Sentiment Analysis of Songs Lyrics

An NLP approach using deep learning

David Gamez, Joseph Mercer
david.gamez@berkeley.edu, jlmercer@berkeley.edu

UC Berkeley School of Information. Master of Information and Data Science. W266 Natural Language Processing with Deep Learning

Submitted December 7, 2019

Abstract

We consider the task of sentiment analysis of songs lyrics, useful for the new ways in which people are listening to their preferred music. We develop several models to classify each song lyrics in one of the four emotion categories of Russel's Valence-Arousal model. These categories can be grouped in *positive* and *negative* which allows for a simpler binary classification. For binomial classification, a deep learning model based on BERT got an accuracy of 78.25% -accuracy on positive/negative (valence)- beating the 76.29% maximum accuracy on valence reported up to now. However, for multiclass classification, we were not able to beat a classical Naive Bayes model. The greatest confusion of the model was found between *relaxed* and *sad* emotions. A larger corpus would probably help dealing with this difficulty.

Keywords: Lyrics Sentiment Analysis, Lyrics Mood Classification, CNN, RNN, BERT

Introduction

The objective of this work is to develop a deep learning model for sentiment analysis of song lyrics which can be used to provide recommendations of songs based on a user's mood.

The sentiment classification of any document is a challenging task and even more difficult in the case of song lyrics where the text is short, the language is mostly informal, from a small vocabulary and only one of the two dimensions -audio and lyrics- of the song is analyzed.

However, it is a useful task. As the music streaming services are becoming more popular and making recommendations to the users based on the context on and not just on the genre or the artist improves the user experience. Having a way to classify songs would help these services to make better recommendations to the users, or the other way around, streaming services could learn the mood of the listener based on the last set of songs played and then provide music that best fits the user's current mood.

The model will classify each song lyrics in one of the four emotion categories of Russel's Valence-Arousal model [5]: ***angry***, ***happy***, ***sad*** and ***relaxed***. These categories can be grouped in ***positive*** (*happy*, *relaxed*) and ***negative*** (*angry*, *sad*) which allows for a simpler binary classification.

Background

Circumplex Model of Emotions

As described in [1], "one of the most popular dimensional models is the planar model of Russell [5]", shown in figure 1. The model is based on two dimensions: Valence and Arousal.

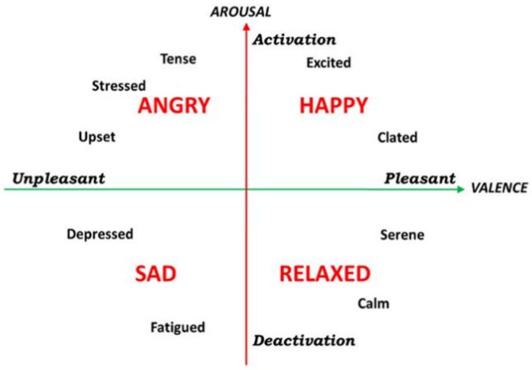

Figure 1: Circumplex model of emotions [1]

In this model, Arousal represents how strongly the emotion is felt, while Valence shows positive or negative intensity of an emotion [1].

Ground Truth Datasets

In order to train our model, a dataset with labeled sentiment information for every song lyrics is needed. This information is hardly available. We use two datasets: first, a corpus of collection of 771 song lyrics collected from Allmusic, annotated using human evaluators and user tags [4] and, second, a collection of 2,500 song lyrics annotated using content words of lyrics and their valence and arousal norms in affect lexicons only (*MoodyLyrics* dataset [1]). These datasets do not contain the lyrics information for copyright reasons, so we web scrapped the data. Intensive analysis was required to clean the data (spelling errors, lack of a common format for comments, chorus, repeated parts of words, as well as different languages -only English ones are used for this experiment), leading to a final corpus of 2574 songs, 1255 positive mood and 1319 negative mood lyrics in English. In table 1, we provide the distribution of songs according to the mood category, valence and source they belong to.

Table 1: Songs per mood category and source

Mood	Valence	Source 1 [4]	Source 2 [1]	Total
Happy	Positive	211	465	667
Angry	Negative	204	468	672
Sad	Negative	204	443	647
Relaxed	Positive	148	440	588

It is important to notice that because both sources have a different generating procedure, we may introduce some noise in the labeled information when using them together. Indeed, the procedure to generate source 2 applied to source 1 for mood classification reported an accuracy of 74.25%, not very good for a dataset to be considered a ground truth[1]. Nevertheless, we use both of them to get a larger dataset, relatively small though.

Previous research

There have been several attempts to automate songs lyrics mood classification, as in [1], [2], [4] and [6]. [1] and [4] are -totally or partially- based on using affect lexicons -like ANEW- were values for *Valence*, *Arousal* and *Dominance* can be found, and [4] uses information from the title of the songs, so they are not comparable to the task in this project. [2] and [6] have used a machine learning or deep learning NLP approach and their result will be compared to ours, though [2] also researched the combined analysis of audio and lyrics.

