

Comp 6781
Music Genre Classification System

Yeghia Koronian (27635001)
Chintan Sarvaiya (27148879)

Description of the program

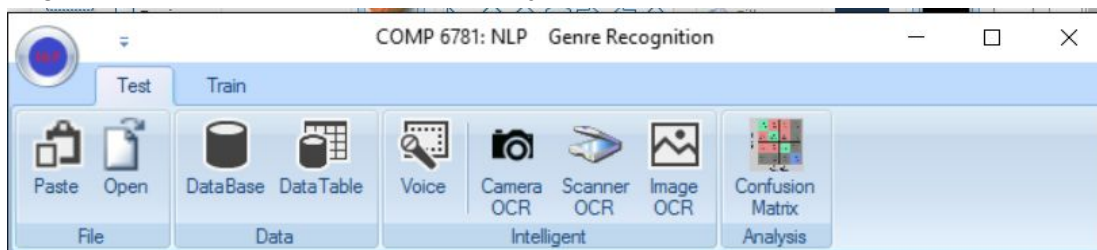
For this project, we used one of the most famous programming language C# for its ease of use syntax and built-in functionalities, and the overall knowledge we have with the language. The code used for the project relied on the following fundamental data structures: arrays, hash tables, data tables. As input, the program reads the lyrics(training) from the database using structured query language(sql) and builds database tables for each genre(output genre) containing the frequency of the word in that genre, these tables will finally serve as determining the output class for each lyric(testing)

Instructions to run the program

After extracting the zip file containing the assignment, you will find a folder named “NLP Genre Recognition” inside this folder there is a file “NLP Genre Recognition” of type “Microsoft Visual Studio Solution” if you have visual studio installed on your computer by double clicking this file it will open automatically all the solution, and if you don’t have visual studio you can download[1] it here, before running the code we need to import the sql database, find the **genrerecognition.SQL** database in the root folder.

Now we are all set to do some testing on the program.

Following is the basic functionalities of our project.



We have two types of datasets : Train and Test

Train Tab :

Database button : will create the training dataset from the database

Test Tab :

Database button : will retrieve data from the testing dataset from the database and run the analysis

Paste button : Paste the lyrics and system will give you the genre of that lyrics

Open button : Open the file containing lyrics, the system will scan the words and give you the genre.

Voice button : You can sing a song ! Based on the words you spoke (sang!) our system will analyze the words and give you the genre. We're currently working on that.

Camera & Scanner OCR button: Scan the lyrics from camera or scanner and our system will analyze the words and give you genre. We're thinking about these in future.

Image OCR button: Open the lyrics-image file and based on OCR technology our system will analyze the words and give you the genre.

Confusion Matrix : Shows the confusion matrix of the testing data

Goal of the work & experiments

The questions that we experimented with:

Should we remove stopwords ?

Stopwords have high frequency as we saw in our first assignment. When we compared our two results with stop-words and without stop-words, it gave us a big difference. The result without stopword were more promising. In our system, we get remarkable change in accuracy by removing stopwords.

Should we remove punctuations ?

Punctuations are used only to facilitate the reader or the listener so they can understand better. While using that for a system or machine learning experience, these punctuations are not useful to predict the result. So, it was better to remove those.

Our Corpora

Our corpora has almost 3000 songs in total. We used a QuickLyric mobile application to get the lyrics. From the mobile application and using reverse engineering, we exported that data into mysql database. It was a tedious task to copy the songs from sqlite (mobile application sql database) to mysql database. Every time we needed to input the songs manually. QuickLyric downloads the lyrics but it doesn't give 100% hit rate on the lyrics. For example, out of 1000 rap songs we were able to download almost 600 songs.

We maintained 70-30% ratio of the dataset. We used 70% of the songs out of all the songs as training data set so that we could have enough information to build our model. 30% of the songs are taken as testing dataset. This 70-30% ratio is applied to all of our genre tables (Rock,

Country, Rap, Reggae, Religion). We followed **2 fold cross validation** on training dataset. So, from our 100% training dataset, we use 50% as training data and 50% as held out data and vice versa

Methodologies and parameters

We used Naive Bayes to achieve the project goals. We used multinomial distribution as our data is discrete.

