Contents

3   Introduction

It can be observed that technology has yet to make a significant contribution towards aiding performance review in learning environments and indeed learning environments on the whole. This can largely be attributed to the lack of effectiveness of the existing tools, whereby much of the information obtained is neither insightful nor particularly meaningful.

Traditional approaches to the incorporation of technology in learning environments have tended to focus more on interactivity as opposed to analysis, which has resulted in a lack of progress in the areas of performance review, analysis, and overall utility (Cho et al. 2020). This is due in part to the complexity of such environments, with many subjective metrics and mediums that, at first glance, appear to be difficult to study and quantify.

In this project, speech emotion has been identified and selected as a suitable metric for analysis. Speech emotion is a language agnostic metric that transcends many of the barriers posed by the more obvious metrics such as spoken / body language or facial expressions.

Despite the many nuances of speech, such as tonality or pitch, there are clear patterns that can be derived from studying how emotions are conveyed (Angrick et al. 2019). A spectrogram is a visual representation of the spectrum of frequencies (in this case, the frequencies are converted to the mel scale) across a signal as a function of time. Consider the following mel spectrograms produced from two different male actors saying a short sentence in a blatantly angry and then sad manner.
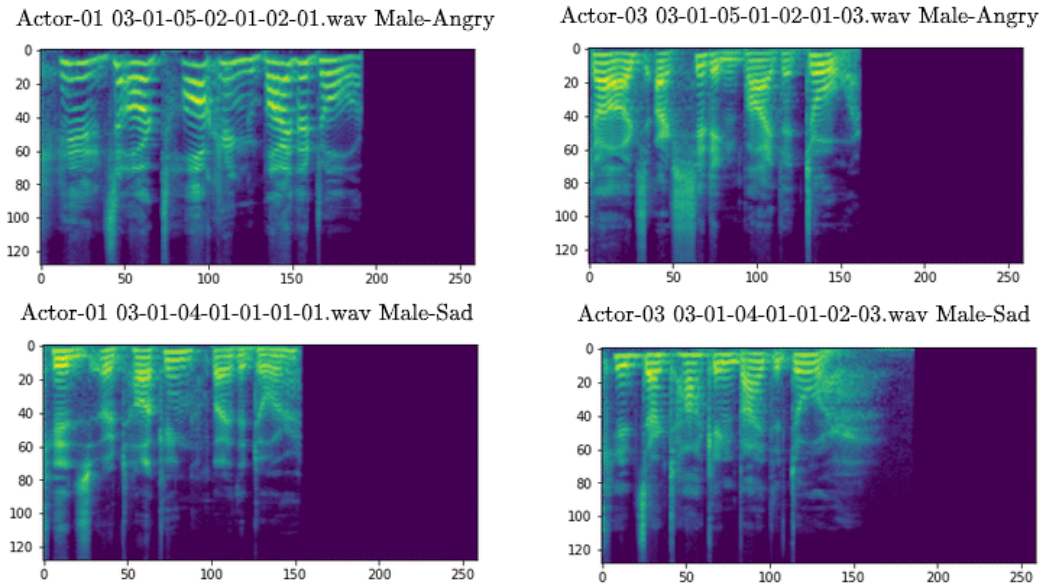
Figure 1 - spectrogram comparison.

Even from a macro perspective, it is possible to make out some visible similarities. Take for example the large circular shapes that can be made out from the two angry clips, or the distribution of the yellow (corresponding to pitch across the mel scale) lines in the sad clips. The spectrogram is represented by a matrix of values, which is where the vast, more subtle patterns can be identified and observed - making it an ideal candidate for a CNN approach.

Prior to the increased popularity of machine learning, conventional methods for analysing and processing speech involved the usage of Gaussian Mixture Models (GMM) and Hidden Markov Models (HMM), both of which produced results that were simply too inaccurate to ever put into production (Sainath et al., 2013; Venkataramanan & Rajamohan, 2019). The accessibility of machine learning techniques has lead to novel approaches such as the application of Recurrent Neural Networks (RNN) and CNNs, resulting in improved accuracy and introducing completely different approaches to the problem.

In this paper, inspired by the notable success and improvements seen from the application of CNNs in particular, a relatively accurate classification model has been trained through applying CNNs to Mel Spectrogram representations of audio files. The Mel Spectrogram effectively allows us to transform the classification problem from the audio domain into the visual domain, where CNNs have demonstrated to excel. As previously shown in figure 1, there are many subtle patterns to be found in the data, but it is a difficult task to manually calculate and label these ourselves. Hence a deep learning approach, where the model learns to recognise and extract features itself, becomes a clear suitable choice.

The final trained model has then been deployed into a general purpose graphical tool that can perform predictions across pre-recorded audio files or through recording on the fly. The graphical tool has been purposefully designed with simplicity in mind and with the intention of being easy to use and incorporate into learning environments.

4 Research Objectives

- To research, experiment, and develop an understanding with deep learning using industry standard frameworks.

- To develop a plausible speech emotion classifier using state of the art techniques.

- To deploy the speech emotion classification model into an accessible GUI application.

- To assess the feasibility of such a tool for use in learning environments.

5 Literature Review

5.1 Automatic Speech Emotion Recognition Using Machine Learning

Automatic Speech Emotion Recognition Using Machine Learning takes a comparative look at SER systems. The underlying method to extract and process the speech signals (using Mel-frequency cepstrum coefficients and modulation spectral features in conjunction with feature selection) remains the same whilst the machine learning paradigms differ. The paper first looks at a recurrent neural network approach and goes on to compare it against both multivariate linear regression and support vector machine approaches. The paper makes the claim that such approaches were selected due to their existing popularity in the field.

The initial criticism of the paper is that it fails to consider the CNN approach, despite its notable popularity and differences to the other methods. However, the paper is still concise and informative enough to warrant a full review. One positive quality that immediately stands out is how clear the information is presented, both in terms of looks and in use of language. An additional outstanding quality of this paper is the lack of pre-requisite knowledge required of the reader; many similar such papers assume at least a fundamental understanding and generally a lot more, whilst this paper describes most of the necessary knowledge required to grasp the content.

