Emotion Classification for Cross-lingual song lyrics

Maliha Bushra Hoque
Dept. of CSE
BRAC University
Dhaka, Bangladesh
maliha.bushra.hoque@g.bracu.ac.bd

Sanjoy Dev
Dept. of CSE
BRAC University
Dhaka, Bangladesh
sanjoy.dev@g.bracu.ac.bd

Shishir Kumar Das
Dept. of CSE
BRAC University
Dhaka, Bangladesh
shishir.kumar.das@g.bracu.ac.bd

***Abstract*—Online music databases are available throughout the whole world, but these databases can not get the emotions from the music based on the contents of the lyrics. That is why many companies have come up with a solution for getting emotions from songs and categorizing them with the help of human experts [1]. Though this process was not that efficient. This paper is an attempt to detect the emotions of songs without any human experts, only using machine learning. Classification models have been applied and the songs are categorized into five emotional moods. Our results are encouraging for mining the lyrical content of songs for specific emotions since they produce classification models that are understandable to humans and produce conclusions that are consistent with commonsense perceptions of particular emotions. One area of research that integrates various classifiers of musical emotion, such as acoustics and lyrical content, is the meaning of lyrics, which is the topic of this work.**

***Index Terms*—Machine Learning, Text Mining, Text Classification, Categorize, Musical Emotions, Lyrical Text, Lyrical Content.**

I. Introduction

With the advancement of computers and Internet technology, the productivity of digital sound sources is rising every day. Additionally, the arrival of smartphones has improved access to music. In fact, several businesses are attempting to profit from this situation by providing their clients with music through such phones. Text-based sentiment analysis has become more popular in recent years because of its applications in fields such as marketing, politics, and psychology. Most sentiment analysis methods are developed to identify net positive or negative sentiment rather than more subtle, ambiguous feelings like surprise, nostalgia, or anticipation (Jongeling et al., 2017). Because of this, current models frequently do not properly reflect the existence of many emotions within a text sample, leading to a local representation of a person's actual feelings. Songs are frequently written to provoke deep emotional reactions from listeners, making this an attractive field of study for those interested in understanding complex emotions. The current music search techniques for many music sources have progressed beyond the search for fundamental music knowledge to suggestions of music that people want to hear. The metadata gathered from music information searches is then used to create an index, including the genre, title, lyric, singer, and album name. When it comes to music recommendation techniques, there are algorithmic measures of popularity, approaches based on the most recent registration, and user-defined albums where users pre-select songs. Researchers have long researched the classification of musical genres, but they have only recently become interested in the classification of emotions obtained from music [2] [3] [4]. Previous studies on the categorization of emotion have made use of musical elements such as rhythm, speed, beat, song, and musical notes. But lyrics are also significant: Even when melodies are similar, songs' lyrical contents might have a varied effect on listeners [5].

II. Research Objective

The selection of music for public spaces like hospitals or restaurants to perhaps increase the emotional well-being of staff, patients, or patrons is only one example of how sentiment prediction of contemporary music might be used in modern culture. Moreover, in the past, humans used songs for different important purposes besides entertainment. For example, many rebels created different songs during the different revolutions, the same as war songs, political songs, and social awareness songs. Songs represent both culture and history which is a great advantage to understanding a nation or particular groups of people. Even though in this research we focused more on the happy and sad parts of emotions but same can also be implemented for many other different cases.

III. Problem Statement

The goal of this project was to create a recommendation system that can determine if a song is happy or sad based on its lyrics for one or multiple languages. This system may be used to search song libraries and choose music based on emotion in a variety of social scenarios. In the system, 1. Using a naive Bayes classifier to predict song sentiment solely from song lyrics. 2. Using the algorithm as the foundation for a music recommendation system. 3. Identifying cheerful music with high accuracy using text qualities gleaned from song lyrics.

