Multi-label Emotion Classification of Movie Subtitles

Prathyusha Charagondla
pcharagondla@berkeley.edu

Ziwei Zhao
ziweiz@berkeley.edu

Abstract

Multi-label, multi-class emotion classification is an underdeveloped area of NLP. We take the XED data set, which contains 17,530 movie subtitles labeled one or more of Plutchik's eight emotions, and use CNN, LSTM, and BERT models for multi-label classification. In our experiments, we explored the BERT without the CLS token, averaging the vectors from the 'last_hidden_state', and the RoBERTa model. Our RoBERTa model achieves the best micro f1-score of 0.548, slightly surpassing the original paper's 0.536 (Öhman et al., 2020). We explore the differences between the model predictions and the true labels and discuss potential future steps for improving multi-label emotion classification.

1 Introduction

While binary (positive or negative) or ternary (including neutral) sentiment analysis have been well explored in natural language processing since the early 2000s (Mäntylä et al., 2016)), there remains much room for development in the field of multi-label emotion classification.

Conventionally, a lot of sentiment analysis use product or movie reviews and tweets, mostly self-contained short form text that does not depend on other context (Öhman et al., 2020). We are interested in exploring emotion classification in a more complete story that involve multiple characters, especially a movie. In the film industry, there is a lack of representation of women and minority voices, both in the number of lead characters and the lack of emotional complexity of existing characters. Improving the ability for machines to classify emotions in text could help with script analysis to ensure that underrepresented characters are fully fleshed and experience a range of emotions.

Emotion data sets require human annotation and are difficult to create because the more categories of emotions there are, the more likely it is for different annotators to disagree. Moreover, emotions can be difficult to detect for humans, especially to separate distinct but easily conflated emotions such as disgust and anger. We were not able to find complete movie scripts with emotions annotated, which would best suit our analysis. Instead, we use the English portion of the XED dataset, a multilingual dataset with movie subtitles, each with one or more of eight emotion labels. We attempt to improve the highest micro f1-score of 0.536 for 17,528 English subtitles (Öhman et al., 2020). We experiment with CNN, LSTM, and transformer models BERT and RoBERTa to achieve an micro f1-score of 0.548, and explore ideas for further future improvements.

2 Related Work

A typical emotion dataset might contain 6-8 emotions. The XED dataset that we will be using consists of the 8 primary emotions from the psychologist, Robert Plutchik's theory of Wheel of Emotions, anger, anticipation, disgust, fear, joy, sadness, surprise and trust. The most categories we have seen is 27, in Demszky et al. (2020), which includes emotions such as approval, realization, confusion and grief. In this study, the authors created GoEmotions, the largest manually annotated dataset of 58k English Reddit comments. The highest F1 score achieved is 0.46 using a BERT model.

Huang et al. (2019) uses a bidirectional encoder-decoder system called Seq2Emo model that contains modules such as ELMo and DeepMoji that is able to predict multiple emotion labels on one tweet. They achieve a micro F1 score of 70.9%. Zhou et al. (2020) uses the same Twitter dataset SemEval2018 and achieves a micro F1 score of 71.6% by using a proposed model EmNet, an encoder-decoder structures that can learn word emotions along with the context that the word is in. Tweets

Sentence	Label
Such a beautiful bride .	joy
I regret the things that I done .	disgust, sadness
He should be part of our celebration .	trust, anticipation
Don't you know me ?	fear, surprise
For the love of God , [PERSON] . !	anger

Table 1: Examples from data with respective labels

Label	Emotion	Count	Percentage
0	anger	3,828	17.1
1	anticipation	3,400	15.2
2	disgust	2,317	10.3
3	fear	2,439	10.9
4	joy	2,833	12.6
5	sadness	2,464	11.0
6	surprise	2,442	10.9
7	trust	2,699	12.0

Table 2: Label mapping and frequency

are commonly used in sentiment and emotion analysis, but because of their length limit and the use of hashtags and emojis that help with prediction, the results tend to not be as transferable to other text including movie scripts (Öhman et al., 2020).

3 Data

We are using a data set named XED that contains 17,530 unique movie subtitles from the OpenSubtitles Corpus (Helsinki-NLP). Each subtitle is labelled with one or more of the eight core emotions from Plutchik's theory - anger, anticipation, disgust, fear, joy, sadness, surprise and trust. They were labelled by over 100 university student annotators with minimal instructions and without any context for where in the movie or even which movie the quote originated from (Öhman et al., 2020). The data had been pre-processed to include a space between words and punctuation, and names of people and places are replaced with '[PERSON]' and '[LOCATION]' tokens. Table 1 shows a few example sentences and their labels from the data.

