Predicting sentiment in Expeditie Robinson

Simona Dimitrova, Wesley van Gaalen, Kacper Janczyk, Matache Vlad Bogdan Narcis, and Jakub Cyba

Breda University of Applied Sciences

Academic Year 2023-24 Y2C

Alican Noyan

April 12th

2024

Abstract

In our current world, where we interact with technology in increasingly diverse ways, Natural Language Processing aims to bridge the gap between human conversation and computers. By employing the power of Deep Learning, complex algorithms learn to understand the connection between words, their meaning and context. Chatbots, support tools and even research has been drastically changed in recent times due to advances in the field.

Predicting sentiment in Expeditie Robinson

Introduction

Emotions can play a significant role in daily human interactions. From spoken language to a difference in tone can allow us to easily grasp and understand the speakers emotions. Recent progress in natural language processing (NLP) has allowed for emotions to be identified and classified more easily. Emotion detection is one of the most important application of NLP, allowing for improvement in decision-making process and better understanding of consumers' emotions.

In recent years, machine learning has made considerable progress within NLP, allowing computers to generate, and interpret human speech and text with greater accuracy. Within this field, emotion detection is a crucial aspect of Sentiment Analysis. It goal is to extract subjective qualities from text, e.g. emotions. This combination of technology and psychology allows for a nuanced understanding of human expression.

Early researches , such as Paul Ekman's study "Universals and cultural differences in the judgments of facial expressions of emotion." (1971), have laid the groundwork for modern emotion classification models. Ekman identified six primary emotions: happiness, sadness, fear, anger, surprise, and disgust . These categorical models provide a structured framework for classifying emotions, which allows for precise analysis and interpretation.

With that in mind, Banijay Benelux, a prominent media company, and 3Rivers, a leading media consultancy firm, have collaborated to explore the practical implications of emotion detection. Their joint endeavor aims to enhance viewer engagement by analyzing the elements of TV series that resonate with audiences. By carefully classifying content, including actors, actions, and expressed emotions, they aim to decode the complexities of viewer preferences and reactions. With this innovative approach, they are confident that they can provide a more personalized and enjoyable viewing experience for their audience.

Our project goal is to automate emotion labeling for the TV series Expeditie

Robinson, also known as Survivor, which is one of the flagship programs of Banijay Benelux. We plan to achieve this by developing an extensive emotion classification model that can accurately identify Ekman's six key emotions within each segment of the show, using existing emotion-tagged data. Our goal is to develop a pipeline that utilizes speech-to-text technology to extract and categorize emotions. This will provide the client with a deeper comprehension of viewer engagement patterns.

To ensure the accuracy of the emotion classification model, we will conduct a thorough comparison against emotion tags labeled by 3Rivers as a benchmark. This iterative process ensures that we refine and optimize our methodology, paving the way for more accurate and insightful emotion detection in media content.

In summary, our goal is to combine advanced technology and media analytics to uncover the emotional aspects of viewer engagement. We will achieve this through automated emotion classification, which will provide new insights into audience behavior. Our ultimate aim is to enhance the viewer experience and inform content creation strategies.

Data Processing and Exploration

In this section, we'll dive into explanation of the datasets used for this project and how they were processed. Moreover, we'll explore insights of the data.

Data Collection

With regards to this project's objectives, we were provided with various datasets containing sentences and emotions assigned to them. We can split this data into three distinct types: training data, test data and use-case data. Let's start with description of the training data.

The training data serves as a core to our model training process. We were provided with an overall amount of 9 different datasets, which contain sentences from various sources, such as Reddit, Twitter, TV show "Friends" and fairytales. We also used an additional dataset, called "LIAR" which contains 12.8K manually labelled short statements

used for fake news detection. These datasets and the number of annotated emotions can be seen in the Table 1 below.

Table 1

Training datasets and number of annotated emotions

Dataset	Number of Emotions
GoEmotions	27 + Neutral
SMILE	5 + nocode and not-relevant
Friends	6 + Neutral
MELD	6 + Neutral
CARER	6
Affective Text	6
Daily Dialogue	5 + no emotion
EmoBank	NA
Affect data	6 + Neutral
LIAR	6

The more detailed table, containing description of each dataset can be found in the Appendix section.

We assessed the effectiveness of our models by testing them on a dataset obtained from Kaggle. This dataset contains 1436 sentences, each labelled with a sequence ID and an integer value, along with the text itself. Our goal was to predict emotions based on these sentences. To evaluate the models' performance, we utilised the F1 score metric. Detailed discussion on this evaluation will be provided later in the report.

Finally, to fulfil the project's use case, we received "Expeditie Robinson" data from our client, Banijay Benelux. This dataset comprises 17 video files. Our objective was to extract the audio from these files, transcribe it, translate it from Dutch to English, and segment it based on the start and end times of each segment. This crucial information was provided in a CSV file containing three key columns: "Start Time," "End Time," and

"Emotions."

