Advantages of XLM-R Model for Urdu Sentiment Multi-Classification

Mingcan Guo, Zhongyuan Han*, Leilei Kong, Zhijie Zhang, Zengyao Li, Haoyang Chen and Haoliang Qi

Foshan University, Foshan, China

Abstract

Sentiment Multi-Classification detection has gained much attention in recent years. The multi-label sentiment detection task refers to additional comments based on social media or shopping platforms, which usually contain different personal solid emotions. How to classify these reviews into multiple sentiment types using efficient methods such as machine learning is the main content of this type of task. We describe our XLM-R based method for tracking emotion detection task at FIRE 2022 in this paper. The system uses the XLM-R pretrained model to extract semantic features from Urdu text. After using dynamic learning rate-based tuning, we found that the model is more stable in performance and has a higher score on the test set. In the final result, our system achieved a Micro F1 score of 0.759 and a Macro F1 score of 0.687 in this task and won the first rank in the FIRE2022 track emotion detection task.

Keywords

sentiment, Multi-Classification, XLM-R, social media

1. Introduction

With the continuous iteration of Internet technology, more and more people choose to express their emotional remarks and opinions on social media such as Twitter and Facebook [1]. People hope to openly express their views on social economy, culture, politics, etc. from multiple perspectives. How to classify these emotional speeches and give them different emotional labels will have a positive effect on managing and regulating the community and improving the user experience. It can also help companies and governments to collect rich emotional information. At the same time, it has important implications both in the research and industrial fields of artificial intelligence. Therefore, some method needs to be used to identify and divide these speeches, which is the task of multi-label sentiment detection [2].

With the advent of the Internet age, the world is connected as a whole, people from different countries can communicate freely on the Internet, and thousands of speeches are posted daily, providing rich language resources for the emotion detection system in past research. Urdu is the official language of Pakistan. It is the 10th-most widely spoken language in the world, with 230 million total speakers[1]. It is widely used in many countries, such as India and Nepal. Because of the large number of speakers and the wide range of application areas of Urdu, it is crucial to understand the native speakers of Urdu by analyzing and studying the sentiment classification of Urdu by language processing systems.

In the emotions and threat detection shared tasks at FIRE 2022 task [3, 4], our team uses the Urdu script dataset [5] and a pre-trained XLM-R based language processing system for multi-label classification. To achieve faster model convergence and better model performance, we found that the

[1] https://en.wikipedia.org/wiki/Urdu

FIRE 2022: Forum for Information Retrieval Evaluation, December 9–13, 2022, India

EMAIL: gmc9812@163.com(M. Guo); hanzhongyuan@gmail.com (Z. Han)(*corresponding author); kongleilei@fosu.edu.cn (L. Kong); zhangzhijie5454@gmail.com(Z. Zhang); lzy1512192979@gmail.com(Z. Li) hoyo.chen.i@gmail.com(H. Chen); haoliang.qi@gmail.com(H. Qi)

ORCID: 0000-0002-4977-2138 (M. Guo); 0000-0001-8960-9872 (Z. Han); 0002-4636-3507(L. Kong); 0000-0002-4854-0618 (Z. Zhang); 0000-0001-8472-4150 (Z. Li); 0000-0003-3223-9086 (H. Chen); 0000-0003-1321-5820 (H. Qi)

XLM-R model has a strong normalization ability for non-mainstream languages like Urdu, and can well meet the needs of the task after learning rate tuning, so we use model tuning and adaptive learning rate tuning to verify the effectiveness of sentiment classification in this dataset.

The overall result distribution of this paper is as follows. In the second part, we will introduce the related work and give an overview of the related historical research. We will present our experimental method in the third part and submit the practical steps and associated details in the fourth part. Finally, we will analyze the results of our experiments on the official dataset and add a summary to the whole paper.

2. Related Work

People have been working on multi-label sentiment classification for years [6]. People usually divide sentiments according to rules and dictionaries in traditional sentiment classification. For example, J. Blitzer et al. [7] extended the correspondence learning (SCL) algorithm [8] to sentiment classification to detect Amazon's products of different product types. Regarding the sentiment of comments, K. Deneck studied the multi-domain sentiment analysis problem based on SentiWordNet as a lexical resource [9]. G. Xu et al. proposed an extended dictionary-based Chinese sentiment analysis method on the rule-based analysis method [10]. Their experiments build a vast sentiment lexicon covering five domains: hotel, number, fruit, clothing, and shampoo. However, traditional methods do not work well on social media platforms that generate many comments of different types daily.

