guoym at SemEval-2020 Task 8: Ensemble-based Classification of Visuo-Lingual Metaphor in Memes

Yingmei Guo[1], **Jinfa Huang**[2], **Yanlong Dong**[1], **Mingxing Xu**[1]
[1]Department of Computer Science&Technology, Tsinghua University
[2]School of Electronics and Computer Engineering (SECE), Peking University
(guoym18, dongyl17)@mails.tsinghua.edu.cn
xumx@tsinghua.edu.cn, jinfahuang@stu.pku.edu.cn

Abstract

In this paper, we describe our ensemble-based system designed by guoym Team for the SemEval-2020 Task 8, *Memotion Analysis*. In our system, we utilize five types of representation of data as input of base classifiers to extract information from different aspects. We train five base classifiers for each type of representation using five-fold cross-validation. Then the outputs of these base classifiers are combined through data-based ensemble method and feature-based ensemble method to make full use of all data and representations from different aspects. Our method achieves the performance within the top 2 ranks in the final leaderboard of *Memotion Analysis* among 36 Teams.

1 Introduction

Memotion analysis is a task to understand the emotions of memes (Sharma et al., 2020). A meme is an idea, behavior, or skill that can be transferred from one person to another by imitation: stories, fashions, inventions, recipes, songs, ways of plowing a field, or throwing a baseball or making a sculpture (Blackmore and Blackmore, 2000). With the popularization of the Internet and instant message applications, Internet memes on social media platforms such as Facebook, Instagram, and Twitter have become an effective way of communication. (Shifman, 2014) However, the latest Internet memes have facilitated the prevalence of hate speech in online social media which makes this study problem increasingly significant (Williams et al., 2016). Detecting an offensive meme is more complex than detecting an offensive text because it involves visual and language understanding. So it is necessary to find a hybrid approach for the computational processing of Internet memes.

Similar to the text content on social media, memes need to be analyzed and processed to extract the information and the emotion conveyed. (Peirson et al., 2018; Oliveira et al., 2016) have tried to automate the meme generation process. (French, 2017) have tried to extract its inherent sentiment in the recent past. On this basis, more studies need to be done to distinguish fine-grained emotions in memes.

In this paper, we describe our approach to SemEval-2020 task 8, *Memotion Analysis*, which aims to identify the emotions of memes. There are three subtasks: task A aims for sentiment classification, task B aims for humor classification and task C is to quantify the scales of semantic classes. The system performance is evaluated by the macro F1 score for task A and the macro-averaged F1 score for task B and task C. According to our observation of the dataset and single classifier's performance, we design an ensemble-based classification system (Rokach, 2010) which is composed of multiple based classifiers. These base classifiers are combined using data-based ensemble method and feature-based ensemble method which are two steps of voting to get final predictions. Our ensemble-based method has achieved a macro F1 score of 35.2% for task A, a macro-averaged F1 score of 51.46% for task B, and a macro-averaged F1 score of 32.25% for task C which ranked 2nd, 2nd and 1st in three subtasks of *Memotion Analysis* among 36 Teams.

The rest of the paper is organized as follows. In Section 2, we describe our system including feature extraction, classifier, and ensemble method. In Section 3, detailed experiments, evaluation results of our

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 1120–1125
Barcelona, Spain (Online), December 12, 2020.

system, and the effect of data-based ensemble method and feature-based ensemble method are discussed. Section 4 gives the conclusion.

2 System Description

Our system consists of three parts: feature extraction, classifier and ensemble.

2.1 Feature Extraction

The images with size 224×224 and channel RGB are used as the visual input. We employ Resnet50 (He et al., 2016) network pre-trained on ImageNet 2012 classification challenge dataset (Deng et al., 2009) to extract visual features. The output of the final convolutional layer is used as region features and we use average pooling to get a 2048 dimensional representation for each image.

As for the textual context, we first use a 2-layer bi-directional Gated Recurrent Unit(Bi-GRU) (Cho et al., 2014) to accumulate contextual information from two directions for each sentence. The input of the Bi-GRU consists of 300-dimensional pre-trained GloVe vectors extracted from global word-word co-occurrence statistics (Pennington et al., 2014). We get the representation of a sentence by:

$$h_i^l = [\overrightarrow{h_i^l}; \overleftarrow{h_i^l}] \tag{1}$$

where l is the number of layers, i is the index of sentence, $[;]$ is the vector concatenation along the last dimension of features, and $\overrightarrow{h_i^l}$ and $\overleftarrow{h_i^l}$ are the last hidden states of forward and backward GRU of the lth layer, respectively.

