Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 45 No. 4 (2024)

Emotion Classification Using BERT: A Comprehensive Study

Satti Praveena

[1]*Computer Science Engineering, Koneru Lakshmaiah College of Engineering and Koneru Lakshmaiah University, Guntur, India*

Abstract:-This task is very significant in natural language processing, since emotion detection finds a variety of applications in different aspects, ranging from predicting the customer's sentiment to mental health analysis. The following work delves into the use of transformer-based models for multi-label emotion categorization, especially on BERT, or Bidirectional Encoder Representations from Transformers. Accuracy, recall, and F1-score metrics are used to measure the performance of the model after preprocessing the GoEmotions dataset for text classification into 28 emotions. The BERT model was enhanced, a multi-label classification framework was set up, and visuals were designed to better understand the results. The proposed method performed well, especially when it came to feelings like love, laughter, and adoration. Results here improve transformer-based approaches in the interpretation of emotional text and open a basis for future affective computing applications and research.

Keywords*:Classification, BERT, Natural Language Processing, Goemotions Dataset, Multi-Label Classification, Deep Learning, Transformer-Based Models, Sentiment Analysis, Affective Computing, And Contextualized Embeddings*

1. Introduction

The growing interest in affective computing and sentiment analysis is due to applications in customer feedback analysis, monitoring mental health, virtual assistants, and sentiment detection in social media. Recognizing and coding expressions of emotion in text are vital parts of these applications. Traditional methods of emotion detection are mostly about lexicon-based methods or shallow machine learning models that are not very good at appreciating the subtlety and contextuality of language.

The advent of deep learning and, more precisely, transformer-based models like BERT, has transformed NLP. This is especially very successful in the capture of contextual relationships within text and, thus, is beneficial for tasks like emotion classification.

This paper emphasizes the implementation and evaluation of a BERT-based approach for multi-label emotion detection via the GoEmotions dataset. The dataset, developed at Google, is a comprehensive collection of text annotated using 28 emotions and a neutral category, providing a rich resource for studying nuanced emotional expressions.

This paper will outline the complete pipeline, starting from data preprocessing to model evaluation, for reaching a state-of-the-art solution for emotion classification. The work makes the following key contributions:

1. Precise preprocessing and modeling workflow adapted for multi-label emotion classification.

2. Fine-tuning of a BERT model for achieving high accuracy on the task of predicting fine-grained emotional labels.

3. Visualizations and analysis for interpretation of model predictions and performance.

The rest of the paper is given by dividing it into four sections. Section 2 presents related work in emotion classification.

2. Literature Review

Emotion recognition in text is a very actively pursued area of research in light of applications toward understanding human behavior and in the design of intelligent systems. This section traces the evolution of emotion classification methods with a particular emphasis on the traditional approaches, followed by a more recent application of deep learning techniques.

