STONY BROOK UNIVERSITY

Music Genre Classification Using Lyrics

Raghav Arora
Sameer Gupta

May 22, 2014

1 INTRODUCTION

1.1 MOTIVATION

Lyrics of a song contains some vital information that is imperative for comprehending the message being passed by the song. We strongly believe that lyrics exhibit some easy to understand structures that, in case of its audio, would be difficult to analyze.
Music classification system is used by online music providers like iTunes, Spotify to recommend music to the user based on the previous history of the user or the current tracks in his playlist to suit the user's mood but most of these applications classify the song based on different "tags" created by users itself. This brings us to the thought of why not classify the songs based on its lyrics itself since the words and the structure of the lyrics indicates the mood and emotions.
As a part of course project, we develop classifier for classification of songs into different genre and further predict the decade when the song was released.

1.2 PROBLEM STATEMENT

A lot of work has already been carried out for "genre classification" problem. So, what is new that we are trying to do ?
Most the the previous work is based on audio features, or a combination of audio and lyrics. Our project is based on "only lyrics" based classification. The primary focus of the project was to develop and implement classifiers that could be used to classify the songs into six different genres; country, jazz, metal, alternative, electronic, folk. We also classify the songs according to the year when it was released using a different set of data. Along with this, we

also tried to predict the decade in which the song was released but due to limitation in the size of the corpus for this problem, we decided to predict the songs into one of these time periods - (1950-1970, 1970-1990, 1990-2010). The data on which the experiment was performed is described in next step. We added new features to make classification more accurate using other non-vocal attributes of the songs for both the cases.

As an initial step, we started with k-means clustering, k-nearest neighbors algorithm for classification based on the lyrics of the songs. We clustered the songs into six different clusters and three clusters in case of year prediction. Later, we analyze the results and try to improve them using different NLP techniques like Naive Bayes, Maximum-Entropy, POS tagging and SVM(Support Vector Machine) and developing new feature vectors incrementally.

Along with this, we analyzed the relationship between the different attributes of the songs and its effect on the problem we are trying to resolve. For example - analyze if length of the song has a relation with its genre or how will our classifier work if we include this property of the song along with the lyrics. Similarly, many different components of the lyrics of the songs were analysed that are discussed in detail in following sections. Similar kind of steps were followed for the "year prediction" problem to predict the time of release of a song.

Later, we also try to analyze whether there is a measurable change topical words in the song over different years. We observe some interesting trends in the lyrics with time.

Notable Points :

1. We implement 10-fold cross validation for SVM and Maxent classifiers so as to provide more accurate results.

2. We used Weka for k-means clustering and k-NN, Multiclass Support Vector Machine for SVM, and Stanford Classifier for MaxEntropy.

3. We wrote Python scripts to perform all the tasks like extracting the data from MySql or sqlite, creating a uniform dataset, and creating feature vectors to be fed to different tools mentioned above for performing these NLP techniques.

2 DATASET

The dataset for the project was constructed in three different stages.

STAGE I

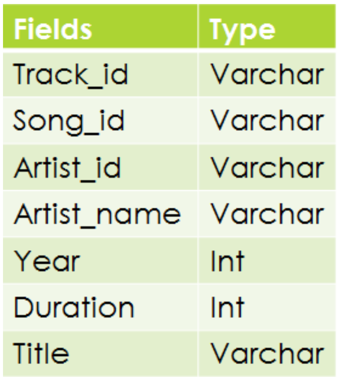
First, we were able to get our hands on Million Song Dataset[12] which is provided by Columbia University researchers with a lot of third party datasets (musiXmatch, lastFm, musiXmatch, The Echo Nest) which contains information like track_id, word_count, song_title etc for the analysis.

In MSD, we have the relevant data distributed among four seperate files :

1. mxm_dataset_train - MusiXmatch dataset, the official lyrics dataset of the Million Song Dataset. The file contains the track_id, mxm_track_id and word count of top words ocuuring in the songs.

2. msd_genre_dataset - The data is collected from Million Song Dataset. The file provides the genre, track_id, artist_name, title, loudness, duration, and many more attributes of the song.

