Using Hip-Hop Lyrics for Artist Identification: Case Studies on Watch The Throne, Blackstar, and What A Time To Be Alive

Nicholas Abad
Lancaster University
n.abad@lancs.ac.uk

Abstract

Within the field of authorship analysis, many studies have previously implemented natural language processing and machine learning techniques when trying to decide the author of a specific song. However, within this study, a new approach is introduced in which three classifiers are all trained on the solo discographies of two individual artists. With these trained classifiers, their performances are then tested on the verses of these two artists' joint album with an objective to predict which artist sang a specific verse. Within this study, the accuracy, precision, and recall measures are all recorded in addition to the performances of the three classifiers. For these case studies, three joint hip-hop albums, *Watch The Throne*, *Blackstar*, and *What A Time To Be Alive*, each of which were co-created by two artists with vast solo discographies, are then tested.

1 Introduction

Within the topic of information retrieval, stylometry, which can be defined as the statistical analysis in literary styles between different authors, has played an integral roll in a number of different disciplines, ranging anywhere from the academic research domain to cyber security. Recent studies into stylometry as a whole, however, have focused solely on the authorship of text documents, which is done through the collection of an author's previous works, the creation of important features of the words within these works, and the use of different statistical methods to output a prediction as to whether that text belongs to the author in question. Although authorship analysis techniques can be implemented within a number of different literary works, one interesting implementation of these methods is within the musical domain, specifically when trying to decide the author of a specific song or verse through the use of only lyrics.

Despite there being much research within music information retrieval in general, this experiment sets out to determine whether the author of specific verses within a joint album by two artists can be easily identifiable using a combination of different statistical methods. By training these chosen classifiers on the lyrics within the solo discographies of two different artists, these classifiers will then be given song verses within a joint album of these two artists to see how well they could predict the author of that verse, which can be generally categorised as a binary classification problem. Throughout this experiment, several metrics, such as accuracy, precision, and recall, will thus be recorded to see how well these classifiers perform together as an ensemble.

To give a general sense of the layout of this paper, *Section 2: Related Works* gives a brief overview of several key previous works in relation to this experiment while *Section 3: Methodology* describes the natural language processing and machine learning methods that will be used throughout the analysis. Additionally, *Section 4: Data Collection and Preprocessing* and *Section 5: Tokenisation* will detail the steps and reasoning behind the collection, preprocessing, and tokenisation of the musical data/lyrics while *Section 6: Results and Findings* will give an in depth analysis as to what was found and what inferences could be made. Finally, in *Section 7: Conclusion*, final remarks and future works are specified in an attempt to showcase what more can be done in the future.

2 Related Works

Before delving into the actual workings of the experiment, it is important to note previous studies that have went into great detail regarding the binary classification of artists based on their lyrics. Of these works, one of the most influential was written by Sebastiani (2002) in which support vector machines, neural networks, logistic regression, and Naive Bayes Theorem were all used as classifiers when trying to determine the artist of a song on a multi-class level. Within this highly successful study, Sebastiani used these classification techniques to discover that neural networks performed well in the context of his experiment but this single method did not outperform that of support vector machines or regression-based methods, both of which performed best and similar to one another.

Echoing Sebastiani's results regarding the strength of neural networks, support vector machines, and regression-based methods was also a research experiment conducted by Whitman et al. (2001) in which it was found that when given the same problem of artist identification using song lyrics, support vector machines returned a better accuracy measure despite their neural network performing well in their own right with an accuracy of 85%. According to Whitman et al., they claimed that a single support vector machine worked best because they generalise well within a binary case but do not quite work as well in a multi-class (3 or more labels) setting.

Continuing on the topic of music information retrieval, a study conducted by Li and Ogihara (2006) computed the term frequency - inverse document frequency measure, which is further discussed in *Section3.3*, for each word and thus selected the top 200 words to use as their features, also noting that stemming operations were not applied. After conducting the experiment, it was then found that after splitting their songs into three separate groups, the accuracy, precision, and recall measures were as follows:

	Accuracy	**Precision**	**Recall**
Group 1	.507	.500	.384
Group 2	.507	.382	.464
Group 3	.644	.541	.617
Average	.553	.474	.488

Table 1: *Performance metrics that will further be used as the experiment's baseline.*

These three previous works, in particular, are of great importance because this study will be using some of the aforementioned statistical methods in an attempt to match or outperform the results of *Table 1*.

3 Methodology

3.1 Experimental Procedure

Through the inspiration of these previous works, the main objective of this study will be to use an ensemble of three classifiers, which have been chosen to be support vector machines, random forests, and logistic regression, in order to try to predict the singer of a particular verse within a joint album after training these classifiers on songs within both artists' individual song lyrics.