Besides, we utilize Bidirectional Encoder Representations from Transformers(BERT) (Devlin et al., 2018) and Deep contextualized word representations(ELMo) (Peters et al., 2018)pre-trained on big datasets to get representations of sentences with a better sense of context. Furthermore, text features extracted by Bi-GRU and image features extracted by the Resnet50 network are concatenated to get the fusion features for memes.

2.2 Classifier

Every base classifier in our system shares the same architecture Figure 1(a). A dropout layer follows the first dense layer with a rectified linear unit(ReLU) (Nair and Hinton, 2010) as an activation function. The last dense layer with softmax as activation function is used to get the probability distribution over all predicted classes. The input of the base classifier is the representation extracted by the feature extraction module. The representation can be text features extracted by Bi-GRU, BERT, or ELMo, or image features extracted by Resnet50 network or fusion features of text and images. We use cross-entropy loss function in all base classifiers. When base classifiers are trained, the problem of data imbalance is considered. We give weight to the classes simply by multiplying the loss of each example by a certain factor depending on their class.

2.3 Ensemble

There are two methods of ensemble in our system.

Data-based ensemble method First, we use 5-fold cross-validation to estimate the generalization error of the chosen model configuration and choose the best hyper-parameters. In this procedure, five different models are trained on five different subsets of the training data. These five models then are saved and used as members of an ensemble. In our system, as shown in Figure 1(b), for each base classifier with a specific type of representation as input, five models are trained and combined together to get the probability distribution of each meme.

Feature-based ensemble method Besides, we get different models by feeding different types of data representation into models. There are five different models that take five different types of features as input, respectively. These features are text features extracted by Bi-GRU, BERT and ELMo, image features extracted by Resnet50 network, and fusion features of text and images. As shown in Figure 1(b), five models with different types of representation are fused together to get the final probability distribution of each meme.

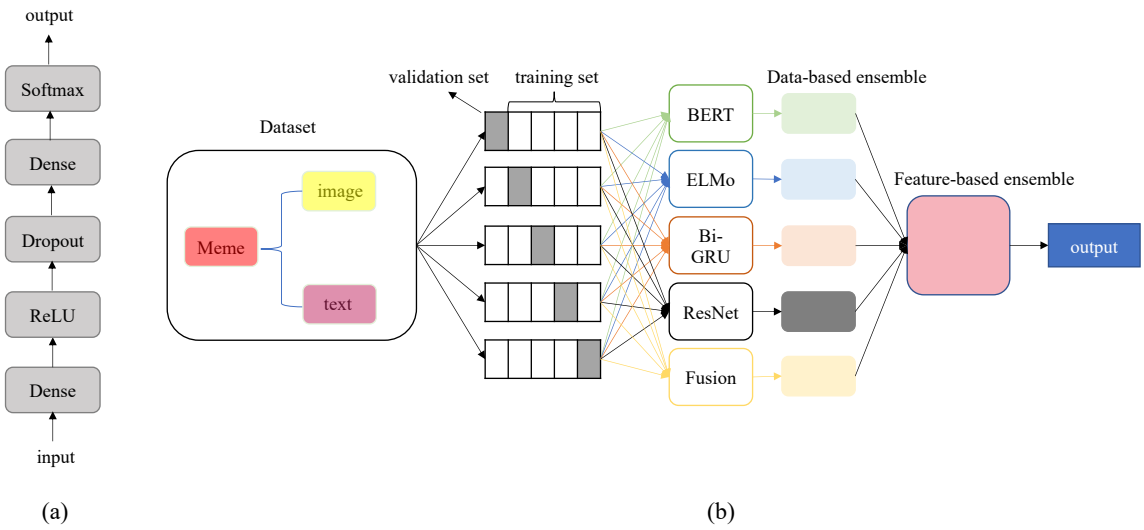


Figure 1: The architecture of base classifiers and overall system.

In our system, we utilize the soft voting method in the data-based ensemble stage and hard voting method in feature-based ensemble stage. It should be noted that due to models with different types of features as input achieve different performances in tasks, so we assign different weights $w \in \{0, 1\}$ to them according to their macro-F1 scores in validation sets.

Besides, our dataset is unbalanced especially for Task A. We find that the number of memes which is neutral or negative is less than the number of memes which is positive. We train two binary classifiers. The first one is to classify memes as positive instances or others which includes neutral and negative and the second one is to classify memes as negative instances or neutral instances. In the test stage, we first feed the meme to the first classifier. If it is predicted as a positive instance, we think it is a positive meme. On the other hand, if it is predicted as others, we feed it into the second classifier to get the final prediction.

