**COMP8420 – Advanced Natural Language Processing**

Major Project (ASSESSMENT TASK) **- 3**

**Integrating Audio Analysis with NLP for Enhanced Sentiment and Intent Recognition in Customer Interactions**

BY

RODULFO II DELA PAZ ALFAFARA (47747420)

rodulfoiidelapaz.alfafara@students.mq.edu.au

&

JELENE GUZMAN (47499095)

jelenedelossantos.guzman@students.mq.edu.au

Session-1, 2024

A close-up of a person

Description automatically generated

MACQUARIE UNIVERSITY

SYDNEY, AUSTRALIA

This is the Project Report Summary. No Codes and only highlighted figures are included here. For the comprehensive analysis inclusive of all codes & figures, please refer to the github repository link provided under the References section towards the end of this report.

**INTRODUCTION**

**Background**

In our first assignment, we were tasked with using a pre-trained sentiment classifier model. When we were evaluating the performance of the sentiment classifier model, we encountered a notable challenge: our classifier identified a recorded audio of an individual who was supposed to be in distress or having a negative experience as calm. This discrepancy highlighted a significant gap in the classifier's ability to accurately interpret the emotional state conveyed through text context alone. Motivated by this experience, we set out to explore ways to improve the accuracy of identifying and tagging angry callers. Our solution aims to merge Text Context Analysis and Audio Analysis for a more robust sentiment classifier. Humans inherently rely on multiple cues to gauge emotions—text content, sighs, panting, breathing, and tone or volume of speech. Similarly, a truly effective sentiment classifier should integrate both textual and audio signals to emulate this holistic approach.

The real-world applications of this enhanced sentiment analysis are extensive, particularly in contact centers. Every major brand, irrespective of the industry, operates contact centers to address various customer intents. However, the potential of this solution extends beyond customer service. It can be pivotal in law enforcement, security systems, and violence prevention, providing a comprehensive tool for identifying and responding to distress signals in diverse scenarios. By combining text and audio analysis, we aim to develop a sentiment classifier that mirrors human perceptive capabilities, offering a more accurate and nuanced understanding of emotional states.

**Scope & Limitations**

The scope of this project is focused on developing a model that can accurately classify spoken language as "angry" or not. The model will be trained on various datasets of audio recordings of spoken language, with the goal of identifying statements that exhibit an angry tone. In contrast, the project will not attempt to classify emotions beyond anger, such as happiness, sadness, or fear. This scope is intentionally narrow to ensure a high degree of accuracy within the constraints of time and available datasets. The model will not be able to detect sarcasm or irony in spoken language, nor will it be able to classify emotions beyond anger.

**METHODOLOGY**

The text context analysis and audio analysis are two crucial components of the proposed solution. For the text context analysis, the team leveraged on pre-trained sentiment classifier models, which have been shown to be effective in sentiment analysis tasks. These models have already been well-tested, hence, they can be leveraged to achieve good performance. For the audio analysis, the team has conducted extensive research and has identified several features that can be used to analyse audio signals. The team used these features to analyse the audio signals and extract relevant information. The outputs of the text context analysis and audio analysis were then used as inputs to a neural network model. This model highlighted the relationships between what is being said and how it is being said, allowing the team to demonstrate the effectiveness of the proposed solution.