IV. Literature Review or Related Works

Numerous research articles and publications inspired us to work on this issue; a few are covered in depth here. The authors Minho Kim and Hyuk-Chul suggested feature selection classification as a method of identifying lyrics-based emotion. The methods used to find music nowadays have advanced beyond the search for basic musical knowledge to include recommendations of music that listeners might enjoy. The genre, title, lyric, singer, and album name are all included in the index that is made using the metadata received from music information searches. There are algorithmic measurements of popularity, methods based on the most recent registration, and user-defined albums where users pre-select songs when it comes to music recommendation systems. Recently, a method for producing music recommendations based on the exact emotions and moods that customers prefer has become more and more common. Authors Serhat Hizlisoy a, Serdar Yildirim b, and Zekeriya Tufekci suggested employing convolutional long short-term memory deep neural networks to recognize the emotions in music. An annotated emotional music database is required in order to construct a music emotion recognition system. The two methods of categorizing emotions in music are category and dimensional. In the categorical method, discrete names for emotions like sadness, happiness, anger, and fear are used to describe them [2]. Emotions are portrayed in dimensional space in the second method. Arousal and valence are the two dimensions of Russell's [3] paradigm. In the dimensional method, Russell's circumplex model, which Thayer's model [4] is based on, incorporates dimensions for valence and arousal., and is typically utilized for musical definition. These components alter the underlying stimuli that may affect the reactions in mood [5] [6] [7] [8]. While the valence axis shows the degree of pleasure against displeasure, the arousal axis depicts emotions in a spectrum from calm to exuberant. Since category numbers are not agreed upon, categorical models are troublesome. With dimensional models, there is no such issue. Comparing dimensional models to the categorical approach, the advantage is the reduced uncertainty. This paradigm offers a valid method for categorizing emotion into two different categories. Therefore, the two-dimensional model is used to annotate musical excerpts. Panda et al. [9] offered a library of 903 audio samples that were classified into 5 emotion groups using the MIREX mood classification task. Y.-C. Lin et al. created a database of 7922 musical compositions utilizing tags from AMG and labeling them with 183 emotional descriptions [11]. There are 1240 pieces of Chinese pop music that have been assessed for valence and arousal in a database created by Y.H. Yang et al. [13]. M. Soleymani et al collection's of 744 songs make up another sizable database [14]. A selection of music snippets from the free music archive was used in this database (FMA).

V. Dataset, Data description, and pre-processing

The dataset was collected from Kaggle. The dataset used here has 18000 songs, which were categorized. The features here are Artist, Title, Year, Genre, Lyrics, Length, and Labels. These 7 categories are retrieved from the dataset. From the Labels category, 5 types of emotion labels are found. These emotions are - Sadness, Romance, Violence, Obscene, and Feelings.

Any procedure performed on raw data to prepare it for subsequent processing is referred to as ”data preparation”. This has been regarded for a long time as the most important first stage in the data mining process. Data preprocessing techniques have been used more recently to train artificial intelligence and machine learning models as well as to execute judgments against those models. Machine learning and AI development pipelines have used these techniques, as they can provide definitive conclusions.

The following procedures have been employed to preprocess the data:

- **Label Encoder** - Labels are normalized using LabelEncoder. It is a significant supervised learning preprocessing step for the structured dataset. Label Encoders are vastly used to turn labels into numerical ones, in the case when they are hashable and comparable. This means, to make it quite understandable by the machines, labels are transformed into the numeric form [15]. It makes the action better for the machine learning process to be comprehensive.

- **Porter-Stemmer** - Hybrid Porter-Stemmer-Tokenizer breaks down sentences into words (or tokens), and then each token is subjected to Porter stemming algorithm. Additionally, tokens made up only of punctuation characters are eliminated. The only tokens that are saved are those that have more than one letter [16]. Its primary function is to normalize terms as part of the process of building up information retrieval systems, which is what it does. The NLTK library is imported to use the Porter-Stemmer process.

- **Stopword removal** - Stopword removal is hugely applied among the preprocessing techniques in numerous NLP implementations. Stop words add noise to the features most of the time. In order to create a cleaner dataset with better features for a machine learning model, stop words should be removed. Stop words did not impact the used dataset here [17]. That is why the decision of removing them was taken. Although, this preprocessing is not mandatory for every dataset.

