Unified Humor Detection Based on Sentence-pair Augmentation and Transfer Learning

Minghan Wang[1], **Hao Yang**[1], **Ying Qin**[1], **Shiliang Sun**[2], **Yao Deng**[1]
Huawei Translation Service Center, Beijing, China
East China Normal University, Shanghai, China
{wangminghan, yanghao30, qinying, dengyao3}@huawei.com
slsun@cs.ecnu.edu.cn

Abstract

We propose a unified multilingual model for humor detection which can be trained under a transfer learning framework. 1) The model is built based on pre-trained multilingual BERT, thereby is able to make predictions on Chinese, Russian and Spanish corpora. 2) We step out from single sentence classification and propose sequence-pair prediction which considers the inter-sentence relationship. 3) We propose the Sentence Discrepancy Prediction (SDP) loss, aiming to measure the semantic discrepancy of the sequence-pair, which often appears in the setup and punchline of a joke. Our method achieves two SoTA and a second-place on three humor detection corpora in three languages (Russian, Spanish and Chinese), and also improves F1-score by 4%-6%, which demonstrates its effectiveness in multilingual humor detection tasks.

1 Introduction

Machine learning has been adopted in computational linguistic for understanding natural languages for several decades. With the development of representation learning, rich semantics can be encoded into the dense vectors named as embedding, which significantly improves the ability of algorithms in understanding fine-grained emotions, for example, judging whether a sentence is humorous, often formulated as a binary classification problem. There can be many applications of humor detection such as language understanding in dialogue system and sentiment classification in social network platforms. In this paper, we focus on humor detection based on deep learning methods.

Figure 1: An example from HAHA corpus shows that the semantic discrepancy exists in a joke, where the **urinated stones** is a disease in the left picture and is an action in the right image, originated from the second and the third sentence in the joke:
"-Doctor, my kidney hurts a lot
*-Have you **urinated stones**?*
*-Yes doctor, I **urinated stones**, cars, trees, posts ..."*

Many algorithms has been used to solve these problems such as conventional machine learning algorithms like TF-IDF representation with SVM classifier, or deep learning based like BERT (Devlin et al., 2019). However, most of these algorithms are typically designed for universal tasks but ignoring the difference (e.g. the paragraph structure and semantic features) between humor detection and other document classification tasks.

From a linguistic perspective, there are two critical features that often appear in jokes, which inspire us to model them explicitly and make specific optimization for the task:

- Good setup and a punchline is the core of many jokes. The setup can be considered as the background of a story, and the punchline is the surprise or the exception that is commonly contradict to intuition, which is the trigger to make the reader laugh. The punchline often appears at the ends of the joke, should be short enough, and often has signif-

icant semantic discrepancy to the setup. The discrepancy could be a turning or a reinforcement. For example, *"One of the most wonderful things in life is to wake up and enjoy a cuddle with somebody; unless you are in prison."* or *"A wife is like a hand grenade. Take off the ring and say good bye to your house."*, another example is shown in Figure. 1. Therefore, we may try to decompose the joke to model the setup and the punchline separately.

- The topic of the joke determines whether it is funny for most of the people. Social events, politics and daily life are mostly used as materials to write a joke, which means there are usually commonsense in the joke and requires prior knowledge to understand the conflict in the punchline. Because jokes are often very short, where items, roles and activities must be widely understood by readers. Therefore, a pre-trained language model is fairly appropriate for this task as it could provide better language representation learned from large corpus.

By reviewing features of jokes, we can start our study by making two assumptions. 1) Most of jokes have punchline, and can be appropriately modeled. 2) Most of punchlines have semantic discrepancies with setup, and can be considered as a factor in the determination of humorous.

Therefore, we propose a method for humor detection which can be described as three stages. **1)** Data augmentation with paragraph decomposition. **2)** Fine-tuning BERT on the task specific labels with the help of Sentence Discrepancy Prediction (SDP). **3)** Making predictions based on decomposed paragraphs. The contribution of our work can be summarized as following:

- We propose a data augmentation method named paragraph decomposition which is specifically appropriate for humor detection tasks.
- We propose a method to explicitly detect the semantic discrepancy in sentence pairs, named SDP.
- The proposed method is evaluated on three languages, which demonstrate its effectiveness in multilingual scenarios.

