Self-Attention on Sentence Snippets Incongruity for Humor Assessment

Di Cao¹, *
¹ School of Computing and Information Systems, University of Melbourne
* dccao@student.unimelb.edu.au

ABSTRACT: Researches on humor identification can advocate a better understanding of human language. Many studies focused on the categorical classification problem of humor, which is less sensible to the intermediate level funny content. Previous work captured the incongruity between words but not sentence snippets. In this paper, a novel method is proposed to exploit snippet-level incongruity features from different aspects, combined with the sentence snippets representations to predict funniness scores. The experiment result shows that this model outperforms most of the competitive methods.

1. INTRODUCTION
With the spring-up of social media and social platforms, such as Twitter and Facebook, the sentiment detection on short text has aroused more attention. Humor, as an important and pervasive ingredient of human language, requires a deeper semantic understanding of text. Humor may vary from diverse cultures. In addition, cracking a joke, even an off-color one, employs sarcasm, timing, irony, and all elements strongly connected with social awareness and a rather wide knowledge base. Since humor is a highly creative activity, developing techniques, such as computational humor, enables computers to understand humor in human conversation or written text spur the progress in cognitive science.

Previous work in computational humor has mainly focused on the tasks of humor classification, especially in the study of binary classification tasks on annotated datasets [1-5]. However, from the perspective of computation, humor regression appears to be significantly subtler. Because such categorical prediction is actually unable to capture the intermediate level humor character, hence it is incapable of distinguishing the ambiguous sallies from trenchant ones. The Microcredit was released as a SemEval Task of 2020, a novel dataset for computational humor research [7]. Different from nearly all existing humor datasets annotated with categorical labels, this dataset assigned each sample with a score ranging from 0 to 3, inspiring researchers to focus on the humorous effect of atomic changes and the tipping point between regular and humorous text. In order to construct the possible latent structures behind humor, previous studies establish the humor-specific stylistic features and content-based manual extracted features [8, 9]. Buhscitu [10] team exploited such hand-crafted features together with word representations to estimate the funniness score. However, such features may be sensitive to the data, and they lack semantic information of context. To avoid intricate feature engineering work, Hitachi team exploited an ensemble of PLMs, BERT, GPT-2, RoBERTa, XLNet etc., fine-tuned and selected 20 best-performing instances per PLM [11]. The YUN-HPCC [12] system employed 11 different encoding methods to present the text. Though with the representation powers of the ensembled models and encoding methods, they occupied the front ranks in SemEval Task 7. However, the humor-specific features remain unemployed. To leverage both the semantic and humor-specific information, the ECNU
system adopted the incongruity theory of humor but focused on the word-level incongruity information. When it comes to the humorous sentences with incongruity, not between words but semantic snippets, their method may be inefficient and result in a low recall.

The intention of this paper is to assess the humor by capturing both the semantic representation and incongruity information of sentence snippets. A novel method is introduced, which applies the self-attention mechanism on sentence snippets to extract incongruity information. This model also combines the deep semantic vectors of original sentence snippets generated by pre-trained language models (PLMs), to estimate the mean funniness of the edited headlines in the Humicroedit dataset. The experiment result shows that the proposed method surpasses most of the competitive models with concise architecture. Ablation experiments and model analysis are also conducted to demonstrate the effectiveness of this model. In summary, the contributions of this work are as follows:

- A new deep method is proposed to exploit the snippet-level incongruity characteristic of funny edited headlines. Also, empirical evidence is brought in support of this method in application to general humor intensity assessment tasks, without building manual extracted linguistic features of humor.
- This proposed method achieves fairly equivalent performance to the ensemble methods using tens or hundreds of fine-tuned models, but have more compact structures and less training expenses. And it also exceeds other methods by injecting immediate information of funniness. Besides, the model visualization part presents that this model conforms to human intuition.

2. METHOD

In this section, humor rating problem is first defined, and then the proposed methods are described, which is Multi-Head Attention for Incongruous Snippets (MAIS) combined with pre-trained language models.

