YNU-HPCC at SemEval-2020 Task 8: Using a Parallel-Channel Model for Memotion Analysis

Li Yuan, Jin Wang and Xuejie Zhang
School of Information Science and Engineering
Yunnan University
Kunming, China
Contact:{wangjin, xjzhang}@ynu.edu.cn

Abstract

In recent years, the growing ubiquity of Internet memes on social media platforms, such as Facebook, Instagram, and Twitter, has become a topic of immense interest. However, the classification and recognition of memes is much more complicated than that of social text since it involves visual cues and language understanding. To address this issue, this paper proposed a parallel-channel model to process the textual and visual information in memes and then analyze the sentiment polarity of memes. In the shared task of identifying and categorizing memes, we preprocess the dataset according to the language behaviors on social media. Then, we adapt and fine-tune the Bidirectional Encoder Representations from Transformers (BERT), and two types of convolutional neural network models (CNNs) were used to extract the features from the pictures. We applied an ensemble model that combined the BiLSTM, BIGRU, and Attention models to perform cross domain suggestion mining. The officially released results show that our system performs better than the baseline algorithm. Our team won nineteenth place in subtask A (Sentiment Classification). The code of this paper is availabled at : https://github.com/YuanLi95/Semveal2020-Task8-emotion-analysis.

1 Introduction

In recent years, memes that combine pictures and text have been widely used in social media. Using memes can help users to express richer meaning and emotion compared with using text or images alone; hence, it is worthwhile to analyze the sentiment expressions of memes.Moreover, recognizing and analyzing the meaning and sentiment of memes is much more difficult than analyzing social texts or pictures.

In SemEval-2020 Task 8:Memotion Analysis (Sharma et al., 2020), the organizers hoped that the task would increase the research attention given to the topic. The task is divided into three subtasks.

• Task A- Sentiment Classification: Given an Internet meme, the first task is to classify its sentiment polarity.

• Task B- Humor Classification: Given an Internet meme, the system has to identify the type of humor expressed.

• Task C- Scales of Semantic Classes: The third task is to quantify the extent to which a particular effect is being expressed.

Memes and this issue have attracted the attention of researchers. In a previous study, Borth (2013) pioneered the sentiment analysis of visual content with SentiBank. Another study implemented Optical Character Recognition (OCR) to extract the text captions of memes and then classified the sentiment polarity of the text using the Naive Bayes algorithm (Amalia et al., 2018). For a similar meme sentiment analysis task, Zhao (2019) developed a multimodal sentiment analysis method for image-text posts, and their experiments showed that this method achieves excellent performance on the Flickr benchmark dataset. Hu and Flaxman (2018) used GloVe to map the text to a high dimensional space and fine-tuned the pictures through Inception (a pretrained deep convolutional neural network).

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 916–921
Barcelona, Spain (Online), December 12, 2020.

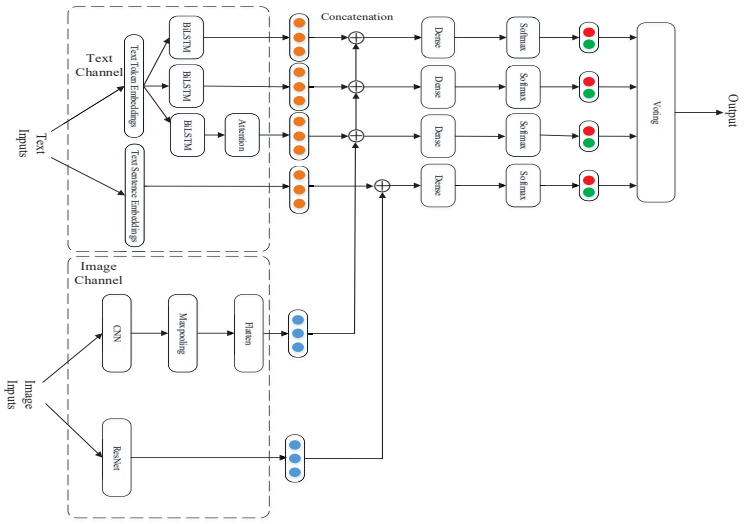


Figure 1: The Multimodal architecture of the parallel channel model.