Methods

Preprocessing

The first part of this experiment was cleaning up the data to be processed by the models. Songs that were not in the English language were removed from the dataset. There were a number of additional characters that needed to be removed from the lyrics using regex methods due to the lyrics being scraped from websites. Representing music via text also introduces some oddities such as repeating syllables or representing other singers. For the purposes of this experiment, those were removed.

Dataset split

The dataset was randomly split into train/validation/test (1850/400/324 samples, 72%/16%/12%), and used for following models.

Target: positive/negative vs 4 quadrants

The fact that the dataset has labels for both valence (2 classes) and mood (4 classes) allowed to investigate binomial and multiclass text classification algorithms. In this sense, every algorithm was trained and tested for positive/negative valence and happy/angry/sad/relaxed classification.

Models selection

Baseline. An obvious baseline model in classification is just using the most frequent group as the only solution, so that any more sophisticated algorithm should be able to beat this very basic one in order to be considered as providing any added value. In this sense, we had 51.24% for positive valence in binomial classification and 20.78% for happy mood in multinomial classification.

Naive Bayes (NB). Though the objective of this work is to develop a deep learning model, it is a good practice to compare the result with a more classical approach, based on a simpler model like NB, and use it as a second baseline or reference. We performed both binomial and multiclass NB classification, using Countvectorizer -unigram and bigram- and Tf-idf to find features from lyrics. This models are easy to build and fast to train and use, moreover with a small dataset. We used

CNN. Using the idea first presented in [3] to classify text, a Convolutional Neural Network was applied. First, we cut off lyrics after 250 words, use embedding size of 200. Our CNN model used a (0.2) spatial dropout after the embedding layer prior to a 1 dimensional convolutional (64 filters with window size of 5) and MaxPooling layers. Finally, after a flatten layer we built a dense layer with dropout before the output one. Adam method was used as optimizer and rectifier linear unit (‘relu’) as activation function, except in the last layer, where sigmoid/softmax was used for binomial/multiclass cases respectively.

RNN. Recurrent Neural Network were used in this project to learn the sequential information of the song lyrics, remembering the previous information in hidden states and connecting it to our classification task. Specifically, we used a Bidirectional Long Short Term Memory(Bi-LTSM)network, a subclass of RNN. After an initial embedding layer size 200 and 1D features map dropout (0.2 spatial dropout), we used a 64 dimensionality LSTM per direction, a dense layer, 0.2 dropout and, finally, the output layer. As in the CNN models, Adam method was used as optimizer and rectifier linear unit (‘relu’) as activation function, except in the last layer, where sigmoid/softmax was used for binomial/multiclass cases respectively. Finally, binary/categorical cross entropy was used as the loss function for binomial/multiclass cases respectively.

BERT. A recent development in NLP, BERT is a model pre-trained on tasks and corpora. It has achieved great results both broadly and for this specific task. The underlying model used a pooled output layer, a single hidden layer, then a dropout layer (0.1) and lastly a softmax layer. The model can be adjusted via its hyperparameters. The best results were obtained when the warm-up proportion was 0.1, batch size was 16, the learning rate was 0.000002 (2e-6).

Results and discussion

In order to analyze and compare the results of the models, we divide the models in a) positive/negative mood classification and b) four mood categories classification.

Positive vs negative mood classification

The metrics for every model on the test dataset are shown in table 2. Every single model beats the baseline and each accuracy is in the 70-78% range. It is interesting to notice that a very simple Naive Bayes model beats every other model, except BERT. Regarding the deep learning neural network models, the results improve as more recent approaches in the field are used. This is true not only for the absolute metrics values, but for the homogeneity for positive vs negative values, especially compared with NB model: precision and recall values for NB model are in [0.61, 0.91] range, while they are in [0.75, 0.80] in the case of BERT model. Finally, it is also interesting that all models are better (f1-score) when dealing with negative lyrics -lower valence, *angry* and *sad*- than with positive ones.

For future work, it could make sense to research whether an ensemble NB-BERT model could lead to better results.

Table 2: positive/negative classification metrics per model (best/worst results in bold **black**/**red** color)

Model	Metrics				
	class	precision	recall	f1-score	#
Baseline	Accuracy: **0.48**				515
NB	pos	**0.88**	**0.61**	**0.72**	265
	neg	**0.69**	**0.91**	0.78	250
	Accuracy: 0.7553				515
CNN	pos	**0.65**	**0.82**	0.72	155
	neg	0.78	**0.59**	0.78	169
	Accuracy: **0.7006**				324
Bi-LSTM RNN	pos	0.72	0.72	0.72	155
	neg	0.74	0.74	**0.74**	169
	Accuracy: 0.7315				324
BERT	pos	0.80	0.75	**0.76**	256
	neg	**0.795**	0.77	**0.78**	259
	Accuracy: **0.7825**				515

Four emotion categories classification

The metrics for every model on the test dataset are shown in table 3.