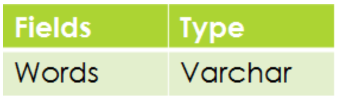
3. mxm_779k_matches - Matches are provided by musiXmatch based on artist names and song titles from the Million Song Dataset. This file contains MSD track id, artist name of MSD, MXM track id, artist name of MXM.
4. File that matches track with the year of release.
Once we have the text files for the dataset, we preprocess this data before using it. The data is divided into 3 databases :
1. **Track_metadata.db**
Songs table – track_id, title, song_id, release, artist_id, artist_mbid, artist_name, duration, year
Artist_Mbtag table – artist_id, mbtag
Artist_Term table – Artist_id, term
2. **mxm_dataset.db**



Fields	Type
Track_id	Varchar
Song_id	Varchar
Artist_id	Varchar
Artist_name	Varchar
Year	Int
Duration	Int
Title	Varchar

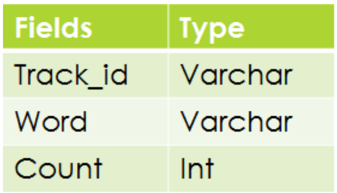
Figure 2.1: Songs Table

Words table – words (list of all the words in all lyrics)
Lyrics table – track_id, mxm_tid, word, count
3. **Lastfm_tags.db**



Fields	Type
Words	Varchar

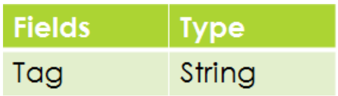
Figure 2.2: Words Table : Contains the list of words in all the lyrics



Fields	Type
Track_id	Varchar
Word	Varchar
Count	Int

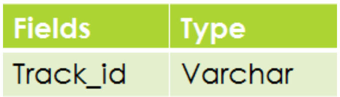
Figure 2.3: Lyrics Table : Contains track_id, the words that appear in that song along with it's count

Last table – tag, tid, tid_tag
The dataset freely available for public and can be downloaded from
http://labrosa.ee.columbia.edu/millionsong/pages/ getting-dataset#subset. .



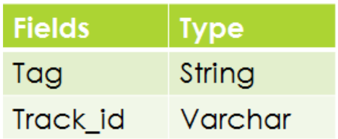
Fields	Type
Tag	String

Figure 2.4: TAG Table : Provides list of all genres present in the dataset



Fields	Type
Track_id	Varchar

Figure 2.5: TID Table : Contains the list of track_ids in the dataset



Fields	Type
Tag	String
Track_id	Varchar

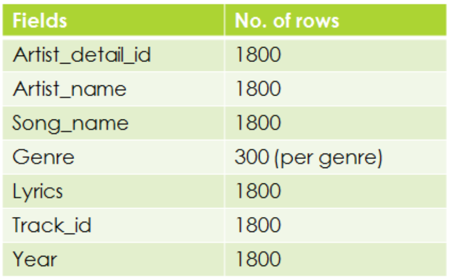
Figure 2.6: TID_TAG Table : Provides the mapping of track_id with it's corresponding genre

Stage II

Eventhough the dataset mentioned above contains a large amount of data, it has certain constraints that makes it inappropriate for use on advanced NLP techniques. The dataset contains bag of words instead of complete lyrics of the songs along with the frequency count of each word in the song. Hence, the dataset cannot be used for bigram, POS tagging, analysis of sentence structure for predicting the etc. For applying such techniques and provide some credible results to support our analysis, we decide to create another dataset by crawling web to fetch complete lyrics of the songs and there corresponding genre.
For this, we used CAL10k dataset[10] that provided us details for 10k songs that included artist name, song name,genre and track_id which can be used to connect MSD with CAL10k data. We requested for the dataset authors[7] and using it with there permission.

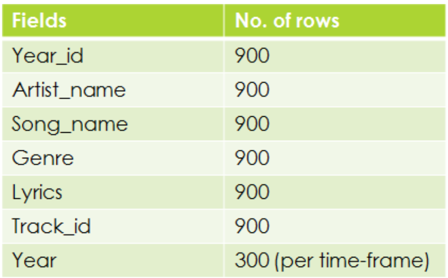
Stage III

Even till this stage, we are facing a major limitation since we do not have a lyrics corresponding to CAL10k dataset. Therefore, to extract the lyrics we created our own web crawler which we used to extract data from some APIs like chartlyrics.com/SearchLyricsDirect.asmx[11] and http://lyrics.wikia.com/api.php[13] which takes artist name and song name as input and returns the complete lyrics as a response. The complete data retrieved was later saved in our MySql database. As some of the track_id's used in CAL10k were misssing in the MSD dataset, we used echonest API[15] to fetch third party data for CAL10k dataset. The final dataset now contains six genres - electronic, metal, folk, alternative, country and jazz .