Formulas :

- Training set:
 - Compute probability of each genre :

$$P(G_i) = \frac{\text{Count of Training Lyrics of Specific Genre}_i}{\text{Count of all Training Lyrics}}$$
 - Compute conditional probability of each word in lyrics of genre :

$$P(w_i | c_j) = \frac{\text{Count of Word}_i \text{ in a Genre}}{\text{Count of all Words in a Genre}}$$
 - Fill the vocabulary using the words above :

$$V = V \cup W_i$$
- Testing set:
 - Get the song lyrics and trim out all the words from it which are not in vocabulary :

$$L_i = L_i \cap V$$
 - Get the genre of trimmed lyrics which will be the genre with highest score:

$$G = \operatorname{argmax}_{G_j} P(G_j) \prod P(w_i | G_j)$$

However there are some words which are not in certain genre, for that we need to apply delta smoothing. From our experiments we found out that $\delta = \mathbf{0.07}$ gives us the best result.

Results and Analysis

Till now we have all our lyrics tokenized with its corresponding probability and ready for analysis. Before jumping into conclusion, we wanted a good measure to check the accuracy of our system. For this reason we used confusion matrix to compute the accuracy of each genre classified and overall accuracy.

	<u>RAP</u>	<u>COUNTRY</u>	<u>RELIGION</u>	<u>ROCK</u>	<u>REGGAE</u>
<u>Rap</u>	96.18	0.64	0.64	2.55	0.00
<u>Country</u>	18.92	42.57	18.24	19.59	0.68
<u>Religion</u>	4.30	1.08	89.25	5.38	0.00
<u>Rock</u>	21.64	11.94	8.21	58.21	0.00
<u>Reggae</u>	24.71	16.47	21.18	12.94	24.71

Overall Accuracy : 62.18 %

As we can see from the matrix above, Rap and Religion have pretty good accuracy. This is not surprising to us, a quick glimpse to the words in the lyrics of these genres will show us that these two genres have discriminating words which help us to achieve high accuracy.

On the other hand Reggae genre has a very poor accuracy as we can see it has been labeled almost evenly among all the genres, this could only mean two things:

- Reggae does not have discriminating words.
- Reggae genre is not recognized from its words but rather from its instrumental acoustics.

Just to test this theory, we read 10 reggae songs (with monotonical voice) to 3 individuals and they were only able to identify its genre 10% of the time (mostly it was identified as rap)

Rock has 58.21% accuracy however it has been labeled 21.64% as rap and 11.94% as country. After looking the words in our database, we saw that in heavy metal (part of rock) the words were usually curse words, which was also very common in rap, and in soft rock the words were common with country. This could explain why we did not achieve very high accuracy in rock.

We were not satisfied with overall accuracy of 62.18 % , thus we wondered if we have applied the bayesian classifier in a wrong manner or this classifier is not well suited for music genre classifier. To answer these questions, first we compared our accuracy with the accuracy obtained using audio as input (no instrumental acoustics) and applying bayesian classifier, second we applied our same data set on different classifiers and we compared the accuracy.

Analysis 1: Audio Input vs Text Input

First of all, in using all of the audio features presented above with a Bayes Net learner, we were able to achieve 61.125% accuracy in overall classification, The confusion matrix for this method is shown below:

		Classified As							
Correct Genre		Blues	Country	Disco	Hip Hop	Metal	Pop	Reggae	Rock
	Blues	63	12	8	0	0	4	4	9
	Country	6	73	8	0	1	0	2	10
	Disco	0	9	60	7	3	4	7	10
	Hip Hop	1	0	6	67	6	7	12	1
	Metal	0	0	0	0	85	0	1	14
	Pop	1	8	4	9	2	65	4	7
	Reggae	2	7	12	7	3	6	57	6
	Rock	5	17	7	1	13	2	7	48