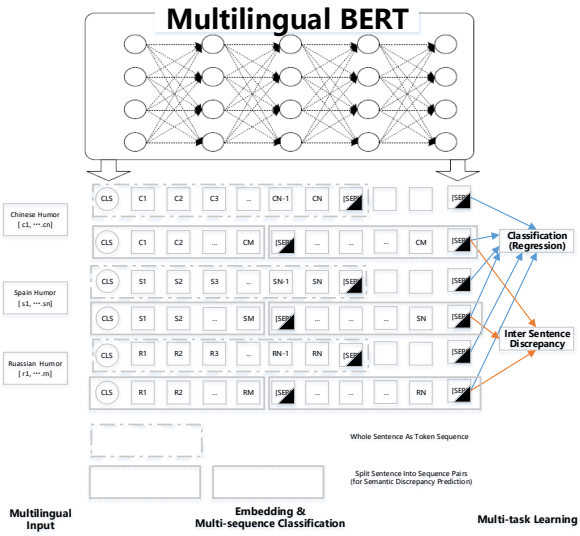

Figure 2: The architecture of our model, where two types of inputs are from three languages. The first type is normal sequence without being decomposed, and is only optimized by a classification/regression loss. The second type is decomposed sequence with additional inter sentence discrepancy loss as well as the classification/regression loss. All forms of inputs are encoded with a unified model based on the multilingual BERT.

2 Related Work

In recent years, many studies on humor detection have been published. Some researchers focuses on employing state-of-the-art studies like BERT (Devlin et al., 2019) to make better predictions, others attempts to improve simple networks like LSTM (Hochreiter and Schmidhuber, 1996) and CNN (Krizhevsky et al., 2012) or even conventional machine learning algorithms to compete with deep neural networks. At the same time, researchers have made available several high-quality datasets in different languages which significant help investigations on this area.

(Weller and Seppi, 2019) propose a BERT based humor detection model, fine-tuned on corpus collected from Reddit, Short Jokes and Pun of the Day (Yang et al., 2015), which achieves significant improvement on the performance comparing with many CNN based models.

(Chiruzzo et al., 2019) summaries a series of works from teams who build models and conduct experiments on HAHA dataset in the IberLEF 2019. (Ismailov, 2019) propose the method based on a pre-trained multilingual BERT, and further pre-train it on the domain dataset. Finally, the model is fine-tuned with task specific labels. Apart from that, they combine the prediction of Naive Bayes with TF-IDF and NN outputs with logistic

regression to produce the final prediction, which achieves the best result in the HAHA 2019 challenge. Other teams also follows the framework by combining deep pre-trained models with conventional algorithms to acquire competitive predictions.

(Blinov et al., 2019; Chiruzzo et al., 2019; Yang et al., 2015) release large corpus in different languages like Russian and Spanish, which give chances for researchers to build and evaluate their models on more diverse datasets. At the same time, they evaluate their datasets with proposed models and make detailed analysis which successfully demonstrates the good quality of the corpus.

By reviewing previous works and analyzing their results, we choose to follow a similar pipeline to start our work based on the pre-trained multilingual BERT and evaluate our method on three datasets in different languages aiming to investigate whether the feature of punchline exists in jokes from different cultures and can be detected with the model.

3 Approach

In this section, we introduce details of our method in the three stages which is shown in Figure 2, and we also discuss the advantages of our method comparing with others.

3.1 Paragraph Decomposition

We have briefly introduced the feature of a joke in the introduction section and pointed out the importance of the punchline. However, there is no publicly available large dataset with exact labeled location of the punchline sentence, which stops us from decomposing the joke into the setup and the punchline directly. Therefore, we apply two ways to decompose a joke into a sentence pair.

- **Decomposing from the middle**. The first method is the simplest way, which inserts a [SEP] token in the middle of the paragraph without considering real punctuations of the paragraph. We use $\mathbf{PD_M}$ to represent such method.
- **Decomposing from the last sentence**. The second way is to insert the [SEP] before the last sentence of the paragraph. We use $\mathbf{PD_L}$ to represent such method.

The major purpose of decomposing paragraphs into segment pairs is to convert the problem of a single document classification problem to paragraph pair classification. Two benefits can be achieved. 1) Tasks which heavily depend on understanding the semantic relationship between consecutive segments can be benefit from PD, such as natural language inference and humor detection. 2) From the experiment, we find that treating a long sequence (e.g. more than 300 tokens) as a single paragraph (without [SEP] in the middle) will dramatically drop the performance of BERT in a humor classification task; however, by adding [SEP] at the appropriate position, the performance can be optimized. We assume that in the pre-training of Next Sentence Prediction (NSP) in BERT, the [SEP] could affect the self-attention to attend tokens in the pre-/post-segment separately, which somewhat decreases the context length.

