Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/43/20230847

Multilingual sentiment analysis during the pandemic using deep learning models

Yichen Chang

School of Mathematics and Physics, Xi'an Jiaotong-Liverpool University, Suzhou, Jiangsu, 215123, China

yichen.chang19@student.xjtlu.edu.cn

Abstract. High mood swings and increased mortality with the COVID-19 outbreak, mood changes are a concern. COVID-19 is extremely contagious, and by December 20, 2021, the global death toll from COVID-19 exceeded 5.35 million. Under such circumstances, people changed their original way of life and isolated themselves from home to work online. Most people in this situation expressed their emotions through social media. In order to understand the changes of people's emotions during the pandemic, this research analyzed the emotions of people's messages posted on social media in different countries through the SenWave dataset by different deep learning models. The dataset contains 6 languages (English, Arabic, Spanish, French, Italian and Chinese) with more than 70,000 tweets, in this paper, only tweets collected from Twitter are concerned, which contains 11 labels. The dataset will be preprocessed and transferred into numerical vector, then TextCNN, BERT, and BERT-TextCNN models are implemented for the above problems. BERT model reached 0.69 and 0.53 on precision and recall respectively on Twitter dataset.

Keywords: Multilingual sentiment analysis, multi-label classification, deep learning.

1. Introduction

1.1. Background

Sentiment analysis is the opinion mining or polarity analysis based on the text, using different machine learning or deep learning methods to analyse and classify the text into specific emotions [1]. Sentiment analysis on social media is often used to analyse people's opinions, emotions and attitudes posted on social media towards key topics by using machine learning and deep learning methods, which has been employed in many fields, such as public opinion monitoring [2]. After the break out of Covid-19, people have been voicing more opinions about the government, the epidemic, community management, and the health sector on social media to support or oppose polices [3]. These views were of great significance in helping the government respond and reassure the people by utilizing their emotional state through real-time detection [4]. This research aims to employ different deep learning models to analysis the sentiment of people's social media posts during the pandemic.

1.2. Problem Statement

During the epidemic, many entertainment facilities were closed, and many people had to work or attend classes from home instead of having face-to-face discussions with colleagues and classmates; thus, people's mental health problems may suffer as a result [3]. Residents were confined to their homes and were prone to negative emotions when faced with a shortage of supplies. Research shows that lack of interaction and communication can lead to anxiety, anger, panic and other negative emotions [5]. According to research [6], the percentage of college students in the United States with major depression grew to 23.7 percent during the pandemic, and 33 percent of college students were prone to major anxiety. In order to help the government to timely observe residents' mood fluctuations, we trained multiple deep learning models on multi-lingual text sentiment analysis dataset to make predictions. To visually show the effect of these models, different charts will be used to visualize different categories of different sentiment classification.

1.3. Motivation, Aims, and Objectives

This research aims to apply deep learning models to analyze people's emotions, to be able to understand the current general public sentiment. The main objectives of this project are the following:

(1) Use different NLP techniques for data preprocessing.

(2) Employ different deep learning models on multi-label dataset.

(3) Analysis and validation of different deep learning models and algorithmic in the context of their suitability in sentiment analysis.

2. Related Work

2.1. Machine Learning Models

Mohammad et al. [7] used different models (SVM, LSTM, Bi-LSTM, Linear Regression, etc.) for sentiment classification of English, Arabic, and Spanish tweets, which contain 11 labels that the model based on deep neural network had a better performance. The highest accuracy achieved by NTUA-SLP team with deep neural network on English tweets was 58.8%. Law and Ghosh [8] reformed the binary tree classifier and strategically divided the input data into two subsets of subsequent sub nodes to preserve the problem of label dependency and class imbalance. This binary tree classifier for multi-label classification outperformed the other 14 multi-label classifiers on 14 datasets. Machova et al. [9] used machine learning methods to detect suspicious reviewers, and dictionary-based sentiment analysis methods to identify typical emotions of troll comments. Support vector machines obtained the best results of machine learning methods with an accuracy of 0.986 and F1 value of 0.988. Lasri et al. [10] applied six different machine learning methods (Random Forest, multinomial Naive Bayes classifier, Logistic Regression, Decision Tree, Linear Support Vector classifier, and Extreme Gradient Boosting) to classify tweets collected from Twitter about the University of Morocco into three labels (positive, negative, and neutral). The results showed that the accuracy of random forest classifier was up to 90%, and the classification performance was the best among the six machine learning methods.

