NLP Final Project Report: Emotion Detection in Text

Peter Charles Bailey (Charlie)

CU ID: peba2926

Abstract

Emotion is one of the crucial human feedback mechanisms that is difficult to transmit with it's full potential in the written form. This project, demonstrates the efficacy of a **fine-tuned BERT model** at detecting and distinguishing between six core human emotions (sadness, joy, love, anger, fear, and surprise) in textual data. The BERT-base-uncased model was fine-tuned using the **dair-ai/emotion** dataset from Hugging Face. The model **achieved an accuracy 93.3% and an F1 score 0.932**, confirming that BERT-based emotion detection in text is highly accurate. These results serve as a proof-fo-concept, suggesting that social media platforms could leverage this technology to bridge digital communication gaps and reduce misinterpretation in our increasingly polarized world.

Introduction

The primary problem addressed by this project is the difficulty of accurately interpreting nuanced human emotions conveyed in written text, an essential yet challenging task given the lack of non-verbal cues. This problem is especially relevant as our world becomes increasingly digital and polarized. Misunderstandings in text-based communication frequently lead to unnecessary conflict and reduced social cohesion. The goal of this project is to fine-tune a BERT model to reliably classify textual content according to six core human emotions (sadness, joy, love, anger, fear and surprise), aiming to achieve a high accuracy rate, thus demonstrating the efficacy of this method in real-world implementations. Ultimately, the ability to better convey and decode emotions via text could significantly improve digital interactions and reduce communication gaps.

Related Work

Transformer-based models, especially BERT, have significantly advanced text-based emotion detection. Acheampong et al. (2021) provided a comprehensive survey of multiple studies that utilized a BERT architecture, highlighting their effectiveness in emotion recognition tasks. Among the highlighted works, Huang et al. (2019a) proposed a contextual emotion classifier (EmotionX-KU). Their model consisted of three main phases: casual utterance modeling, model pre-training using BERT, and fine-tuning. They used the EmotionLines and Friends datasets and obtained micro-F1 scores of 81.5 and 88.5 on each dataset respectively.

In a separate study, Al-Omari et al. (2020) presented the EmoDet2 system, an ensemble model containing BERT and BiLSTM components, which demonstrated robust performance on dialogue-based emotion detection. Similarly, Abas et al. (2022) proposed a BERT-CNN hybrid model that significantly outperformed baseline methods on benchmark datasets. Additionally, Demszky et al. (2020) introduced GoEmotions, a carefully curated and annotated large-scale emotion dataset, which provides a reliable benchmark for fine-grained emotion detection research.

Collectively, these works illustrate the effectiveness of BERT-based approaches for accurately classifying nuanced emotions in textual data. This reinforces their suitability for implementation in a digital communication context.

Data

Description

For this project, I will be using the **dair-ai/emotion** dataset from Hugging Face. There are two configurations of this dataset available—a split version (with predefined subsets) and unsplit version.

Name	Train	Validation	Test
split	16,000	2,000	2,000
unsplit	416,809	N/A	N/A

I will utilize the `split` configuration, using the `train` and `validation` subsets for fine-tuning and evaluating final results on the `test` subset.

Below is an example of the dataset structure:

	text	label
0	i didnt feel humiliated	0
1	i can go from feeling so hopeless to so damned...	0
2	im grabbing a minute to post i feel greedy wrong	3
3	i am ever feeling nostalgic about the fireplac...	2
4	i am feeling grouchy	3
5	ive been feeling a little burdened lately wasn...	0
6	ive been taking or milligrams or times recomme...	5
7	i feel as confused about life as a teenager or...	4
8	i have been with petronas for years i feel tha...	1
9	i feel romantic too	2

Both versions of this dataset have two features:

1. a text string derived from English Twitter messages
2. a numeric classification label with the following possible values: `sadness (0), joy (1), love (2), anger (3), fear (4), surprise (5)`

