Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 1511–1516
Marseille, 11–16 May 2020

Affect in Tweets: A Transfer Learning Approach

Linrui Zhang, Hsin-Lun Huang, Yang Yu, Dan Moldovan
Lymba Corporation
901 Waterfall Way Building 5, Richardson, TX 75080
{lzhang, hhuang, yyu, moldovan}@lymba.com

Abstract

People convey sentiments and emotions through language. To understand these affectual states is an essential step towards understanding natural language. In this paper, we propose a transfer-learning based approach to inferring the affectual state of a person from their tweets. As opposed to the traditional machine learning models which require considerable effort in designing task specific features, our model can be well adapted to the proposed tasks with a very limited amount of fine-tuning, which significantly reduces the manual effort in feature engineering. We aim to show that by leveraging the pre-learned knowledge, transfer learning models can achieve competitive results in the affectual content analysis of tweets, compared to the traditional models. As shown by the experiments on SemEval-2018 Task 1: Affect in Tweets, our model ranking 2^{nd}, 4^{th} and 6^{th} place in four of its subtasks proves the effectiveness of our idea.

Keywords: Natural Language Processing, Deep Learning, Transfer Learning

1. Introduction

In recent years, the interest in analyzing Twitter has grown exponentially among the NLP community and a substantial amount of related events and workshops (Rosenthal et al., 2015) (Nakov et al., 2016) (Rosenthal et al., 2017) (Barbieri et al., 2018) (Van Hee et al., 2018) have been organized. An essential part towards analyzing Twitter is to detect the emotions and the intensity of these emotions that are contained or can be inferred from tweets. For example, from the following two tweets, (1) *I'm in tears. This is so heartbreaking*, (2) *You don't know how to love me when you're in sober*. we should know that both tweets convey sadness, and the second tweet implies sadness to a lesser extent.

SemEval-2018 Task 1: Affect in Tweets (Mohammad et al., 2018) presents an array of subtasks where participating systems need to automatically determine the (intensity of) emotions and (intensity of) sentiments from the corpora provided by the organizers. Meanwhile, the organizers also summarized the methods and resources used by the participating teams. From their summarization, we observed that most of the participants chose to solve the problems with feature-based machine learning algorithms, which implement systems with a tremendous amount of linguistic features, pre-learned vectors and extra corpora (Park et al., 2018) (Baziotis et al., 2018) (Meisheri and Dey, 2018). For instance, the top one performer SeerNet (Duppada et al., 2018) implemented its system with pre-trained DeepMoji (Felbo et al., 2017) vectors, Skip-Thought vectors (Kiros et al., 2015) and Sentiment Neuron vectors (Radford et al., 2017), as well as a substantial amount of linguistic features, such as AFINN (Nielsen, 2011), NRC Affect Intensities (Mohammad, 2018) and Emotion Lexicon (Mohammad and Turney, 2010). Such a task specific architecture is not only difficult to be well adapted to different tasks or domains, but also needs considerable manual effort in feature engineering and system design.

Transfer learning is a machine learning strategy where a model trained on several related tasks is re-used as the starting point for a new task, so that the model can take advantage of the pre-learned knowledge from the previous tasks to make predictions for the new tasks. In recent years, transfer-learning has dominated a wide range of NLP tasks including Question Answer (Lan et al., 2019), Information Retrieval (Nogueira et al., 2019) and Text Understanding (Raffel et al., 2019). A lot of pre-learned structures have been proposed, such as ELMo (Peters et al., 2018), BERT (Devlin et al., 2019), and XLNet (Yang et al., 2019). In this paper, we designed and implemented a transfer learning system based on BERT and applied it to solving four subtasks related to the affect analysis of tweets from SemEval-2018 Task 1. Meanwhile, we compared its performance with that of other top performers that used traditional feature-based machine learning or deep learning methods. We aim to show that by leveraging the pre-learned knowledge and a very limited amount of fine-tuning effort, transfer learning models can achieve competitive to state-of-the-art results compared with the tradition models with excessive feature engineering. The primary contributions of our paper are as follows:

- We demonstrate the effectiveness of the transfer learning mechanism in the Twitter affectual content analysis task.
- We show that pre-trained representations can significantly reduce the need for much heavily engineered effort in designing task-specific architectures.
- Our model can achieve competitive results compared with the state-of-the-art systems in the SemEval-2018 Task 1, as illustrated by the experimental results in which our model ranks 2^{nd}, 4^{th} and 6^{th} place in four of its subtasks.