2.1 TASK DEFINITION

The objective of this task is to identify the humorous effect of micro-edit of a headline from professional news sources. Formally, for a given text with one token replaced by the substitute word, supposing the edited headline containing \( l \) words, \( X = \{x_1, x_2, x_3, ..., x_l\} \), where \( x_i \) represents the \( i^{th} \) word in the edited headline. The model is expected to estimate the mean funniness of the edited headlines on the range from 0 to 3, where higher grades indicate funnier variation from the original headlines.

2.2 MODEL OVERVIEW

Figure 1 gives an overview of this model, which is composed of three modules: sentence snippets embedding extraction module, semantic incongruity extraction module, and output module. The sentence snippets embedding extraction module serves as a sentence snippets encoder, which in this model, exploits BERT [14] and RoBERTa [15] to give sentence representation and then apply the convolution layer to transform the sentence embedding into token snippets representation. And the semantic incongruity extraction module applies the self-attention mechanism to capture the incongruity of the token snippets in a sentence, where the snippets representation is derived from the GloVe [16] word vectors. In the output module, snippets representation is concatenated from BERT and the snippets incongruity information based on GloVe to produce the estimated average funniness score of an edited headline.
2.3 SNIPPET EMBEDDING EXTRACTION MODULE

BERT is adopted as one way to acquire the representation of the edited headlines, considering the empirical improvements due to transfer learning with it have demonstrated its capability as a generalized word or sentence embedding tool. As such, the output $E \in \mathbb{R}^{l \times e}$ from BERT is a sequence of word embeddings, where $e$ represents the embedding dimension. Since in most cases, a single token will not address enough semantic information, but a sentence snippet which consists of several consecutive words. As the representation of sentence snippets need to be required, the convolutional layer provides a good choice for it can be applied to grasp contextual local features by implementing convolution operation between the convolution kernels and a sequence of word embeddings. Given a convolutional filter $f \in \mathbb{R}^{w \times e}$, where $w$ is the window size, the sentence snippets representation of the input sequence vector $E \in \mathbb{R}^{l \times e}$ now becomes $C \in \mathbb{R}^{(l-w+1) \times e}$, when the stride equals 1. And the $C$ is actually a set of vectors, which can also be represented as $C = \{c_1, c_2, ..., c_{l-w+1}\}$, where each element $c_i$ is calculated as the following formula:

$$c_i = \sum E_{i,e} \odot f_{w,e}$$

where $\odot$ means element-wise product. And $c_i \in \mathbb{R}^e$ represents the $i^{th}$ sentence snippet embedding.

2.4 SEMANTIC INCONGRUITY EXTRACTION MODULE

Since incongruity is widely accepted as one of the essences of humor [20], we intend to capture the incongruity caused by micro-edit on the headlines as additional injected information. Sentence incongruity, to a certain extent, can be thought as opposition or contradiction between the semantic chunks of the sentence. For example, the following sentence [21] presents an incongruous structure, resulting in a humorous effect.

*A clean desk is a sign of a cluttered desk drawer.*

In order to grasp the semantic distinction between the sentence snippets, the multi-head attention is applied to get the representation of snippets contradiction. Generally, multi-heads attention captures semantic differences from multiple aspects, which inject semantic distinction information into the
representation of sentence snippets. In this module, we first acquire the sentence level representation from GloVe [16], which is fixed during the training. All the out-of-vocabulary tokens are substituted with \='<unk>'\'. And to get a uniform sentence length, each edited headline is either truncated or padded with \='<pad>'\'. Both the embeddings of unknown word symbols and padded token symbols are initialized with random vectors. Then we adopt the convolutional layer to create the sentence snippets vectors and feed them to the transformer layer to get the semantic incongruity information of the edited headlines. The attention vector $a_i$ from one self-attention structure is computed as follows:

$$a_i = \sum_{j=1}^{t-m} \frac{Q_i K_j^T}{\sqrt{e_i}} V_j$$

where $e_i$ is the dimension of the $Q_i$, and $j$ is the number of sentence snippets. The $Q, K, V$ stands for three different types of encoded sentence snippets representation from GloVe. And they can be calculated as follows:

$$S_{Q,K,V} = W_{Q,K,V}^i \ast c^i + b_{Q,K,V}^i$$

where $c^i$ is the $i^{th}$ snippet representation, $W_{Q,K,V}^i$ is the corresponding projection matrix, and $b_{Q,K,V}^i$ is the bias term.