The paper starts by explaining the importance behind SER technology, the main argument can be reduced to how effective and convenient it is to obtain information pertaining to the emotional state of an individual in comparison to other methods (such as facial expressions and physiological signals). This argument can be applied to the idea of using such technology in learning environments based on the documented effectiveness of the results in similar scenarios. The paper even uses learning environments as an example of a potential application of such technology; Kerkeni et al. (2019) state that "a teacher can use SER to decide what subjects can be taught and must be able to develop strategies for managing emotions within the learning environment".

The authors then describe their system from a top-down perspective, this includes which emotions are in scope as well as the decision making process behind which algorithms are selected. It appears that the researchers were restricted by their datasets as to which emotions ended up in-scope. Whilst this does not affect the content of this particular paper, a more careful consideration will have to be made when applying this to learning environments (as per this project).

Delving further into the paper, the authors describe how the different algorithms are applied to solve the problem, which is achieved by mathematical notation and descriptions of important variables. This is of course a necessity for such a paper, although the notation is clear enough for even those with weaker mathematical backgrounds to gain insight. The paper concludes by displaying and contrasting between the results obtained from the experiments; the main conclusion is that SER produces the best recognition results with a limited dataset whilst RNN performs better when a higher volume of data is available. Such results will prove vital when comparing against and assessing the effectiveness of our SER system.

5.2 A CNN-Assisted Enhanced Audio Signal Processing for Speech Emotion Recognition

Due to the notable omission of a CNN approach in the first reviewed paper, this paper has primarily been selected to gain practical insight into such an approach. The paper starts by defining the motivation behind it, using similar arguments to the previously reviewed paper. One particular area of interest is the point about "extracting hidden information", where Kwon (2020) refers to uncovering previously undetected, new information using a CNN approach. At first glance, it appears that the author looks at the issue more through the lens of human computer interaction (HCI), contrasting with the previous paper's theoretical approach.

An immediate positive point of the paper is that it describes its approach clearly and in plenty of detail, though unlike the first paper, this paper requires the reader to possess fundamental knowledge concerning machine learning terminology and methods. The paper goes on to discuss the proposed methodology, which is essentially a model comprising of input layers (taking in 2D speech signal spectrograms), several convolutional layers flattened down, and two fully connected layers with the softmax classifier applied at the end. Compared to the first paper, this approach seems easier and requires a lot less pre-processing work applied to a given dataset (since the audio just needs transforming into a 2d spectrogram). The main downside to this paper is that it lacks technical and theoretical insight, providing only a single equation unrelated to the classifier itself.

Cutting to the end results of the paper, given the clear lack of training applied to the model, the results achieved are not bad. Although lower than one might expect given some of the cutting-edge results achieved through CNNs on other projects; the lack of training surely impacts this. In conclusion, the paper has provided an insightful and inspiring approach to be considered for this project. As with the prior paper, the end results will be interesting to compare against in the end.

5.3   Emotion Recognition from Speech

Emotion Recognition from Speech has been monumental in aiding the architectural decisions taken for building the classifier used in this project. In the paper, the authors build and compare an assortment of comprehensive approaches to speech emotion recognition, comparing many of the industry standard methods, such as HMMs, Mel-frequency cepstrum coefficients (MFCCs), and CNNs. In particular, the authors find that a 4 layer model with batch normalisation and max pooling yields the highest accuracy across their dataset. The authors additionally discuss filtering and data processing techniques, such as applying Wiener filtering to the audio to help filter out noise. The paper provides an extremely useful framework of architectural decisions and comparisons that have proven very useful throughout the development of this project.

5.4   Factors Affecting Technology Integration in the Classroom

This paper provides solid reasoning on the critical factors affecting the integration of technology in learning environments. The paper consists of 5 key reasons as to why classrooms benefit less from technology than many other environments: poor infrastructure, inadequate technology, lack of sufficient, effective professional development, low self-efficacy, and teacher perceptions. Each reason stated in the paper is insightful, but the points regarding inadequate technology and teacher perceptions are of the most relevance to the project. As previously mentioned, one of the goals of this project would be to put forth a valid and viable application of technology in learning environments (as to build some interest around the idea). Harrel & Bynum (2018) state that teachers "perceive the effort needed to learn the new technology and practicality of it as a significant consideration in whether they use it or not", so in order for this to work, the educators must be convinced of the technology.

5.5 Modeling learners' cognitive and affective states to scaffold SRL in open-ended learning environments

Whilst this paper largely covers an area unrelated to the project, the points regarding relationships between cognitive and affective states are interesting and will be of use to the analysis stage of the project. The paper goes on to describe an interesting correlation between two affective states, boredom and delight, and academic performance. As one might expect, an exposed state of boredom significantly decreases cognitive and subsequently academic performance, as documented by the paper. With these statistics in mind, prioritising the prediction of the states of boredom and delight from speech will be areas of particular interest to focus training the model on.

6 Methodology

Selecting the correct tools to tackle this project was challenging due to the many pros and cons of the existing frameworks available. Python was the obvious language of choice due to the massive amount of well documented machine learning libraries available. Had the dataset been larger however, Python and Python libraries would have been switched out for a lower level approach, using C++ for example, for efficiency. Initially PyTorch was selected as the deep learning framework of choice due to the pythonic syntax and therefore ease of use. Whilst an implementation could be made using PyTorch, the end model was fundamentally flawed. The documentation of PyTorch, especially for a beginner, is not the most clear; many parameters are not sufficiently explained and a lot of prior knowledge is assumed of the reader.

Eventually PyTorch was switched for Keras and TensorFlow, with much more success. Whilst TensorFlow is not quite as elegant or lightweight as PyTorch, the documentation of the library is unmatched. Consequently, multiple successful deep learning models could then be built and deployed without much debugging required. Besides the machine learning libraries, the popular signal processing library Librosa was used to handle the processing and transformation of audio files. Librosa is well documented and can handle complicated audio transformations in only a few lines of code.

7 Implementation

The development of this project can be broken down into 5 key stages: dataset selection and comprehension, data processing, model architecture and training, testing, and finally, GUI development.

7.1 Foreword

A lot of the time spent on this project was spent prior to the key stages on personal development; this involved developing a theoretical and practical understanding of machine learning techniques. In particular, two LinkedIn Learning courses by Fernandes (2019) and Geitgey (2017) both played a large role in understanding how the technology worked at both the practical and theoretical levels. Once a core understanding was developed,

consistent literature review of technical papers on speech emotion recognition helped to further shape the understanding towards the specific subject matter.