Data Processing

Before model selection and implementation, it was necessary to process the data we were provided with. Therefore, from the training dataset, we chose to merge the MELD, CARER, GoEmotions, SMILE, Affective, Affect, LIAR and Daily Dialogue datasets together. We've decided not to include Friends and EmoBank datasets, due to low data quality, which was assessed during individual analysis of each dataset.

We used Python's Pandas and NumPy packages to complete our data processing chores. First, we loaded the datasets into our environment. Following that, we began a data cleansing step by removing duplicate entries and doing a thorough check for missing values, which were then removed in order to maintain data integrity. We standardised the emotional labelling across our datasets using Ekman's six major emotions: happiness, anger, surprise, disgust, sadness, and fear. This meant creating a mapping mechanism in which various emotion labels were linked with these six categories for each relevant text input. We next performed a dataset aggregation procedure to combine all individual datasets into a single format. To ensure clarity and unity, we renamed the columns to distinguish between sentences and their associated emotion. Finally, the new dataset was stored in CSV format.

Data Exploration and Analysis

Since the objective of this project was to create the emotion classification model, it was necessary to first check the distribution of each emotion in the dataset. This can be seen in a Table 2 below.

Table 2

Distribution of each emotion in the dataset

Emotion	Frequency
happiness	224422
sadness	143454
anger	77918
fear	59695
surprise	35989
neutral	21237
disgust	5550

We can see that happiness is the most common emotion in the dataset, with 224422 sentences labelled as such. What really stands out is that disgust appears far less frequently, with only 5550 labelled sentences. This suggests a potential bias in the model, which could affect its ability to accurately predict sentences associated with "disgust."

After examining the distribution of emotions in the dataset, we further investigated the average sentence length for each emotion category. The results are as follows, seen in Figure 1 below.