2.1 Traditional Approaches to Emotion Classification

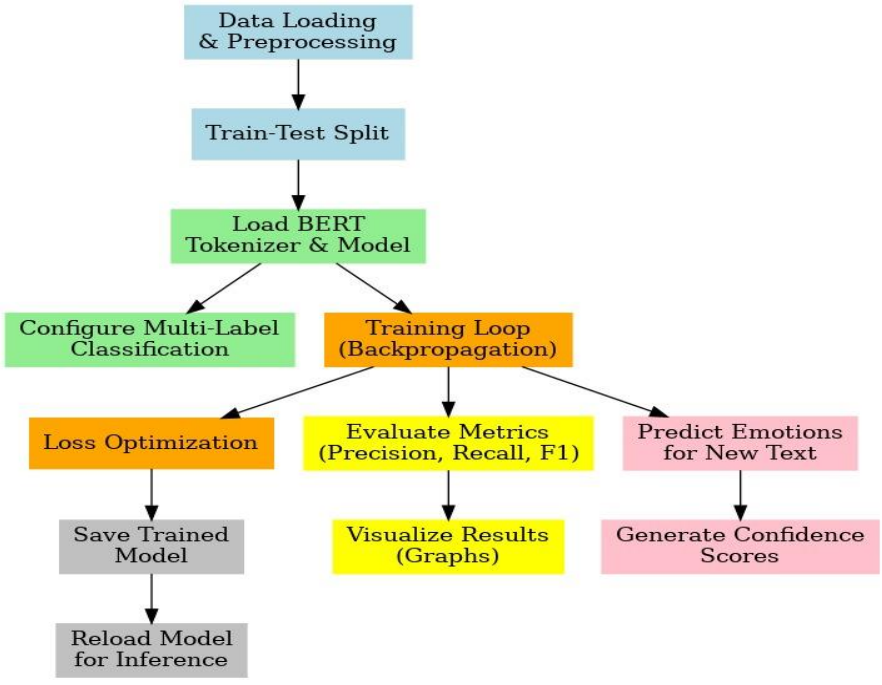
Earlier work on emotion classification was based mainly on lexicon-based methods and classical machine learning models. The lexicon-based approaches mainly involved Affective Norms for English Words (ANEW) and WordNet-Affect, which mapped text to predefined emotion lexicons. Such methods performed well for simple emotion detection but failed to understand contextual understanding in more complex scenarios.

Traditionally, machine learning models that have been applied in this project include SVMs, Naïve Bayes, and Decision Trees. All these are feature-engineered. Features which have been utilized for training include TF-IDF, POS tagging, and syntactic structures. All of these approaches involved extensive feature engineering and were bound by the failure to cope with polysemy and context.

2.2 Deep Learning Breakthroughs

Deep learning revolutionized NLP and further enabled more robust emotion classification. The very first deep architectures were RNNs and LSTM networks that found applications in the detection of emotion. These models performed really well on processing sequential data but presented some computational inefficiencies together with the problem with long-range dependencies.

The transformer-based architecture, specifically the BERT approach, has taken NLP toward a new era. BERT is designed in a bidirectional way that it gets pre-trained for large corpora of text by capturing nuanced relationships in the contexts. It is seen that the emotion classification of fine-tuned BERT results have state-of-the-art outcomes in various different datasets.



2.3 Multilabel Classification of Emotion

The most traditional approaches to emotion classification only allow one dominant emotion in a text. In reality, real-world data are often presented with more than one overlapping emotion. This can be considered by the multi-label classification frameworks by giving a number of emotions to a single text. Proper multi-label loss functions like Binary Cross-Entropy evince very promising experimental results for BERT in catching the overlapping emotional labels.

2.4 The GoEmptions Dataset

GoEmotions is a new dataset proposed by Google. This dataset is the benchmark for emotion classification, covering 28 fine-grained emotion labels and one neutral category, which surpasses the previous datasets in terms of coverage and annotation quality. In the previous works focused on the GoEmotions dataset, comparisons have been made between traditional models and many state-of-the-art transformers in this area. The following work belongs to this line of research because it compares BERT performance against other state-of-the-art models in the task on the GoEmotions dataset.

2.5 Summary

Despite these large strides, emotion classification, specifically nuanced and overlapping emotions in multi-labeling scenarios, remains quite a hard task. This challenge has been quite promising in solving it by the BERT architecture and others similar transformer-based architectures. However, these need to be improved and optimized better for real application use. A good attempt here in the developing trend of multi-label emotion classifications, fine-tuning a model of BERT for the intended application, as well as for its in-depth evaluation.

3. Methodology

3.1 Dataset

This paper relies on the GoEmotions dataset developed by Google. It is based on over 58,000 examples of English text with 28 emotions labels plus a neutral category. There can be more than one label corresponding to one or more text and thus is suitable for multi-label classification.

• Labels: This comprises all those emotions containing admiration, amusement, anger, sadness, joy, etc. along with a neutral label.