3.2 Sentence Discrepancy Prediction

As already stated, the punchline of a joke often has semantic discrepancy to the setup. Therefore, we explicitly model it by using original classification label as the SDP label, which means paragraphs labeled as humours (positive sample marked as 1) are considered to have a setup and a punchline with large semantic discrepancy. On the other hand, a negative sample (marked as -1) is considered to have no setup and punchline thus has no discrepancy between any sentences or sub-sentences inside the paragraph.

Specifically, we define $v_{i,\text{cls}}$ and $v_{i,\text{sep}}$ as the representation of the sentence pair from joke i, which can be obtained with the representation of [CLS] at the beginning and the [SEP] of the decomposed position, respectively.

Then, we choose to use the **cosine** as the scoring function to measure the semantic similarity of v_{cls} and v_{sep}. denoted as:

$$s_i = \cos(g(v_{i,\text{cls}}), g(v_{i,\text{sep}})), \tag{1}$$

where g is a linear transformation.

Finally, we define the SDP loss as L_{SDP}:

$$\mathcal{L}_{\text{SDP}} = \frac{1}{N}\sum_{i}^{N}(y_i + s_i)^2, \tag{2}$$

where $y \in \{-1, 1\}$ is the label comes from the binary classification task but scaled into -1 to 1. The purpose of this loss is to leverage the vector of two segments in the semantic space to the opposite direction if the paragraph is a joke (i.e. the paragraph

has a punchline thus the angle of the pre and the post segment should be large), and to the same direction (i.e. small angle for a non-humorous paragraph) if there is no discrepancy.

3.3 Fine-Tuning

Instead of simply fine-tuning the model with a single loss computed from the predicted logits and ground-truth, we fine-tune the model with two tasks sharing same labels but providing different contributions. The first loss comes from the conventional classification task, and the second one is from the sentence discrepancy prediction.

We define a task specific prediction heads implemented by a linear transformation, denoted as f; the input of the prediction head is the representation of the [CLS] token, represented as v_{cls}; $\hat{y}$ denotes the predicted logits. More formally:

$$\hat{y} = f(v_{\text{cls}}; \theta_f), \tag{3}$$

Weighted cross-entropy is used as the loss function to deal with the imbalance of the datasets; the label weights are calculated as follows:

$$w_c = \frac{N}{N_c \times C}, \tag{4}$$

where N is the number of samples in the training set; C is the number of classes (e.g. 2 for binary classification) and N_c is the number of samples classified as c. Therefore, the loss function can be rewritten as:

$$\mathcal{L}_{\text{CLS}} = -\frac{1}{N}\sum_{i}^{N}\sum_{c}^{C} w_c y_{i,c} \log P(y_{i,c}|x_i) \tag{5}$$

To train the model with two tasks, we define $\mathcal{L}(\theta)$ as:

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{CLS}} + \lambda \mathcal{L}_{\text{SDP}} \tag{6}$$

where λ is the factor to scale the SDP loss. Note that the parameters of BERT aren't frozen and can be updated during the fine-tuning.

3.4 Segment Ensemble

Although the paragraph decomposition could change the view of the model to encode the paragraph, it might also introduce noise and cause the damage on the semantic representation. Therefore, we use another BERT, fine-tuned on the **undecomposed** corpus to produce vanilla prediction, and ensemble it with the decomposed prediction. An average pooling is performed on the logits of two models. Note that the vanilla fine-tuned BERT is also considered as the **baseline** model; we use **SE** to represent segment ensemble for simplicity.

		Train	Dev	Test
FUN (RU)	samples	246,415	5,000	61,794
	tokens	17.69	17.59	18.17
	positive	50.00%	50.48%	50.0%
HAHA (ES)	samples	22,000	2,000	6,000
	tokens	15.48	15.56	16.35
	positive	38.59%	38.20%	39.03%
CCL (ZH)	samples	11,494	1,642	3,284
	tokens	38.33	39.15	38.71
	positive	70.34%	69.49%	70.34%

Table 1: Details about three datasets. ZH, RU and ES are the abbreviation of Chinese, Russian and Spanish respectively. Tokens are the average tokens per line in specific subset. Positives are the proportion of the positive samples in specific subset, which indicates that HAHA and CCL is relatively imbalanced comparing with FUN.