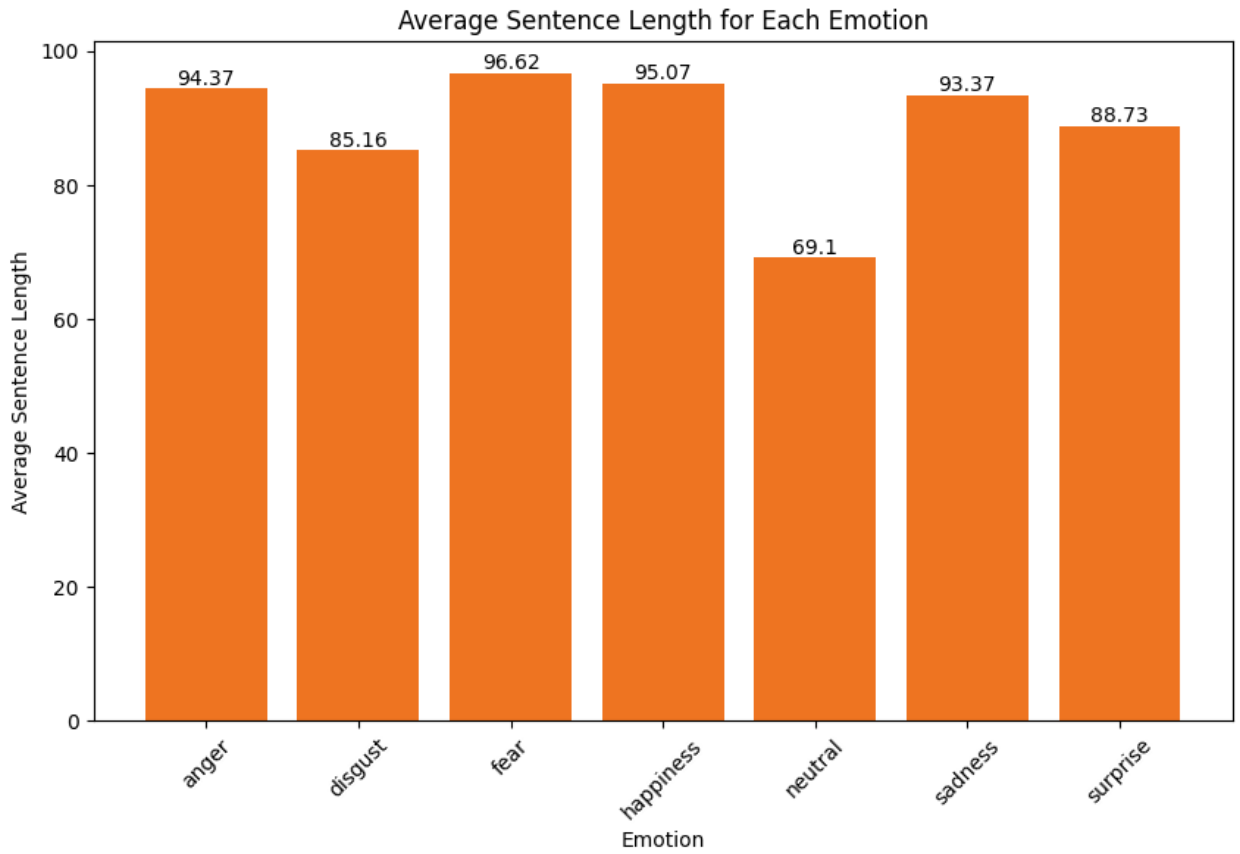

Figure 1

Average sentence length per emotion

These findings offer additional insights into the dataset, suggesting potential variations in sentence structures and expression lengths across different emotional states.

In the context of natural language processing (NLP), n-grams represent contiguous sequences of n items (typically words) within a text. These sequences provide insights into the syntactic and semantic structures of language, aiding tasks such as language modelling, text generation, and sentiment analysis (Jurafsky & Martin, 2024). In the provided dataset, 2-grams (also known as bi-grams) reveal frequent pairs of words occurring adjacently. This can be seen in a Figure 2 below.

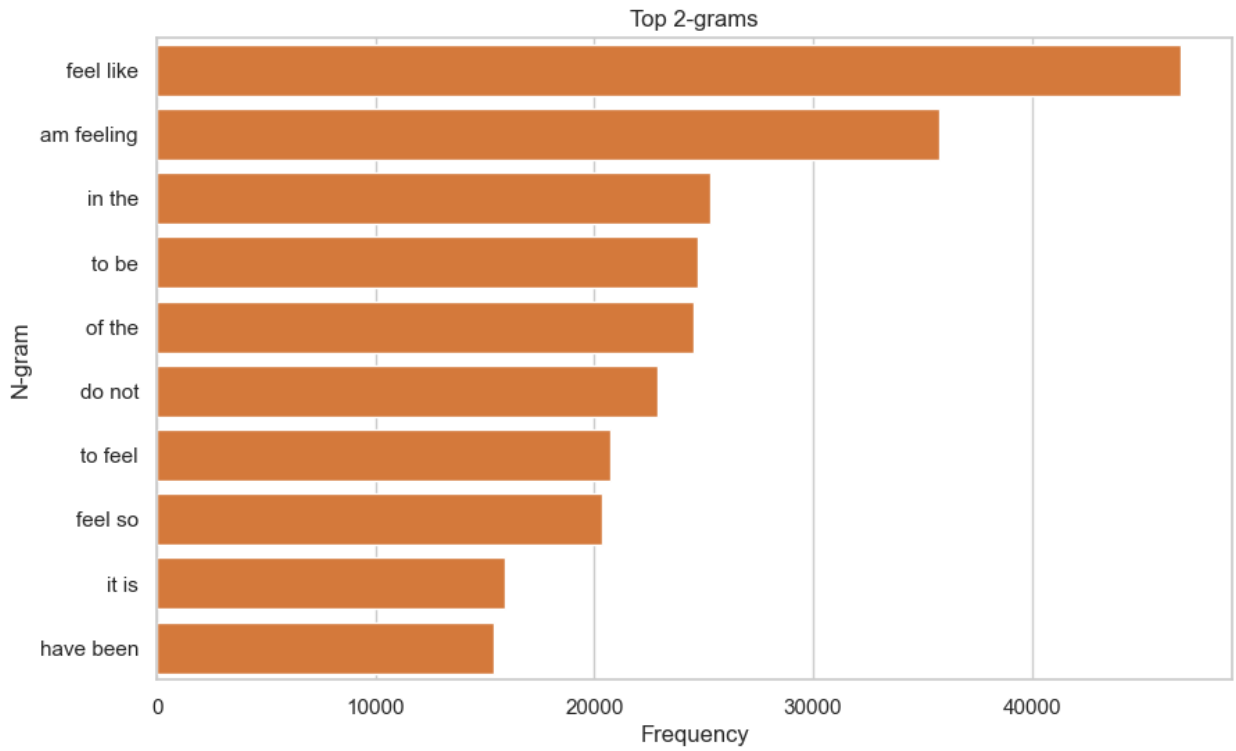

Figure 2

Top 10 most present 2-grams

Among the notable 2-grams, "feel like" emerges as the most frequent, occurring 46816 times. This combination suggests expressions related to subjective experiences or opinions. Other frequent 2-grams such as "in the," "to be," and "of the" represent common syntactic structures or collocations in the language. By analysing the distribution and patterns of these n-grams, NLP models can gain valuable insights into the linguistic features and contextual cues.

