Predicting emotion in music: Harnessing multimodal artificial intelligence to predict emotional dominance, valence and arousal

Lara Elvevåg

25th February 2024

TABLE OF CONTENTS

Acknowledgements ………………………………………………………. page 3

Abstract ………………………………………………………. page 4

Introduction ………………………………………………………. page 5

Methods ………………………………………………………. page 10

Results ………………………………………………………. page 16

Discussion ………………………………………………………. page 26

Conclusion ………………………………………………………. page 27

References ………………………………………………………. page 28

ACKNOWLEDGEMENTS

This project would not have been possible without the fantastic mentorship of Dr. Chelsea Chandler. I met her many years ago when she was studying for her PhD and I feel enormously privileged that she agreed to be my teacher for this exciting project. She has guided me regularly and meticulously for the last year and it has been a spectacularly interesting and enjoyable learning experience. Therefore my biggest thanks and gratitude go to Chelsea, thank you!

Since the age of eight, I have been a student at the online school FunTech (https://funtech.co.uk/) following their intense fast-track coding curriculum. Last summer, when I was thirteen, I sat my computer science Cambridge iGCSE exams and got a 96% mark. I also have computer science as my elective subject at Sommerlyst skole, and am very grateful to my teacher Martin Lamvik there who has supported my interests and provided much-appreciated encouragement.

Recently I had a fabulous work experience week at Kongsberg Satellite Services (KSAT) which was truly amazing. There I worked on revamping their PDF report generation software for documenting possible events (such as oil spills, moving ice or illegal ship traffic) from satellite images using Python programming. Thank you to KSAT for this wonderful opportunity. This experience certainly confirms my commitment to pursuing computer science in my career. I hope that those who read this report enjoy it as much as I enjoyed the process of doing this project.

I confirm that this report reflects my own work and that all the writing is my own.

The Jupyter notebooks coded specifically for this project are available at my GitHub repository (professorlara):
https://github.com/professorlara/Predicting-emotions-in-music-using-artificial-intelligence

ABSTRACT

Music has a profound impact on human emotion, with both lyrics and melody playing pivotal roles in eliciting powerful feelings. These emotions may be evoked through the affective qualities of songs, which we can automatically predict with a variety of artificial intelligence methods that include natural language processing (NLP) and machine learning (ML). Generally, the modalities of acoustics (sound) and language (the lyrics) are analysed separately when making these predictions, so it is unknown what combining sound and text would result in. This project sought to determine whether the prediction of the sentiment associated with a song would improve if acoustics and language were combined. To answer this question, this study analysed approximately 10,000 songs that were labelled with sentiment ratings by humans using NLP and ML models (for example a neural network) to establish the accuracy of predicting three dimensions of emotion: dominance (the amount of control), arousal (the intensity), and valence (the pleasantness). To do this, a dataset was created with 10,000 English songs by combining three datasets that are freely available: (i) the musical sentiment dataset (MuSe), (ii) the Kaggle song lyrics dataset and (iii) AcousticBrainz's API. Importantly, these datasets contain varied features associated with the songs, namely ratings of the three dimensions of sentiment, song lyrics and audio features, respectively. It was anticipated that the accuracy of prediction would be higher when combining two data types, compared to just one. As expected, combining more than one modality of data enriches and improves the sentiment prediction. Current applications in the music industry, such as music recommendations, are likely to match customers' preferences better if a multimodal approach to analysing music is used. If it is possible to predict the emotions that a song will evoke, in the future it will likely be possible for artificial intelligence to tailor music recommendations more accurately.

INTRODUCTION

Listening to music evokes emotions and feelings in us. We all have the experience of feeling happy or sad based on the music we are listening to. Also, music can arouse us such that we feel calm or excited. It can also have the effect of us feeling controlled by the music. These features are measurable and are called dominance (the amount of control), arousal (the intensity: calm or excited), and valence (the pleasantness: happy or sad) respectively (Warriner et al., 2013). Importantly, these emotional features tend to be consistent and universal in humans, such that a piece of music that evokes sadness in one person is also likely to evoke sadness in another person. Naturally, the intensity of these emotions will vary to some extent. Nonetheless, with enough data, in principle, these sentiment labels can be predicted. For example, the music streaming service Spotify tracks its users' listening habits to make recommendations based on what kinds of songs users listen to. In part, the success of their method is likely because the sentiment of the songs is evaluated in the algorithms that are used to produce recommendations (Björklund et al., 2022). Put differently, they examine the emotions associated with each song to predict what kind of song the user will likely want to listen to next. Songs (comprised of lyrics and melody) contain numerous features that can be used to measure the emotions they evoke. Examples of features include the number of words, number of adjectives, beats per minute, key signature, and wavelengths. Given the vast quantity of complex data and features, Spotify uses artificial intelligence methods to classify songs as happy, sad, calm or upbeat, and can then use this information to suggest similar songs to the user.

ARTIFICIAL INTELLIGENCE

Artificial intelligence (AI) is a general term for computer systems that show intelligent behaviour (see terms and definitions in Table 1). There is a lot of excitement these days about AI and machine learning. Machine Learning (ML) is a subset of AI that uses statistical algorithms to learn from data. Natural Language Processing (NLP) is another subset of AI that uses both statistical and linguistic knowledge to understand human language. The potential applications of AI are vast, notably automating tasks and improving decision-making processes. It is a massive, fast-moving, multi-billion-dollar industry.

Table 1: Definitions of key terms (derived from Chandler et al., 2020 & Warriner et al., 2013).

Artificial intelligence (AI)	AI is a general term for computer systems that exhibit intelligent behaviour and can learn, explain, and advise their users.
Machine learning (ML)	ML is a subset of AI that harnesses statistical algorithms to learn *features* of data and the associated importance of each (or simply the associated importance of user-defined features). Once the features and their weights, as well as other *hyperparameters* are set, the model can predict some outcome or clinical classification on new, unseen data.
Natural Language Processing (NLP)	NLP is another type of AI that incorporates both statistical and linguistic knowledge to understand human language.

Neural Network	A system of nodes, composed in layers, where each node learns some nonlinear equation on some subset of training data and when all nodes are combined, a categorical or real-valued output can be computed. Modern neural networks are deep, meaning they have hundreds to thousands of nodes and layers and are trained on large datasets.
Sentiment analysis	Sentiment analysis is a branch of AI that analyses a piece of data to predict the emotion that is associated with the data.
Valence, arousal and dominance	Some important aspects of sentiment are arousal, valence and dominance. Arousal is the intensity of emotion and ranges from calm (low) to excited (high). Valence is the level of pleasantness that an event generates. Finally, dominance is the level of control exerted by a stimulus.

MACHINE LEARNING METHODS

Numerous machine learning methods exist notably traditional models (such as linear and logistic regression, support vector machines and decision trees) and more contemporary models such as deep neural networks (Choi et al., 2020). In this project, both types of models were explored.

Traditional machine learning methods use statistical information and complex correlation functions to predict values or to classify. One type of traditional machine learning model is a decision tree. Regarding Figure 1 below, decision trees are hierarchical tree-like models that are composed of a root node (start point), decision nodes and leaf nodes (endpoint). Often the root node is a specific question that leads to branches holding potential answers or further questions. This process repeats until the data reaches a leaf node. Decision trees are a popular machine learning model because they mimic how humans think when making a decision, so their logic is easier to understand compared to other more complicated models.

