University of Utrecht

Humanities

Language Identification in French Afro-Trap

The challenge of code-switching for automated language identification

7.5 ECTS Bachelor thesis BSc Artificial Intelligence

Author:
Cyril de Kock

First assessor:
Frans Adriaans

Second assessor:
Stella Donker

June 27, 2019

Contents

1 Introduction

Language identification (LI) is the art of using computational methods to automatically detect the language of a given text. This technology is used in applications collecting data of which the language is not known beforehand. An example of this would be researchers employing an automated tool to gather Spanish data from the web. Another service that uses LI are machine translation methods like Google's translate which tries to determine the language you want to translate from as you type (Lui, Lau, & Baldwin, 2014).

Most LI tools expect monolingual input. However, most people in the world speak more than one language. Conversations, media and music are often a blend of languages which continuously mix and interact. These instances where people alternate between languages are called code-switches or code-mixes. This is reflected in the worlds data which is for a large part multilingual. This poses a problem to most LI methods as they assume each document in their input to be monolingual and produce only a single language output per document. There is, for this reason, a need for tools which can make the distinction between different languages. Linguistics researchers trying to collect corpora of low resource languages often have to deal with the data they seek being mixed with a more prevalent language such as English (Jauhiainen, Lui, Zampieri, Baldwin, & Lindén, 2018). Low resource languages are defined as languages of which data nor descriptive information is not widely available. Multilingual LI would allow these researchers to automatically collect the vast amounts of data they require.

Code-switching and code-mixing have been extensively studied in the context of psycholinguistics and sociolinguistics (Milroy & Muysken, 1995; Gardner-Chloros, 2009) but research on code-switched data using language technology has only started in the past decade (Solorio & Liu, 2008b). Some of these studies focus on LI itself while others use it as a tool to annotate words with grammatical labels named part-of-speech (POS) tags. Research in LI is mostly focused on natural language processing and machine learning. Algorithms are taught to differentiate between languages using labeled data and then tested on data they have not seen before. This form of learning is called supervised learning and is the most prominent in LI research.

Most of these algorithms function as classifiers. These are mathematical functions designed to map values to a certain class. In the field of LI these values could be, for example, word characteristics and the class a language. These values are called features and bundled together in a vector named the feature set. These features allow a classifier to gather information on the relations between them and the correct class a described item belongs to. A classifier is trained by learning the relations between these features and the classes they indicate. Subsequently, a classifier can use this information to work on new data.

Computational studies on code-switching often focus on bilingual data using high resource languages such as Spanish, Hindi and English (Vyas, Gella, Sharma, Bali, & Choudhury, 2014). Tools for low resource languages are rare as dealing which such data is inherently problematic. There is little to no labeled data available to use in training and dictionary based look-up methods often are not an option. Despite this, low resource languages are especially interesting to the problem of LI as often dealing with such data inherently requires one to tackle the problem of code-switching. This is due to the fact that speakers of these low resource languages often code-switch to a lingua franca (Piergallini, Shirvani, Gautam, & Chouikha, 2016).

Building a model that can identify such low resource languages could give new insights into what features are useful when training a classifier to model code-switched instances of such data. These features can then be used in similar LI tasks. The resulting models can be applied in domains such as corpus building, machine translation and in POS tagging code-switched texts in which the language is often passed as a feature (Jamatia, Gambäck, & Das, 2015).

1.1 Aims

This study will tackle the problem of LI in code-switched data with low resource languages. The aim is to build a classifier that can identify all the different languages present in the data. To train the classifier I will use supervised learning methods. Using these methods will require assembling a corpus of annotated data. Annotating data requires a lot of effort but still saves time compared to unsupervised learning. The latter doesn't require annotations but implementing the correct methods and tweaking their parameters still requires more time than the scope of this study can afford.

The linguistic domain of choice will be Afro-trap, a subdivision of French rap where many languages blend together. The code-switched nature of the genre and the lack of resources for many of the languages used make it a fitting choice of data. To make use of the data I will first determine what languages are present in the data and what annotations are required to mark those languages. Subsequently each word in the data will have to be annotated with a matching language tag.

To properly classify the languages in any data set a classifier requires a set of features describing each token in the data. Features can be designed using the context a word appears in, but also just the to be classified word itself. I will explore both approaches and strive to identify what features are useful in the task of identifying languages for both of these tasks. This will be accomplished by applying successful features from previous literature on the subject in conjunction with my own features specific to this problem.

A lot of different approaches exist when it comes to classifying data. Different classifiers each divide data into categories in their own way. There is no exact way of determining which classifier is best for a particular problem. Often studies employ empirical tests to determine which algorithm suits their problem (Sequiera, Choudhury, & Bali, 2015). After finding a suitable feature set I will make the comparison between different classifiers to determine which suits the problem of LI best.

1.2 Overview

The next section provides some background for the choice of data and will examine existing work regarding code-switching and LI. Section 3 will report on the process of building a corpus of lyrics suitable as data for this study. Section 4 will subsequently report on the linguistic categories and annotations they require. In section 5 I will discuss different feature sets potentially useful to a classifier in section 5 and report the results of experimentation in section 6. Concluding, section 7 will summarise the results and section 8 will reflect on the research process while making suggestions for future work.

2 Background & Related Work

Afro-Trap is a musical phenomenon combining Afrobeat, trap and French hiphop (Hammou & Simon, 2018). The genre was pioneered by Mohammed Sylla (MHD) in 2015 after he went viral releasing a freestyle rap to a tune by the Nigerian band P-square (ARTE, 2016). MHD integrated the American influence of trap which had been rising since the early 2000's with the Ivory coast's popular genre Coupé-Décalé.

Inspired by street and football culture, the genre combines American and African influences and texts with a French lyrical base, leading to a diverse lexicon and song lyrics that contain an abundance of slang and code-switching. Rappers in the genre often hold warm feelings towards their respective African roots and use this to colour and enrich their music (Mancioday, 2012). For example:

La vie na ngai, Mma vie à moi

La vie na ngai, Nzambe nde ayebi

My life, this life of mine
My life, only god knows

I will explore the variety of languages used to code-switch to section 3. First I will elaborate on the multiple reasons for picking Afro-Trap as a source of data. The first reason is the multilingual nature of the genre and the many instances of code-switching in the lyrics as a result. Second is the convenient documentation that song text databases such as Genius provide for music which makes it easy to select relevant data from such websites.