7.2 Dataset Selection and Comprehension

Prior to embarking on any practical work for this project, consideration was made to the type of data that would be required to use for both testing and training the speech emotion classifier. One initial idea was to collect and sample our own data, but the ongoing global pandemic made this option infeasible. In general, the choices for such a dataset were highly limited, with only one option standing out in particular; The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS). RAVDESS is an open-source dataset (Kaggle, 2019) consisting of 1440 unique audio clips of actors speaking a sentence whilst blatantly conveying one emotion out of: calm, happy, sad, angry, fearful, surprise, and disgust.

The selection of the RAVDESS dataset for this project meant that the emotions in scope were pre-determined, which, whilst not inherently detrimental, meant that prior work related to selecting the emotions best suited to the field of learning environments became redundant. Fortunately however, the more general emotions that had been selected prior were covered by the dataset with later developments showing that inferences about the more complicated in-scope emotions could be attained from the broad parent classes, as well as through studying the multi-class weighted predictions.

The dataset is already labelled such that the labels effectively describe the key content of the audio clip, as shown below in figure 2.
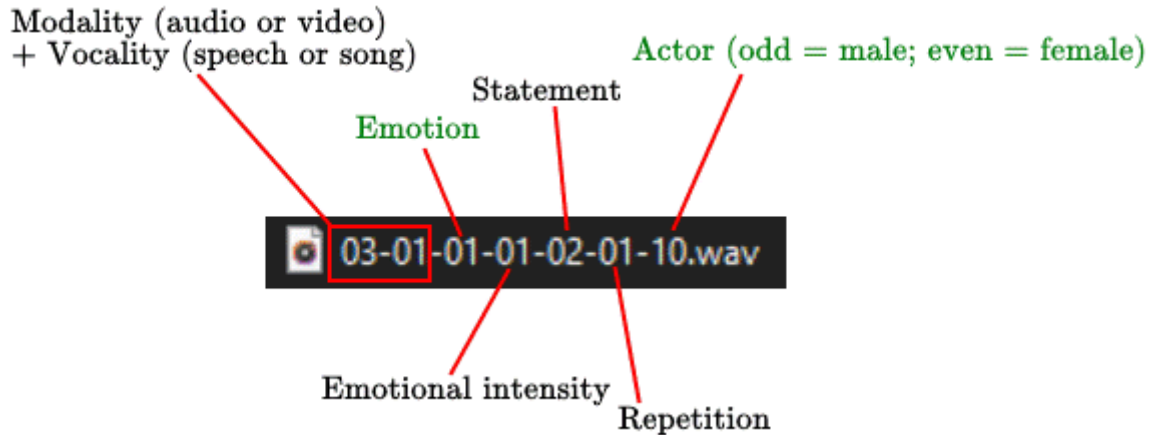

Figure 2 - an example audio file from the dataset; only the sections marked in green (emotion and gender) need to be taken into consideration.

Since the dataset comes labelled and is well structured, the only administrative task was to extract the key labels from each of the audio files; this was achieved through the `get_label_RAVDESS` helper function (code in the Appendix). At first glance, the emotional intensity label may seem like a factor to consider, however, due to the limitations in terms of the size of the dataset, creating additional categories would only serve to make the classification more ambiguous.

With regards to categorisation, this was initially set to include all of the default labelled emotions in the dataset, but later work revealed that this yielded worse results. The

likely explanation for this is that the distribution of emotions throughout the dataset is uneven, meaning that not enough data can be studied to build viable connections. Upon further studying the minority emotions (calm and surprise), it was found that these could feasibly be converted in efforts to improve the neutral and happy emotion categorisation respectively.

Original Label	Converted Label
Neutral	Neutral
Calm	**Neutral**
Happy	Happy
Sad	Sad
Angry	Angry
Fearful	Fearful
Disgust	Disgust
Surprised	**Happy**

Figure 3 - the final emotional labels used to train the model; in bold text are the conversions.

7.3 Data Processing

Since the dataset comes in the form of labelled audio (.wav) files, the main task is to convert these to spectrograms, in particular mel spectrograms, that can later be used to train the model. Using the mel scale as opposed to the hertz scale is important when working with human audio; the mel scale is a scale derived from tests with human listeners, such that the scale of pitches is judged to be equal in distance from one another.

The first task is converting the audio frequency from the hertz scale to the mel scale, this is given by the equation $m = 2595 \log_{10}\left(1 + \frac{Hz}{700}\right)$. Once this conversion is made, a mel spectrogram can be obtained from an audio signal by computing the squared magnitude of the Short-Time Fourier Transform (STFT) applied to a signal, substituting the frequency of the signal for the mel scale conversion. Finally, the mel spectrogram is scaled to decibel units according to $10 \times \log_{10}(mel\ spectrogram)$. This in turn corresponds to a weighted matrix, as in figure 4 below.

```
Shape: (128, 259, 1)
[[[-51.84466685]
  [-53.62410729]
  [-49.66521989]
  ...
  [-76.23304767]
  [-76.23304767]
  [-76.23304767]]

 [[-42.06875396]
  [-41.09654384]
  [-39.05476856]
  ...
```

Figure 4 - a log-scaled mel spectrogram obtained from an audio file from the dataset.
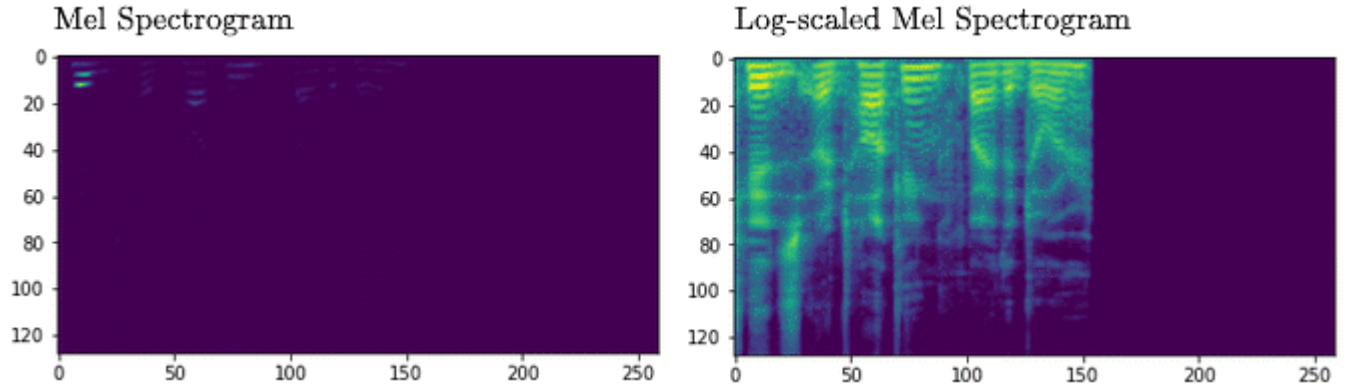

Figure 5 - illustrating the importance of the decibel conversion.

