Journal of Software Engineering and Applications, 2019, 12, 447-459
https://www.scirp.org/journal/jsea
ISSN Online: 1945-3124
ISSN Print: 1945-3116

Towards Understanding Creative Language in Tweets

Linrui Zhang, Yisheng Zhou, Yang Yu, Dan Moldovan

Lymba Corporation, Richardson, TX, USA
Email: lzhang@lymba.com, yzhou@lymba.com, yyang@lymba.com, moldovan@lymba.com

How to cite this paper: Zhang, L.R., Zhou, Y.S., Yu, Y. and Moldovan, D. (2019) Towards Understanding Creative Language in Tweets. *Journal of Software Engineering and Applications*, **12**, 447-459.
https://doi.org/10.4236/jsea.2019.1211028

Received: October 7, 2019
Accepted: November 15, 2019
Published: November 18, 2019

Abstract

Extracting fine-grained information from social media is traditionally a challenging task, since the language used in social media messages is usually informal, with creative genre-specific terminology and expression. How to handle such a challenge so as to automatically understand the opinions that people are communicating has become a hot subject of research. In this paper, we aim to show that leveraging the pre-learned knowledge can help neural network models understand the creative language in Tweets. In order to address this idea, we present a transfer learning model based on BERT. We fine-turned the pre-trained BERT model and applied the customized model to two downstream tasks described in SemEval-2018: Irony Detection task and Emoji Prediction task of Tweets. Our model could achieve an F-score of 38.52 (ranked 1/49) in Emoji Prediction task and 67.52 (ranked 2/43) and 51.35 (ranked 1/31) in Irony Detection subtask A and subtask B. The experimental results validate the effectiveness of our idea.

Keywords

Natural Language Processing, Deep Learning, Transfer Learning

1. Introduction

The social media messages have been commonly used to share thoughts and opinions about the surrounding world and have become a new form of communication [1]. Unfortunately, understanding social media messages is not straightforward. The language used in these messages is very informal, with creative vocabularies and expressions, such as creative spellings, #hashtags and emojis, making it more challenging to understand than traditional text, such as newswire [2]. For example, the tweet "Monday mornings are my fave :) # not" is an irony with negative sentiment, but it may be considered as a positive one with traditional

sentiment analysis model [3]. With the resurgence of Deep Learning, the recent study of social media understanding mainly focuses on using neural network models. For instance, [4] proposed a connected LSTM model for Tweet Irony Detection. [5] predicted Emojis from Tweets using RNNs with attention. [6] constructed a model with various neural network models, including DeepMoji [7], Skip-Thought [8], etc. to infer the effectual state of a person from the tweets. Even though neural network models can offer reasonably efficient computation, as well as better modeling of sequence, they also suffer from several issues. One major problem is that the training process of these models is purely data-driven, *i.e.* the knowledge they gained is entirely from the corresponding training data. Such training mechanism may work well for traditional text genres with formal sentences; however, it usually achieves an unsatisfiable performance with informal text, such as social media data. In addition, preparing substantial high-quality training data set also requires a lot of manual effort.

Transfer Learning is a machine learning strategy that stores knowledge gained in solving some upstream tasks and then applies the stored knowledge to solving some new but related downstream tasks [9]. It can be used to solve the above-mentioned issues in neural network models since 1) the transfer learning brings extra knowledge to the target tasks, which may be used to handle the social media messages and 2) the extra knowledge gained from the upstream tasks do not rely on the training data of the target tasks, which can reduce the human effort in preparing the corresponding training set. To validate this idea, we proposed a transfer learning-based model and tested it with two Twitter messages understanding tasks provided by SemEval 2018. We aim to evaluate whether leveraging the pre-learned knowledge can help neural network models understand the creative language in Tweets.

In literature, there are various works focusing on social media Analysis. From SemEval-2013 [10] sentiment analysis in Twitter to SemEval-2018 [11] affect Twitter, social media analysis has been a continuous hot topic in SemEval competition. However, the approaches proposed in these competitions are either rule-based or feature-based (traditional machine learning-based) [12], which usually require considerable manual efforts to develop. Some recent models though leveraged some naïve deep learning technologies, still cannot reach a satisfactory performance [13]. In this paper, we proposed a transfer learning-based system, which requires very limited human effect, but can achieve a state-of-the-art performance. The primary contributions of our paper are as follows:

- We demonstrate the effectiveness of transfer learning model in understanding creative language in Tweets.
- Our model advances the state of the art models in Irony Detection task and Emoji Prediction task in Tweets described in Sem Eval 2018, exceeding the top performer at Emoji Prediction task by 2.53% in F-score and surpassing the 2-ranked and 1-ranked performer at Irony Detection subtask A and subtask B by 0.7% and 0.3% respectively.