The third reason satisfies the criterium of working with low resource data. African languages and music are underrepresented on the internet with even famous musicians not having all of their music properly documented. The same applies to Afro-Trap where songs with code-switches to any African languages are often either partially documented or not at all. One other cause for this is the fact that a lot of rappers are still working from the Parisian underground and do not have widespread following willing to transcribe or upload their lyrics.

Another reason for choosing Afro-Trap as a source was the familiarity of the author with French and English respectively. This saves a lot of time confirming the origin of words when annotating. Finally, using this kind of data over the more popular and standard social media data sets such as tweets is that while social media data is often bilingual it does not regularly provide the latitude of languages this study requires.

2.1 Code-switching

There is no clear terminology when discussing alternation between languages yet. Studies in the field of linguistics often differentiate between code-switching and code-mixing but no agreement has been reached yet. Code-switching has been defined as the mixing of words and phrases from two grammatical systems across sentence boundaries and code-mixing is the mixing of words and affixes into the structure of another language and which requires participants to reconcile their hearing and recognition (Bokamba, 1989). However, code-mixing has also been defined as intrasentential code-switching (Poplack, 1980) and often the terms are used interchangeably. Another important differentiation includes interword code-switching. This often occurs at morpheme boundaries and more often than not creates compounds of different tokens (Hosain, 2014).

In the rest of this particular study I will refer to the mixing of languages as code-switching and differentiate between intersentential, intrasentential and inter-word code-switching whenever necessary. Various rules regarding the use of code-switching such as the free morpheme constraint and the equivalence constraint (Berk-Seligson, 1986) exist but due to the absence of POS-tags in the corpus I will not be able to make use of such grammatical constraints. This study makes the distinction between identifying languages due to code-switching and identifying languages simply due to different languages being present in a document (Chittaranjan, Vyas, Bali, & Choudhury, 2014). My data should be the former. The latter would allow creating an LI model but would not prepare the model for intrasentential code-switching.

2.2 Automated processing of code-switched data

The reason for using a classifier when modelling such data is that music and language are constantly evolving. Describing the problem with a set of rules might work for a specific instance but would not deal well with variations in the data. Different studies have experimented with and shown successful processing of code-switched data using machine learning techniques (Solorio & Liu, 2008a; Sequiera et al., 2015; Barman, Wagner, Chrupała, & Foster, 2014).

A widely used classifier in LI research is the Naive Bayes (NB) classifier.The features are passed as a vector of values to the classifier. The NB then uses each of the provided features as a probability indicating a language. The probabilities are multiplied for each language and the highest probability language is chosen (Jauhiainen et al., 2018). NB is often used because it is simple to implement and has quick processing times. Despite the simple approach, it proves to be quite effective in the domain of LI with Tan et al. (2014) obtaining 99.97% accuracy on a 6-language data set.

Multiple different implementations of NB exist. Bernoulli NB (BNB) models represent features as binary inputs marking whether a feature applies to a word or not. Multinomial NB (MNB) models use frequencies as features and model each language to be classified as samples drawn from a multinomial distribution (Giwa, 2016). Juahiainen et al. (2018) survied a vast collection of LI research and found no studies using a Bernoulli model. This is most likely due to previous research showing both regular and multinomial models to be more effective (McCallum, Nigam, et al., 1998; Eyheramendy, Lewis, & Madigan, 2003). Other classification algorithms like Logistic Regression (LG) (Acs, Grad-Gyenge, & de Rezende Oliveira, 2015) and (Linear) Support Vector Machines (SVM) have been shown to be effective for LI as well (Kim & Park, 2007).

LI can be approached from multiple levels. Document-level classification is concerned with assigning a label to a collection of text. An example of this is classifying the language of tweets (Lui & Baldwin, 2014). Lower level classification includes tagging sentences and words. As code-switching can occur intrasententially this study is only concerned with word-level annotations.

Word-level classification has two distinct approaches in itself which are both necessary to develop adequate models. Words can be identified using the context of the structures they appear in or without. The former is desirable as it has shown features based on context improve word-level classification (Barman, Das, Wagner, & Foster, 2014; Vyas et al., 2014; Nguyen & Doğruöz, 2013; Piergallini et al., 2016). However, often context is a luxury. An example would be the automatic identification of languages in Google's translation service. Users often wish to translate just one word which makes the algorithms rely on just the word itself to identify its language.

3 Building the Corpus

I collected a corpus of data to train and test the classifiers. The corpus consisted of French Afro-Trap lyrics taken from a list of candidate songs composed beforehand. The data was scraped from three different websites with Genius as the main source. Utrecht University Research Data Management Support was contacted after concerns were raised about whether such data collection was legal due to lyrics being copyrighted material. In compliance with their policies, the data were collected for educational purposes only and will not be distributed. A list of all songs in the corpus is included in the appendix. All examples used in this study are taken from this list.

Not all songs from the candidate list had their lyrics transcribed online due to the artists being little known and not publishing the lyrics themselves. This proved to be a obstacle in providing a diverse corpus. In total the corpus contains 46 songs. 26 of these have MHD as the sole performing artist. Figure 1 visualises the distribution of performing artist in the corpus. Each section represents the artists performing. The red sections include one song per artist, the green sections two and the purple one includes the 26 by MHD. This lack of inter-artist diversity forms no problem for any of the experiments using classifiers. Although it may prevent the model from generalising to other data.

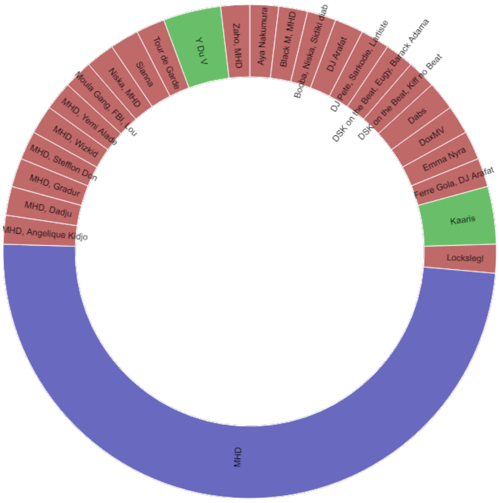

Figure 1: Artist distribution in the corpus

3.1 Cleaning

Before starting the task of annotation some restructuring of the data had to be conducted. Some lines were not transcribed fully with missing text marked as '[?]'. All of these instances were removed from the corpus as incomplete lines could skew results when exploiting context. For example:

Étant tit-pe je mangeais les [?]