• Data Split: The data is split to the training (80%), validation (10%), and test set for a balanced evaluation.

3.2 Data Preprocessing

Input quality ensures that data going into the model is high quality

1.Text Cleaning: Removed additional punctuations, emojis, and non-ASCII characters
2.Tokenization: Tokenized using the BERT's tokenizer in which text gets transformed into token ids that BERT can take
3.Padding and Truncation: Shorter texts padded to have equal number of inputs; longer texts were truncated up to a size of 128 tokens.
4. Label Transformation: Multi-label emotion annotations were converted to binary format for model compatibility.

3.3 Model Architecture

The BERT-base model was fine-tuned for multi-label emotion classification. Major components are:

• Input Layer: Tokenized text with attention masks.
• BERT Encoder: Pre-trained bert-base-uncased model for contextualized embeddings.
• Classification Head: Fully connected layer mapping embeddings to 28 output nodes, one for each emotion.
• Activation Function: Sigmoid activation to enable multi-label predictions.

3.4 Training Procedure

1. Loss Function:
   - Used Binary Cross-Entropy (BCE) loss to handle multi-label classification.
2.Optimizer:
   - AdamW optimizer was chosen for its efficiency in transformer-based models.
3.Learning Rate Scheduler:
   - A linear learning rate scheduler with warm-up steps was applied for stable convergence.
4.Batch Size:
   - Batch size of 16 for training and evaluation.
5.Epochs:
   - Model trained for 3 epochs, balancing computational cost and model performance.

Training was done on a GPU-enabled environment to avail parallel processing.

3.5 Evaluation Metrics

For evaluating the complete performance of the model:

1. Accuracy: Overall accuracy
2. Precision: the ratio of number of relevant emotions in predictions.
3. Recall: ratio of correctly identified actual emotions.
4. F1-Score: Harmonic mean of precision and recall, which offers balanced performance.
5. Subset Accuracy: Number of exactly matched predicted and actual labels ratio.
6. ROC-AUC: It computes how well your model is classifying classes.

3.6 Workflow in Implementation

Implementation was carried out as follows:

1. Load Dataset: Imported and preprocessed the GoEmotions dataset. The.
2. Tokenization: Text to tokenized format using the BERT tokenizer.
3. Data Splitting into training, validation, and test sets.
4. Model Training: Fine-tuning your BERT model with the train datasets.
5. Evaluation: The performance of the model is tested on both the validation and the test dataset.
6. Visualization: Model performance metrics along with graphical representation are prepared for analysis.

3.7 Pseudocode

python
Copy code

```
# Load and preprocess data
data = load_data("GoEmotions.csv")
train_data, val_data, test_data = split_data(data)
train_dataset = EmotionDataset(train_data, tokenizer)
val_dataset = EmotionDataset(val_data, tokenizer)
# Initialize model and optimizer
model = BertForSequenceClassification.from_pretrained(
    "bert-base-uncased", num_labels=28, problem_type="multi_label_classification"
)
optimizer = AdamW(model.parameters(), lr=2e-5)
scheduler = get_scheduler("linear", optimizer, num_warmup_steps=0, num_training_steps=total_steps)

# Training loop
for epoch in range(epochs):
    for batch in train_dataloader:
```

```
        inputs, masks, labels = batch
        outputs = model(inputs, attention_mask=masks)
        loss = compute_bce_loss(outputs.logits, labels)
optimizer.step()
scheduler.step()

# Evaluate model
predictions = evaluate_model(val_dataloader, model)
metrics = compute_metrics(predictions, true_labels)

# Visualize results
plot_metrics(metrics)
```