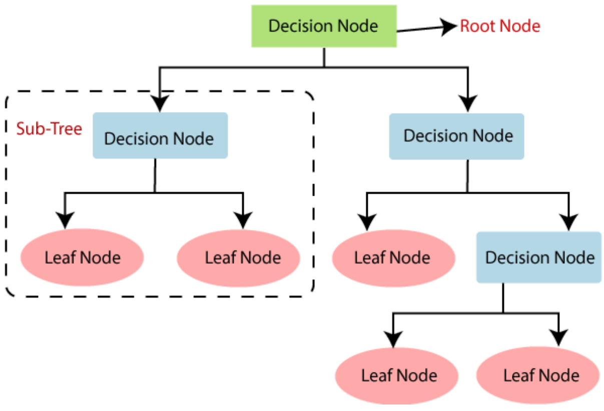

Figure 1: Simplified representation of a decision tree (from Panigrahi, 2023).

One way to categorise models is supervised and unsupervised. In a supervised model, the input data is labelled. The model goes through a training process where it has to make predictions and it is corrected when the predictions are wrong. This process is repeated several times until the model achieves the desired accuracy. Then the model is tested on the test data. In an unsupervised approach, the model is given unlabeled data and asked to find trends on its own. All of the eight traditional models explored in this project were supervised, and they are described below:

Linear regression is a type of regression method which takes in two variables and uses a best-fit line (regression line) to see how well they correlate. **Ridge regression** is a type of regression model that is based on linear regression, but with a modification of the loss parameter. **Lasso (Least Absolute Shrinkage and Selection Operator) regression** is also an extension of linear regression that produces accurate yet simple models that have few parameters. The **Gradient Boosting Regressor** is an ensemble of decision trees, and the final prediction is calculated from the average of all the decision trees' predictions. The **AdaBoost (Adaptive Boosting) regressor** is an ensemble model where additional regressors are added for each instance during training, and the weights of these regressors are adjusted based on the error of the current prediction. A **multi-layer perceptron** is a neural network that has at least three layers. Even though it is classified as a deep learning method, it is not as complicated as some of the other neural networks developed in this project. Therefore it has been grouped under the 'simple' traditional models in this report. **Support Vector Regression** is the regression variant of Support Vector Machines (that are classifiers). Support Vector Machines group values into classes by finding a hyperplane inside a high-dimensional figure that will best separate the values. A similar approach is used by Support Vector Regression for regression analysis. Finally, **random forests** are ensembles of decision trees that are each built randomly.

Regarding Figure 2 below, a neural network is a method used in AI that teaches computers to process data in a way that is inspired by the human brain. It is a system of nodes that is composed of layers, where each node learns an equation on some subset of training data. When all the nodes are combined, the output is computed. Modern neural networks have hundreds to thousands of nodes and layers and are trained on large datasets. An issue when training a neural network is overfitting. This means that the model learns the statistical noise in the training data, which causes the models to generalise poorly leading to poor performance when tested on test data. Therefore, a regularisation method called 'dropout' is often implemented to reduce the risk of overfitting. When dropout is used, some nodes are randomly ignored or 'dropped out' during training which makes it more robust. (For an overview of machine learning modelling methods see Ng, 2023).

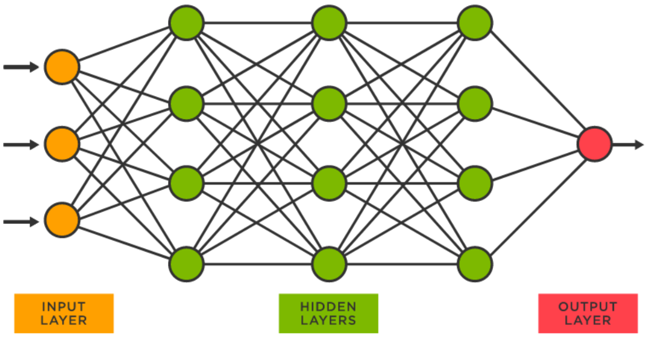

Figure 2: Simplified representation of a neural network (from Kumar, 2023).

MULTIMODAL MODELLING

Multimodal modelling is combining two or more modalities (e.g. language, audio and images) in a model. In this project, the two modalities that were used were text (songs' lyrics) and audio (pre-computed audio features). Multimodal modelling techniques allow models to process and analyse data from several modalities, creating a more complete and accurate understanding of the data (Rosidi., 2023).

One way of combining models is ensemble modelling. Ensemble modelling operates by combining multiple diverse models to create one prediction (Opitz & Maclin, 1999). By combining several diverse models, the strengths of one model will compensate for the other's weaknesses. This results in a more robust and accurate final prediction. A majority vote or a weighted average can combine the individual models' predictions. A weighted average means that the models with a smaller error were weighed higher (and accounted for more of the final prediction) compared to the models that had higher error rates. As mentioned in the Machine Learning Methods section above, random forests are an example of ensemble models. Random forests combine multiple decision trees that are all trained on different parts of the dataset. The final output of the model is determined by taking the average of all the trees.

Another modelling approach is multimodal learning, where the goal is to combine information from different modalities into one model. This approach is useful if the modalities provide complementary information about each other. In the case of this study, that would mean training one model on both the language and acoustic features. This approach would work well if one language feature and one audio feature have a strong relationship that is beneficial to the prediction.

SENTIMENT ANALYSIS

Sentiment analysis is the formal process of measuring the emotion associated with text, audio and other modalities (such as images and video). Previously, machine learning models have been used to analyse pieces of data to predict dimensions of the emotion associated with it. Important prediction variables in sentiment analysis are arousal, valence and dominance (for definitions, see Table 1 above). Arousal is the intensity of emotion and ranges from calm (low) to excited (high). Valence is the level of pleasantness that an event generates. Finally, dominance is the level of control exerted by a stimulus.

However, current methods for sentiment recognition are unable to detect the nuances in language, for example sarcasm and irony. This may be because the approaches are generally only examining one aspect of sentiment at a time (either lyrics or audio) or are only predicting sentiment in a binary manner (positive or negative). Indeed, the accuracy of sentiment recognition is likely to be superior if more than one datastream is examined simultaneously to compute the sentiment. Therefore, this project sought to combine two modalities (specifically text and audio features) to evaluate if the accuracy of predicting sentiment labels (valence, arousal and dominance) of songs is superior to a prediction model based on just one modality. Specifically, this project built several ML models that use both the lyrics and the audio features (i.e. multimodal) and examined whether this would more accurately determine aspects of a song's sentiment compared to unimodal models, or even pre-trained models of sentiment.