When I was small I ate the [?]

Similarly, artists often use echoes or (vocal) background sounds to add an extra layer to their music. Such vocal instances were transcribed as 'line (echo)' in the data. These instances were split onto a new line with the parentheses used to indicate the echo or background vocals being removed. For example:

Ça va aller (ça va aller)

It will be okay (It will be okay)

Resulting in:

Ça va aller
Ça va aller

The corpus was cleaned of any lines indicating song structure. For example:

[Intro : Sidiki Diabaté & Niska]

Numbers in the corpus were mostly written numerically and thus give no indication of which language they are pronounced in. Songs that did contain not fully spelled out numbers were all individually checked and the numbers were transcribed in their respective language. One song was removed from the data due to the fact that all online audio copies of the the song were deleted and thus the spelling of numbers used couldn't be checked. The reason for this remains unknown. French numbers were written using the revised 1990 spelling rule dictating hyphens should connect all numbers. Time indications in French are often written as *number+H* with the number indicating the particular time and the *H* indicating the word *heures*. For example *dix heures*. Instances like these were written out in full as well. Lastly all punctuation was removed except for the apostrophe and the hyphen as those are key to French word structure. This normalised the data to one format as not all sources of lyrics used the same punctuation standards.

4 Annotation

In this section I will explore the different linguistic categories present in the corpus, defining each category by its characteristics. The aim is to provide an overview of what the corpus looks like and what annotations are required for classification.

4.1 Linguistic categories

There have been studies defining French hiphop as containing four main linguistic categories. Most define the genre as a blend of French, Verlan, English and Arabic (Hassa, 2010). Verlan is a form of speech play were parts of a word are inverted (Lefkowitz, 1989). Others have taken a different approach and put Verlan into the broader category of Argot or French slang (Paine, 2012). Argot is the French term for all language that is considered to be non-standard. Afro-Trap adds to the multi-dimensionality of French hip-hop by introducing a whole new category of African languages. This section will explore these categories one by one.

4.2 English

American hip-hop artists are by far the most well-known and their influence on Afro-Trap is clearly visible. Trap music originates from the mid 2000' United States and its sound laid part of the foundation for the Afro-Trap genre. ARTE described Afro-Trap it as *"the music of a generation raised between the sounds of Africa and America"* (2016). This is demonstrated in the thematic exploration of the disconnect between these rappers and the French Republic whom they feel ignores their presence and often works against them. Rap has become their medium to voice their criticism about a system they see as discriminatory. This is reflective of American rap music which is very critical about the systematic racism which works against them (Sarkar, Winer, & Sarkar, 2005). The other way in which English manifests itself is when Afro-Trap rappers draw on their multilingual culture and using lots of English insertions and alterations as well as complete English verses in their lyrics.

4.3 French slang

This section examines French slang, often named Argot. It has been used with the same definition in other languages as well but in this study I will be using it to describe solely non-standard French language. Rough and vulgar language is often grouped under this label as well as the dropping of word parts such as:

Et j'suis le meilleur comme d'hab.

And as usual I am number one

Where *d'hab* is replacing *d'habitude*. A lot of code-switching is labeled as Argot as well but this study will look at these instances in the light of their own respective language. Although this is an indicator of the limitations of trying to annotate text with singular linguistic categories as all of these categories overlap and interact with one another. This is nevertheless required for the task of classification and the aim is to make these categories encapsulate enough information without being too restrictive. The most prevalent subcategory of Argot in the corpus is perhaps Verlan. As previously stated, Verlan is the art of inverting words. Subsequently sounds can be added to these inverted words or final vowels can be dropped (Jamin, 1998). Examples of this are:

On est venu pour faire la te-fê on pose les mes-ar on lève les rres-ve.

Where *fête* is inverted to *te-fê (party)*, *armes* to *mes-ar (weapons)* and *verres* to *rres-ve (drinks)*.

4.4 Niger-Congo

As mentioned in the introduction, the characteristic feature of Afro-Trap is the code-switching to a variety of African languages. There is a lot of variation between songs and between artists as languages used often represent each artists origins. Manual survey of a collection of Afro-Trap songs revealed the presence of following languages: Soninké, Fon-Gbe, Diakhanké, Bambara, Chichewa, Nyanja, Lingala, Swahili and Yoruba. All of these languages fall under the Niger Congo language family. This language family is by estimate the largest phylum in the world and its reach covers many of the former French colonies including Niger, Ivory Coast and Senegal. It is characterised by similar noun class systems, verbal extensions and a universal basic lexicon (Williamson & Blench, 2000). Afro-Trap rappers use their respective languages of origin to enrich their music by code-switching:

Laisse-les kouma la mala c'est pour nous

Let them talk the money is ours

With *kouma* being Diakhanké for talking. Sometimes entire verses are included:

Wanyinyin de ndo nou we o
Wzon gbe tche yon de kpe lo
Wanyinyin de ndo nou we o
Wzon gbe tche yon de kpe lo

It's the love I have for you
That fills my life with beauty
It's the love I have for you
That fills my life with beauty

It cannot be confirmed that all rappers fully speak the languages they are using though it is suspected they do not as often when complete verses in one of these languages are realised this is done by a guest role, such is the case in the previous example. As each rapper or supporting artist has a different cultural heritage none of these languages vastly outnumber the others.

4.5 Spanish

Spanish is not frequently seen in the music analysed but occurs often enough to justify its own category. Basic Spanish phrases like *vida, hola* and *adiós* appear with moderate frequency in the releases of different artists. Just as American hiphop has Mexican and Puerto Rican influences it is suspected this is simply the influence from Spain as a neighbour and the prevalence of Spanish in popular music.

4.6 Arabic

Arabic in not overly present in Afro-Trap for most artists use no more than a couple phrases with most of them being inserted in mostly French sentences. For example:

Toujours en activité wallay billay j'connais pas la grève

Always active by Allah by Allah don't know how to rest

These may just be particular words appearing as part of Argot. However, it could also be that these have a deeper connection to the roots of the artists. On the subject of Arabic code-switching in French rap Hassa (2010) wrote:

"The use of Arabic in French rap suggests an identification with North African community".