Fields	No. of rows
Artist_detail_id	1800
Artist_name	1800
Song_name	1800
Genre	300 (per genre)
Lyrics	1800
Track_id	1800
Year	1800

Figure 2.7: Final dataset for genre classification



Fields	No. of rows
Year_id	900
Artist_name	900
Song_name	900
Genre	900
Lyrics	900
Track_id	900
Year	300 (per time-frame)

Figure 2.8: Final dataset for time prediction

3 BASELINE

A lot of analysis has already been done by researchers to classify the genre and other features of music based on the lyrics of the music. We will uniformly split the data in ratio of 8:2 for training and testing respectively. The accuracy would be measured based on the results obtained after running test data on different classification models. We would further analyze the accuracy of different models and if there are any changes observed on using different attributes of songs for classification.

Since we are classifying the lyrics into six different genres and lyrics into six different decades, the baseline would be 16.6% (chosen by randomly distributing genres into six different classes) for genre classification. The baseline was set to 33% for the second problem, since we have 3 different classes in that case.

4 SELECTING NEW FEATURES

I. Bag of Words
Having considered lyrics as most crucial property in classifying genre as well as time prediction of the song, bag of words was supposedly the main feature of our feature vector. We know that each lyricist has his own style of writing and this style may change with the change in genre and time. Different artists are usually inclined towards use of a different vocabulary that could helpful in classifying a song into a particular genre. Further, the usage of words also change with time, that may help us predicting the time of release as well for the song. For example, "country" songs have many occurences of words like village, baby as compared to others. This count may indicate how strongly does a given song may belong to this genre.

II. Part of Speech Tagging
Part of Speech of the lyrics is an important feature that shows the structure of the song. The POS tagging indicates the writing style of the lyrics. We assume that the structure of a song across the a genre would be similar and thus use part of speech as one feature while building the feature vector. For example, more use of noun in a song means more references being made to different objects while high frequency of adjectives indicates that song is very descriptive.

III. Average number of words in a line
The average number of words in a line gives the length of the song. This attribute provides us the rhythm and flow of the song. We assume that shorter length of a lines in a song indicates more emphasis is being given to each word. We included average number of words in a line as one of our features as well.

IV. Average number of lines
We thought that length of the complete song may also give us some information regarding the genre or the time of release of the song in the similar way as above.

V. Duration
Duration of the song is one of the non-lyrics feature that we used to improve our feature vector. It is the entire length of the song in "seconds" that we retrieve from million song dataset. We observed that length of the songs differ with the genre. Along with that we also noticed that length of the songs changed considerably with time as well.

VI. Repetition of words and lines in a song Finally, we decided to add this last feature for genre classification. We observed that "metal" songs had repeated lines in the song. If a line or a word appeared more than once, we increased the count by one with every occurrence. Repetitions were more in Metal and electronic as compared to other.

5 RESULTS

5.1 Genre Classification

We started with 3000 songs from Million Song Dataset(only contained some count of words and not the lyrics) that were evenly distributed among all six genres. We started with k-means clustering followed by k-nearest neighbor. We created a list of words that occurred in more than 80 songs. The feature vector was created based on this list of words. We set the value as '1' if the word exists in the song, else set it as '0'. Next, we performed k-NN technique to classify the songs. The result was expectedly bad in case of k-means (being an unsupervised method) as compared to k-NN.
Moving forward we started with Naive Bayes Classification Model on our final dataset that had 300 lyrics of each of the genres. The dataset was split into 2 parts, training set with 240 songs(for each genre) and test set with 60 songs.
We contructed a feature vector(bag of words) using lyrics and then made an initial analysis and comparison by applying unigram, bigram, k-means, k-NN, SVM and MaxEntropy classifiers.