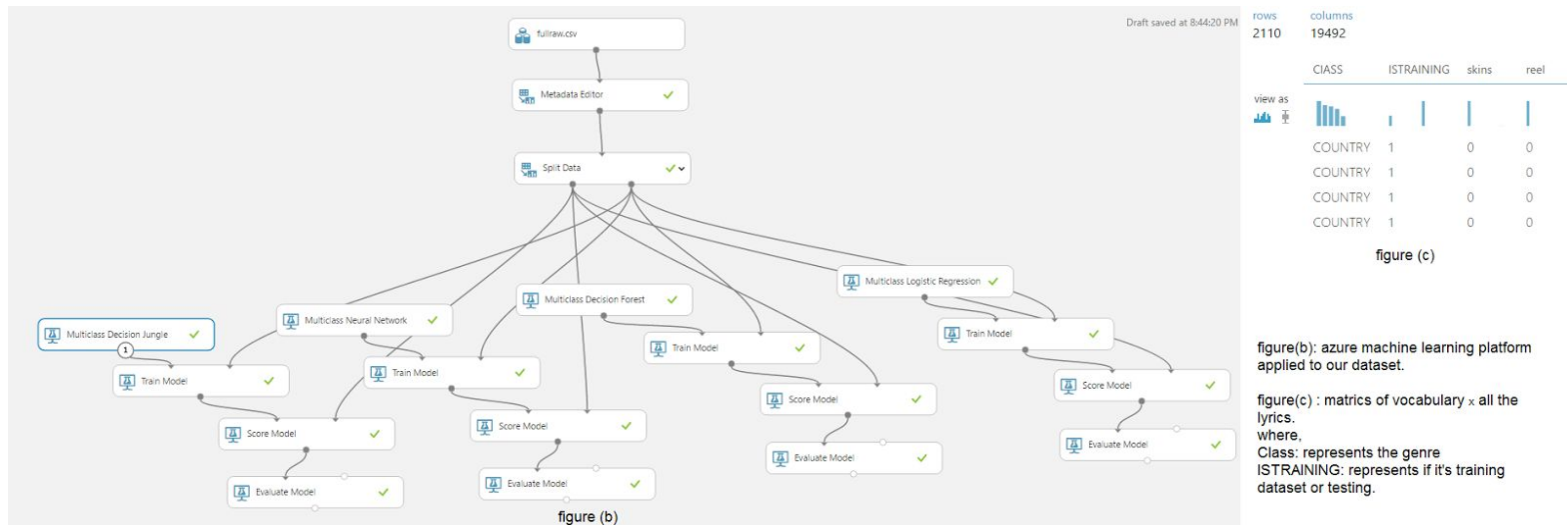
Confusion Matrix

By comparing the overall accuracy of our system and the system above we could see that, our system outperforms by 0.9% which is negligible difference. Thus we could conclude that we did program our classifier in the good way, however it's important to note that the experiment done above has been done using 8 genres which gives the system more room to make wrong predictions. Another interesting thing is to pinpoint that the above system used audio, we don't have enough information to know if the speech to text conversion has been done at high accuracy which lead to 61.125% accuracy

Analysis 2

In this section we perform music genre classification using the same dataset but different classifiers to see which classifier is the best suited for this application.

To do this task we used azure machine learning platform (we have invited you to collaborate on this experiment by email)



Few Notes about figure(b) we used our vocabulary to construct a table where the words from the vocabulary are our features, and the rows are the lyrics of the songs, column 1 represents the actual class and column 2 represent whether the row is for training or testing, the other columns will have frequencies as weights.

MultiClass Decision Jungle classifier

Decision jungles are a recent extension to decision forests. A decision jungle consists of an ensemble of decision directed acyclic graphs (DAGs).

Decision jungles have the following advantages:

- By allowing tree branches to merge, a decision DAG typically has a lower memory footprint and a better generalization performance than a decision tree, albeit at the cost of a somewhat higher training time.
- Decision jungles are non-parametric models, which can represent non-linear decision boundaries.
- They perform integrated feature selection and classification and are resilient in the presence of noisy features.



Only when changing the depth of the decision jungle classifier to 32(previously 16) the system surpassed with the accuracy of genre rock with respect to our system, but still the overall accuracy 51.30% is still lower than the naive bayes classifier, decision jungle works on the logic of decision tree having said that, an explanation of this result could be the words are not very discriminating in country and reggae.

Conclusion: Naive Bayes classifier outperforms decision jungle

Multiclass Neural Network

Experiment created on 4/7/2016 > Evaluate Model > Evaluation results

Metrics

Overall accuracy	0.545603
Average accuracy	0.818241
Micro-averaged precision	0.545603
Macro-averaged precision	0.577753
Micro-averaged recall	0.545603
Macro-averaged recall	0.52862

Multiclass Neural Network

Confusion Matrix

		Predicted Class				
		COUNTRY	RAP	REGGAE	RELIGION	ROCK
Actual Class	COUNTRY	18.9%	2.0%	6.8%	9.5%	62.8%
	RAP	1.9%	83.9%	1.9%	1.3%	11.0%
	REGGAE	9.5%	1.2%	32.1%	15.5%	41.7%
	RELIGION	1.1%		5.4%	57.0%	36.6%
	ROCK	10.4%	2.2%	6.7%	8.2%	72.4%

Properties Project

Multiclass Neural Network

Create trainer mode

Single Parameter

Hidden layer specification

Fully-connected case

Number of hidden nodes

100

The learning rate

0.1

Number of learning iter...

