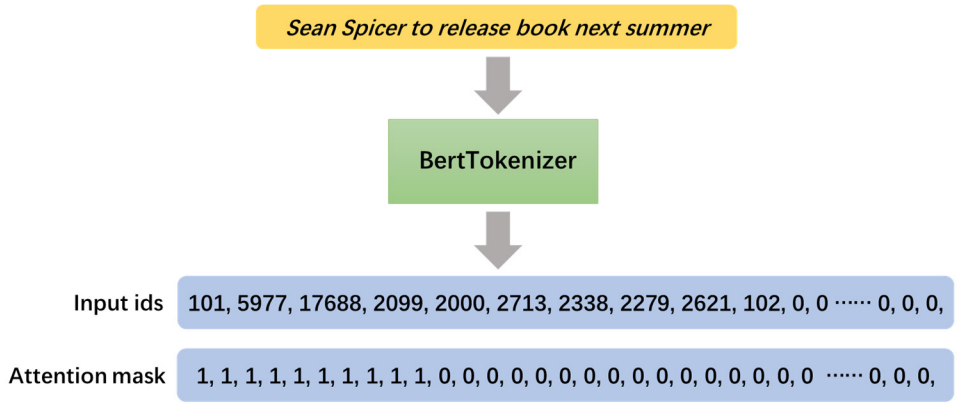

Figure 2. An example of the use of BertTokenizer.

As shown in figure 2, in HIMA-Net, we use the BertTokenizer to encode the text data into a two-dimensional feature vector. One dimension is the input ids that represent the text content, another dimension is the attention mask of padding and truncation since we padding the sequences to the same length. Note that we choose not to padding the text data of three datasets to one unique length because of the differences between these data sets. For the sentences in Short Jokes and Headline, we padding them to 72, and for those in the Pun dataset, we only padding them to 48.

Finally, taking the Short Jokes dataset as an example, the output size of the BertTokenizer is:

$$[batch\_size, feature\_num = 2, sequence\_length = 72]$$

2.21d-CNN

Convolutional Neural Network (CNN) has been widely used in various fields of deep learning because of its outstanding capability in local information extraction. 1d-CNN is the CNN that performs convolutional operations in only one dimension, and it has been widely used in many natural language processing tasks [12]. In HIMA-Net, 1d-CNN is used to extract local information related to humor in text, which is the parts in the sentences that contribute more to humor. These highly relevant features to humor are of great help in our humor detection task.

The CNN module in HIMA-Net contains two layers of 1d-CNN. The first layer of 1d-CNN increases the feature dimension of data from 2 to 12 by using the convolutional kernel size 3. The second layer has the convolution kernel size of 1 and further increases the feature dimension to 24 so that the model can extract higher-level features related to humor. We use Batch Normalization and ReLU after each 1d-CNN layer to alleviate overfitting.

The final output size of the CNN model is:

$$[batch\_size, feature\_num = 24, sequence\_length = 72]$$

2.3Att-BLSTM

The Attention-Based Bidirectional Long Short-Term Memory Networks (Att-BLSTM) [13] is the structure adding attention layer to the traditional bi-directional LSTM network. This kind of attention mechanism is useful in the relation classification task. It is implemented in this way. Firstly, consider a matrix H consisting of output vectors$[h_1, h_2, \ldots, h_T]$ that the BLSTM layer generated, where T is the length of the sequence. Then, by a weighted sum of these vectors, the representation r is formed:

$$M = tanh(H) \tag{1}$$

$$\alpha = softmax(w^T M) \tag{2}$$

$$r = H\alpha^T \tag{3}$$

where $H \in \mathbb{R}^{dw} \times T$, d^w is the feature dimension corresponding to each node on the length, w is parameter vector that needs to be learned in training and w^T is the transpose of w. The dimension of w, α, r is d^w, T, d^w respectively. The final sequence representation from:

$$r = H\alpha^T \tag{4}$$

With this Att-BLSTM module, HIMA-Net can further extract the global features of sentences and take into account the position and order information in sentences. The attention mechanism allows the model to assign more weights to humor-related features and improves the performance of humor prediction.

Finally, the output size of Att-BLSTM is

$$[batch\_size, feature\_num = 24]$$

3. EXPERIMENT AND RESULTS

In this section, we will introduce our experiments, including the datasets, the setup of the experimental virtual environment, and the results of the experiments.

3.1Datasets

We used three datasets that can be used for short text humor recognition to train and test our model, which are Headline (Humicroedit [14] + FunLines [15]), Pun of the day [16], and Short Jokes [5][10]. Table1 shows the basic information of these datasets.

