



Association Extraction and Recognition of Multiple Emotion Expressed in Social Texts

Jiajun Zou¹, Sixing Wu², Zhongliang Yang^{3(✉)}, Chong Chen⁴, Yizhao Sun⁵,
Minghu Jiang¹, and Yongfeng Huang³

¹ School of Humanities, Tsinghua University, Beijing 100084, China

² North China Electric Power University, Beijing 100096, China

³ Beijing National Research Center for Information Science and Technology,
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
yangz115@tsinghua.org.cn

⁴ North Automatic Control Technology Institute, Taiyuan 030006, China

⁵ Multimedia Communications and Signal Processing,
Friedrich-Alexander-Universität Erlangen-Nürnberg, 91058 Erlangen, Germany

Abstract. Detecting the sentiment people present in social media such as tweets is important for politics, commerce, education and so on. The task of multiple emotion recognition in texts is to predict a set of emotion labels that expressed in sentences. There are still some shortcomings in the current works: 1) the dependencies among emotions are not well modeled due to the complex combinatorial features of them, 2) the semantics of emotion labels as well as the semantic correlations between emotion labels and sentences are not fully considered. In this paper, in the purpose of capturing the dependencies between emotions, we propose a new method by using Graph Convolutional Network (GCN) based on a label co-occurrence matrix building from the dataset, and a Convolutional Neural Network (CNN) is used to capture the syntactic and semantic information in the sentences through different convolutional filters, the outputs of GCN and CNN are multiplied together to fuse their features as the last output. Experiments on SemEval2018 Task1: E-c multi-label emotion recognition problem show that metrics have been significantly improved, and our approach obviously obtains the dependencies among emotions described by Pointwise Mutual Information (PMI) which measures the correlations between emotions both in the true test labels and predicted labels.

Keywords: Multi-label emotion recognition · Graph convolutional network · Sentiment analysis · Deep learning

1 Introduction

Nowadays, with the development of Internet, people can express their thoughts, attitudes and emotions through social media such as Twitter, Facebook and

Weibo. Analyzing these subjective information is an important task in natural language processing (NLP) which has received a lot of attentions from many researchers recently. Sentiment analysis or opinion mining is the computational study of people’s opinions, sentiments, emotions, appraisals, and attitudes towards entities such as products, services, organizations, individuals, issues, events, topics, and their attributes [1, 5, 6, 13, 17, 22], the sentiment can be the polarity or an emotion state such as joy, anger or sadness [9]. The target of sentiment recognition is to recognize the categories of these emotions in sentences.

Judging from the number of emotions in a sentence, we can divide sentiment recognition into single-label recognition and multi-label recognition. Single-label recognition has been extensively studied such as [18, 19, 23]. However, although the results of single-label sentiment recognition have achieved a superior performance, it ignores the reality that a sentence itself may have multiple emotions at the same time, which result in losing some subjective information during processing these sentences. The task of multiple emotion or multi-label emotion recognition aims to predict a set of emotion labels that expressed in a sentence, which is different from single-label recognition. Due to the combinatorial nature of the emotional output space, that is, the emotions normally cooccur in a sentence with more complex dependencies which is more challenging than single-label task. How to dig out the dependencies between emotions efficiently is a key to solving the problem of multi-label sentiment recognition.

In order to overcome the challenges of multi-label emotion recognition in social texts such as Tweets, there have been several excellent works done by researchers. Most of current methods for this task convert the multi-label recognition problem into a set of binary or multi-class recognition problems, which is called problem transformation, to predict whether each label is or not a true label, and then the predictions are combined into multi-label predictions [9]. A bidirectional Long Short-Term Memory (LSTM) neural network with attention mechanism was used by Baziotis *et al.* [2] to deal with multi-label emotion recognition task of SemEval-2018 Task1, which won the competition. Their model utilized a set of word2vec word embeddings trained on a large collection of 550 million Twitter messages. Mohammed Jabreel *et al.* [9] proposed a novel method to transform the problem into a binary recognition problem and exploited a deep learning approach to solve the transformed problem, achieving a new accuracy score on the same dataset. Hardik Meisheri [15] combined three different features generated from deep learning models—a word-level bidirectional LSTM with attention as well as a traditional method in support vector machines. Ji Ho Park *et al.* [20] transferred the emotional knowledge by exploiting neural network models as feature extractors, they used these representations for traditional machine learning models such as support vector regression and logistic regression to capture the correlations of emotion labels, and it treated the multi-label problem as a sequence of binary recognition problems, thus the current classifier could use previous classifier’s output, namely classifier chain [21].