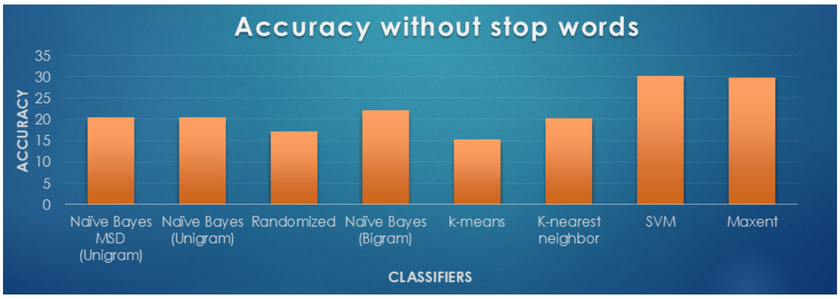


Figure 5.1: Accuracy without stop words

After analyzing the accuracy of our feature vector (frequency of words), we included certain features to further improve accuracy of our classifier. We included the features over bag of words in the following order.

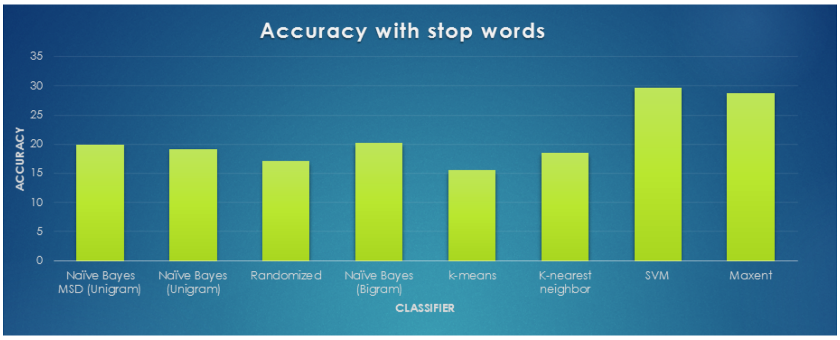


Figure 5.2: Accuracy with stop words

I. Part of Speech Tagging
We used NLTK in Python to tag the words in the lyrics. The grammar was used from "Penn Treebank Tag Set". The results as shown below improved the accuracy both for SVM and MaxEnt when used with the bag of words. This shows that POS is a good indicator of the structure of songs among different genres. Example : Country music that is known for telling stories through song contains most no. of adjectives while electronic has maximum nouns. We also observed that POS structure of electronic was very similar to the metal and led to the confusion for classifier while distinguishing the song into one of these genres.

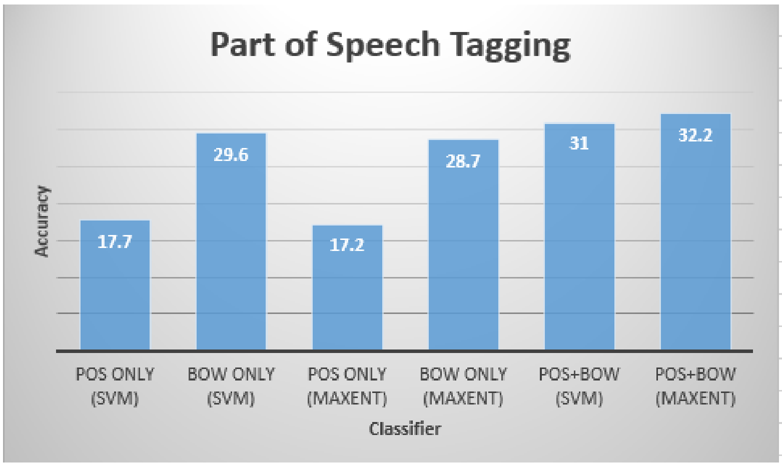


Figure 5.3: Accuracy using Part of Speech Tagging

II. Average number of words in a line
We included average number of words in line in our feature and observed that when used with bag of words, the accuracy of SVM increases on addition of this feature while it was almost similar in case of Maximum Entropy. We are not able to provide a reason for this behavior but can say that addition of this feature helps increasing our overall accuracy.

III. Average number of lines
Number of lines in a song proved to a good feature for our classifier. It was not able to make much difference when used alone but increased the accuracy when used with the bag of words. On further analysis we found that metal had a lowest 31 lines per song, as these are known for its fast paced songs. No. of lines in folk music were maximum while for jazz and electronic, it was same. Since there was a considerable difference in the number of lines for all the genres expect two, it performed well. We also noticed that instead of completely repeating the lines, we have only CHORUS 1, CHORUS 2, etc. written in our lyrics. Therefore, the number of lines for some of the songs was not exactly the same as in original.