2. Task Description

SemEval-2018 Task 1 presents an array of tasks for the affectual content analysis of tweets. Specifically, it contains five subtasks: (1) Emotion Intensity regression task (**EI-reg**), (2) Emotion Intensity Ordinal classification task (**EI-oc**), (3) Sentiment Regression task (**V-reg**), (4) Sentiment

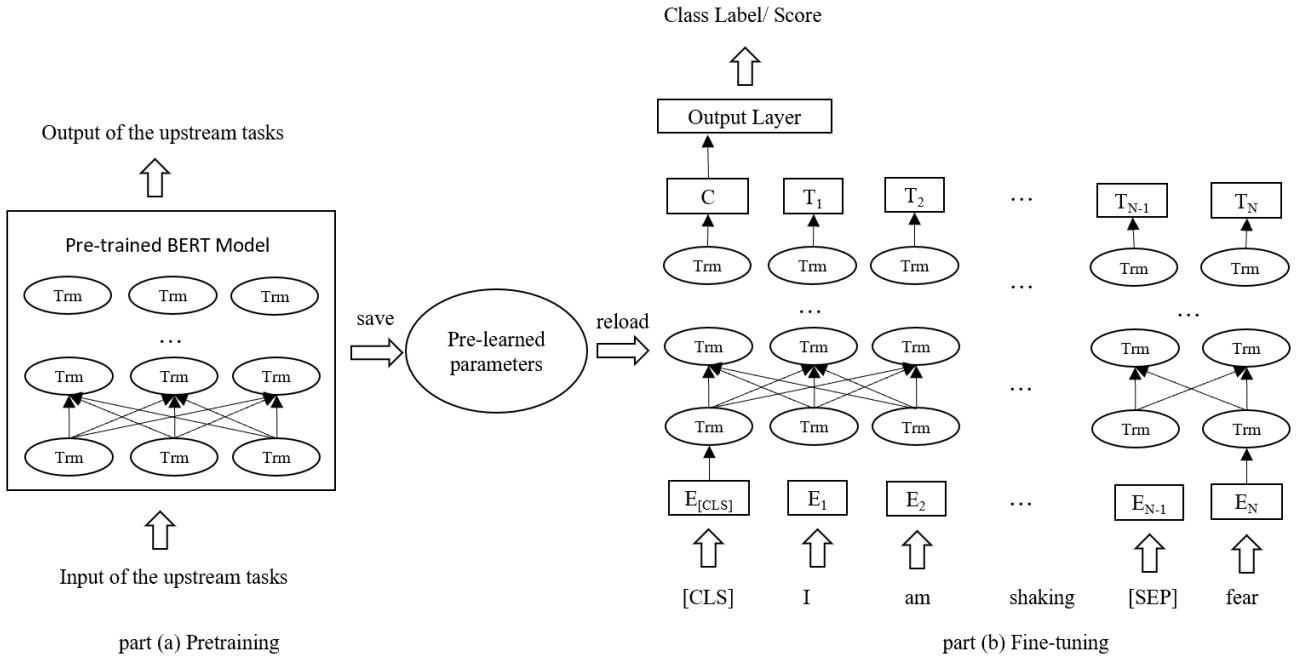

Figure 1: The main structure of our system.