**Approach (Technologies Implemented, Datasets Utilized)**

**1. Text Context Analysis –** codes can be found in Text Semantic Analysis Jupyter notebook

First, we created a program designed to determine whether a caller is angry based on the sentiment of their transcribed speech. It begins by transcribing the input audio file into text using a transcription function. Then, it utilizes the sentiment analysis capabilities of the transformers library, specifically leveraging the twitter-roberta-base-sentiment-latest model from Cardiff NLP [1]. This is a pre-trained language model that has been trained on a massive dataset of approximately 124 million tweets from January 2018 to December 2021. This extensive training dataset allowed the model to learn generalizable patterns and relationships in language, enabling it to perform well on a wide range of natural language processing tasks. Then, the model has been specifically fine-tuned for sentiment analysis using the TweetEval benchmark, which is a widely-used evaluation metric for sentiment analysis tasks. TweetEval provides a standardized framework for evaluating the performance of sentiment analysis models on Twitter data. By fine-tuning the RoBERTa-base model on this dataset, the model has learned to recognize and classify sentiment in tweets with high accuracy [1,2]. To validate the robustness of this model, its performance was closely compared with the badmatr11x/hate-offensive-speech model, which was specifically fine-tuned on datasets containing annotated tweets, categorized into three classes: hate-speech, offensive-speech, and neither [3]. This model has been shown to excel, particularly in identifying hate speech and imminent physical altercations. However, since the primary objective of this project focuses on customer calls, certain angry calls were found to be missed by this model, as they did not meet the criteria for hate or offensive speech. In contrast, these calls were correctly identified as negative by the Cardiff NLP model. This discrepancy led to our decision to move forward with the Cardiff NLP model. For the actual comparison of both models’ evaluations, please refer to the Comparing the badmatr11x/hate-offensive-speech V.S. cardiffnlp/twitter-roberta-base-sentiment-latest section of the Evaluation Jupyter notebook.

The chosen pre-trained model, accessed through a pipeline, analyses the sentiment of the transcribed text and returns a score and a label indicating the sentiment.If the sentiment label is 'negative,' the program infers that the caller is angry. Otherwise, it concludes that the user is calm. The function's effectiveness is enhanced by the robustness of the pre-trained sentiment analysis model, which has been trained on a diverse and extensive dataset. This model's performance has been globally recognized and validated, excelling particularly in identifying overtly negative texts, such as those containing curse words, hate speech, and offensive remarks.Due to its high accuracy and widespread use, the sentiment analysis model employed in this function did not require additional training. Its proficiency is evident in the upcoming ablation study, where its performance is shown to be nearly as good as more complex solutions that also consider audio signals.

**2. Audio Signals Analysis -** codes can be found in Audio Signals Analysis Jupyter notebook

Zhu-zhou et al. employed Convolutional Neural Network (CNN) models for audio-based violence detection, utilizing a variety of audio features to capture the nuances of audio signals. Their approach focused on extracting Mel-Frequency Cepstral Coefficients (MFCCs) and Delta MFCCs (ΔMFCCs), which assess the intensity and tone variations in speech through spectral envelopes. Additionally, they analysed pitch, indicative of vocal cord vibrations and changes in voice; Zero Crossing Rate (ZCR), which measures frequency signal transitions; Spectral Rolloff (SR), which calculates frequency energy distributions to determine the shape of the spectral envelope; Spectral Centroid (SC), which examines spectral balance and brightness to indicate energy concentration in frequency ranges; and Spectral Flux (SF), which identifies rapid transitions by assessing frequency range changes, among many others [6]. Inspired by this methodology, we decided to observe these features and use them as inputs in a CNN model. Our research led us to a similar approach adopted by Aditya, a Speech Emotion Recognition researcher [4].

Aditya's approach to speech emotion recognition involves a series of detailed steps aimed at accurately identifying the emotional state of a speaker through audio analysis. The first step is Speech Input, where the user's voice is recorded. This step captures the raw audio data, which is the foundation for further analysis. Next, the audio undergoes Pre-processing. This crucial step involves cleaning the audio data by removing noise and potentially isolating specific speech segments. Noise removal ensures that the audio features extracted later are not contaminated by irrelevant background sounds, which can interfere with accurate emotion detection. Isolating specific speech segments can help focus the analysis on parts of the audio that are more likely to convey emotional cues. Feature Extraction follows, where key features related to emotions are extracted from the pre-processed audio [4]. These features are categorized into three main types:

* Prosodic Features: These include pitch, intonation, volume, speaking rate, and pauses. Prosodic features are essential as they capture the expressive aspects of speech that convey emotions. For instance, a higher pitch might indicate excitement or anger, while a slower speaking rate with longer pauses might suggest sadness or tiredness.
* Spectral Features: These features involve analyzing the spectrum of the voice, including Mel-Frequency Cepstral Coefficients (MFCCs). MFCCs are particularly important because they emphasize qualities of the audio that are perceptually relevant to human hearing, such as the timbre and tonal qualities of the voice. By capturing these details, spectral features help in distinguishing between different emotional states.
* Voice Quality Features: These include jitter and shimmer, which are small variations in voice quality. Jitter refers to the frequency variation from cycle to cycle, and shimmer refers to the amplitude variation. These features can indicate stress or tension in the voice, which are important cues for emotion recognition.