RELATED WORK

To the best of my knowledge, the approach used in this project is unique, as there is no research reported on harnessing multimodal sentiment analysis to use both audio and language features to predict arousal, valence and dominance in songs. Multimodal sentiment analysis is a relatively new field (Soleymani et al., 2017), and few studies have been conducted where multimodal sentiment analysis has been used to analyse music. The first paper that mentions the term was published in 2011 (Morency et al., 2011) where they analysed videos (that include acoustic, visual and language features) to predict how happy the subject in the video was. In terms of music, studies have mostly focused on classifying songs as positive or negative (i.e., happy or sad) in a binary manner. There have been several interesting findings in this field recently. For example, a study from 2016 used multimodal sentiment analysis on songs (Abburi et al., 2016) and found that analysing the first 30 seconds of the songs produced superior classifications of the two sentiments than analysing the entire song. However, their dataset was small (<100 songs) and the songs were in Telugu. A study from 2018 used multimodal deep learning to classify music by genre (Oramas et al., 2018). Their approach is interesting as they used the audio tracks, text reviews of the songs and cover art images to train their models. However, they did not include the songs' lyrics as a feature for the model to be trained on. As far as I am aware, there has been no research so far published on using multimodal artificial intelligence to predict emotional dominance, valence and arousal. However, in 2017 a study was done where arousal, valence and dominance were jointly predicted using acoustic features (Parthasarathy & Busso, 2017). Even though this project only examined one modality (audio), papers like this set the stage for research such as this to be conducted.

The **main goal** of this project was to improve the accuracy of sentiment recognition in songs by using a multimodal approach that combines language and acoustic features. The **first hypothesis** was that if unimodal sentiment models were ensembled, the final prediction would be more accurate than the predictions of these individual models alone. The **second hypothesis** was that if a multimodal model (that takes in both acoustic and language features) was created, the final prediction would be more accurate than any unimodal model. Finally, the **third hypothesis** was that the ensemble and multimodal models' final predictions would be more accurate than any pre-trained fine-tuned model. The **dependent variable** was the performance of the model when evaluated by making predictions on the test dataset. This was measured by the mean absolute error, which is a proxy of how accurate the predictions the model makes are. The **independent variables** were the features extracted and the type of model that was making the predictions. Several different model types were evaluated, and the utilized feature and hyperparameters of said models were found using machine learning methods (namely feature selection and grid search).

METHODS

The goal of this project was to develop models that accurately predict three dimensions of sentiment (dominance, arousal and valence) using multimodal data (acoustics and language). In total, throughout this project, eight different models were developed to predict these dimensions of sentiment. These include two acoustic models based on the auditory signal from the songs, four language models based on the lyrics in the songs and two multimodal models (based on both acoustics and language).

DATASET

Three different datasets were used:

(1) The musical sentiment (MuSe) dataset contains about 78,000 unique songs with arousal, valence and dominance ratings (Akiki & Burghardt, 2021). The ratings were on a scale of 0.2 to 8.5.

(2) The Kaggle lyrics dataset contains songs with lyrics from over 4000 artists (Shah, 2021).

(3) The AcousticBrainz API (application programming interface) was used to extract pre-computed acoustic features for the songs (Porter et al., 2015).

These three datasets were combined, resulting in 10,000 unique songs with lyrics, labels, and acoustic features. Then the data were split into train (70%), validate (15%) and test (15%) datasets. The reason for this is to be able to use a portion of the data to train and optimise the language and acoustic models, and the remaining data to test the accuracy of the models. In the development of machine learning models, the models are often trained on the train dataset and tested on the validate dataset, iteratively updating parameters until one achieves a decent accuracy. Once one has determined the features, model architecture, and hyperparameters that optimise the accuracy, it is then retrained on all data from the train and validate sets, and tested on the held-out test dataset to determine the accuracy.

DATA DISTRIBUTION

The MuSe dataset contains sentiment information derived from the social tags given to that song on the website Last.fm (an online music database), derived through the Warriner et al. (2013) database, and expressed across the three dimensions: dominance, valence, and arousal.

The range of possible values for dominance was 0.23 - 7.44, the average was 5.40 and the standard deviation was 1.04. For valence the range was 0.24 - 8.47, the average was 5.72 and the standard deviation was 1.47. Finally, for arousal the range was 0.13 - 7.27, the average was 4.31 and the standard deviation was 1.05. Density plots showing the distribution of the three labels are given below in Figure 3.

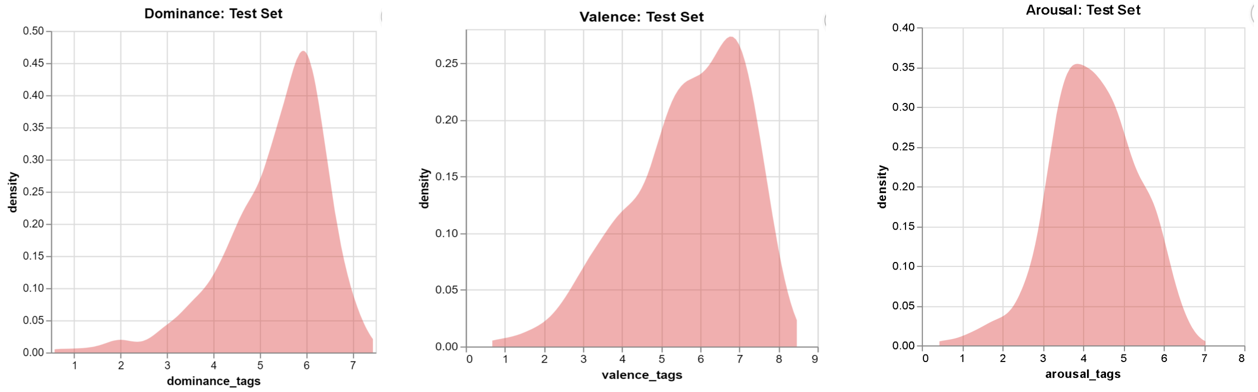

Figure 3: Density plot of the distribution of values for the dominance, valence and arousal ratings.

The density plots show that the range of values in the three sentiment labels was quite limited. This impacted the study, as there was a limited number of songs with extreme ratings (either very low or very high). Even though the axes are different across the three labels, it is apparent that the valence label has a wider range and more even distribution of values compared to the other two sentiment labels. It is also interesting to note that the arousal label has the smallest range of values. The density plot for the dominance label has the steepest peak, which suggests that the range of values is not distributed evenly and that a high percentage of the values lie in the interval between 5 and 6. Smaller variability in a prediction variable may cause machine learning models to be unable to learn the nuances of the relationship between various features and the prediction variable.

FEATURES

Both the acoustic and language-based models developed in this project were feature-based, and some of these features are described below.

(1) Acoustic features:

Since raw audio files for songs are not freely available, the acoustic features were retrieved from an API that contains pre-computed acoustic measurements. Specifically, these features were pre-computed by acousticbrainz (Porter et al., 2015), and are broken down into three core categories: low-level, rhythm, and tonal features.

The low-level features include key signature, length, loudness and mel frequency cepstral coefficients (MFCCs) of the songs, among many others. MFCCs are coefficients derived from sound waves and computed using Fourier transforms and logarithms (Sato & Obuchi, 2007). Another low-level feature is the low-level spectral contrast coefficient. The spectral contrast coefficient is a measure of the distance between the peaks and valleys in a sound wave (Yang et al., 2003). Next are rhythm features, which include the number of beats in the song, beats per minute and danceability. Beats per minute is an important measurement regarding the tempo of a song, which also affects the danceability measure. The last type of acoustic feature are tonal features. These include number of chords, strength of chords, key signature and whether the song is in major or minor. This is relevant to sentiment prediction because songs in major keys tend to be happier, whereas songs in a minor key are often classified as negative and sad.

(2) Language features:

Unlike the acoustic dataset, the lyrics dataset did not include precomputed features. Therefore, features such as the number of words, the number of unique words, the lexical richness, the content density, the frequency of various parts of speech types, and the number of named entities were extracted from the dataset for each song.