In addition to analyzing 2-grams, a word cloud was generated to visually represent the most frequent words in the dataset. Remarkably, the word cloud corroborated the insights collected from the 2-grams analysis. Words such as "feel,","feeling," and other emotionally charged terms featured prominently in the word cloud, aligning with the prevalence of related bigrams like "feel like" and "am feeling." This consistency reinforces the notion that these expressions are central to the dataset's emotional content. Moreover,

the word cloud provided a holistic view of the dataset's linguistic landscape, highlighting the salient words that contribute significantly to the emotional context. This wordcloud can be seen in the Figure 3 below.

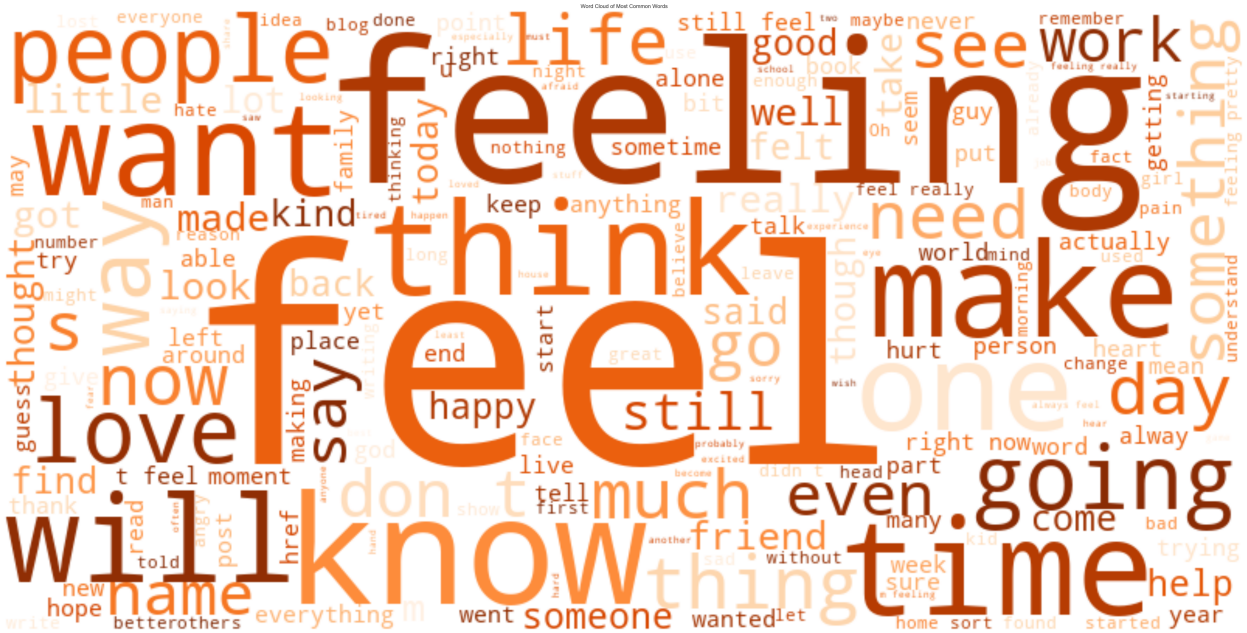

Figure 3

Wordcloud representing most frequent words

Model Selection and Implementation

After completing pre-processing and exploring the data, the next step was to choose and train a model. In our group project, we're tackling the challenge of multi-class emotion classification, specifically focusing on identifying six primary Ekman "Universals and cultural differences in the judgments of facial expressions of emotion.," 1971 emotions: haprpiness, anger, surprise, disgust, sadness, and fear from textual data.

The challenge was to train a multi classification model that is able to accurately capture and predict all 6 of these emotions on the student-generated test set and subsequent utilization of the model on the Expeditie Robinson episodes.

Simple Neural Network

For the multi-class emotion classification task, the first thing we've decided to use was a simple neural network model. There are few reasons behind this choice. Firstly, we wanted to have an understanding of how to work with the datasets we obtained. By using a simple neural network model, we could efficiently train it on the data we have, since the structure of such model makes it faster to train. Another reason behind using it was to have a base to work on and iterate on. By choosing to develop simple neural network model first, we had more room for improvements and iterations for the scope of this project. This enabled us to use more sophisticated models later on. The model produced us F1 score of 0.65. This is already a achievement, since this score value already met client's requirement, which was a score of 0.60. This performance laid down a solid starting point for us to start benchmarking and exploring various alternative methodologies and models. However, the goal was to create as accurate model as possible, therefore we've decided to take a look into transformer models, which are widely used in the industry.

For the data preparation for this task, a few steps were taken:

- **Tokenization and Padding**: The text data is tokenized into word indices using a tokenizer. Then, sequences are padded to ensure uniform length, which is essential for inputting data into neural networks since they typically require fixed-size inputs.

- **Label Encoding**: The emotion labels are encoded into numerical values using label encoding. This transforms categorical data into a format that the model can understand.

