university of groningen / faculty of arts

An Explorative Study into the Field of Lyrics-based Author Classification.

Tomer Gabay

Master thesis
Information Science
Tomer Gabay
s2726769
May 5, 2020

ABSTRACT

This paper tackles the barely researched field of Lyrics-based Author Attribution (LbAA) and the unresearched fields of Lyrics-based Gender Classification and verse classification in collaborative songs. Classical author attribution features (word and character n-grams), discriminative music genre classification features (POS tags, text statistics) and less conventional features (phonetic and abstract representations) are all applied to LbAA in this study. Subsequently, these features are discussed to get an insight into how this research field is similar and dissimilar from related research fields. Using an iterative and explorative method, combined with Support Vector Machines and analysis of its most informative features, new insights on Lyrics-based Author Classification has been acquired. For the first time the reliability of punctuation in lyrics is questioned, as a small annotation research and its influence on the classification scores raised doubts about the lack of interpretability of punctuation when transcribing lyrics. In contrast to other LbAA papers, (nick)names of artists are deliberately removed using blacklisting, in order to increase the difficulty and it results in the appearance of new informative lexical features, such as the lyrical-related stop words *yeah* and *uh*.
An alternative method to the previously successful method of adding multi-artist songs to the data to increase the scores is invented, and turns out to be more effective: add only the relevant verses of multi-artist songs.
Using grid search techniques for parameters and different sets of the previously mentioned features, optimal models are constructed, which make use of more than just word and character n-grams. These optimal models, based on Support Vector Machines and expressed in Macro-F1 scores, scored respectively 0.91 and 0.79 on two eight-class LbAA data sets, 0.84 on gender classification, and respectively an average of 0.79 and 0.90 on two collaborative album data sets.

CONTENTS

PREFACE

Now, at the age of 23, my period as a student has come to an end. I have enjoyed my period as a student enormously. Many new friendships have been made, and not unimportant, I have gained a lot of knowledge and acquired numerous amounts of skills which I didn't have before, such as programming, usability design and writing papers at a scientific level. Throughout my entire bachelor and master Information Science at the Rijksuniveriteit Groningen I have been a happy man. Of course, some obstacles like having to be able to program in Prolog had to be overcome. In the end, with the aid and support of my family, friends, fellow students and the professors, everything has worked out more than fine, with relative ease, and so I'd like to thank each and every one of them for there support, advice and guidance.
Besides my time as a student at the RUG, I also have enjoyed and learned a lot of my part-time job as a teacher assistant in the Python related programming courses, and I cannot promise that I won't return for a PhD, if I'll be allowed to of course ;).
My most enjoyed course, both for the bachelor and the master, in contrast to many students I presume, is writing the thesis. Herein you can combine all skills you have acquired, and due to the large amount of freedom in subject choice, you can apply them to your favourite subject. In my particular case, this means applying Machine-Learning to politics for my bachelor, and to music for my master. Therefore, I can honestly say that I enjoyed writing this thesis and I feel proud of its result. Obviously, my special thanks goes out to Malvina Nissim, who did not only guide me throughout my master thesis, but also throughout my internship and lectured, in my opinion, the most interesting course of the master: Learning from Data.
I hope that everyone who reads this thesis enjoys it, and that my research truly contributes to the research field of Lyrics-based Author Classification, and that it inspires others to study this research field as well.

1 INTRODUCTION

Music is not just a form of leisure or entertainment, it is central to the very formation and reproduction of human societies (Shepherd and Wicke, 1997). Every culture on earth produces music. The total worth of the music industry was $19.1 billion in 2018 (McIntyre, 2019), rising by almost 10% compared to 2017. Hence, music is both is essential for human societies as well as commercially viable. Of course, such an important phenomenon sparks the interest of scientists too. Music Information Retrieval (henceforth called MIR) is a very broad research area, with sub domains such as classic similarity retrieval, genre classification, visualization of music collections, and user interfaces for accessing audio collections (Mayer et al., 2008). Unlike in prose[1], where Author Attribution is a widely researched field, only a few studies cover Author Attribution in MIR.
Author Attribution is the process of determining the writer of a document (Bozkurt et al., 2007). What the document is varies per research, from blogs and Tweets (Overdorf and Greenstadt, 2016) to articles and essays (Mosteller and Wallace, 1964). If we apply Bozkurt et al.'s definition to lyrics, we will get: Lyrics-based Author Attribution in the process of determining the author of the lyrics of a song. The challenge here lays in the fact that often the artist of a song is not the author of the lyrics of that song, and who the actual writer of that lyrics is, is often unknown. This makes it rather difficult to collect data to train on. To avoid this problem, Mara (2014) only focuses on rap artists, which are known to mostly write their own lyrics. Also, rap is a lyrics-focused music genre (Fell and Sporleder, 2014), and thus well suited for Natural Language Processing tasks, such as Author Attribution.
Hence, this study will follow Michael Mara's decision in focusing on the rap genre. Mara (2014) did an exploratory research in Author Attribution of lyrics on several rap artist data sets of different sizes and compositions, utilizing Naive Bayes and Support Vector Machines. In a way, the present research will be a comprehensive and improved version of Mara's exploratory research in 2014, with numerous new features explored, techniques applied and lots of new findings (see Section 1.3).

1.1 MOTIVATION

Lyrics-based Author classification is a barely researched field. Still, there is strong empirical evidence that it is worthwhile to use lyrical properties for analysing and classifying music (Fell and Sporleder, 2014), because classifiers that use both lyrics and audio outperform audio-only classifiers on most classification tasks (Mayer and Rauber, 2011). Potential use for such classifiers include style similarity analysis and ghostwriting detection, in which artists secretly do not write their own lyrics.

[1] a form of language that exhibits a natural flow of speech and grammatical structure, thus 'normal' language

1.2 RESEARCH QUESTIONS

This section covers the main research questions with some elaboration on each question.

1. *Which features are the most important for Lyrics-based Author Attribution?*
   This research question is suggested by Mara (2014) as future work. His own paper is not extensive in this matter. It only uses a bag-of-words model and Part of Speech bigrams as features. Hirjee and Brown (2010) already proved that rappers can be distinguished from each other using advanced rhyme detection algorithms, and due to the high complexity of rhyme and advanced required knowledge in phonology, such a rhyme-based model is not replicated in the present paper. Instead, features based on Author Attribution and music genre classification are applied.

2. *Are the most important features for Author Attribution on prose also the most important features for Author Attribution on lyrics?*
   E.g. Stamatatos (2009) gives a nice overview of important features on Author Attribution on prose, such as word and character n-grams. Whether these features are also important to LbAA has not been researched yet.

3. *Are the most important features for music genre classification also the most important for LbAA?*
   Both Mayer et al. (2008) and Fell and Sporleder (2014) go detailed into which features are important to determine the music genre based on lyrics, which extend beyond the usual Author Attribution features on prose. Especially text statistics are very informative to distinguish different music genres.

4. *Are artists of the same gender or ethnicity harder to distinguish than artists of different gender or ethnicity?*
   In papers like Miller et al. (2012) it is proven that a person's writing can reveal his/her gender, while Pennacchiotti and Popescu (2011) argue that linguistic characteristics can help an algorithm to distinguish Afro-Americans from non-Afro-Americans. Such research falls into the Author Profiling field, in which there's an attempt to determine to which classes an author belongs, such as age, gender or native language (Santosh et al., 2013). In the present paper, a lyrics-based gender classification task is implemented, the first in this field. Also, confusion matrices are interpreted to determine whether the lyrics of two female artists are more easily confused than that of a man and a female, and whether the lyrics of two Afro-American artists are more easily confused than that of a Caucasian and Afro-American artist.

5. *How accurately can a classifier determine the artist of a verse in a multi-artist song?*
   Instead of classifying at song level, verse level classification[2] is applied for this research question. Test data instances will be shorter, which usually increases the classification difficulty. But perhaps as important, the verses come from the same song, and thus, the same topic is expected to be addressed. Also, co-writing is more likely to have occurred. Therefore this task is expected to be significantly harder than solo-artist song-based classification.

[2] Though technically incorrect, all lyrics sections are referred to as verses in this paper for generalisation purposes, including hooks and choruses.

1.3 CONTRIBUTIONS

In this work, a comprehensive feature exploration on LbAA is presented, which include phonetic and abstract feature representations. For the first time, artists names and nicknames are deliberately removed from the lyrics to make the task more interesting and challenging. Also, this paper aims to raise a discussion on the reliability of punctuation in lyrics, which has not got any attention yet. The research field of gender classification is as of now expanded to song texts, while the influence of gender and ethnicity on LbAA is also examined for the first time. At last, this study is the first to apply author attribution to verses in multi-artist songs.

1.4 READER'S GUIDE

The next chapter is Related Work (2), in which the fields of Author Attribution and Author Profiling on prose are examined and compared to the fields of LbAA and music genre classification.
In chapter Data Collection (3) the used database is explained (3.1), how its API is used with additional song filtering for extra reliability (3.2) and what data sets are assembled (3.3). In Method (4) the baselines and evaluation methods are explained (4.1), necessary preprocessing steps are discussed (4.2), and the most informative lexical n-grams are applied in (4.3) to gather more information about the lyrics, which helps with optimizing parameters and detecting extra preprocessing steps that must be taken. Section 4.4 raises the question whether punctuation in lyrics can be considered reliable, through the conduct of a small research and informative character n-grams analysis. Mara (2014)'s use of extra training songs, despite being noisy through the presence of other artists' lyrics, is discussed in 4.5, and an alternative method is proposed. The lack of value of statistical features on LbAA and the problems stumbled upon in phonetic representations are of subject in respectively 4.6 and 4.7. Section 4.8 covers gender classification and the implementation of van der Goot et al. (2018)'s abstract representations. The task of verse classification in multi-artist songs is elaborated in 4.9, and some side-experiments are mentioned in Section 4.10. In Method's last section the final models are presented (4.11). The results of LbAA, gender classification, and verse classification in multi-artist songs are displayed and discussed in respectively Sections 5.1, 5.2 and 5.3. The conclusions of this paper can be found in 6 and suggested future work is mentioned in Chapter 7.
Please consult the Appendices for extra tables and graphs on the data sets (A), feature analysis (B), and the results (C) when they are mentioned, as they help understanding the text.

2 RELATED WORK

Stamatatos (2009) describes the development of author attribution (AA) throughout the modern history, starting in the 19th century with AA research on plays of Shakespeare. Until 1964, in which Mosteller and Wallace published *Inference and disputed authorship: the Federalist*, statistics were used for AA. Mosteller and Wallace's paper initiated nontraditional authorship attribution studies, as opposed to traditional human-expert-based methods. Computer-assisted stylometry was by far the most dominant field within AA from 1964 until the late 1990s. Stylometry is described in Stolerman et al. (2014) as "a form of authorship attribution that relies on the linguistic information found in a document". After the late 1990s, computer-assisted stylometry turned into computer-based stylometry studies, e.g. Overdorf and Greenstadt (2016) and Stolerman et al. (2014). A cause of this was the rise of internet and thus much more possible applications of AA, for instance on e-mail messages, blogs and social media, because now anyone with an internet connection could spread their own written texts. Through features on e.g. lexical, character, syntactic and semantic level, author's of such texts can be identified (Stamatatos, 2009), with especially function words being very informative (Argamon and Levitan, 2005).

Next to the task of AA, other related author classification tasks started to arise. Due to the sudden possibility of writing anonymously on social media platforms as Twitter[1] and Reddit[2], the field of author profiling has gained mayor interest as well. Author profiling (henceforth AP) is described in Mitkov and Oakes (2019) as "the analysis of people's writing in an attempt to find out which classes they belong to, such as gender, age group or native language". To simplify even further: AA is about determining *who* is the author while AP is about *the characteristics* of the author. One of the most applied AP tasks is gender classification, in which the classifier tries to predict the gender of the writer of instances. Especially character n-grams turn out be very informative to predict someone's gender (Basile et al., 2017; Miller et al., 2012).

To conclude on the relevant areas on prose, there have been many studies on both AA and AP on online posts, ranging from AA tasks on blogs and Reddit comments (Overdorf and Greenstadt, 2016) to AP tasks such as dialect and gender classification on tweets (Rangel et al., 2017).