Once these features are extracted, Aditya's model employs a trained machine learning model to analyse them. This model uses classification algorithms to identify the associated emotion based on the input features. The model learns to associate specific patterns in the features with particular emotions through a training process involving a large and diverse dataset. Finally, the system outputs the detected emotion. Aditya's approach demonstrates the importance of a comprehensive analysis that considers various aspects of the audio signal, ensuring a robust and accurate emotion recognition system. This multi-faceted feature extraction process, combined with advanced machine learning techniques, enables the system to perform effectively in identifying human emotions from speech [4].

Following Aditya’s recommendations, the audio features analysed by our proposed solution include:

• Zero Crossing Rate (ZCR): This feature captures the noisiness or harmonic content of an audio signal. A higher ZCR value indicates a noisier signal, while a lower value indicates a cleaner signal.

• Chroma STFT: This feature captures the harmonic and melodic characteristics of an audio signal. Chroma STFT is a type of spectrogram that represents the frequency content of an audio signal over time.

• Mel-Frequency Cepstral Coefficients (MFCCs): These features are used to distinguish between different sounds and are commonly used in speech recognition and audio classification tasks.

• Root Mean Square Value (RMS): This feature captures the dynamic range and energy profile of an audio signal. A higher RMS value indicates a louder signal, while a lower value indicates a softer signal.

• Mel Spectrogram: This feature provides a time-frequency representation of an audio signal, emphasizing perceptually relevant frequencies.

By leveraging these comprehensive features, both Zhu-zhou et al.'s and Aditya's approaches’ effectiveness on integrating detailed audio feature analysis into CNN models for applications in emotion recognition is fully demonstrated in the Audio Signals Analysis Jupyer notebook.

The dataset used to train the model is the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS), a comprehensive collection designed to aid in the study and development of emotion recognition systems. The RAVDESS dataset is renowned for its high quality and extensive range of emotional expressions in speech and song. It includes audio-only files of speech at a 16-bit depth and 48 kHz sampling rate. The full dataset, which encompasses both audio and video recordings, occupies 24.8 GB and is available on Zenodo. This dataset has been meticulously constructed and validated perceptually, as detailed in an open-access paper published in PLoS ONE. The specific portion of the RAVDESS dataset used includes 1440 audio-only files. The dataset comprises recordings from 24 professional actors (12 females and 12 males), each performing 60 trials, leading to a total of 1440 distinct files. These actors vocalize two lexically-matched statements in a neutral North American accent. The speech covers a range of seven emotions: calm, happy, sad, angry, fearful, surprise, and disgust. Each emotion is expressed at two levels of intensity (normal and strong), except for the neutral emotion, which has no strong intensity variant [5].

**3. Accent Classifier** - codes can be found in the Accent\_Classifier Jupyter notebook

To further enhance our proposed solution, we aim to recommend which agents could best handle a particular caller, especially focusing on addressing angry callers. This solution involves not only detecting agitated customers but also ensuring their cases are managed effectively. For instance, it is increasingly important to assign callers to agents who share similar tones or vocabulary to better understand and address customer needs. Therefore, our recommendation system will incorporate an accent classifier to determine the customer's accent and subsequently assign them to an agent who speaks similarly.