- We perform an ablation study to analyze the factors that affect the performance of our transfer learning model and compare our model with other state-of-the-art learning models in literature on the given tasks.

2. Task Description

In this section, we discuss the two selected tasks and introduce the related works.

2.1. Task 1: Emoji Prediction

Emojis are graphic symbols that represent ideas or concepts used in electronic messages and web pages [14]. Currently, they are largely adopted by almost any social media service and instant messaging platforms. However, understanding the meanings of emojis is not straightforward, *i.e.* people sometimes have multiple interpretations of emojis beyond the designer's intent or the physical object they evoke [15]. For instance, 🙏 intends to mean pray, but it is mis-used as *high five* in many occasions. A misunderstanding of emojis can reverse the meaning of sentences and mislead people. Therefore, effectively predicting emojis is an important step towards understanding text content, especially for the emoji-enriched social media messages, e.g. Twitter. SemEval-2018 Task 2 [16] introduced an Emoji Predication Task. Given a text message including an emoji, the goal is to predict that emoji based exclusively on the textual content of that message. Specifically, the messages are selected from Twitter data and assume that only one emoji occurs inside each tweet. **Figure 1** illustrates an example of a tweet message with an emoji at the end.

2.2. Task 2: Irony Detection in Tweets

Irony Detection of Tweets is a challenging task. Previous works, either rule-based systems or supervised machine learning systems, mainly focused on exploiting lexical features from text [17] [18]. This makes the machine has difficulty in assessing the semantic meanings of ironic text and only interprets the text in its literal sense. For example, the sentiment of the tweet *Love these cold winter mornings* 🙂 *best feeling everrrrr*! has a high chance to be classified as positive by tradition irony detection systems, since a positive feeling can be inferred from the word *love*, *best feeling* and 🙂. However, for human readers, it is obvious that the author does not enjoy the cold winter at all.

SemEval-2018 Task 3 presented a task on Irony Detection in Tweets. Given a tweet, the task aims to determine whether the tweet is ironic (Subtask A) and which type of irony is expressed (Subtask B). Specifically, Subtask A is a binary classification task that requires the submitted systems to predict whether a given tweet is ironic or not. Examples (1) and (2) illustrate an ironic and non-ironic tweet.

National Siblings Day #WeAreFamily #HappyNationalSiblings Day #SisterLikeUs @ TimeSquare… 😜

Figure 1. Example of a tweet with an emoji at the end.

1) **Ironic**. *Yay for another work at* 4 *am* 😐

2) **Non-ironic**. *On my lunch break so sleepy* 😴

Subtask B describes a multiclass irony classification task to define whether a tweet contains a specific type of irony. Examples (3) to (6) provide an explanation of each irony class, associated with one example. For more details about how each kind of irony is defined, please refer to the original paper.

3) **Verbal irony by means of a polarity contrast**. This category applies to instances containing an evaluative expression whose polarity is inverted between the literal and the intended evaluation.

Example: *I love waking up at* 8 *am on a Saturday morning after going to bed at midnight.*

4) **Other verbal irony.** This category contains instances that show no polarity contrast between the literal and the intended evolution, but are nevertheless ironic.

Example: @someuser Yeah keeping cricket clean, that's what he wants #Sarcasm

5) **Situational irony.** This category contains instances that describe situations that fail to meet some expectations.

Example: *As if I need more water in my pasture.*

6) **Non-ironic. This category contains instances that are not ironic.**

Example: On my lunch break so sleepy 😴.

3. Model Description

Currently, almost all popular transfer learning-based NLP models follow a pre-training and fine-tuning strategy. The model is pre-trained unsupervisely over different upstream tasks and then is customized to solve related downstream tasks by fine-tuning the model parameters with the training data provided by the downstream tasks. Typical transfer-learning based NLP models include: GPT [19], BERT [20] or XLNet [21]. BERT is currently the most popular model in NLP community. It takes the advantage of bi-directional training and can outperform the performance of GPT and meanwhile the BERT based model needs less computational power than XLNet. In this case, we decided to build our system based on BERT.