All of the previous works tried to use transformation method to deal with this task and played an important role in multi-label emotion recognition, however, there are still some shortcomings. On the one hand, due to the limitation

of predicted emotion label combination, the correlations existing in emotions could not be well modeled or they even lose the dependencies between emotion labels, such as binary relevance [21]. On the other hand, the semantics of emotion labels or even the semantic correlations between the emotion labels and the texts are not considered, and we think it can provide more additional information for multi-label emotion recognition. For example, in sentence “Oh, hidden revenge and anger...I remember the time”, the emotion labels for it are “anger” and “disgust”, we can obviously find that the labels and the sentence are semantically related, and even label-“anger” appears in it. Therefore, we argue that considering the above two aspects is crucial to achieve the goal of multi-label emotion recognition which motivates us to design a new method to overcome the weakness.

In this paper, we propose a novel Graph Convolutional Network (GCN) [11] based model to capture the label correlations for multi-label emotion recognition with a emotion label co-occurrence matrix. We use a Convolutional Neural Network (CNN) with different convolutional filters to further obtain the syntactic and semantic features in sentences. In order to take advantage of semantic correlations between sentence and labels, we also take the labels’ semantics into account which are used as the representations of nodes in the input of GCN. Experiments conducted on SemEval-2018 task 1 dataset show that our approach can improve multi-label emotion recognition metrics, and the dependencies between labels are captured, which are observed through visualization analysis.

This paper is organized as follows: In Sect. 2, we explain the methodology. The experiments and results are reported in Sect. 2. At last, conclusions are presented in Sect. 4.

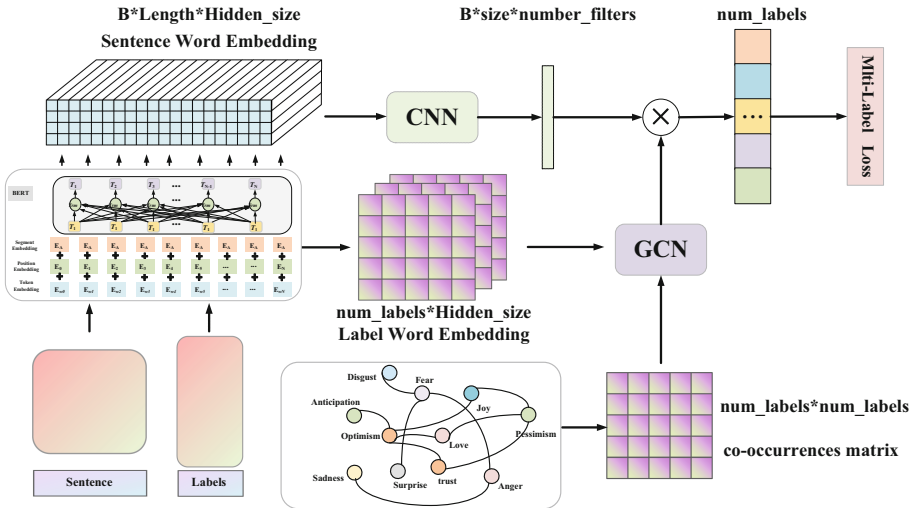


Fig. 1. The proposed multi-label emotion recognition model based on GCN.

2 The Proposed Method

In this Section we will first introduce the overall architecture of the proposed model, then the details of the CNN module and the GCN module which compose the model will be described. At last, we will introduce the output layer.