2.2. Deep Learning Models

The Fan et al. [11] used the TextCNN model for automatic labeling and classification of environmental complaints, and achieved good performance in issuing classification in feature dictionary. Compared with other deep learning models, this method performed best with a precision of 86.2%, a recall of 85.7% and F1 value of 85.9%. Guo et al. [12] improved the classification performance of TextCNN by assigning multiple weights to each word and applying these weights to the word embedding of the word respectively. The accuracy of the model reached 86.6%.

Cao et al. [13] proposed a new recognition method to identify the emotional tendency of consumers in the evaluation of agricultural products through speech rules, which is based on improved symmetric structure of BERT model. Comparing with the original BERT model, the F1 value increased by 7.05 in the test set. By comparing various machine learning and deep learning techniques and word embedding

DOI: 10.54254/2755-2721/43/20230847

methods, Kumar et al. [14] used trained models to predict emotions in Hindi texts. The accuracy of mBERT model reached 91.84% in the test data, comparing with other machine learning and deep learning methods, mBERT showed the best performance.

Zhang and Ma [15] proposed a hybrid model based on albert, TextCNN and PCA to enhance topic knowledge for emotional analysis of sudden disasters. The model performance was evaluated on microblog data of rainstorm disasters in China and the highest accuracy received with ALBERT-TextCNN combining hierarchical attention mechanism with LDA was around 0.8879. Abdillah et al. [16] adopted a combination of heterogeneous deep learning and machine learning to perform multi-label extended text classification on 15 different single models, including 3 deep learning algorithms (CNN, LSTM, and BERT) and 4 machine learning algorithms (SVM, kNN, Decision Tree, and Naive Bayes), as well as different text representations (TF-IDF, Word2Vec, and FastText). The results showed that the comprehensive model was better than the single model for biomedical quality assurance classification.

3. Methodology

3.1. Word Embedding

The features required by the machine model are digitized, and the features applicable to the machine model can be extracted from the text to process the data through word embedding. BERT is analysed in this paper.

BERT uses the structure of transformer and considers the information of the left and right words in the sentence (sequence) of word to realize the characterization of the word [17]. Bert encodes word order information in transformer by means of attention method and training of mask language model as well as embedding. Through the above way, both context and word order information are considered and word embedding is realized. BERT can take one or two sentences as input and expect special tokens to mark the beginning (<CLS>) and end of each sentence (<SEP>) [18], a sample of two sentences input is shown in Figure 3-1.

Figure 3-1. Input Sentence of BERT: The first token of every sentence is always a <CLS> token, to separate sentences in sentence pair using <SEP>, at the end of the sentence pair using <SEP> to denote the ends.

Tokenizer first check whether the word is in the corpus, if not, it will attempt to break down words into the largest possible sub words contained in the vocabulary and, as a last resort, into individual characters [19]. Then tokenizer matches these tokens against the index in the vocabulary. Thus, the input_ids (the index in the corpus for each input word), attention_mask (if the sentences vary in length, fill them with 1 for uniform length), and token_type_ids (distinguish the sentences in sentence pairs, 0 corresponds to the first sentence and 1 corresponds to the second sentence) [18]. These three vectors are input into the model, and the model outputs three embeddings, token embeddings, the segmentation embeddings, and the position embeddings, and final input embeddings are the sum of these three embeddings.

3.2. Deep Learning Architectures

Three models are used for sentiment analysis on multi-label dataset. The flow chart is shown in Figure 3-2.