The SpaCy model (Vasiliev, 2020) was used to retrieve the named entities from the dataset. Named entities are a word class of proper nouns. For example places, people and addresses. The reason for extracting this feature was that it seemed logical to assume that songs that contained several named entities would be more personal (which could relate to the song's sentiment). Another feature that was computed is lexical richness. Lexical richness is a measure of the diversity in a text and is often calculated using the type-token ratio (Toruella & Capsada, 2013). This ratio is found by dividing the total number of different words (types) by the total number of words (tokens). A high type-token ratio (close to 1) suggests high lexical richness, whereas a low value (close to 0) suggests that there is little variation in the text. Another important feature that was extracted is the content density. This metric is quite similar to lexical richness. The content density of a text was calculated by dividing the total number of verbs, nouns, adjectives and adverbs by the total number of words in the song.

(3) Sentiment values from pre-trained models:

A pre-trained model is a model that has been previously trained on a large dataset and can then be fine-tuned to solve a specific task. In this study, pre-trained models were used to compute sentiment values for each of the songs in the dataset. These sentiment values were used as features that the models developed in this project use to predict the sentiment labels. First, the Distilbert sentiment model (distilbert-base-uncased-finetuned-sst-2-english) was used to calculate sentiment values for the songs' lyrics. (Sanh et al., 2019). This model is a text classifier that outputs two values (how positive and negative the text is) that range between 0 and 1.

The RoBERTa sentiment model (siebert/sentiment-roberta-large-english) was also used to calculate how positive and negative the song lyrics in the dataset were (Liu et al., 2019). This model is a version of the popular RoBERTa model that is trained specifically for sentiment analysis. Another type of sentiment analyzer that was used is VADER. VADER is a rule-based model for sentiment that was created to deal with text from social media. (Hutto & Gilbert, 2014). The model outputs three sentiment values: how positive, neutral and negative the text is. These values range between 0 and 1. In addition, the VADER model computes the average sentiment of the text. This value ranges between -1 and 1.

FEATURE SELECTION

Feature selection was used to find which subset of features were best suited to predicting the target values (dominance, valence and arousal labels). To do this, both correlation and mutual information approaches were used. Correlation is a measure of how two variables change together. In this case, the two variables were sentiment labels and the relevant features. Feature selection was also explored using a mutual information approach. Mutual information is calculated between two variables and measures the reduction in uncertainty for one variable given a known value of the other variable. In other words, it is the amount of information one variable gives about the other (Vergara & Estévez, 2013). The reason to perform feature selection is to limit the input that the model receives to just the relevant features. Feature selection was performed not only to find which features were most predictive of the three sentiment labels, but also the optimum number of features that should be given as input.

HYPERPARAMETER OPTIMIZATION AND GRID SEARCH

Hyperparameter optimization is the process of finding the optimal combination of hyperparameters for a model. In a neural network, hyperparameters are used to determine how the model is training (e.g. number of layers, number of neurons and learning rate). In this project, hyperparameter optimization was performed using grid search. Grid search is essentially going through all the possible combinations of hyperparameters to find the best combination (Feurer & Hutte, 2019). The keras tuner library (O'Malley et al., 2019) was used to conduct a grid search for the two neural networks

Grid search was also performed for the two traditional models, but instead of testing different hyperparameter combinations, the model types and feature combinations were evaluated. For grid search on these two traditional models, the scikit-learn library was used (Pedregosa et al., 2011). The eight different types of traditional models (described above in the Machine Learning Methods section in the Introduction on page 7) were evaluated during the grid search. These include three classic regression models (linear, ridge and lasso regression), two ensembles of classic regression models (gradient boosting regressor and ada boost regressor), one simple neural network (multi-layer perceptron), and two other models (support vector regressor and random forests). For each of the eight traditional model types, different combinations of hyperparameters were tested to see which combination had the lowest error.

After using grid search to find the best hyperparameters combinations, the models were defined with said hyperparameters. Then the models were trained on the train data and tested on the validation set with different parameters. The goal when training a model is to maximise the accuracy on the validation set, so lots of combinations of these parameters are tested. During the training of the neural networks specifically, different versions of the model were saved at 'checkpoints'. Once the neural network has finished training one can determine which set of hyperparameters was best, and which checkpoint of training to use by evaluating the model on the validate set, and finally testing that model on the test dataset.

MODELS

Throughout this project, several different types of machine learning models were investigated to see what would be best suited to this task of predicting sentiment values. In this project, both traditional models and neural network models were explored, in addition to both feature-based and non-feature-based approaches. Two models were made for predicting sentiment using acoustic information (due to the aforementioned limitations of the acoustic data), and four models were made using language features. In addition, two types of ensemble models were created. Therefore these eight models are presented below.

(1) Traditional feature-based model using acoustic features:

Acoustic features were used alongside a grid search methodology to train and optimise traditional machine learning models for predicting the sentiment labels.

(2) Feature-based neural network using acoustic features:

The second type of model that was explored was a neural network that predicts sentiment based on the acoustic features. The same features that were extracted for model (1) were used in this model. Similar to model (1), a grid search was used to determine which neural network hyperparameters would be best at predicting the target values.

(3) Traditional feature-based model using language features:

As in the first approach, feature selection was performed on the language feature set, and a grid search methodology was used to train and optimise traditional machine learning models.

(4) Feature-based neural network on language:

A language-based neural network model was also developed. The same features that were extracted for model (3) were used in this model. As in the previous models, a grid search approach was used to find the best combination of hyperparameters for the model structure.

The next two models are unique to the language modality as the raw lyrics can be harnessed directly (whereas with the acoustic modality, only pre-computed features could be used for modelling).

(5) Traditional model based on BERT embeddings:

In addition, the BERT (Bidirectional Encoder Representations from Transformers) vector embedding of each song was computed. The BERT model is a popular open-source NLP model that was developed by researchers at Google (Devlin et al., 2018). The BERT model can take in 512 tokens (words from the songs) and outputs a vector that contains a long string of numbers. Each token has a unique vector, and by computing the cosine distance between two vectors one gets a measure of coherence. In this study, the BERT vector was calculated for each song in the dataset. Grid search was performed to see which type of traditional machine learning model was best suited to predicting sentiment values using the BERT vectors. As with the models above, after finding the type of model with the highest accuracy, it was tested on the test dataset.

(6) RoBERTa fine-tuned language model:

A large language model (RoBERTa) was fine-tuned to predict the arousal, valence and dominance values in the dataset. Fine-tuning is a technique used to adapt pre-trained neural network models for a specific task or dataset. The RoBERTa model is trained on general text, and it does not have experience in sentiment prediction. Therefore, the model was fine-tuned so that it was better suited to predict arousal, valence and dominance of songs.

(7) Ensembled multimodal model:

One approach to ensemble modelling is to combine the outputs of several base models to create one final prediction. Each unimodal machine learning model mentioned above was ensembled together to make one final model. The six outputs of the models were combined by calculating the mean. Averaging is a common way of combining models as all the six base models are taken into account in the final prediction.