While their frequency may be relatively low these words will be considered as a category of their own.

4.7 Summary

If all languages in the corpus were to be annotated individually this would result in a set of at least twelve different annotations excluding any additional ones for names, words of unknown origin and any other language not yet accounted for. Working with language categories is more productive as this will allow me to make the best of the data and working with many low-frequency languages would not allow fitting a proper model.

This study will consider all of the languages falling under the Niger-Congo language family a one "linguistic category". The reasons are manifold, first there are not simply not enough tokens in the available data per language for proper classification. One way of working around this problem is grouping them together. This is done with the knowledge of these languages sharing a basic lexicon and the code-switching structures in which their respective words appear being similar. This basic lexicon is a problem of its own as the origin of some words can be traced to multiple languages. For example the word *kouma (talking)* can be traced back to Diakhanké as well as Bambara. A lack of resources makes it difficult if not impossible to reach consensus on which annotation is correct.

Furthermore the corpus requires annotations for standard French, French slang, English, Spanish and Arabic. All of the Arabic included is written in the Roman alphabet so no transliteration is required. The reason French slang is annotated separately from standard French is an experimental one. French slang does differ from standard French linguistically as mentioned in section 4.2 and classifiers have been demonstrated to be able to separate language varieties (Zampieri, Gebre, & Diwersy, 2013). Another reason for making the distinction is that it defines another language category in the corpus. Furthermore it reduces the relative frequency of French providing additional challenge when classifying. It also makes standard French a less muddled category allowing more accurate dictionary look-up. This will be further discussed in section 6.

Finally, named entities are not considered part of any language group and thus need a separate tag. Examples of named entities include people, countries and brands. Non-lexical sounds such as *wooh*, *ounga* and *paw* are often used in hip-hop and are not considered part of a language category in this study. Instead they are put in a category of their own. Any tokens whose meaning cannot be ascertained will be annotated as unknown and removed from the corpus before classification.

4.8 Annotating the corpus

Previously, I identified the characteristic linguistic groups in Afro-Trap and defined a set of tags to accurately categorise these groups and separate them from one another an any other data. This required not only to differentiate those groups but also to account for the irregularities of musical data. The following tag-set was defined before starting the task of annotation:

- **FRA:** tag for standard French.
- **FRS:** tag used to mark any word defined as part of Argot.
- **ENG:** tag used for any English words corpus.
- **AFR:** tag used for any words in the Niger-Congo family of languages.
- **ARAB:** tag used for any Arabic words in the corpus.
- **ESP:** tag used for any Spanish words in the corpus.
- **NAME:** tag for named entities.
- **NLEX:** tag used for any non-lexical vocables in the corpus.
- **UNK**: tag used to define words would meaning couldn't be ascertained.

All non- standard French words were annotated manually. Any words whose meaning or language category was unclear were looked up using online dictionaries. Whenever phonetic transcriptions were provided they were used to confirm the language of a given word. Additional tags were added when encountering Jamaican Patois (**CAR**), Corsican (**COR**), Dutch (**NED**) and Italian (**ITA**) respectively. Stop words were annotated according to their context, thus if a given stop word appeared in a English sentence it would be annotated as such. Once the annotation process was completed all remaining words were annotated as French automatically using regular expressions.

After the annotation process was completed all lines containing either unknown or too low frequency tags were removed. These were all the tags added in the process of annotation with each appearing no more than 10 times. A total of 103 lines were removed. The remaining corpus consists of 3002 lines and 19904 words. 1517 of these lines are monolingual French which means about half of the corpus consists of code-switched sentences. It should be noted that this excludes sentences in which words are either tagged with NLEX or NAME. These sentences would not be considered code-switching when speaking in linguistic terms but will be seen as such by a classifier. Table 1 displays the tags present in the corpus, their frequency of appearance and their frequency as a percentage of the whole tag set.

Tag	Frequency	Percentage
FRA	16105	80.88%
FRS	516	2.59%
NAME	892	4.48%
AFR	894	4.49%
ENG	1200	6.03%
ARAB	69	0.35%
ESP	47	.24%
NLEX	181	0.91%

Table 1: The tags present in the corpus, their total amount and their relative frequency as a percentage of the whole set

5 Classification

This study models LI task as a classification task. All data in the corpus was annotated to train a classifier using supervised learning. Supervised learning works by showing a classifier a series of labeled examples called training data. The training data for each given input consists of all features of the given input and the correct annotation. The features describe the information provided to the classifier. These are usually comprised of booleans, integers, strings and more complex types consisting of a collection of such values. In this case all these values describe a particular word.

Designing features only describing the word at hand is called 'classification without context'. However, in the corpus words very rarely occur on their own. They appear in lines with a lot of other information hidden in the words preceding and following them. One would expect exploitation of context to provide better results and often this is the case (Barman, Das, et al., 2014; Piergallini et al., 2016). Not all contextual features are helpful and too many features can prove counterproductive (Chittaranjan et al., 2014). Another source of information is the annotations themselves. These can be used as an additional contextual resource when training a classifier.

In this study I will address the problem of word identification in two different ways. The reasoning behind this is discussed in section 2.2. I will test different combinations of feature sets for both classification with and without exploitation of context.

For both tasks multiple sets of features are defined and evaluated. The NLTK and scikit-learn (Pedregosa et al., 2011) Python libraries provide a vast array of different classifiers, evaluation metrics and other resources to tackle the problem at hand. Accuracy will be used as evaluation metric to compare different classifiers and a confusion matrix is employed to highlight the flaws and strengths of the model.

To ensure accurate results I used ten-fold cross validation when training and testing each classifier. n-fold cross validation splits the data in n folds and then uses $n-1$ of these for training and saves the remaining fold for testing. This is done n times and each time a different fold is left out for testing. This reduces variability caused by simply slicing the data in a training and test set.