We selected the pre-trained BERT model designed for sequence classification (a.k.a *Bert for Sequence Classification*) and fine-tuned the model parameters using the labeled data of the target tasks. The main structure of our model is illustrated in **Figure 2** and the detailed implementation is introduced in the following sections.

3.1. Input Preprocessing

Since there is a large variation of vocabulary used in Tweets, such as creative spellings, URLs, #hashtags, etc., we have to normalize the inputs before sending them to the system. We utilized the *ekphrasis* tool [22] as the Twitter processor.

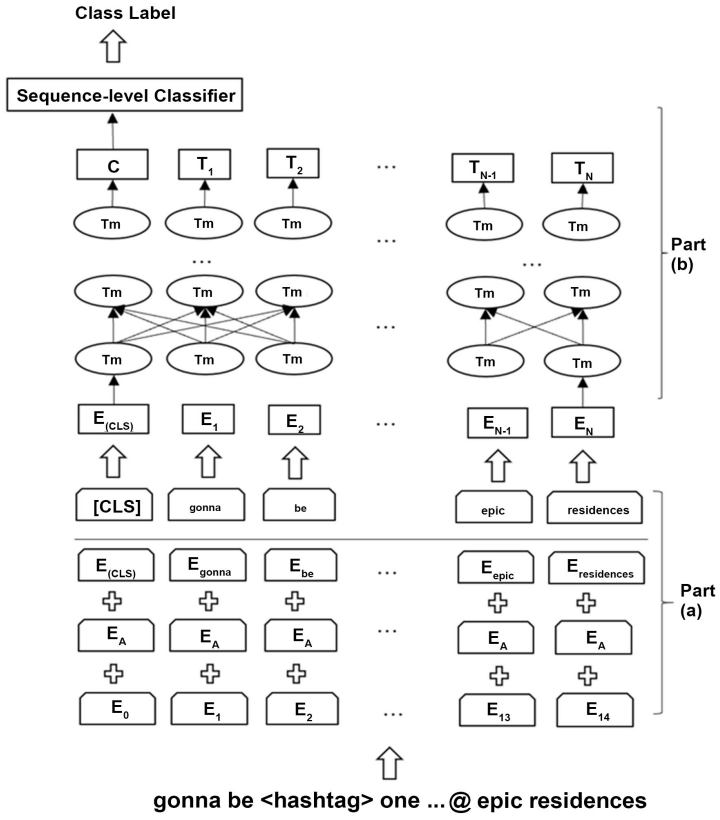

Figure 2. The main structure of our system.

It can perform tokenization, word normalization, word segmentation and spell correction for Twitter. There are several benefits of using the preprocessed tweets as inputs. For example, *ekphrasis* can recognize URLs (e.g. https://t.co/10PEnv2pz5) and substitute them with labels (<url>). This can significantly reduce the vocabulary size of the system without losing too much information. Table 1 gives an example of a tweet processed by *ekphrasis* tool.

3.2. Pre-Training Step

To make the paper self-contained, we first briefly introduce the pre-training step of BERT. More details about the model structure and the pre-training step could be found in the original paper.

BERT is pre-trained with two unsupervised upstream tasks: *Masked LM* and *Next Sentence Prediction* (*NSP*). In the *Masked LM* task, the system randomly masked out 15% of the tokens in each sequence and used the rest of the content to predict the masked tokens. This allows the model to learn a deep bidirectional representation of a sequence. In the *NSP* task, the system is trained to predicate if a sentence *B* is directly following another sentence *A* from a corpus. This allows the model to learn the relationship between the two sentences.

3.3. Fine-Tuning Step

The fine-tuning step aims to fine-tune the parameters of the per-trained model

Table 1. An example of a tweet processed by ekphrasis tool.

Original	Gonna be #oneepicsummer 3 Days|May 7th 2016 https://t.co/10PEnv2pz5 @Epic Residences
Processed	Gonna be <hashtag> one epic summer </hashtag> <number> <allcaps> days <url> @ epic residences

in order to customize the model for our new tasks. The details of the Fine-tuning step are discussed in this section.

3.3.1. Structure

The structure of our model is illustrated in **Figure 2**. It contains two components: (a) sequence representation and (b) sequence classification.

The aim of part (a) is to process the tweets into the format that can be accepted by BERT. The BERT input embeddings are the concatenation of three different kinds of embeddings: 1) Token Embeddings (e.g. E_{gonna}), 2) Segmentation Embeddings (e.g. E_A) and 3) Position Embeddings (e.g. E_1). The token embeddings are initialized with the pre-learned word embeddings during the pre-training step. The segmentation embeddings indicate whether the input sentence belongs to the first or the second sentence in the pre-training step. The position embeddings describe the positions of the tokens in the input sentences.

