Genre Classification from Lyrics: Methods

group 2

March 2017

1 Methods

1.1 Materials

In this study a lyrics database of 600 songs was used. These songs were equally distributed over six different genres: Blues, Country, Electric Dance Music (EDM), Metal, Rap, and Rock. The lyrics were gathered by six persons. Each person gathered 100 songs of one genre. Within the broader genres, like metal and rock, the songs gathered were all of one specific sub-genre, to avoid discrepancies within one genre. The data of each genre exists out of songs from ten artists, with ten songs per artist. These songs were randomly picked. The database exists only out of the lyrics, the title, and the artists of a song. No further musical features were added.

1.2 Measurements

Within this experiment, two identical classifiers were trained on two different feature spaces. One classifier was trained on baseline Term Frequency-Inverse Document Frequency (TF-IDF) Vectors, whereas the other classifier was trained on both these TF-IDF vectors, and additional numerical features of the lyrics.

The TF-IDF vectors were created using the function TfidfVectorizer() from the python library Sklearn. This function first removes all stopwords. These are all words that appear in many texts and do not carry meaning alone. This includes words like: 'the', 'with', and 'of'. Then it makes a bag-of-words of the lyrics of a song. A bag-of-words is a vector of tuples, in which every tuple contains a word and the amount of times the word has been seen in the lyrics.

The Term Frequency is calculated by using the following formula:

$$tf_{t,d} = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}$$

Where $f_{t,d}$ is the count of term t in document d.

The inverse document frequency is used to give a higher weight to terms that

are seen less in the entire corpus of lyrics. It does this by multiplying the term frequency with the inverse document frequency: a logarithmic function that gets bigger the less a term is seen in the corpus. The formula for the final vectors is the following:

$$tf - idf_{t,d} = tf_{t,d} * \log \frac{n}{df_t}$$

Where n is the total amount of documents (or lyrics in our case), and df_t is the document frequency: The number of documents that contain term t.

These calculations are done for all terms t in a document d and put in one vector. This is the TF-IDF vector of document d, and these vectors were used as baseline in our experiment.

For our classifier we used a technique called *One vs Rest*. This means that instead of having a single classifier that classifies all genres at the same time, there is a classifier for every genre, that gives the certainty a song is its specific genre. These classifiers are binary: They classify whether a song is a specific genre (eg. blues), or not a specific genre (eg. not blues). The genre that a song is classified as is equal to the genre that had the certainty.

1.3 lyric-specific features

There were many lyric-specific features tried, to see if they would improve the baseline features. All features will be explained here.

Total Word Count: The number of words in a set of lyrics.

Unique Line Ratio: The amount of unique lines, divided by the total amount of lines.

Unique Word Ratio: The amount of unique words, divided by the total amount of words.

Characters Per Word: The average amount of characters per word. The total amount of characters divided by the total amount of words.

Words Per Line: The average amount of words per line. Total amount of words divided by the total amount of lines.

1.4 analyses