5.1 Word-based features

Previous literature has explored different feature sets which are effective at LI (Barman, Das, et al., 2014; Barman, Wagner, et al., 2014). N-grams, dictionary presence, word length and capitalisation features have all proven to be resourceful. N-grams are a tried and tested approach for LI and n-grams (n = 1 to 5) plus the word itself have proven to be the most informative (King & Abney, 2013). As such this study will not try to find the optimal n-gram range. However, not all studies agree on including the word itself as part of the n-gram features (Piergallini et al., 2016) and thus it will be a feature set of its own. The following sets of features were used, each set is marked with a letter for identification in result tables:

1. **N-grams (N):** As mentioned n-grams (n = 1 to 5) are used as features.
2. **Word (W):** The lowercase word is used as feature. This is done with the purpose of treating words such as *bonjour* and *Bonjour* the same.
3. **Length of a word (L):** Raw word length is used as a measure. Other measures have been experimented with (Wagner et al., 2014) but due to the limited scope of this study I was not able to implement such features.
4. **Capitalisation (C):** The same three standard boolean features for capitalisation as Barman et al. (2014) are used. These indicate whether the first letter of a word is capitalised, whether all letters of a word are capitalised and whether any letter in a word

is capitalised. These features are employed to aid in the recognition of names in the corpus.

5. **Presence in Dictionaries (D):** The Py-enchant module is used to access online dictionaries and check them for the presence of a word. The module includes dictionaries for both English and French. Boolean features indicate whether a word is present in a dictionary or not. A feature was added to indicate whether all individual words in a French compound were present in the French dictionary. 86.91% of the corpus consists of French and English words dictionary look-up is expected to achieve accuracy proportional the relative frequency of these languages.

6. **Word characteristics (F):** Suffixes and affixes of up to $n = 3$ are passed as features. This feature is never used in combination with n-grams as it is a subset of the information n-grams provide.

7. **Punctuation (P):** French specifically uses a great amount of apostrophes and hyphens to combine words into single tokens. Boolean features indicate whether either is present in a word.

5.2 Context-based features

The following features were employed to exploit the context of a sentence:

1. **Index (I):** The position of a word in a given line is passed.

2. **Context words (C_1):** The n words ($n = -3$ up to 3, $n \neq 0$) surrounding a given word are passed as individual feature values. If a word is the first in it is line the $n-1$ value is passed as <START> and all the others down to $n = -3$ are passed as empty strings. Similarly if the word is the last in the line the $n+1$ value is passed as <END> and the others up to $n = 3$ as empty strings.

3. **Context n-grams (C_2):** Following the work of Barman et al. (2014) the previous and next token are each combined with the current word, generating two sets of n-grams. It was experimentally shown this is the optimal combination and as such no further experimentation will be done in this study.

4. **History (H):** As mentioned before, the annotations of any words appearing earlier in a line can be combined to form a history of annotations. These previous classifications can subsequently be used to annotate any future tokens in the sequence. This is known as greedy sequence classification (Bird, Klein, & Loper, 2009). This study uses the entire annotation history of the line a word occurs in thus far.

5.3 Classifiers

I used a NB classifier for the comparison of different feature sets. The reasoning behind this choice is twofold. Multiple studies have shows NB to be adept at language classification (Zampieri et al., 2014; Bhattu & Ravi, 2015) and it is far more rapid than any other algorithm. This combination of efficacy and efficiency is ideally suited to the comparison of all possible combinations of features.

After determining the feature set best suitable to this problem I will compare four different classifiers against Naive Bayes to determine which performs best. These are the algorithms discussed in section 2.2. MNB, BNB, LG and a Linear SVM. LG predicts the likelihood of input belonging to a certain class by using a logistic function to transform the linear input into a probability. In this case each input will have a probability indicating its likelihood of belonging to any of the language groups in the corpus.

SVM's separate two classes by drawing an imaginary field in the vector-space. While designed for binary classification, they can be used on multi-dimensional problems as well. This is done by projecting input data onto a high-dimensional inner product space. This is convenient as SVM's have proven to be effective at LI (Kim & Park, 2007; Barman, Wagner, et al., 2014; Barman, Das, et al., 2014). I use a Linear SVM as this is the predominant approach in LI (Jauhiainen et al., 2018).

6 Results

First I tested combinations of feature sets in classification without context. Then the best resulting feature set was used as baseline when comparing contextual features as per example of Barman at el. (Barman, Wagner, et al., 2014; Barman, Das, et al., 2014). While they proved the use of non-contextual features as baseline to be an effective strategy, it did not generalise to this instance. Previous literature has demonstrated this can occur (Nguyen & Doğruöz, 2013) and I change course by adding non-contextual features that were left out of the baseline set. As a result accuracy scores improve.

6.1 Classification without exploiting context

Table 2 displays the accuracy measures for all different combinations of feature sets used in training. Inspired by previous research I start using either n-grams or the word itself as a base measure and then test all combinations of the base layer + *n* features. Surprisingly the combination of word and n-grams results in the worst score out every feature set, despite this combination being effective in previous LI research. The word itself seems to be the most effective feature as the top five feature sets are in increasing order: **N**, **W**, **WD**, **WC**, **WPC**. Adding any additional features to an n-gram base did not improve performance in any case. I proceed with **WPC** as the baseline set in future evaluations.

Features	Accuracy	Features	Accuracy	Features	Accuracy	Features	Accuracy
N	94.4%	**W**	94.5%	**WLP**	93.7%	**WDF**	92.3%
NP	94.1%	**ND**	92.3%	**WLF**	92.2%	**WPF**	92.2%
NL	93.2%	**NC**	93.8%	**WFLP**	92.3%	**NCDP**	93.8%
NW	87.8%	**WL**	93.5%	**NDLP**	93.2%%	**NCDL**	93.3%
WD	94.6%	**WP**	94.3%	**NCLP**	93.3%	**WCDP**	93.2%
WC	94.6%	**WF**	92.2%	**WDLP**	93.2%	**WCDL**	92.9%
NLP	93.2%	**NDL**	91.9%	**WCLP**	94.4%	**WCDF**	93.5%
NDP	94.1%	**NCD**	92.6%	**WDFP**	93.7%	**WDFL**	93.6%
NPC	92.3%	**NCL**	93.3%	**WCFP**	91.8%	**WCFL**	92%
WCF	91.8%	**WDP**	92.3%	**NCDLP**	92.2%	**WCDLF**	93.5%
WDC	93.2%	**WDL**	92.1%	**WCFLP**	92%	**WCDLP**	92.6%
WPC	94.7%	**WCL**	94.3%	**WCDFP**	93.6%	**WDFLP**	93.6%

Table 2: The average classifcation scores per feature set using only non-contextual features.