In the the field of Music Information Retrieval, AA and AP are barely researched. Mara (2014), as mentioned in the previous chapter, applied AA to rap lyrics. Three data sets of different sizes are assembled and used, ranging from respectively 4, to 12, to 348 artists in the data sets. By using a bag-of-words representation and classical Machine-Learning algorithms in the form of Naive Bayes en Support Vector Machines, accuracies of up to 0.877 are achieved on the four artist data set. However, as Hirjee and Brown (2010) noted, rappers are lexically easy to distinguish as they regularly mention their own name in songs. Mara (2014) does not seem to be aware of this. Next to the bag-of-words model, he also applied Part of Speech tags, but they did not aid his classifier.

Hirjee and Brown (2010) also applied AA to rap lyrics, but they constructed a very advanced model to detect rhyming patterns that are capable of identifying a rap artist or group. In their data set of 25 artists, a weighted F1-score of 0.516 was accomplished. Due to the absence of other papers on Lyrics-based Author Attribution, a lot of potential features which are successful in AA on prose have not been explored yet.

[1] https://twitter.com/
[2] https://www.reddit.com/

However, there are some vast difference between AA and AP on prose compared to AA and AP on lyrics. Lyrics contain traits such as rhyme and repetitive structures, and are confined to rhythm. Due to the lack of papers on AA and AP on lyrics, information about lyrics-based classification must be searched elsewhere. Luckily, music genre classification has gotten serious attention. Mayer et al. (2008) most successful model scores an accuracy of 0.335 on ten music genres with the use of Support Vector Machines, combined with rhyme features, Part of Speech tags and text statistics. Interestingly, while the bag-of-words model works very well for Mara (2014)'s LbAA, its success for Mayer et al. (2008)'s music genre classification is limited. Fell and Sporleder (2014) on the other hand, uses more advanced lexical features, such as n-grams combined with an tfidf vectorizer, which achieves an F-scores of 0.49 on eight music genres, still with the use of classical Machine-Learning. Their best model combines lexical features with text statistics and style features, amongst a few other features, scoring 0.53 on the eight genres, outscoring human participants. Not only did they apply classification to lyrics on genre, but also to lyrics ratings and their approximate publication time.
citetsaptsinos2017music is the first to apply a neural network to music genre classification, in the form of Long-Short-Term-Memory (LSTM) and a Hierarchical Attention Network. With accuracies up to 0.46 on a 117 genre data set and 0.50 on a 20 genre data set, their scores are high when taken the amount of classes into account[3].

[3] Some extractions of this chapter originate from Gabay (2019) and Gabay (2020)

3 DATA COLLECTION

Almost all papers related to music genre classification and music author attribution mention the lack of a publicly available lyrics data sets (Mara, 2014; Fell and Sporleder, 2014; Guo and Khamphoune, 2013; Mayer and Rauber, 2011). Therefore, all data sets built (see Section 3.3) in the present paper are downloadable through Github[1].

3.1 GENIUS

Following Mara (2014) and Guo and Khamphoune (2013), the Genius' data base is used through it's API to create the artist data set. Genius[2] was founded in 2009 as a crowd-sourced hip-hop website named Rap Exegesis. After four months its name was changed to Rap Genius, as exegesis was too difficult for users to spell (Wiedeman, 2014). In July 2014 Rap Genius was renamed to simply Genius, as it wanted to extend their content beyond just rap music, into the fields of e.g. pop music, R&B and literature (Carlson, 2014). Thus, references to Rap Exegesis or Rap Genius in previous literature refer to what is now Genius. In its essential, Genius is a crowd-sourced lyrics website, on which users[3] can add lyrics and annotate them to explain the meaning of a certain word, line or verse. To increase the quality of the lyrics and annotations three strategies have been put into place:

- **Extensive guidelines**: Genius provides extensive guidelines on how to write lyrics, partly for generalization purposes. For example, contributors should always write *okay* rather than *O.K.* or *ok*. Other rules include rules on when to write numbers in their digit-representation and when in their letter-representation, and how to use punctuation. The complete guideline can be found on: https://genius.com/Genius-how-to-add-songs-to-genius-annotated.

- **Genius IQ**: Genius IQ is a point based rewarding system. A user's Genius IQ determines their possible actions on the website. For example, a Genius IQ of 100 is necessary in order to be allowed to write the lyrics of a new song. IQ can be earned by adding e.g. annotations to a line of an already existing song lyrics on Genius. If an annotation is being 'upvoted' by other users, the writer of the annotations earns IQ for each vote. The higher the IQ of the voter, the higher the impact on the earned IQ of the annotator, ranging from two to ten IQ points per vote. On the contrary, if a contributor's annotation receives 'downvotes', points are subtracted from their IQ.

- **Restrictions on certain songs**:

  - Songs which are getting over 5,000 views require extra IQ points to be able to edit its lyrics or to add annotations to it.

  - Songs can be 'locked' by the staff, in which only top Genius contributors or the staff themselves can contribute to the lyrics. Usually this applies to very popular songs.

  - The actual artist can verify lyrics or annotations, after which Genius users won't be able to make any adjustments.

[1] https://github.com/sTomerG/lyrics_based_author_classification_datasets
[2] https://genius.com/
[3] Users who conribute to Genius by adding lyrics and/or annotations are henceforth called contributors.

As in basically all forms of data, crowd-sourced data has advantages and disadvantages in comparison to non-crowd-sourced data. Usually crowd-sourced data is less reliable, but much larger, and thus depending on the task a choice must be made. However, in the case of lyrics, there is hardly ever the original lyrics as written by the author available. In the few cases the author does publish some of their own lyrics in written form, such as in Jay-Z (2010), Genius utilizes these lyrics and annotations. Due to the fact that the amount of views on a lyrics raises the bar of Genius IQ and/or restrictions, popular artists' lyrics are likely to have a higher quality and thus should be preferred for lyrics-based researches.

3.2 SONG FILTERING

Using the *search_artist* function of Genius' API, a list of all instances and their meta data such as album and featuring artists are returned. These instances are mostly songs, but can also include textual explanations of album cover art or interview transcriptions. A simple suggested method, but still quite effective, is to check whether an instance is linked to an album. Non-songs are most often not linked to any album, thus using that meta-data helps to a large extent to filter out the non-song instances.
There are also numerous songs with different versions, such as remixes or live versions. As they are often very similar to the original version, and thus contribute to duplicate lyrics, only one version of each song is used in this paper, conform Tsaptsinos (2017) and Fell and Sporleder (2014). The different versions are filtered out based on song the similarity of song titles.
Despite filtering for different versions and album related songs, there are still undesirable instances in the data set: intros, outros, interludes and skits. On rap albums, intros and outros are often interviews or spoken statements, and thus do not count as lyrics. Interludes and skits are often funny conversations, which were partly used to have more tracks on a CD, which back in the CD era lead to an increased appeal of the CD for potential buyers (Rytlewski, 2012). Fortunately, allmost all intros, outros, interludes and skits do mention its type in the song title, and so songs with these words in the title are filtered out.
Songs can also have multiple artists. In these songs, parts of the lyrics are not by the main artist and thus noise. At first, these songs are filtered out, but they are utilized later on, which is explained in Section 4.5.

3.3 DATA SETS

At the core of this study are four artist data sets. As argued in Section 3.1, the quality of the data is higher on popular artists. All artists used for song classification are used in Mara (2014), and a high proportion of them also in Hirjee and Brown (2010), which both focuses on the most popular rap artists. The exact composition of each data set is displayed in Appendix A.

3.3.1 Diverse Artist Data Set (DADS)

The first data set is the *Diverse Artist Data Set* (DADS). This data set is build out of 1038 songs of eight different artists (see Figure 1). Four Afro-American male, two Afro-American female, and two white American male artists. The choice of for this composition is based on Research Question 4: *Are artists of the same gender or ethnicity harder to distinguish than artists of different gender or ethnicity?* Looking at the confusion matrices shows whether songs are more often confused within their respective artist 'class' or not. With eight artists the amount of classes is similar to

the amount of classes used in other music classification tasks (Fell and Sporleder, 2014; Mara, 2014; Mayer et al., 2008). The choice of the actual artists are based on Mara (2014) and the precondition that there are at least two of each of the three umbrella classes: Afro-American male, white American Male and Afro-American female. Due to the lack of popular Caucasian female rap artists with at least 50 amount of songs, they are not included in this data set.

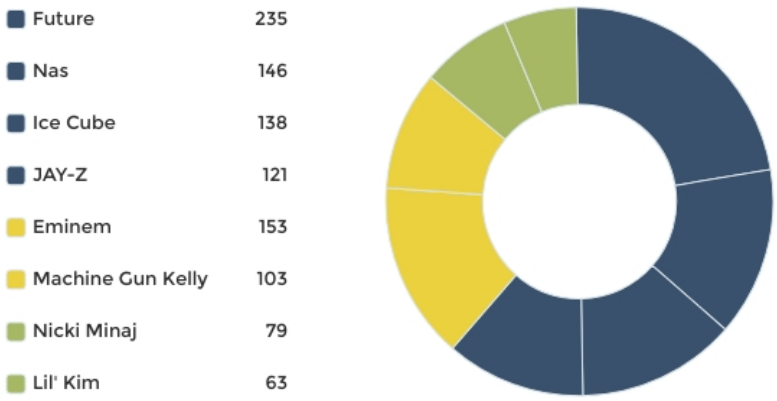

Figure 1: Distribution of DADS. Dark blue represents Afro-American male rappers, yellow represents white American male rappers and green represents Afro-American female rappers.

3.3.2 Afro-American Male Data Set (AAMDS)

The second data set is the Afro-American Male Data Set (AAMDS). In order to have a better indication whether ethnicity and gender influences the classifiers, AAMDS consists of only Afro-American male rappers, which makes it potentially lyrically more homogeneous and thus perhaps a more difficult task to classify. The four Afro-American male artists from DADS are included, plus four more are added: Lil Wayne (338), 2Pac (100), 50 Cent (168) and Snoop Dogg (157).

3.3.3 Gender-based Data Set (GBDS)

The third data set is the Gender-based Data Set (GBDS), to perform gender classification for Research Question 4. This data set is based on seven of the previously mentioned male rappers, and seven female rappers. On top of Nicki Minaj (79) and Lil' Kim (63), Missy Elliot (43), Cardi B (28), Iggy Azalea (49), Queen Latifah (40) and MC Lyte (58) are added. In total there are 363 songs for the women. To prevent any influence from class imbalance a total of 363 songs of the men are extracted, which together make up this data set. For extra balancing, the male artists and female artists are split into pairs, which results in the exact same artist proportions (See Appendix A, Table 13).

3.3.4 Collaborative Album Data Set (CADS)

The fourth data set is the Collaborative Album Data Set (CADS). This data set consists of two sub-data sets, each consisting out of one collaborative album as test data; *Watch the Throne* by JAY-Z and Kanye West, and *Distant Relatives* by Damian Marley and Nas.
Their own individual songs will be used as training data. This data set will be used to answer Research Question 5: *Is it possible to automatically determine the artist of a verse in a multi-artist song?*. Hence, the songs on the collaborative albums will be split into verses, with the author being either of two the artists of that album. The choice to use these collaborative albums as test data is that these artists have worked in close cooperation to create the songs. Therefore the chance of them covering the same subjects in the songs and complementing each other through co-writing is expected to be larger than when two artists collaborate on just one song, and thus the difficulty on collaborative albums is expected to be higher. Another extra layer of difficulty is that in both these sub datasets, the most represented artist in training is the least represented artist in the test data, by large margins (see the tables below).

Artist	Training songs	Added verses	Only verses	Test verses
JAY-Z	121	191	770	11
Kanye West	70	142	471	19
TOTAL	191	333	1241	30

Table 1: Collaborative Album Data Set a. The 'only verses' column includes songs from the training songs split into verses.

Artist	Training songs	Added verses	Only verses	Test verses
Nas	146	68	711	20
Damian Marley	39	19	224	30
TOTAL	185	333	935	50

Table 2: Collaborative Album Data Set b. The 'only verses' column includes songs from the training songs split into verses.

4 METHOD

All programming is done in Python3, using Scikit-learn (Pedregosa et al., 2011) for Machine-Learning. Several classification algorithms such as Naive Bayes, Random-Forest and Logistic Regression have been attempted at different stages. Due to a LinearSVC systematically outperforming these other methods, the focus of this paper is solely on the LinearSVC. All random states are saved with a seed of 50.