DOI: 10.54254/2755-2721/43/20230847

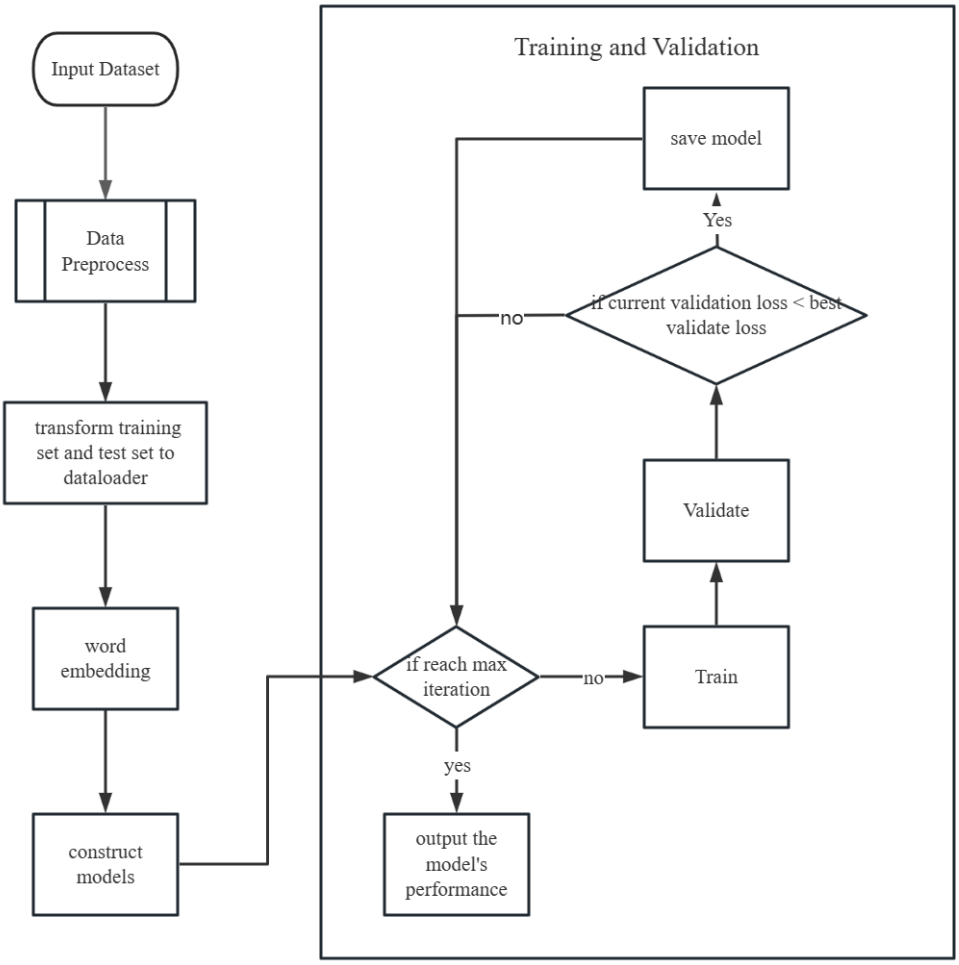

Figure 3-2. Project Flow Chart.

3.2.1. BERT

Google researchers developed BERT in 2018, based on the transformer, with pre-training on BookCorpus via transfer learning [18]. There are currently 22 BERT models, based on SenWave's multilingual dataset [20] [21]. The project used the "bert-base-multilingual-cased", which contains 104 languages, 12 encoder layers, 768 hidden, 12 heads and 100M parameters [18]. The architecture of BERT is shown in Figure 3-3. BERT is pre-trained on a large corpus of multilingual data through self-supervision, randomly masking 15% of input words using MLM, and then predicts them by running sentences with masked words through the model [22]. During the pretraining process, the model combined two masked sentences into one input, and then check whether these two sentences are consistent through NSP [23].

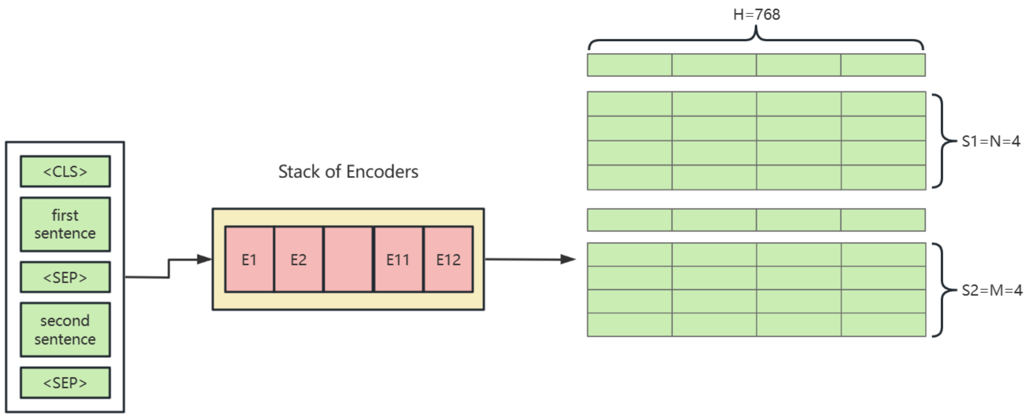

Figure 3-3. Architecture of BERT.

3.2.2. *TextCNN*

TextCNN is yet another text classification model proposed based on CNN model, which contains a convolutional layer and a maximum pooling layer [24]. The output will be external sigmoid function for classification.