After the input sequences are formed into BERT input format, they are passed through the sequence classification component (part b). We specifically selected the pre-trained *Bert for Sequence Classification* Model as the sequence classifier.

Bert for Sequence Classification is a fine-tuning model that includes Bert Model and a sequence-level classifier on top of the Bert Model. The parameters of the Bert Model are initialized with the same parameters from the pre-training step and the parameters of the sequence-level classifier are waited to be trained in the fine-tuning step.

3.3.2. Training

During the fine-tuning step, the pre-processed Twitter messages are feed into the system to generate the probability of the candidate class labels. We used back propagation and Adam optimizer to train the system. Since both downstream tasks are classification problems, we select cross-entropy loss as the object function, which is calculated as follows:

$$Loss = -\sum_{i=1}^{n}\sum_{j=1}^{m} y_i^j \log p_i^j \quad (1)$$

where, y is a binary indicator (0 or 1) indicating whether a class label is correctly classified. P is the predicted probability of the correctly classified label. n is the number of training examples and $i \in [1, n]$ is the index number of the training examples. m is the total number of the class labels and $j \in [1, m]$ is the index number of the class labels.

3.3.3. Hyperparameter

The only new parameters introduced during fine-tuning step are the *number_of_labels* in the output layer. We set it to be the same as the number of class

labels of the corresponding tasks. The rest of the hyperparameters are set as default values in BERT. Table 2 shows the hyperparameters used for fine-tuning.

4. Experiments and Results

4.1. Emoji Prediction Task

4.1.1. Corpus

For the Emoji Prediction Task, we used the corpus provided by SemEval-2018. It collected roughly 550 K tweets (500 K for training and 50 K for testing) that include one of the twenty emojis that occur most frequently in the Twitter data. The relative frequency percentage of each emoji in the train and test set is shown in Table 3. From the table, we could observe that the corpus is not balanced. In order to handle this imbalanced issue, we selected macro-averaged F1-measure as the evaluation matrix of our system.

4.1.2. Experimental Results

Table 4 illustrates the performance of our model compared with the top performers in SemEval-2018 Task 2. The macro-averaged precision recall and F-score are presented.

We selected the top 1 performer Tubingen-Oslo [23], top 2 performer NTUA-SLP top 4 performer EmoNLP [24], top 6 performer UMDuluth-CS8761 [25] and the top 7 performer BASELINE system as the comparison system.

From the results, we can observe that our model can achieve state-of-the-art performance, exceeding the top performer Tubingen-Oslo (using SVM) by 2.53% in F-score, as well as the top neural network-based model NTUA-SLP (using RNNs) and the BASELINE model by 3.16% and 7.54%.

4.1.3. Effectiveness of Fine-Tuning Set Size

A key factor in the pretrain-finetune model is the size of the fine-tuning data. In this case, we present the F-score of our system against the training set size in Figure 3.

Table 2. Hyperparameters for fine-tuning.

Hyperparameter	Value
Max_sequ_length	128
Train_batch_size	32
Learning_rate	2e-5
Num_training_epochs	3
Number_of_labels (Emoji PredictionTask)	20
Number_of_labels (Irony Detection Task A)	2
Number_of_labels (Irony Detection Task B)	4
Pre-trained BERT model	Bert-base-uncased
Optimizer	BERT Adam
Lower case	True

Table 3. The distribution of the emoji labels.

#	Emoji	Train	Test	#	Emoji	Train	Test
1	❤️	22.4%	21.6%	11	📷	3.2%	2.9%
2	😍	10.3%	9.7%	12	🇺🇸	3.0%	3.9%
3	😂	10.2%	9.1%	13	🌞	2.9%	2.5%
4	💕	5.5%	5.2%	14	💜	2.6%	2.2%
5	🔥	4.9%	7.4%	15	😉	2.7%	2.6%
6	🙂	4.7%	3.2%	16	💯	2.7%	2.5%
7	😎	4.3%	4.0%	17	😁	2.6%	2.3%
8	✨	3.6%	5.5%	18	🎄	2.6%	3.1%
9	💙	3.4%	3.1%	19	📷	2.6%	4.8%
10	😘	3.2%	2.4%	20	😜	2.6%	2.0%

Table 4. Comparison of the participating systems with our system by precision, recall and F-score (in percentage) in the test set of SemEval-2018 task 2.