- **Count Vectorization** - This process of vectorization involves counting instances of each word in a document. The process actually makes it simple to use text data in machine learning directly. Count Vectorizer breaks down a sentence or any text into words. This includes conducting preprocessing operations, changing all words to lowercase, and deleting special characters [18]. Documents and texts are converted into term or token counts in Count Vectorization.

- **Tfidf Vectorization** - The Term Frequency and Inverse Document Frequency (TF-IDF), help to measure the importance of a word by analyzing a dataset. The term frequency refers to how frequently each word appears in the text or dataset. And the term inverse document frequency analyzes the importance of the word in a dataset [19].

  TF(t) =$\frac{(Totalnumberofappearancesoft)}{(Summationoftermsinthedocument)}$

  IDF(t) =$\log_{10}(\frac{TotalDocuments}{Summationofthedocumentswithterm`t'})$

  TF * IDF = $\frac{(Totalnumberofappearancesoft)}{(Summationoftermsinthedocument)}$ X

  $\log_{10}(\frac{TotalDocuments}{Summationofthedocumentswithterm`t'})$

- **The α smoothing parameter** - The smoothing constant is Alpha. The speed of smoothing depends on the value of Alpha. This value of Alpha is between 0 to 1. The process of smoothing proceeds more slowly when it is close to 0. Where the demand forecast is more sensitive when the smoothing constant is higher. In that case, the dataset will be larger [20]. Also, working with the value of Alpha being zero, the set of the previous smoothed value is the current smoothed point, while dealing with the value of Alpha being one (1), the set to the latest point is the current point.

VI. Methodology

A. Model Selection

For lyrics classification for cross-lingual songs, we chose three Naive Bayes models. Naive Bayes classifiers are known for being productive with small sample sizes [21] and are currently being utilized for tasks that require the categorization of binary text, such as e-mail spam detection [22]. Additionally, some research investigations have demonstrated that these classifiers perform similarly to SVMs for categorizing text [23] [24]. On the 3765 song training set, grid search and cross-validation were used to assess the model performances using a number of feature combinations and preprocessing methods. Naive Bayes is a type of supervised learning and it is based on Bayes theorem.

From the general posterior probability for Naive Bayes classification we know:

posterior probability = $\frac{(ConditionalprobabilityXpriorprobability)}{evidence}$

$P(a_j|x_i) = \frac{P(x_i|a_j)P(a_j)}{P(x_i)}$

The goal of this model is to maximize the posterior probability from the training dataset. Here, the class-conditional probability of witnessing feature x_i belonging to class a_j is given by $P(x_i|a_j)$:

predicted class label¸$argmax_{j=1,\ldots,n}P(a_j|x_i)$

Multi-Variate Bernoulli Bayes(MVB):

A discrete probability function with conditional properties is known as a class conditional probability function (for a discrete random variable). Based on the binary feature vectors, the class conditional probability of the MVB Naive Bayes model's class-conditional probability is shown:

$P(x|a_j) = \Pi_{i=1}^{m} P(x_i|a_j)^b (1P(x_i|a_j))^{1-b}$

$\hat{P}(x|a_j) = \frac{df_{x\,i,y}+\alpha}{df_y+\alpha n}$

Suppose, $P(a_j|x_i)$ is the maximum-likelihood of a particular token x_i, belongs to class a_j. For the second one, x_i is the feature of the training dataset which belongs to the class a_j. On the other hand, $df_{xi,y}$ is the amount of lyrics of the training dataset. Again, df_y is the number of lyrics that contains the feature x_i and belongs to class a_j. Finally, n represents the feature vector's number of components.

Multinomial Naive Bayes (MNB) with term frequency features:

The Multinomial Naive Bayes classifier (MNB) is appropriate for word counts of text classification. Based on the term frequencies and inverse term frequency, or tf-idf, a multinomial naive Bayes model was assessed. In place of binary numbers, the term "frequency" (tf(t, d)) can be used to represent text files. It treats a document D as a feature vector of vocabulary-sized x, where the number of terms in the x_i document D is represented by each member. This vector x then follows a multinomial distribution by definition, giving rise to the distinctive classification function of MNB.