For the accent classifier, we utilized a pre-trained model, english\_accents\_classification by dima806 [7]. This model performs well in identifying whether a speaker's accent is from the US, England, India, Australia, or Canada. However, to broaden its capabilities to identify a wider range of English accents, we decided to fine-tune the model using the Common Voice dataset. This dataset is a corpus of speech data consisting of audio recordings of people reading text from various public domain sources, such as user-submitted blog posts, old books, movies, and other public speech corpora. The Common Voice dataset is organized into several subsets for user convenience. Subsets labeled "valid" contain high-quality audio clips that have been verified by at least two listeners, who agree that the audio matches the text. These clips are suitable for training and testing ASR (Automatic Speech Recognition) systems. Conversely, subsets labeled "invalid" contain low-quality clips where listeners disagree on the audio-text match, making them unsuitable for ASR training or testing. Clips labeled as "other" have insufficient votes to be classified as valid or invalid and are also unsuitable for ASR training or testing. The "valid" and "other" subsets are further divided into three groups (Dev, Train & Test sets) [8]. The Common Voice corpus is a valuable resource for ASR system research and development, offering a diverse set of speech data to enhance ASR accuracy and effectiveness [8]. Due to computing resource constraints, we utilized only the test set of the Common Voice dataset, containing 3,995 audio clips, and further split this data into training and testing sets for model evaluation. For details on the training and evaluation process, please refer to the Accent\_Classifier Jupyter notebook. By integrating this accent classifier, we enhance our ability to match callers with agents who can best address their needs, improving overall customer service and satisfaction.

**4. Age & Gender Classifier -** codes can be found in the Age & Gender Classifier Jupyter notebook

To further complement the handling of angry callers, our proposed solution also incorporates a gender and age classifier to determine the caller's demographic profile and subsequently assign them to an agent who can best relate to their needs. For more details, please refer to the Gender\_Age\_Classifier Jupyter notebook.

By integrating this gender and age classifier, we enhance our ability to match callers with agents who can best address their needs. This demographic matching ensures that callers are connected with agents who are more likely to understand their concerns and communicate effectively, thus optimizing the customer service experience.

**5. Neural Network Model-** codes can be found in the Neural\_Network Jupyter notebook

Our proposed solution integrated features from text context analysis and audio signal analysis to develop a robust sentiment classification model. By reusing code from the Text Semantic Analysis and Audio Signals Analysis Jupyter notebooks, we were able to seamlessly extract and combine these features for input into a neural network model. The text context analysis provided semantic features by generating sentiment scores, reflecting the emotional tone of the text. Meanwhile, the audio signal analysis extracted the same set of features discussed in #2 above to capture nuances in the speaker's voice. To combine these features into a single tensor for the neural network, we expanded the dimensions of the sentiment scores and labels to match the audio features tensor. We then concatenated these tensors along the feature dimension, resulting in a comprehensive input tensor that includes both text and audio features. The neural network model was designed using TensorFlow and Keras. It consists of multiple dense layers with ReLU activation functions, dropout layers to prevent overfitting, and a final dense layer with a sigmoid activation function for binary classification. The model was compiled with the Adam optimizer and binary cross-entropy loss function, with accuracy as the evaluation metric. To ensure optimal training performance, we implemented early stopping and model checkpointing, ensuring we retain the most effective model for evaluation. The model was trained with a batch size of 8 and a validation split of 20%, using these callbacks to achieve the best results without overfitting. By combining text and audio features, our sentiment classification model tries to mimic human understanding, leveraging both textual and audio cues to identify and manage customer emotions effectively. This comprehensive approach provides a robust solution for detecting and addressing customer sentiments. For the dataset used for training, please refer to Evaluation Outcomes That Supported The Use of A Neural Network Model sub-section under the Experimental Results section.

**6. Proposed Classifier**

The proposed solution comprises a suite of integrated systems designed to accurately identify whether a caller is angry and subsequently assign the most suitable agent to manage the caller's needs effectively. This comprehensive approach utilizes a combination of advanced technologies, including a Sentiment Classifier, a Convolutional Neural Network (CNN) for Audio Signal Analysis, and a Neural Network that synthesizes the features generated by these classifiers. The process begins with the Sentiment Classifier and CNN Audio Signal Model working in tandem to analyze both textual and audio inputs from the caller. The outputs from these models are then fed into a Neural Network, which integrates these multimodal features to determine the emotional state of the caller. If the caller is identified as not being angry, the standard operational procedures are followed. In instances where the caller is determined to be angry, additional classifiers, namely, the Accent Classifier and the Age & Gender Classifier are employed. The Accent Classifier utilizes a model to identify the caller's accent, ensuring that the caller can be matched with an agent who speaks in a similar manner, thus facilitating better communication and understanding. Concurrently, the Age & Gender Classifier aids in further refining the agent assignment by considering demographic factors that may enhance rapport and empathy between the caller and the agent. This multi-faceted approach ensures that the system not only detects anger in callers with high accuracy but also optimally assigns agents who are best equipped to handle the caller's emotional and communicative needs, thereby improving overall service efficiency and customer satisfaction.