Team	Precision	Recall	F-score
Ours	**40.64%**	**41.76%**	**38.52%**
Tübingen-Oslo	36.55%	36.22%	35.99%
NTUA-SLP	34.53%	38.00%	35.36%
EmoNLP	39.43%	33.70%	33.67%
UMDuluth-CS8761	39.90%	31.37%	31.83%
Baseline	30.34%	33.00%	30.98%

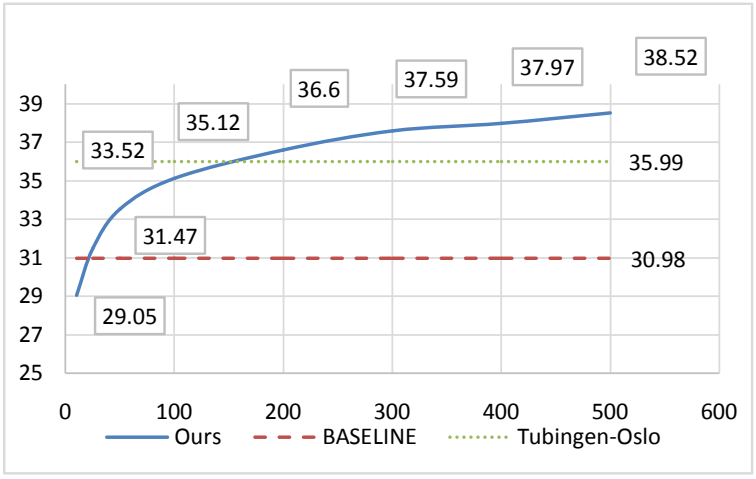

Figure 3. Learning curve of our model against the training set (×1000 instances). Horizontal axis indicate the size of the fine-tuning data and the vertial axis indicates the system performance in F_1 score.

From Figure 3, we can observe that the performance curve of our model is increasing with the increasing size of the fine-tuning set. Our model can surpass the BASELINE model and the top performer when the fine-tuning set size

reaches to around 20 K and 200 K. Intuitively speaking, this indicates that our model is gradually customized to adapt to the target task.

4.2. Irony Detection Task

4.2.1. Corpus

SemEval-2018 Task 3 presents the task on irony detection. It contains two subtasks: subtask A determines whether a tweet is ironic or not and subtask B determines the irony types of the tweet. For both tasks, a training corpus of 3834 tweets, as well as a test set of 784 tweets, is provided. Subtask A is a binary classification problem, so we used regular F-score that only reports results for the class specified by positive label. Subtask B is a multi-label classification problem and has the same corpus imbalance issue with emoji predication task, so we used macro-averaged F-score as the evaluation matric. The distribution of the different irony types of Subtask B experimental corpus is presented in **Table 5**.

4.2.2. Experimental Results

Table 6 demonstrates the experimental results of our model compared with other participants. The top three performers THU_NGN, NTUA-SLP [26] and WLV [27] on Subtask A and UCDCC [28], NTUA-SLP and THU_NGN on Subtask B are selected as the comparison models.

From the results, we can observe that our model can obtain competitive to state-of-the-art result on Subtask A and state-of-the-art results on Subtask B.

4.2.3. Effectiveness of Training Epochs

Unlike the previous emoji prediction task that contains as large as 500 K training samples, the iron detection task only contains roughly 4 K training samples. This implies that the learning model is more likely to overfit the data. In order to illustrate the influence of overfitting, we show the system performance against the training epochs in **Figure 4**. From the table, we can observe that the overfitting appears after three training epochs.

5. Results Analysis

From the experimental results, we can have the following observations:

- Our model can achieve state-of-the-art performance on Emoji Predication task and Irony Detection subtask B described in SemEval-2018. The reason is that the pre-learned knowledge from BERT provides a high starting point for the downstream tasks to begin with. By leveraging these pre-trained knowledges, the system can better understand the semantic meanings of the input data.

Table 5. Distribution of the different irony categories in the corpus.

Class labels	# of instances
Verbal irony by means of a polarity contrast	1728
Other types of verbal irony	267
Situational irony	401
Non-ironic	604

Table 6. Comparison of the participating systems with our system by precision, recall and F-score in the test set of SemEval-2018 task 3.