### 2.5 OUTPUT MODULE

In this stage, the first step is to concatenate feature vectors from both the sentence snippets extraction module and semantic incongruity extraction module. Then the joint representations are fed to the prediction layer, which consists of two linear layers, where the former linear layer serves as an encoder while the latter actually performs the prediction.

$$\text{Score} = \Phi(\text{Dropout}(\text{relu}(W_1 l + b_1)))$$

where $W_1$ is the weight matrix of the first linear layer and $b_1$ is the bias term. The input vector of this module is $l$, which concatenates the representations from the previous two modules. The function $\Phi$ is determined by the ultimate prediction task. In this regression task, it just performs linear combination on the encoded vectors from the first layer and produce the estimated funniness score.

### 3. EXPERIMENT RESULTS AND ANALYSIS

In this section, the experiment dataset, experiment settings and experiment results are introduced. Further, an ablation study is performed to demonstrate the effectiveness of the sentence snippets. And then, model analysis is applied to verify the influences of sentence snippet length and snippet incongruity embedding dimension. Also, the attention weights are visualized to illustrate the working principle of this model.

#### 3.1 EXPERIMENT ON HUMICROEDIT

##### 3.1.1 DATASETS

We employ the Humicroedit dataset [7] to estimate the funniness rating of an edited headline. This dataset contains about 5000 original headlines collected from Reddit via the popular subreddit world news and politics, where headlines sampled from 25 major English news sources, published between 01/2017 and 05/2018, and each of them is between 4-20 words long. Each original headline has three modified, potentially funny versions by replacing a verb/noun/entity with a single word. The funniness score of an edited headline is the average of the ratings from its five judges. As the examples listed in Table 1, the funniness score range from 0 to 3, where 0 means not funny, 1 means slightly funny, 2 means moderately funny, and 3 indicates funny.

| ID | Original Headline (Replaced word in bold) | Substitute | Rating |
|----|------------------------------------------|------------|--------|
| R1 | 4 arrested in Sydney raids to stop terrorist **attack** | kangaroo | 2.6 |
For the score prediction task, the Humicroedit dataset is randomly sampled into a train dataset with 9,653 samples, validation dataset with 2,420 samples, and test dataset with 3,025 samples. Only the training data and validation data are labeled with funniness scores, while the funniness scores for the test data are hidden, and it is expected to give the estimated approximate humor ratings. The test result of the best single model is reported, and for the model analysis, the performances on the validation set are reported for it is assumed that the distribution of funniness score is approximately the identical in validation dataset and test dataset.

3.1.2 EXPERIMENTAL SETTINGS

Our model runs on a NVIDIA Tesla T4 GPU. In the snippet embedding extraction module, the BERT tokenizer is employed, and all the tokens in the edited headlines are lowered. The fixed length of tokenized sentences is set to 40, which is a little larger than the maximum length to leave a space for future longer tokenized headlines. In the sentence snippets incongruity extraction module, all stopwords are removed to mitigate the effects of noise and set the fixed sentence length to 24, where shorter headlines are padded with ‘<pad>’. The sentence snippet length \( w \) is set to 4 for sentence representations from BERT and 3 for sentence representations from GloVe. The number of attention structures in one encoder layer is set to 5, and the number of encoder layers in the transformer architecture is also set to 5. The drop rate of the dropout layer in the prediction layer is 0.1. During the training, we choose the batch size of 64 and use the Adam optimizer [21] with learning rate \( lr = 2e^{-5} \) and \( \varepsilon = 10^{-8} \).