- **One-Hot Encoding**: One-hot encoding is applied to the numerical labels. This converts each integer label into a binary vector, where each element represents a different emotion class. This is commonly used for multi-class classification tasks to represent categorical variables.

Transformer models

In our search for a solution , we focused on investigating other techniques. After testing statistical and neural models for our emotion classification task we decided to dig deeper into what advanced NLP tools could offer. This prompted our exploration of the Hugging Face Transformer Library Wolf et al., 2020. The Hugging Face Transformer Library is an open-source library that provides a vast array of pre-trained models primarily focused on NLP. Transformers provide APIs to quickly download and use those pretrained models for fine-tuning them to any tasks. We centered on fine-tuning pre-trained language models such as BERT, RoBERTa, and LLaMA, which were originally developed for diverse language understanding tasks. "Fine-tuning a Neural Network explained," n.d. refers to the process of adapting these models to our specific emotion classification task by training them on emotion-labelled datasets. This process allows the models to learn the patterns associated with different emotions in textual data. Our objective is to improve these pre-trained models' ability to reliably predict emotions in text through using the comprehensive representations they have learnt and adapting them for our domain-specific purpose.

BERT

Our motivation to choose "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," 2018 for the task of multi-class emotion classification stems from its remarkable success in various natural language understanding tasks. BERT, developed by Google researchers, advanced the field of natural language processing (NLP) by introducing a bidirectional context understanding mechanism. One of the key advantages of BERT is its ability to capture contextual information from both left and right directions of a word, allowing it to grasp the dependencies and relationships within a sentence. This bidirectional approach enables BERT to understand the context in which words appear, leading to more accurate representations of text. Additionally, BERT's pre-training objectives, such as masked language modelling and next sentence prediction,

imbue it with a deep understanding of language semantics and syntax. This pre-training on vast amounts of text data from diverse sources endows BERT with a broad knowledge base, making it well-suited for transfer learning to downstream tasks like emotion classification. BERT's architecture, based on transformers, enables efficient parallelisation during training and inference, making it scalable to large datasets and computationally feasible for complex tasks. Firstly, we create dummies variables from the labels (emotions) then we tokenized the the train and test set. After tokenizing the train and test sets, we feed them into the model along with the dummy variables derived from the labels (emotions). The model then learns from this input to make predictions on new data. This process involves several steps:

- **Tokenization**: Converting the raw test data into numerical tokens that can be understood by the model. This involves breaking down the text into individual words or sub-words and representing them as unique integers.

- **Attention Masks**: These masks indicate which elements in the input sequence should be attended to by the model and which should be ignored. They help the model focus on relevant tokens and avoid processing padding tokens.

- **Input IDs**: These are numerical representations of the tokenized text data. They serve as the input to the model for processing.

- **Token IDs**: Each token in the input sequence is assigned a unique identifier. These token IDs are used to index into the embedding matrix to retrieve the corresponding word embeddings.

- **Targets**: These are the labels or the target values associated with each input instance. In the case of emotion classification, the targets represent the emotions expressed in the text.

RoBERTa

After fine-tuning pre-trained model BERT our curiosity drove us to explore alternatives options and we looked into RoBERTa as a possible replacement for BERT model. The RoBERTa model was proposed in RoBERTa: A Robustly Optimized BERT Pretraining Approach Liu et al., 2019 It is based on Google's BERT model released in 2018. Since RoBERTa is trained on larger dataset, understands language patterns better and it is proven to be more robust version of BERT, we fine-tuned RoBERTa model for our emotion classification task. Even though RoBERTa and BERT transformers model share the same model architecture they learn in different ways. RoBERTa mainly concentrates on understanding language patterns, trained on diverse datasets. During training, RoBERTa maintains its simplicity by repeatedly masking words, improving a straightforward learning process, BERT engages in a guessing game with some of the words it come across. These opposing approaches make RoBERTa more trustworthy and efficient in comprehending linguistic nuances. As a results, we chose RoBERTa for our emotion classification task, anticipating its deep comprehension would produce better results.

Model and data optimisation

To increase the performance of our model's predictions we have tried to apply various techniques.

Data Optimisation

To support the model in areas where it was under performing due to a shortage of data, multiple different techniques were tried, which are explained below.

Back Translation Augmentation Back-translation augmentation works by translating sentences to another language and then translating them back to English to obtain a slightly different sentence. In this project, we tried this approach by translating sentences to French and then back to English. However, this method proved to be detrimental to the predictions, dropping the F1 score from 0.81 to 0.52. Therefore, this approach was only tested and not applied in our final solution. Example of back

translation augmentation for a sentence containing the emotion surprise. **Original sentence**: "I am feeling very surprised". **Back-Translated:** "I feel very surprised".