Task	Team	Precision	Recall	F-score
A	Our	0.604	0.765	0.675
	THU_NGN	0.630	**0.801**	**0.705**
	NTUA-SLP	**0.654**	0.691	0.672
	WLV	0.532	0.836	0.650
B	Our	0.529	**0.527**	**0.514**
	UCDCC	**0.577**	0.504	0.507
	NTUA-SLP	0.496	0.512	0.496
	THU_NGN	0.486	0.541	0.495

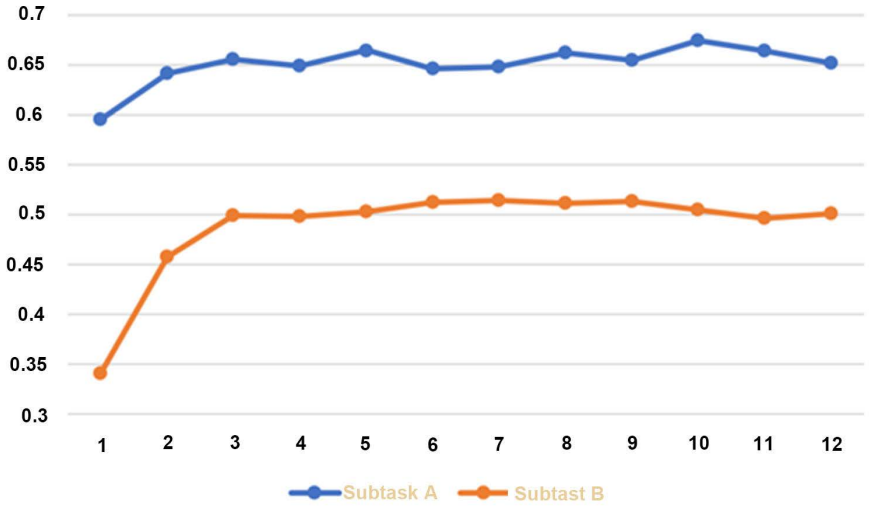

Figure 4. Performance curve of our model against the training epochs. Horizontal axis indicate the training epochs and the vertial axis indicates the system performance in F_1 score.

- The performance improvement is more obvious in Emoji Predication task than Irony Detection task. This indicates that fine-tuning set size is a very important factor in transfer learning. In Emoji Prediction task, we note that only after the system is fine-tune with 200 k samples, can it surpass the performance the state-of-the-art model. On the contrary, in the Irony detection task, we only have 4 k fine-tuning data. The system will overfit on the data before it can surpass the best performer in literature. From **Figure 4**, we can notice that the overfitting appears when the training epoch research to 3. As a result, our model can only rank 2nd place in Irony Detection subtask A and only beyond the best model by 0.7% on subtask B.
- Except from the pre-learned knowledge, the complex structure of BERT also contributes to the superior performance of the transfer learning model. Compared with the LSTM-based model, BERT contains 12 to 24 layers of transformer [29]. The multiple transformer layers enable BERT to learn a

more complicated representation of the input sentences. This makes it easier for the classification layer to classify the input instances in the high dimensional space.

- The complex structure of BERT also leads to several issues. The first issue is the overfitting problem we have discussed earlier that we need more training data to fine-tune the system. The second issue is the time complexity. According to our experiment, the speed of the fine-tuning procedure is 100 examples/second on one NVIDIA Titan RTX GPU, which is much slower than the LSTM-based models

6. Conclusions

In this paper, we implemented a transfer learning based system with BERT and applied it to two social media understanding tasks, the Emoji Predication task and the Irony Detection task. Experimental results have shown that leveraging the pre-learned knowledge can significantly increase the ability of neural network model in understanding the creative language used in social media messages. We also analyzed the features that have an effect on the transfer learning-based model and concluded that the quality of the model is highly dependent on the size of the fine-tuning set.

There are several avenues of future work. The primary work should be the optimization of the fine-tuning set. We will crawl more data, specifically focusing on social media genre, from online resources in order to improve the quality and gain the quantity of the fine-tuning data. Besides, we plan to integrate linguistic features into the system so as to leverage the lexical and syntactic information to improve the system performance. In addition, reducing the training and predicting time of the model so that our research can be more suitable for industrial applications, is another desirable improvement.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Rosenthal, S., Farra, N. and Nakov, P. (2017) SemEval-2017 Task 4: Sentiment Analysis in Twitter. *Proceedings of the* 11*th International Workshop on Semantic Evaluation* (*SemEval*-2017), Vancouver, Canada, August 2017, 502-518. https://doi.org/10.18653/v1/S17-2088