In recent years, with the continuous development of neural network and convolutional network technology and the emergence of transformer models with excellent self-attention mechanisms [11], rule-based analysis methods have gradually been replaced by feature-supervised learning methods based on pre-trained models, S. Mahata et al. [12] proposed a model based on bidirectional LSTM and language tagging using FastText embedding to generate word vectors to train the model, trying to solve the sentiment analysis problem of English-Tamil code mixed data, B. R. Chakravarthi et al. [13] propose Long Short Term Memory (LSTM) networks and language-specific pre-processing, they involved applying an attention layer on contextualized word embeddings and fine-tuning a model pretrained on the training data of the previous version, DravidianCodeMix-2020, to recognize Tamil and Malayalam in social media comments Emotions in mixed languages.

In addition to traditional rule-based sentiment classification and popular neural network pre-trained model-based sentiment classification, Y. Wu et al. [14] proposed a multimodal sentiment classification method based on cross-modal prediction centered on text modality, two types of information are mined from speech modalities and image modalities to assist text modalities, and a text-centric multimodal feature fusion mechanism is designed to perform feature fusion on multimodal features.

In previous sentiment analysis tasks, transformers-based models often achieve good results in different tracks [15]. For example, Y. Bai et al. [16] combined the fine-tuning method of XLM-RoBERTa and CNN through downstream tasks and obtained first place in the mission of Sentiment Analysis of Dravidian Languages in Code-Mixed Text. In shared task, study by L. Khan et al. [17] shows that the combination of word n-gram features with LR outperformed other classifiers for sentiment analysis task, obtaining the highest F1 score of 82.05% using combination of features. In related supervised classification problems, the content-based word unigram method used by I. Ameer et al. [18] outperforms other content-based feature-based methods. L. Khan et al. [19] used four text representations: word n-grams, char n-grams, pretrained fastText, and BERT word embeddings to train the classifier. Their proposed mBERT model with BERT-pretrained word embeddings outperforms deep learning, machine learning and rule-based classifiers and achieves an F1 score of 81.49%.

3. Methodology

3.1. Model Description

In this task, our method uses a pre-trained model based on XLM-RoBERTa [20], referred to as XLM-R. It inherits the method of XLM and draws on the ideas of RoBERTa [21]. Compared with XLM, XLM-R expands not only the language but also the training data. Therefore, similar to other transformer structures, the architecture of XLM-R can be and is better suited for text classification tasks. It takes as input a sequence of no more than 512 tokens and outputs a representation of that sequence. The first token of the sequence is always [CLS], which contains the particular categorical embedding. As shown in Figure 1, Urdu tweets are passed into token classification, a linear classification layer that takes the token sequence and the final hidden state of each Urdu text as input and assigns it to each token. The cards generate the label output, and the top softmax classifier is used to predict the probability of label C, which is finally classified into different sentiment labels.

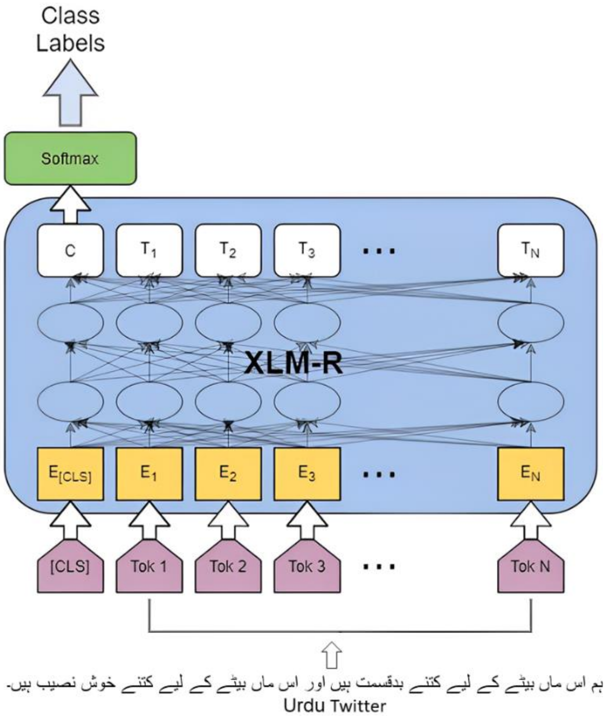

Figure 1: Urdu text classification architecture with XLM-R [22]

3.2. Neural Network Tuning

Among many optimization methods, learning rate-based tuning is usually given priority. M. D. Zeiler [23] uses SGD, Momentum, ADAGRAD, and ADADELTA in a supervised manner. The neural network is trained to minimize the cross-entropy between the network output predictions and the target labels, and the results show that the model is sensitive to the parameters of the learning rate, and an accurate learning rate can quickly converge the model error around the optimal performance that occurs with momentum.