4.1 EVALUATION

The baselines are calculated by dividing 1 by the amount of possible classes, following Mara (2014) and Hirjee and Brown (2010). Due to the class imbalance in DADS and AAMDS, evaluations are based on the Macro-F1 score. The Macro-F1 score gives equal weight to every class, without taking the amount of instances per class in account. The baselines, expressed in Macro-F1 are thus 0.125 for DADS and AAMDS (based on eight classes), and 0.5 for GBDS and CADS (based on two classes).
DADS, AAMDs and GBDS are each split into 80% training, 10% development and 10% test data. The optimal feature settings are compiled using grid-searches combined with k-vold cross validation (k=9). The reason to use k-vold rather than just the development data is that the chance of overfitting significantly decreases. K is set to 9 to replicate the train-test ratio in the final data set without consideration of the extra verses; 7.3 training instances for every test instance. The development set is used to determine whether adding extra training verses helps (see Section 4.5).

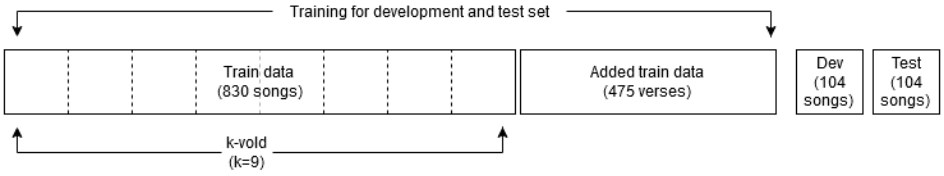

Figure 2: Distribution of train, development and test data in DADS. AAMDS and GBDS are similarly set up.

4.2 PREPROCESSING

Due to the novelty of research on Lyrics-based Author Attribution, preprocessing steps are done in an explorative and experimentative manner with the aid of word n-grams and their displayed informative features (see Section 4.3). This method is applied instead of simply grid-searching to find the optimal lexical n-gram settings, to get a better understanding of what the classifier is learning. By looking at the most informative features of a classifier, conclusions can be made about what the classifier has learned in order to link lyrics to an artist, and extra filtering can be applied if necessary. Due to the hypothesis that a more homogeneous artist data set in harder to classify than a diverse artist set, the preproccesing steps are based on AAMDS. On the first run, without any preprocessing steps, a LinearSVC combined with an tfidf-vectorizer and word unigrams scored a Macro-F1 score of 0.884.

However, Genius' lyrics contains non-lyrical data, such as explained and seen in Figure 4. Other non-lyrical data include notes about the music; **The song changes to: The Doors - Five to One** or *[Instrumental Break]* or *[Produced by Tay Keith]*. Regexes are used to remove these notes from the lyrics. This causes the Macro-F1 score to drop to 0.832.
At this point, all data in the lyrics is actually the lyrics itself. Despite no other lyrics-based classification paper mentioning converting digits into their letter representation (e.g. 50 to fifty), there are two main arguments to apply this conversion:

1. Statistical features such as average word length interpret *1996* as four characters, despite actually consisting out of 18: *nineteen ninety-six*.
2. The character and phonetical similarity between e.g. *eight* and *weight* cannot be detected without converting *8* to *eight*.

Therefore, numbers that are written through digits are converted to words with the use of the inflect library[1]. With the use of regexes numbers like *1996* are first split into *19* and *96*, which better reflects its most probable pronunciation. The same applies to the in rap often appearing *911* (Mayer et al., 2008) which is the present study is converted to *9 1 1* before converting to *nine one one*. The effectiveness of this method is implied by a increasing Macro-F1 score of up to 0.018 when using phonetic character n-grams representations.

4.3 OPTIMIZING FOR WORD N-GRAMS

When looking at the most informative features of the lyrics, each artist's name and/or their nicknames were the top unigram word features. Hirjee and Brown (2010) noticed this pattern as well, but did not take any effort to remove them, while Mara (2014) does not seem to be aware of this phenomenon at all. As there has been no study that tried LbAA without the influence of artists' their own (nick)names, and to investigate whether there are other important features that can help, artists' (nick)names are replaced by a placeholder in this study. To remove the (nick)names of the artist in their lyrics, a blacklist is composed using Wikipedia. For each artist, their artist name, their nicknames and their official names are blacklisted and are replaced by a *own_name* placeholder. Some nicknames are manually added to the blacklist after consulting the informative word unigrams after the initial blacklist still resulted in some a few nicknames appearing. After the removal of all the (nick)names, the Macro-F1 score drops to 0.764.
Interesting features are now starting to arise (see Table 3). 2Pac's *25* is an extraction of the verdict on murder, which sentences someone '25 to life'[2]. Words as *codeine* and *molly* in Future's features are referring to drugs. *Hollygrove* is the neighborhood in Louisiana where Lil Wayne grew up, while *crip* refers to a Los Angeles-based gang. As such, removing (nick)names from the lyrics causes lexical features to appear which reveals an artists' distinct background and/or discussed topics.

Artist	Top 10 most informative word unigrams									
2Pac	25	knows	rebel	homies	ciety	troubles	thug	friends	troublesome	motherfuckers
Future	codeine	bands	smash	molly	some	racks	ain	took	these	on
Lil Wayne	hollygrove	yea	young	da	pussy	em	yeah	but	like	and
Snoop Dogg	momma	shoulders	fadin	cool	coolaid	gangsta	cuz	beach	game	crip

Table 3: Top 10 most informative word unigrams of four artists from AAMDS. The most-right unigram in each row is the most informative.

[1] https://pypi.org/project/inflect/
[2] At this stage in the research numbers in their digit representation were not yet converted to their letter-representation

Still, there are questionable features visible in Table 3. 2Pac's *ciety* is a word unigram, while *ciety* is neither in the Cambrige English Dictionairy[3] nor in the Urban Dictionary[4]. Manuel searching for the word *ciety* in 2pac's lyrics reveals it is only used in one line in one song a numerous amount of times, as a repetitive fade-out of *society*. Such incidental n-grams can cause a classifier to overfit on the training data (John et al., 1994). Hence, following Mara (2014)'s example, words that do not occur in at least 5 songs are removed from the vocabulary. This causes the Macro-F1 score to increase by 2.7% to 0.757. Other minimal document frequency values for word unigrams did not pass the score of 0.757.

Other possible settings for the word n-gram tfidf-vectorizer include the removal of stop words, lowercasing and n-gram range. Stop words are usually removed for topic-based classification, as they often don't carry semantic information (Stamatatos, 2009). However, for author discrimination the most common words (and thus often stop words) are among the best features to discriminate between authors in prose (Argamon and Levitan, 2005). Interestingly, the removal of English stop words in these lyrics hardly decreased the Macro-F1 score, which at first seems to imply that such words might be less discriminative in LbAA, compared to very discriminate in prose. Still, *but, on* and *and* are all visible in Table 3, and therefore it is more likely than in lyrics there simply are enough non-stop words that help the classifier if stop words are removed. Also, stop words in lyrics are not identical to stop words in prose, e.g. *da, yeah,* and *em* are arguably stop words, but are not likely to be filtered out by scikit-learn's stop word removal.

By default all letters are lowercased in sklearn's tfidf-vectorizer, as for most classification tasks you probably do not want to distinguish between a capitalized word at the beginning of a sentence and the same word somewhere else in a sentence (e.g. Computer and computer). For this LbAA task though, the Macro-F1 score actually increases with 0.013 when lowercasing is turned off. The reason for this is that named entities such as *Long Beach* and *Queens* are then discriminated from their non-named entity counterparts: *long, beach* and *queens*, causing e.g. *Beach* to appear in the most informative word unigrams. Another reason why stopping the default of lowercasing might help is that the position of a word in the sentence seems to matter. Every artist in the AAMDS utters the word *nigga*, but still *Nigga* manages to be one of 50 Cent's top features, because he is using the word much more in the beginning of sentences than the other artists.

With the word n-gram range changed from unigrams to unigrams and bigrams the Macro-F1 score on AAMDS reached 0.769. Other word n-gram ranges were tested but performed worse than the combination of uni- and bigrams.

With word n-grams being optimized for AAMDS, the same settings and parameters were applied to DADS. Due to the classifier performing low on recall for the two least represented classes, the Macro-F1 score on DADS only reached an 0.675. All parameters and settings were re-evaluated to determine whether a change in them would help the classifier to detect the two least represented classes, but it wouldn't. As Mara (2014) noticed that adding more lyrics, despite being noisy, helps the classifier, steps were taken to add lyrics (see Section 4.5), and due to the success of character n-grams, character n-grams are implemented in an attempt to increase the score on DADS as well.

[3] https://dictionary.cambridge.org/
[4] https://www.urbandictionary.com/

4.4 PUNCTUATION IN LYRICS

In a similar method to word n-grams the optimal settings for character n-grams are determined. However, the optimized character n-grams based on AAMDS scored surprisingly low on DADS, with Macro-F1 scores around 0.6. Therefore, the informative character n-grams of DADS were examined. Interestingly, quite some punctuation, such as exclamation marks and commas, turned out in several top informative character n-grams. This was expected, as Mayer et al. (2008) already suggested special characters might be of interested. However, one must keep in mind that lyrics are merely a transcription of the words uttered by the artist, and almost never the actual lyrics as written by the artists themselves (Leight, 2019).
Therefore, to have an indication how interpretive punctuation in lyrics is, a small research was conducted. Three daily listeners of rap music, with an university-level education and a high proficiency in English were asked to participate individually. Their task was to add punctuation to a verse. Due to all three participants being regular listeners of Eminem, and in Genius it's expected that a person who writes the lyrics of a new song of artist x regularly listens to artist x, a random Eminem song was selected from the data set. Each participant was given the first two verses of the song *Who Knew* as it appears on Genius[5], hence with punctuation included, as a reference. The third verse had all its punctuation removed, with the exception of ′ in concatenated words such as *don't* and *I'd*. The participants were asked to add the punctuation to the third verse in the same style as it is in the first two verses, with regard to the instructions on punctuation in lyrics by Genius[6] which were provided to them and asked to be studied carefully. Then each participant would listen to the first two verses of the song once while reading the according lyrics to be able to get an extra indication of how the punctuation is linked to the lyrics. No time limit was set and they could replay the song as many times as they thought was necessary, while adding punctuation to the third verse. Genius' punctuation rules were also available to them throughout the entire experiment. The annotated third verse is displayed in Figure 3.

As can be noted from the annotations, even in one verse, while the other two were given as an example, there are a lot of differences between each third verse's punctuation. This indicates that punctuation in lyrics is more likely to resemble the style of Genius' contributor, than the style of the writer of the song. Of course, in this research only one verse is annotated and thus statistically is not proving much. And even though AAMDS' Macro-F1 scores were barely influenced by the punctuation removal, the Macro-F1 score of the development set of DADS increased from 0.596 to 0.683, again implicating that punctuation in lyrics might be more related to the contributor's writing style rather than the artist's. In an extra attempt to test this, Genius' lyrics were compared to other databases' lyrics, such as LyricFind[7]. These lyrics turn out to be often identical on punctuation, however, there are regularly lawsuits in the lyrics database industry (Leight, 2019), including one of Genius suing LyricFind of copying their lyrics plus providing evidence for it (Deahl, 2019). Therefore the attempt to compare punctuation between databases was halted for this study, but lyrics variation between databases is definitely an interesting topic in future research.
The removal of punctuation was done using regexes. If a punctuation mark is preceded or followed by a space or newline, the punctuation mark is removed. Punctuation between alphanumeric characters are preserved, such as the dot in *L.A* or the hyphen in *Four-door* (see Table 4). Removing inter-alphanumeric punctuation causes the Macro-F1 score to drop, so it appears that these punctuation marks do matter and are not, or less, dependent on a contributor's writing style.