Our aim is to explore different classifiers and parameters to build a classifier with the highest micro f1 score. In our primary exploration of the data set, we found that the data was not equally distributed across the different categories (Table 2), with anger most represented at 17.1% and disgust least represented at 10.3%.

We split the data into train, dev, and test sets using a ratio of 60%, 20%, and 20%.

4 Methods

4.1 Convolutional Neural Network (CNN)

As we are working with multi-class text classification, we use the Convolutional Neural Network (CNN) as our baseline model. First, we tokenize the data using Keras' tokenizer and convert them to sequences and pad them, setting max length to 100. Then, we trained our CNN using Keras. The model consists of 3 ”filters,” each comprising of a Keras 1D convolution layer and a Global Max Pooling operation, a Keras dropout layer and 2 Dense layers with rectified linear lnit (reLU) activation function, followed by an output layer with a sigmoid activation function, so that each of the eight labels can independently be predicted. The threshold is set at 0.5, so a label will be predicted if its probability from the sigmoid function in the last layer is greater than 0.5.

After building our initial CNN, we tuned our parameters. In order to optimize our hyper-parameter tuning, we evaluate both the grid search and random search methods. In the end, we decided to continue using Random Search to optimize our hyper-parameters, as it tends to find better models in most cases and requires less computational power as well (Bergstra and Bengio, 2012). The hyper-parameters that we've explored are number of filters, the kernel sizes for each of these filters, number of epochs and dropout rate. We built the CNN randomizing on select parameters, while holding all other variables constant. The final CNN we used was trained for 11 epochs at a dropout rate of 0.7 with number of filters 12, 40, 37, each of sizes 7, 9, 7 respectively. In parameter tuning, we find that 11 epochs seems to result in the best classifier, as we added epochs the model seemed to overfit on the training data.

4.2 Long Short-Term Memory (LSTM)

Working off of our base CNN model, we explore the Long Short-Term Memory (LSTM). Using the tokenized input data from the CNN, we train our LSTM using Keras. We try different structures of LSTMs, including bidirectional LSTM. The structure we settle on is the bidirectional LSTM with a bidirectional LSTM, a dropout layer, followed by an output layer with a sigmoid activation function. Similar to the CNN, we used random search to optimize our hyper-parameter tuning. The parameters we've explored are epochs and dropout rate. The final LSTM we used was trained for 10 epochs at a dropout rate of 0.5.

4.3 BERT

4.3.1 Base Model

Next we turn to explore BERT, which is used by the authors of the original XED paper to achieve their best results. We use the pre-trained 'bert-base-uncased' model from Huggingface, tokenizing sentences in the data using BertTokenizerFast, and add one Keras dropout layer and one Keras dense layer on top of the BERT layer for classification. The max length of tokenized sentences is set at 100. We use an Adam optimizer with various learning rates and batch sizes of 32 and 64 to optimize the BERT model. We use the binary cross entropy loss function and sigmoid activation function in the last layer.

4.3.2 Experiments

BERT without CLS To find potential improvements in the BERT model, we conduct a few experiments. First, we still use 'bert-base-uncased' as the model, but instead of using the default CLS token for classification, we keep 'last_hidden_state' from the BERT layer, which contains all vectors that represent each token of the inputs. In our case, for one input sentence, this would be 100 vectors, each with a length 768. Then we take the average of these 100 vectors to represent the inputs using an added Keras GlobalAveragePooling1D layer before feeding into the dropout layer and final layer for classification.

RoBERTa Next we try the model RoBERTa, or "Robustly Optimized BERT", which was trained with more data and longer sequences. It also removed next sentence prediction as an objective during training and used dynamic word masking. (Liu et al., 2019). We use the pre-trained 'roberta-base' model and tokenizer RobertaTokenizerFast, while keeping other model parameters the same as out base BERT model. We also use the default CLS token for classification.

5 Results

5.1 Scores

The main metrics we use to measure model performance are micro f1-score and accuracy calculated as hamming score, which is the ratio between the length of the intersection of true and predicted labels and the union of the two. The score penalizes predictions that are incorrect even if it captures all the correct labels as well (Read and Hollmén, 2017). We discuss the scores from each model below, and they are summarized in Table 3.