The convolution layer can do a point-wise multiplication and sum between the matrix corresponding to the word vector and the convolution kernel, and finally output a feature map by constantly translating n distances [25]. The position of each convolution kernel slide is a complete word, ensuring the rationality of word as the minimum granularity in the language. In this project, the kernel size = $[2, 3, 4]$, that is each window can contain 2, 3, and 4 words, respectively. Every kernel output 2 channels, which means the number of convolutional kernels of each kind is 2. The results are then passed into the max-pooling layer via the sigmoid function. After obtaining feature map, select a maximum value as the output, which is max-pooling. The output result of max-pooling layer is used as the input of the full connected layer, and the final prediction result is obtained through sigmoid function. The architecture of TextCNN model is shown in Figure 3-4.

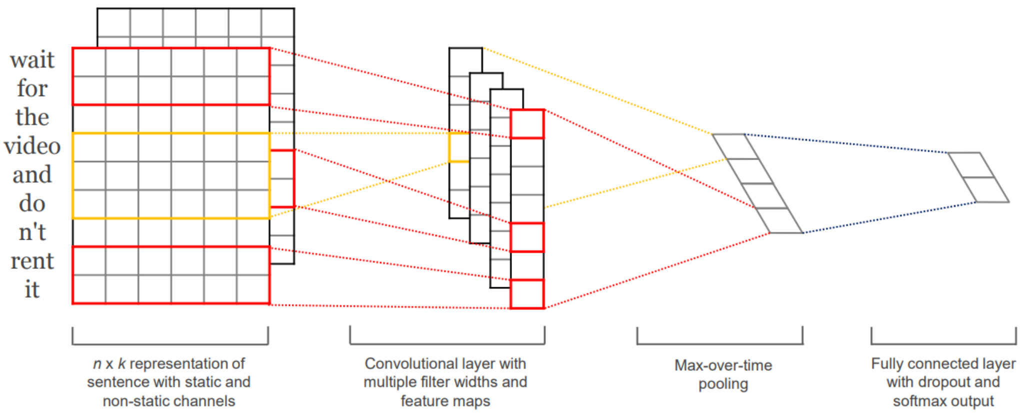

Figure 3-4. TextCNN Architecture [24]: n represents the length of the sentence and k is the word vector dimension of the sentence, number of channels is 2.

3.2.3. *BERT-TextCNN*

Due to BERT's tendency to better categories cryptic sentences, while TextCNN tends to be more sensitive to keywords, an attempt is made to combine the BERT model and TextCNN model to obtain better results [26]. First, BERT output the pretrained word vector, whose shape is $[batch -$

$size, max\_length, 768]$. As TextCNN needs to obtain four-dimensional data, channel dimension is added so that the input shape is $[batch - size, channel, \max\_length, 768]$. Then use the convolution kernel with width $[2, 3, 4]$ and length of 768 to convolve with the input $[batch - size, channel, \max\_length, 768]$. The output results are concat together. Finally, a full connection layer is connected to output the corresponding labels.

3.3. Evaluation of the Proposed System

In the training process, MultiLabelSoftMarginLoss function is set to judge the training condition of the model. Loss functions are calculated by [27]:

$$\begin{aligned} &MultiLabelSoftMarginLoss(x, y) \\ &= -\frac{1}{C}\sum_{i} y[i] * \log((1 + \exp(-x[i]))^{-1}) \\ &+(1 - y[i]) * \log\left(\frac{\exp(-x[i])}{1 + \exp(-x[i])}\right) \end{aligned} \qquad (3-1)$$

where, y represents the actual label and x represents the predicted label.

The results of the models will be evaluated by accuracy, precision, recall, and F1 score. For each sample, the accuracy is correct only if the predicted value is exactly the same as the true value. Thus, for multi-label dataset, the accuracy is defined by:

$$Accuracy(f; D) = \frac{1}{n}\sum_{i=1}^{n}(f(x_i) = label_i) \qquad (3-2)$$

where $f(x_i)$ and $label_i$ are all in list form.

As for multi-label dataset, partial correctness cases need to be considered. Sometimes when the sample has two labels, the model outputs only one correct label. Although the model does not predict all the labels, it does partial prediction correctly. Thus, precision and recall help to get the partial correctness rate.

For each sample, precision is the proportion of the number of predicted correct labels in the total number of predicted correct labels [28], and recall is the number of predicted correct labels as a proportion of the total number of predicted correct labels [29].

$$Precision = \frac{1}{|S|}\sum_{s \in S}\frac{|y_s \cap \widehat{y_s}|}{|\widehat{y_s}|} \qquad (3-3)$$

$$Recall = \frac{1}{|S|}\sum_{s \in S}\frac{|y_s \cap \widehat{y_s}|}{|y_s|} \qquad (3-4)$$

where, y_s represents the true labels, and $\widehat{y_s}$ represents the correct label predicted by the model.