[5] https://genius.com/Eminem-who-knew-lyrics
[6] https://genius.com/Genius-how-to-add-songs-to-genius-annotated
[7] https://www.lyricfind.com/

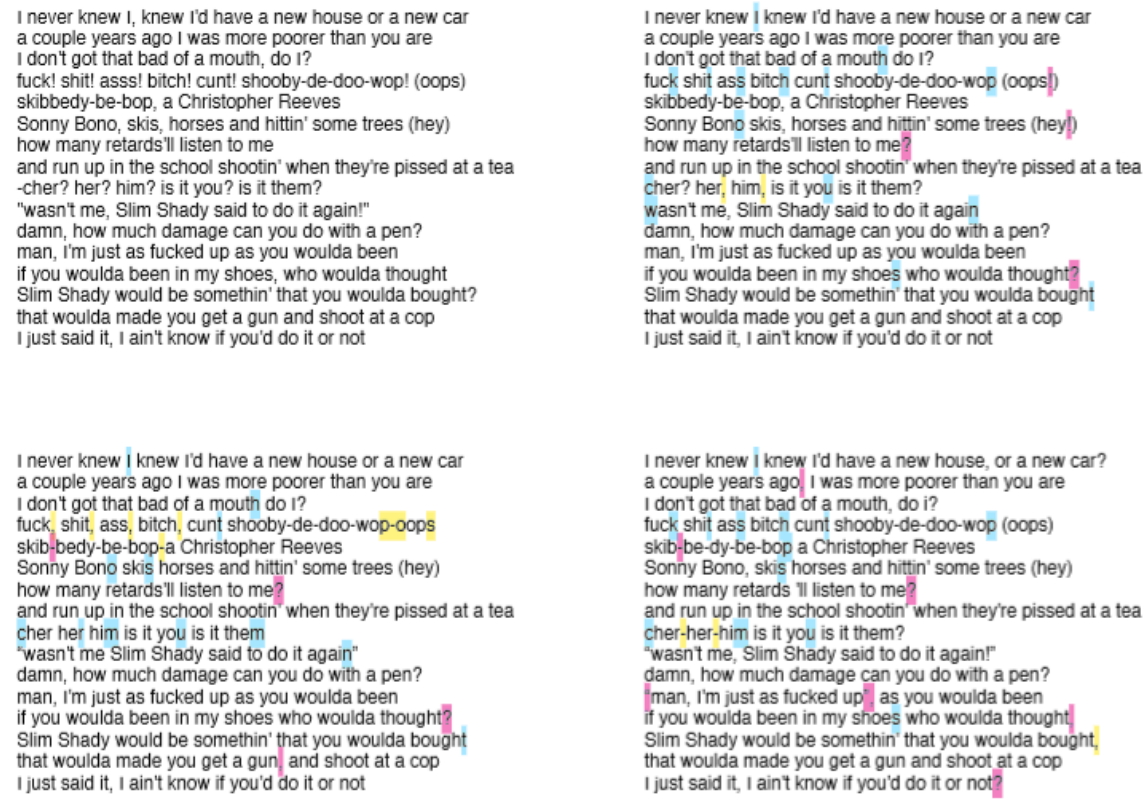

Figure 3: Third verse of *Who Knew* by Eminem. On the top left is the verse as it appears on Genius. The other three texts are the verses with added punctuation by the participants. The blue colour indicates a that a punctuation mark is missing. Yellow indicates that on the right spot a punctuation mark is added, but a different one. Purple indicates that a punctuation mark was added where on Genius there isn't one.

4.5 ADDING VERSES

At first, similar to the present research, Mara (2014) started with a lyrics data set in which for each song there was only one artist. In an experiment where he added lyrics with featuring artists, as long as the main artist was the relevant artist, Mara saw the scores going up. This is somewhat remarkable, since the training data set of an artist now also contains lyrics sang by other artists. Mara (2014) does not analyze why this noisy data helps the classifier, hence two possible notions to why adding this noisy data works are elaborated here:

1. In collaborative songs, the main artist (partly) writes the other artists' lyrics as well. Hence, despite lyrics being sang by someone else, the lyrics still reflects the main artist. If this is the case, low scores are expected on the verse classification of multi-artist songs.
2. Despite containing lyrics written by other artists, lexical features that link to the main artist are still identified by the classifier, while most lexical features written by other artists are righteously ignored.

In an attempt to build on Mara (2014)'s findings, multi-artist songs in which the main artist is the relevant artist are gathered, but verse filtering is applied to add only the lyrics that is attributed to main artist. Hence, the advantage of adding data is preserved, while the disadvantage of noisy data is overcome. The extraction of the relevant verses is possible through annotations in Genius' lyrics which indicate who raps/sings a particular verse (see Figure 4). Using regex these annotations are located. In some verses two artists are continuously alternating sentences, and therefore both names are in the annotation. Therefore only if one name is present in the annotation the verse is deemed reliable. Verses with less than 20 words are disregarded. As the present task is song classification, the verses are only added to the training set. Data sets with verses added to the training data are marked with a +, hence: AAMDS+ is AAMDS with verses added to the training set. What must be noted is that now in those data sets a substantial amount of the training data is in a different format (verses) than the development and test data (songs). Because songs (usually) consists out of multiple verses, absolute statistical features such as word and sentence count are no longer of use. However, due to the lack of contribution of statistical features to the classifier (see Section 4.6), this is not much of a disadvantage.

```
[Verse 1: Jay-Z]
Yeah
Yeah, I'm out that Brooklyn, now I'm down in Tribeca
Right next to De Niro, but I'll be hood forever
I'm the new Sinatra, and since I made it here
I can make it anywhere, yeah, they love me everywhere
I used to cop in Harlem - hola, my Dominicanos
(...)
```

Figure 4: Sample from *Empire State of Mind* by JAY-Z and Alicia Keys. The information in the square brackets indicates who's rapping/singing a particular verse.

The addition of verses contributes to an increase of the Macro-F1 score on the final models of respectively 0.043 for AAMDS+ and 0.077 on DADS+. An advantage of adding these verses is that it contributes to class balancing, especially on DADS+, as the least represented classes are gaining relatively more instances than better represented classes (see Appendix A, Table 11). In this paper, the added verses contribute to an increase of amount of instances is 53.8%, comparing to an increase of 74.6% in Mara (2014). However the increase of Mara's best model with the

extra songs resulted in an increase of 2.3% on accuracy while by adding verses the accuracies of DADS+ and AAMDS+ are raised by respectively 7.1% and 4.1%. This implies that adding less lyrics with minimal noise should be preferred over more lyrics with more noise. Classifications attempts in the present research in which only the least represented classes have their verses added perform worse than when all classes are added.

4.6 STATISTICAL FEATURES

Statistical features, such as words per line, characters per word and type/token ratio are used as early as in the late 1800s for authorship attribution (Mendenhall, 1887), and are still used today, for both author attribution as well as author profiling, e.g. for gender (Oliveira and Neto, 2017; Alrifai et al., 2017), but are unreliable to be used without other features (Stamatatos, 2009). These statistical features are also often used in music genre classification tasks. E.g. rap music is relatively easy distinguishable from other music genres with statistical features, as the word utterance speed is much higher Mayer et al. (2008), which leads to word count being a very informative feature to detect rap music through lyrics (Fell and Sporleder, 2014).
As mentioned in the previous section, absolute count statistics such as word and sentence count lose their value due to the combination of verses and songs in the training data. This is confirmed through these features being unable to significantly pass the baseline of 0.125. Other statistical features, based on Mayer et al. (2008), Fell and Sporleder (2014) and Stamatatos (2009), using relative statistics such as unique word ratio and repeated sentence ratio are examined, but none can significantly aid the lexical n-gram-based classifier in achieving higher scores. This is partly caused by generalizing choruses, hooks and verses into one category: 'verses'. (see Section 1.2). Every artist has both long and short 'verses' and songs, and short verses might lead to a higher unique word ratio, as there is less room for repetition, while choruses are usually more repetitive. This high variance in lyrics is the most probable cause of statistical features not aiding the classifier. Punctuation counts (or ratio's) as used in Mayer et al. (2008) are disregarded since punctuation has been deliberately removed from the lyrics in the present study (see Section 4.4).

4.7 PHONETIC REPRESENTATIONS

As mentioned in Section 4.4, lyrics are merely a transcription of the sounds uttered by an artist transformed to e.g. the English language. Sounds produced by human language are ambiguous in natural language alphabets. For example, there are two pronunciations for *data*, but their difference cannot be written using the Latin alphabet. This difference can be written using a phonetic alphabet. Hence, a perfect transcription of a song text should actually be written using phonetics, but this would be clearly inconvenient for most people as phonetic languages are not commonly understood. In this paper attempts are made to convert lyrics into their phonetic representation using three different algorithms (see below). The goal of classifying on phonetic representations is to determine whether artists are distinguishable through there phonetics. E.g *cough, tough, dough, through, thorough* and *plough* all share the same lexical character n-gram *ough*. Phonetically however, each words ending is pronounced differently[8]. On the other hand, *two* and *to* are not detectable as similar through character n-grams larger than one, while phonetically they are identical. Hence, a phonetic representation can show both similarities and dissimilarities which do not appear in its lexical representation.

[8] https://pronunciationstudio.com/7-pronunciations-ough/

- NLTK's **Carnegie Mellon Pronouncing Dictionary Corpus Reader** is a dictionary-based phonetic converter. Its advantage is that it is very accurate. Each word that is inserted returns the phonetic representation(s) of the word. In the present paper the first option of multiple representations is chosen. The dictionary consists of 12,7069 words[9]. Hence, its disadvantage is that words that are not in their dictionary cannot be converted to a phonetic representation and are thus left out in this representation. This is a drawback, especially for rap lyrics, in which a lot of slang is used (Fell and Sporleder, 2014).
- **Soundex** was developed for indexing names by sound as pronounced in English. The exact workings of the algorithm are described on `https://www.archives.gov/research/census/soundex`. Through Soundex, some words with identical pronunciation such as *to* and *two* are both converted to the identical Soundex encoding *tooo*.
- **Metaphone** is a phonetic algorithm based on Soundex for indexing words by their English pronunciation (Philips, 1990) but it more accurately encodes words by using variations and inconsistencies in English spelling.

All three phonetic algorithms are applied and grid-searched for their optimal parameters. None of these phonetic features outperform the optimal character n-gram feature. A possible cause for this is that in order to rhyme, rappers regularly creatively divert from the official pronunciation. E.g., Eminem rhymes *Batman* with *Saddam* and the 'unrhymable' *oranges* with *hinges, syringes* and *inches* by pronouncing it creatively (see Figure 5). Therefore, converting lyrics to its phonetic representation does not necessarily represent the actual phonetics of the song, which is a probable cause why phonetic representations are incapable of outperforming the lexical. Still, NLTK's CMPDCR-based representation made in into multiple final models as it did add some valuable information to the classifier (see Section 4.11).

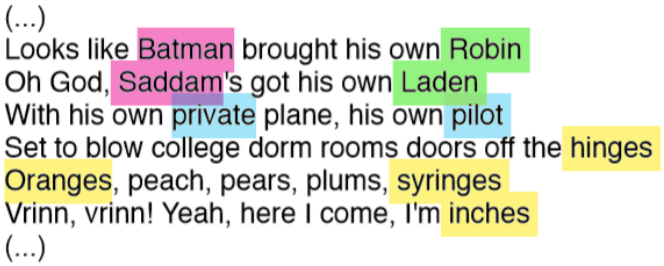

Figure 5: Extraction of the first verse of *Business* by Eminem. Words that are pronounced to rhyme with each other are marked per colour.

4.8 LYRICS-BASED GENDER CLASSIFICATION

No lyrics-based (or poetry-based) gender classification has been performed before, hence all knowledge is based on prose-based gender classification. On prose, character n-grams are often the most informative for gender classification (Basile et al., 2017; Rangel et al., 2017). Therefore, the usually successful features, such as word n-grams and character n-grams, are implemented. Next to the classical gender classification features a relatively new and less conventional technique is applied: 'bleaching'. In a research by van der Goot et al. (2018) bleaching techniques are applied to cross-lingual gender classification on Twitter data, in which lexical strings are transformed to more abstract features. These bleached representations turn out

[9] `https://www.nltk.org/_modules/nltk/corpus/reader/cmudict.html`, accessed on 25-04-2020

to be better transferable across languages than lexical features. Even though lexical features still outperform bleached representations in a one language setting, bleached representations still managed to beat the baseline of 50% by achieving 68.7%.
Six different bleached representation are presented in van der Goot et al. (2018), of which five are used in the present study (see Table 4). One representation is left out as its distinction is that it takes emoticons and emojis into account, which are not present in lyrics.

Original	Four-door	Maybach	Four-door	I	drive	anything	Ye
Frequency	1	1	1	3	2	1	2
Length	09	07	09	01	05	08	02
PunctC	W-W	W	W-W	W	W	W	W
Shape	ULLXLL	ULL	ULLXLL	U	LL	LL	UL
Vowels	CVVCOCVVC	CVCCVCC	CVVCOCVVC	V	CCVCV	VCCCCVCC	CV

Table 4: Bleached representations of a line in *Mask Off* by Future

These abstract representations are build at different stages during preprocessing. In experiments where these representations are created without the artist name removed first, some representations manage to capture distinctive spelling names, such as 2pac. Therefore, all names are replaced by the own_name placeholder before building these representations.