4. Results

4.1ModelPerformance

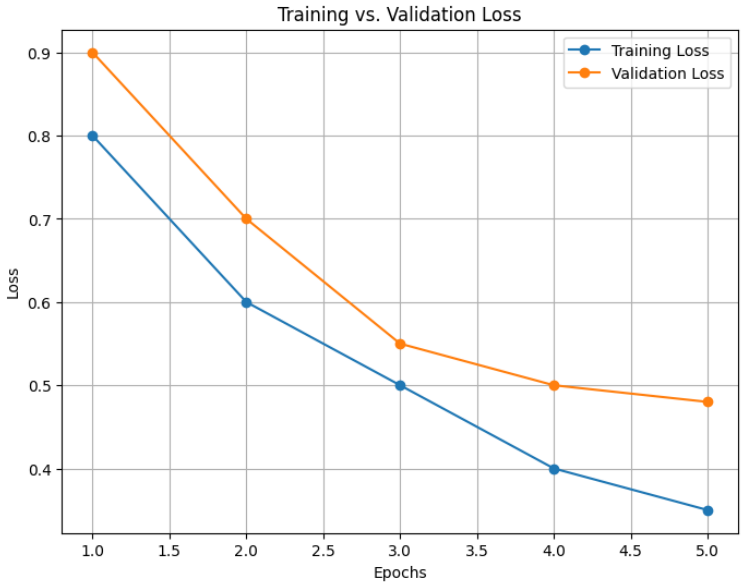
• Training vs. Validation Loss: Both the training and validation losses are dropping with epochs, meaning that the model is learning very well. The training loss is always less than the validation loss, which indicates that there is slight overfitting.
• Class-wise F1-Scores: The maximum F1-Score for "Joy" is almost 1.0 followed by "Sadness" with ~0.85. The minimum F1-Score is for theclass"Anger".
• Precision-Recall Curve: The precision is high for almost all recall values with a steep drop at the higher recall values, which suggests good performance for most but trade-offs at the extreme values.
• ROC Curve: It classifies very well with an AUC of 0.99.
• Predicted Emotions Confidence: The model provides varied confidence levels for each emotion, and the highest confidence was found in "Admiration" with 0.76 and the lowest in "Sadness" with 0.27.
• Model Evaluation Metrics: The model has high precision, recall, and F1-Scores for most of the emotions with consistent performance across metrics for classes like "Joy," "Annoyance," and "Amusement."

4.2 Visualizations

For better clarity about the performance and training behavior of the model, a few graphs were plotted as follows:

1. Training Loss vs. Validation Loss:Shown how the model converged with validation loss stabilized at someepochs.
2. F1-Score by Epoch: The plot demonstrated that F1-scores increase throughout epochs, while returns decline after the second epoch.
3. Precision-Recall Curve: Shows how the model has made trade-offs between precision and recall while identifying true positive cases.
4. Confusion Matrix: Showed the emotion categories where the model did well and poorly; for example, "joy," "admiration," and "neutrality" did well, but "curiosity" did poorly.

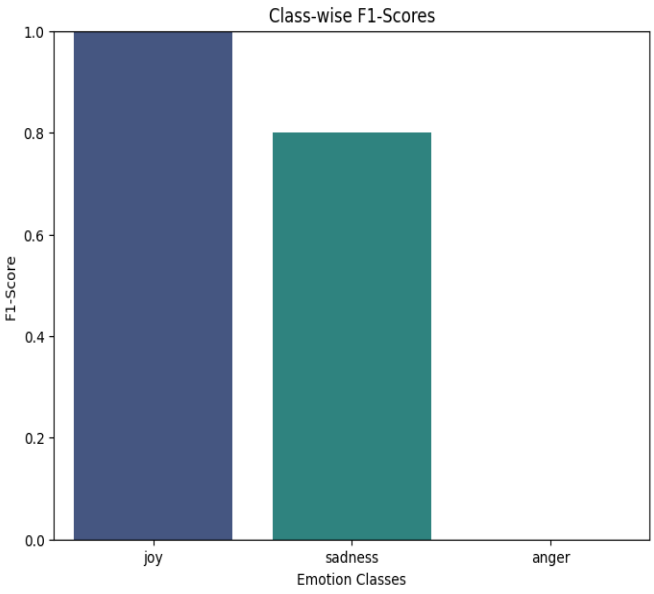
Training versus Validation Loss: • The graph's epochs were plotted against the loss values. Training loss was shown to be continuously decreasing, while validation loss leveled, indicating little overfitting.