The F-score is a composite metric that is a weighted harmonic average of precision and recall [30]. The F1 score is based on a harmonic average of precision and recall, calculated by:

$$F1 = \frac{1}{|S|}\sum_{s \in S}\frac{2 \times |y_s \cap \widehat{y_s}|}{|\widehat{y_s}| + |y_s|} \qquad (3-5)$$

4. Experiment Results

4.1. Dataset Description

SenWave is a multi-lingual dataset collected by Yang et al. [20] [21] that contains six datasets of six languages (English, Arabic, Spanish, French, Italian and Chinese) with more than 70,000 tweets. Except for Chinese dataset, others are collected from Twitter that contains 11 labels. Each tweet may have more than one label, including optimistic, thankful, empathetic, pessimistic, anxious, sad, annoyed, denial, official report, surprise, and joking. In this research, the Twitter datasets were combined to one dataset to implement TextCNN, BERT, and BERT-TextCNN models in the following sections.

DOI: 10.54254/2755-2721/43/20230847

4.2. Data Preprocessing

In order to let model be used in multi-lingual sentiment analysis, the whole Twitter datasets would be preprocessed before training. After removing links, symbols, and numbers on Twitter datasets, tweets will be tokenized, stop words removed, pos tagged and lemmatized. Then the dataset would be transformed to numerical vectors using BERT. An example of tweets after data preprocessing and word embedding is shown in Table 4-1.

Table 4-1. Tweets preprocessing.

	An example from Twitter dataset
Original tweet	Psychopaths stupid policies cannot prevent the spread of coronavirus infection.
After preprocessing	['psychopaths', 'stupid', 'policy', 'prevent', 'spread', 'coronavirus', 'infection']
input_ids	[101, 84970, 18115, 22394, 28780, 90695, 10162, 38572, 25259, 29320, 10105, 25609, 10108, 31206, 37715, 10251, 55223, 119, 102, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
attention_mask	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
token_type_ids	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

4.3. TextCNN/BERT/BERT-TextCNN models

In the process of training TextCNN, the text information is first converted into digital vectors by using Word2Vec, and then the data set is divided into training set and validation set by the ratio of 0.8 and 0.2. After 10 epochs' training and validation, the model would evaluate its performance. BERT and BERT-TextCNN models are trained by 4 epochs with learning rate of 1e-5. The datasets were divided to training set and validation set at rates of 0.8 and 0.2, respectively. The models used training set to train the models, and validation set to check the loss and accuracy in each training. During each training section, each neuron in BERT and BERT-TextCNN models had a 0.3 chance of not excited. If the validation loss is less than the minimum validation loss, then save the current model. After training and validation, the model would evaluate its performance.

The activation function of training is sigmoid function. Before calculating the accuracy of the model on the Twitter dataset, we set a threshold (0.5), that is, if the value output by the model through sigmoid is greater than 0.5 at last, we reassign the value to 1, which means that we consider the content of the tweet to have the emotion corresponding to the label. The accuracy, precision, recall and F1 score of TextCNN, BERT and BERT-TextCNN models are shown in Table 4-3.

Among these models, BERT showed the best performance, with an accuracy of 0.28, a precision of 0.69, a recall of 0.53 and F1 score of 0.59.

Table 4-2. Evaluation of TextCNN/BERT/BERT-TextCNN.

	Twitter dataset			
	Accuracy	Precision	Recall	F1 score
TextCNN	0.1212	0.24	1.00	0.37
BERT	0.2812	0.69	0.53	0.59
BERT-TextCNN	0.1740	0.57	0.29	0.32

Precision, recall, and F1 score are weighted average.

Due to poor computer performance, BERT and BERT-TextCNN models can only train 4 epochs on Twitter dataset, resulting in the weight was not well adjusted. From Table 4-3 we can know that the TextCNN earned the highest value of recall but the precision was only 0.24 on Twitter dataset, which may be caused by the model can only predict one correct label for each sample if the sample has more than one label. The multi-label confusion matrix is shown in Figure 4-1. Refer to the confusion matrix,

DOI: 10.54254/2755-2721/43/20230847

the TextCNN model failed to predict the negative class in every label, which may be caused by the failure to debug the model parameters. BERT-TextCNN has an accuracy of 0.174 and precision of 0.57, that is well-performed in multi-label classification problems, reaching the pass line. However, the recall and F1 value are much lower than the precision, which may be caused by a high threshold (0.5) or many difficult samples in the data set, i.e., a text containing two or three labels with different emotions. We would try to reduce the threshold value to 0.45 and 0.4 to check whether the recall has been improved and continue to dig difficult samples for continuous training of the model to improve the model ability.

