

# **An in-Depth Error Analysis**

## **1. Introduction**

Understanding how people react to different moments in content is becoming a big part of how media and entertainment evolve. Being able to use AI to analyze sentences pulled from various sources of human language, whether it's a TV episode, a tweet, or a product review, it opens the door to smarter insights. To know the preferences of the audience, to know what moment in an episode is the most popular and attention-grabbing, the best moment for product placement, or how an audience is feeling in real time can shape how stories are told and how content is marketed. Training AI to recognize and classify emotions across different forms of media helps us better understand viewer engagement, customer feedback and social media trends, with the help of sentiment analysis and emotion classification.

The model subjected to this error analysis is the best iteration we trained, we specifically used the pre-trained transformer-based model RoBERTa, which is an improved version of BERT, designed for text-based classification, sentiment analysis and question answering, it's adept at natural language understanding. There were many other model types tested such as Naïve Bayes, LSTM's, RNN's but the model which yielded the best results ended up being the transformer based one, the same one being analyzed here.

The data used for training were some large language datasets compiled for the use of training, all of which with emotion labels attached to each sentence. The (1) first dataset, called Go Emotions, is a collection of reddit comments compiled into one large dataset, with a range of emotions as labels. The second dataset (2) consists of a Kaggle dataset we found while in search of datasets. It's an assortment of phrases intended for model training. The third dataset is the grouping of all of the outputs of the Content Intelligence Agency (CIA) pipeline from other groups except for our group's set, which is saved for testing and validation of the models training process.

The task of the model is to predict from a range of emotions (happiness, sadness, anger, surprise, disgust, fear + neutral) which one is the right classification for each sentence in the dataset. Let's test it.