IV. Duration Only duration showed some improvement in the accuracy but when used with the bag of words, there was a decrease in the accuracy for both SVM and MaxEntropy. We

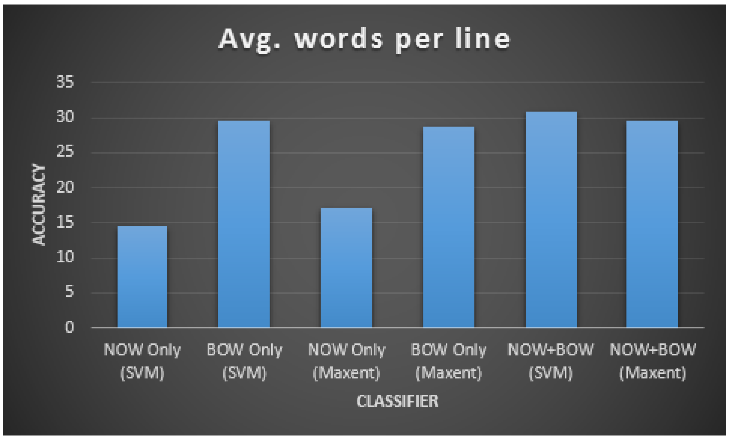


Figure 5.4: Accuracy using Average number of words in a line

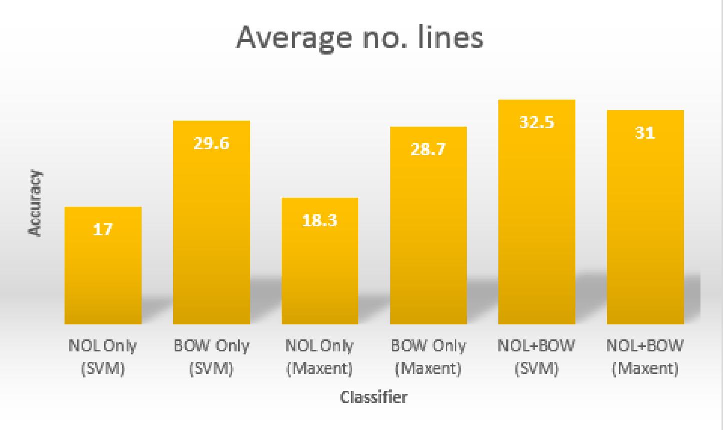


Figure 5.5: Accuracy using Average number of lines

used a normalized value of the duration of the song as the feature vector. Eventhough, there was some difference in the running time of the song from different genres, it may not have improved the accuracy because the change wasn't very significant leading to confusions while classification of a song. Example : Average Duration of Country is larger than Electronic.

V. Repetition of words and lines in a song

We thought that this feature may make a considerable difference for our classifier, but it did not perform well. The result for both SVM and MaxEnt was same without increasing the accuracy of the classifier when used with bag of words. The results could have been a lot different in case we had the complete lyrics (instead of CHORUS 1 CHORUS 2). Complete lyrics would have given us the entire repetition in a song that could make a difference while classification of the song. We observed that in the given dataset, repetition is more in Metal/Electric as compared to rest.