The last step of the data processing stage was creating the testing and training data through applying the labelling and spectrogram transformation functions to each audio file. The testing data was selected through a random sampling ($n = 180$) of assorted audio files containing all emotions in scope and all actors except actor 24. Actor 24 was excluded in an attempt to introduce some completely unknown data into the testing dataset.

7.4 Model Architecture & Training

The approach towards building the model involved researching existing models and simply experimenting with different variables. Research documented in the literature review, with particular regard to the paper by Venkataramanan & Rajomohan (2019), showed that the convolutional neural network approach, applied to visualisations of signals, achieved by far the best results. Convolutional neural networks work by analysing discrete features within a tensor (which the spectrogram, as a matrix, can easily be converted to), which is then abstracted to a feature map that contributes towards the input of the next layer. The reason that this type of neural network is feasible for this project is due to the conversion of the audio signal to the spectrograph, essentially transferring the problem from the audio domain to that of the visual.

The core architecture, consisting of 4 2-dimensional convolutional layers with max pooling, batch normalisation, dropout regularisation, and an exponential linear unit (ELU) activation function, therefore remained the same throughout the experiments. The Adam optimiser was selected due to the adaptive learning rate and categorical cross entropy was selected as the loss function (since the goal is to classify across multiple classes).

```
Model: "SER_CNN_2D_DJ"
_____
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 128, 259, 128)     1280

 batch_normalization (BatchNo (None, 128, 259, 128)    512

 activation (Activation)     (None, 128, 259, 128)     0

 max_pooling2d (MaxPooling2D) (None, 64, 129, 128)     0

 dropout (Dropout)           (None, 64, 129, 128)      0

 conv2d_1 (Conv2D)           (None, 64, 129, 128)      147584

 batch_normalization_1 (Batch (None, 64, 129, 128)     512

 activation_1 (Activation)   (None, 64, 129, 128)      0

 max_pooling2d_1 (MaxPooling2 (None, 16, 32, 128)      0

 dropout_1 (Dropout)         (None, 16, 32, 128)       0

 conv2d_2 (Conv2D)           (None, 16, 32, 64)        73792

 batch_normalization_2 (Batch (None, 16, 32, 64)       256

 activation_2 (Activation)   (None, 16, 32, 64)        0

 max_pooling2d_2 (MaxPooling2 (None, 4, 8, 64)         0

 dropout_2 (Dropout)         (None, 4, 8, 64)          0

 conv2d_3 (Conv2D)           (None, 4, 8, 64)          36928

 batch_normalization_3 (Batch (None, 4, 8, 64)         256

 activation_3 (Activation)   (None, 4, 8, 64)          0

 max_pooling2d_3 (MaxPooling2 (None, 1, 2, 64)         0

 dropout_3 (Dropout)         (None, 1, 2, 64)          0

 flatten (Flatten)           (None, 128)               0

 dense (Dense)               (None, 12)                1548
=================================================================
Total params: 262,668
Trainable params: 261,900
Non-trainable params: 768
```

Figure 6 - summary of the complete speech emotion recognition model.

Once the final design had been decided, the model was then trained several times using a multitude of different batch and epoch size combinations. The main goal was to strike a balance between optimisation and the over-fitting problem; 125 epochs at batch size 16 produced the most optimal results of the tests performed (including in terms of computational efficiency).

The training steps and code can be viewed from the SER Classifier Jupyter Notebook with link in the appendix.

7.5   Testing

After the final model was trained, the test dataset was loaded and tested blindly on the model, yielding 180 predictions. The categorical accuracy across this dataset was 68.88%, which is a comparatively high score. Whilst the accuracy is most certainly inflated due to the similarities of the test dataset (all of the speakers being North American actors speaking English and saying similar sentences), the results are still impressive considering both the lack of training data and the fact that the majority of the test data consisted of actors that had not been used to train the model at all.

Actual value	Prediction
22. female sad	22. female neutral
28. female angry	28. female neutral
45. female disgust	45. male sad
72. male happy	72. female happy
80. male sad	80. male neutral
100. male fearful	100. male happy
152. female angry	152. male angry

Figure 7 - a sample of the incorrect predictions, showing reasonable mistakes (i.e. emotional similarity or right emotion; wrong gender). *Values taken from SER_Classifier notebook.*

After the model had been tested on sample data from the same dataset, the next step was to test it on completely unique audio. The main source of this audio was from myself and from family and friends. From personal tests using my own voice, the results yielded were positively surprising. In particular, the model had shown to have learned the negative emotions particularly well; correctly predicting male anger for each of the sample clips (included in the GitHub repository). What was also impressive was that the model wasn't just predicting the correct emotion with a slight edge over the other categories; the model would predict anger with over 90% accuracy over the other categories (with male disgust, female anger, and male sad often taking up the remainder of the prediction for this category).
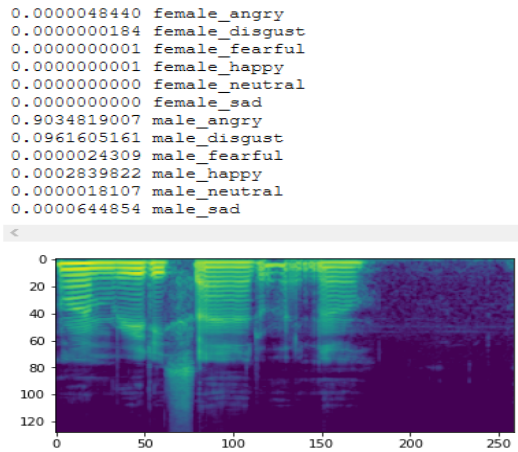
```
0.0000048440 female_angry
0.0000000184 female_disgust
0.0000000001 female_fearful
0.0000000001 female_happy
0.0000000000 female_neutral
0.0000000000 female_sad
0.9034819007 male_angry
0.0961605161 male_disgust
0.0000024309 male_fearful
0.0002839822 male_happy
0.0000018107 male_neutral
0.0000644854 male_sad
```


Figure 8 - output predictions using an audio clip of myself talking angrily, available through the appendix.