3.1.3 BASELINE MODELS

- **Fine-tuning PLM**: The model leverages the capabilities of pre-trained language models and utilizes the first token vector of the last layer as the sentence representation. And a prediction module takes the sentence-level representation to give the result. The parameters of the model will be adjusted to accommodate specific tasks during the training procedure. The results of ablation experiments given by Duluth system are referred [29].

- **The LMML and ECNU System**: These two systems utilized the importance of replaced words and replacement words in the task dataset against the contextual words to predict the funniness score [13, 23]. In ECNU system, they use BERT as the sentence encoder and feed the sentence representation to a bidirectional LSTM and an attention layer to extract the representation of both replaced words and replacement words to produce the prediction [13]. In LMML system, they utilized BERT to generate the contextual representations of both replaced words and replacement words and also combined them with the global embeddings of these two types of tokens produced by delivering the sentence representation to an attention layer to get the funniness rating [23].

- **The BERT-Flair System**: The model is proposed by the MLEngineer team [24]. They fine-tuned and exploited four BERT sentence regression models with a naïve bayes regressor to estimate the funniness rating, and combined it with the estimated rating from the flair library which utilized RoBERTa embeddings and a Naïve Bayes regressor to generate the ultimate rating, which is the average of two predictions.

- **The YUN-HPCC System**: This model employed an ensemble method, which used multiple preprocessing techniques (case vs. uncased, removing punctuation vs. keeping punctuation) and
encoded processed headlines with multiple encoders, including FastText, Word2Vec, ELMo, and BERT. The final representation incorporated 11 different encodings and was used for training a bidirectional GRU [12]. The predictions from bidirectional GRU were concatenated and fed to an XGBoost regressor.

- **The Hitachi System**: The SOTA model exploited an ensemble of PLMs BERT, GPT2, RoBERTa, XLNet, Transformer-XL, and XLM [11]. They fine-tuned 50 instances with a unique hyperparameter setting for each PLM. After applying 5-fold cross-validation, they selected the best 20 models per PLM to constitute the final predictor, which leveraged the prediction power of 700 models.

### 3.1.4 EXPERIMENTAL RESULTS

The proposed model is compared with the baseline models on root mean squared error (RMSE) between the annotated funniness ratings and the estimated ratings. Since the metric describes the distance from the estimated funniness score to the actual score, we expect it to be lower as possible. Table 2 shows the results of the models and other baseline models on the Humicroedit dataset.

The proposed model achieves second place among all baseline models on Humicroedit. Compared with the fine-tuning pre-trained language models, this model provides more accurate predictions on edited headlines. RoBERTa with MAIS decreases the RMSE score by 0.018 for each sample compared with fine-tuning RoBERTa. In contrast to LMML and ECNU systems, the global incongruity information of sentence snippets of edited headlines are implied instead of global attention information of replaced and replacement words. As the result shows, this proposed method outperforms these systems, owing to the involvement of modeling sentence snippets incongruity. Also, this model shows slightly better performances than the systems using the ensembled method. More specifically, this model contains more concise structure and demands less time during training process, but produces better prediction results than the BERT-Flair System, which applies four fine-tuning BERT regression models and a RoBERTa regression model with naïve bayes regressors to estimate the results. Moreover, our proposed method also surpasses the YUN-HPCC System, which employs 11 embedding methods to estimate the funniness of edited headlines.

From the comparison between the baseline models, we notice that the additional information is necessary for improving the performance of predicting the funniness level of edited headlines. The top-level systems either apply different encoding methods and incorporate the prediction power of multiple regression models or inject semantic information of replaced (replacement) words. Similarly, the proposed method also involves incongruity information, and it is proved to be effective. And self-attention on the sentence snippets rather than the tokens is performed in the LMML and ECNU system.