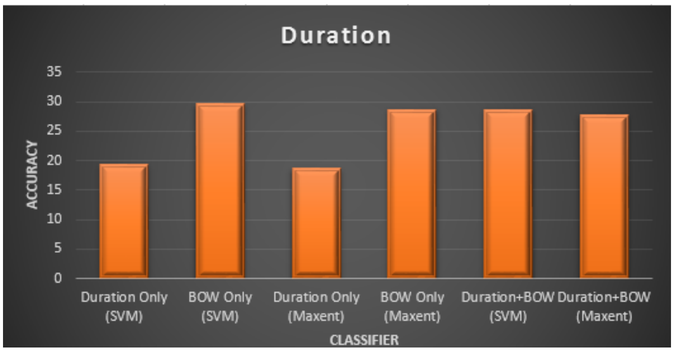


Figure 5.6: Accuracy with duration of a song

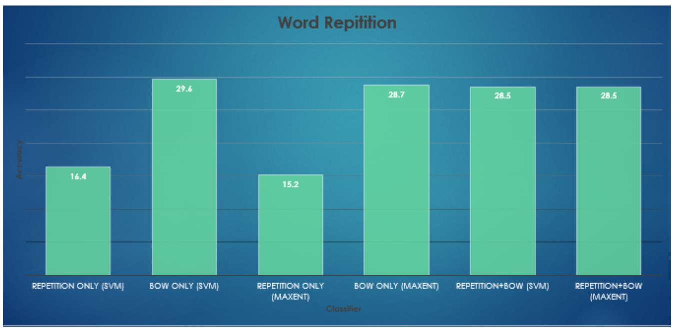
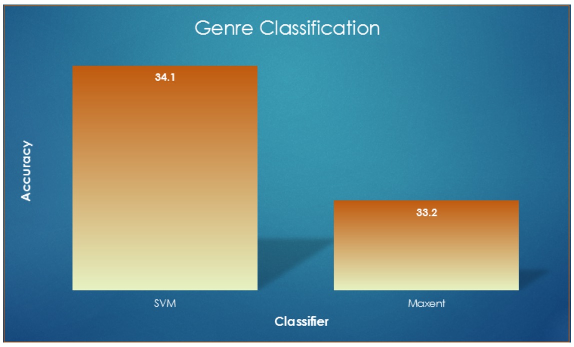


Figure 5.7: Accuracy with repetition of words and line in a song

Finally, we combined all the features and analyzed the performance of our classifier. The performance of SVM for classifying a song in one of the six genres was better than that of Maximum Entropy. After fine-tuning, the C parameter is set to 3*10^3 with L1 regularization for SVM. The result obtained has been shown in the figure below.



5.2 Predicting time of release of a song

We performed similar set of steps for this problem as followed above. The dataset is constructed with the 300 songs from three time-periods (1950-70; 1971-90; 1991-10) having training set of 240 songs and test set of 60 songs from each of 3 classes. We analyzed and compared the result with different features as descibed below.

I. Bag of words

It was the most important feature and all the features were included for it to achieve a better accuracy. We believe that there has been a change in the use of words with time and thus lyrics of the song can best indicate this change. We performed Naive Bayes Unigram and Bigram classification to it.

II. Duration

We observed that there was some noteable change in the length of the song with time. The length of the song increased in 1980s and then started decreasing again. This feature performed well here as compared to its performance in genre classification.

III. Part of Speech Tagging

The results obtained after POS were not in accordance to our intuition. We believed that the structure of the songs would have undergone a considerable change with the time, but the results indicate it being not an effective feature for time prediction. The main reason for this could be the limited data available for this problem. If we manage to get more lyrics from different years, we may get better results using POS tagging.

IV. Average Number of Lines

This feature is somewhat related to the duration of the song. If the duration of the song is more, it is most probable that the number of lines in the song will also be higher. The performance of the feature was also good.

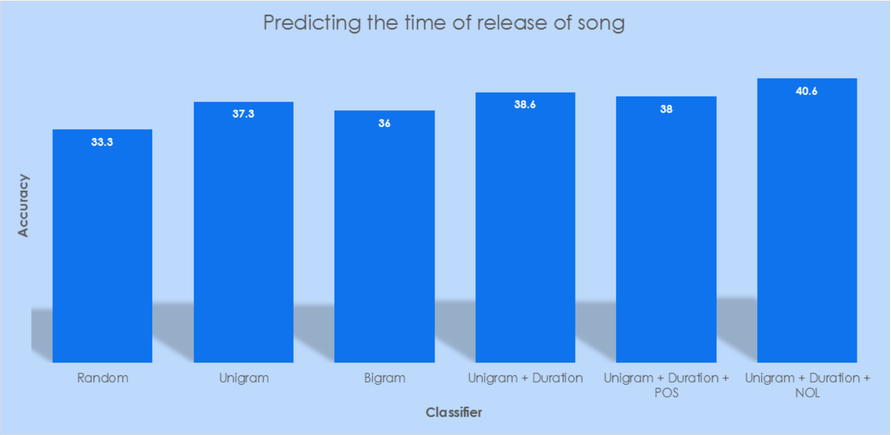
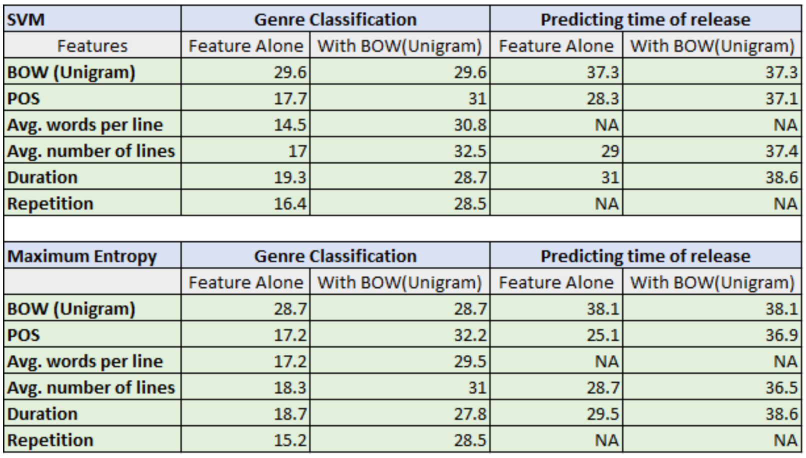