6.2 Classification exploiting context

To keep results tables clear the best feature set from the previous classification task **WPC** is written as **B** in any subsequent tables. Table 3 displays the scores of this baseline set combined with features exploiting the context of a word. The reason the baseline set has a different accuracy score than in the previous section is the different structure of the data. To make use of context, combinations of a line and an index are passed to the classifier with the index indicating the position of a particular word in the line. This is done in sequence for every word in a line. In the last section all words were extrapolated from their lines, scrambled and then passed to the classifier. When not using contextual features the latter produces a higher accuracy. This is most likely due to the fact that the probability of appearance during training of words repeated in a line is higher when scrambling.

Previous research on LI has shown exploitation of context to provide effective features in classification (Barman, Das, et al., 2014; Piergallini et al., 2016) and that features effective in non-contextual classification can be used as a baseline when considering the context of a word. This did not apply to this case as accuracy dropped for every combination of features and dipped below dictionary look-up rates in two instances. No set of features performed better than the baseline set. To correct this strategy the word-based features are combined with other non-contextual features in an attempt to produce a better model.

Feature sets	Accuracy	Feature sets	Accuracy
B	93.9%	**BI**	93.9%
$\mathbf{BC_1}$	93.4%	$\mathbf{BC_2}$	85.4%
$\mathbf{BC_1I}$	89.7%	$\mathbf{BC_2I}$	84.9%
$\mathbf{BC_1C_2}$	89.9%	$\mathbf{BC_1C_2I}$	89.7%
BH	93.2%	**BIH**	93.0%
$\mathbf{BC_1H}$	92.4%	$\mathbf{BC_2H}$	82.8%
$\mathbf{BC_1IH}$	89.7%	$\mathbf{BC_2IH}$	82.7%
$\mathbf{BC_1C_2H}$	89.1%	$\mathbf{BC_1C_2IH}$	88.77 %

Table 3: The average classification scores using a Naive Bayes classifier with feature sets exploiting context and the best non-contextual feature set.

Table 4 displays the result of combining the non-contextual features **L**, **D** and **F** with the baseline features and surrounding words. Double decimal digits were used in the table as the differences between scores were much smaller. The addition of the suffix and affix features increased accuracy by two percent. Each additional feature resulted in a slight increase in accuracy. One of these instances could be due to chance but the continuous increase clearly show the usefulness of the added features. Additional checks were performed to investigate whether adding set **H** or removing $\mathbf{C_1}$ helps classification. Employing history as a feature increases accuracy but the context words results in a significant drop in accuracy.

Feature sets	Accuracy	Feature sets	Accuracy
$\mathbf{BC_1F}$	95.56%	$\mathbf{BC_1FL}$	95.62%
$\mathbf{BC_1IFL}$	95.64%	$\mathbf{BC_1DFIL}$	95.96%
$\mathbf{BC_1DFHIL}$	96.12%	**BDFHIL**	93.34%

Table 4: The average classification scores using a Naive Bayes classifier with feature sets exploiting context and additional word-level features.

Table 4 confirms the hypothesis that features based on word context are important when classifying. Table 3 and 4 demonstrate how classification with and without context are different tasks. This indicates generalising features which work well on the latter may not work as well when context is involved as they may not provide a classifier with sufficient information. This could also be the result of the limited data available or my choice of annotations.

Lastly I compared different classifiers using the best feature set $\mathrm{BC_1DFHIL}$. Bernoulli Naive Bayes performed far worse than any other classifier. This was within expectations as no research on LI uses this algorithm. A Linear SVM achieved the highest accuracy out of all classifiers reaching over 98%.

Classifier	**Accuracy**	**Classifier**	**Accuracy**
MNB	94.6%	**BNB**	90.7%
LSVM	98.1%	**LG**	97.1%

Table 5: A comparison of accuracy measures for MNB, BNB, Linear SVM and LG classifiers trained on feature set BC$_1$DFHIL

To help understand the model I generated a confusion matrix from a random run of the Linear SVM. The rows of the matrix represent the correct labels and the columns show the values assigned by the classifier. The classifier performs well across the board except for the English words, which are often wrongly tagged as French. This may be due to peculiarities of this run but could also indicate the classifier has trouble separating the languages based on their linguistic similarities.

	FRA	**ENG**	**NAME**	**AFR**	**FRS**	**ESP**	**NLEX**	**ARAB**
FRA	<1667>	2	2	.	1	.	2	.
ENG	14	<88>	.	.	.	.	.	.
NAME	.	1	<69>	<1>	.	.	2	.
AFR	4	.	.	<64>	2	.	.	.
FRS	4	.	1	.	<38>	.	.	.
ESP	1	.	.	.	.	<4>	.	.
NLEX	.	.	.	.	.	.	<3>	.
ARAB	.	.	.	.	.	.	.	<2>

Table 6: A confusion matrix with the rows representing the reference values and the columns the assignments made by the classifer

7 Conclusion

The aim of this study was to identify languages in a code-switched corpus with low resource languages. The amount of languages and lack of accurate descriptive information required a custom coarse set of annotations to enable accurate classification. Classification was performed in two ways. The first was classification without context, which omits working with the code-switched structure of the data. The second used the structures the different languages appeared in and tried to exploit these to gain information. For both of these I compared different feature sets to determine which were informative. Both ways can be applied in fields such as machine translation, POS-tagging and corpus construction.

Table 2 indicates the best features for word classification without context were the word itself, capitalisation features and booleans indicating use of punctuation in a word. An explanation for the effectiveness of the word itself might be due to the repetition found in musical data. As words occur more often with a specific language tag it makes it easy for a classifier to map the relation between a word and a tag. Capitalisation features aid in the differentiation of names from the rest of the data. The punctuation based features show the importance of domain knowledge when training a classifier. Hyphens and apostrophes are key to grammatical structure of written French and can be effective in separating French and slang variants from the other language groups.

When it comes to exploitation of context, the baseline set resulting from the non-contextual task was used in combination with index, history, n-gram and word based contextual features. These combinations performed poor with scores dipping below dictionary look-up. This poor performance indicates classification with and without context are two different tasks that require different approaches.

Performance improved after adding back features left out from the baseline set. The combination of the baseline set, all contextual features except for n-grams, dictionary look-up, word suffixes and affixes and word length showed to be the most informative to the classifier. This confirmed the work of earlier research asserting context is an important factor when performing LI (Barman, Das, et al., 2014; Piergallini et al., 2016; Jauhiainen et al., 2018). Using n-grams at any point in classification only resulted in worse performance. This is surprising given their usual effectiveness in LI.