- The **Frequency Representation** represents each word as its binned frequency in the training data. In van der Goot et al. (2018) the bins are sized by orders of magnitude. Due to a smaller data set, different bin sizes were attempted for the present paper, but none of the Frequency Representations appeared meaningful to the classifier.

- The **Length Representation** is implemented after the conversion of digit-represented numbers to letter-represented numbers, since the length of *10* and *70* is both two, but the length of *ten* and *seventy* are respectively three and seven. For tweets this difference is not relevant, but as lyrics are based on vocalizing words, this distinction matters and thus the letter-represented number representation is preferred (Section 4.2).

- The **PunctC Representation** merges all consecutive alphanumeric characters to one 'W' and leaves other characters be. Due to the removal of punctuation at both ends of words this representation is largely stripped from its value. Still, this representation is capable of detecting abbreviations, e.g. *L.A.* is represented by *W.W* and *N.Y.C* as *W.W.W*, and is used in almost every final model (see Section 4.11).

- The **Shape Representation** represents lowercase characters with 'L', uppercase characters with 'U', digits with 'D' and all other characters with 'X'. Consecutive repetitions of characters in their transformed state are condensed to a maximum of two for greater generalization. Due to this representation making use of digits, the representation is build before the conversion of digit-represented numbers to letter-represented numbers.

- The **Vowels Representation** converts vowels to 'V', consonants to 'C' and other characters to 'O'. This representation is constructed after the conversion of digit-represented numbers to letter-represented numbers, for similar reasons as explained for the Length Representation.

Using comprehensive grid searches all features were individually examined on GBDS (see Table 5). What can be noted is that all bleached representations significantly outperform the baseline of 0.5. Due to their relatively high n-gram range

Representation	Analyzer	Vectorizer	n-gram Range	Min df	Macro-F1 Score
Length	word	Tfidf	1-4	5	0.668
PunctC	char	Tfidf	7-10	1	0.621
Shape	word	Tfidf	6-10	10	0.665
Vowel	char	Tfidf	8-8	5	0.773
Lexical	char_wb	Tfidf	6-6	10	0.832

Table 5: Optimized parameters for each representation on GBDS.

these bleached representations appear to extract the most information from relatively long sequences, which is not very surprising as short sequences are more likely to have identical bleached representations. The relatively high score for the Vowel Representation is likely due to the fact that the Vowel Representation is the closest related to the highest scoring representation: lexical. Nevertheless, the Vowel Representation n-grams are difficult to link to actual words (see Appendix B, Figure 8), and attempts to trace Vowel Representations to their original lexical form by looking at the lexical informative character n-grams failed. It is noteworthy to mention that every final model in the present paper makes use of at least one of these abstract representations, regardless of classification task, confirming the successfulness of these representations.

4.9 VERSE CLASSIFICATION ON COLLABORATIVE ALBUMS

There are some differences with the classification task on CADS than on the other data sets. Due to the limited amount of songs on the collaborative albums, there is no development set. Instead, two models are applied to the test set to see which works best.

1. A model optimized using k-vold cross validation on both the songs and the verses of the artists. Contrary to classification on DADS and AAMDS, the test instances in CADS are verses. Therefore the verses can be added to the training set before cross validation.

2. A model optimized using k-vold cross validation on verse level. Songs are split into their verses to have the same format in the training data as the test data: verses.

Both models are optimized using grid searches on CADS.a, as that data set consists out of two Afro-American rappers, indisputably related to hip-hop, and thus more homogeneous than CADS.b, which includes Damian Marley, a Jamaican-based rapper and singer, affiliated to both reggae and rap. As Fell and Sporleder (2014) noted, reggae artists are relatively easy detectable through their often Jamaican slang and Rastafarian terms.

4.10 EXPERIMENTS

A few more small experiments that are inspired by relevant literature have been conducted and are explained here.
With Part of Speech (henceforth POS) tags being very informative for music genre classification (Mayer et al., 2008; Fell and Sporleder, 2014), POS tags are attempted to aid the classifier in the present paper for LbAA, but nearly all attempts failed, which is similar to Mara (2014)'s findings.
Hirjee and Brown (2010) mentioned that it is relatively easy to apply author attribution to rap lyrics, as rappers tend to drop their (nick)names and associated music labels and groups regularly. With their (nick)names already removed, spaCy's[10] Named Entity Recognition tagger is applied to create a representation without those references, and without other named entity references, such as mentions of the neighbourhood they grew up in. Instead, a name is convert to *PERSON*, an organization to *ORGANIZATION* and so on. Even though the accuracy of this NER-tagger on lyrics is unknown, the average Macro-F1 score based on lexical n-grams of AAMDS shows no noteworthy change when the named entities are bleached out. Therefore, despite what Hirjee and Brown (2010) wrote is true, this does not mean that these names entities are necessary for a classifier to distinguish artists from each other.
Due to the set up of CADS, its classification task can also be converted to author verification instead of author attribution. In this particular author verification task, verses are compared to determine whether they belong to the same artist. As this is rather a side-experiment, Hürlimann et al. (2015)'s successful Groningen Lightweight Authorship Detection (GLAD) model is used[11], instead of building a new author verification model. With Macro-F1 scores of around 0.7, GLAD outperformed the baseline significantly. To dive deeper into GLAD to see if this model, or the lyrics can be optimized for it to achieve higher scores is beyond the scope of the present paper, but could definitely be interesting for future research.

4.11 FINAL MODELS

Four different final models are used: one for author attribution, one for author profiling on gender and two for author classification on verses in collaborative albums. All models are based on a linear SVM with the same preprocessing steps, such as (nick)name removal and punctuation filtering. The default of lowercasing is turned off for arguments explained in Section 4.3. The features are assembled by starting with the best individual feature for each of the classification tasks mentioned above. Another feature is added and the classifier runs. If the Macro-F1 score improves with the extra feature, the extra feature is added to the feature set. The other way around, in which the classifier starts with all features and by deleting each feature at a time the impact of that feature is detected, resulted in a lower final Macro-F1 score.

[10] https://spacy.io/
[11] https://github.com/pan-webis-de/glad

4.11.1 Final Song-lyrics Author Attribution Model

The model for Lyrics-based Author Attribution on song level is displayed in the table below (6). The optimal feature set is optimized on k-vold classification on AAMDS due to the hyptohesis that that data set is more difficult due to homogeneity. The Macro-F1 score on the development set is 0.853 and 0.864 for respectively DADS+ and AAMDS+. The regularization parameter of the linear SVM (often denoted as C) is set to 200, which means that the hyperplane that is used to separate the classes is relatively small. Three of van der Goot et al. (2018)'s abstract representations made the final feature set, despite their intent to use for the gender classification task, and not LbAA.

Feature	Analyzer	Vectorizer	n-gram Range	Min df	Verses	Macro-F1
Lexical n-grams	char_wb	Tfidf	3-5	5		
Lexical n-grams	word	Tfidf	1-2	5		
PunctC repr.	char	Tfidf	4-8	5	yes	0.859
Shape repr.	char	Tfidf	4-8	5		
Vowel repr.	char	Tfidf	4-8	5		

Table 6: Final feature set for Lyrics-based Author Attribution. The Macro-F1 score is the average score of the classifier on the development sets of AAMDS+ and DADS+.

4.11.2 Final Author Profiling Model for Gender Classification

Contrary to the previous model, the gender-classification model did not profit from additional verses in the training set. The addition of verses to the training set might only benefit imbalanced data sets if the verses contribute to balancing. Due to the perfect balance between both genders in this research, the verses, due to their smaller vocabulary, rather hurt than help the classifier.

Feature	Analyzer	Vectorizer	n-gram Range	Min df	Verses	Macro-F1
Lexical n-grams	char_wb	Tfidf	5-5	10		
Lexical n-grams	word	Tfidf	1-1	5		
Phonetic n-grams	word	Tfidf	1-1	5	no	0.819
Length repr.	word	Tfidf	1-4	5		
PunctC repr.	char	Tfidf	7-10	1		
Shape repr.	char	Tfidf	9-15	5		

Table 7: Final feature set for gender classification on lyrics.

4.11.3 Final Collaborative Album Models

In Tables 8 and 9 are the final feature sets for CADS. Interestingly, when learning solely from verses (Table 9), the classifier relies mostly on word n-grams, and with a relatively large range. As no development set is created for this task, the scores on mentioned in the tables below are not on the development set, but of the k-vold classification on the training data of CADS.a.

Feature	Analyzer	Vectorizer	n-gram Range	Min df	Songs	Macro-F1
Lexical n-grams	char_wb	Tfidf	6-6	5	yes	0.901
Lexical n-grams	word	Tfidf	1-2	5		
Phonetic n-grams	word	Tfidf	1-3	5		
Length repr.	char	Tfidf	7-7	1		
POS repr.	char	Tfidf	7-7	5		

Table 8: Final feature set for collaborative album artist attribution based on songs with added verses.

Feature	Analyzer	Vectorizer	n-gram Range	Min df	Songs	Macro-F1
Lexical n-grams	char_wb	Tfidf	3-7	5	no	0.774
Lexical n-grams	word	Tfidf	1-2	5		
Phonetic n-grams	word	Tfidf	1-2	5		
Soundex n-grams	word	Tfidf	1-1	5		
PunctC repr.	word	Tfidf	1-6	5		
Shape repr.	word	Tfidf	1-6	5		

Table 9: Final feature set for collaborative album artist attribution based solely on verses.

This major difference in Macro-F1 score is most likely a result of verses being harder to classify than having a harder time to train on only verses. This is further elaborated in Section 5.3.

5 RESULTS AND DISCUSSION

The results of the final models are shown and discussed in this chapter. The most interesting and relevant figures and tables are added in here. Consult Appendix C for extra details on the classification results of each task and each class. Appendix B contains extra tables and figures on the informative features.

5.1 AUTHOR ATTRIBUTION

The Macro-F1 scores on the test set of AAMDS+ and DADS+ are respectively 0.899 and 0.794. While AAMDS+'s Macro-F1 test score rises with over 0.035 compared to the development data, DADS' declines with 0.07. A possible explanation for this difference is that the model was optimized on AAMDS+ and then applied to DADS+, hence the parameters might not be the most optimal for DADS+. However, in an attempt to confirm or debunk this another classifier is optimized and applied on DADS+, resulting is a Macro-F1 score of 0.803 instead of 0.794. Due to this small difference it can be concluded that the model generalizes properly, and that DADS' test data is just more difficult for the classifiers than the development data.
Through analyzing the confusion matrix of DADS+, the conclusion can be made that ethnicity and gender plays no role in this data set. Neither the two women (*Kim and Minaj*) nor the two Caucasian males (*Emi* and *MGK*) have been misclassified with each other (see Figure 6). Therefore the hypothesis that gender and ethnicity have an influence in Lyrics-based Author Attribution is, based on this study, unlikely.

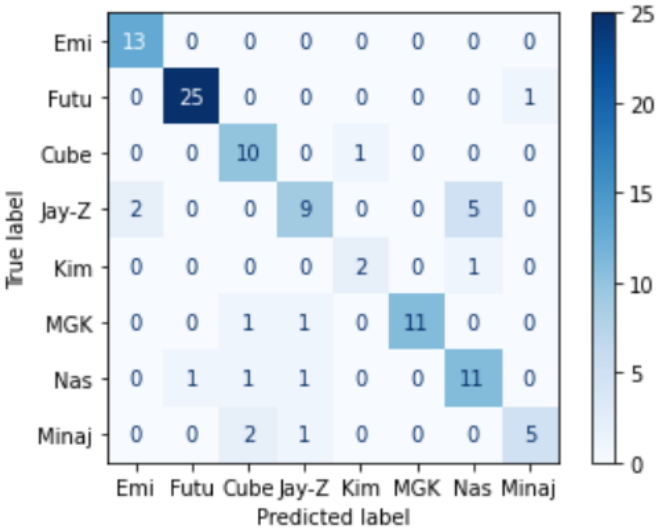

Figure 6: Confusion matrix of DADS+.

The only obvious misclassification that can be derived from the confusion matrix is the classifier labeling five out of sixteen Jay-Z songs lyrics as being of Nas. Surprisingly, this only happens two out of fourteen times in AAMDS+.