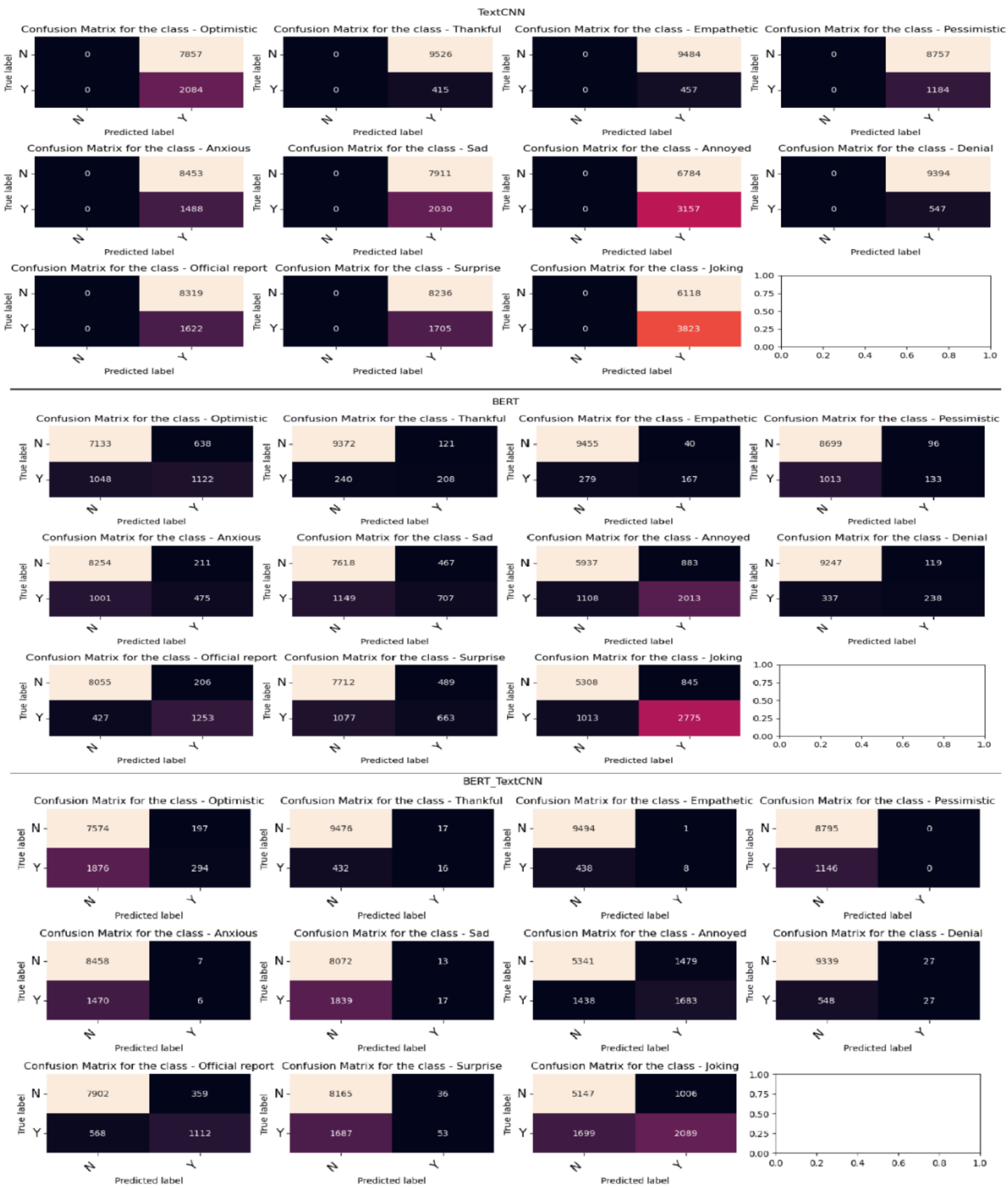

Figure 4-1. Confusion Matrix of TextCNN, BERT, and BERT-TextCNN models on Twitter dataset: N denotes negative and Y denotes positive, that is TN is at position (N, N) and TP is at position (Y, Y).