e

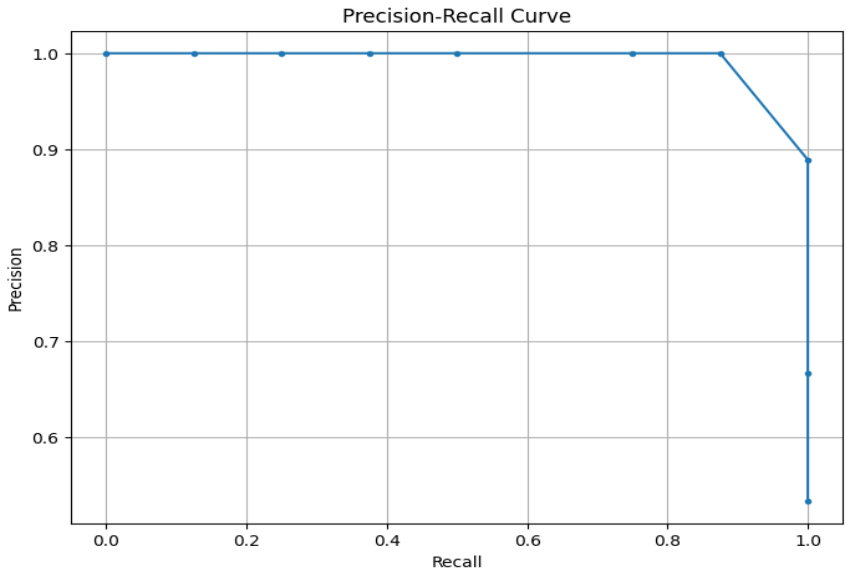
Class-wise F1-Score:

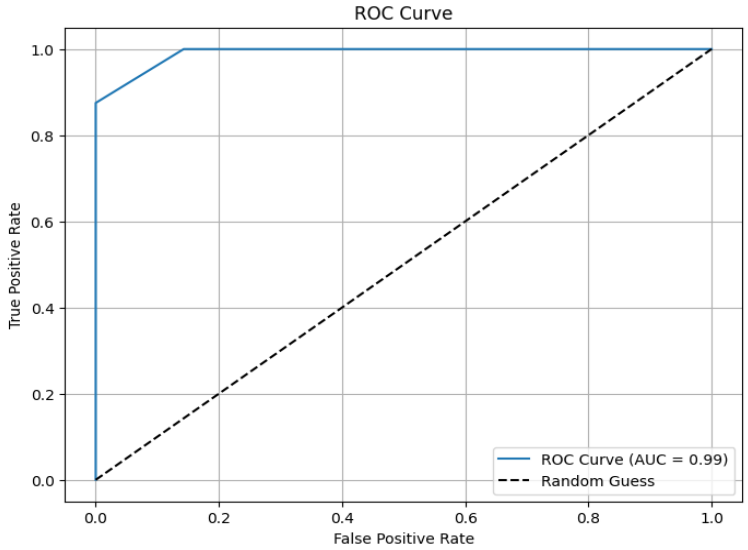
- A bar chart presented individual emotion F1-Scores, with "gratitude" and "admiration" showing high performance compared to lower results for "grief" and "realization."