$P(x_i|a_j) = \frac{\sum tf(d\epsilon a_j)+\alpha}{\sum N_{d\epsilon a_j}+\alpha n}$

Here, $N_{d\epsilon aj}$ is the summation of the term frequencies of the training dataset under aj. The naive analysis of conditional independence between features allows the Bernoulli and Multinomial naive bayes models to calculate the class-conditional probability of the lyrical text x as the product of the probabilities of the distinct terms.

$$P(x|a_j) = P(x_1|a_j)P(x_2|a_j)P(x_n|a_j)$$

Gaussian Naive Bayes

In the Naive Bayes classification, we can deal with continuous attributes using Gaussian distribution [25]. Again, it also represents the likelihoods of the features based on the classes. A Gaussian Probability Density Function may therefore help us to characterize each feature as

$$X_i \sim N(\mu, \sigma 2)$$

The Gaussian PDF is also known as Normal Distribution and it has the shape of a bell. The equation can be defined as:

$$N(\mu, \sigma 2)(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Here, mean and variance are represented by μ and $\sigma 2$ respectively. Naive Bayes requires parameters of the order of O(nk), where n represents the number of features and k as the number of classes. We must define a normal distribution in particular. $P(Xi|C)$ N(μ, σ2). These parameters can be gained with:

$$\mu X_i|C_{=c} = \frac{1}{N_c} \sum_{i=1}^{N^c} xi$$

$$\sigma^2 \mu Xi|C_{=c} = \frac{1}{N_c} \sum_{i=1}^{N^c} xi^2 - \sigma^2$$

Here, N is the total amount of lyrics used for training set and N_c is the number of lyrics where C = c. We can calculate, P(C = c) for all classes using relative frequencies such that

$$\text{P(C = c)} = \frac{N_c}{N}$$

B. Description of the implementation

The training data includes words with 33.2 percent violence, 28.57 percent sadness, 33.56 percent obscene, 4.60 percent romance, and 2.75 percent sentiments. After pre-processing, sadness, violence, and obscene words were marked as Negative words, and the rest were marked as positive words. The percentage was the same for all in the validation dataset. Initially, the dataset consisted of around 10,000 data. The final model's performance was assessed using the 450-song validation dataset after it had been trained using the whole training dataset. However, the model was selected using grid search and cross validation on a 2054 training dataset in order to maximize performance via F1-score. The receiver operating characteristics area under the curve (ROC auc), recall, accuracy, F1-score, and precision were the five metrics used to evaluate performance. Grid search on three distinct naïve bayes models is the first step. These three models used Multi-Variate Bernoulli Bayes, Multinomial Bayes, and Gaussian NB for feature extraction. The three models used term frequency features instead of term frequency counts as feature vectors. Each of the three models was independently optimized using grid search, and the performance of the top models in each of the three groups was then evaluated using ROC analysis. Finally, the best model was picked for grid search optimization.

VII. Experimental Result Analysis:

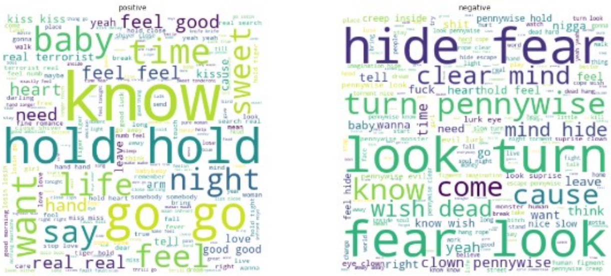

Figure 1

The most frequent words from the above word cloud visualizations from the training dataset are (Posi ve: know, hold, go, feel) and (Negative: hide, fear, look, turn) (Figure 1). Plotting according to the music genre, it was, noticed that more negative words are used than, positive ones. Only 106 words out of 2054 were positive, out of the total 1948 negative terms. In Figure 2 The most negative words are used most frequently in the "pop" genre.