5.2 LYRICS-BASED GENDER CLASSIFICATION

The gender model scored a Macro-F1 score of 0.84 on the test set of GBDS and thus performing slightly better than on the development set (0.819). For the first time in can be considered proven that an algorithm can detect someone's gender from lyrics far beyond the rate of chance (0.5), though these scores are lower than gender classification tasks on prose, such as tweet-based gender classification (0.99) (Miller et al., 2012), and blog-based gender classification (0.89) (Mukherjee and Liu, 2010). The most informative unigrams and bigrams of the male and female artists are displayed in Figure 7. It is worthy to note that the most informative unigrams for females are referring to women: *bitches* and *girls*, while the most informative male unigrams are referring to men: *nigga* and *niggas*, which suggests (rap) artists tend to talk mostly about their own gender.
What must be kept in mind is that with only seven individuals representing each gender, unintended artist bias can occur, which is likely why *Brooklyn* is in the females most informative word n-grams. This unintended artist bias should be limited however, due to this data set being completely balanced artist-wise (see Section 3.3).

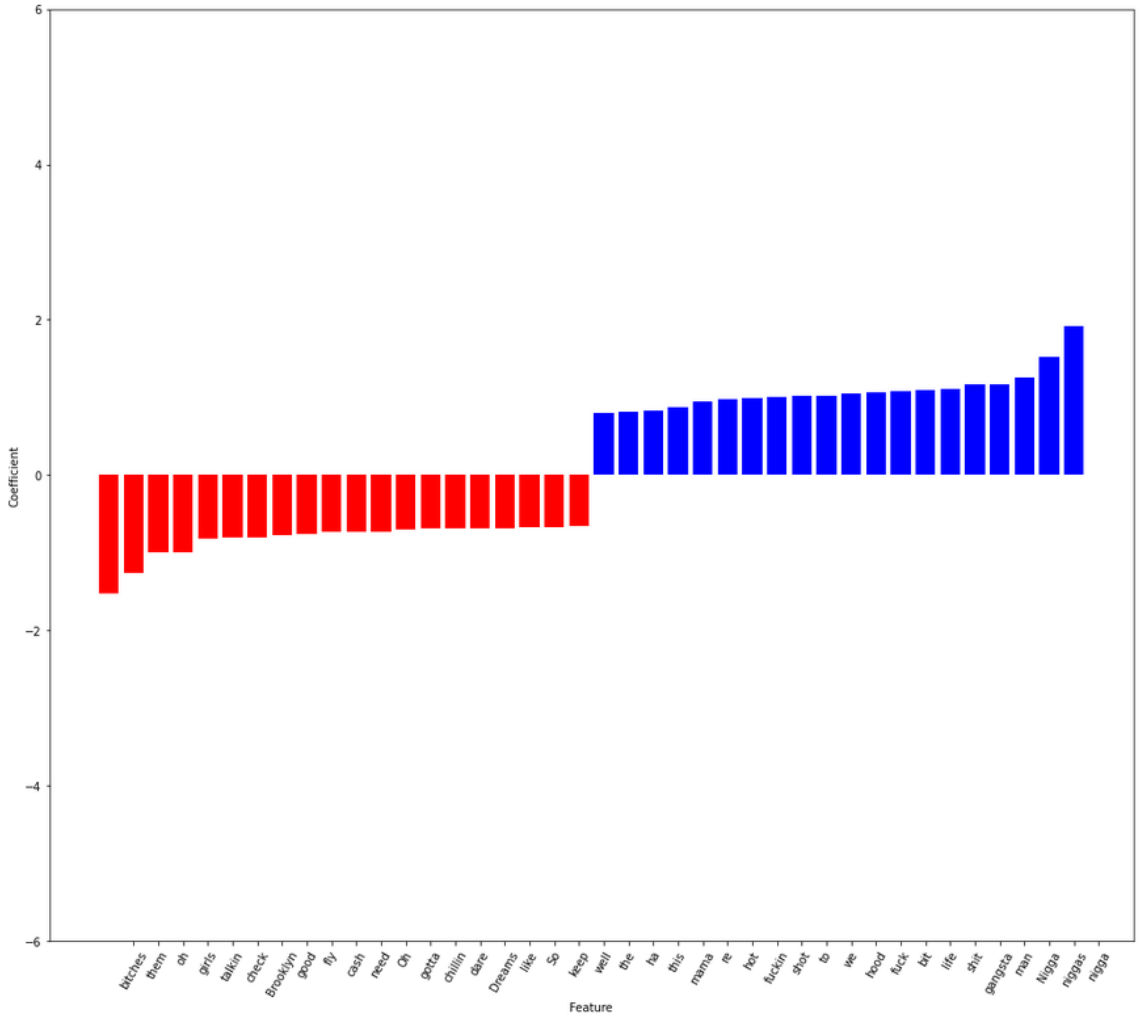

Figure 7: Most informative features of females (red) and males (blue) in GBDS

5.3 COLLABORATIVE ALBUM DATA SET

Despite having a completely opposite class distribution in the training and test set, both classifiers still perform quite well, with Macro-F1 scores of around 0.80 on CADS.a and 0.90 on CADS.b. Interestingly, the differences in scores between training on songs and verses and training on verses only can arguably be neglected (see Tables 19 and 20 in Appendix C.1). Conform the expectations, the scores on CADS.b are higher than on CADS.a, due to Jamaican Patois[1] (*dem, mi, fi*) and Rastafarian terms (*Zion, Jah*). These in English atypical word unigrams are helping the classifier to distinguish Damian Marley from Nas. Despite the neglectable differences in scores between the model trained on both songs and verses and the model trained only on verses, the most informative word n-grams differ per model (see Table 2). When trained on verses only, nouns are more prominently present (*power, one love, Zion, Jah, trouble*), while when trained on both songs and verses function words are the most dominant (*my, the, what, this, no, pon, fi*).

Artist	Songs	Informative unigrams and bigrams							
Nas	yes	my	nigga	niggas	was	the	what	this	black
	no	my	this	that	nigga	niggas	power	one love	black
Marley	yes	dem	mi	And	no	well	ancient	pon	fi
	no	And	Well	dem	Zion	mi	Jah	upon	trouble

Table 10: Most informative word n-grams on CADS.b The most informative are on the left side of the table.

[1] Jamaican Patois is an English-based Creole language with West-African influences.

6 CONCLUSION

The most informative features for Lyrics-based Author Attribution are word and character n-grams, conform Mara (2014) (Research Question 1). Nevertheless, less conventional representations such as phonetic representations and van der Goot et al. (2018)'s abstract representations can still help the classifier reach performances not achievable through lexical features alone. Mara (2014)'s findings that POS-tags are not adding valuable information to the classifier for LbAA is confirmed in the present paper. Even though Hirjee and Brown (2010) rightfully notes that rappers are easily distinguishable through e.g. their (nick)names, record label names and group names, the present study proves that by removing those features using blacklisting and NER-tagging the artists are still well distinguishable through other words such as function words, which is similar to author attribution on prose (Argamon and Levitan, 2005).

Author Attribution features on prose are thus also relevant to LbAA (Research Question 2). Differently from author attribution on prose, the present study argues that punctuation might not be a valid feature for LbAA, as its transcription is too interpretable. Lowercasing is discouraged as well, because lowercasing erases the differences between words in the beginning of a sentence and uttered somewhere else, and it erases the differences between some named entities from their lexically identical non-named entity counterparts (*Queens* and *queens*).

Features relevant for music genre classification are less applicable to LbAA (Research Question 3). Mayer et al. (2008)'s most successful features for music genre classification using Support Vector Machines were POS tags, text statistics and rhyme. With the exception of rhyme, which Hirjee and Brown (2010) already proved to be informative for LbAA, POS tags and text statistics turn out to be of little use in LbAA. Due to similar important features for music genre classification in Fell and Sporleder (2014), it can be concluded that LbAA is much closer related to author attribution on prose than it is to music genre classification.

Gender and ethnicity turn out to have minimal, or perhaps no impact at all on LbAA in a data set consisting out of artists of different genders and ethnicities (Research Question 4). Still, an artist's gender can be detectable for an algorithm through their lyrics with a high accuracy (0.84), in which each gender is most identifiable through gender related words.

Despite artists cooperating on songs in collaborative albums, their verses are still well distinguishable, with accuracies between 0.8 and 0.9 (Research Question 5). Therefore it can be concluded that in such collaborative songs, the artists still largely retain their own lexicon and style. This is an extra conformation that adding collaborative song lyrics should preferably be based on individual verses, instead of entire songs, as in Mara (2014).

7 FUTURE WORK

Due to the novelty of the research field of Lyrics-based Author Classification (LbAC), the amount of potential future work is enormous. An interesting option which could greatly aid LbAC is to combine this paper's top models with Hirjee and Brown (2010)'s rhyme-based top model, as every models is capable of significantly outscoring the baseline scores. Semantic features have not been applied yet to any LbAC, and so its potential is yet to be discovered.

In the current available LbAC papers, only classical Machine-Learning algorithms such as Naive Bayes and Support Vector Machines have been implemented. The more novel and usually high performing Deep Learning Networks are perhaps expected to outperform these classical algorithms on LbAC, though for example on Cross-Domain Author Attribution these classical algorithms based on character n-grams can still systematically outperform Deep Learning Networks (Kestemont et al., 2018).

In the present paper, which features the first Lyrics-based Gender Classification task, a relatively small amount of fourteen unique persons are in the data set, seven for each gender. Therefore in future research gender-based data sets in which more artists are taken into account are perhaps intriguing, as there each artist's influence declines.

The first steps into the field of verse-level author attribution have been taken in this study. But, the used data sets of the verse-level author attribution task lends itself perfectly for other variances of author classification, such as author verification applied in Hürlimann et al. (2015) and author diarization, tackled e.g. in Sittar et al. (2016). Nevertheless, also in the author attribution on verse-level much more research is still possible by using e.g. more than two classes.

What must not go unnoticed is that all LbAC so far is based on rap music, which is known to be heavily focused on lyrics, and thus is likely to be less difficult for author classification tasks than other music genres. Therefore LbAC on non-rap lyrics would be interesting, however, it requires the often unknown knowledge as to who actually wrote the lyrics. Still, with enough effort and expertise, creating such a data set should be feasible.

BIBLIOGRAPHY

Alrifai, K., G. Rebdawi, and N. Ghneim (2017). Arabic tweeps gender and dialect prediction. In *CLEF (Working Notes)*.

Argamon, S. and S. Levitan (2005). Measuring the usefulness of function words for authorship attribution. In *Proceedings of the 2005 ACH/ALLC Conference*, pp. 4–7.

Basile, A., G. Dwyer, M. Medvedeva, J. Rawee, H. Haagsma, and M. Nissim (2017). N-gram: New groningen author-profiling model. *arXiv preprint arXiv:1707.03764*.

Bird, S., E. Klein, and E. Loper (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Bozkurt, I. N., O. Baglioglu, and E. Uyar (2007). Authorship attribution. In *2007 22nd international symposium on computer and information sciences*, pp. 1–5. IEEE.

Carlson, N. (2014). The inside story of how rap genius fired a cofunder and just raised $40 million (annotated!). *Business Insider*. https://www.businessinsider.com/the-inside-story-of-how-rap-genius-fired-a-cofounder--and-just-raised-40-million-annotated-international=true&r=US&IR=T, accessed on 16-04-2020.

Cassidy, F. G. (1966). Multiple etymologies in jamaican creole. *American speech 41*(3), 211–215.

Deahl, D. (2019). Genius sues google over allegedly stolen song lyrics. *The Verge*. https://www.theverge.com/2019/12/3/20993621/genius-google-lawsuit-stolen-lyrics-lyricfind, accessed on 03-05-2020.

Fell, M. and C. Sporleder (2014, August). Lyrics-based analysis and classification of music. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, Dublin, Ireland, pp. 620–631. Dublin City University and Association for Computational Linguistics.

Gabay, T. (2019). Literature review on authorship attribution and author profiling.

Gabay, T. (2020). Literature review on lyrics-based classification and author attribution.

Guo, S. and S. Khamphoune (2013). "i'm different, yeah i'm different": Classifying rap lyrics by artist.

Hirjee, H. and D. Brown (2010). Using automated rhyme detection to characterize rhyming style in rap music.

Honnibal, M. and I. Montani (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in Science & Engineering 9*(3), 90–95.

Hürlimann, M., B. Weck, E. van den Berg, S. Suster, and M. Nissim (2015). Glad: Groningen lightweight authorship detection. In *CLEF (Working Notes)*.

Jay-Z (2010). *Decoded*. Random House.