2.1 Overall Architecture

Assume the input sentence with n words is represented as:

$$S = \{x_1, x_2, \dots, x_n\}, \quad (1)$$

x_i is the i -th word in the sentence. The label set is formed as:

$$G = \{g_1, g_2, \dots, g_N\}, \quad (2)$$

where N is the number of emotion labels. The goal of this task is to predict a label subset belongs to G according to the input sentence S .

Figure 1 illustrates the overall architecture of the proposed model for the task. In the sake of extracting plentiful features from the input sentence, we adopt Bidirectional Encoder Representation from Transformers (BERT) [4], an excellent pre-trained language model, as the word embedding layer to calculate the embeddings of words in sentences and emotion labels. Then a CNN module is used to capture the local information through several convolutional filters attempting to take full advantage of the syntactic features. According to the dataset, we calculate the co-occurrence matrix of the emotion labels, which is normalized and put into the GCN with emotion label word embeddings, acting as the edge matrix and node matrix, respectively. Then we make the matrices from CNN and GCN multiply each other as the last output, thus fusing the correlations of different emotion labels and the features of sentences.

2.2 CNN Structure

We design a CNN architecture in the model. The input of CNN can be denoted as $H \in \mathbb{R}^{B \times L \times d_B}$, where B is the batch size, L is the max length of sentence we pad, and d_B is the hidden size of BERT.

The convolution operations are applied on these vectors to produce new feature maps. In the proposed model, the convolution operation involves several filters so as to capture different local features, because the emotions are often expressed by a sequence of words in different parts in sentences. We concatenate the results after the convolution. At the first filter, a two-dimensional convolution is used and we employ global max-pooling to obtain the feature r_1 :

$$r_1 = \mathbf{F}(H; \theta) \in \mathbb{R}^{B \times d_m}, \quad (3)$$

where d_m is the number of out channels in CNN, θ indicates model parameters. This operation is the same as the other filters. Thus we can get the last output of the CNN as follows:

$$r = [r_1; r_2; \dots; r_n] \in \mathbb{R}^{B \times (d_m \times n)}, \quad (4)$$

where n represents the number of filters.

2.3 GCN Module

GCN [11] is designed to deal with data containing graph structure, which is constructed by nodes and edges. In each GCN layer, a node iteratively aggregates the information from its one-hop neighbors and update its representation [7, 12, 25, 26].

In this task, we first count all labels in the dataset to construct the label co-occurrence matrix $A \in \mathbb{R}^{k \times k}$ through automatic method. Then we use a GCN to extract the co-occurrence features of emotion labels. For the first layer of GCN, we take the labels' word embeddings $E \in \mathbb{R}^{k \times d_B}$ and normalized label co-occurrence matrix A as the input, which denotes the nodes and edges, respectively, where k is the number of labels. The node features updates as follows:

$$h_i^l = \sigma\left(\sum_{j=1}^k A_{ij} W^l h_j^{l-1} + b^l\right), \quad (5)$$

where l is the l -th layer of GCN, h_i and h_j represent the state of node i and j , respectively. W^l is a linear transformation weight, b^l is a bias term, and $\sigma(\cdot)$ is a nonlinear function, such as ReLU. The output of the last layer in GCN is $H \in \mathbb{R}^{k \times (d_m \times n)}$ which represents the aggregated informations among emotion labels.

In this way, the features among emotion nodes can be aggregated through the GCN module.

2.4 The Output of the Whole Model

At last, we make the output of CNN and GCN multiply together as follows:

$$y = rH^T \in \mathbb{R}^k, \quad (6)$$

where H^T is the transposed matrix of H and y is the last output of the proposed model.

3 Experiments and Results

3.1 Dataset

We evaluated our model on a benchmark dataset: SemEval-2018 Task1 (Affect in Tweets) [16], which contains 10,983 sentences combined with training set (6,838 samples), validation set (886 samples), and testing set (3,259 samples), there are 11 emotion labels in this dataset which are more difficult to conduct recognition task. The statistic of emotion labels in training dataset is shown in Table 1.