A particularly interesting test was conducted using a friend of mine with diagnosed Asperger's syndrome who notably struggles to perceive the emotions of others. For the test, she was asked to send some audio clips of herself conveying an impression of the various emotions in scope. The overall accuracy initially seemed poor, with some predictions failing to even fit the binary classification of positive/negative. Upon looking further into this and analysing the audio files together, we agreed that actually she herself had misclassified emotions that the predictions from the model were more reflective of.
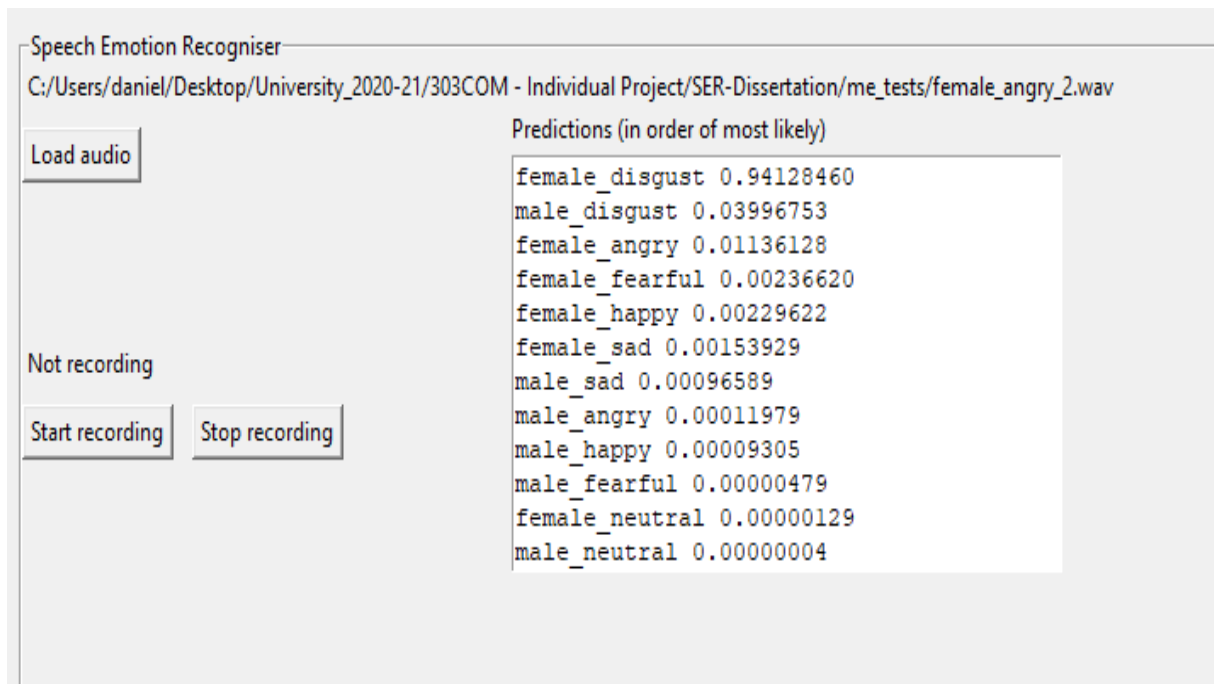

Figure 9 - an example of one of the tests on a friend with autism who had herself misclassified the audio (and a preview of the GUI tool).

The model was also tested on German and Swiss German language audio to check for any variance in results when straying from English. In these tests, sentences were said exclusively in the respective language whilst an emotion from the list was conveyed. The

model generally performed well, correctly predicting the emotion for angry / sad / disgust but made some errors on the positive emotions tested. It is worth noting that in the cases where it failed, the target emotion was always within at least the top 5 prediction classes.

7.6   GUI Development

Once the model had been developed, tested, and evaluated, the final step of this project was to deploy the model into a functional tool for end-users. The primary objective of the tool was that a user with next to no technical background could easily obtain results from the model without having to write code or tweak files. The core functionality of the tool was relatively easy to determine; the user should be able to analyse pre-recorded audio and should be able to record audio directly so that the audio would be analysed and the results clearly visible. Since accessibility was an important factor, the tool was developed with simplicity and efficiency in mind. To achieve this, the tool was built natively within Python using Tkinter and with minimal libraries / graphics. The tool at first glance is shown in figure 10 and figure 9 shows a working example.
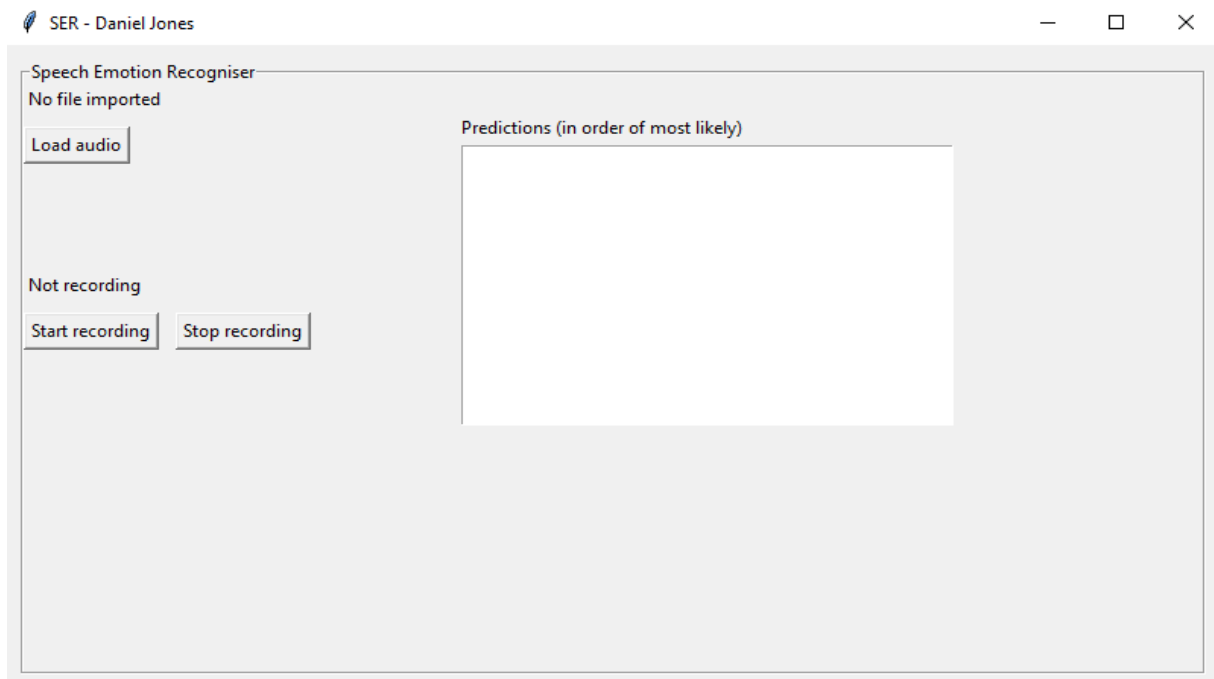