Classifier	Micro f1	Accuracy
CNN	0.298	0.226
LSTM	0.340	0.276
BERT-CLS	0.513	0.462
BERT-Avg	0.514	0.447
RoBERTa	0.543	0.500
RoBERTa-filled	0.548	0.525

Table 3: Results by Classifiers

The first model, our initial base model resulted in a low accuracy score, however through hyper-parameter tuning, we find that the model trained for 11 epochs at a dropout rate of 0.7 with number of filters 12, 40, 37, each of sizes 7, 9, 7 respectively seemed to perform the best with an micro F1 score of 0.298, macro F1 score of 0.280 and an accuracy score of 0.226. In this model, of the 8 emotion labels, surprise has the lowest F1 score of 0.121 and joy has the highest F1 score of 0.490.

Using the CNN as the baseline to beat, we trained an LSTM model. Through hyper-parameter tuning, we find that the model trained for 10 epochs at a dropout rate of 0.7 seemed to perform the best with an micro F1 score of 0.340, macro F1 score of 0.325 and an accuracy score of 0.276. In this model, of the 8 emotion labels, disgust has the lowest F1 score of 0.208 and joy has the highest F1 score of 0.504.

In the base BERT model, a learning rate of 5e-5 and batch size of 64 gives an accuracy of 0.462 and a micro-f1 score of 0.513. For individual labels, disgust has the lowest f1-score at 0.403 and joy has the highest at 0.643. By replacing the CLS token with the average of the sentence tokens from the last hidden state and using the same learning rate and batch size, the micro f1-score increases only slightly to 0.514 while accuracy decreases to 0.447. Disgust and joy still have the lowest and highest individual f1-scores at 0.417 and 0.652.

Using RoBERTa with a learning rate of 5e-5 and batch size of 64 gives us better results than all previous classifiers. The micro f1-score is 0.543 and the accuracy is 0.500. Disgust still has the lower f1-score, but it is improved drastically to 0.489. Joy has the highest f1-score at 0.657.

By our model construction, a label is only predicted if it has a probability of more than 0.5, so it is possible for some inputs to have no labels predicted. This occurs for 9.7% of the test data. Since

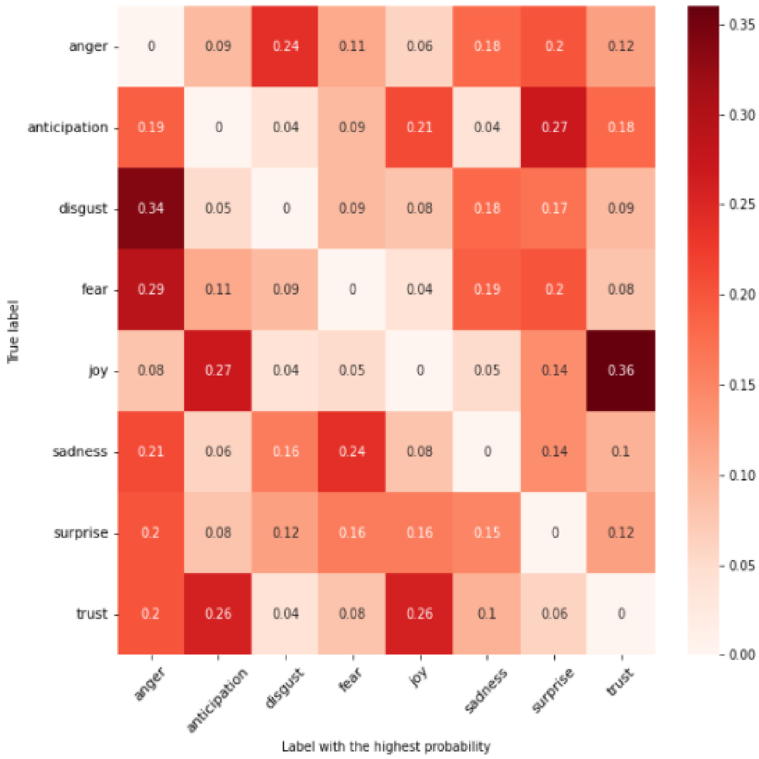

Figure 1: Confusion matrix - test data that failed to correctly predict true label(s)

all input sentences in the XED data set have labels, we assign the one label with the highest probability of the eight probabilities for the 9.7% data, even if the estimate is less than 0.5. We call this the ”RoBERTa-filled” classifier and it increases the micro-f1 score to 0.548 and the accuracy to 0.525 - our best results.