Common learning rate change strategies include preset rule learning rate change methods, including fixed, step, exp, inv, multistep, poly, sigmoid, etc. Compared with non-adaptive learning rate transformation methods, the model's absolute value is reduced, and performance impact is. In this task, we use ReduceLROnPlateau [24] in Keras to adjust the learning rate strategy, based on the Adam [25] adaptive learning rate algorithm, we detect the change of the Loss index at each epoch. The learning rate adjustment will be triggered when the Loss no longer decreases within a certain period. The adjustment strategy is shown in formula 1, where λ is the attenuation multiplication factor.

$$new\_lr = \lambda \times old\_lr \tag{1}$$

The learning rate decay in our experiments is shown in Figure 2. The value on y-axis in the figure is magnified by 10^5 times. According to the algorithm, after it is detected that the Loss no longer decreases for a certain period, the learning rate will be automatically lowered in the next epoch to make the model converge to the best performance continuously.

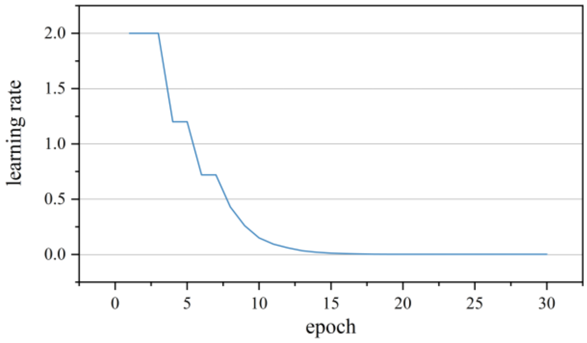

Figure 2: Learning rate varies with the epoch

4. Experiments

4.1. Data and Pre-Processing

The dataset provided by the task is obtained through Twitter [2], tweets are collected in a CSV file through the Twitter open API, other kinds of languages are excluded, and only the purest Urdu tweets are kept. These comments can be divided into seven emotions: anger, disgust, fear, sadness, surprise, happiness, and neutral. A total of over 10,000 tweets collected are divided into 7,800 training sets and 1,950 test sets. Table 1 shows statistical data according to different categories. All Urdu texts are normalized with diacritics removed and spaces added after numbers, punctuation, and stop words.

Table 1

The division of train and test sets for different emotion types

Datasets	Labels	Training	Test
Urdu Tweets	neutral	3014	753
	happiness	1046	261
	surprise	1550	388
	sadness	2190	548
	fear	609	152
	disgust	761	190
	anger	811	203
	Total	7800	1950

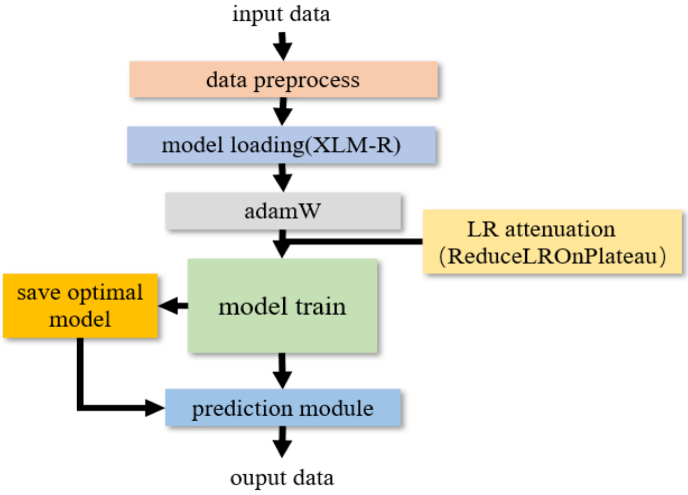

Figure 3: Experimental process

We use the XLM-R model to train the dataset and add the adaptive learning rate change strategy tuning in this work. The hyperparameters of the XLM-R model are set to lr=2e-5, batch_size=32, max_len=128, hidden_size=768, epochs=15, learning rate decay parameters are set to factor=0.6, cooldown=0, min_lr=0, eps=1e-08.

After loading the model, we choose adamW as the optimizer of the model and cross entropy loss as the loss function. During training, we set the model to be verified every 100 steps and added the gradient reset to save the optimal model when confirming the set. The learning rate gradient decays, and finally, we use the activation function in the output layer to output the predicted label of the model. Figure 3 shows the overall architecture of the experimental process.