Figure 10 - The tool upon execution, showing the two key functions as buttons.

8   Evaluation

The end result of this project is both a promising speech emotion recognition classifier (by account of the tested accuracy, both on the test dataset, and according to personal tests) and a fully functional tool that can be used by end users in a variety of different environments. In this section, I will relate both the successes and failures of the project back to the original research objectives as defined in the project proposal document under the "Primary Research Plan".

8.1 Research Objectives

The first and primary research objective was to create a capable speech emotion classifier based on current literature. By account of the various tests and experiments, I believe that a viable model has been successfully built and trained utilising methods that produce competitive results. This has been achieved through constant literature review and through personal research on theoretical machine learning and machine learning best practices. The end model and all of the associated work contained in the notebook should serve as a clear and concise approach to tackling the problem, demonstrating what works and what could be improved.

The next research objective was to improve and modify the model through conducting formal tests and through conducting research in learning environments. Unfortunately this objective was unsuccessful, as these tests could not take place. However, I do believe that the end product is in a capable condition, where it could easily be picked up from where it is currently left to be further developed. Since the deployment of the product and core architecture are taken care of, this would be only be a matter of additional data collection, processing, and re-training, which are comparatively inexpensive.

The final research objective, to create a tool that could be used within learning environments by end users has, I believe, been achieved. A GUI tool has been produced that is very lightweight, yields results quickly, and requires little dependencies once deployed. The GUI is intentionally simple, and through testing with a user with autism, has proven to add at least some unique and useful insights. With regards to the final GUI, since it is intentionally simple and lightweight (according to research regarding technology in learning environments), I do not believe that it could be further improved without further adjusting the model or gaining real world feedback from learning environments.

Overall I believe that good work has been made in achieving the research objectives, whilst not achieving them all perfectly, progress has been made and the end result is in a particularly nice stage that I intend to personally pick up, carry on with, and develop further in a future project.

9 Discussion and Conclusion

The project was originally structured around three key pillars; building a speech emotion recognition classifier, building an accessible tool to utilise the classifier, and testing it within learning environments to document the findings. The idea behind this structure was that, upon completion of any stage, there would at least be something to document and contribute back to the field. Unfortunately, the third stage of the project was not able to take place due largely to the global pandemic, but also due to a personal underestimation of the length of time required to even begin understanding the fields of signal processing and machine learning. However, the amount of personal development and new knowledge I have been able to gain through undertaking this project have been of immeasurable value to me.

The main strength of the project is the final product itself, which I believe offers a functionality that is, in some cases, insightful enough to be used as is. For example, the tests performed with my friend with autism directly show that the tool can offer useful insights previously unobtainable. Furthermore, the model architecture and trained model both provide a useful checkpoint that could easily be enhanced further through additional data volume. If nothing else, I believe that the model and final application developed alongside this project provide a clear and well documented framework to anyone looking to enter speech emotion recognition or gain a head start in developing their own approach.

Was I to embark on this project from the beginning again, I would certainly choose less ambitious targets and goals. The core development of the project took far more time than originally anticipated, which I should have taken into consideration when planning the project. I would also change my approach to form a better foundation before starting on the practical work, because through doing this, a lot of debugging time could have been saved and concentrated elsewhere, resulting in a better overall project.

Despite the lack of new research linked to the usage of the tool in learning environments, the tool itself has been successfully developed and documented, and could easily be picked up from where it has been left off to continue further work with the project. Combining this with the vast amount I have learned makes this project a huge personal success.

10 Project Management

A waterfall-like project management methodology, namely IPERKA, was selected due to the structure of the project (as discussed in the **Discussion** section) and because of prior work experience using the methodology. IPERKA stands for "Informieren, Planen, Entscheiden, Realisieren, Kontrollieren, and Auswerten" which translates to "informing, planning, deciding, implementing, controlling, and evaluating". The methodology places lots of importance on project planning via the first three stages, which I felt was beneficial here due to the lack of experience prior to commencing the project.