Generated Sentences

OpenAI their large language model ChatGPT OpenAI's ChatGPT, 2024 was used to generate sentences for specific emotions. The input to the large language model was: 'Generate a sentence that expresses emotion.' This approach appeared to have a positive effect on the predictions, as further explained in the Error Analysis section. However, the model's accuracy in predictions began to decline rapidly when more than 1000 sentences for one emotion were added. An example sentence generated by the large language model for the emotion 'surprise' is: 'Her jaw dropped as she took in the unexpected turn of events.

Syntax-free Manipulation Augmentation Syntax-free manipulation involves altering the structure of a sentence while maintaining its meaning. By using this approach we sampled data from our training set and generated using ChatGPT OpenAI's ChatGPT, 2024 3 paraphrased versions of the original sentence. For example, considering the original sentence "I feel passionately that this activity should be punished." Through Syntax-free Manipulation Augmentation, this sentence can be paraphrased as "I strongly believe that this behavior should be penalized." or "This behavior has to be penalized, in my opinion.". This approach extended the datasets and improved the model's capacity to handle a variety of linguistic patterns. As a result, this method produced a slight improvement in the F1 score, which increased from 0.82 to 0.84. It should be noted that this increase was slight due to the relatively small size of the augmented sample when compared to the entire dataset.

Evaluation Metrics and Results

To evaluate the performance of the trained models, the following variety of evaluation metrics were used together with elaboration on why.

Accuracy Accuracy was used to see how many classes we were able to predict correctely in total.

The formula for accuracy is given by

$$Accuracy = \frac{\text{True Positives+True Negatives}}{\text{True Positives+True Negatives+False Positives+False Negatives}}$$

Precision

Precision was used to calculate the proportion of accurately classified occurrences or samples among those classed as positives is measured by precision. $Precision = \frac{\text{True Positive}}{\text{True Positive+False Positive}}$. The formula for precision is given by

High precision would mean that when the model predicts a specific emotion, it's likely to be correct.

Recall

Recall was used to identify all data points in relevant cases. The formula for recall is given by $Recall = \frac{True\ Positive}{True\ Positive+False\ Negative}$.

High recall ensures that the model identifies most instances of a particular emotion.

F1 score

F1 score balances precision and recall, which is essential for this project(emotion classification). Since we want the model to be reliable across various emotional states. It ensures the model does not favour detecting one emotion over others, maintaining a balanced sensitivity to all emotions. The formula for the F1 score is given by $F1 = \frac{2\times\text{precision}\times\text{recall}}{\text{precision+recall}}$.

When it comes to evaluating the model, we used Kaggle as our testing platform. Kaggle's structured competitions provide an ideal environment for robust evaluation, with well-defined evaluation criteria like the F1 score in our case. Our evaluation involves assessing the model on a student-generated test set to ensure its validity in real-world settings such as the Expedition Robinson data. Throughout the evaluation phase, the model was tested on a representative subset of 30% of the whole test set.

Error Analysis

To get a deeper understanding of the errors our model was making, we performed various ways of error analysis and, with the results, iterated on our model to improve its

performance.

We used classification matrices throughout the error analysis to get insight into the performance of the model per emotion. These classification matrices show the precision, recall, and F1 score per specific emotion.

Table 3 shows that surprise has a low F1 score, and from the high precision, we can conclude that if the model predicts a sentence has the surprise emotion, it's likely to be correct. However, the recall shows that we were able to predict only around 65% of the surprise occurrences correctly.

Emotion	Precision	Recall	F1-Score
Anger	0.77	0.91	0.83
Disgust	0.91	0.83	0.87
Fear	0.80	0.93	0.86
Happiness	0.80	0.97	0.87
Sadness	0.93	0.76	0.83
Surprise	0.92	0.65	0.76

Table 3

Classification Matrix of first iteration RoBERTa. Total F1 score: 0.849

This indicates that the model is experiencing trouble identifying sentences with surprise emotion. To counter this issue, we added approximately 500 sentences with the surprise emotion generated with ChatGPT. This had a very positive effect on the recall of surprise, along with the overall F1 score, as shown in table: 4

Emotion	Precision	Recall	F1-Score
Anger	0.79	0.89	0.84
Disgust	0.90	0.81	0.85
Fear	0.87	0.88	0.88
Happiness	0.86	0.94	0.89
Sadness	0.87	0.77	0.82
Surprise	0.85	0.84	0.85

Table 4

Classification Matrix after adding more surprise sentences to training. Total F1 score: 0.853

We performed this iteration a number of times, where in every iteration we added more sentences for the emotion with the lowest recall. This led to an F1 score of 0.874. We noticed that a huge part of the sentences in the Kaggle competition test set were ChatGPT generated. The proportion of ChatGPT-generated sentences in our dataset was rather low; therefore, we decided to remove 100,000 rows (which were not GPT-generated) from our dataset to increase the proportion of ChatGPT-generated sentences. This led to an increase in all emotions, as displayed in table 5