| Model                        | RMSE  |
|------------------------------|-------|
| Fine-Tuning BERT (Jin et al., 2020) | 0.536 |
| Fine-Tuning RoBERTa (Jin et al., 2020) | 0.534 |
| ECNU (Zhang et al., 2020)     | 0.522 |
| LMML (Ballapuram 2020)       | 0.520 |
| MLEngineer (Shatnawi et al., 2020) | 0.519 |
| YUN-HPCC System (Tomasulo et al., 2020) | 0.517 |
| Hitachi System (Morishita et al., 2020) | **0.497** |
| BERT+MAIS                    | 0.528 |
| RoBERTa+MAIS                 | **0.516** |

Table 2. Experiment results in Humicroedit dataset. The Top-2 results are in bold.
3.2 ABSTRACTION STUDY
In order to explore the effect of snippet-level semantic and incongruity information, a series of ablative experiment is conducted. In this proposed method, the underlying assumption is that snippet-level information performs better than token-level information. Since the convolutional layer serves as the sentence snippet extractor, the convolutional layers of snippets extraction module and semantic incongruity extraction module are firstly removed. In other words, the token embeddings are accepted rather than the snippets embeddings. The model is denoted as RoBERTa (w/o conv) + MAIS (w/o conv). Then we keep the convolutional layer of sentence snippet extraction module but eliminate the convolutional layer of semantic incongruity extraction module, which means we exploit the sentence snippets representation but only token-level semantic incongruity information. And the model denotes as RoBERTa + MAIS (w/o conv).

Table 3 gives the results of the ablative experiments. It shows that this proposed method RoBERTa + MAIS achieves the best result when involving sentence snippets embeddings and snippet-level incongruity information. The model RoBERTa (w/o conv) + MAIS (w/o conv) gives the worst result in the experiments, which demonstrates the significance of convolutional layer and proves that snippet-level representation and incongruity information outperform the token-level ones. Additionally, comparing with RoBERTa (w/o conv) + MAIS (w/o conv), the model RoBERTa + MAIS (w/o conv) shows a better performance, which illustrates sentence snippet embeddings provides a better choice than the word embeddings.

Table 3. Ablation experiment results. The best results are in bold.

| Model                                      | RMSE |
|--------------------------------------------|------|
| RoBERTa (w/o conv) + MAIS (w/o conv)       | 0.535|
| RoBERTa + MAIS (w/o conv)                  | 0.529|
| RoBERTa+MAIS                               | 0.516|

3.3 MODEL ANALYSIS
In this section, a comprehensive analysis of this model is conducted to measure the effect of sentence snippets length and the dimension of snippet representation in the semantic incongruity extraction module. Also, model visualization is performed on the self-attention layer of MAIS to prove its effectiveness.

- **The effect of sentence snippet length**
  In the proposed method, sentence snippets are applied in both the embedding extraction module and incongruity information extraction module. Thus, the length of sentence snippet is vital to model performance. A small length \( m \) indicates this model only focuses on short sentence snippets, which may lose some important information for incongruity identification, resulting in an unideal performance of this model. In contrast, a large sentence snippet length may bring some redundant information, which may also confuse our model. To verify the effect of snippet length, contrastive experiments are carried out. First, the snippet length is fixed to 3 in the incongruity extraction module, and the sentence snippet length is changed in the embedding extraction module from 1 to 6. As shown in the Figure 2(a), the RMSE score reaches a bottom point when \( m \) is equal to 4. And the performance turns to be worse when \( m \) continues to grow. So, we pick \( m \) to 4 in the snippet embedding extraction module. Then we employ the optimal snippet length of embedding extraction module and change the snippet length in the incongruity information extraction module from 1 to 6. From the Figure 2(b), it can be concluded that the optimal sentence snippet length is 3 in the incongruity information extraction module.
Figure 2. The performances with the snippet length from 1 to 6 in IEM (a) and EEM (b).

- **The effect of snippet incongruity embedding dimension**
  In the incongruity information extraction module, the GloVe is employed to produce the sentence snippet representations and further transform them into snippets incongruity information embeddings. Therefore, the dimension of GloVe representation acts as an important factor in deciding how much incongruity information will be injected into the prediction part of the model. To identify the optimal dimension of snippet incongruity information, we experiment on both BERT based and RoBERTa based MAIS methods using the dimension of 100, 200 and 300. Figure 3 shows the results of the comprehensive experiments. The dimension of 100 gives the lowest RMSE score on these two methods, which means 100-dimension snippet incongruity information has an adequate force to assist the prediction on the funniness level of edited headlines.

