COMP9444 Project Summary

Lyrics Mood Analysis

Term3 2022

Calvin Long(z5255352)
Alon Moss(z5160732)
Rongbo Zhao(z5286178)
Zhengye Ma(z5158505)
Hong Zhang(z5257097)

Introduction/Motivation

Currently there is no technology to be able to create a playlist of music based on a specific mood. If we can have song classification which is based on the mood, it can be of great convenience to those who want to listen to different mood music. Through analysing multiple Neural Network models our team has identified the opportunity to do so and enhance enjoyment of music.

In this report, we will discuss and compare several deep learning models such as the TextCNN model, Bert-based model(GoEmotion) and BiLSTM model. Then, we will choose the model with high performance to analyse the lyrics mood. Based on the model prediction of music lyrics, classify the song.

Literature Review

GoEmotions is a Fine-grained statement classification data set. In the past 10 years, the NLP group had already provided some data sets based on the language of emotion classification. Most of the data set is manual. In an Essay called "GoEmotions: A Dataset of Fine-Grained Emotions", these researchers use PCCA Algorithm to better understand agreement among raters and the latent structure of the emotion space[1]. This picture in that essay shows the clearly processes of the PCCA algorithm:
[2]

Algorithm 1 Leave-One-Rater-Out PPCA

$R \leftarrow$ set of raters
$E \leftarrow$ set of emotions
$C \in \mathbb{R}^{|R| \times |E|}$
for all raters $r \in \{1, ..., |R|\}$ **do**
  $n \leftarrow$ number of examples annotated by r
  $J \in \mathbb{R}^{n \times |R| \times |E|} \leftarrow$ all ratings for the examples annotated by r
  $J^{-r} \in \mathbb{R}^{n \times |R|-1 \times |E|} \leftarrow$ all ratings in J, excluding r
  $J^{r} \in \mathbb{R}^{n \times |E|} \leftarrow$ all ratings by r
  $X, Y \in \mathbb{R}^{n \times |E|} \leftarrow$ randomly split J^{-r} and average ratings across raters for both sets
  $W \in \mathbb{R}^{|E| \times |E|} \leftarrow$ result of $PPCA(X, Y)$
  for all components† $\mathbf{w}_{i \in \{1,...,|E|\}}$ in W **do**
    $\mathbf{v}_i^r \leftarrow$ projection‡ of J^r onto $\mathbf{w}_i$
    $\mathbf{v}_i^{-r} \leftarrow$ projection‡ of J^{-r} onto $\mathbf{w}_i$
    $C_{r,i} \leftarrow$ correlation between $\mathbf{v}_i^r$ and $\mathbf{v}_i^{-r}$, partialing out $\mathbf{v}_k^{-r} \forall k \in \{1, ..., i-1\}$
  end for
end for
$C' \leftarrow$ Wilcoxon signed rank test on C
$C'' \leftarrow$ Bonferroni correction on $C'(\alpha = 0.05)$
†in descending order of eigenvalue
‡we demean vectors before projection

According to the picture, we can see that the PCCA algorithm is using a linear model function to get the result of Maximum likelihood estimation. Basically, the correctness of the algorithm is around 60%. In our group project, we will use Hugging Face Transformers to increase the percentage of correctness.

Models and/or Methods

There are three models used in this project. Firstly the TextCNN model which uses convolutional neural network to classify the text, sentences will be represented by matrices, after that the matrices

[1] https://arxiv.org/pdf/2005.00547.pdf page 4
[2] https://arxiv.org/pdf/2005.00547.pdf page 5

will be calculated by the model which is formed from several convolutional layers, max pooling layers and dense layers, the output is the probability distribution of the categories. Secondly, the Bert-based model, it can be described as a language representation model, its main model structure is a stack of transformer encoder, which is essentially a two-stage framework, pretraining, and fine tuning on specific tasks. Thirdly, the BiLSTM model which is a type of Recurrent Neural Network, is a combination of forward LSTM and backward LSTM, LSTM has three stages which are selective forget stage, selective memory stage and output stage. In short, it can remember the long-term memory and forget the unimportant information.

Experimental Setup

In order to train and test the multiple models, we selected two input data files. The first was sourced from Kaggle, under “The Emotions Dataset for NLP”, which contains 20,000 inputs (2,000 for testing, 16,000 for training and 2,000 for validation). The second input file that was used was the GoEmotions Dataset which was developed by the Google Research team and includes just under 50,000 inputs (5,000 for testing and 43,000 for training). At the end, after the model is completed, we also need a song lyrics source with text format, hence, we utilised MusixMatch which contains a database of over 8 million song lyrics. For evaluation, we calculated average accuracy, loss, macro F1-Score, Micro F1-Score, and Weighted F1-score for each model. The reason we choose F1 score is that the project is a multiple classification and F1 scores can balance the influence of precision and recall rate to evaluate the model comprehensively.

Results

Quantitative Analysis:

When evaluating NN models qualitatively we have to look at a couple of factors and values which can determine a models effectiveness and accuracy for a particular data set. Looking into accuracy we see that we generally can use 2 values to determine this; Loss and Average. Looking at a model's average will tell us how much of the evaluation data a model has gotten correct or within an acceptable range, and loss will tell us how far away from the correct values our NN model has guessed results to. To evaluate a model's effectiveness we use F1-scores. An F1-score is the harmonic mean (weighted average) of the models predict and recall components which it uses to determine how well the model works for a particular data set. With F1-scores the higher the F1-score the more effective the model is at guessing the correct values and thus the more effective it is.