100

The initial learning weig...

0.1

The momentum

0

The type of normalizer

Min-Max normalizer

☒ Shuffle examples

Random number seed

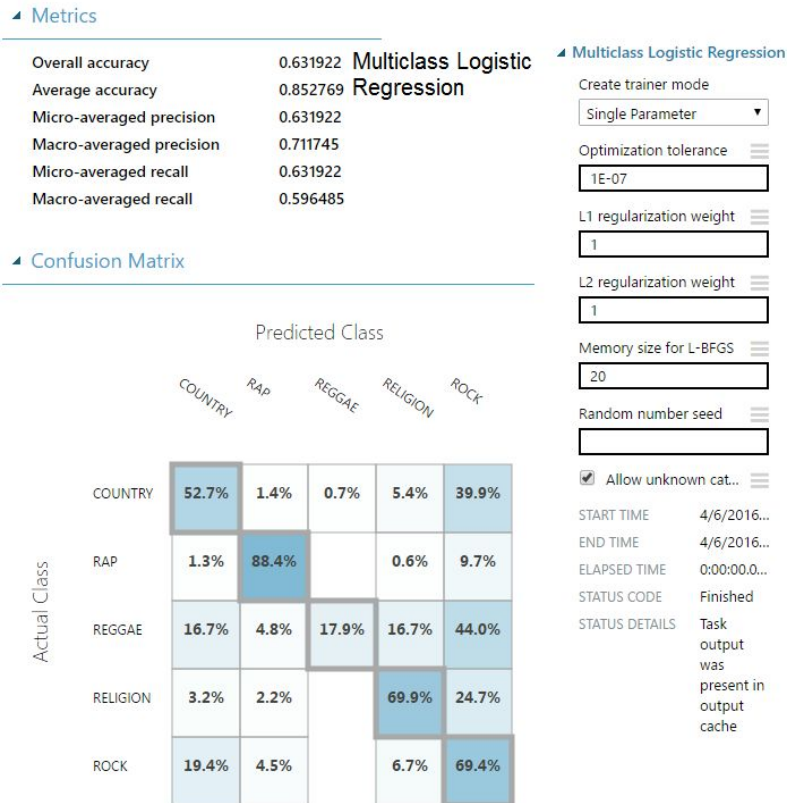
☒ Allow unknown cat...

We setup a neural network with 100 hidden nodes, 0.1 learning rate, 0.1 initial weight and we perform 100 epochs . However we did not get the results we were expecting for. Neural network is one of the most robust and accurate classifier but as it shows from the results it does not classify music genre from lyrics very good. One can argue that the initial weights might have hindered the results but doing 100 epochs would be enough to adjust the weights.

Conclusion: Naive Bayes classifier outperforms neural network by a whopping 10%

Multiclass Logistic Regression

Logistic regression is a well-known method in statistics that is used to predict the probability of an outcome, and is particularly popular for classification tasks. The algorithm predicts the probability of occurrence of an event by fitting data to a logistic function. For details about this implementation, in multi class logistic regression, the classifier can be used to predict multiple outcomes.



Among all other models, this model has given the highest accuracy. According to the analysis between our system and this model, we are achieving near accuracy. From the matrices, reggae has the lowest accuracy here and rap has the highest one which is similar to our results. Reason logistic regression outperformed all the given classifiers above because just like naive classifier it considers all the features(words) are independent, and it calculates probability of individual outcome.

Conclusion: logistic regression outperforms all the given classifiers mentioned above when it comes to music genre classification.

Final conclusion: Our naive bayes classifier has been programmed and the parameters are set properly since all the other classifiers give similar results, furthermore naive bayes is a good classifier for this application.

References and Citations:

<https://msdn.microsoft.com/en-us/library/azure/dn905963.aspx>

<https://msdn.microsoft.com/en-us/library/azure/dn905853.aspx>

<https://en.wikipedia.org/wiki>

<https://studio.azureml.net>

<http://josh-jacobson.github.io/genre-classification/>

<http://go.microsoft.com/fwlink/?LinkId=403675>

<http://go.microsoft.com/fwlink/?LinkId=403677>

[1] <https://www.visualstudio.com/en-us/downloads/download-visual-studio-vs.aspx>

<http://josh-jacobson.github.io/genre-classification/>

"I certify that this submission is my original work and meets the Faculty's Expectations of Originality"

Yeghia Koronian (27635001)

Chintan Sarvaiya (27148879)