Figure 3. The performance of two methods with different snippet incongruity embedding dimension.

- **Model Visualization**
  In this section, the attention weights in the snippet incongruity extraction module are visualized to illustrate how this proposed method works. Several humorous examples are selected from the validation dataset and use the following three edited headlines to explain in detail.

- “Kushner to visit therapist following latest Trump tirades.”
- “Iowa monkey who tried to vote for Trump twice pleads guilty to election misconduct.”
- “Trump Calls For ‘Bill Of Love’ Allowing DACA Recipients To cuddle.”
The actual scores for these three examples are 2.8, 2.6, and 2.2. And the proposed model gives prediction deviations of 0.43 and 0.36 for the first two examples but 1.14 for the last example. Figure 4 visualizes how this self-attention module works. As shown in Figure 4, the snippets “latest Trump tirades” and “to election misconduct” from the first two examples have the highest attention weights, while the largest attention weight is assigned to the snippet “Trump calls for” in the third example. And in the first two examples, the snippets with the largest attention weights show incongruity with the snippets containing the edited word. However, the snippets that gain more attention in the third example show no explicit contradiction to the snippet with the edited word. Thus, this proposed model is powerful in capturing the incongruity information within a sentence, but has flaws in identifying the humorous sentence not caused by incongruity.

Figure 4. The visualization of snippet attention weights for given examples.

4. CONCLUSION
In this paper, a novel method is proposed by introducing a sentence snippet incongruity information extraction module combined with sentence snippets representations to predict the funniness score of an edited headline. It alleviates the computational complexity incurred by bonding multiple fine-tuning models and injects direct information of humor to help identify the funny headlines. The proposed model achieves a better performance than the competitive ensemble methods. Further, the ablation experiments have demonstrated the effectiveness of the model settings and the proposed method.

5. FUTURE WORK
The predicted variations mostly center on the edited headlines with large funniness scores in the minority of the training samples. For future content like this, more uniformly distributed labeled humor data are advocated. Another direction worth noticing is to better capture the significant features using a few samples. In this circumstance, few-shot learning is considered to be applied. Since the incentives of humor may vary in types, it is also believed that focusing on representations of the specific forms of humor, such as irony, puns, and superiority, would be advantageous. Additionally, to generalize this proposed method to accommodate more NLP tasks, multi-task learning is also taken into consideration to produce more general representations of both snippets and semantic information.

REFERENCES
[1] Mihalcea, Rada, and Carlo Strapparava. "Making computers laugh: Investigations in automatic humor recognition." In Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, pp. 531-538. Association for Computational Linguistics, 2005.
[2] Kiddon, Chloe, and Yuriy Brun. "That’s what she said: double entendre identification." In Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies, pp. 89-94. 2011.

[3] Bertero, Dario, and Pascale Fung. "Deep learning of audio and language features for humor prediction." In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16), pp. 496-501. 2016.

[4] Zhang, Renxian, and Naishi Liu. "Recognizing humor on twitter." In Proceedings of the 23rd ACM international conference on conference on information and knowledge management, pp. 889-898. 2014.

[5] Barbieri, Francesco, Horacio Saggon, and Francesco Ronzano. "Modelling sarcasm in twitter, a novel approach." In Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 50-58. 2014.

[6] Barbieri, Francesco, and Horacio Saggon. "Modelling irony in twitter." In Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics, pp. 56-64. 2014.

[7] Hossain, Nabil, John Krumm, and Michael Gamon. "'President Vows to Cut< Taxes> Hair': Dataset and Analysis of Creative Text Editing for Humorous Headlines." arXiv preprint arXiv:1906.00274 (2019).

[8] Mihalcea, Rada, and Carlo Strapparava. "Learning to laugh (automatically): Computational models for humor recognition." Computational Intelligence 22, no. 2 (2006): 126-142.