Before the official release of the test set with labels, we divided the training set into 9:1 for model training and model validation.

4.2. Results

We use a variety of evaluation metrics to verify the performance of our model, including Accuracy, Precision, Recall, Macro F1, Micro F1, and Loss. And we choose Accuracy and Loss among these metrics to validate the model in train set.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{2}$$

$$Loss = -w * [p * log(q) + (1-p) * log(1-q)] \tag{3}$$

In training the model, we added the methods without learning rate optimization for comparison. After each epoch, we call the index evaluation method in the sklearn method library to output the evaluation of the model on the validation set. As shown in Figure 4 and 5, on the two evaluation indicators of Accuracy and Loss, the model tuned by the adaptive learning rate can converge to near the best performance to obtain the highest evaluation score. With the adjustment of the learning rate, the model’s score is gradually stabilized at the optimum on the premise that the Loss function does not increase, and the performance can be kept stable in the process of continuous iteration. The performance indicators of the trained model on the validation data set can reach the best, of which the Accuracy reaches 0.676, and the Loss reaches 0.178.

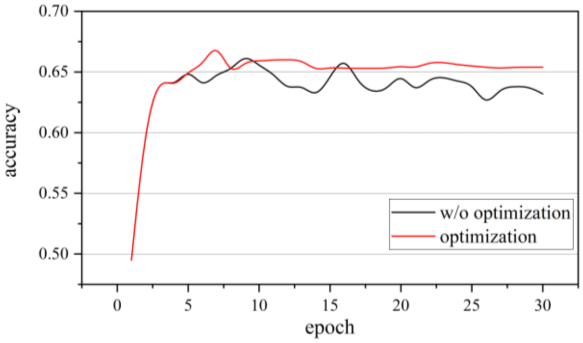

Figure 4: Adjust the learning rate on Accuracy

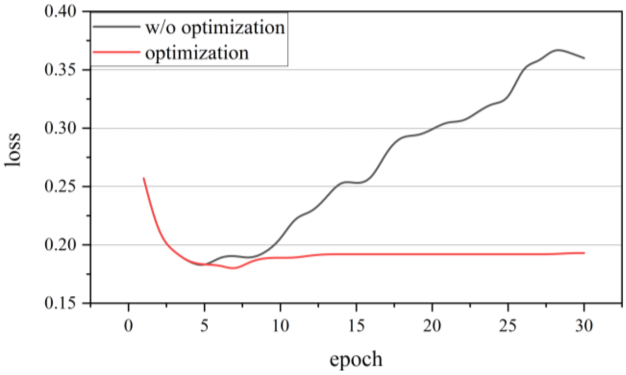

Figure 5: Adjust the learning rate on Loss

As shown in Table 2, the final evaluation of our model on the test set is 0.636 for Accuracy, 0.759 for Weighted F1, 0.759 for Micro F1, 0.687 for Macro F1, and 0.088 for Loss, ranking first among all eight participating teams, which effectively verifies the reliability of our model.

Table 2
Final top six results

Rank	Team	Accuracy	Weighted F1	Micro F1	Macro F1	Loss
1	FOSUNlpTeam	0.636	0.759	0.759	0.687	0.088
2	Team2	0.616	0.743	0.749	0.669	0.088
3	Team3	0.612	0.709	0.742	0.615	0.092
4	Team4	0.582	0.696	0.692	0.603	0.113
5	Team5	0.593	0.699	0.720	0.599	0.092
6	Team6	0.385	0.611	0.477	0.466	0.340

5. Conclusion

This paper mainly introduces our work results on Emotion and Threat detection in Urdu task. Our work combines a pre-trained XLM-R model with adaptive learning rate optimization to solve the multi-label classification problem of Urdu text. The final ranking effectively validated our method. However, combined with the final labeled dataset, we noticed that our method still needs to be optimized, such as paying more attention to the implementation of downstream tasks and the adjustment of model parameters, ignoring the possible impact of preprocessing such as text filtering on the score. Our work on sentence type classification and sample equalization processing are still relatively lacking. The next step will be to preprocess sentence weighting and classification and combine profound learning aspects with building a more robust model processing system. Our code is available on GitHub[2].

6. Acknowledgments

This work is supported by the National Social Science Fund of China (No.18BYY125).

7. References

[1] H. Slim, M. Hafedh, Social media impact on language learning for specific purposes: A study in english for business administration, Teaching english with technology 19 (2019) 56–71.