Ordinal Classification task (**V-oc**), and (5) Emotion Classification (**E-c**) task. In this paper, we focus on four of the subtasks from (1) to (4) and the details of each subtask are sketched below:

2.1. EI-reg

Given a tweet and an emotion E (anger, fear, joy, sadness), determine the intensity of E that best represents the mental state of the tweeter, which is a real-valued score between 0 and 1. For example:

- Tweet: @hesham768 that's the spirit #optimism.
- Emotion: joy
- Score: 0.340

2.2. EI-oc

Given a tweet and an emotion E (anger, fear, joy, sadness), classify the tweet into one of four ordinal classes (0, 1, 2, 3) of intensity of E that best represents the mental state of the tweeter. For example:

- Tweet: I am shaking now.
- Emotion: fear
- Label: 3 (high amount of fear)

2.3. V-reg

Given a tweet, determine the intensity of sentiment or valence (V) that best represents the mental state of the tweeter, which is a real-valued score between 0 (most negative) and 1 (most positive). For example:

- Tweet: God, I've been so physically weak the whole day. So much shaking :(
- Score: 0.172

2.4. V-oc

Given a tweet, classify it into one of seven ordinal classes (from -3 to 3), corresponding to various levels of positive and negative sentiment intensity, that best represents the mental state of the tweeter. For example:

- Tweet: And here we go again
- Label: -2 (moderately negative emotion)

For more details about SemEval-2018 Task 1, please refer to the abovementioned paper (Saif Mohammad et al., 2018).

3. Approach

BERT (Devlin et al., 2019) is a novel language representation model, which pre-trains a deep representation of text from unlabeled data and is used as a prototype for many other state-of-the-art models for a wide range of NLP tasks. In this paper, we fine-tune this model by adding an additional output layer and apply the customized model to our tasks. Specifically, our approach consists of four steps: (1) pre-train BERT on several unsupervised upstream tasks and save the training parameters, (2) process the downstream tasks' data and transform them into BERT accessible format, (3) construct new structures for the downstream tasks and initialize the system with the pre-learned parameters, and (4) fine-tune the system structures with the preprocessed data of the downstream tasks. The training process and the main structure of our system is illustrated in Figure 1.

3.1. Pre-training

BERT is pre-trained with two unsupervised tasks: *Masked LM* and *Next Sentence Prediction (NSP)*. In the *Masked LM* task, the system will randomly mask out 15% of the tokens

in each sequence and use the rest of the context to predict the masked tokens. This allows the model to learn a deep bidirectional representation of the text. 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. After the pre-training step, the training parameters will be saved for future use.

3.2. Data Processing

We select a training example from the EI-oc task to illustrate the construction process of BERT's input representation. The input instance includes a tweet *I am shaking* and an emotion *fear*. We separate the tweet and emotion with a *[SEP]* token and add a start token *[CLS]* in front of the entire sequence.

Each token in the sequence is constructed by summing the corresponding token, segment, and position embeddings. The token embeddings (e.g. $E_{shaking}$) represent the meaning of the token and are initialized by (Wu et al., 2016). The segment embeddings indicate whether the given token belongs to sentence *A* or sentence *B* from the pre-training step. For example, as shown in Figure 2, token *shaking* appears before *[SEP]*, so it belongs to sentence A (E_A). The position embeddings illustrate the index of the token in the input sequence. A visualization of this construction can be seen in Figure 2.

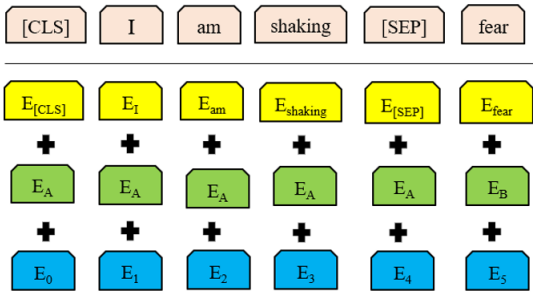

Figure 2: A visualization of the BERT input representation procedure.