To exemplify this further, consider the hip-hop album *Watch The Throne*, which is a joint studio album released in 2011 by rappers Jay-Z and Kanye West. In order to train the three aforementioned classifiers, individual song lyrics were scraped using methods found in *Section 4: Data Collection and Preprocessing* on both of these artists' individual albums (i.e. Kanye West's *College Dropout*, Kanye West's *Graduation*, etc., Jay-Z's *Reasonable Doubt*, Jay-Z's *The Black Album*, etc.). With these three classifiers properly trained on Kanye West's and Jay-Z's individual albums, the testing set will then be comprised of verses within the songs of their joint album *Watch The Throne*. Because each verse is sung by either Kanye West or Jay-Z, the goal of the ensemble would then be to accurately predict the correct rapper of that verse.

For each of these verses, the logistic regression classifier, the support vector machine classifier, and the random forest classifiers will all independently predict who a verse belongs to, which will then result in three predictions of that verse. By using a simple unweighted voting metric in an ensemble, the final prediction will thus be the artist that has the majority of the predictions. Going back to the *Watch the Throne* example, if logistic regression predicts that a particular verse belongs to Kanye West, while both the random forest and the support vector machine predict Jay-Z, the final prediction will then be that the verse belongs to Jay-Z due to the two classifiers outnumbering the single logistic regression classifier.

Using the same methods described above, the two other joint albums that this experiment will be

testing will be *Blackstar*, which is a hip-hop album created by rappers Mos Def and Talib Kweli, and *What A Time To Be Alive*, which is a hip-hop album created by the rapper Future and the singer Drake.

Wanting to stay within the hip-hop genre, these three joint albums were chosen due to not only their popularity within the genre itself but also due to the plethora of individual albums that each of these six artists have created before and after their joint albums have been released, specifically since classifiers typically perform better when given more data.

3.2 Metrics

In order to compare how well the ensemble does in comparison to *Table 1*, the accuracy, precision, and recall scores will be calculated for each of the three joint albums. In particular, these metrics can be computed by the following equations:

$$Accuracy = \frac{TP + FP}{TP + FP + TN + FN} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

in which TP, FP, TN, and FN respectively represent a true positive, a false positive, a true negative, and a false negative value.

3.3 Term Frequency-Inverse Document Frequency (TF-IDF)

Rather than using the simple bag-of-words metric, the term frequency-inverse document frequency, or tf-idf for short, will be used within this study to give each word a corresponding weight or score. Firstly, this method measures how many times a particular word appears in a document, which subsequently describes the "term frequency" portion of the algorithm name. Secondly, frequent terms, such as "the", "and", and "of", are negatively weighed particularly since they appear a countless amount of times but have little importance within a document. On the other hand, however, rare words are scaled up for the converse reason, which is because of their importance. In particular, the tf-idf score can be computed by the following equation:

$$W_{i,j} = TF_{i,j} \times log(\frac{N}{df_i}) \quad (4)$$

in which $W_{i,j}$ represents the weight of word *i* in document *j*, $TF_{i,j}$ represents the term frequency of word *i* in document *j*, N represents the total number of documents, and df_i represents the number of documents containing the word *i*.

4 Data Collection and Preprocessing

In order to collect the necessary song lyrics for a given artist, a data scraper was created from scratch and was used on *www.MetroLyrics.com*, which is a widely-used and extensive lyrics database for not only hip-hop music but for lyrics within all genres as well. When using this website, a majority of an artists' lyrics and songs were readily available to scrape but it should be noted that for some songs, lyrics were not available and/or were not yet uploaded. In the case of the latter, these songs were not present within the database so no action needed to be done but in the case of the former, the text file that was created had to be manually deleted due to it containing no information. To exemplify a link with no information, a picture of a link in which the lyrics are not available can be found in *Figure 1*.

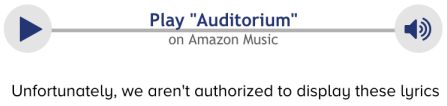

Figure 1: *Non-working link to a particular song*

Despite this inconvenience and as previously noted, this was only the case for a small fraction of songs for each individual artist and was not prevalent for any songs within these artists' joint albums.

Additionally, because the joint album songs appeared as songs on both of the individual artists' lyrics page, the joint album songs also had to be manually deleted as well so that the classifiers will not be trained on parts of their testing set. For the three case studies that this experiment will study, overall statistics of the training sets could be found in the tables below:

Table 2: *Case 1 - Watch the Throne*

Case 1	**Training Songs**	**Testing Verses**
Jay-Z:	356	17
Kanye West:	333	12
Total:	699	29

Table 3: *Case 2 - Blackstar*

Case 2	**Training Songs**	**Training Verses**
Mos Def:	120	11
Talib Kweli:	149	12
Total:	169	23

Table 4: *Case 3 - What A Time To Be Alive*

Case 3	**Training Songs**	**Testing Verses**
Drake:	373	13
Future:	275	11
Total:	548	24

5 Tokenisation

Once all of the music data was collected using the web scraper, the next step was to then tokenise the lyrics in a way that seemed the most appropriate. After much thought and consideration, a basic whitespace tokeniser was chosen mainly due to how it deals with contractions, which are commonly used in music and, in particular, within hip-hop quite frequently. Unlike the TreeBankWordTokeniser or the PunktWordTokeniser for example, the whitespace tokeniser keeps contractions together rather than splitting them into several individual tokens, which can be beneficial in this specific case. However, it should be kept in mind that the main drawback when choosing this tokeniser was that because the lyrics itself on *www.MetroLyrics.com* were not all seemingly written by the same person, there is a lot of variability in terms of punctuation usage. Therefore, since the tokeniser splits on a whitespace, the word ”don’t” and ”don’t,” for example, would not be considered to be the same token, which has the potential to alter the results.