In this paper, we propose a parallel channel model that includes a text channel, which is implemented to process the text in memes, and an image channel for image analysis. The text channel implements the BiLSTM, BiGRU, and BiLSTM with attention models. For the image channel, a multilayer CNN model and ResNet152 (He et al., 2016) were applied to capture the image features. Then, the information in the two modalities is combined by a dense layer after concatenation. The experimental results show that our approach achieved good performance.

The remainder of this paper is organized as follows. Section 2 describes the proposed parallel channel model, and Section 3 presents the implementation details and experimental results. The conclusions of this study are presented in Section 4.

2 Parallel Channel Model

2.1 Overview

As illustrated in Figure 1, the proposed model consists of two channels: the image channel and the text channel. We propose two different types of pretraining vectors and three different models in the text channel and two different models in the image channel as a way to extract picture features. We combined multiple models, use the soft voting mechanism, and output the results. For an input meme, w_s represents the extracted text and I is the image. Then, the proposed model can be expressed as follows:

$$\begin{aligned} h_i^T &= f_i^T(w_s) \\ h_j^I &= f_j^I(I) \\ f(w_s, I) &= voting\left[h_i^T \oplus h_j^I\right] \end{aligned} \tag{1}$$

where $i \in (1, 2, 3, 4)$ and $j \in (1, 2)$, f^T and f^I represent the way to obtain special text and image features. h_i^T and h_j^I are the text vector and the image vector, respectively,and $f(w_s, I)$ is the final result.

2.2 Text Channel

Embedding Layer. The embedding layer is the first layer of the text channel. We constructed the word vectors from a 768-dimensional BERT vector. Then, a word vector matrix was loaded into the embedding layer and then fed into different hidden layers. For longer posts, we only keep the first 128 words, which is a reasonable choice since 90% of the posts in the dataset contain less than 128 words. We also use the sentence-level vectors from a 768-dimensional BERT vector as the text features and fed them into the fully connected layer.

Bidirectional Long Short-Term Memory (BiLSTM) (Greff et al., 2017) is a special Recurrent Neural Network. The LSTM model can better capture the long-distance dependencies. There are various novel

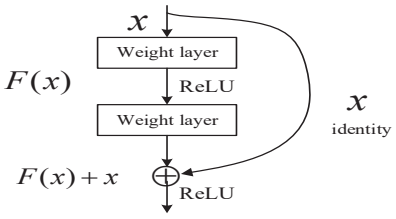


Figure 2: Residual learning: a building block.

models based on LSTM, for instance: Wang (2020) proposed a tree-structured regional CNN-LSTM model for valence-arousal (VA) prediction. A capsule tree LSTM model introduces a dynamic routing algorithm to construct sentence representations (Wang et al., 2019), and experiments prove that the method improves the performance of the tree LSTM and the basic LSTM model. BiLSTM is based on LSTM and can better capture forward and backward semantic dependencies. We show how a memory block calculates the hidden state h_t^T and output C_t using the following equations.

- **Gate**

$$\begin{aligned} f_t &= \sigma(W_f \cdot [h_{t-1}^T, x_t] + b_f) \\ i_t &= \sigma(W_i \cdot [h_{t-1}^T, x_t] + b_i) \\ o_t &= \sigma(W_o \cdot [h_{t-1}^T, x_t] + b_o) \end{aligned} \tag{2}$$

- **Transformation**

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}^T, x_t] + b_c) \tag{3}$$

- **Status update**

$$\begin{aligned} C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t \\ h_t^T &= o_t * \tanh(C_t) \end{aligned} \tag{4}$$

here, x_t is the input vector; C_t is the cell state vector; W and b are cell parameters; f_t , i_t and o_t are gate vectors; and σ denotes the sigmoid function.

Gated Recurrent Unit (GRU) (Cho et al., 2014) is a variant of LSTM that combines the forget gate and the input gate into a single update gate. It also mixes cell states and hidden states. The final model is simpler than the standard LSTM model. The effect is similar to LSTM but with fewer parameters, and it is not easy to overfit.

Attention mechanism (Bahdanau et al., 2015) breaks the limitation that the traditional encoder-decoder structure depends on a fixed-length vector when encoding and decoding. Its implementation retains the intermediate output results of the input sequence via the LSTM encoder, trains a model to selectively learn these inputs and associates the output sequence with it when the model is output. Attention mechanisms have been widely used in various NLP fields such as the Transformer (Vaswani et al., 2017), Neural Machine Translation (Yang et al., 2016) and aspect-level sentiment analysis (Tang et al., 2019).