(8) A single multimodal model:

The final type of model explored in this project was one multimodal model that combines information from different modalities into a single model. In contrast to the previous model (7) where all the **outputs** from the different models were combined in the prediction, in this model all the **features** (from both the language and acoustic modalities) were combined into one feature set. Feature selection was performed to see which features would be most useful to include in the model's input. As in the above models, both feature selection and grid search were used to create one final multimodal model. This model type works well if the modalities provide complementary information about each other. This could be because one language feature and one audio feature have a strong relationship that is beneficial to the prediction.

After developing and testing the above models on the train and validate datasets, they were tested individually on the held-out test set.

EVALUATION

While developing and testing these machine learning models, mean absolute error (MAE) was used to evaluate how the model was performing. As a part of the training, the model will make predictions of the sentiment. The MAE is a measure of how far off the prediction is to the true value (which is labelled by humans). In this project, the goal was to predict sentiment values that range from 1 to 9. A MAE of 1.5 would mean that on average the model's predictions are 1.5 points away from the real score. This measure is useful as it provides insight into how the model is performing. Other key measures include the smallest/largest prediction and the standard deviation.

TOOLS

The project code was written and run in Google Colaboratory (Bisong, 2019) and Jupyter Notebook (Kluyver et al., 2016) and is **available at my GitHub repository (professorlara):** https://github.com/professorlara/Predicting-emotions-in-music-using-artificial-intelligence These are web-based interactive platforms for writing, running and sharing code. To develop the traditional ML models, the scikit-learn library was used (Pedregosa et al., 2011). Scikit-learn is a widely used library for creating machine learning models in Python. The Keras library was used to develop the neural networks (Chollet et al., 2015). Keras is a free but highly powerful API, used for developing deep learning models (such as neural networks) in Python. For importing and manipulating the datasets, the Pandas library was implemented (McKinney, 2010).

RESULTS

FEATURE SELECTION

Results summary (Table 2, Panels A and B): In summary, in Table 2 [A] on acoustics it is clear that the low-level spectral contrast coefficient is the feature that correlated the highest to all three sentiment labels. In Table 2 [B] the barkband feature was most predictive of dominance, whereas the melband feature was most predictive of valence and arousal. In Table 2 [A] and [B], arousal was the label that had the strongest relationship to the acoustic features, as the correlation and mutual information scores were considerably higher than valence and dominance.

Table 2: Top 5 ***acoustic*** *features for dominance, valence and arousal ratings using correlation [A] and mutual information [B]. The scores range from 0-1. (For further descriptions of each of the features see Bogdanov et al., 2013).*

	Dominance		**Valence**		**Arousal**	
	Feature	Score	Feature	Score	Feature	Score
1	acousticbrainz_lowlevel_spectral_contrast_coeffs_dvar2	0.189	acousticbrainz_lowlevel_spectral_contrast_coeffs_dvar2	0.211	acousticbrainz_lowlevel_spectral_complexity_dmean	0.419
2	acousticbrainz_lowlevel_spectral_contrast_coeffs_dvar3	0.183	acousticbrainz_lowlevel_spectral_contrast_coeffs_dmean2	0.189	acousticbrainz_lowlevel_spectral_complexity_dmean2	0.416
3	acousticbrainz_onset_rate	0.175	acousticbrainz_lowlevel_spectral_contrast_coeffs_dvar3	0.178	acousticbrainz_lowlevel_melbands_spread_dmean	0.408
4	acousticbrainz_lowlevel_spectral_contrast_coeffs_dmean3	0.171	acousticbrainz_lowlevel_spectral_contrast_coeffs_dmean3	0.174	acousticbrainz_lowlevel_melbands_spread_dmean2	0.407
5	acousticbrainz_lowlevel_spectral_contrast_valleys_dmean2	0.158	acousticbrainz_onset_rate	0.174	acousticbrainz_lowlevel_barkbands_spread_dmean2	0.398

[A]

	Dominance		Valence		Arousal	
	Feature	Score	Feature	Score	Feature	Score
1	acousticbrainz_lowlevel_barkbands_skewness_var	0.046	acousticbrainz_lowlevel_melbands_median29	0.062	acousticbrainz_lowlevel_melbands_spread_dmean	0.139
2	acousticbrainz_lowlevel_spectral_centroid_dmean	0.042	acousticbrainz_lowlevel_barkbands_spread_dmean	0.058	acousticbrainz_lowlevel_melbands_spread_dmean2	0.131
3	acousticbrainz_lowlevel_spectral_energyband_high_dvar2	0.041	acousticbrainz_lowlevel_melbands_median21	0.055	acousticbrainz_lowlevel_spectral_complexity_dmean	0.126
4	acousticbrainz_lowlevel_spectral_contrast_valleys_dvar3	0.041	acousticbrainz_lowlevel_barkbands_flatness_db_mean	0.054	acousticbrainz_lowlevel_spectral_centroid_dmean	0.124
5	acousticbrainz_lowlevel_spectral_centroid_dvar2	0.041	acousticbrainz_lowlevel_spectral_contrast_coeffs_dmean2	0.053	acousticbrainz_lowlevel_barkbands_spread_median	0.124

[B]

Results summary (Table 3, Panels A and B): In Panel A, the average sentiment of the song correlated highly with dominance and valence. For arousal, content density correlated highest. Panel B shows that verb count scored highly for dominance, the number of 5-grams was best for valence, and adjective count in the song had the highest mutual information score for arousal. A high mutual information score reflects the fact that the feature and the label share a lot of information, so the feature can be used to predict the label. As in Table 2, arousal had the strongest relationship to the language features. (For further descriptions of the language features, see the Features section above on pages 11-12).

Table 3: Top 5 ***language*** *features for dominance, valence and arousal using correlation [A] and mutual information [B].*

	Dominance		Valence		Arousal	
	Feature	Score	Feature	Score	Feature	Score
1	Average sentiment from Vader model, computed individually by song line	0.178	Average sentiment from Distilbert model, computed individually by song line	0.212	Content density	0.235
2	Average sentiment from Distilbert model, computed individually by song line	0.154	Ratio of positive sentiment from Distilbert model, computed individually by song line	0.210	Total verb count	0.231
3	Ratio of positive sentiment from Distilbert model, computed individually by song line	0.154	Average sentiment from Vader model, computed individually by song line	0.204	Number of lines	0.230
4	Ratio of positive sentiment from Roberta model, computed individually by song line	0.149	Ratio of positive sentiment from Roberta model, computed individually by song line	0.202	Number of 5-grams	0.226
5	Average sentiment from Roberta model, computed individually by song line	0.148	Average sentiment from Roberta model, computed individually by song line	0.202	Number of unique 5-grams	0.227

[A]

	Dominance		Valence		Arousal	
	Feature	Score	Feature	Score	Feature	Score
1	Total verb count	0.037	Number of 5-grams	0.039	Adjective count	0.051
2	Median sentiment from Roberta model, computed individually by song line	0.032	Number of unique 4-grams	0.0385	Number of words	0.049
3	Average sentiment from Roberta model, computed individually by song line	0.031	Number of trigrams	0.0384	Number of unique 5-grams	0.0483
4	Number of lines	0.027	Number of words	0.0378	Number of 4-grams	0.0482
5	Base verb frequency	0.025	Number of 4-grams	0.0377	Number of unique bigrams	0.047

[B]

GRID SEARCH:

Results summary (Table 4, Panels A to F): The results in Panel A show that a Random Forest Regressor model that is composed of 68 estimators (decision trees) is the best type of traditional model for predicting dominance using acoustic data. Panel B suggests a Gradient Boosting Regressor model type which, like the model in Table 4 [Panel A], is an ensemble model. Panels C and D show the results from the grid search to find the best architecture for the neural networks. For both model types, the use of dropout during training was recommended. Panel E suggests that a model using Lasso regression is best for predicting the sentiment label using BERT embeddings. Finally, for the multimodal model [Panel F], Support Vector Regression yielded the lowest error.