Regular weekly meetings with my supervisor were organised and held over Microsoft Teams each week to discuss both ideas, problems, and progress. I found these meetings extremely useful as they motivated me to stay on track, organise my thoughts, and discuss the feasibility of ideas. I dedicated approximately 2 days each week (requiring more particularly towards the end of the project) towards working on the project. An example of the notes taken during these sessions is shown below.
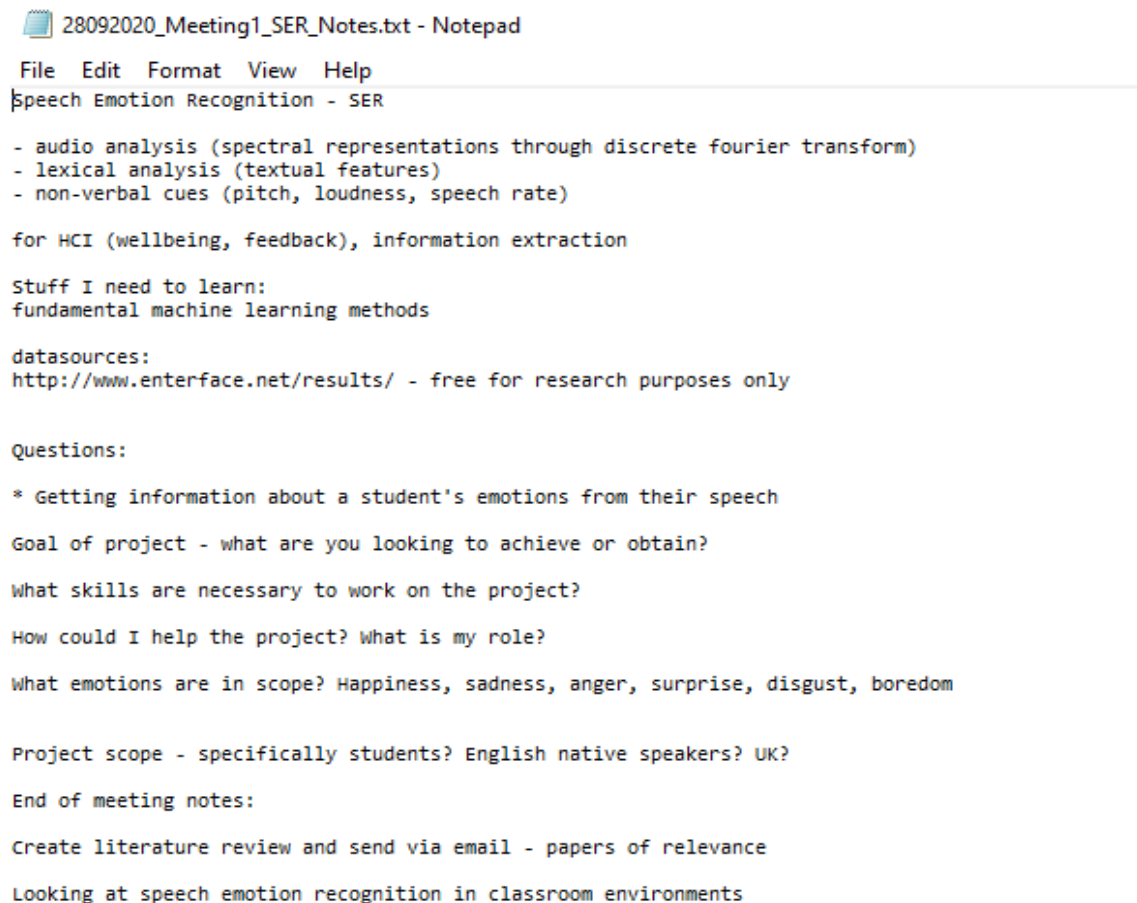

Figure 11 - an example of notes taken from the very first meeting with my supervisor, discussing initial requirements and ideas.

10.1 Gantt Chart

To help stick to a schedule and plan, I worked predominantly according to a Gantt chart, drafted in the early stages of the project. Whilst I wasn't able to exactly follow the Gantt chart, the structure and clarity provided by it proved to be worthwhile throughout the project, highlighting a clear path towards the final goals.
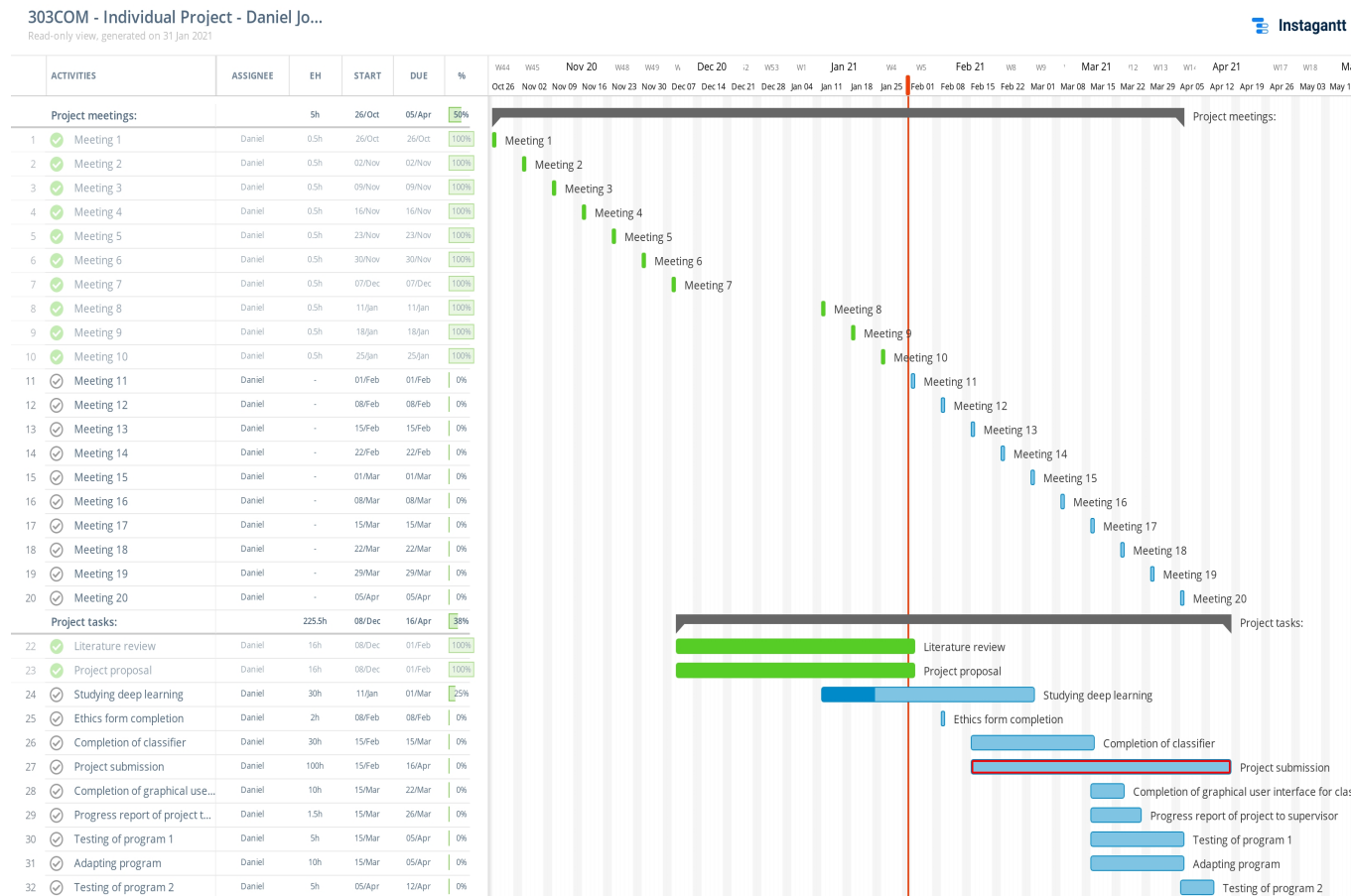