4 Experiments

In this section, we introduce the details of the datasets, as well as the experimental setup.

4.1 Data

We perform experiments on three following datasets organized in three languages respectively. The detail can be found in Table 1

4.1.1 CCL

This dataset is published in the CCL2019 Chinese Humor Detection Competition[1], which has two subsets where the first one is composed of 21,552 samples for binary classification. 21,885 jokes in the second subsets are labeled in three levels and can be formulated as a tri-class classification problem. However, we only perform experiments on the first subsets for compatibility with other two datasets. Note that the golden labels of development set and test set are not released, and can only be assessed by the competition organizer, therefore, we randomly split a dev and test set from the original train set for convenient. The experimental results reported later is from the test set on our own splitting, and we also present the score on the leaderboard of our model. Overlength jokes are removed from the training set and are trimmed to 512 tokens during validation. Macro F1-score is used as the evaluation metric.

[1] https://github.com/DUTIR-Emotion-Group/CCL2019-Chinese-Humor-Computation

Method	CCL	FUN	HAHA
Random (baseline)	0.5844	0.4991	0.4314
Fasttext (baseline)	0.8267	0.7982	0.7302
(2019)QingBoAI (ensemble)	0.9488	-	-
(2019)**ours (ensemble)**	0.8968	-	-
(2019)SanQunWuDui (ensemble)	0.8683	-	-
SVM	-	0.798	-
(2019)ULMFun	-	0.9070	-
(2019)adilism (ensemble)	-	-	0.821
(2019)Kevin & Hiromi (ensemble)	-	-	0.816
(2019)bfarzin (ensemble)	-	-	0.810
BERT (baseline)	0.8468	0.9022	0.7896
BERT-SDP ($\mathbf{PD_L}$)	0.8635	0.9115	0.7975
BERT-SDP ($\mathbf{PD_M}$)	0.8692	0.9126	0.8120
BERT-SDP ($\mathbf{PD_M}$+SE)	**0.9017**^{2nd}	**0.9138**^{1st}	**0.8217**^{1st}

Table 2: Our method achieves top 2 result in all three datasets comparing with both ensemble and single models published in 2019, which demonstrates the effectiveness of our approach in scenarios like multilingual and imbalanced data. Note that the second group are from the leaderboard of CCL competition which we participated in and achieved the second place. The third group is the result published in original FUN (Blinov et al., 2019). The fourth group is from the report of HAHA at IberLEF 2019 (Chiruzzo et al., 2019), which we didn't participate in and is shown for comparison purposes.

4.1.2 FUN

FUN is proposed in (Blinov et al., 2019), mainly collected from several Russian social network websites; it only contains binary labels (i.e. classifying whether a paragraph is humorous). Note that FUN is the largest dataset in our experiment, consisting of more than 313,210 samples, where 1877 are manually labeled and considered as golden truth which is not used for evaluation due to its limited size. 5000 samples are further split as a dev set from the train set. Macro F1-score is the evaluation metric.

4.1.3 HAHA

HAHA (Chiruzzo et al., 2019) is a Spanish corpus collected from twitter for the competition of IberLEF 2019. There are 30,000 samples where 11,595 tweets are labeled as humorous (38.7%). The humorous tweets are further annotated with real number scores in the range of 1 to 5. We only do the first task (i.e. binary classification) aiming to make comparable settings among three datasets with macro F1-score. In addition, we further split the train set into train and dev for tuning hyperparameters.

4.2 Experimental Setup

The BERT model we used is implemented with transformers (Wolf et al., 2019). All three datasets are encoded with BERT-base-multilingual-cased. We use pytorch[2] to implement the classification head f and the SDP head g after the BERT encoder. The model is trained on 4 Titan Xp GPUs where each has 12 GB memory, the batch size is set to 96. We use the AdamW (Loshchilov and Hutter, 2019) as the optimizer with the peak learning rate of 1e-4.

We perform experiment on the BERT baseline as well as 3 variants of our approaches, including two decomposition strategies and the segment ensemble. Besides the baseline BERT, all 3 variants use the SDP loss with $\lambda = 0.1$.