3.3. System Architecture

As depicted in Figure 1 part (b), the system contains three components: (1) a data processing layer, (2) a sentence representation layer, and (3) an output layer. The data processing layer has been detailed in the previous section. Its output goes into the sentence representation layer. The sentence representation layer aims to generate the sequence representation of the input sequence. The output layer, built on top of the sentence representation layer, then accepts the sequence representation and generates class labels or regression scores, depending on the tasks.

The core part of the architecture is the sentence representation layer. It is constructed with 12 layers of transformers (same as the BERT base model) and the parameters of these transformers are initialized with the pre-learned parameters from the BERT pre-training step (part (a) in Figure 1).

3.4. Fine Tuning

The purpose of the Fine-Tuning step is to fine-tune the pre-learned parameters so as to customize the system to the downstream tasks. Our downstream tasks can be categorized into two types of problems: (1) EI-oc and V-oc as classification problems, and (2) EI-reg and V-reg as regression problems.

For the classification problem, we use cross-entropy loss as the object function to fine-tune the model, 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 predicted. P is the probability of the correctly predicted 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.

For the regression problems, we use mean-squared-error loss as the object function to fine-tune the model, which is calculated as follows:

$$Loss = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \quad (2)$$

where Y_i is the gold annotated score and $\hat{Y}_i$ is the system predicted score. $i \in [1, n]$ is the index number of the training examples.

After the model is fine-tuned with the training data provided by the downstream tasks, we will evaluate the model with the corresponding testing data.

Dataset	Train	Dev	Test	Total
EI-reg, EI-oc				
– anger	1701	388	1022	3091
– fear	2252	389	986	3627
– joy	1616	290	1105	3011
– sadness	1533	397	975	2905
V-reg, V-oc	1181	449	937	2567

Table 1: The statistic of the corpora

Parameters	Value
Max_Seq_length	128
Train_batch_size	32
Learning_rate	2e-5
Num_training_epoch	3
Number_of_labels (EI-oc)	4
Number_of_labels (V-oc)	7
Number_of_labels (EI-reg/V-reb)	1
Pre-trained BERT Model	Bert-base-uncased
Optimizer	BERT Adam

Table 2: The system's parameters

Task	Rank	Team Name	average	anger	fear	joy	sadness
EI-reg	-	Ours	78.5	80.0	78.1	78.3	74.2
	1	SeerNet	79.9	82.7	77.9	79.2	79.8
	3	NTUA-SLP	77.6	78.2	75.8	77.1	79.2
	23	Median Team	65.3	65.4	67.2	64.8	63.5
	37	SVM-Unigrams	52.0	52.6	52.5	57.5	45.3
	46	Baseline	-0.8	-1.8	2.4	-5.8	2.0
EI-oc	-	Ours	68.3	69.8	65.6	71.2	66.5
	1	SeerNet	69.5	70.6	63.7	72.0	71.7
	2	PlusEmo2Vec	65.9	70.4	52.8	72.0	68.3
	3	psyML	65.3	67.0	58.8	68.6	66.7
	17	Median Team	53.0	53.0	47.0	55.2	56.7
	26	SVM-Unigrams	39.4	38.2	35.5	46.9	37.0
	37	Baseline	-1.6	-6.2	4.7	1.4	-6.1

Table 3: The experimental results (in percentage) of our system and other SemEval-2018 participants on the EI-reg and EI-oc Tasks.

4. Experiments

4.1. Corpus

We evaluate our system with the data provided by SemEval-2018 Task 1. As demonstrated in section 2, it contains four corpora, EI-reg, EI-oc, V-reg and V-oc. For the EI-reg and EI-oc tasks, the corpora of the four emotions (anger, fear, joy and sadness) are provided separately. The statistics of the corpora of the four subtasks are shown in Table 1.