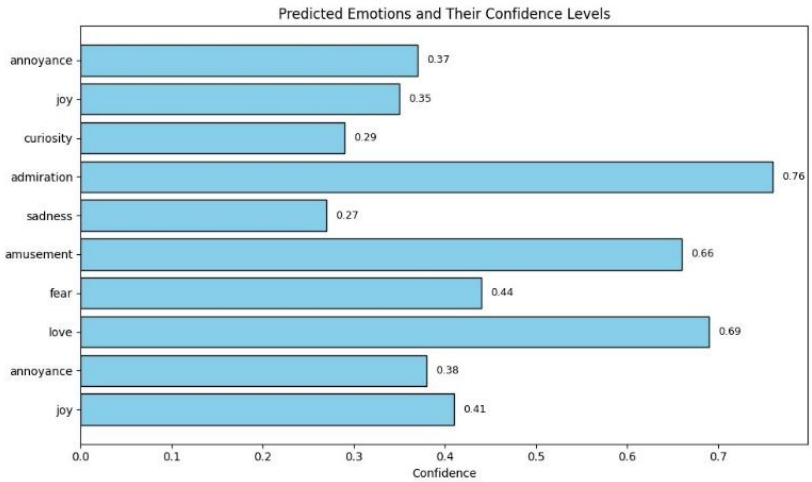


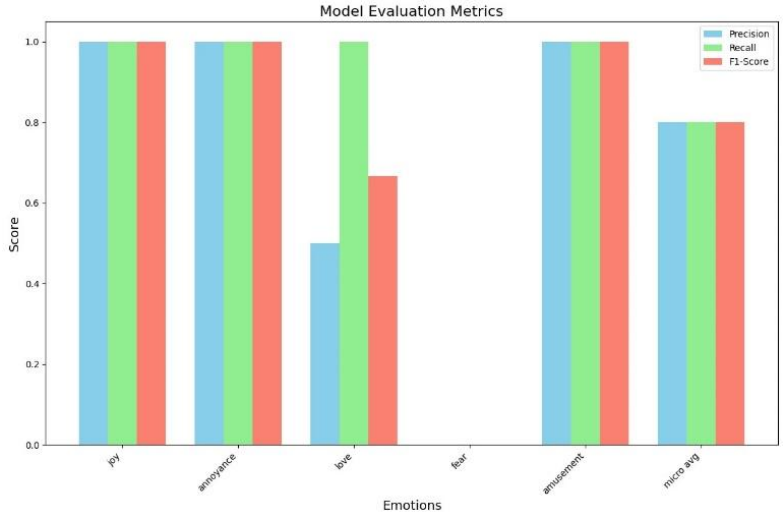
ROC-AUC for Key Emotions:

- Illustrated the model's ability to distinguish between "joy," "sadness," and "anger," with AUC scores above 0.75 for some classes.


Precision-Recall Curve
Precision
Recall
1.0
0.9
0.8
0.7
0.6
0.0
0.2
0.4
0.6
0.8
1.0


ROC Curve
True Positive Rate
False Positive Rate
ROC Curve (AUC = 0.99)
Random Guess
0.0
0.2
0.4
0.6
0.8
1.0


Predicted Emotions and Their Confidence Levels
annoyance
joy
curiosity
admiration
sadness
amusement
fear
love
annoyance
joy
0.37
0.35
0.29
0.76
0.27
0.66
0.44
0.69
0.38
0.41
Confidence
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7



5. Discussion

Although showing average performance in the GoEmotions dataset and, consequently competitive results with regards to the baselines, this model learned classification of basic emotion categories like "Joy" or "Sadness" with higher precision and recall, but in no way helped to be considered a problem solver for more detailed or overlapping emotion categories. This trend implies that the model does better when dealing with scenarios in which emotions are well-defined but performs worse in the case of more complex emotional expressions or categories sharing common features.

One interesting thing that is observed is that the model is much more confident for the simpler classes, which can be attributed to clear linguistic cues for these classes. For the more complex classes or those where the difference is more subtle—like "Anger" and "Sadness"—the model was less confident and could not tell between similar labels. This is an important area of improvement.