**A screenshot of a computer screen

Description automatically generated**

For the demonstration of the flow depicted from the diagram above, please refer to the Final Proposed Solution Using Trained Neural Network Model section of the Evaluation Jupyter notebook where the nn\_verify\_customer\_is\_calm function is being called on an audio recording that mimics a customer call; and the function correspondingly 1) identifies whether the caller is angry; and 2) to which agent the caller should be assigned to.

**EXPERIMENTAL RESULTS**

**Evaluations Performed**

The evaluations performed involve a systematic approach to ensure that the model accurately identifies whether a caller is angry. First, a dictionary is created to map each file name to its true label, extracted from the Data Frame. For each collected file path, the file name is extracted and the corresponding true label is retrieved from the dictionary. Then, the function that predicts whether the caller is calm or angry are called on each evaluated data point to make a prediction. The results, including the file name, true label, and predicted label, are stored in a list. The accuracy of the model is then evaluated by comparing the predicted labels with the true labels. The number of correct predictions is counted, and the accuracy is calculated as the ratio of correct predictions to the total number of predictions. This accuracy is printed as a percentage.

**Evaluation Outcomes That Supported The Use of A Neural Network Model**

Our decision to use a neural network model for the final design of our proposed solution was not made lightly. Initially, our solution relied solely on a program called verify\_customer\_is\_calm, which determined that a customer was angry if either the text semantic classifier model or the CNN audio signals model identified the caller as angry. However, after conducting thorough evaluations, we discovered that this approach missed several cases of angry callers. Specifically, the program failed to identify certain non-angry text phrases as indicative of anger when spoken in a particular tone, such as “this is unacceptable” or “I would like to speak with your supervisor.” This realization led us to conclude that the best way to capture the nuanced relationship between what is being said and how it is being said was through neural network models. Neural networks can learn complex representations of data, making them well-suited for training on input features that capture both text and audio cues. Indeed, as demonstrated by our subsequent tests, the neural network model we trained on these features successfully learned the intricate relationship between the content and delivery of speech. Despite having a relatively small training dataset, the model achieved high accuracy on the test data, validating our approach and highlighting the effectiveness of neural networks in this context. With this said, in this section, we will be discussing the different evaluation outcomes that inspired the final proposed solution.

For the initial purpose of evaluating on entirely new and unseen data, we created a dataset of 109 recorded calls, 62 of which were angry calls. This dataset was used for subsequent evaluations. However, as previously mentioned, we decided to utilize a neural network model after analyzing the evaluation outcomes. Therefore, we trained the neural network model on this dataset, splitting it into training, validation, and test sets. We chose this dataset over the RAVDESS or the Common Voice dataset for three main reasons:

1. Specificity to Customer Calls: This dataset was specifically designed to mimic customer calls, making it more relevant to our application.
2. Focused Scope: The dataset focuses exclusively on angry or non-angry calls, excluding other emotions that are outside the scope of our study and could potentially confuse the model given our limited training data.
3. Inclusion of Tricky Examples: The dataset contains challenging examples where the text context is neutral and the voice is not overtly angry but should still be tagged as angry, providing crucial training scenarios that better reflect real-world conditions.

By using this tailored dataset, we ensured that our neural network model was trained on data that closely resembles the actual use case, enhancing its ability to accurately detect and classify angry callers.

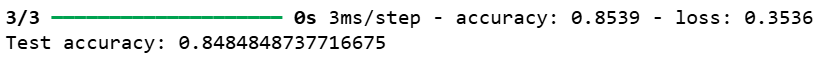
**Comparison of the Sentiment Classifier Models (badmatr11x/hate-offensive-speech V.S. cardiffnlp/twitter-roberta-base-sentiment-latest )**

By referring to the Comparing the badmatr11x/hate-offensive-speech V.S. cardiffnlp/twitter-roberta-base-sentiment-latest section of the Evaluation Jupyter notebook, we can observe that the cardiffnlp/twitter-roberta-base-sentiment-latest model outperformed the other model in terms of accuracy with a score of 68% against 63%. This outcome led to the team’s decision to move forward with the cardiffnlp/twitter-roberta-base-sentiment-latest model for the sentiment classifier component of the final proposed solution design.