4.2. System Parameters

The only new parameter introduced during fine-tuning is *number_of_labels* in the output layer. For the classification tasks, we set it to be the number of candidate class labels. For the regression tasks, we set it to be 1 since the only output is a score number. The rest of the parameters are set as the default values in BERT. Table 2 illustrates the complete parameter set used in fine-tuning.

4.3. Experimental Results

This section shows the results of our experiments. We use the official evaluation method Pearson *r* as the evaluation metric and compare our system with the top performers in SemEval-2018.

4.3.1. Results of the EI-reg and EI-oc task

In Table 3, we show the details of the experimental results of our system and other representative systems SeerNet (Duppada et al., 2018), NTUA-SLP (Baziotis et al., 2018), PlusEmo2Vect (Park et al., 2018), Media Team, SVM-Unigrams and psyML (Gee and Wang, 2018) for the EI-reg and EI-oc tasks.

In the EI-reg task, we achieved 78.5% averaged Pearson *r* score (ranked 2/46), which is 1.4 percentage point (p.p) behind the top performer, SeerNet, and 0.9 p.p ahead of the 2^{nd} performer NTUA-SLP.

In the EI-oc task, we obtained 68.3% averaged Pearson *r* score (ranked 2/37), which is 1.2 p.p less than the top performer, SeerNet, and 2.4 p.p more than the 2^{nd} performer, PlusEmo2Vect.

4.3.2. Results of the V-reg and V-oc task

In Table 4, we show the details of the experimental results of our system and other representative teams, Median Team, SVM-Unigrams, TCS Research, Yuan, and Amobee (Rozental and Fleischer, 2018) for the V-reg and V-oc tasks. From the table, it can be observed that our system achieved 84.0% and 80.5% Person *r* score in the V-reg and V-oc tasks, respectively, ranking 6^{th} and 4^{th} places among the participating teams.

Task	Rank	Team Name	Pearson *r*
V-reg	-	Ours	84.0
	1	SeerNet	87.3
	2	TCS Research	86.1
	5	Amobee	84.3
	6	Yuan	83.6
	18	Median Team	78.4
	31	SVM-Unigrams	58.5
	35	Baseline	3.1
V-oc	-	Ours	80.5
	1	SeerNet	83.6
	2	PlusEmo2Vec	83.3
	3	Amobee	81.3
	4	psyML	80.2
	18	Median Team	68.2
	24	SVM-Unigrams	50.9
	36	Baseline	-1.0

Table 4: The experimental results (in percentage) of our system and other SemEval-2018 participants on the V-reg and V-oc Tasks.

5. Error Analysis

Even though our system can achieve a satisfactory performance, it still cannot surpass the top performer on the given tasks. We analyzed the errors and listed four reasons that should be responsible for the non-optimal performance, which are shown as follows:

- **Pretrain-finetune discrepancy.** The BERT model is pre-trained with data from Wikipedia and multiple corpora in the domain of books. Texts in these domains are usually written in standard English orthography. However, the downstream tasks involve twitter messages, which usually contain informal language, so the data in the pre-train and fine-tune steps are not consistent with each other. In this case, the prelearned knowledge from the upstream tasks cannot be well adapted to the downstream tasks. This makes our model suffer from the pretrain-finetune discrepancy issue and leads to a bad performance.

- **Data Genre.** As mentioned earlier, the language used in twitter messages is usually informal, with genre-specific terminology and abbreviations. Working with these informal text genres presents challenges for natural language processing beyond those typically encountered when working with traditional text genres.

- **Overfitting.** As shown in Table 1, only a very limited amount of data (roughly 2000 per task) is available for fine-tuning the model. In this case, the customized model will encounter the issue of overfitting before it reaches a satisfactory performance.

- **Linguistic Features.** In our approach, we do not leverage any linguistic features to develop our system. However, other top performers all rely on a variety of features and corpora, such as sentiment lexicons, word/character ngrams, dependency/parse features and extra unlabeled corpora. These features and corpora, despite the need of considerable manual effort to acquire them, can provide extra information in predicting system outputs and lead to a better system performance. Combining linguistic features to improve our system performance will be one of our future goals.