Table 3: four emotion categories classification metrics per model

Model	Metrics				
	class	precision	recall	f1-score	#
Baseline	Accuracy: **0.2563**				515
NB	happy	0.59	0.74	**0.66**	132
	angry	0.84	0.72	**0.77**	143
	sad	0.51	0.66	**0.58**	122
	relax	0.57	0.33	0.42	118
	Accuracy: **0.6233**				515
CNN	happy	0.60	0.62	0.61	90
	angry	0.63	0.79	0.70	85
	sad	0.46	0.46	0.46	70
	relax	0.60	0.42	0.49	79
	Accuracy: 0.5802				324
Bi-LSTM RNN	happy	0.53	0.57	**0.55**	90
	angry	0.68	0.66	**0.67**	85
	sad	0.35	0.34	**0.35**	70
	relax	0.52	0.51	**0.51**	79
	Accuracy: **0.5277**				324

Every single model beats the baseline -*happy* mood for every prediction- and accuracies are in the 52-62% range. However, in this multi-classification task, we have not been able to build a deep learning model which can beat a simple NB model in terms of accuracy: the only class where a deep learning model got a better f1-score than the NB one was for *relax* class. In this multinomial classification, best results are found for high arousal -*happy* and *angry*-. This makes *angry* mood the easiest to classify. Taking a look at the normalized confusion matrix for the Bi-LSTM RNN model in table 4, we can see how *sad* and *relaxed* emotions are often confused with each other or classified as *happy*.

Table 4: Bi-LSTM RNN normalized confusion matrix (percentages)

True\Pred	Happy	Angry	Sad	Relaxed
Happy	**56.67**	14.44	13.33	15.56
Angry	10.59	**65.88**	17.65	5.88

Sad	25.71	14.29	**34.29**	25.71
Relaxed	22.78	3.80	22.78	**50.63**

Discussion

The fact that *angry* songs are the easiest to classify is explained because of the very frequent use of rude vocabulary and expressions that are not common in any of the other classes. However, the boundaries between a *sad* and *relaxed* song, or between a *relaxed* and *happy* song are not always so clear, even for a human being.

To make a deep learning network be able to capture those subtle nuances for a multinomial classification, a larger corpus is recommended: we trained them with 1850 songs lyrics, which means roughly 925 songs lyrics per class for the multinomial classification and just 462 for the binomial one. It is our understanding that this last value has turned out to be scarce and prevented better results in the multinomial task.

Other likely reasons which prevent the algorithms to have a better performance in this task might be the following:

1. Use of two databases with different labelling criteria (human vs automated annotation), though this was needed to increase the size of the corpus as much as we were able to.
2. The fact that some songs are really a story with an outcome not known until the end may require to take the whole song lyrics and not only the first part of them in the very long ones.
3. A song has lyrics and audio and this task is only focused on the first one. Better results were reported in [2] when an approach like ours was combined with audio analysis.

Conclusion

For binomial classification, we developed a deep learning model which was able to beat the classical model, which it is a better result than the one reported in [2], where a combination of lyrics and audio models was required to beat a classical NLP approach -on lyrics only-. On top of that, our BERT model got a **78.25%** accuracy on positive/negative (valence), beating the 76.29% maximum accuracy on valence reported in [6] and even the 77.23% reported on arousal, which we did not train for.

However, for multiclass classification, we were not able to beat a classical model. The greatest confusion of the model was found between *relaxed* and *sad* emotions. A larger corpus would probably help dealing with this difficulty.

References

[1] Çano, Erion; Morisio, Maurizio (2017). MoodyLyrics: A Sentiment Annotated Lyrics Dataset. In: 2017 International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence, Hong Kong, March, 2017. pp. 118-124 (Link [1]).

[2] Delbouys, R., Hennequin, R., Piccoli, F., Royo-Letelier, J., Moussallam, M. (2018). Music Mood Detection Based On Audio And Lyrics With Deep Neural Net. arXiv e-prints arXiv:1809.07276 (Link [2]).

[3] Kim, Yoon (2014). Convolutional Neural Networks for Sentence Classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics (Link [3]).

[4] Malheiro, Ricardo & Panda, Renato & Gomes, Paulo & Paiva, Rui Pedro. (2016). Classification and Regression of Music Lyrics: Emotionally-Significant Features. 45-55. 10.5220/0006037400450055 (Link [4]).

[5] Russell, James A.(1980). A circumplex model of a_ect. Journal of personality and social psychology, 39(6):1161-1178, 1980 (Link [5]).

[6] Zaanen, Kanters (2010). Automatic mood classification using tf*idf based on lyrics. Proceedings of the 11th International Society for Music Retrieval Conference, pages 75-80 (Link [6])