We find that the output of task B can be used in Task C. We train four binary classifiers to identify the type of humor expressed by memes for Task B. For Task C, we train three three-category classifiers to quantity the extent with "slightly ", "mildly" and "very" to which a particular effect is expressed. If a meme is predicted to be not humorous in Task B, we classify it to be not humorous in Task C. On the other hand, If a meme is considered to express a certain typer of humor in task B, we feed it into the corresponding classifier in Task C to quantify the specific extent.

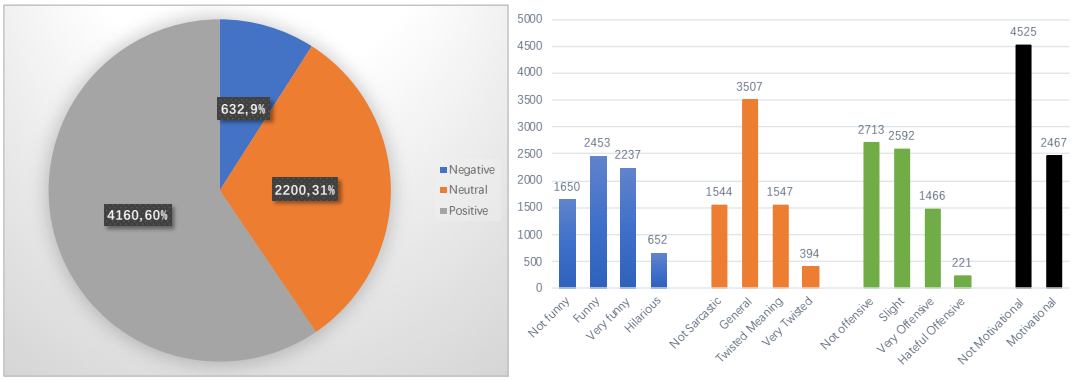
3 Experiment and Discussion

3.1 Dataset and metric

Dataset and Tasks Task organizers have released a new memotion dataset, where the training set contains 6992 memes and the test set contains 1878 memes. The task A is to classify each meme into a positive, negative or neutral meme. As for task B, we need to identify the type of humor, which can be "sarcastic", "humorous", "offensive" or "motivation". It should be noted that a meme can have more than one type. The task C is to quantify the extent with "not", "slightly", "mildly" and "very" to which a particular effect is being expressed. Figure 2 shows the class distributions of the dataset for Memotion Analysis.

We use 5-fold cross-validation which randomly divides the dataset into a training set and a validation set according to a four-to-one ratio. Our division ensures that the class distributions of training set and validation set are the same. Class distributions of training set and validation set for task A are shown in Table 1.

Metric We adapt macro F1 score as the evaluation metric for task A. For task B and task C, we adapt macro F1 score for each of the subtasks, and then average them to obtain macro-averaged F1.



Class distributions of dataset for task A

Class distributions of dataset for task B&C

Figure 2: Class distributions of dataset for Memotion Analysis

Dataset	Neutral	Positive	Negative	Total
Training set	1760	3328	506	5594
Validation set	440	832	126	1398

Table 1: Class distributions of training set and validation set for task A

3.2 Preprocessing

There are two modalities in memes, including images and text. As for images, we resize images with different sizes to 224×224 using bilinear interpolation. As for textual content, we notice that there are many tags, HTML entities, non-alphabets, and any other kind of characters that might not be a part of the language in sentences. Hence, we remove all tags, urls, and numbers in the text which seem to be not useful for memotion analysis. Then we transform all the letters in text into lowercase and sentences are padded to the length of the longest sentence in the dataset.

3.3 Implementation Details

Tensorflow (Abadi et al., 2016) is used to develop our models. We use BERT-base model in our experiments. For network optimization, we choose Adam algorithm (Kingma and Ba, 2014).The hyper-parameters of the classifiers are shown in Table 2. For different types of representation, the learning rates are different. For text features extracted by Bi-GRU, the learning rates of the classifiers are 0.0001, and for image features and text features extracted by BERT and ELMo, the learning rates of classifiers are 0.00001.

Parameters	batch size	hidden size of Bi-GRU	dropout rate	l2
Value	32	150	0.5	1e-4

Table 2: Parameters of the system

3.4 Results and Discussion

Quantitative Analysis. Table 3 shows the performance of baselines and our ensemble-based system in Memotion Analysis. Our system has achieved a macro F1 score of 35.2% for task A, a macro-averaged F1 score of 51.46% for task B, and a macro-averaged F1 score of 32.25% for task C which ranked 2nd, 2nd and 1st in three subtasks, respectively.