We pre-processed each tweet in the dataset like [9], a list of regular expressions was used to recognize the meta information in tweets so as to clean up the unnecessary symbols.

Table 1. The number of labels in training dataset.

Anger	Anticipation	Disgust	Fear	Joy	Love
4282	1916	4563	2124	3876	1401
Optimism	Pessimism	Sadness	Surprise	Trust	Total
3456	1586	3567	760	845	28376

3.2 Compared Models

We compared our model with other five previous related models used to do the this task:

1. **SVM-unigrams** [16]: SVM-unigrams used word unigrams as features and support vector machine to deal with this task.
2. **TCS** [15]: TCS introduced a bidirectional LSTM with attention mechanism in the same task.
3. **PlusEmo2Vec** [20]: PlusEmo2VEc adopted a model with classifier chain and won the third place in SemEval-2018 Task1.
4. **Transformer** [10]: Transformer used a large pre-trained language model to recognize the emotions in sentences.
5. **BNet** [9]: BNet transformed the multi-label task into a binary recognition problem with deep learning, obtaining a better results.

3.3 Evaluation Metrics

In this work, we reported the Jaccard index, Macro-F1 and Micro-F1 for performance evaluation.

When we used the predicted results to calculate the best Macro-F1 and Micro-F1 score, the corresponding thresholds were also selected at the same time, which were represented by $threshold_ma$ and $threshold_mi$, respectively, then we took the average of the two thresholds as the last threshold to determine Jaccard index, that was to say, for a sentence, the emotion labels were predicted as positive if the results of them were greater than $(threshold_ma + threshold_mi)/2$.

3.4 Training and Parameters Setting

At last, we trained our model by using the multi-label recognition loss as follows:

$$loss = \sum_{j=1}^k y_j \log(\sigma(\hat{y}_j)) + (1 - y_j) \log(1 - \sigma(\hat{y}_j)), \quad (7)$$

where $\sigma(\cdot)$ was the sigmoid function. During training, we used the pre-trained “bert-base-uncased” model, where the number of transformer layers was 12 and hidden size d_B was 768. AdamW [14] was used as the optimizer, the learning

rate was set to be $5e-5$, and batch size was 4, epoch number was 20. We padded the sentence with the same length of 128. The out channels of CNN was set to be 200, and we adopted 3 convolution filters which size were 2,3 and 4, respectively, the output size of GCN was decided by the number of filters and the out channels of CNN.

3.5 Results

As shown in Table 2, the best results are bolded, we can know obviously that our model performs very well compared to other models on the same dataset especially on Micro-F1 and Macro-F1 indicators which obtain the highest scores, when comes to compare Jaccard index, our model achieves the second place. These results prove our method is effective.

Table 2. Compared results of our model and other models.

Model	Jaccard index	Micro-F1	Macro-F1
SVM-Unigrams [9]	0.442	0.57	0.443
Transformer [9]	0.577	0.69	0.561
TCS [9]	0.582	0.693	0.53
PlusEmo2Vec [9]	0.576	0.692	0.497
BNet [9]	0.590	0.692	0.564
Our model	0.586	0.701	0.587

We conducted an ablation study to verify the importance of modules in our model. The results are listed in Table 3. When we removed the co-occurrence matrix, the performance of the model was the most worse on Jaccard index. And we replaced GCN with attention mechanism [24], the results were also worse, it was the same as removing CNN. This illustrates the importance of the modules we designed.

Table 3. Ablation study of the proposed model.