6 Results and Findings

For each of the three joint albums, the accuracy, precision, and recall measures could all be found in *Table 5* while the accuracy measures for each of the individual classifiers could be found in *Table 6*, both of which are located below. It should be noted that within these tables, **Case 1** denotes the the joint album *Watch The Throne*, **Case 2** denotes the joint album *Blackstar*, and **Case 3** denotes the joint album *What A Time To Be Alive*.

Table 5: *Joint Album Final Metrics*

	Accuracy	**Precision**	**Recall**
Case 1	0.72413	0.73557	0.74019
Case 2	0.69565	0.70833	0.68939
Case 3	0.83333	0.83217	0.83217
Average	0.75104	0.75869	0.75391

Through these results, it is easy to conclude that the ensemble method worked extremely well by producing an average accuracy rate of 0.75104 across the three albums. Upon inspection of the results of each of the cases, it can be easily observed that **Case 2: Blackstar** performs the worst in each of the three metrics, which can be thought to be understandable given that the classifiers in this case had significantly less training data to work on. With only 169 available songs in the corpus in comparison to the 699 and the 548 of its counterparts, this result seems justifiable and within reason.

On the other hand, however, despite **Case 3: What A Time To Be Alive** having 100 less songs of training data in comparison to **Case 1: Watch The Throne**, it seemed a bit odd that the metrics were all higher in favor of the latter. Thinking that this may have to do with the amount of lyrics within songs rather than the raw song count in the training data, the combined file sizes of each of these artists’ were then investigated but did not quite prove anything. With an initial hope that **Case 3: What A Time To Be Alive** had more data in terms of bytes in comparison to **Case 1: Watch The Throne** despite having less songs, this idea was soon discredited once the combined file size of Case 1 was found to be roughly $1,929,000$ bytes while the combined file size of Case 3 was found to be roughly $1,679,000$ bytes.

Because of this, coming to the conclusion that training size significantly effects the performance of these classifiers cannot be concluded due to not being able to generalise this in both cases. Despite this being the case however, this experiment’s approach through the use of a data scraper, Whitespace tokenisation, and an ensemble method far out-performed the initial benchmark created in Li and Ogihara (2006), which can also be found earlier in *Table 1*.

Table 6: *Individual Classifier Final Results*

	LR Accuracy	SVM Accuracy	RF Accuracy
Case 1	0.72413	0.68965	0.51724
Case 2	0.60869	0.69567	0.69565
Case 3	0.83333	0.75	0.75
Average	0.75104	0.75869	0.75391

Secondly, when analysing the results in *Table 6*, it is important to take note of the performance of the support vector machine classifier in comparison to the logistic regression classifier as well as the random forest classifier. According to multiple previous works, it was commonly stated that within their experiments, the performance of the SVM always exceeded that of all other classifiers, except for a neural network which was not implemented within this project. However, in this specific case, the SVM, on average, performed the best but not significantly better with only outperforming logistic regression by 0.00765 and a random forest by 0.00478, which contradicts what was previously found. In a general sense, however, there did not seem to be a consistent pattern with the chosen classifiers.

7 Conclusion and Future Works

In conclusion, it was found that by training three classifiers on the artist's individual albums and proceeding to test the performance of these classifiers on verses within their joint albums, the ensemble approach works exceptionally well, resulting in an average accuracy rate of 0.75104, an average precision rate of 0.75869, and an average recall rate of 0.75391. These findings suggest that in general, hip-hop artists seemingly use the same linguistic vocabulary throughout their careers, which is exemplified by the apparent success of this study. Despite this success however, results were not very conclusive in terms of the optimal individual classifier, with no classifier consistently exceeding the performance of the others.

In order to further continue this study, it would be beneficial to try to implement a neural network that has the same or similar architecture to that of previous works, particularly to investigate the performance it has in comparison to the performance of the support vector machine classifier. Due to not being able to find this architecture when conducting a literature review, this classifier was purposely left out. Additionally, it would be interesting to test other joint albums to see if this could be generalised to not only other hip-hop artists and albums but to other musical genres as well. As a final recommendation of potential future works, conducting this study on a multi-class (3+ labels) dataset may interesting to investigate as well.

References

Tao Li and Mitsunori Ogihara. 2006. Toward intelligent music information retrieval. *IEEE Transactions on Multimedia* 8(3):564–574.

Fabrizio Sebastiani. 2002. Machine learning in automated text categorization. *ACM computing surveys (CSUR)* 34(1):1–47.

Brian Whitman, Gary Flake, and Steve Lawrence. 2001. Artist detection in music with minnowmatch. In *Neural Networks for Signal Processing XI, 2001. Proceedings of the 2001 IEEE Signal Processing Society Workshop*. IEEE, pages 559–568.