Figure 5.8: Accuracy on combining all feature

The table below shows the final comparitive results(% accuracy) of different classifiers using SVM and Maximum Entropy.



SVM	Genre Classification		Predicting time of release	
Features	Feature Alone	With BOW(Unigram)	Feature Alone	With BOW(Unigram)
BOW (Unigram)	29.6	29.6	37.3	37.3
POS	17.7	31	28.3	37.1
Avg. words per line	14.5	30.8	NA	NA
Avg. number of lines	17	32.5	29	37.4
Duration	19.3	28.7	31	38.6
Repetition	16.4	28.5	NA	NA
Maximum Entropy	Genre Classification		Predicting time of release	
	Feature Alone	With BOW(Unigram)	Feature Alone	With BOW(Unigram)
BOW (Unigram)	28.7	28.7	38.1	38.1
POS	17.2	32.2	25.1	36.9
Avg. words per line	17.2	29.5	NA	NA
Avg. number of lines	18.3	31	28.7	36.5
Duration	18.7	27.8	29.5	38.6
Repetition	15.2	28.5	NA	NA

Figure 5.9: Accuracies in % for all the featues with SVM and Maximum Entropy for Genre Classification as well as Time Predication of the song

5.3 Changes in topical words with genre

In our analysis, we get our hands on few words that dominate a given genre. For example 'baby, momma, wanna, village' are quiet common in country genre where as words like 'life, night' belongs to the electronic genre. Also words like 'f**k', 'death', 'black, 'dirty' dominates the metal genre, which often associate this genre with masculinity, aggression and machismo. This result verifies the apriori knowledge we have about these genres.

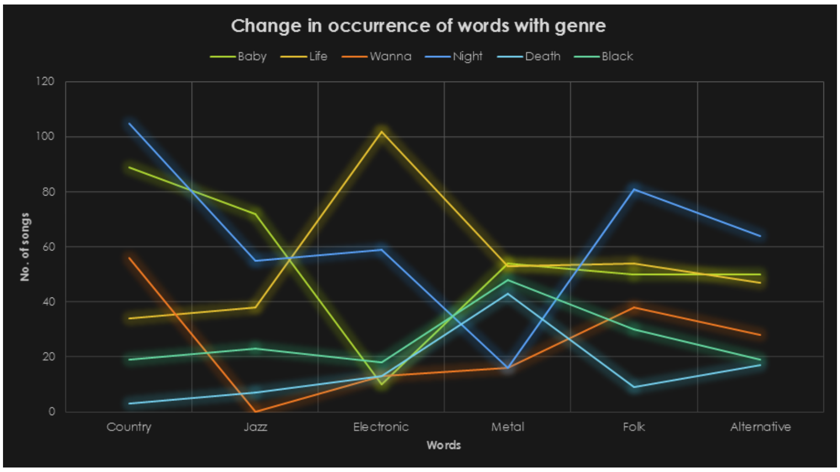


Figure 5.10: Change in topical words with different genres

5.4 CHANGES IN TOPICAL WORDS WITH TIME

We also analyzed the change in topical words over the period of time.For example the words related to 'war/peace' were more common in lyrics in the era of 60s-70s, 'money' in 80s-90s and 'slang words' like [Scrub, Bootylicious, Womanizer, Fly, Tight, No Diggity, Maneater, Hey Ya, Bling Bling] are more common is the lyrics in the current era. Also words like 'love' are quiet commonly used in the lyrics and does not show any significant change with time. These results may be following certain trends like during the time of attack on USA in early this century, we observe the increase in songs containing war/peace. Similarly, with the increase of slangs in our language, same trend is being followed in songs as well. More deeper analysis and larger dataset of lyrics may yield more interesting insights.