Comparison between different classifiers demonstrated a Linear SVM achieves the highest accuracy. This is in line with other LI studies where Linear SVM's have been used with great success.

8 Discussion

One important decision I made in the process of creating my corpus was to group all languages in the Niger-Congo family under one annotation instead of creating a separate tag for each language. This was done due to a lack of data per individual language and a lack of descriptive information per language. This could also be tackled annotating individual languages removing any that are low in frequency or indiscernible. However, further reducing the corpus seemed liked poor practice. A corpus of twenty thousand words is minimalist in the field of computational linguistics and reducing it would affect the ability of the model to generalise to other data.

Another decision was to annotate French and French slang as separate language groups. This was mainly done to introduce more variety in the data. This seems contradictory to the decision to create one tag for the Niger-Congo Language family. Nevertheless these decisions produced the linguistic diversity I sought while retaining the content of the corpus. This contradiction could be taken as a sign that another source of data might be more suitable for this kind of research. Social media data is often used in LI research and relatively simple to gather and may provide more clearly definable language categories.

The major drawback of this study is the lack of data for many of the languages present in the corpus. Aside from those languages grouped in the Niger-Congo family, Spanish and Arabic suffer from a lack of data in the corpus as well. While the Spanish and Arabic lexicon is limited in French hiphop and the model may extrapolate well to new data, it raises the question whether such data is suitable to draw conclusions about LI. Further research may want to either increase the amount of data to achieve proper representation or use a more evenly distributed source of data.

There are multiple other ways in which future research could add or improve upon this study. One improvement would be to use more elaborate evaluation metrics than just accuracy. Standard deviations could measure how consistent feature sets are in the results they generate. Furthermore, precision and recall may be informative measures when working data that is largely biased towards one class as is the case with my data. Precision is a measure of how many instances of a class have correctly been caught. Recall on the other hand captures the amount of misses when classifying. Both would be informative measures to this study and further research.

Another area of improvement concerns annotation. In this study only the author was performing annotation. Despite careful scrutiny and multiple revisions, only one person performing this task can easily lead to mistakes or misinterpretations when researching unknown tokens. It is preferable to use multiple people in the annotation process. This can be done by having the annotators manually check each others work or by having each individual annotate the whole set and then choosing the correct tag based on some measure of agreement.

Further additions could be made in the form of more advanced features. Names will always be a part of data such a music or social media content. Using the output of a Named Entity Recogniser as feature could help in separating these names from languages to be identified. More advanced models could improve results as well. Combining the outputs of multiple classifiers in joint models or using Hidden Markov Models to generate probabilities for sequences of annotations may prove more effective than the simple models used in this study.

Finally, I used off-the-shelf classifiers with no additional tweaking for comparison. However, most classifiers are complicated mathematical functions with many parameters that can be tuned. Discovering which parameters fit the best model is a daunting task and warrants its own study.

To conclude, this study demonstrates that LI requires different approaches dependant on available data. Features that are standard in some applications may prove less informative applied to other domains. The usefulness of language-specific features such as punctuation markers shows the importance of domain knowledge. I recommend only using standard feature sets as a baseline when they have been demonstrated to work for domain of the data used and recommend the use of Linear SVM's as the algorithm of choice.

References

Acs, J., Grad-Gyenge, L., & de Rezende Oliveira, T. B. R. (2015). A two-level classifier for discriminating similar languages. In *Proceedings of the joint workshop on language technology for closely related languages, varieties and dialects* (pp. 73–77).

ARTE, T. (2016, Mar). *Le raz-de-marée afro trap - tracks arte.* YouTube. Retrieved from `https://www.youtube.com/watch?v=2-VuypErtWI`

Barman, U., Das, A., Wagner, J., & Foster, J. (2014). Code mixing: A challenge for language identification in the language of social media. In *Proceedings of the first workshop on computational approaches to code switching* (pp. 13–23).

Barman, U., Wagner, J., Chrupała, G., & Foster, J. (2014). Dcu-uvt: Word-level language classification with code-mixed data. In *Proceedings of the first workshop on computational approaches to code switching* (pp. 127–132).

Berk-Seligson, S. (1986). Linguistic constraints on intrasentential code-switching: A study of spanish/hebrew bilingualism. *Language in society*, *15*(3), 313–348.

Bhattu, S. N., & Ravi, V. (2015). Language identification in mixed script social media text. In *Fire workshops* (pp. 37–39).

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

Bokamba, E. G. (1989). Are there syntactic constraints on code-mixing? *World Englishes*, *8*(3), 277–292.

Chittaranjan, G., Vyas, Y., Bali, K., & Choudhury, M. (2014). Word-level language identification using crf: Code-switching shared task report of msr india system. In *Proceedings of the first workshop on computational approaches to code switching* (pp. 73–79).

Eyheramendy, S., Lewis, D. D., & Madigan, D. (2003). On the naive bayes model for text categorization.

Gardner-Chloros, P. (2009). *Sociolinguistic factors in code-switching.* Cambridge University Press.

Giwa, O. (2016). *Language identification for proper name pronunciation* (Unpublished doctoral dissertation). North-West University (South Africa), Vaal Triangle Campus.

Hammou, K., & Simon, P. (2018). Rap en france et racialisation. *Mouvements*(4), 29–35.

Hassa, S. (2010). Kiff my zikmu: Symbolic dimensions of arabic, english and verlan in french rap texts. *Languages of global hip hop*, 44–66.

Hosain, I. (2014). Code-mixing in the fm radio in bangladesh: A sociolinguistic observation.

Jamatia, A., Gambäck, B., & Das, A. (2015). Part-of-speech tagging for code-mixed english-hindi twitter and facebook chat messages. In *Proceedings of the international conference recent advances in natural language processing* (pp. 239–248).

Jamin, M. (1998, Feb). *Introduction à l'argot: argot et verlan.* The University of Sunderland. Retrieved from `https://eserve.org.uk/tmc/contemp1/argot.htm`

Jauhiainen, T., Lui, M., Zampieri, M., Baldwin, T., & Lindén, K. (2018). Automatic language identification in texts: A survey. *arXiv preprint arXiv:1804.08186*.