John, G. H., R. Kohavi, and K. Pfleger (1994). Irrelevant features and the subset selection problem. In W. W. Cohen and H. Hirsh (Eds.), *Machine Learning Proceedings 1994*, pp. 121 – 129. San Francisco (CA): Morgan Kaufmann.

Kestemont, M., M. Tschuggnall, E. Stamatatos, W. Daelemans, G. Specht, B. Stein, and M. Potthast (2018). Overview of the author identification task at pan-2018: cross-domain authorship attribution and style change detection. In *Working Notes Papers of the CLEF 2018 Evaluation Labs. Avignon, France, September 10-14, 2018/Cappellato, Linda [edit.]; et al.*, pp. 1–25.

Leight, E. (2019). https://www.rollingstone.com/music/music-features/genius-lawsuit-google-922666/. *Rolling Stone*. https://www.rollingstone.com/music/music-features/genius-lawsuit-google-922666/, accessed on 03-05-2020.

Mara, M. (2014). Artist attribution via song lyrics.

Mayer, R., R. Neumayer, and A. Rauber (2008). Rhyme and style features for musical genre classification by song lyrics. In *Ismir*, pp. 337–342.

Mayer, R. and A. Rauber (2011). Musical genre classification by ensembles of audio and lyrics features. In *Proceedings of International Conference on Music Information Retrieval*, pp. 675–680.

McIntyre, H. (2019). The global music industry hit $19 billion in sales in 2018, rising by almost 10%. *Forbes*. https://www.forbes.com/sites/hughmcintyre/2019/04/02/the-global-music-industry-hits-19-billion-in-sales-in-2018-jumping-by-almost-10/#160c271618a9, accessed on 14-04-2020.

Mendenhall, T. C. (1887). The characteristic curves of composition. *Science 9*(214), 237–249.

Miller, Z., B. Dickinson, and W. Hu (2012). Gender prediction on twitter using stream algorithms with n-gram character features. *International Journal of Intelligence Science 2*(04), 143.

Mitkov, R. and M. P. Oakes (2019). *Author Profiling and Related Applications*. Oxford University Press.

Mosteller, F. and D. Wallace (1964). Inference and disputed authorship: The federalist.(1964).

Mukherjee, A. and B. Liu (2010). Improving gender classification of blog authors. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, USA, pp. 207–217. Association for Computational Linguistics.

Oliveira, R. R. and R. F. O. Neto (2017). Using character n-grams and style features for gender and language variety classification. In *CLEF (Working Notes)*.

Overdorf, R. and R. Greenstadt (2016). Blogs, twitter feeds, and reddit comments: Cross-domain authorship attribution. *Proceedings on Privacy Enhancing Technologies 2016*(3), 155–171.

pandas development team, T. (2020, February). pandas-dev/pandas: Pandas.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12*, 2825–2830.

Pennacchiotti, M. and A.-M. Popescu (2011). A machine learning approach to twitter user classification. In *Fifth International AAAI Conference on Weblogs and Social Media*.

Philips, L. (1990). Hanging on the metaphone. *Computer Language 7*(12), 39–43.

Rangel, F., P. Rosso, M. Potthast, and B. Stein (2017). Overview of the 5th author profiling task at pan 2017: Gender and language variety identification in twitter. *Working Notes Papers of the CLEF*, 1613–0073.

Rytlewski, E. (2012). Phasing out the skit: How hip-hop outgrew one if its most frustrating traditions. *A.V. Club*. https://music.avclub.com/phasing-out-the-skit-how-hip-hop-outgrew-one-of-its-mo-1798229914, accessed on 17-04-2020.

Santosh, K., R. Bansal, M. Shekhar, and V. Varma (2013). Author profiling: Predicting age and gender from blogs. *Notebook for PAN at CLEF 2013*.

Shepherd, J. and P. Wicke (1997). *Music and cultural theory*. Polity Press Cambridge.

Sittar, A., H. R. Iqbal, and R. M. A. Nawab (2016). Author diarization using cluster-distance approach. In *CLEF*.

Stamatatos, E. (2009). A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology 60*(3), 538–556.

Stolerman, A., R. Overdorf, S. Afroz, and R. Greenstadt (2014). Breaking the closed-world assumption in stylometric authorship attribution. In *IFIP International Conference on Digital Forensics*, pp. 185–205. Springer.

Tsaptsinos, A. (2017). Music genre classification by lyrics using a hierarchical attention network. ICME.

van der Goot, R., N. Ljubešić, I. Matroos, M. Nissim, and B. Plank (2018). Bleaching text: Abstract features for cross-lingual gender prediction. *arXiv preprint arXiv:1805.03122*.

Waskom, M., O. Botvinnik, P. Hobson, J. B. Cole, Y. Halchenko, S. Hoyer, A. Miles, T. Augspurger, T. Yarkoni, T. Megies, L. P. Coelho, D. Wehner, cynddl, E. Ziegler, diegoo020, Y. V. Zaytsev, T. Hoppe, S. Seabold, P. Cloud, M. Koskinen, K. Meyer, A. Qalieh, and D. Allan (2014, November). seaborn: v0.5.0 (november 2014).

Wes McKinney (2010). Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman (Eds.), *Proceedings of the 9th Python in Science Conference*, pp. 56 – 61.

Wiedeman, R. (2014). Genius idea. *New York Magazine*. https://nymag.com/intelligencer/2014/12/genius-minus-the-rap.html, accessed on 16-04-2020.

Appendices

A DATA SETS

Artist	Total Songs	Train songs	Development songs	Test songs	Extra train verses
Future	235	190	23	26	80
Eminem	153	120	15	13	74
Nas	146	119	14	14	68
Ice Cube	138	109	15	11	22
JAY-Z	121	94	10	16	91
MGK	103	82	8	13	47
Nicki Minaj	79	67	9	8	47
Lil' Kim	63	49	10	3	46
TOTAL	1038	830	104	104	475

Table 11: Diverse Artist Data Set.

Artist	Total Songs	Train songs	Development songs	Test songs	Extra train verses
Lil Wayne	338	267	31	40	226
Future	235	193	25	17	80
50 Cent	168	139	12	17	79
Snoop Dogg	157	126	16	15	149
Nas	146	118	12	16	68
Ice Cube	138	110	14	14	22
JAY-Z	121	90	17	14	91
2Pac	100	79	13	8	123
TOTAL	1403	1122	140	141	838

Table 12: Afro-American Male Data Set.

Artist	Gender	Train songs	Development songs	Test songs	Extra train verses
50 Cent Nicki Minaj	Male Female	67	5	7	47
Snoop Dogg Lil' Kim	Male Female	51	6	6	46
Eminem MC LYte	Male Female	44	10	4	5
Nas Iggy Azalea	Male Female	39	3	7	32
Ice Cube Missy Elliot	Male Female	39	5	2	19
JAY-Z Queen Latifah	Male Female	26	5	9	4
2Pac Cardi B	Male Female	24	2	2	9
TOTAL TOTAL	Female Male	290	36	37	81
TOTAL	Both	580	72	74	162

Table 13: Gender-based Data Set.

B FEATURE ANALYSIS

Artist	Most informative word n-grams						
Eminem	goodbye	My name	Remember	as	this	to	and
Future	These	dab	gon	her	Em	Pluto	out the
Ice Cube	Don	neighborhood	gangsta	wanna	fat	War	got to
JAY-Z	said	Fella	Roc Fella	flow	Marcy	and and	uh
Lil' Kim	pussy	do do	Patron	on racks	Put your	bitch	Brooklyn
MGK	fuck don	round here	Ayy	my	Kiss	Ay	up
Nas	hood	they	black	women	The	She	Queens
Nicki Minaj	mma	am	stupid	automatic	boom	blazin	Barbie

Table 14: Extraction of the most informative word n-grams of DADS+.

Artist	Most informative word n-grams						
2Pac	ghetto	Thug	cause	die	Minnie	Society	thug
50 Cent	chase the	Man	calm	Nigga	at Where	shit	shot
Future	Gotta	Yeah Yeah	take	my ho	molly	bands	racks
Ice Cube	fat	Don	how am	Cause	wanna	West	got to
JAY-Z	him	chick	Roc Fella	Fella	Marcy	uh	uhh
Lil Wayne	Orleans	Young Money	pussy	em	Hollygrove	Yeah	But
Nas	made	the man	Queensbridge	of	are	Yo	with
Snoop Dogg	thang	the homie	drip	Beach	dip	homie	game

Table 15: Extraction of the most informative word n-grams of AAMDS+.

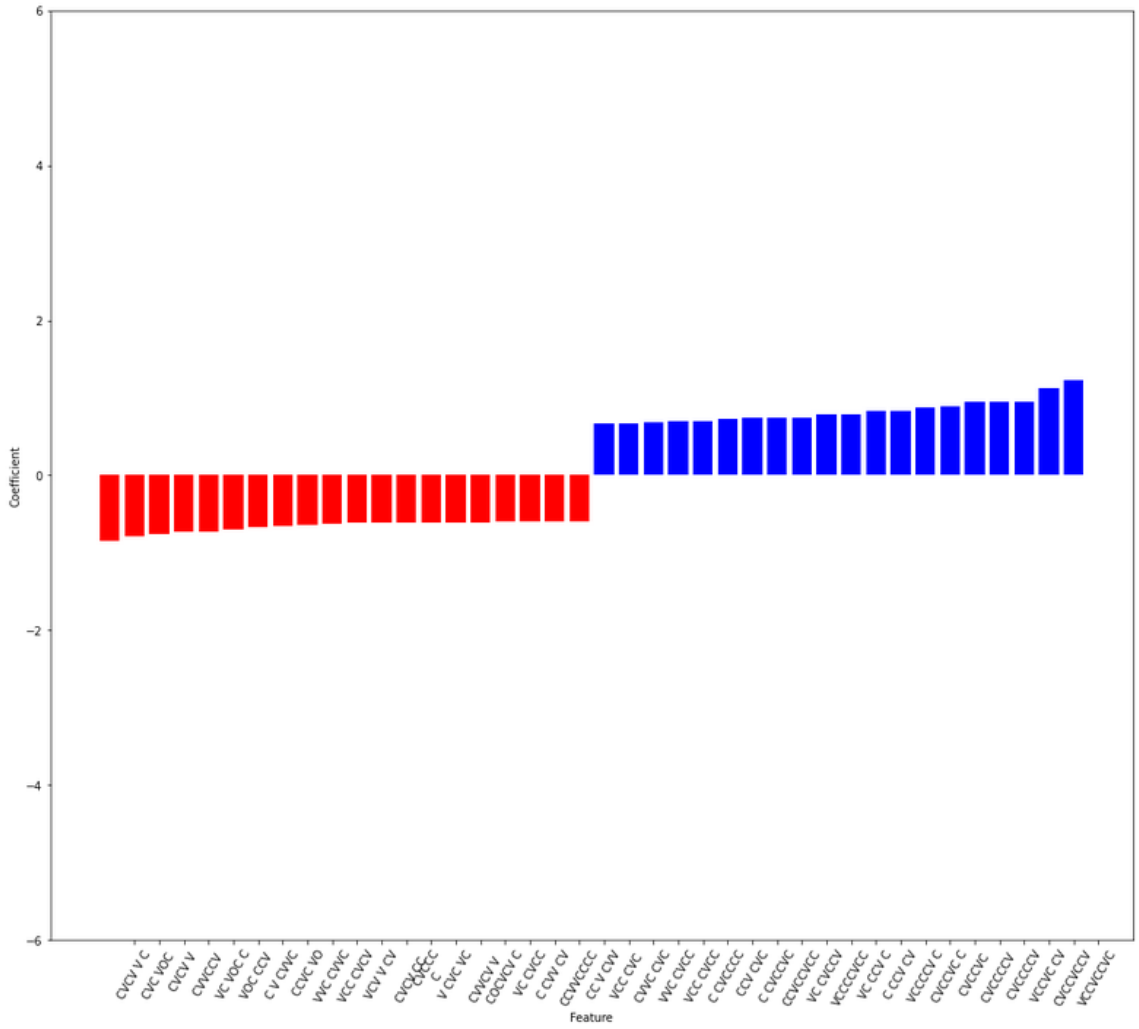

Figure 8: Most informative vowel representations for GBDS. Red resembles female, blue male.