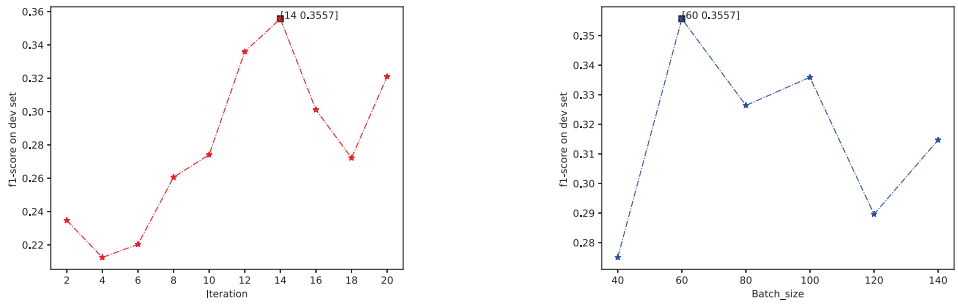
2.3 Image Channel

Convolutional neural networks (CNNs) (Krizhevsky et al., 2012) are often used to extract image representations. A CNN is usually divided into convolution layers and pooling layers. The convolution layers are used to extract n-gram features from the picture pixels. Pooling selects a part of the input matrix and chooses the best representative for the region. The max pooling layer selects the max feature.

ResNet model (He et al., 2016) is one of the widely used image recognition models, and it solves the deep vanishing gradient problem. The basic structure of the residual is shown in Figure 2. We used PyTorch's pretrained ResNet152 model for the feature extraction from pictures.

3 Experiments and Evaluation

In this section, experiments were conducted to evaluate the proposed models on both subtasks. We also report the results of the official review. The details of the experiment are described as follows.



(a) Fine tune of epochs (b) Fine tune of batch size

Figure 3: Fine tune of epochs and batch size.

	Model-Name	$h_{1,2}$	h_3	r	d
Text-channel	BILSTM	300	160	0.4	128
	BIGRU	300	300	0.3	128
	BILSTM-Attention	300	160	0.3	64
Picture-channel	**Model-Name**	c	m	l	p
	Multilayer CNN	6	64	3	0.2

Table 1: The best-tuned parameters for Task A.

3.1 Data Preparation

The organizers provided 7K human annotated Internet memes labeled with semantic dimensions, namely, sentiment and the type of humor that is sarcastic, humorous, or offensive. For subtask A and subtask B, the data distributions are a little unbalanced, which make the tasks much harder. We randomly used 20% of the memes from the provided data as the dev set to fine-tune the parameters. The Stanford tokenizer toolkit was employed to process the memes-text into an array of tokens. Meanwhile, before feeding the token array to any neural networks, they are preprocessed by following procedures:

- Punctuation marks, websites URLs and mailing addresses are removed,
- Common nonstandard expressions are restored, and
- Non-English letters are treated as unknown words represented by <unk>.

3.2 Implementation Details

This experiment used Keras with the TensorFlow backend. For subtask A, we used two different pretrained word vectors, and we introduced other models. For subtask A, we tried different batch sizes and attempts, and the results are shown in Figure 3.The best batch size is 60, the best number of training epochs is 14 and the learning rate is set as 1e-5. We use Scikit-Learn to execute the grid search (Pedregosa et al., 2011) to adjust the hyperparameters, through which we can find the best parameters for evaluating the system. The parameters given are as follows: the time step of the RNN for hidden layers 1, 2 ($h_{1,2}$) and 3 (h_3); the dimension of the dense layer (d); and the dropout rate (r). For the image channel, we also have the number of convolution layers (c), the number of filters (m), the length of the filter (l) and the pool (p). Table 1 summarizes these fine-tuned parameters.