Kim, S., & Park, J. (2007). *Automatic detection of character encoding and language* (Tech. Rep.). Technical Report, Machine Learning, Stanford University.

King, B., & Abney, S. (2013). Labeling the languages of words in mixed-language documents using weakly supervised methods. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 1110–1119).

Lefkowitz, N. J. (1989). Verlan: talking backwards in french. *The French Review*, *63*(2), 312–322.

Lui, M., & Baldwin, T. (2014). Accurate language identification of twitter messages. In *Proceedings of the 5th workshop on language analysis for social media (lasm)* (pp. 17–25).

Lui, M., Lau, J. H., & Baldwin, T. (2014). Automatic detection and language identification of multilingual documents. *Transactions of the Association for Computational Linguistics, 2*, 27–40.

Mancioday. (2012, Mar). *Qui est mhd, le prince de l'afro-trap ?* Retrieved from `https://www.lesinrocks.com/2016/04/19/musique/musique/mhd-prince-parisien-de-lafro-trap/`

McCallum, A., Nigam, K., et al. (1998). A comparison of event models for naive bayes text classification. In *Aaai-98 workshop on learning for text categorization* (Vol. 752, pp. 41–48).

Milroy, L., & Muysken, P. (1995). *One speaker, two languages: Cross-disciplinary perspectives on code-switching.* Cambridge University Press.

Nguyen, D., & Doğruöz, A. S. (2013). Word level language identification in online multilingual communication. In *Proceedings of the 2013 conference on empirical methods in natural language processing* (pp. 857–862).

Paine, S. (2012). The quadrilingual vocabulary of french rap. *Multilingualism in Popular Arts, 3*(1), 48–69.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12*, 2825–2830.

Piergallini, M., Shirvani, R., Gautam, G. S., & Chouikha, M. (2016). Word-level language identification and predicting codeswitching points in swahili-english language data. In *Proceedings of the second workshop on computational approaches to code switching* (pp. 21–29).

Poplack, S. (1980). Sometimes i'll start a sentence in spanish y termino en espanol: toward a typology of code-switching1. *Linguistics, 18*(7-8), 581–618.

Sarkar, M., Winer, L., & Sarkar, K. (2005). Multilingual code-switching in montreal hip-hop: Mayhem meets method, or,'tout moune qui talk trash kiss mon black ass du nord'. In *Isb4: Proceedings of the 4th international symposium on bilingualism* (pp. 2057–2074).

Sequiera, R., Choudhury, M., & Bali, K. (2015). Pos tagging of hindi-english code mixed text from social media: Some machine learning experiments. In *Proceedings of the 12th international conference on natural language processing* (pp. 237–246).

Solorio, T., & Liu, Y. (2008a). Learning to predict code-switching points. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 973–981).

Solorio, T., & Liu, Y. (2008b). Part-of-speech tagging for english-spanish code-switched text. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 1051–1060).

Vyas, Y., Gella, S., Sharma, J., Bali, K., & Choudhury, M. (2014). Pos tagging of english-hindi code-mixed social media content. In *Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp)* (pp. 974–979).

Wagner, J., Arora, P., Cortes, S., Barman, U., Bogdanova, D., Foster, J., & Tounsi, L. (2014). Dcu: Aspect-based polarity classification for semeval task 4. In *Proceedings of the 8th international workshop on semantic evaluation (semeval 2014)* (pp. 223–229).

Williamson, K., & Blench, R. (2000). Niger-congo. *African languages: An introduction, 1*, 42.

Zampieri, M., Gebre, B. G., & Diwersy, S. (2013). N-gram language models and pos distribution for the identification of spanish varieties (ngrammes et traits morphosyntaxiques pour la identification de variétés de l'espagnol)[in french]. *Proceedings of TALN 2013 (Volume 2: Short Papers), 2*, 580–587.

Zampieri, M., Tan, L., Ljubešić, N., & Tiedemann, J. (2014). A report on the dsl shared task 2014. In *Proceedings of the first workshop on applying nlp tools to similar languages, varieties and dialects* (pp. 58–67).

Appendix

Artist(s)	Song	Artist(s)	Song
MHD	La Moula	**MHD**	Afro Trap Pt. 2(Kakala Bomaye)
MHD	XIX	**MHD**	Afro Trap Pt. 3(Champions league)
MHD	Amina	**MHD**	Afro Trap Pt. 4(Fais le Mouv)
MHD	Bella	**MHD**	Afro Trap Pt. 5(Ngatie Abedi)
MHD	Tout Seul	**MHD**	Afro Trap Pt.6 (Molo Molo)
MHD	Porsche Panamera	**MHD**	Mort ce Soir
MHD	RogerMilla	**MHD**	Maman j'ai Mal
MHD	Encore	**MHD**	Rouler
MHD	Papalé	**MHD**	Bravo
MHD **Fally Ipupa**	Ma Vie	**MHD**	Afro Trap Pt. 8(Never)
MHD	A Kele n'ta	**MHD** **Wizkid**	Bella
MHD **Dadju**	Bebé	**MHD** **Yemi Alade**	Aleo
MHD **Stefflon Don**	Senseless Thing	**MHD** **Angelique Kidjo**	Wanyinyin
Lockslegl	Lo Ma Def	**Emma Nyra**	Rotate
Y Du V	Ayez	**Y Du V**	Petit Délire
Tour de Garde	Ninguin	**DoxMV**	#CESTFACILE // AFRO TRAP
Niska **MHD**	Versus	**Ferre Gola** **DJ Arafat**	Azalaki Awa - Remix
Dabs	Magie	**Aya Nakamura**	Love
DJ Pete **Sarkodie** **Lartiste**	Tu mérites	**DSK on the Beat** **Eugy** **Barack Adama**	Ya oh Gyal
DSK on the Beat, **Kiff no Beat**	Fais ton Malin	**Black M** **MHD**	a l'ouest
Booba **Niska** **Sidiki diabaté**	Ca va aller	**Moula Gang** **FBI** **Lou**	Y'a du goût
DJ Arafat	Enfant Béni	**Kaaris**	Je suis gninnin, je suis bien
Zaho **MHD**	Laissez-les-Kouma	**Sianna**	Siannaararabica

Table 7: All songs and their peforming artist(s) in the corpus. Each artist collaborating on a song is written on a new line.