	Jaccard index	Micro.F1	Macro.F1
BERT	0.572	0.693	0.573
w/o CNN	0.574	0.694	0.578
w/o Co-oc	0.566	0.686	0.575
Replace GCN with ATT	0.577	0.694	0.565
Whole model	0.586	0.701	0.587

In order to further examine the performance of our model, we calculated the precision score, the recall score and F1 score of every emotion label, which are

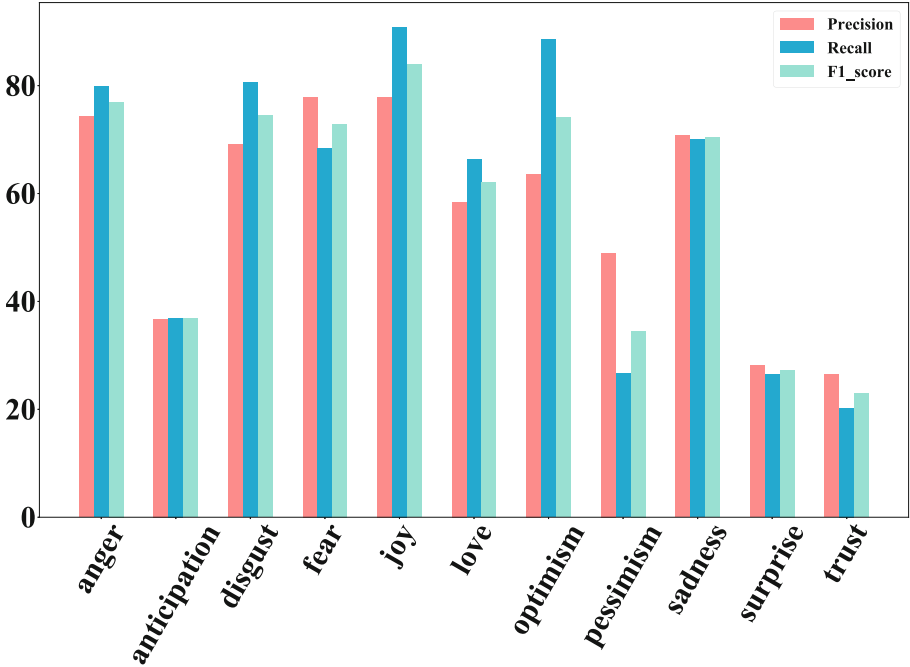


Fig. 2. The performance of our model.

plotted in Fig. 2. As we can see from it, our model clearly recognizes the emotion labels, such as “anger”, “disgust”, “fear”, “joy”, “love”, “optimism” and “sadness”. However, the performances on “anticipation”, “pessimism”, “surprise” and “trust” are worse. We speculate that the reason for this phenomenon may come from the dataset itself, from Table I, we can see the number of these labels are fewer than others resulting in the model not being able to fully learn the emotional characteristics.

Furthermore, we calculated the Pointwise Mutual Information (PMI) [3] by using the co-occurrence matrix built from the test dataset and predicted emotion labels, the PMI can be written as follows:

$$PMI_{ab} = \log \frac{p(a,b)}{p(a)p(b)} = \log \frac{p(a|b)}{p(a)}, \quad (8)$$

where positive values indicated that emotion labels occurred together more than would be expected under an independence assumption and negative values indicated that one emotion label tended to appear only when the other did not [8].

The visualization results of PMI for the true emotional labels in test dataset and the predicted results are shown in Fig. 3 and Fig. 4, respectively. Each grid in the pictures represents the correlation of every corresponding pair of labels. The lighter the color, the greater the correlation. From the comparison of the two figures, we can easily find that the image of predicted results is very similar

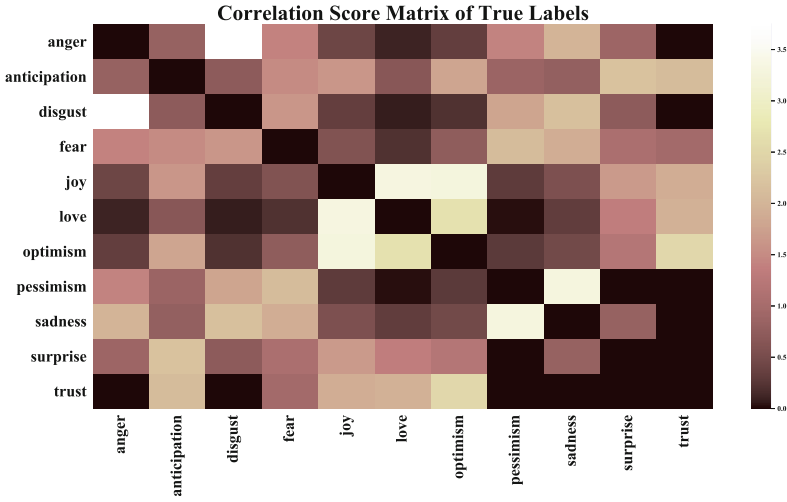


Fig. 3. The visualization of PMI for true labels.