5. Conclusion and Future Work

In recent years, Covid-19 has been a hot topic of discussion among people. Although China had implemented the opening-up policy, people still concern about the coronavirus. In this project, we used three deep learning models (TextCNN, BERT and BERT-TextCNN models) to train Twitter dataset that people posted post on social media during the pandemic. On this multi-label dataset, the highest accuracy reached 0.2812, precision reached 0.65, recall reached 0.57 and F1 score reached 0.58 with BERT model.

In the future, different parameters and training epochs will be applied to improve the performance of BERT model. Besides, we plan to create a web-based interface social media websites to visualize real-time sentiment analysis, so that the government can timely understand the emotions of residents and take countermeasures after the occurrence of major events.

References

[1] G. Zammarchi, F. Mola, and C. Conversano, 'Using sentiment analysis to evaluate the impact of the COVID-19 outbreak on Italy's country reputation and stock market performance', *Stat. Methods Appl.*, Apr. 2023, doi: 10.1007/s10260-023-00690-5.

[2] Y. Madani, M. Erritali, and B. Bouikhalene, 'A new sentiment analysis method to detect and Analyse sentiments of Covid-19 moroccan tweets using a recommender approach', *Multimed. Tools Appl.*, pp. 1–20, Feb. 2023, doi: 10.1007/s11042-023-14514-x.

[3] 'Feeling Positive About Reopening? New Normal Scenarios From COVID-19 US Reopen Sentiment Analytics | IEEE Journals & Magazine | IEEE Xplore'. https://ieeexplore.ieee.org/document/9154672 (accessed Sep. 15, 2023).

[4] 'Deep Learning Model for COVID-19 Sentiment Analysis on Twitter | SpringerLink'. https://link.springer.com/article/10.1007/s00354-023-00209-2 (accessed Sep. 16, 2023).

[5] 'Studying the psychology of coping negative emotions during COVID-19: a quantitative analysis from India | SpringerLink'. https://link.springer.com/article/10.1007/s11356-021-16002-x (accessed Sep. 16, 2023).

[6] 'Trends and prevalence of suicide 2017-2021 and its association with COVID-19: Interrupted time series analysis of a national sample of college students in the United States - PubMed'. https://pubmed.ncbi.nlm.nih.gov/35987067/ (accessed Sep. 16, 2023).

[7] 'SemEval-2018 Task 1: Affect in Tweets - ACL Anthology'. https://aclanthology.org/S18-1001/ (accessed Sep. 16, 2023).

[8] A. Law and A. Ghosh, 'Multi-Label Classification Using Binary Tree of Classifiers', *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 6, no. 3, pp. 677–689, Jun. 2022, doi: 10.1109/TETCI.2021.3075717.

[9] K. Machova, M. Mach, and M. Vasilko, 'Comparison of Machine Learning and Sentiment Analysis in Detection of Suspicious Online Reviewers on Different Type of Data', *Sensors*, vol. 22, no. 1, Art. no. 1, Jan. 2022, doi: 10.3390/s22010155.

[10] I. Lasri, A. Riadsolh, and M. Elbelkacemi, 'Real-time Twitter Sentiment Analysis for Moroccan Universities using Machine Learning and Big Data Technologies', *Int. J. Emerg. Technol. Learn. IJET*, vol. 18, pp. 42–61, Mar. 2023, doi: 10.3991/ijet.v18i05.35959.

[11] 'Environmental Complaint Text Classification Scheme Combining Automatic Annotation and TextCNN | IEEE Conference Publication | IEEE Xplore'. https://ieeexplore.ieee.org/abstract/document/9727661 (accessed Sep. 18, 2023).

[12] B. Guo, C. Zhang, J. Liu, and X. Ma, 'Improving text classification with weighted word embeddings via a multi-channel TextCNN model', *Neurocomputing*, vol. 363, pp. 366–374, Oct. 2019, doi: 10.1016/j.neucom.2019.07.052.

[13] Y. Cao, Z. Sun, L. Li, and W. Mo, 'A Study of Sentiment Analysis Algorithms for Agricultural Product Reviews Based on Improved BERT Model', *Symmetry*, vol. 14, no. 8, Art. no. 8, Aug. 2022, doi: 10.3390/sym14081604.