Table 1. The statistical characteristics of the three datasets

Dataset	Type	Total number	#Pos	#Neg
Headline	Regression	38438	\	\
Pun	Binary classification	32003	16001	16002
Short Jokes	Binary classification	475302	237651	37651

The Headline dataset contains 23,343 headlines, of which 15,095 are from FunLines, and 8,248 are from Humicroedit. Each headline contains both original and edited versions. Five volunteers rated the humor of the edited headline on a scale from 0-3, representing “not funny at all” to “very funny”. We use the mean of the five scores as labels to train our model and let the model predict the humor ratings of the headlines. Table 2 is an example from the Headline dataset. In this headline, the “man” is changed to "toddler". Obviously, the edited headline is funnier than the original one. On this dataset, our model predicts the humor level of the edited headline.

Table 2. Example of Headline dataset

Data	Grades	Mean Grade
Sen.Rand Paul assaulted at his Kentucky home, **man** arrested	32222	2.2
Sen.Rand Paul assaulted at his Kentucky home, **toddler** arrested		

The Pun of the Day dataset was constructed from the Pun of the Day website. It contains 16001 puns and 16002 not-pun sentences. Puns, also known as paronomasia, is a relatively esoteric form of humor that uses multiple meanings of a word or harmonizes words to achieve the expected humorous or rhetorical effect. The negative sample for this dataset was collected from news websites. Table 3 illustrates an example from Pun, where a label of 1 means pun and 0 means not-pun.

Table 3. Example of Pun of the Day dataset

Data	Label
I’m thinking of something like an animal hospital she said	0
Let me share with you a couple of keys to playing piano	1

The Short Jokes dataset is the largest of the three datasets we used, and it comes from an open database on a Kaggle project. It contains 231,657 short jokes from different joke websites, and their length ranging from 10 to 200 characters. These jokes are positive samples. For the negative samples, they are from WMT162 English News. These carefully selected negative samples contain words that all appear in the positive samples, and their average text length is close to the humorous ones. Table 4 shows an example from the Short Jokes dataset.

Table 4. Example of the Short Jokes dataset

Data	Label
The laker’s led – after one quarter despite committing eight turnovers in the first	0
Why do rabbits make good mathematicians because they are constantly multiplying	1

3.2Experiment settings

Our experiments are conducted in a python 3.7 environment, and the model framework is based on Pytorch 1.6. We divided the dataset into train set, dev set, and test sets in the ratio of 8:1:1, and trained our model based on the train set, using the dev set to fine-turning the hyperparameters and tested the final performance of the model on the test set. We set the initial epochs to 200 and used early stopping, using a mini-batch of size 30.

3.3Evaluation

Since the Headline dataset uses humor scores as labels, we use the Root Mean Squared Error (RMSE) as the evaluation criterion for this dataset, with a smaller RMSE indicating the better performance of the model.

$$RMSE = \sqrt{\frac{1}{M}\sum_{m=1}^{M}(y_m - \hat{y}_m)^2} \tag{5}$$

Where y_m is the actual value, $\hat{y}_m$ is the predicted value, and M is the size of the test sample.

Our work on the Pun dataset and the Short Jokes dataset is entirely based on the binary classification. Therefore, we used the corresponding evaluation metrics to make a comprehensive assessment of our model, including Accuracy, Precision, Recall, and F1-score.

In typical binary classification scenes, we assume that the given samples can be divided into two categories: Positive and Negative. The evaluation metrics we used is defined as below:

$$RMSE = \sqrt{\frac{1}{M}\sum_{m=1}^{M}(y_m - \hat{y}_m)^2} \tag{6}$$

$$Precision = \frac{TP}{TP+FP} \tag{7}$$

$$Recall = \frac{TP}{TP+FN} \tag{8}$$

$$F1\ score = \frac{2*Precision*Recall}{Precision+Recall} \tag{9}$$

Where TP, TN, FP, FN represent the correct predicted positive sample, the correct predicted negative sample, the wrong predicted positive sample, and the wrong predicted negative sample, respectively.

In our humor binary classification task, the mentioned metrics could be interpreted as the following. The Accuracy represents the percentage of texts categorized into the correct category by our classifier. The Precision represents the percentage of actual funny texts in all texts categorized as funny by our classifier. The Recall represents the percentage of correctly labelled texts in all funny texts from our corpus, and the F1-score is the combination of Precision and Recall.

3.4Results

To prove the effectiveness of the HIMA-Net, we selected a series of baselines and compared their performance on three datasets. There are 4 models as baselines, including CNN, BLSTM, Att-BLSTM, and CNN-BLSTM. They are all classical models in the field of deep learning and natural language processing. The experiment result shows that each of our improvements to the baseline improves the performance.

Table5, Table6, and Table7 show the performance of our model and the baseline model on the Headline (Humicroedit + FunLines) dataset, the Pun dataset, and the Short Jokes dataset, respectively. The baseline BLSTM scores the lowest in all evaluation metrics, whereas our proposed model scores the highest. Each additional component we add to the baseline BLSTM shows a positive influence on overall performance. CNN and attention layer both help to improve the Accuracy and Recall scores significantly, but less so in Precision. From the tables, it is clear that our proposed model HIMA-Net achieves the best RMSE in the regression task of predicting the level of humor, and the best results in all four evaluation-metrics in the classification task of judging humor or not, which indicates that our proposed HIMA-Net outperforms the baselines in its ability to extract humor-related information and process humor-related information the text more effectively.