[2] Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F. and Stoyanov, V. (2016) SemEval-2016 Task 4: Sentiment Analysis in Twitter. *Proceedings of the* 10*th International Workshop on Semantic Evaluation* (*Semeval*-2016), San Diego, CA, June 2016, 1-18. https://doi.org/10.18653/v1/S16-1001

[3] Van Hee, C., Lefever, E. and Hoste, V. (2018) Semeval-2018 Task 3: Irony Detection in English Tweets. *Proceedings of The* 12*th International Workshop on Semantic Evaluation*, New Orleans, LA, June 2018, 39-50. https://doi.org/10.18653/v1/S18-1005

[4] Wu, C., Wu, F., Wu, S., Liu, J., Yuan, Z. and Huang, Y. (2018) THU_NGN at Semeval-2018 Task 3: Tweet Irony Detection with Densely Connected LSTM and Multi-Task Learning. *Proceedings of The* 12*th International Workshop on Semantic Evaluation*, New Orleans, LA, June 2018, 51-56. https://doi.org/10.18653/v1/S18-1006

[5] Baziotis, C., Nikolaos, A., Kolovou, A., Paraskevopoulos, G., Ellinas, N. and Potamianos, A. (2018) NTUA-SLP at SemEval-2018 Task 2: Predicting Emojis Using RNNs with Context-Aware Attention. *Proceedings of The* 12*th International Workshop on Semantic Evaluation*, New Orleans, LA, June 2018, 438-444. https://doi.org/10.18653/v1/s18-1069

[6] Paetzold, G. (2018) UTFPR at IEST 2018: Exploring Character-to-Word Composition for Emotion Analysis. *Proceedings of the* 9*th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, Brussels, Belgium, October 2018, 176-181. https://doi.org/10.18653/v1/W18-6224

[7] Felbo, B., Mislove, A., Søgaard, A., Rahwan, I. and Lehmann, S. (2017) Using Millions of Emoji Occurrences to Learn Any-Domain Representations for Detecting Sentiment, Emotion and Sarcasm. *Proceedings of the* 2017 *Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark, September 2017, 1615-1625. https://doi.org/10.18653/v1/D17-1169

[8] Kiros, R., Zhu, Y., Salakhutdinov, R.R., Zemel, R., Urtasun, R., Torralba, A. and Fidler, S. (2015) Skip-Thought Vectors. In: *Advances in Neural Information Processing Systems*, 3294-3302.

[9] West, J., Ventura, D. and Warnick, S. (2007) Spring Research Presentation: A Theoretical Foundation for Inductive Transfer. Brigham Young University, College of Physical and Mathematical Sciences, 32.

[10] Whlson, T., Kozareva, Z., Nakov, P., Rosenthal, S., Stoyanov, V. and Ritter, A. (2013) SemEval-2013 Task 2: Sentiment Analysis in Twitter. *Proceedings of the International Workshop on Semantic Evaluation*, Atlanta, GA.

[11] Mohammad, S., Bravo-Marquez, F., Salameh, M. and Kiritchenko, S. (2018) Semeval-2018 Task 1: Affect in Tweets. *Proceedings of the* 12*th International Workshop on Semantic Evaluation*, New Orleans, LA, June 2018, 1-17. https://doi.org/10.18653/v1/S18-1001

[12] Zhu, X., Kiritchenko, S. and Mohammand, S. (2014) Nrc-Canada-2014: Recent Improvements in the Sentiment Analysis of Tweets. *Proceedings of the* 8*th International Workshop on Semantic Evaluation*, Dublin, Ireland, August 2014, 443-447. https://doi.org/10.3115/v1/S14-2077

[13] Tang, D., Wei, F., Qin, B., Liu, T. and Zhou, M. (2014) Coooolll: A Deep Learning System for Twitter Sentiment Classification. *Proceedings of the* 8*th International Workshop on Semantic Evaluation*, Dublin, Ireland, August 2014, 208-212. https://doi.org/10.3115/v1/S14-2033

[14] Cappallo, S., Mensink, T. and Snoek, C.G. (2015) Image2emoji: Zero-Shot Emoji Prediction for Visual Media. *Proceedings of the* 23*rd ACM International Conference on Multimedia*, Brisbane, Australia, 26-30 October 2015, 1311-1314. https://doi.org/10.1145/2733373.2806335