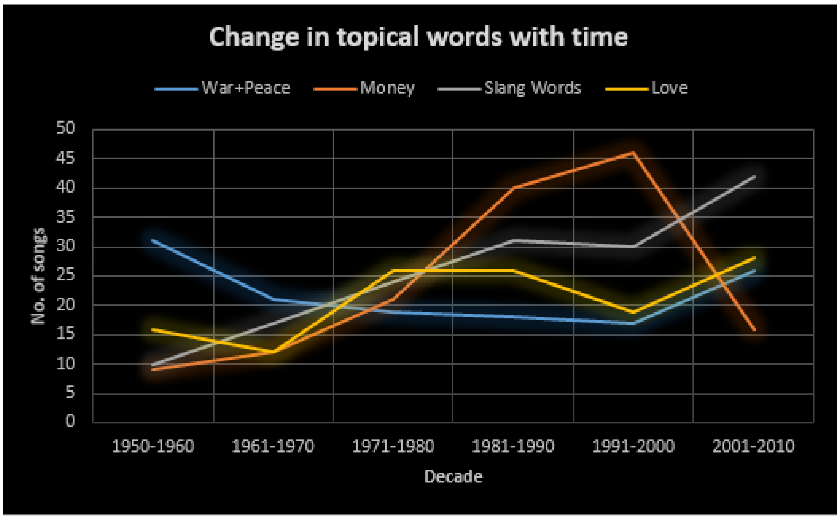


Figure 5.11: Change in topical words with time

6 CHALLENGES AND ISSUES

Some of the major challenges we faced during the project are as follows :
1. Data collection was the biggest challenge at the start of the project.
2. Data limitation was one of the issues, since having larger dataset could have given us more accurate results.
3. Maintaining the quality of dataset.
4. Selecting attributes that may affect the accuracy of the classifier for creating feature vector.

7 FUTURE WORK

We tried out considerable number of features but if we had more time, we would have liked to analyze the effect of certain tasks :
1. Improve feature vector for "Time Prediction" to get better accuracy.

2. Classify the lyrics of a song into a category using WordNet. Further, use incorporate the meaning into our feature vector.
3. If time permits, we will try to perform sentiment analysis of a song, i.e. categorize it based on the mood it indicates.

8 REFERENCES

PAPERS

1. Bertin-Mahieux, Thierry, et al. "The million song dataset." ISMIR 2011: Proceedings of the 12th International Society for Music Information Retrieval Conference, October 24-28, 2011, Miami, Florida. University of Miami, 2011.

2. Bou-Rabee, Ahmed, Keegan Go, and Karanveer Mohan. "Classifying the Subjective: Determining Genre of Music From Lyrics." (2012).

3. Oudenne, Ashley M., P. A. Swarthmore, and Sarah E. Chasins. "Identifying the Emotional Polarity of Song Lyrics through Natural Language Processing."

4. Liang, Dawen, Haijie Gu, and Brendan O'Connor. "Music Genre Classification with the Million Song Dataset." Machine Learning Department, CMU (2011).

5. Adam Sadovsky,Xing Chen. "Song Genre and Artist Classification via Supervised Learning from Lyrics." (2006).

6. Talupur, Muralidhar, Suman Nath, and Hong Yan. "Classification of music genre." Project Report for 15781 (2001).

7. Turnbull, Douglas, et al. "Towards musical query-by-semantic-description using the cal500 data set." Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2007.

8. Diekroeger, Danny. "Can Song Lyrics Predict Genre?."

9. Silla, C. N., Alessandro L. Koerich, and Celso AA Kaestner. "Feature selection in automatic music genre classification." Multimedia, 2008. ISM 2008. Tenth IEEE International Symposium on. IEEE, 2008.

API'S USED FOR COLLECTING DATASET

10. http://cosmal.ucsd.edu/cal/projects
11. http://api.chartlyrics.com/apiv1.asmx
12. http://labrosa.ee.columbia.edu/millionsong/pages/getting-dataset
13. http://lyrics.wikia.com/api.php

14. http://www.student.dtu.dk/ s093020/dataAnalysisWebsite/
15. http://developer.echonest.com/docs/v4/track.html