[9] Yang, Diyi, Alon Lavie, Chris Dyer, and Eduard Hovy. "Humor recognition and humor anchor extraction." In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 2367-2376. 2015.

[10] Jensen, Kristian Nørgaard, Nicolaj Filrup Rasmussen, Thai Wang, Marco Placenti, and Barbara Plank. "Buhscitu at SemEval-2020 task 7: Assessing humour in edited news headlines using hand-crafted features and online knowledge bases." In Proceedings of the Fourteenth Workshop on Semantic Evaluation, pp. 824-832. 2020.

[11] Morishita, Terufumi, Gaku Morio, Hiroaki Ozaki, and Toshinori Miyoshi. "Hitachi at SemEval-2020 task 7: Stacking at scale with heterogeneous language models for humor recognition." In Proceedings of the Fourteenth Workshop on Semantic Evaluation, pp. 791-803. 2020.

[12] Tomasulo, Joseph, Jin Wang, and Xuejie Zhang. "YNU-HPCC at SemEval-2020 task 7: Using an ensemble biGRU model to evaluate the humor of edited news titles." In Proceedings of the Fourteenth Workshop on Semantic Evaluation, pp. 871-875. 2020.

[13] Zhang, Tiantian, Zhixuan Chen, and Man Lan. "ECNU at SemEval-2020 task 7: Assessing humor in edited news headlines using biLSTM with attention." In Proceedings of the Fourteenth Workshop on Semantic Evaluation, pp. 995-1000. 2020.

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

[15] Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. "RoBERTa: A robustly optimized bert pretraining approach." arXiv preprint arXiv:1907.11692 (2019).

[16] Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global vectors for word representation." In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp. 1532-1543. 2014.

[17] Kim, Yoon. "Convolutional neural networks for sentence classification." arXiv preprint arXiv:1408.5882 (2014).

[18] Kolchinski, Y. Alex, and Christopher Potts. "Representing social media users for sarcasm detection." arXiv preprint arXiv:1808.08470 (2018).

[19] Pan, Hongliang, Zheng Lin, Peng Fu, and Weiping Wang. "Modeling the incongruity between Sentence Snippets for Sarcasm Detection."
[20] Lefcourt, Herbert M. *Humor: The psychology of living buoyantly*. Springer Science & Business Media, 2001.

[21] Mihalcea, Rada, and Carlo Strapparava. "Making computers laugh: Investigations in automatic humor recognition." In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 531-538. Association for Computational Linguistics, 2005.

[22] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980* (2014).

[23] Ballapuram, Pramodith. "LMML at SemEval-2020 task 7: Siamese transformers for rating humor in edited news headlines." In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pp. 1026-1032. 2020.

[24] Shatnawi, Fara, Malak Abdullah, and Mahmoud Hammad. "MLEngineer at SemEval-2020 task 7: BERT-flair based humor detection model (BFHumor)." In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pp. 1041-1048. 2020.

[25] Riloff, Ellen, Asgequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. "Sarcasm as contrast between a positive sentiment and negative situation." In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 704-714. 2013.

[26] Tay, Yi, Luu Anh Tuan, Siu Cheung Hui, and Jian Su. "Reasoning with sarcasm by reading in-between." *arXiv preprint arXiv:1805.02856* (2018).

[27] Xiong, Tao, Peiran Zhang, Hongbo Zhu, and Yihui Yang. "Sarcasm Detection with Self-matching Networks and Low-rank Bilinear Pooling." In *The World Wide Web Conference*, pp. 2115-2124. 2019.

[28] Hossain, Nabil, John Krumm, Michael Gamon, and Henry Kautz. "SemEval-2020 Task 7: Assessing humor in edited news headlines." *arXiv preprint arXiv:2008.00304* (2020).

[29] Jin, Shuning, Yue Yin, Xiane Tang, and Ted Pedersen. "Duluth at SemEval-2020 task 7: Using surprise as a key to unlock humorous headlines." In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pp. 986-994. 2020.