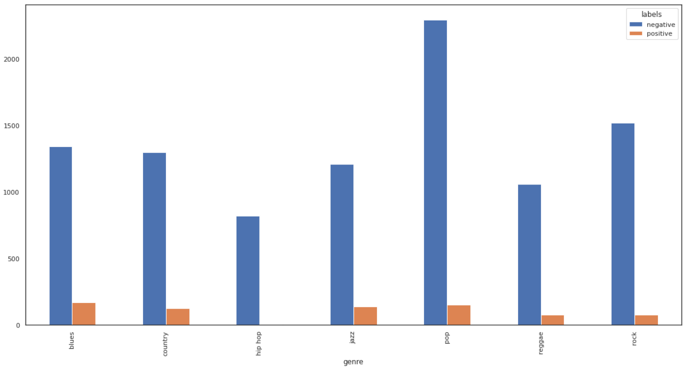

Figure 2

Following the grid search, three different Naive Bayes classification models performed nearly equally, as shown in (Figure 3). After performing necessary preprocessing steps such as extra stop word removals and Porter stemming for suffix stripping, the multinomial naive Bayes classifier with tf-idf feature representation had the best performance (average ROC auc 0.70). Further investigation found that the smoothing parameter, the minimum term frequency cut-off value, and the maximum vocabulary size had no significant effect on the execution of the chosen classification model (Figure 3c to e). However, the attempt to improve the classification performance by extending the n-gram range was clearly detrimental (Figure 3f). Following model selection, the ultimate classifier was trained on the complete dataset while performance w evaluated using the validation dataset. The precision performance of the mood classifier was 93.84 percent on the training set and 4.23 percent over 450 validation sets which indicates that it might be overfitting.

	ACC(%)	PRE(%)	REC(%)	F1(%)	ROC AUC(%)
Training	78.48	93.84	48.50	68.45	74.80
Validation	45.53	84.23	10.82	28.89	48.57

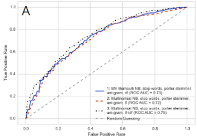

Figure 3a

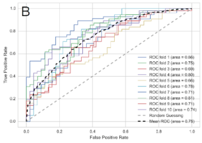

Figure 3b

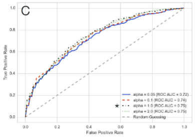

Figure 3c

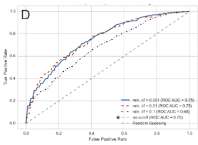

Figure 3d

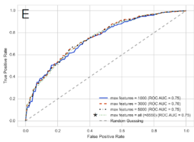

Figure 3e

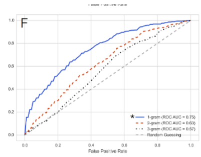

Figure 3f

VIII. Conclusion

In this paper, we investigate various methods for the multi-lingual emotion classification of music. The process of emotion classification was examined in this study using emotion ontology and machine learning techniques. The current study, in contrast to other research, used emotions derived from song lyrics as learning elements. The findings indicate that a naive Bayes model can accurately predict the positive class (happy) in lyrics based on mood categorization, which might be useful for searching a vast music collection for cheerful music with a low false positive rate. A music library that has been filtered in this way may also be used as an input for genre classification to filter music based on individual tastes. Extensions to the mood classification web application to include more features are planned for the future. ROC curves of various lyrics classification models were tested using cross-validation on a dataset of 1,000 randomly selected songs. Songs that were accurately rated as cheerful and those that were incorrectly classified as sad were used to compute the true positive rate and false positive rate, respectively.

References

[1] Juslin, Patrik N. “What Does Music Express? Basic Emotions and Beyond.” Frontiers, 16 Aug. 2013, www.frontiersin.org/articles/10.3389/fpsyg.2013.00596/full.

[2] LyricWikia - http://lyrics.wikia.com/Lyrics$_{W}iki$.

[3] Million Song Dataset - http://labrosa.ee.columbia.edu/millionsong/pages/getting-dataset.