5 Analysis

The experimental results is shown in Table 2, which is separated into three groups. The first group contains baseline methods including a random predictor and a fasttext (Bojanowski et al., 2016) model. The second group are SOTA methods in CCL 2019 competition, where the second place is obtained by our ensemble model. The third group is published in original FUN (Blinov et al., 2019), where SVM is their baseline and ULMFun is a fine-tuned ULMFiT (Howard and Ruder, 2018). The fourth group are results published in the report of IberLEF 2019 (Chiruzzo et al., 2019), which we didn't participate in, and is shown for comparison purposes. The last group are the ab-

[2]https://pytorch.org/

ZH	猫似乎只是在削尖他们的爪子。 [SEP] 实际上， 他们正在锻炼腿部肌肉。	0.61
	猫似乎只是在削尖他们的爪子。 实际上， 他们正在锻炼腿部肌肉。[SEP]	0.55
EN	Cats seem to be just sharpening their claws. [SEP] In fact, they are exercising leg muscles.	0.61
	Cats seem to be just sharpening their claws. In fact, they are exercising leg muscles. [SEP]	0.55

Table 3: An example shows that correctly decomposing the joke could encourage the model to produce higher probability for the correct class.

lation study evaluated on a BERT baseline and 3 variants of our approach. Note that the score gap on the CCL column in the second and last group is caused by the different test set. We can see all of them have the improvements of performance comparing with baselines.

We find a representative case from CCL dataset, which is shown in Table.3. We can see that decomposing the joke from the start of the second sentence achieves higher probability and the second sentence is actually the punchline of this joke.

Although the score of HAHA is acceptable, we find some cases showing that the tweets published in HAHA is relatively unclean, with noisy characters like hashtags or being barely readable even by human, which also happens in FUN. As shown in Table. 4, repeatedly appeared "JA" and hashtags may corrupt the paragraph decomposition algorithm and produce unreasonable paragraph pairs. At the same time, BERT is not pretrained on tweets or corpus from social networks which means the token representations of FUN and HAHA is insufficient to encode correct semantics.

6 Conclusion

We propose the SDP and paragraph decomposition to for humor detection, by linking the classification label to the inter-sentence discrepancy prediction. Our proposed method achieves competitive performance on three dataset with different languages. Although our SDP algorithm has achieved great performance on humor detection tasks, how to generalize it to other NLP tasks remains as our future work.

ES	¿Tu? ¿Gustarme? JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JÁ Tengo que disimular un poco mas.
	#20CosasQueHacerAntesDeMorir: Enseñarles la diferencia entre: -Hay de haber -Ahí de lugar -Ay de exclamar - Ai se eu te pego.
	Rt con el pollo asado #PremiosFenix
EN	¿Your? ¿Like me? JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JA JÁ I have to hide a little more.
	#20Things to do before you die: Teach them the difference between: -There is a place -Ay to exclaim - I hit you there.
	Rt with roast chicken #PremiosFenix

Table 4: An example shows that uncleaned tweets from HAHA could dramatically corrupt the performance of paragraph decomposition and BERT encoder

References

Blinov, Vladislav, Valeria Bolotova-Baranova, and Pavel Braslavski. 2019. Large dataset and language model fun-tuning for humor recognition. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4027–4032.

Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *CoRR*, abs/1607.04606.

Chiruzzo, Luis, Santiago Castro, Mathías Etcheverry, Diego Garat, Juan José Prada, and Aiala Rosá. 2019. Overview of HAHA at iberlef 2019: Humor analysis based on human annotation. In *Proceedings of the Iberian Languages Evaluation Forum co-located with 35th Conference of the Spanish Society for Natural Language Processing, IberLEF@SEPLN 2019, Bilbao, Spain, September 24th, 2019.*, pages 132–144.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Hochreiter, Sepp and Jürgen Schmidhuber. 1996. LSTM can solve hard long time lag problems. In *Advances in Neural Information Processing Systems 9, NIPS, Denver, CO, USA*, pages 473–479.

Howard, Jeremy and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 328–339.

Ismailov, Adilzhan. 2019. Humor analysis based on human annotation challenge at iberlef 2019: First-place solution. In *Proceedings of the Iberian Languages Evaluation Forum co-located with 35th Conference of the Spanish Society for Natural Language Processing, IberLEF@SEPLN 2019, Bilbao, Spain, September 24th, 2019.*, pages 160–164.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States.*, pages 1106–1114.

Loshchilov, Ilya and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

Weller, Orion and Kevin D. Seppi. 2019. Humor detection: A transformer gets the last laugh. *CoRR*, abs/1909.00252.

Wolf, Thomas, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Yang, Diyi, Alon Lavie, Chris Dyer, and Eduard H. Hovy. 2015. Humor recognition and humor anchor extraction. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 2367–2376.