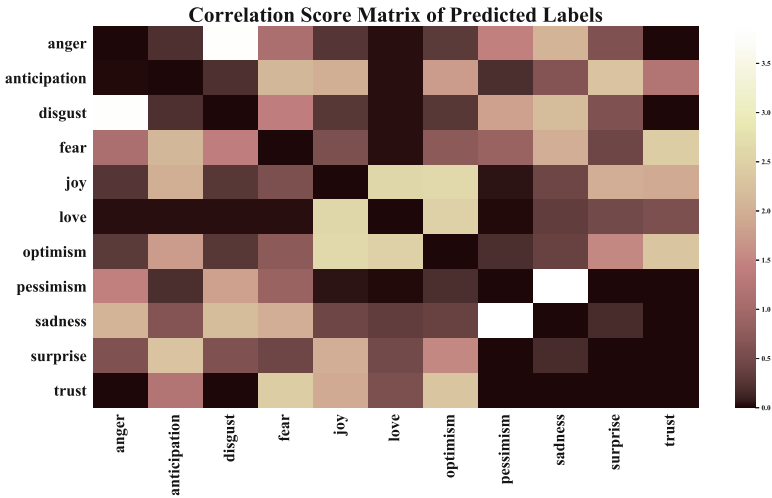


Fig. 4. The visualization of PMI for predicted labels.

to the true labels, meaning our model has captured the dependencies among the emotion labels. On closer inspection from those figures, we can see that the relationship between the emotions “anger” and “disgust” and the relationship between “pessimism” and “sadness” are obvious (the corresponding grid in the picture is lighter), which can correspond to our real life. Furthermore, the correlations among “joy”, “love” and “optimism” are very obvious, which explains

why the samples of “love” is also less but the result is still better than “anticipation”, “pessimism”, “surprise” and “trust” as shown in Fig. 2, because the model has unearthed the dependences between “love” and “joy” and “optimism”.

4 Conclusion

In this paper, we propose a novel method for multi-label emotion recognition based on CNN and GCN. The CNN is used for capturing syntactic and semantic information from sentence word embeddings. In order to effectively mine the correlations characterized by PMI among emotion labels, we build a co-occurrence matrix as well as the labels’ word embeddings acting as the inputs of GCN. At last, the product of their results is taken as the final output. Experimental results show that our model outperforms other methods, and the correlations among emotional labels are captured obviously, demonstrating the feasibility and effectiveness of our approach. In the future, it will be expectant to combine other advanced methodologies such as Graph Attention Network to design a better architecture for this task.

Acknowledgements. This work is supported the National Key Research and Development Program of China (No.2018YFC1604000/2018YFC1604002).