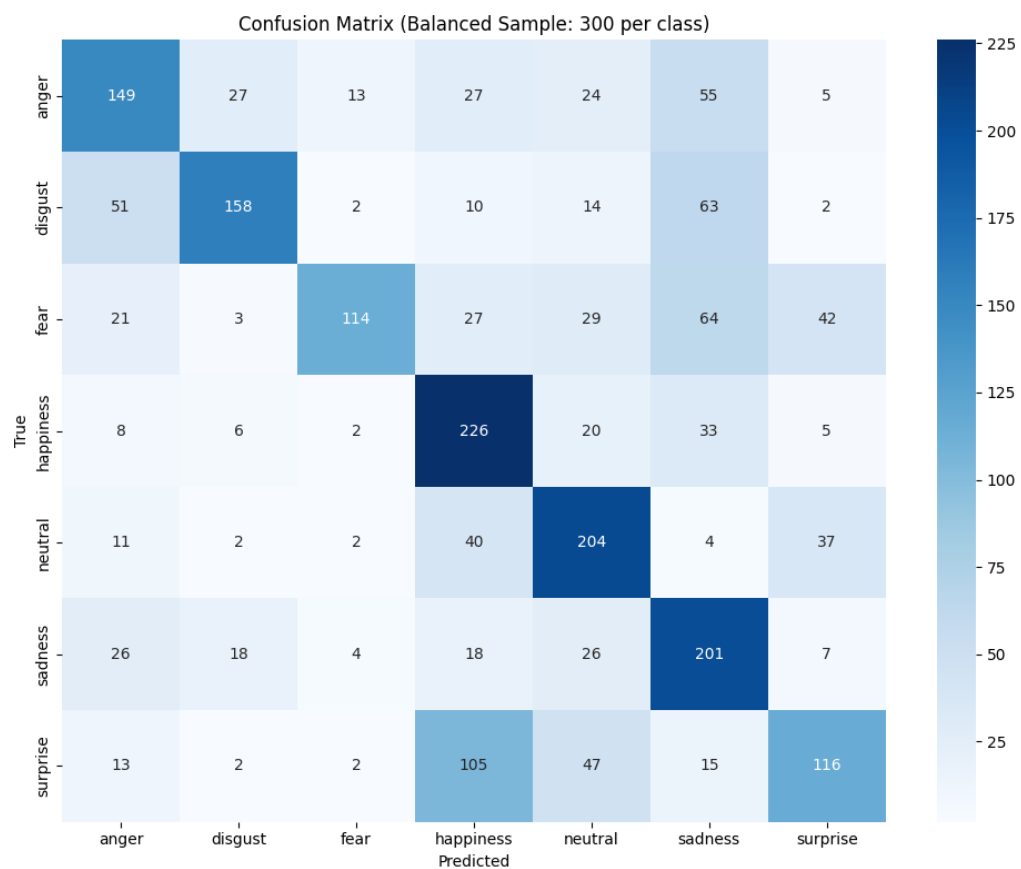
## **2. Performance Overview**

For the initial training evaluation, the highest F1 score recorded was .51 with its accuracy being about the same. After generating a classification report of the model, the results were the following:

Classification Report:				
	precision	recall	f1-score	support
anger	0.53	0.50	0.51	300
disgust	0.73	0.53	0.61	300
fear	0.82	0.38	0.52	300
happiness	0.50	0.75	0.60	300
neutral	0.56	0.68	0.61	300
sadness	0.46	0.67	0.55	300
surprise	0.54	0.39	0.45	300
accuracy			0.56	2100
macro avg	0.59	0.56	0.55	2100
weighted avg	0.59	0.56	0.55	2100

### 1. Classification report of the model

As you can see, after generating the classification report on a neutral unseen dataset, the weighted average f1 score across all labels was .55-.56 which is a slight bit higher than the original training score. And after equally distributing 300 samples to each label, all labels scored very similarly, the worst being surprise and the best being neutral and disgust.



### 2. Confusion matrix of the model

As for the confusion matrix, it follows the distribution that the classification report had, as the confusion matrix forms a nice diagonal line of true positives, but it's still showing many misclassifications done by the model.

### 3. Detailed Error Analysis

After doing analysis on a neutral test dataset using the model to predict the emotion of each sentence and comparing it to the ground truth, we tried to find any patterns that might be leading the model to misclassifying certain sentences from the dataset. We took 300 random samples from a new unseen neutral dataset and made the model predict each sample. Then we saved only the misclassifications made by the model with the ground truth to compare and then observed if any meaningful patterns emerged.

The class with the highest number of *False Positives* for the predictions made would be the **Happiness** class followed by the **Surprise** class. And the class with the highest number of *False Negatives* was by far the **Neutral** class then followed by the **Anger** class.

Class	False Negatives	False Positives
Neutral	60	24
Anger	29	15
Happiness	20	45
Sadness	11	11
Surprise	15	40
Disgust	7	8
Fear	4	3

#### 3. Table of False positives and negatives by class

'Neutral' has the most FN out of any other class, which was the expected outcome. 'Neutral' is inherently subtle and quite context-dependent so it's very easy for the model to confuse it with other more low-intensity emotions such as sadness or surprise. The class that was most mistaken for 'Neutral' was 'Happiness' followed by surprise.

Example: "What a sweet story."

This was predicted as 'Happiness', which is a prediction that has some backing to it. These types of phrases carry politeness or mild positivity which can end up confusing the model, this is especially the case if the training data does not have any differentiation between neutral politeness and real emotional positivity.

Another fact could be that classes such as anger, fear or disgust often contain emotionally charged words (e.g., "hate", "terrified", "gross"). 'Neutral' lacks such strong indicators, which is seen in the true cases from the used dataset. Due to these factors the model may struggle to learn the subtlety of 'Neutral' or it might default to a more distinct emotion if it's under any uncertainty.

Another pattern found was that the model struggled with short and ambiguous inputs such as “No!” which was predicted as ‘Surprise’ instead of ‘Anger’ or “Oh well” which was predicted as ‘Happiness’ instead of ‘Neutral’. These very short texts lack enough context for the model to accurately give a good prediction for them. Such examples can express multiple emotions depending on tone and situation. This goes into the next found point, which is contextual dependence. There are some sentences that depend heavily on context and due to the lack of it, the model can’t generalize well for them as there can be many meanings for one sentence.

Example: “*Why? Did you write it?*”

This sentence is vague on its own, but the tone could swing toward curiosity (surprise), sarcasm (disgust), or neutral.