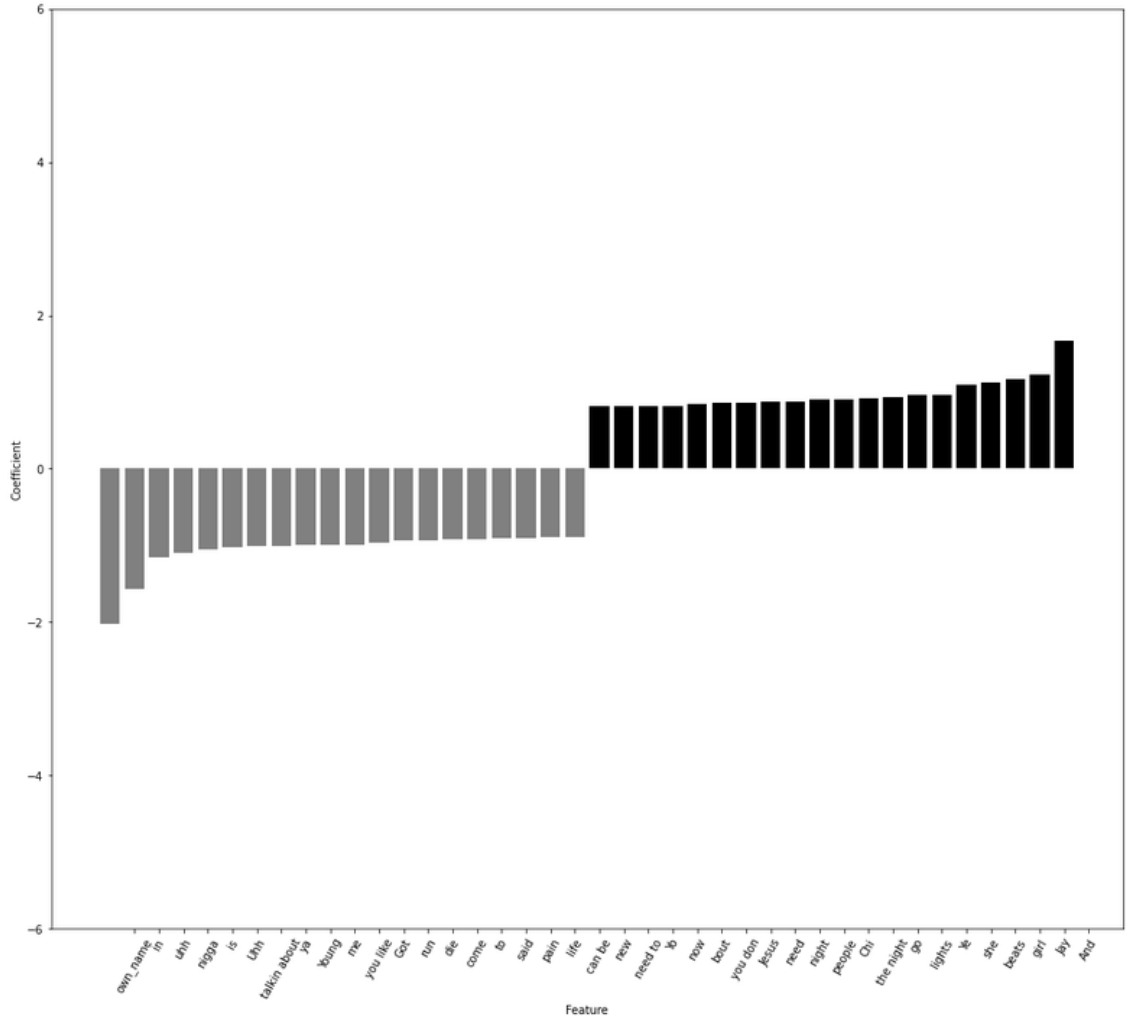

Figure 9: Most informative word n-grams of CADS.a. Grey resembles Jay-Z, black Kanye West

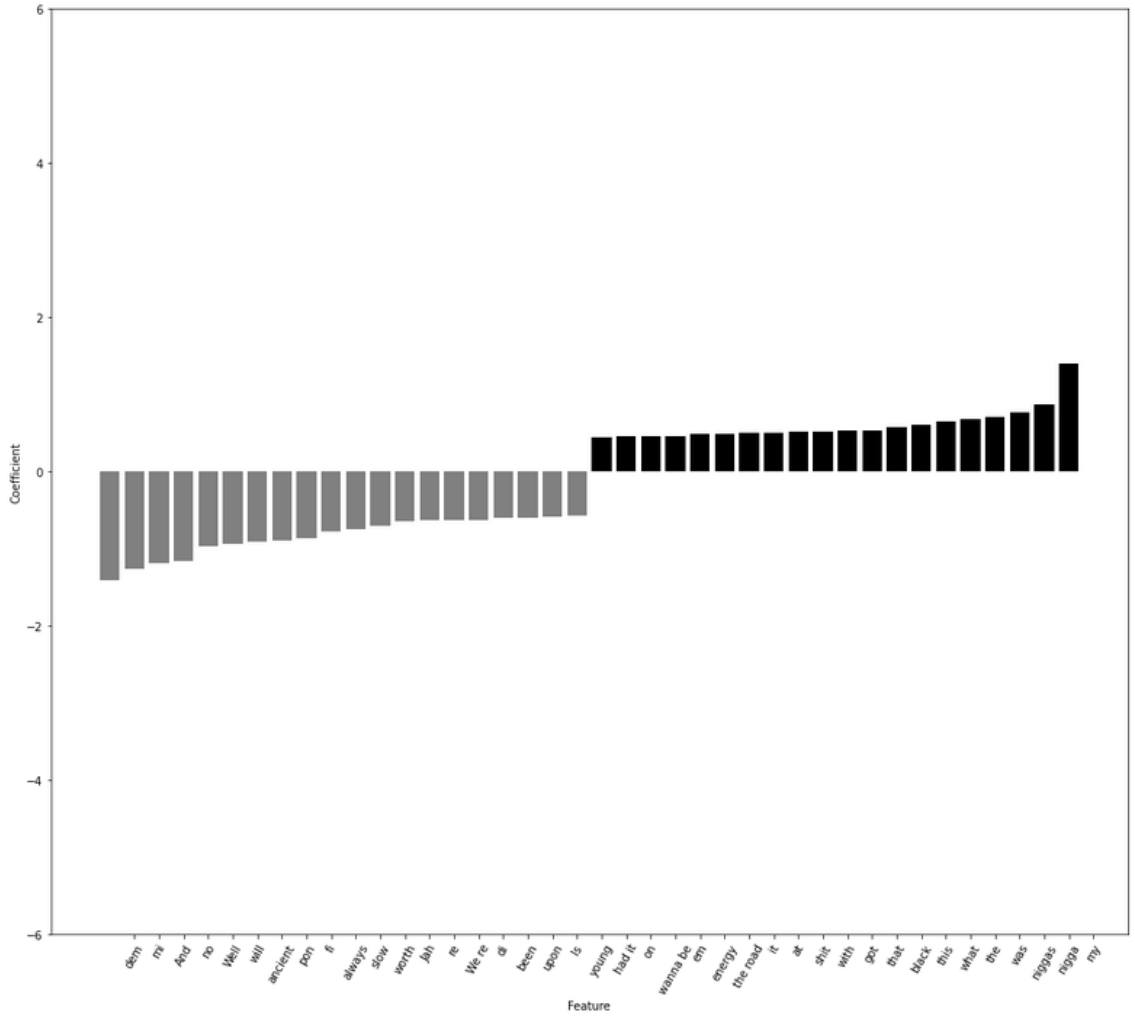

Figure 10: Most informative word n-grams of CADS.b. Grey resembles Damian Marley, black Nas.

C RESULTS

C.1 CLASSIFICATION REPORTS

Artist	Precision	Recall	f1-score	Support
2Pac	0.89	1.00	0.94	8
50 Cent	0.94	0.88	0.91	17
Future	0.94	0.88	0.91	17
Ice Cube	0.88	1.00	0.93	14
JAY-Z	0.92	0.86	0.89	14
Lil Wayne	0.93	0.97	0.95	40
Nas	0.88	0.88	0.88	16
Snoop Dogg	0.85	0.73	0.79	15
accuracy			0.91	141
macro avg	0.90	0.90	0.90	141
weighted avg	0.91	0.91	0.91	141

Table 16: Classification rapport of AAMDS+.

Artist	Precision	Recall	f1-score	Support
Eminem	0.87	1.00	0.93	13
Future	0.96	0.96	0.96	26
Ice Cube	0.71	0.91	0.80	11
JAY-Z	0.75	0.56	0.64	16
Lil' Kim	0.67	0.67	0.67	3
MGK	1.00	0.85	0.92	13
Nas	0.65	0.79	0.71	14
Nicki Minaj	0.83	0.62	0.71	8
accuracy			0.83	104
macro avg	0.80	0.79	0.79	104
weighted avg	0.84	0.83	0.82	104

Table 17: Classification rapport of DADS+.

Gender	Precision	Recall	f1-score	Support
female	0.84	0.84	0.84	37
male	0.84	0.84	0.84	37
accuracy			0.84	74
macro avg	0.84	0.84	0.84	74
weighted avg	0.84	0.84	0.84	74

Table 18: Classification rapport of GBDS.

Artist	Precision		Recall		f1-score		Support
	S&V	V	S&V	V	S&V	V	
JAY-Z	0.73	0.69	0.73	0.82	0.73	0.75	11
Kanye West	0.84	0.88	0.84	0.79	0.84	0.83	19
accuracy					0.80	0.80	30
macro avg	0.78	0.79	0.78	0.80	0.78	0.79	30
weighted avg	0.80	0.81	0.80	0.80	0.80	0.80	30

Table 19: Classification rapport of CADS.a, trained on songs and verses (S&V) or only on verses (V).

Artist	Precision		Recall		f1-score		Support
	S&V	V	S&V	V	S&V	V	
Damian Marley	0.90	0.86	0.93	0.90	0.92	0.88	30
Nas	0.86	0.89	0.85	0.90	0.87	0.88	20
accuracy					0.90	0.90	50
macro avg	0.90	0.89	0.89	0.90	0.89	0.90	50
weighted avg	0.90	0.90	0.90	0.90	0.90	0.90	50

Table 20: Classification rapport of CADS.b, trained on songs and verses (S&V) or only on verses (V).

C.2 CONFUSION MATRICES

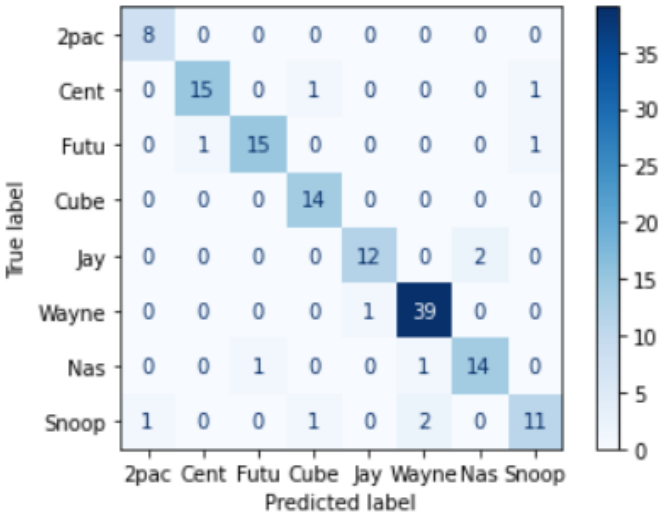

Figure 11: Confusion matrix of AAMDS+.

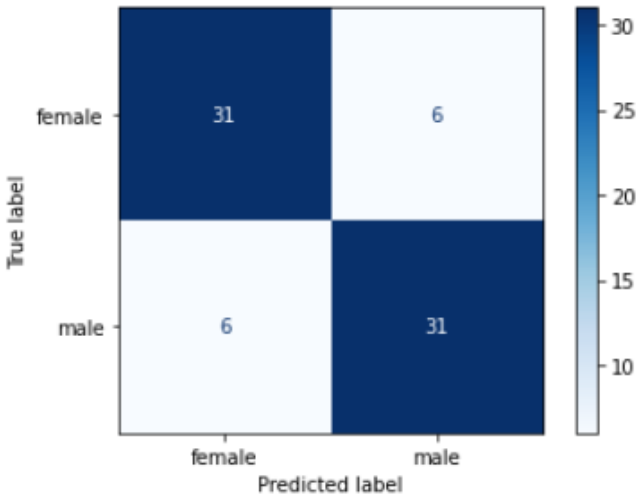

Figure 12: Confusion matrix of GBDS.