*Table 4: Grid search results illustrating which model type and combination of hyperparameters were best at predicting **dominance** (Panels [A] to [F] for each of the model types excluding the fine-tuned RoBERTa model and the ensembled model).*

Estimator	Mean absolute error	Hyperparameters				
Random Forest Regressor	0.7557	minimum samples per leaf	number of estimators			
		10	68			
Gradient Boosting Regressor	0.7566	minimum samples per leaf	number of estimators	criterion	learning rate	loss
		8	200	mse	0.1	ls
Random Forest Regressor	0.7575	minimum samples per leaf	number of estimators			
		8	55			

[A] Grid search for traditional acoustic model

Estimator	**Mean absolute error**	**Hyperparameters**				
Gradient Boosting Regressor	0.7588	criterion	learning rate	loss	minimum samples per leaf	number of estimators
		squared_error	0.1	absolute_error	8	100
Gradient Boosting Regressor	0.7592	criterion	learning rate	loss	minimum samples per leaf	number of estimators
		squared_error	0.1	absolute_error	10	100
Gradient Boosting Regressor	0.7609	criterion	learning rate	loss	minimum samples per leaf	number of estimators
		squared_error	0.1	absolute_error	8	200

[B] Grid search for traditional language model

Mean Absolute Error	**Hyperparameters**								
0.76800	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1			
	tanh	True	0.0055	2	224	32			
0.76802	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	nodes in layer 4
	tanh	True	0.0015	5	256	448	384	416	32
0.76803	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	nodes in layer 4
	tanh	True	0.0012	5	416	384	64	32	128

[C] Grid search for acoustic-based neural network

Mean Absolute Error	**Hyperparameters**								
0.73785	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	nodes in layer 4
	tanh	True	0.00024	5	128	320	320	288	128
0.75586	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1			
	tanh	True	0.00014	2	32	512			
0.76619	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	
	tanh	True	0.00082	4	512	288	480	320	

[D] Grid search for language-based neural network

Estimator	Mean absolute error	Hyperparameters
Lasso Regression	0.772	alpha
		0.001
Lasso Regression	0.784	alpha
		0.01
Lasso Regression	0.794	alpha
		0.0001

[E] Grid search for traditional model based on BERT embeddings

Estimator	Mean absolute error	Hyperparameters			
Support Vector Regression	0.8049	Strength of regularisation parameter	Degree of polynomial kernel function	epsilon	kernel
		0.01	1	0.2	linear
Support Vector Regression	0.8058	Strength of regularisation parameter	Degree of polynomial kernel function	epsilon	kernel
		0.05	1	0.2	linear
Support Vector Regression	0.8066	Strength of regularisation parameter	Degree of polynomial kernel function	epsilon	kernel
		0.01	1	0.5	linear

[F] One multimodal model

Results summary (Table 5, Panels A to F): The results from Table 5 suggest that Ridge Regression and Support Vector Regression model types are well-suited traditional models to predicting the valence label. The results also show that the acoustic-based neural network should use dropout as part of the training, whereas the language-based neural network should not.

Table 5: Grid search results illustrating which model type and combination of hyperparameters were best at predicting ***valence*** *(Panels [A] to [E] for each of the model types excluding the fine-tuned RoBERTa model and the ensembled model).*

Estimator	Mean absolute error	Hyperparameters		
Ridge Regression	1.12560	alpha	fit intercept	solver
		0.0001	True	svd
Ridge Regression	1.12567	alpha	fit intercept	solver
		0.001	True	cholesky
Ridge Regression	1.12574	alpha	fit intercept	solver
		0.00001	True	cholesky

[A] Grid search for traditional acoustic model

Estimator	Mean absolute error	Hyperparameters			
Support Vector Regression	1.1528	Strength of regularisation parameter	Degree of polynomial kernel function	epsilon	kernel
		0.05	1	0.5	linear
Support Vector Regression	1.1530	Strength of regularisation parameter	Degree of polynomial kernel function	epsilon	kernel
		0.01	1	0.5	linear
Support Vector Regression	1.1539	Strength of regularisation parameter	Degree of polynomial kernel function	epsilon	kernel
		0.01	1	0.2	linear

[B] Grid search for traditional language model

Mean Absolute Error	Hyperparameters								
1.1414	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	nodes in layer 4
	tanh	True	0.0004	3	384	128	160	160	384
1.1507	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	nodes in layer 4
	tanh	False	0.0003	5	192	448	128	384	32
1.1532	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	
	tanh	False	0.0027	1	320	128	512	448	

[C] Grid search for neural network based on acoustics

Mean Absolute Error	Hyperparameters								
1.0915	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	
	tanh	False	0.00059	4	128	256	224	64	
1.0918	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2		
	tanh	False	0.00099	3	96	288	384		
1.0919	activation	dropout	learning rate	layers	nnodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	nodes in layer 4
	relu	True	0.00016	5	160	32	480	448	224

[D] Grid search for neural network based on language

Estimator	Mean absolute error	Hyperparameters		
Lasso Regression	1.1393	alpha		
		0.0001		
GradientBoosting Regressor	1.1469	learning rate	minimum samples per leaf	number of estimators
		0.1468	6	71
GradientBoosting Regressor	1.1471	learning rate	minimum samples per leaf	number of estimators
		0.1468	6	65

[E] Grid search for a traditional model based on BERT embeddings

Estimator	Mean absolute error	Hyperparameters				
Gradient Boosting Regressor	1.0760	criterion	learning rate	loss	minimum samples per leaf	number of estimators
		squared_error	0.1	absolute_error	8	200
Gradient Boosting Regressor	1.0767	criterion	learning rate	loss	minimum samples per leaf	number of estimators
		squared_error	0.1	absolute_error	10	100
Gradient Boosting Regressor	0.0781	criterion	learning rate	loss	minimum samples per leaf	number of estimators
		squared_error	0.1	absolute_error	8	200

[F] One multimodal model

Results summary (Table 6, Panels A to F): The results in Panels A and B suggest that Ridge Regression and Support Vector Regression model types are well-suited traditional models to predicting the valence label. Similar to what was reported in Table 5 above, the results in Table 6 Panels C and D also show that dropout should be used when training the acoustic-based neural network, whereas in the language-based neural network, it should not be used. Panel E in Table 6 suggests that a Lasso Regression model for predicting valence using the BERT embeddings is preferable. Lastly, similar to the findings reported in Table 5 Panel F above, the results in Table 6 Panel F support the idea that a Gradient Boosting Regressor would be the best traditional model type for the multimodal model.