Figure 12 - A screen-shot taken from the Gantt chart showing meeting plans and tasks to be completed.
Link to full size image
Link to latest edition of the chart

10.2 Presentation

Several progress presentations took place throughout the project life-cycle, particularly when milestone accomplishments (such as finishing the model and the GUI) had been accomplished. In the first presentation, a now redundant feature was demonstrated with positive feedback and with nothing to add or change. In the second presentation, the primary model architecture was demonstrated, receiving positive feedback. In the second presentation, the main comment was on the emphasis of deploying the model into a GUI tool with some discussion about the core functionality of the tool. In the final presentation, the GUI tool and the core functionality of the tool was presented through demonstrating the recognition capabilities using my own voice along with some pre-recorded clips. The feedback was completely positive with nothing to add or change.

10.3 Consideration of social, legal, professional, and ethical issues

The potential social, legal, professional, and ethical issues that could develop from this project must certainly be paid careful attention to. The implications of a tool that can predict emotions from speech could have negative implications if careful disclaimers are not disclosed regarding the project. That is to say that, whilst emotion can be predicted with a relatively high accuracy, the key word is "predicted"; the model of course is prone to errors and even results that clearly do not make sense. An example of misuse would involve any attempt to use the project to draw absolute conclusions about speech emotion, such that the results would then influence additional decision making. Contained within the project repository is a disclaimer highlighting some of the key points (what the tool is and isn't) along with a fair usage license (MIT License). Essentially, the tool is a novel device that is made exclusively with guidance in mind. If the tool can be useful, for example to users with relevant disabilities or within learning environments, then it should only be used to prompt (through the range of the predictions) or to offer an alternative perspective.
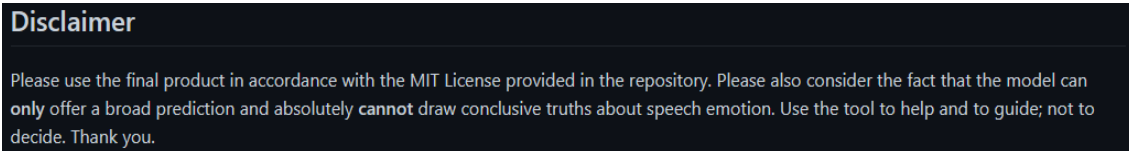

Figure 12 - the disclaimer included within the README.md file of the associated GitHub repository.

11   References

- Cho, V., Mansfield, K. C., Claughton, J. (2020). The past and future technology in classroom management and school discipline: A systematic review. *Teaching and Teacher Education, 90*, 103037. https://doi.org/10.1016/j.tate.2020.103037

- Angrick, M., Herff, C., Johnson, G., Shih, J., Krusienski, D., Schultz, T. (2019). Interpretation of convolutional neural networks for speech spectrogram regression from intracranial recordings. *Neurocomputing*, 342, 145-151.

- Sainath, T. N., Mohamed, A. R., Kingsbury, B., Ramabhadran, B. (2013). Deep convolutional neural networks for LVCSR. In *2013 IEEE international conference on acoustics, speech and signal processing* (pp. 8614-8618). https://ieeexplore.ieee.org/abstract/document/6639347

- Venkataramanan, K., Rajomohan, H. R. (2019). Emotion Recognition from Speech *arXiv preprint arXiv:1912.10458*. https://arxiv.org/pdf/1912.10458.pdf

- Kerkeni, L., Serrestou, Y., Mbarki, M., Raoof, K., Mahjoub, M.A., Cleder, C. (2019). Automatic Speech Emotion Recognition Using Machine Learning. In *Social media and machine learning*. IntechOpen. https://doi.org/10.5772/intechopen.84856

- Kwon, S. (2020). A CNN-assisted enhanced audio signal processing for speech emotion recognition. *Sensors 20(1), 183*. http://doi.org/10.3390/s20010183

- Harrell, S., Bynum, Y. (2018). Factors affecting technology integration in the classroom. *Alabama Journal of Educational Leadership*, 5, 12-18

- Lim, W., Jang, D., Lee, T. (2016). Speech emotion recognition using convolutional and recurrent neural networks. In *2016 Asia-Pacific signal and information processing association annual summit and conference (APSIPA)* (pp. 1-4). IEEE. paper2016.pdf

- Munshi, A., Rajendran, R., Ocumpaugh, J., Biswas, G., Baker, R. S., Paquette, L. (2018). Modeling learners' cognitive and affective states to scaffold SRL in open-ended learning environments. In *Proceedings of the 26th conference on user modeling, adaptation and personalization*(pp. 131-138). https://doi.org/10.1145/3209219.3209241

- Fernandes, J. (2019, October). *PyTorch Essential Training: Deep Learning* [Video]. LinkedIn. https://www.linkedin.com/learning/pytorch-essential-training-deep-learning/

- Geitgey, A. (2017, August). *Building and Deploying Deep Learning Applications with TensorFlow* [Video]. LinkedIn. https://www.linkedin.com/learning/building-and-deploying-deep-learning-applications-with-tensorflow/what-is-tensorflow

- Kaggle. (2019). *RAVDESS Emotional speech audio* https://www.kaggle.com/uwrfkaggler/ravdess-emotional-speech-audio

12 Appendix

- **https://github.com/danielj0nes/SER-Dissertation** - GitHub repository containing all of the referenced files, such as the Jupyter Notebook SER_Classifier.ipynb (where the model was built) and SER_Application.py where the model was deployed. The repository also contains all of the personal audio files using for testing and documenting results, the source code and images for this document, and will be made available from the 30th of April 2021 18:00 GMT (according to final submission deadline).

- **RAVDESS Emotional speech audio** - Primary testing and training data used in the project (available for download via the link).