6. Conclusion

In this paper, we proposed a transfer learning model based on BERT and applied it to solving four subtasks related to the affect analysis of tweets provided by SemEval-2018. Our experimental results showed that, with the transfer learning mechanism, we can achieve a competitive to state-of-the-art performance by simply fine-tuning a pre-trained generic model instead of designing a task specific model that requires considerable manual effort. With this discovery, we can eliminate excessive feature engineering procedures in designing relevant machine learning models in related fields.

7. Bibliographical References

Barbieri, F., Camacho-Collados, J., Ronzano, F., Anke, L. E., Ballesteros, M., Basile, V., Patti, V., and Saggion, H. (2018). Semeval 2018 task 2: Multilingual emoji prediction. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 24–33.

Baziotis, C., Nikolaos, A., Chronopoulou, A., Kolovou, A., Paraskevopoulos, G., Ellinas, N., Narayanan, S., and Potamianos, A. (2018). Ntua-slp at semeval-2018 task 1: Predicting affective content in tweets with deep attentive rnns and transfer learning. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 245–255.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (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, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Duppada, V., Jain, R., and Hiray, S. (2018). Seernet at semeval-2018 task 1: Domain adaptation for affect in tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 18–23.

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. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1615–1625.

Gee, G. and Wang, E. (2018). Psyml at semeval-2018 task 1: Transfer learning for sentiment and emotion analysis. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 369–376.

Kiros, R., Zhu, Y., Salakhutdinov, R., Zemel, R. S., Torralba, A., Urtasun, R., and Fidler, S. (2015). Skip-thought vectors. In *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 2*, pages 3294–3302. MIT Press.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Meisheri, H. and Dey, L. (2018). Tcs research at semeval-2018 task 1: Learning robust representations using multi-attention architecture. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 291–299.

Mohammad, S. and Turney, P. (2010). Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 26–34.

Mohammad, S., Bravo-Marquez, F., Salameh, M., and Kiritchenko, S. (2018). Semeval-2018 task 1: Affect in tweets. In *Proceedings of the 12th international workshop on semantic evaluation*, pages 1–17.

Mohammad, S. (2018). Word affect intensities. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.

Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F., and Stoyanov, V. (2016). Semeval-2016 task 4: Sentiment analysis in twitter. In *Proceedings of the 10th international workshop on semantic evaluation (semeval-2016)*, pages 1–18.

Nielsen, F. Å. (2011). A new anew: Evaluation of a word list for sentiment analysis in microblogs. *arXiv preprint*

arXiv:1103.2903.

Nogueira, R., Yang, W., Cho, K., and Lin, J. (2019). Multi-stage document ranking with bert. *arXiv preprint arXiv:1910.14424*.

Park, J. H., Xu, P., and Fung, P. (2018). Plusemo2vec at semeval-2018 task 1: Exploiting emotion knowledge from emoji and# hashtags. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 264–272.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.

Radford, A., Jozefowicz, R., and Sutskever, I. (2017). Learning to generate reviews and discovering sentiment. *arXiv preprint arXiv:1704.01444*.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2019). Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Rosenthal, S., Nakov, P., Kiritchenko, S., Mohammad, S., Ritter, A., and Stoyanov, V. (2015). Semeval-2015 task 10: Sentiment analysis in twitter. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 451–463.

Rosenthal, S., Farra, N., and Nakov, P. (2017). Semeval-2017 task 4: Sentiment analysis in twitter. In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, pages 502–518.

Rozental, A. and Fleischer, D. (2018). Amobee at semeval-2018 task 1: Gru neural network with a cnn attention mechanism for sentiment classification. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 218–225.

Van Hee, C., Lefever, E., and Hoste, V. (2018). Semeval-2018 task 3: Irony detection in english tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 39–50.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

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*.