References

1. Albahli, A.S., et al.: Covid-19 public sentiment insights: a text mining approach to the gulf countries. *Comput. Mater. Continua* **67**(2), 913–930 (2021)
2. Baziotis, C., et al.: NTUA-SLP at semeval-2018 task 1: predicting affective content in tweets with deep attentive RNNs and transfer learning (2018). arXiv preprint, [arXiv:1804.06658](https://arxiv.org/abs/1804.06658)
3. Church, K., Hanks, P.: Word association norms, mutual information, and lexicography. *Comput. Linguist.* **16**(1), 22–29 (1990)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding (2018). arXiv preprint, [arXiv:1810.04805](https://arxiv.org/abs/1810.04805)
5. Hilal, A., Alfurhood, B., Al-Wesabi, F., Hamza, M., Al Duhayyim, M., Iskandar, H.: Artificial intelligence based sentiment analysis for health crisis management in smart cities. *Comput. Mater. Continua* **71**(1), 143–157 (2022)
6. Hnaif, A.A., Kanan, E., Kanan, T.: Sentiment analysis for Arabic social media news polarity. *Intell. Autom. Soft Comput.* **28**(1), 107–119 (2021)
7. Hou, X., Huang, J., Wang, G., Huang, K., He, X., Zhou, B.: Selective attention based graph convolutional networks for aspect-level sentiment classification (2019). arXiv preprint, [arXiv:1910.10857](https://arxiv.org/abs/1910.10857)
8. Islam, A., Inkpen, D.: Second order co-occurrence PMI for determining the semantic similarity of words. In: *LREC*, pp. 1033–1038 (2006)
9. Jabreel, M., Moreno, A.: A deep learning-based approach for multi-label emotion classification in tweets. *Appl. Sci.* **9**(6), 1123 (2019)
10. Kant, N., Puri, R., Yakovenko, N., Catanzaro, B.: Practical text classification with large pre-trained language models (2018). arXiv preprint, [arXiv:1812.01207](https://arxiv.org/abs/1812.01207)

11. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks (2016). arXiv preprint, [arXiv:1609.02907](https://arxiv.org/abs/1609.02907)
12. Lai, Y., Zhang, L., Han, D., Zhou, R., Wang, G.: Fine-grained emotion classification of Chinese microblogs based on graph convolution networks. *World Wide Web* **23**(5), 2771–2787 (2020)
13. Liu, B.: *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge University Press (2020)
14. Loshchilov, I., Hutter, F.: Fixing weight decay regularization in Adam (2018)
15. Meisheri, H., Dey, L.: TCS research at semeval-2018 task 1: learning robust representations using multi-attention architecture. In: *Proceedings of the 12th International Workshop on Semantic Evaluation*, pp. 291–299 (2018)
16. Mohammad, S., Bravo-Marquez, F., Salameh, M., Kiritchenko, S.: Semeval-2018 task 1: affect in tweets. In: *Proceedings of the 12th International Workshop On Semantic Evaluation*, pp. 1–17 (2018)
17. Twitter Arabic sentiment analysis to detect depression using machine learning. *CMC Comput. Mater. Continua* **71**(2), 3463–3477 (2022)
18. Mutanov, G., Karyukin, V., Mamykova, Z.: Multi-class sentiment analysis of social media data with machine learning algorithms. *Comput. Mater. Continua* **69**(1), 913–930 (2021)
19. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? sentiment classification using machine learning techniques (2002). arXiv preprint [cs/0205070](https://arxiv.org/abs/cs/0205070)
20. Park, J.H., Xu, P., Fung, P.: Plusemo2vec at semeval-2018 task 1: exploiting emotion knowledge from emoji and# hashtags (2018). arXiv preprint, [arXiv:1804.08280](https://arxiv.org/abs/1804.08280)
21. Read, J., Pfahringer, B., Holmes, G., Frank, E.: Classifier chains for multi-label classification. *Mach. Learn.* **85**(3), 333 (2011)
22. Suhail, K., et al.: Stock market trading based on market sentiments and reinforcement learning. *CMC-Comput. Mater. Continua* **70**(1), 935–950 (2022)
23. Tang, D., Qin, B., Liu, T.: Document modeling with gated recurrent neural network for sentiment classification. In: *Proceedings of the 2015 conference on empirical methods in natural language processing*, pp. 1422–1432 (2015)
24. Vaswani, A., et al.: Attention is all you need. In: *Advances in Neural Information Processing Systems*, pp. 5998–6008 (2017)
25. Yao, L., Mao, C., Luo, Y.: Graph convolutional networks for text classification. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 7370–7377 (2019)
26. Zhang, C., Li, Q., Song, D.: Aspect-based sentiment classification with aspect-specific graph convolutional networks (2019). arXiv preprint, [arXiv:1909.03477](https://arxiv.org/abs/1909.03477)