5.2 Analysis

We use the predictions of our best classifier RoBERTa-filled to analyze errors. The major reason for mislabeling is due to the conflation of similar emotions such as anger and disgust or fear and sadness. This was expected as it is one of the major obstacles in the field of emotion classification. We created a customized confusion matrix to illustrate this. We analyze examples from the test data where there is one or more true labels that are not being predicted. For each such label, we identify the alternative label with the highest probability in the prediction for the input sentence.

For example, the sentence ”You’ve gone out of your mind .” has true label anger. Our model only estimates a probability of 0.208 for the label anger, therefore does not predict it. We look at the other seven probabilities and find that label with the highest probability is sadness. Therefore, the true label would be anger and the ”predicted label” would be sadness, placing this example in the row 1, column 6 of Figure 1, which indicates that out of all test data with the true label anger that did not correctly predict anger, 18% have sadness as the most likely label.

Notably, disgust and fear are most often mistaken as anger, joy is most often mistaken as trust and anticipation, and anticipation is most often mistaken with surprise.

While hopefully a more sophisticated model can distinguish these similar emotions, it is more understandable to see its occurrence than mislabels of opposite emotions like joy and sadness. One reason is that some short sentences contain a word with a strong sentiment without much context. Our model would predict an emotion that aligns with the key word, while a human eye can decipher the deeper meaning. For example, ”Better than yours .” has true labels anger and disgust, while our model predicts joy. ”Aren’t you satisfied ?” is labeled anger but predicted as joy. ”You must have loved your [PERSON] .” has labels sadness and trust, and our model predicts joy, potentially due to the presence of ”love”.

In some cases, while the model predictions don’t completely align with the true labels, it is debatable whether the predictions are actually incorrect. ”What sort of a man are you ?” is labeled as surprise and the prediction is disgust. ”Still no word from [PERSON] ?” is labeled as anticipation and fear but the prediction is surprise. In these cases, the predicted labels would be reasonable interpretations of the sentences.

The emotion surprise is tricky to identify because it can be both positive or negative and can often reasonably accompany the other emotions. Our model can predict one emotion correctly, but not all. For example, ”This one is getting too popular .” is labeled anger, disgust, and surprise, and our model predicts only disgust. In the original XED paper, the authors even used a model that categorized this data into positive, negative, and surprise as a ternary analysis, or dropped surprise altogether for a true binary model (Öhman et al., 2020).

6 Conclusion

While our best classifier manages to achieve a higher micro f1-score of 0.548 than the original paper’s 0.536 (Öhman et al., 2020), there is still room for improvement. We chose 0.5 as the probability threshold for classification, but this number can be further tuned. Also, to better assess the performance of the model, there could be a custom accuracy metric that weighs errors differently, penalizing predictions that are more emotionally distinct than the true label, like joy and disgust.

In the end, multi-label emotion classification remains a very challenging task. Unlike sentiment analysis, which also has room for human interpretation, emotions can be highly subjective. It is common for human annotators to disagree on the labels. This was made more difficult by the fact that the XED data set includes single line subtitles that have no context, making some sentences harder to decipher. For future studies, there needs to be more labeled emotion data, both in volume and the types of texts labeled. Specific for the studying of film characters’ emotional arcs, it might be helpful to have labeled data that include longer snippets per sample, even entire conversations.

References

James Bergstra and Yoshua Bengio. 2012. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(10):281–305.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. GoEmotions: A dataset of fine-grained emotions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4040–4054. Association for Computational Linguistics.

Helsinki-NLP. Helsinki-nlp/xed: Xed multilingual emotion datasets.

Chenyang Huang, Amine Trabelsi, Xuebin Qin, Nawshad Farruque, and Osmar R. Zaïane. 2019. Seq2emo for multi-label emotion classification based on latent variable chains transformation. *CoRR*, abs/1911.02147.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Mika Viking Mäntylä, Daniel Graziotin, and Miikka Kuutila. 2016. The evolution of sentiment analysis - A review of research topics, venues, and top cited papers. *CoRR*, abs/1612.01556.

Emily Öhman, Marc Pàmies, Kaisla Kajava, and Jörg Tiedemann. 2020. XED: A multilingual dataset for sentiment analysis and emotion detection. *CoRR*, abs/2011.01612.

Jesse Read and Jaakko Hollmén. 2017. Multi-label classification using labels as hidden nodes.

Deyu Zhou, Shuangzhi Wu, Qing Wang, Jun Xie, Zhaopeng Tu, and Mu Li. 2020. Emotion classification by jointly learning to lexiconize and classify. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3235–3245, Barcelona, Spain (Online). International Committee on Computational Linguistics.