[2] N. Ashraf, L. Khan, S. Butt, H.-T. Chang, G. Sidorov, A. Gelbukh, Multi-label emotion classification of urdu tweets, PeerJ Computer Science 8 (2022) e896.

[3] S. Butt, M. Amjad, F. Balouchzahi, N. Ashraf, R. Sharma, G. Sidorov, A. Gelbukh, Overview of EmoThreat: Emotions and Threat Detection in Urdu at FIRE 2022, in: CEUR Workshop Proceedings, 2022.

[4] S. Butt, M. Amjad, F. Balouchzahi, N. Ashraf, R. Sharma, G. Sidorov, A. Gelbukh, EmoThreat@FIRE2022: Shared Track on Emotions and Threat Detection in Urdu, in: Forum forInformation Retrieval Evaluation, FIRE 2022, Association for Computing Machinery, New York, NY, USA, 2022.

[5] M. Amjad, N. Ashraf, A. Zhila, G. Sidorov, A. Zubiaga, A. Gelbukh, Threatening language detection and target identification in urdu tweets, IEEE Access 9 (2021) 128302–128313.

[6] P. Zhao, L. Hou, O. Wu, Modeling sentiment dependencies with graph convolutional networks for aspect-level sentiment classification, Knowledge-Based Systems 193 (2020) 105443.

[7] J. Blitzer, M. Dredze, F. Pereira, Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification, in: Proceedings of the 45th annual meeting of the association of computational linguistics, 2007, pp. 440–447.

[8] J. Blitzer, R. McDonald, F. Pereira, Domain adaptation with structural correspondence learning, in: Proceedings of the 2006 conference on empirical methods in natural language processing, 2006, pp. 120–128.

[2] https://github.com/xiguagaizi/multi_label_classification-main.git

[9] K. Denecke, Are sentiwordnet scores suited for multi-domain sentiment classification? in: 2009 Fourth International Conference on Digital Information Management, IEEE, 2009, pp. 1–6.

[10] G. Xu, Z. Yu, H. Yao, F. Li, Y. Meng, X. Wu, Chinese text sentiment analysis based on extended sentiment dictionary, IEEE Access 7 (2019) 43749–43762.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polo-sukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[12] S. Mahata, D. Das, S. Bandyopadhyay, Sentiment classification of code-mixed tweets using bi-directional rnn and language tags, in: Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages, 2021, pp. 28–35.

[13] B. R. Chakravarthi, R. Priyadharshini, S. Thavareesan, D. Chinnappa, D. Thenmozhi, E. Sherly, J. P. McCrae, A. Hande, R. Ponnusamy, S. Banerjee, et al. Findings of the sentiment analysis of dravidian languages in code-mixed text, arXiv preprint arXiv:2111.09811 (2021).

[14] Y. Wu, Z. Lin, Y. Zhao, B. Qin, L.-N. Zhu, A text-centered shared-private framework via cross-modal prediction for multimodal sentiment analysis, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, 2021, pp. 4730–4738.

[15] Y. P. Babu, R. Eswari, Sentiment analysis on dravidian code-mixed youtube comments using paraphrase xlm-roberta model, Working Notes of FIRE (2021).

[16] Y. Bai, B. Zhang, Y. Gu, T. Guan, Q. Shi, Automatic detecting the sentiment of code-mixed text by pre-training model, Working Notes of FIRE (2021).

[17] L. Khan, A. Amjad, N. Ashraf, H.-T. Chang, A. Gelbukh, Urdu sentiment analysis with deep learning methods, IEEE Access 9 (2021) 97803–97812.

[18] I. Ameer, N. Ashraf, G. Sidorov, H. Gómez Adorno, Multi-label emotion classification using content-based features in twitter, Computación y Sistemas 24 (2020) 1159–1164.

[19] L. Khan, A. Amjad, N. Ashraf, H.-T. Chang, Multi-class sentiment analysis of urdu text using multilingual bert, Scientific Reports 12 (2022) 1–17.

[20] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, arXiv preprint arXiv:1911.02116 (2019).

[21] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692 (2019).

[22] T. Ranasinghe, M. Zampieri, Multilingual offensive language identification with cross-lingual embeddings, arXiv preprint arXiv:2010.05324 (2020).

[23] M. D. Zeiler, Adadelta: an adaptive learning rate method, arXiv preprint arXiv:1212.5701 (2012).

[24] A. Gulli, S. Pal, Deep learning with Keras, Packt Publishing Ltd, 2017.

[25] S. J. Reddi, S. Kale, S. Kumar, On the convergence of adam and beyond, arXiv preprint arXiv:1904.09237 (2019).