3.3 Evaluation Metrics

For the submission to subtask A and subtask B, its performance will evaluated based on the macro-F1 score. The F1-score is often used as an evaluation indicator of unbalanced data, and is defined as follows:

$$F1 = 2 * \frac{P * R}{(P + R)} \tag{5}$$

Model	Metrics	
	Task A	Task B
BiLSTM_{elmo}	0.286	0.498
BiGRU_{elmo}	0.304	0.512
$\text{BiLSTM} + \text{Attention}_{elmo}$	0.311	0.506
BiLSTM_{bert}	0.323	0.523
BiGRU_{bert}	**0.338**	0.531
$\text{BiLSTM} + \text{Attention}_{bert}$	0.328	0.528
Bert + ResNet	0.334	**0.536**
$\text{OurModel}_{\text{ensemble}}$	**0.3557**	**0.541**

Table 2: The dev data experiment results.

where P denotes the precision and R denotes the recall. A higher F1-score indicates better classification performance.

3.4 Results and Discussion

Table 2 shows the detailed results of the proposed our model compared to the other baseline models in ours dev set.

Subtask A. Our system achieved a score that was 0.115 higher than the baseline score (0.2176). The results show that our proposed system significantly outperforms the baseline models. The main reason is that we have combined a variety of information from memes and used the BERT word embedding.

Subtask B. Our model score was lower than the baseline score of 0.5118. We guess that it may be caused by the inconsistent data distribution between the dev set and test set, and so we need to do more research on class imbalance in the future.

4 Conclusion

In this paper, we describe a task system that we submitted to SemEval-2020 for Memotion Analysis. We propose a two parallel channel model. In the text channel, we use 3 RNN models and 2 types of pretraining vectors. In the image channel, we used a pretrained model and a CNN model. We participated in subtasks A and B, and obtained nineteenth place in subtask A. In future work, we will test more novel fusion methods so that the picture features can be better combined with token embedding.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant No. 61966038, 61702443 and 61762091. The authors would like to thank the anonymous reviewers for their constructive comments.

References

Amalia Amalia, Arner Sharif, Fikri Haisar, Dani Gunawan, and Benny B Nasution. 2018. Meme opinion categorization by using optical character recognition (ocr) and naïve bayes algorithm. In *2018 Third International Conference on Informatics and Computing (ICIC)*, pages 1–5.

Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pages 1–15.

Damian Borth, Rongrong Ji, Tao Chen, Thomas Breuel, and Shih-Fu Chang. 2013. Large-scale visual sentiment ontology and detectors using adjective noun pairs. In *MM 2013 - Proceedings of the 2013 ACM Multimedia Conference*, pages 223–232.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pages 1724–1734.

Klaus Greff, Rupesh K. Srivastava, Jan Koutnik, Bas R. Steunebrink, and Jurgen Schmidhuber. 2017. LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 28(10):2222–2232.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 770–778.

Anthony Hu and Seth Flaxman. 2018. Multimodal sentiment analysis to explore the structure of emotions. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 350–358.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc.

Fabian Pedregosa, Ron Weiss, and Matthieu Brucher. 2011. Scikit-learn : Machine learning in Python. *the Journal of machine Learning research*, 12:2825–2830.

Chhavi Sharma, Deepesh Bhageria, William Paka, Scott, Srinivas P Y K L, Amitava Das, Tanmoy Chakraborty, Viswanath Pulabaigari, and Björn Gambäck. 2020. SemEval-2020 Task 8: Memotion Analysis-The Visuo-Lingual Metaphor! In *Proceedings of the 14th International Workshop on Semantic Evaluation (SemEval-2020)*, Barcelona, Spain, Sep. Association for Computational Linguistics.

Jialong Tang, Ziyao Lu, Jinsong Su, Yubin Ge, and Linfeng Song. 2019. Progressive self-supervised attention learning for aspect-level sentiment analysis. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 557–566. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 2017-Decem, pages 5999–6009.

Jin Wang, Liang-Chih Yu, K. Robert Lai, and Xuejie Zhang. 2019. Investigating dynamic routing in tree-structured LSTM for sentiment analysis. pages 3430–3435.

Jin Wang, Liang Chih Yu, K. Robert Lai, and Xuejie Zhang. 2020. Tree-structured regional CNN-LSTM model for dimensional sentiment analysis. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 28:581–591.

Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. 2016. A character-aware encoder for neural machine translation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, number 95, pages 3063–3070.

Ziyuan Zhao, Huiying Zhu, Z. Xue, Zhao Liu, Jing Tian, Matthew Chin Heng Chua, and M. Liu. 2019. An image-text consistency driven multimodal sentiment analysis approach for social media. *Information Processing and Management*, 56(6):102097.