*Table 6: Grid search results illustrating which model type and combination of hyperparameters were best at predicting **arousal** (Panels [A] to [F] for each of the model types excluding the fine-tuned RoBERTa model and the ensembled model).*

Estimator	**Mean absolute error**	**Hyperparameters**			
Support Vector Regression	0.7556	Strength of regularisation parameter	Degree of polynomial kernel function	epsilon	kernel
		0.01	1	0.2	linear
Support Vector Regression	0.7567	Strength of regularisation parameter	Degree of polynomial kernel function	epsilon	kernel
		0.05	1	0.2	linear
Support Vector Regression	0.7590	Strength of regularisation parameter	Degree of polynomial kernel function	epsilon	kernel
		0.05	1	0.5	linear

[A] Grid search for traditional acoustic model

Estimator	**Mean absolute error**	**Hyperparameters**				
Gradient Boosting Regressor	0.8101	criterion	learning rate	loss	minimum samples per leaf	number of estimators
		squared_error	0.1	absolute_error	10	200
Gradient Boosting Regressor	0.8113	criterion	learning rate	loss	minimum samples per leaf	number of estimators
		squared_error	0.1	absolute_error	8	200
Gradient Boosting Regressor	0.8114	criterion	learning rate	loss	minimum samples per leaf	number of estimators
		squared_error	0.1	absolute_error	10	100

[B] Grid search for traditional language model

Mean Absolute Error	**Hyperparameters**									
0.7038	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	nodes in layer 4	nodes in layer 5
	relu	True	0.0004	6	448	160	64	32	288	512
0.7178	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	nodes in layer 4	nodes in layer 5
	tanh	False	0.0002	6	480	320	512	128	224	480
0.7182	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	nodes in layer 4	nodes in layer 5
	tanh	True	0.0001	6	128	320	512	512	96	32

[C] Grid search for neural network based on acoustics

Mean Absolute Error	**Hyperparameters**									
0.7998	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	nodes in layer 4	nodes in layer 5
	tanh	False	0.0003	6	256	224	416	352	512	160
0.8068	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	nodes in layer 4	nodes in layer 5
	tanh	True	0.0003	6	64	192	128	128	32	32
0.7182	activation	dropout	learning rate	layers	nodes in layer 0	nodes in layer 1	nodes in layer 2	nodes in layer 3	nodes in layer 4	nodes in layer 5
	tanh	False	0.0003	6	512	96	224	416	416	192

[D] Grid search for neural network based on language

Estimator	**Mean absolute error**	**Hyperparameters**
Lasso Regression	0.7834	alpha
		0.001
Lasso Regression	0.8030	alpha
		0.0001
Lasso Regression	0.8049	alpha
		0.01

[E] Grid search for a traditional model based on BERT embeddings

Estimator	**Mean absolute error**	**Hyperparameters**				
Gradient Boosting Regressor	0.7075	criterion	learning rate	loss	minimum samples per leaf	number of estimators
		squared_error	0.1	absolute_error	10	200
Gradient Boosting Regressor	0.7079	criterion	learning rate	loss	minimum samples per leaf	number of estimators
		squared_error	0.1	absolute_error	8	200
Gradient Boosting Regressor	0.7084	criterion	learning rate	loss	minimum samples per leaf	number of estimators
		squared_error	0.1	absolute_error	8	100

[F] One multimodal model

RESULTS SUMMARY (TABLES 7, 8 AND 9): As can be seen from Tables 7 to 9 below, the arousal label was the easiest to predict as it had overall the lowest error rate ranging between 0.73 to 0.81. Surprisingly, the valence label was the hardest to predict, with error rates ranging between 1.1 and 1.9. This might be because, out of the three datasets, the valence label had the largest range in the original dataset, which meant that there was more margin for error. The MAE of the models predicting dominance tended to be between 0.77 and 0.84, slightly worse than the arousal models.

*Table 7: Comparison of models: Mean absolute error of all 8 model types when predicting **dominance** on the test dataset*

Model type	Mean absolute error on test dataset
Traditional acoustic model	0.7771
Neural network based on acoustics	0.8273
Traditional language model	0.8214
Neural network based on language	0.8421
Language model based on BERT embeddings	0.8245
Fine-tuned roBERTa model	0.8118
Ensembled multimodal model	0.8174
A single multimodal model	0.7752

*Table 8: Comparison of models: Mean absolute error of all 8 model types when predicting **valence** on the test dataset*

Model type	Mean absolute error on test dataset
Traditional acoustic model	1.1472
Neural network based on acoustics	1.1023
Traditional language model	1.1214
Neural network based on language	1.1959
Language model based on BERT embeddings	1.1188
Fine-tuned roBERTa model	1.1682
Ensembled multimodal model	1.1423
A single multimodal model	1.1406

Table 9: Comparison of models: Mean absolute error of all 8 model types when predicting ***arousal*** *on the test dataset*

Model type	**Mean absolute error on test dataset**
Traditional acoustic model	0.7351
Neural network based on acoustics	0.7331
Traditional language model	0.7892
Neural network based on language	0.7888
Language model based on BERT embeddings	0.7685
Fine-tuned roBERTa model	0.8182
Ensembled multimodal model	0.6911
A single multimodal model	0.7341

DISCUSSION

The music datasets used in this project had arousal, valence and dominance sentiment labels associated with the songs. The goal of this study was to see if a multimodal approach to predicting these sentiment labels would result in more accurate predictions compared to the prediction from unimodal modelling approaches. A challenge faced was that there was a very small range of values for the three sentiment labels, resulting in that the final models made poorer predictions than if they had been trained on a full range of sentiment labels.

The **first hypothesis** in this project was that if the six unimodal models' predictions were ensembled, the final prediction would be more accurate than the individual base models alone. This hypothesis was correct for the ensemble model predicting arousal, but not for the dominance and valence models. The **second hypothesis** was that the three multimodal models' final predictions would have a higher accuracy than any of the six unimodal models. This prediction was correct only for the multimodal model predicting dominance. However, the arousal model had the second-best error rate (after the acoustic-based neural network). The **third hypothesis** was that both the ensemble and multimodal models would outperform the pre-trained and fine-tuned RoBERTa models' predictions. This hypothesis was true for the models predicting valence and arousal, as the mean absolute error for these two models was lower than the fine-tuned models'. However, for the dominance model, the fine-tuned RoBERTa model had a slightly lower error rate than the ensemble model, but had a higher error rate than the multimodal model.

In general, the arousal label was easier to predict: the error rate was significantly lower in models that were predicting arousal compared to the other two labels. Also, when performing feature selection, the arousal label correlated highest to both the language and acoustic features. In contrast, both the unimodal and multimodal models that predicted valence had significantly higher error rates compared to the arousal and dominance labels.

This result is interesting as it suggests that certain aspects of sentiment are better suited to sentiment prediction than others.

One limitation of this study was that it was not possible to access the songs' raw audio files for the acoustic models. This is in contrast to the language part of the project, where since there was access to the full lyrics, features could be extracted that were thought to be interesting and beneficial to this prediction. Instead, the acoustic dataset contained pre-computed audio features from acousticbrainz such as MFCCs, tempo and key signature. Therefore, feature-based models had to be created using these pre-computed features. If the raw audio files had been accessible, there would have been more flexibility in modelling. Nonetheless, the acoustic feature-based models generally outperformed the language-based models (where there was the option of using other modelling techniques).