Table 5. Results on Humicroedit + FunLines

Model	BLSTM	Att-BLSTM	CNN	CNN-BLSTM	**HIMA-Net**
RMSE	0.57161	0.57135	0.57032	0.57025	**0.57010**

Table 6 Results on Pun of the day

Model	BLSTM	Att-BLSTM	CNN	CNN-BLSTM	**HIMA-Net**
Accuracy	0.717	0.752	0.823	0.825	**0.856**
Precision	0.838	0.793	0.824	0.811	**0.844**
Recall	0.512	0.659	0.806	0.832	**0.865**
F1	0.636	0.720	0.815	0.821	**0.854**

Table 7 Results on Short Jokes

Model	BLSTM	Att-BLSTM	CNN	CNN-BLSTM	**HIMA-Net**
Accuracy	0.614	0.619	0.826	0.858	**0.868**
Precision	0.591	0.591	0.831	0.868	**0.876**
Recall	0.746	0.782	0.821	0.845	**0.859**
F1	0.670	0.673	0.826	0.856	**0.867**

4. DISCUSSION

Several findings are obtained from the experiment results, which are discussed in detail below. What we found was that all improvements we made based on the Baseline improve overall performance. The HIMA-Net combining CNN, Bi-LSTM, and Attention performs the best. It has the lowest RMSE and scores the highest in all evaluation metrics we use to assess classification performance.

Our use of BertTokenizer for pre-processing our corpus has improved model performance, which verifies our assumption that BertTokenizer can generate high-quality encoding based on given texts. The combination of CNN and BLSTM scored significantly higher than each alone when looking at the Recall metric, but a bit lower in Precision, which correlates to the different features of CNN and BLSTM. CNN's ability to extract key information helps to locate funniness, while BLSTM's memory capacity may be more useful in separating funny and boring texts. The series structure of CNN and BLSTM could consider both local and global features, bringing two network structures' strengths together. This could explain why combining the two makes finding funny texts (Recall) easier but separating two categories (Precision) slightly more difficult. The additional Attention layer also showed positive influence in almost all metrics. The use of the Attention layer makes the model assign more weight on the key information related to humor, which brings improvement in both classification and regression, helping our model focus more on humor-related features. Similar to that of CNN, only adding Attention to BLSTM also brings a lower score in Precision.

5. CONCLUSION

This paper proposes HIMA-Net, an effective method to detect humor in short texts. We conducted comparative experiments on three short-text datasets. All the results indicate that our model performs better than previous, classical ones, showing that our work could be quite inspiring in the field.

However, current methods can detect humor only by extracting humor-related features from texts. It remains a challenging task for our model to fully understand humor itself. Therefore, our future work will focus on generating more representative features for humor, and hopefully, it could contribute more in the area of humor detection.

REFERENCES

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

[2] Bertero, D., & Fung, P. (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 (pp. 130-135).

[3] Ramakrishna, A., Greer, T., Atkins, D. C., & Narayanan, S. S. (2018). Computational Modeling of Conversational Humor in Psychotherapy. In INTERSPEECH (pp. 2344-2348).

[4] Chen, L., & Lee, C. M. (2017). Convolutional neural network for humor recognition. arXiv preprint arXiv:1702.02584.

[5] Chen, P. Y., & Soo, V. W. (2018). Humor recognition using deep learning. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, (volume 2: short papers) (pp. 113-117).
[6] Charles R Gruner. (1997). The game of humor: A comprehensive theory of why we laugh. Transaction Publishers
[7] Herbert Spencer et al. (1860). The physiology of laughter. Macmillans Magazine pages (pp. 395–402).
[8] Liu, L., Zhang, D., & Song, W. (2018). Modeling sentiment association in discourse for humor recognition. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers) (pp. 586-591).
[9] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
[10] Weller, O., & Seppi, K. (2019). Humor Detection: A Transformer Gets the Last Laugh. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 3612-3616).
[11] Mao, J., & Liu, W. (2019). A BERT-based Approach for Automatic Humor Detection and Scoring. In IberLEF@ SEPLN (pp. 197-202).
[12] Johnson, R., & Zhang, T. (2015). Effective Use of Word Order for Text Categorization with Convolutional Neural Networks. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 103-112).
[13] Zhou, P., Shi, W., Tian, J., Qi, Z., Li, B., Hao, H., & Xu, B. (2016). Attention-based bidirectional long short-term memory networks for relation classification. In Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: Short papers) (pp. 207-212).
[14] 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) (pp. 133–142)
[15] Hossain, N., Krumm, J., Sajed, T., & Kautz, H. (2020). 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 (pp. 256-262).
[16] Yang, D., Lavie, A., Dyer, C., & Hovy, E. (2015). Humor recognition and humor anchor extraction. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (pp. 2367-2376).