According to Table 3, we could discover that our ensemble system shows excellent results in task A, taskB and task C. It means that our system can extract fine-grained semantic information. Besides, improvement also implies that the data-based ensemble method and feature-based ensemble method are highly efficient to make the decision.

Task	A	B	C
Baseline	21.76	50.02	30.09
Ensemble-based system(ours)	**35.20**	**51.46**	**32.25**
Rank	2nd	2nd	1st

Table 3: The performance of baselines and our ensemble-based system.

Ablative Analysis. Furthermore, we do ablative studies to quantify the impact of the two ensemble methods in our system for task B. We compare the ensemble-based model against a set of ablated models with various settings.

Effect of the data-based ensemble method The data-based ensemble method combined five different models obtained from 5-fold cross validation by voting to ensure that the model fully leveraged all training data. As shown in Table 4, for any classifier with specific type of representation of data, data-based ensemble method has impressive effectiveness on the improvement of the model's performance.

Model	BERT	BERT +data	Resnet	Resnet +data	Fusion	Fusion +data	GRU	GRU +data	ELMo	ELMo +data
Hum.	50.86	51.56	52.62	53.06	52.38	53.18	52.57	53.65	50.74	51.03
Sar.	48.50	48.56	51.15	51.44	51.36	51.66	51.11	52.01	48.98	49.48
Off.	50.73	50.94	50.17	50.37	52.30	52.31	51.12	51.81	50.15	51.30
Mot.	50.26	50.34	51.66	52.15	51.93	52.08	51.31	51.37	49.73	50.12
Aver.	50.09	50.35	51.40	51.75	**51.99**	**52.31**	51.53	52.21	49.90	50.48

Table 4: Results of base classifiers and classifiers with data-based ensemble method for Task B.

Effect of the feature-based ensemble method Feature-based ensemble method combines models from data-based ensemble method by voting. As shown in Table 4 and Table 5, models using data-based and feature-based method achieve best performance in identifying the type of humor compared with other models. Models with different types of representation of data as input may be good in different way. They may make different prediction errors. We consider that combining the predictions from these models adds a bias that in turn counters the variance of a single trained model using a certain representation as input.

Model	Humorous	Sarcastic	offensive	motivation
data-based & feature-based method	53.98	52.61	52.34	52.20

Table 5: Results of classifiers with data-based and feature-based ensemble methods for Task B.

Other analysis From Table 4, the results show that the model with fusion features integrating the information of text and images achieves the best performance among five base classifiers. We consider that information of these two modalities complement each other so that the performance of classification has been greatly improved compared to other models using one modality.

4 Conclusion

In this paper, we present our ensemble-based system designed for SemEval-2020 Task 8, *Memotion Analysis*. In our system, we utilize five types of representation of data as input of base classifiers to extract information from different aspects. Data-based and feature-based ensemble methods are used to improve the performance of the system. Experiment results show that the ensemble-based system achieves good performance which ranked 2nd, 2nd and 1st in three subtasks of *Memotion Analysis*, respectively. In the future, it will be important to choose better representations used as input and design better architecture of hybrid networks for leveraging the data and features.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

Susan Blackmore and Susan J Blackmore. 2000. *The meme machine*, volume 25. Oxford Paperbacks.

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. *arXiv preprint arXiv:1406.1078*.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

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

Jean H French. 2017. Image-based memes as sentiment predictors. In *2017 International Conference on Information Society (i-Society)*, pages 80–85. IEEE.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814.

Hugo Gonçalo Oliveira, Diogo Costa, and Alexandre Miguel Pinto. 2016. One does not simply produce funny memes!–explorations on the automatic generation of internet humor. In *Proceedings of the Seventh International Conference on Computational Creativity (ICCC 2016). Paris, France*.

V Peirson, L Abel, and E Meltem Tolunay. 2018. Dank learning: Generating memes using deep neural networks. *arXiv preprint arXiv:1806.04510*.

Jeffrey Pennington, Richard Socher, and Christopher D 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.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Lior Rokach. 2010. Ensemble-based classifiers. *Artificial Intelligence Review*, 33(1-2):1–39.

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.

Limor Shifman. 2014. *Memes in digital culture*. MIT press.

Amanda Williams, Clio Oliver, Katherine Aumer, and Chanel Meyers. 2016. Racial microaggressions and perceptions of internet memes. *Computers in Human Behavior*, 63:424–432.