**Ablation Study**

The focus of the Ablation Study for this project is to assess the importance of the main components of the proposed solution: text context analysis and audio analysis. Therefore, we conducted two ablations: 1) evaluating the performance of the solution without the audio analysis component, and 2) evaluating the performance of the solution without the sentiment classifier model. Referring to the Ablation Study section of the Evaluation Jupyter notebook, we observed that the Sentiment Classifier model without the CNN audio analysis achieved an accuracy of 67%, which is only slightly worse than the full model that included both components. In contrast, the CNN audio analysis model without the Sentiment Classifier performed significantly worse, with an accuracy of only 45%. This clearly indicates that both components are crucial to our solution. It is important to note, however, that the Sentiment Classifier model without the CNN audio analysis component performed only slightly worse than the full model due to the challenging nature of the evaluation dataset. This dataset includes statements that indicate anger without the presence of a shouting voice, making it harder to detect anger through audio analysis. If these nuanced cases were correctly detected by the combined solution, it would have achieved a much higher accuracy. This highlights the importance of analysing the relationship between text context and audio analysis to accurately identify angry callers in a variety of scenarios. This outcome and realization inspired us to employ a neural network model to address these gaps. By leveraging the combined features of text context and audio signals, the neural network model aims to improve the detection of angry callers in more nuanced scenarios. In the next section, we will discuss the evaluation of the proposed solution with the Neural Network model.

**Proposed Solution w/ Neural Network Model Evaluation**



By referring to the evaluation part towards the end of the Neural\_Network Jupyter notebook, we can observe that the performance of the final proposed solution on the test data has a high accuracy of about 85%. This perceived improvement could be an indication that the Neural network model was able to bridge the gaps of the former solution.

**CONTRIBUTION BREAKDOWN**

Rodulfo focused on the Text Context Analysis, Accent Classifier, and the Neural Network of the final proposed solution, while Jelene worked on the Audio Signals Analysis, Gender & Age Classifier, and the CNN trained and evaluated on the RAVDESS dataset for audio signals analysis. Jelene ideated the architecture for integrating the different classifier solutions to determine if a caller is angry, and Rodulfo facilitated the evaluation of the major components. Both shared ideas and discussions on every aspect of the project, from topic selection to implementation. Rodulfo consolidated these ideas into the report paper, while Jelene compiled them into markdowns and comments in the Jupyter notebooks. Both members contributed evenly and fairly to the final outcome of the project.

**References**

**Github Repository Link:** [**https://github.com/judyratfield/nlp\_final\_project**](https://github.com/judyratfield/nlp_final_project)

1. Twitter-roBERTa-base for Sentiment Analysis by Cardiff University: https://huggingface.co/cardiffnlp/twitter-roberta-base-sentiment-latest
2. TimeLMs: Diachronic Language Models from Twitter by Daniel Loureiro, Francesco Barbieri, Leonardo Neves, Luis Espinosa Anke, Jose Camacho-Collados: <https://arxiv.org/abs/2202.03829>
3. hate-offensive-speech model by badmatr11x: https://huggingface.co/datasets/badmatr11x/hate-offensive-speech
4. Speech Emotion Recognition by aditya1220: https://www.kaggle.com/code/aditya1220/speech-emotion-recognition/notebook
5. Livingstone SR, Russo FA (2018) The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. PLoS ONE 13(5): e0196391. <https://doi.org/10.1371/journal.pone.0196391>.
6. Zhu-Zhou, F. et al.: Computationally constrained audio-based violence detection through

transfer learning and data augmentation techniques, https://www.sciencedirect.com/science/article/pii/S0003682X2300436X.

1. english\_accents\_classification model by dima806: https://huggingface.co/dima806/english\_accents\_classification
2. Common Voice: 500 hours of speech recordings, with speaker demographics by Michael Henretty, Tilman Kamp, Kelly Davis & The Common Voice Team: https://www.kaggle.com/code/dima806/common-voice-accent-classification/input