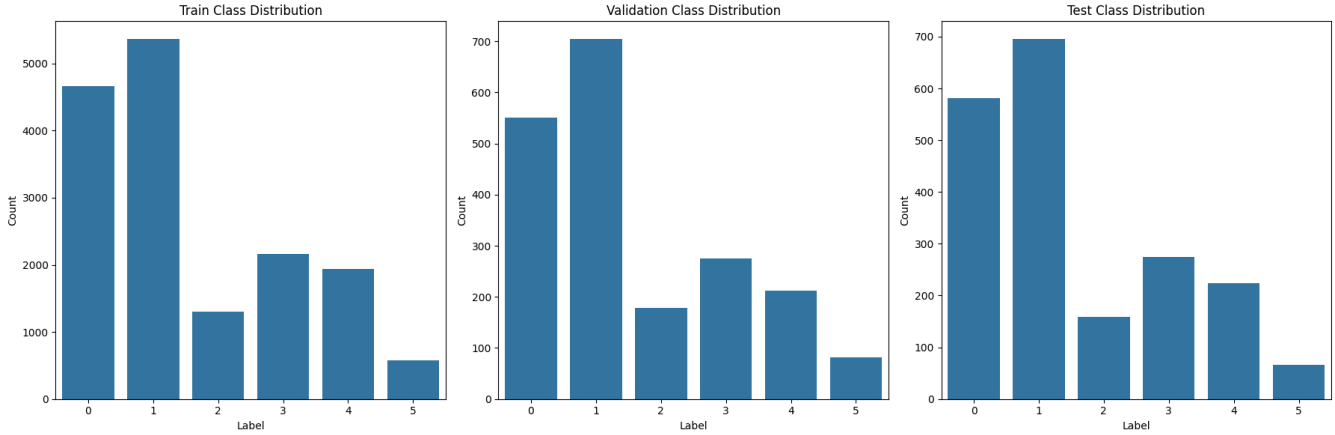
As noted above, the `split` configuration has 16,000 train instances, 2,000 validation instances,and 2,000 test instances—for a total of 20,000 instances overall.

EDA

For the most part, this dataset is cleaned, in the right format, and ready for processing. There is a single duplicate value, but given the size of the dataset, this will have in indistinguishable impact on performance and will therefore be retained.

However, one important point to note are the class imbalances (see below). While there is a similar distribution pattern across all three datasets, within each dataset we see that classes 0 and 1 are heavily overrepresented and class 5 in particular is very underrepresented.

These imbalances will need to be addressed carefully in the model training and evaluation phases to ensure reliable and unbiased performance.



Methodology

Tech Stack

- **PyTorch** framework for running the BERT model and providing integration with my local GPUs.
- **Transformers** library from Hugging Face to utilize the pre-trained BERT models for fine-tuning.
- **Datasets** library from Hugging Face for loading the data into my program.
- **Pandas** library for EDA and data manipulation.
- **Seaborn/Matplotlib** libraries for generating data visualizations.
- **NumPy** package for dealing with vectorized data.
- **Sklearn** library for model evaluation statistics.
- **tqdm** library to easily visualize model training progress.

Approach

I started this project by doing some preliminary EDA to get a better understanding of the dataset I would be working with and ensure it was properly formatted. During EDA, I discovered significant class imbalances across the six emotion categories. To address this issue, I decided to use a **weighted F1 score** as one of the primary metrics in model evaluation to account for these imbalances.

After the initial EDA, I began working on my main BERT fine-tuning workflow. The development of this portion of the project was heavily influenced by the Hugging Face LLM course Chapter 3 on model fine-tuning.

To setup my fine-tuning pipeline, I utilized the `Datasets` library to load in the split version of the dair-ai/emotion dataset. I then initialized the `BertTokenizer` from the `bert-base-uncased` checkpoint. I also initialized the `BertForSequenceClassification` model utilizing the same checkpoint. Finally, I used a data collator to handle dynamically padding inputs of different lengths during training. I had initially written an entire function to determine the max input size after tokenization, but after doing some further research, I realized that Hugging Face had already solved this problem for me!

Once I had all of the core components figured out, I moved on to the main training function. In previous AI/ML projects, I have used PyTorch to move my models onto my Mac M3 GPU for faster training. I was initially planning to do something similar for this project and build using PyTorch, but I realized that Hugging Face had again made everything vastly easier with their `TrainingArguments` class. By default, `TrainingArguments` will utilize my `mps` device—so there is no need to manually move the model. Since the dataset I'm using is relatively small, I decided to use the 'epoch' evaluation strategy rather than 'steps'.

From here, the last component I needed before running the model was the evaluation criteria. For this, I built the `compute_metrics` method. As previously mentioned, I decide to evaluate my model using **accuracy** as well as a **weighted F1 score**. I decided to use the weighted F1 score to evaluate the model while still accounting for the class imbalances.

With these components in place, the model was ready to be trained and evaluated.