[14] T. Kumar, M. Mahrishi, and G. Sharma, 'Emotion recognition in Hindi text using multilingual BERT transformer', *Multimed. Tools Appl.*, Apr. 2023, doi: 10.1007/s11042-023-15150-1.
[15] X. Zhang and Y. Ma, 'An ALBERT-based TextCNN-Hatt hybrid model enhanced with topic knowledge for sentiment analysis of sudden-onset disasters', *Eng. Appl. Artif. Intell.*, vol. 123, p. 106136, Aug. 2023, doi: 10.1016/j.engappai.2023.106136.
[16] A. F. Abdillah, C. B. P. Putra, A. Apriantoni, S. Juanita, and D. Purwitasari, 'Ensemble-based Methods for Multi-label Classification on Biomedical Question-Answer Data', *J. Inf. Syst. Eng. Bus. Intell.*, vol. 8, no. 1, Art. no. 1, Apr. 2022, doi: 10.20473/jisebi.8.1.42-50.
[17] S. Srivastava, B. Paul, and D. Gupta, 'Study of Word Embeddings for Enhanced Cyber Security Named Entity Recognition', *Procedia Comput. Sci.*, vol. 218, pp. 449–460, Jan. 2023, doi: 10.1016/j.procs.2023.01.027.
[18] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, 'BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding'. arXiv, May 24, 2019. doi: 10.48550/arXiv.1810.04805.
[19] B. Heinzerling and M. Strube, 'BPEmb: Tokenization-free Pre-trained Subword Embeddings in 275 Languages'. arXiv, Oct. 05, 2017. Accessed: Apr. 23, 2023. [Online]. Available: http://arxiv.org/abs/1710.02187
[20] 'Rise and fall of the global conversation and shifting sentiments during the COVID-19 pandemic | Humanities and Social Sciences Communications'. https://www.nature.com/articles/s41599-021-00798-7 (accessed Oct. 31, 2022).
[21] Q. Yang *et al.*, 'SenWave: Monitoring the Global Sentiments under the COVID-19 Pandemic'. arXiv, Jun. 18, 2020. doi: 10.48550/arXiv.2006.10842.
[22] J. Qiang, Y. Li, Y. Zhu, Y. Yuan, and X. Wu, 'Lexical Simplification with Pretrained Encoders'. arXiv, Oct. 28, 2020. Accessed: Apr. 16, 2023. [Online]. Available: http://arxiv.org/abs/1907.06226
[23] E. Çelik and T. Dalyan, 'Unified benchmark for zero-shot Turkish text classification', *Inf. Process. Manag.*, vol. 60, no. 3, p. 103298, May 2023, doi: 10.1016/j.ipm.2023.103298.
[24] Y. Kim, 'Convolutional Neural Networks for Sentence Classification'. arXiv, Sep. 02, 2014. Accessed: Apr. 23, 2023. [Online]. Available: http://arxiv.org/abs/1408.5882
[25] B. Lu, Z. Lin, and X. Zhang, 'Text Analysis Method Based on Multi-channel Parallel Classifier', in *2022 IEEE 2nd International Conference on Information Communication and Software Engineering (ICICSE)*, Mar. 2022, pp. 6–12. doi: 10.1109/ICICSE55337.2022.9828968.
[26] X. Jiang, C. Song, Y. Xu, Y. Li, and Y. Peng, 'Research on sentiment classification for netizens based on the BERT-BiLSTM-TextCNN model', *PeerJ Comput. Sci.*, vol. 8, p. e1005, Jun. 2022, doi: 10.7717/peerj-cs.1005.
[27] 'torch.nn — PyTorch 2.0 documentation'. https://pytorch.org/docs/stable/nn.html (accessed Apr. 23, 2023).
[28] J. Deng and F. Ren, 'Multi-Label Emotion Detection via Emotion-Specified Feature Extraction and Emotion Correlation Learning', *IEEE Trans. Affect. Comput.*, vol. 14, no. 1, pp. 475–486, Jan. 2023, doi: 10.1109/TAFFC.2020.3034215.
[29] Q. Lv, W. Liu, R. Li, H. Yang, Y. Tao, and M. Wang, 'Classification of Seismaesthesia Information and Seismic Intensity Assessment by Multi-Model Coupling', *ISPRS Int. J. Geo-Inf.*, vol. 12, no. 2, Art. no. 2, Feb. 2023, doi: 10.3390/ijgi12020046.
[30] H.-D. Le, G.-S. Lee, S.-H. Kim, S. Kim, and H.-J. Yang, 'Multi-Label Multimodal Emotion Recognition With Transformer-Based Fusion and Emotion-Level Representation Learning', *IEEE Access*, vol. 11, pp. 14742–14751, 2023, doi: 10.1109/ACCESS.2023.3244390.