[4] musicXmatch - http://labrosa.ee.columbia.edu/millionsong/musixmatch.

[5] Steven Bird. NLTK: the natural language toolkit. In Proceedings of the COLING/ACL on Interactive presentation sessions, pages 69–72. Association for Computational Linguistics, 2006.

[6] Pedro Domingos and Michael Pazzani. On the optimality of the simple Bayesian classifier under zero-one loss. Machine learning, 29(2-3):103–130, 1997.

[7] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, pages 1–12, 2009.

[8] Zellig S Harris. Distributional structure. Word, 1954.

[9] Sundus Hassan, Muhammad Rafi, and Muhammad Shahid Shaikh. Comparing SVM and Naive Bayes classifiers for text categorization with Wikitology as knowledge enrichment. In Multitopic Conference (INMIC), 2011 IEEE 14th International, pages 31–34. IEEE, 2011.

[10] Perfecto Herrera, X. Amatriain, E. Batlle, and Xavier Serra. Towards Instrument Segmentation for Music Content Description a Critical Review of Instrument Classification Techniques. International Conference on Music Information Retrieval, 2000.

[11] Yajie Hu and Mitsunori Ogihara. Genre classification for million song dataset using confidence-based classifiers combination. In Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval, pages 1083–1084. ACM, 2012.

[12] J D Hunter. Matplotlib: A 2D graphics environment. Computing In Science Engineering, 9(3):90–95, 2007.

[13] Pieter Kanters. Automatic mood classification for music. PhD thesis, Master’s thesis, Tilburg University, Tilburg, the Netherlands, 2009.

[14] Tao Li and Mitsunori Ogihara. Music artist style identification by semi-supervised learning from both lyrics and content. In International Multimedia Conference: Proceedings of the 12 th annual ACM international conference on Multimedia, volume 10, pages 364–367, 2004.

[15] Team, Great Learning. “Label Encoding in Python Explained.” Great Learning Blog: Free Resources What Matters to Shape Your Career!, 16 Dec. 2021, www.mygreatlearning.com/blog/label-encoding-in-python.

[16] Singh, Surya Pratap. “Porter Stemmer Algorithm.” OpenGenus IQ: Computing Expertise Legacy, 21 May 2019, iq.opengenus.org/porter-stemmer.

[17] Malik, Usman. “Removing Stop Words From Strings in Python.” Stack Abuse, 5 Mar. 2020, stackabuse.com/removing-stop-words-from-strings-in-python.

[18] Sung, Joshua. “Natural Language Processing: Count Vectorization and Term Frequency — Inverse Document Frequency.” Medium, 24 Apr. 2018, medium.com/@joshsungasong/natural-language-processing-count-vectorization-and-term-frequency-inverse-document-frequency-49d2156552c1.

[19] Karbhari, V. (2020, February 26). What is TF-IDF in Feature Engineering? Medium. medium.com/acing-ai/what-is-tf-idf-in-feature-engineering-7f1ba81982bd

[20] Rozhenko, Alexander. “A New Method for Finding the Optimal Smoothing Parameter for the Abstract Smoothing Spline.” Elsevier Enhanced Reader, 22 Aug. 2009, reader.elsevier.com/reader/sd/pii/S0021904509001269

[21] Pedro Domingos and Michael Pazzani. On the optimality of the simple Bayesian classifier under zero-one loss. Machine learning, 29(2-3):103–130, 1997.

[22] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, pages 1–12, 2009

[23] Sundus Hassan, Muhammad Rafi, and Muhammad Shahid Shaikh. Comparing SVM and Naive Bayes classifiers for text categorization with Wikitology as knowledge enrichment. In Multitopic Conference (INMIC), 2011 IEEE 14th International, pages 31–34. IEEE, 2011.

[24] Mehran Sahami, Susan Dumais, David Heckerman, and Eric Horvitz. A Bayesian approach to filtering junk e-mail. In Learning for Text Categorization: Papers from the 1998 workshop, volume 62, pages 98–105, 1998.

[25] Mitchell, T: Machine Learning. McGraw-Hill, 1997.