CONCLUSION

This study analysed close to ten thousand songs using various machine learning methods to predict three dimensions of sentiment, namely dominance, arousal, and valence. In general, the arousal label was easiest to predict as evident by the models predicting arousal having lower error rates compared to the other two labels. As expected, in some cases, a multimodal approach of combining data modalities both enriched and increased the accuracy of predicting the sentiment of the songs. The methods that were developed in this project have several implications in the real world. For example, these sentiment analysis techniques could be used by Spotify to improve their music recommendation system such that they make their recommendations even more personal and tailor-made to the user. In addition, these methods could be used in areas beyond just music and songs. For example, voice recognition software, such as Siri and Alexa, might benefit from these concepts as they could enable their technology to properly understand the emotion a user is trying to convey, and thereby adapt their responses accordingly. Furthermore, the approach used in this study could help improve voice recognition software in particular by detecting and interpreting the use of sarcasm and irony in their users, which has long been a limiting factor and models have traditionally performed quite poorly on these dimensions. In conclusion, this project showcases the added value of adopting a multimodal approach to sentiment recognition.

REFERENCES

Abburi, H., Akkireddy, E.S A., Gangashetti, S., & Mamidi, R. (2016). Multimodal Sentiment Analysis of Telugu Songs. *Proceedings of the 4th Workshop on Sentiment Analysis where AI meets Psychology (SAAIP 2016)* (pp. 48-52).

Akiki, C., & Burghardt, M. (2021). MuSe: The Musical Sentiment Dataset. *Journal of Open Humanities Data*, *7*. https://doi.org/10.5334/johd.33

Bisong, E. (2019). Google Colaboratory. In: *Building Machine Learning and Deep Learning Models on Google Cloud Platform.* Apress: Berkeley, CA. https://doi.org/10.1007/978-1-4842-4470-8_7

Björklund, G., Bohlin, M., Olander, E., Jansson, J., Walter, C.E., Au-Yong-Oliveira, M. (2022). An Exploratory Study on the Spotify Recommender System. In *World Conference on Information Systems and Technologies* (pp. 366-378). Cham: Springer International Publishing.

Bogdanov, D., Wack N., Gómez E., Gulati S., Herrera P., Mayor O., ... & Serra, X. (2013). Essentia: An audio analysis library for music information retrieval. *International Society for Music Information Retrieval Conference (ISMIR'13).* 493-498.

Chandler, C., Foltz, P.W., Elvevåg, B. (2020). Using Machine Learning in Psychiatry: The Need to Establish a Framework That Nurtures Trustworthiness. *Schizophr Bull.*, 46(1):11-14. https://doi.org/10.1093/schbul/sbz105

Choi, R.Y., Coyner, A.S., Kalpathy-Cramer, J., Chiang, M.F., Campbell, J.P. (2020). Introduction to Machine Learning, Neural Networks, and Deep Learning. *Transl Vis Sci Technol.*, 9(2):14. https://doi.org/10.1167/tvst.9.2.14

Chollet, F., et al. (2015). Keras. Retrieved from https://keras.io

Devlin, J., Chang, M.W., Lee, K., Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Feurer, M., & Hutter, F. (2019). Hyperparameter optimization. *Automated machine learning: Methods, systems, challenges*, 3-33. https://doi.org/10.1007/978-3-030-05318-5_1

Hutto, C.J. & Gilbert, E.E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Eighth International Conference on Weblogs and Social Media (ICWSM-14). Ann Arbor, MI, June 2014. https://doi.org/10.1609/icwsm.v8i1.14550

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B. E., Bussonnier, M., Frederic, J., ... & Willing, C. (2016). Jupyter Notebooks-a publishing format for reproducible computational workflows. *Elpub*, *2016*, 87-90.

Kotu, V., Deshpande, B. (2015). Data Mining Process. *Predictive Analytics and Data Mining*, 17–36. https://doi.org/10.1016/b978-0-12-801460-8.00002-1

Kumar, A. (2023). Sklearn neural networks example [Online image]. *Vitalflux.* https://vitalflux.com/sklearn-neural-network-regression-example-mlpregressor/

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*. https://doi.org/10.48550/arXiv.1907.11692

McKinney, W., et al. (2010). Data structures for statistical computing in Python. In *Proceedings of the 9th Python in Science Conference* (Vol. 445, pp. 51–56).

Morency, L. P., Mihalcea, R., & Doshi, P. (2011, November). Towards multimodal sentiment analysis: Harvesting opinions from the web. In *Proceedings of the 13th International Conference on Multimodal Interfaces* (pp. 169-176).

Ng, A. (2023). CS229 Lecture notes. Freely available from Stanford University.

O'Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H., Invernizzi, L., & Others. (2019). Keras Tuner. Retrieved from https://github.com/keras-team/keras-tuner

Opitz, D., Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, *11*, 169-198.

Oramas, S., Barbieri, F., Nieto Caballero, O., & Serra, X. (2018). Multimodal deep learning for music genre classification. *Transactions of the International Society for Music Information Retrieval. 2018; 1 (1):* 4-21.

Panigrahi, P. (2023). Decision tree algorithm [Online image]. *Pianalytix.* https://pianalytix.com/decision-tree-algorithm/

Parthasarathy, S., & Busso, C. (2017, August). Jointly Predicting Arousal, Valence and Dominance with Multi-Task Learning. In *Interspeech* (Vol. 2017, pp. 1103-1107).

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Porter, A., Bogdanov, D., Kaye, R., Tsukanov, R., & Serra, X. (2015). Acousticbrainz: a community platform for gathering music information obtained from audio. *16th International Society for Music Information Retrieval Conference.*

Rosidi, N. (2023). Multimodal Models Explained. *KD nuggets. https://www.kdnuggets.com/2023/03/multimodal-models-explained.html*

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108. https://doi.org/10.48550/arXiv.1910.01108*

Sato, N., & Obuchi, Y. (2007). Emotion recognition using Mel-frequency cepstral coefficients. *Journal of Natural Language Processing*, *14*(4), 83–96. https://doi.org/10.5715/jnlp.14.4_83

Shah, D. (2021). Song Lyrics Dataset. Retrieved August 20, 2023 from https://www.kaggle.com/datasets/deepshah16/song-lyrics-dataset

Soleymani, M., Garcia, D., Jou, B., Schuller, B., Chang, S. F., & Pantic, M. (2017). A survey of multimodal sentiment analysis. *Image and Vision Computing*, *65*, 3-14. https://doi.org/10.1016/j.imavis.2017.08.003

Torruella, J., & Capsada, R. (2013). Lexical statistics and tipological structures: a measure of lexical richness. *Procedia-Social and Behavioral Sciences*, *95*, 447-454. https://doi.org/10.1016/j.sbspro.2013.10.668

Vasiliev, Y. (2020). *Natural language processing with Python and spaCy: A practical introduction*. No Starch Press.

Vergara, J.R., Estévez, P.A. (2013). A review of feature selection methods based on mutual information. *Neural Computing and Applications*, *24*(1), 175–186. https://doi.org/10.1007/s00521-013-1368-0

Warriner, A.B., Kuperman, V., Brysbaert, M. (2013). Norms of valence, arousal, and dominance for 13,915 English lemmas. *Behav Res Methods,* 45(4):1191-207. https://doi.org/10.3758/s13428-012-0314-x

Yang, J., Luo, F.L., Nehorai, A. (2003). Spectral contrast enhancement: Algorithms and comparisons. *Speech Communication*, *39*(1-2), 33-46.