[15] Barbieri, F., Ballesteros, M. and Saggion, H. (2017) Are Emojis Predictable? *Proceedings of the* 15*th Conference of the European Chapter of the Association for Computational Linguistics*, **2**, 105-111. https://doi.org/10.18653/v1/E17-2017

[16] Barbieri, F., Camacho-Collados, J., Ronzano, F., Anke, L.E., Ballesteros, M., Basile, V., Saggion, H., *et al.* (2018) SemEval 2018 Task 2: Multilingual Emoji Prediction. *Proceedings of The* 12*th International Workshop on Semantic Evaluation*, New Or-

leans, LA, June 2018, 24-33. https://doi.org/10.18653/v1/S18-1003

[17] Bouazizi, M. and Ohtsuki, T.O. (2016) A Pattern-Based Approach for Sarcasm Detection on Twitter. *IEEE Access*, **4**, 5477-5488. https://doi.org/10.1109/ACCESS.2016.2594194

[18] Van Hee, C., Lefever, E. and Hoste, V. (2016) Exploring the Realization of Irony in Twitter Data. *Proceedings of the Tenth International Conference on Language Resources and Evaluation* (*LREC* 2016), 1794-1799.

[19] Radford, A., Narasimhan, K., Salimans, T. and Sutskever, I. (2018) Improving Language Understanding with Unsupervised Learning. Technical Report, OpenAI.

[20] Devlin, J., Chang, M.W., Lee, K. and Toutanova, K. (2019) BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the* 2019 *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, **1**, 4171-4186.

[21] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. and Le, Q.V. (2019) XLNet: Generalized Autoregressive Pretraining for Language Understanding. arXiv preprint arXiv:1906.08237.

[22] Baziotis, C., Pelekis, N. and Doulkeridis, C. (2017) Datastories at Semeval-2017 Task 4: Deep LSTM with Attention for Message-Level and Topic-Based Sentiment Analysis. *Proceedings of the* 11*th International Workshop on Semantic Evaluation* (*SemEval*-2017), Vancouver, Canada, August 2017, 747-754. https://doi.org/10.18653/v1/S17-2126

[23] Çöltekin, Ç. and Rama, T. (2018) Tübingen-Oslo at SemEval-2018 Task 2: SVMs Perform Better than RNNs in Emoji Prediction. *Proceedings of The* 12*th International Workshop on Semantic Evaluation*, New Orleans, LA, June 2018, 34-38. https://doi.org/10.18653/v1/S18-1004

[24] Liu, M. (2018) EmoNLP at SemEval-2018 Task 2: English Emoji Prediction with Gradient Boosting Regression Tree Method and Bidirectional LSTM. *Proceedings of The* 12*th International Workshop on Semantic Evaluation*, New Orleans, LA, June 2018, 390-394. https://doi.org/10.18653/v1/S18-1059

[25] Beaulieu, J. and Owusu, D.A. (2018) UMDuluth-CS8761 at SemEval-2018 Task 2: Emojis: Too Many Choices? *Proceedings of The* 12*th International Workshop on Semantic Evaluation*, New Orleans, LA, June 2018, 400-404. https://doi.org/10.18653/v1/S18-1061

[26] Baziotis, C., Nikolaos, A., Papalampidi, P., Kolovou, A., Paraskevopoulos, G., Ellinas, N. and Potamianos, A. (2018) NTUA-SLP at SemEval-2018 Task 3: Tracking Ironic Tweets Using Ensembles of Word and Character Level Attentive RNNs. *Proceedings of The* 12*th International Workshop on Semantic Evaluation*, New Orleans, LA, June 2018, 613-621. https://doi.org/10.18653/v1/S18-1100

[27] Rohanian, O., Taslimipoor, S., Evans, R. and Mitkov, R. (2018) WLV at SemEval-2018 Task 3: Dissecting Tweets in Search of Irony. *Proceedings of The* 12*th International Workshop on Semantic Evaluation*, New Orleans, LA, June 2018, 553-559. https://doi.org/10.18653/v1/S18-1090

[28] Ghosh, A. and Veale, T. (2018) IronyMagnet at SemEval-2018 Task 3: A Siamese Network for Irony Detection in Social Media. *Proceedings of The* 12*th International Workshop on Semantic Evaluation*, New Orleans, LA, June 2018, 570-575. https://doi.org/10.18653/v1/S18-1093

[29] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Polosukhin, I., *et al.* (2017) Attention Is All You Need. 31*st Conference on Neural Information Processing Systems* (*NIPS* 2017), Long Beach, CA, 5998-6008.