After training and testing, the accuracy, loss and F1-scores were determined through an evaluation data set, respective to each of the models:

CNN Model		Go-Emotion		LSTM-Keras	
Average:	0.6547	Average:	0.5678	Average:	0.5262
Loss:	0.8219	Loss:	0.7012	Loss:	0.7259
Macro F1-Score:	0.5262	Macro F1-Score:	0.5849	Macro F1-Score:	0.5262
Micro F1-Score:	0.6547	Micro F1-Score:	0.6713	Micro F1-Score:	0.6547
Weighted F1-Score:	0.6430	Weighted F1-Score:	0.6666	Weighted F1-Score:	0.6430

Looking at average and loss for each of the models, we see although CNN has the highest average it also has the highest loss which would introduce additional errors in our measurements. Comparatively the LSTM-Keras model has the lowest accuracy and a moderate loss which is in between that of Go-Emotion and the CNN model. Due to the LSTM-Keras models performance in this aspect we

chose not to use this model in the qualitative analysis. Looking at the Go-Emotion model, we see that we have a moderately high average in comparison to the CNN and LSTM model, and the lowest loss calculated.

Additionally looking at the F1 scores we see that the Go-Emotion model has the highest macro, micro and weighted F1-scores with the CNN model coming in second and LSTM-Keras last in this aspect.

We can see that the CNN model works better based on the average score with high loss and overall lower F1-score than go-emotion, we choose the go-emotion model for lyric analysis in the end.

Qualitative Analysis:

When evaluating whether the NLP analysis was successful with allocating emotional songs, we have to judge whether the emotion of the lyrics of songs for ourselves as humans. With Music being able to express emotion and the ability conveying understanding of that Emotion being a very human expression, evaluating the emotion within songs and by extension the NLP allocation of emotion to song lyrics should be done by humans. Since “Music produces a kind of pleasure which human nature cannot do without ” (Confucius, The book of Rites) we see that if we only compare the values output form this ML model to that of other models, we may only understand what a machine thinks of a song, not the actual human emotion behind it.

This then was compared to a survey which was completed with the 6 emotions as options in order to evaluate each of the songs categorisations by the ML model. By Comparing the NLP analysis with human categorisation we are able to see the accuracy of the model based on human expression and emotion.
For the sample size of songs we chose to look at these 25, which have been categorised by the NLP go-emotion model into these 6 categories below:

1. Anger
    a. 1-800-273-8255 By Logic
    b. 34+35 By Ariana Grande
    c. Come Down By Anderson Paak
    d. Feelings By Lauv
    e. Monster By Kanye West
    f. Rap God By Eminem
    g. Rasputin By Boney M.
    h. The Way I Am By Eminem
2. Fear
    a. 2 Soon By Keshi
    b. Run Boy Run By Woodkid
    c. Without Me By Halsey
3. Sadness
    a. 22 (Taylor’s Version) By Taylor Swift
    b. Bury A Friend By Billie Eilish
    c. Drivers Licence By Olivia Rodrigo
    d. Drunk By Keshi
    e. Glimpse of Us By Joji
    f. Never Gonna Give you Up By Rick Astley
    g. Thank u, next By Ariana Grande
4. Surprise
    a. Diggy Diggy Hole
    b. I Ain’t Worried By OneRepublic
5. Disgust

    a. My Favourite Part By Mac Miller, Ariana Grande
    b. Unholy By Sam Smith
6. Joy
    a. Golden Hour By JVKE
    b. Happy By Pharrell Williams
    c. Hotel California By Eagles

This result was obtained by running the prediction output from go-emotion after training through 20 epoch worth of data. The prediction was then run 1000 times in order to try and accurately represent the songs in a particular emotion category.

The accuracy of the model in comparison to emotion, based on a survey distributed to the public is:

1. Anger: 2/8
2. Fear: 1/2
3. Sadness: 3/6
4. Surprise: 0/2
5. Disgust 1/2
6. Joy: 3/3

Leaving us with a combined accuracy of 40% for evaluating emotions for songs when comparing our NN model with the results from the emotion song survey.

Conclusions

Though this number is a lot lower than we expect, with our expectation being around 60-70% accurate for this lyrics emotion analysis, we see that this lower accuracy could be due to a number of things including but not limited to:

1. The Song tune (no harmonic analysis)
2. The Song speed (no tempo analysis)
3. The Song Artist
4. The Song title

In the future our main focus should be on increasing the accuracy of this model specific to Songs not just song lyrics. We want to include data which will provide our lyrical analysis model additional information and context into the song, with harmonic and tempo analysis introduced into our model. With harmonic and tempo analysis we would be getting a very similar information stream to what a human has when they are listening to music and thus hopefully will allow our model to more accurately predict emotions in songs.

Additionally, the high loss might be jarring to users of our algorithm as they might find some happy songs in a sad playlist, or vice versa. Furthermore, our models might have been inhibited by our usage of Eckman's emotion as realistically having multiple similar classes would minimise the loss totals.

Despite this, our models worked well in terms of speed of running, the fact it was more accurate than a random selection algorithm, as well as the strong relationship between time and accuracy. Overall, we are collectively happy with our work, and look forward to the time saving we are sure to get from our automated playlist curator.