Example: “*I don’t wanna speak, I don’t wanna think.*”

This sentence is also vague in nature but has hints of negative feelings such as sadness. Since the model can’t access dialogue history, it must guess based on patterns it’s seen before, leading to errors.

There have also been many emotional overlaps between categories. For example, ‘Happiness’ vs ‘Neutral’, many misclassified examples show the model confusing politeness, gratitude or admiration for happiness. The same case happens with ‘Surprise’ vs ‘Anger/Fear’. Short exclamations or rhetorical questions can resemble shock or frustration leading to errors.

Example: “*Thanks for coming in today. I appreciate your time.*”

The model predicted ‘Happiness’ for this one, but the true label is ‘Neutral’. The sentence is formal and polite, not expressing much emotion but the model sees “thanks” and “appreciate” and then leans towards ‘Happiness’ as the predicted class.

There was also a correlation between a sample sentence being lengthy and the model misclassifying it as neutral.

Example: “*My Dad says if I spend as much time helping him clean apartments, as I do daydreaming about outer space, he’d be able to afford a trip to the Taj Mahal.*”

The true class is ‘Neutral’, but the predicted class was ‘Happiness’. In this case the issue could be that the sentence length is too large for the model to accurately get a proper prediction, getting confused with all of the different elements present in a single sample, leading it to predict it as ‘Happiness’.

What we think is the main reason behind our model having a mediocre F1 score of .51 is due to poor data quality, specifically for the testing and validation set that was given to us. The data is the output from the Content Intelligence Agency, which originally was an episode from the show “La isla de la tentaciones” which is a Spanish reality TV show, which then

was ran through the CIA pipeline, which returns the translated and labeled data. On initial review a lot of the labels were misclassified and wrong which is not good for validating the actual model, and the translations in cases seemed to be quite off. In fairness to the CIA, the chosen episode is a very emotion packed rollercoaster throughout the entire runtime. People were shouting all the time, they were cutting each other off talking over each other, spoken words and sentences seemed unintelligible at times possibly making it hard for the pipeline to accurately transcribe and classify each thing being said. But due to this quality discrepancy the model performance fell short, even after applying many pre-processing steps to the dataset, manually changing labels, using features for the models, using lexicons, switching between multiple types of models and training data, it all culminated to a maximum F1 score of .51.

To see how good or bad our test set was, we swapped our test set for other groups test sets using the same training data, model type and model architecture to see if performance scores increased. The following results are F1 scores using other groups test sets, chosen at random:

<b>F1 Score (%)</b>
<b>75</b>
<b>74</b>
<b>71</b>
<b>69</b>
<b>64</b>
<b>60</b>

4. Table of F1 scores across different groups

From what was gathered, every other group's test set that was tested gave a better score, not having changed a single thing during the training process. This, we think, is the major reason why not only ours but other groups' models' performance was quite poor and perhaps why some classifications could have been wrong.

## **4. Conclusion**

This error analysis shows that while our RoBERTa-based model can recognize emotional tone, its performance is limited by data quality, ambiguous language, and lack of context. Most errors came from misclassifying neutral sentences, often confusing politeness or subtle tone for positive emotion. Short or vague sentences further challenged the model, which plateaued at an F1 score of 0.51—mainly due to noisy validation data from the Content Intelligence Agency. Testing with cleaner datasets from other groups improved scores to 75%, highlighting how essential high-quality, well-labeled data is. Future work should prioritize cleaner data, better context handling, and possibly multimodal features.

## References:

1. Demszky, D., Movshovitz-Attias, D., Ko, J., Cowen, A., Nemade, G., & Ravi, S. (2020). *GoEmotions: A dataset of fine-grained emotions*. arXiv. <https://arxiv.org/abs/2005.00547>
2. Govi, P. (2018). *Emotions dataset for NLP* [Data set]. Kaggle. <https://www.kaggle.com/datasets/praveengovi/emotions-dataset-for-nlp>
3. OpenAI. (2023). *ChatGPT* [Large language model]. <https://chat.openai.com/>