Emotion	Precision	Recall	F1-Score
Anger	0.89	0.87	0.88
Disgust	0.86	0.95	0.90
Fear	0.86	0.93	0.89
Happiness	0.96	0.95	0.95
Sadness	0.94	0.82	0.88
Surprise	0.88	0.91	0.89

Table 5

Classification Report of our final iteration. Total F1 score: 0.893

Furthermore, we also inspected some of the mistakes our model was still making using LIME (Local Interpretable Model-agnostic Explanations)Ribeiro et al., 2016. An example result generated using LIME is shown in image 4

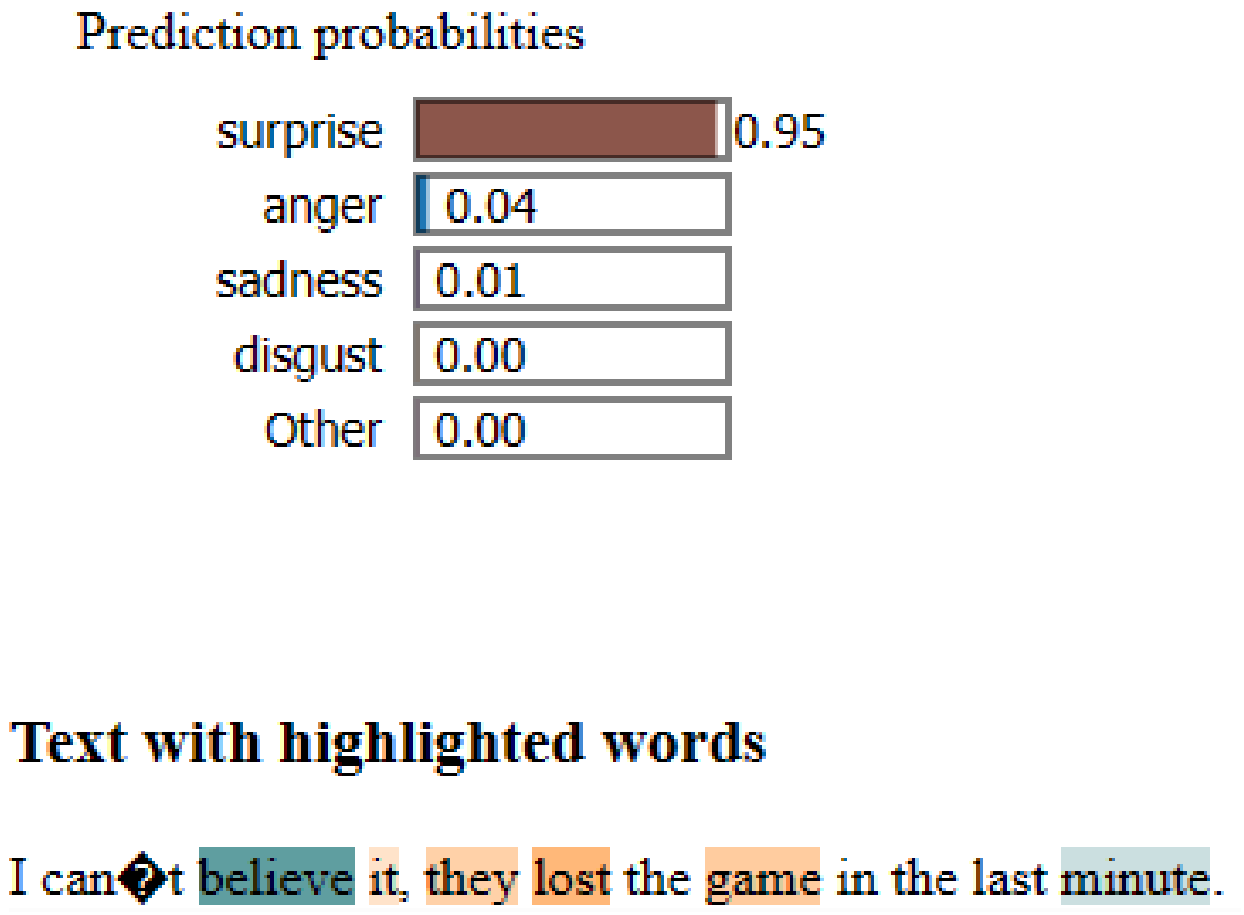

Figure 4

Inspection of a sentence using LIME

Discussion and Conclusion

Results

In this section the results from our selected models are presented and analysed.

Simple NN model

The results from our initial model were reasonably satisfactory, with a F1 score of 0.65.This is already a achievement, since this score value already met client's requirement, which was a score of 0.60.