Results

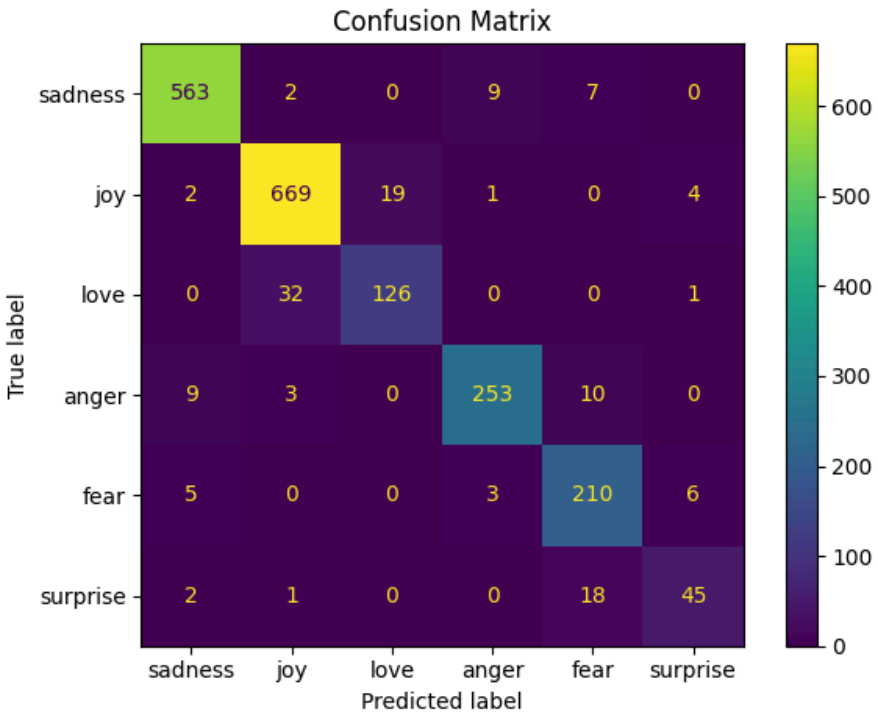
Using the pipeline described above I was able to train the model using my M3 GPU in ~20 minutes. I then evaluated my model on the `test` set of the dair-ai/emotions dataset and obtained the following results:

Accuracy: 93.3%

Weighted F1 Score: 0.932

These results are pretty impressive given that there was very little hyperparameter tuning applied to the model. It is also interesting to note that both the accuracy and weighted F1 score are similar, indicating that the class imbalance did not have a significant impact on classification performance.

Below we can see the confusion matrix for the final checkpoint:



It is interesting to note here that the bulk of the misclassification appears to be between fear/surprise and joy/love. This intuitively makes sense because these are similar emotions that can be difficult to distinguish even for humans.

Discussion

Overall, I am very satisfied with the outcomes of this project. Most challenges arose from initially familiarizing myself with the Hugging Face ecosystem. On several occasions, I began developing custom implementations for pipeline components, only to discover existing Hugging Face solutions that were simpler and more efficient to use. However, each of these instances became valuable learning opportunities, significantly enhancing my proficiency with the Hugging Face Transformers library.

Conclusion

In conclusion, this project serves as a proof-of-concept that BERT-based models can serve as useful tools in performing emotion detection in textual data. Utilizing a BERT model with minimal hyperparameter tuning, I was able to achieve 93.3% accuracy and a weighted F1 score 0.932 on text data from social media posts.

To extend this project in the future, I would like to perform some additional hyperparameter tuning on this model, then scale to test it on the full `unsplit` dataset. From there, I could implement a small web-app that would allow users to classify emotions in short snippets of text.

Bibliography

Abas, A. R., Elhenawy, I., Zidan, M., & Othman, M. (2022). BERT-CNN: A deep learning model for detecting emotions from text. Computers, Materials & Continua, 71(2), 2943–2961

Acheampong, F. A., Nunoo-Mensah, H., & Chen, W. (2021). Transformer models for text-based emotion detection: A review of BERT-based approaches. Artificial Intelligence Review, 54(8), 5789–5829.

Al-Omari, H., Abdullah, M. A., & Shaikh, S. (2020). EmoDet2: Emotion detection in English textual dialogue using BERT and BiLSTM models. In 2020 11th International Conference on Information and Communication Systems (ICICS).

Demszky, D., Movshovitz-Attias, D., Ko, J., Cowen, A., Nemade, G., & Ravi, S. (2020). GoEmotions: A Dataset of Fine-Grained Emotions. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL).

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of NAACL-HLT 2019, pages 4171–4186.