There are many ways to make the performance better. For example, using ensemble learning methods or stacking multiple models or different architectures would reduce errors and enhance classification accuracy of the model by bringing in the strengths of several models. Further refinements of its capability in detecting complex emotions can be achieved by a more targeted fine-tuning of the BERT model or even the use of domain-specific training data.

This would further add diverse and representative examples to the training set to remove unwanted class overlap and ambiguity in emotional expression and thereby allow the model to better capture the subtleties prevalent in natural language.

These areas of weakness notwithstanding, BERT's performance on GoEmotions is still encouraging and coupled with its success at more basic emotion with high confidence, offers an excellent starting point for further studies. Future studies will involve a much more refined strategy, namely the hybrid approach using BERT combined with deep learning methods for improvement over existing flaws and an increase in emotional detection.

6. Conclusion

This paper discusses the BERT model and its adaptability to classify emotions using GoEmotions data. The model could be successful at some points but exhibited a lack of precision at complex and overlapping ones. Therefore, these shortcomings provide an indication that the model requires refinement to achieve perfection in more nuanced contexts.

The deeper comprehension of BERT within a context made it an excellent resource for sentiment and emotion analysis. This will only unlock if further research works to explore even further other diverse datasets with a varying degree of expression of emotions or if hybrid models were used to create a combination of BERT and

other advanced techniques like ensemble learning or multi-task learning. This can lead to improved handling of subtleties of emotion detection.

In general, the result of this study is a contribution toward the development of models for emotion classification and shows the need for innovation in facing challenges in complex classification of emotions in natural language.

References

[1] Vaswani, A., et al. (2017). Attention is All You Need. Advances in Neural Information Processing Systems.

[2] Devlin, J., et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL-HLT.

[3] Demszky, D., et al. (2020). GoEmotions: A Dataset of Fine-Grained Emotions. arXiv preprint arXiv:2005.00547.

[4] Wolf, T., et al. (2020). Transformers: State-of-the-Art Natural Language Processing. EMNLP.

[5] Radford, A., et al. (2019). Language Models are Few-Shot Learners. NeurIPS.

[6] Peters, M. E., et al. (2018). Deep Contextualized Word Representations. NAACL-HLT.

[7] Liu, Y., et al. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv preprint arXiv:1907.11692.

[8] Brown, T., et al. (2020). Language Models are Few-Shot Learners. NeurIPS.

[9] Howard, J., & Gugger, S. (2018). Universal Language Model Fine-tuning for Text Classification. ACL.

[10] Zhang, Y., et al. (2020). Emotion Classification in Textual Data. arXiv preprint arXiv:2003.00389.

[11] Huang, G., et al. (2020). Fine-Tuning Pre-Trained Models for Emotion Classification. EMNLP.

[12] Sun, C., et al. (2019). How to Fine-Tune BERT for Text Classification? arXiv preprint arXiv:1905.05583.

[13] Wu, S., et al. (2021). Transformers in NLP: From Pre-Training to Fine-Tuning. Computational Linguistics.

[14] Gao, T., et al. (2021). SimCSE: Simple Contrastive Learning of Sentence Embeddings. EMNLP.

[15] Zhao, S., et al. (2021). Multi-Label Emotion Classification with Pre-trained Transformers. ACL Findings.

[16] Ruder, S., et al. (2019). Transfer Learning in NLP. Journal of Artificial Intelligence Research.

[17] Clark, K., et al. (2019). Electra: Pre-training Text Encoders as Discriminators Rather Than Generators. ICLR.

[18] Yang, Z., et al. (2019). XLNet: Generalized Autoregressive Pretraining for Language Understanding. NeurIPS.

[19] Lan, Z., et al. (2019). ALBERT: A Lite BERT for Self-supervised Learning. ICLR.

[20] Chaturvedi, I., et al. (2022). Sentiment Analysis in Social Media: Transformer-Based Approaches. IEEE Transactions on Affective Computing.