BERT model

The results from the BERT model are notably impressive, demonstrating a significant increase in performance over our first model, with an F1 score of 0.82, This considerable improvement demonstrates the value of using cutting-edge language pre-trained models such as BERT.

RoBERTa model

So far, RoBERTa has been our best-performing model. In our initial test the model had F1 score of 0.84, before any filtering or data augmentations, proving us that it performed better and was more reliable than simple NN, BERT, and Llama. This convinced us to back our commitment to continue refining and improving this model through ongoing training process. After data augmentation techniques based on generated sentences the model improved significantly with F1 score of 0.89 on the test set.

Discussion

The following section will discuss encountered limitations and our proposed future steps to tackle and improve our solution.

Limitations

As with any project, we encountered a number of limitations that need to be not only acknowledged but their implications need to be considered in order to truly evaluate the solution.

Time Constraints

Due to the project being undertaken as an educational endeavour, that students are graded on, our time was constrained to the span of 8 weeks. Throughout the 8 weeks we are expected to understand the client requirements and also build skills necessary for completion which can lead to a number of problems. Additionally, the client data was provided during week 6 which made it difficult to iterate and improve model performance while also fulfilling all other requirements.

Unequal Distribution of Data

As explained in the EDA section of the report, one of our findings when taking a closer look at the distribution of the emotions across the datasets we see a very big difference between happiness and disgust, the most represented and most under-represented emotions(happiness: 224422; disgust: 5550). In the end our two best submissions were very different in nature, firstly, due to the different models used (RoBERTa vs BERT), but also due to our approaches to the dataset, balanced and unbalanced. The problem arises from the fact that balancing the dataset means sampling all sentences and only choosing a number equal to the number of the least frequent emotion, which limits our available data to train on, resulting in diminished performance. Without balancing we introduce biases which might lead to unforeseen problems.

Lack of Ground Truth

As mentioned above, our test set has been created by students, while large enough to encompass all primary emotions in Ekman's paper it does not represent normal human speech. When evaluating model performance it is important to monitor different metrics and ideally test out real world performance by observing a sample of predictions. When it comes to communication multiple interpretations are possible which makes setting a ground truth difficult. Additionally, our task consisted of predicting emotions in segments of dialogue in "Expeditie Robinson" a TV series who's cast speaks in Dutch. Due to possible transcription and translating errors it is possible that context or data is being lost which can negatively impact out model's accuracy.

Limited Generalization

One of the limitation we faced during the training process was that the model was tested on a small sample of 1436 sentences generated from students. Our project aims not only to train and fine-tune a model but also to predict emotions in Expeditie Robinson episodes. The student-generated data consisted of simple sentences expressing clear emotions, whereas the Expeditie Robinson episodes feature dialogues where emotions are

not always obvious.

Future Steps

While we present a solution in this report, we are aware of the possible improvements that would be possible given more time investment in this project. By allocating more time, we would be able to iterate models and improve training data quality and quantity which could increase our weighted F1 score. Additionally, by labelling some of the "Expeditie Robinsion" data manually we would have a ground truth to evaluate our model on. Another possible improvement we have discussed is using generalized, pre-trained large language models to create even more training data for out model. And finally, as with every AI-driven project more data will always yield better results, to this end we have discussed the possibility of finding more open-source datasets which we can incorporate in our pipeline which would ideally help us solve problems such as the unequal distribution of emotions.

Conclusion

In conclusion, barring the encountered limitations, by using different AI approaches to the task and iterating over our successful attempts we exceeded Banijay's requirement of 0.615 weighted F1 and achieved 0.893 using a combination of a RoBERTa model trained on balanced, processed data. We have managed to create a complete pipeline which has the ability to not only transcribe and translate episodes of their TV series but also accurately predict emotions that are present in specific segments; this enables Banijay and 3River to perform further analysis to better the products they present their consumers with. In retrospective, the project created an exciting and challenging learning experience which pushed us to strive for better results at every step of the way, by researching different models and possible features; improving the quality and quantity of our data, we managed to create a solution that outperforms initial expectations and hopefully helps our clients further their goals.

References

Bert: Pre-training of deep bidirectional transformers for language understanding. (2018). https://arxiv.org/abs/1810.04805

Fine-tuning a neural network explained. (n.d.). https://deeplizard.com/learn/video/5T-iXNNiwIs

Jurafsky, D. J., & Martin, J. (2024, February). *Speech and language processing*. https://web.stanford.edu/~jurafsky/slp3/

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

OpenAI's ChatGPT. (2024).

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "why should i trust you?": Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144. https://doi.org/10.1145/2939672.2939778

Universals and cultural differences in the judgments of facial expressions of emotion. (1971). *Journal of Personality and Social Psychology*.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., ... Rush, A. M. (2020). Transformers: State-of-the-art natural language processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38–45. https://www.aclweb.org/anthology/2020.emnlp-demos.6

Appendix