Abstract

This study focuses on developing four Machine Learning models to classify emotions conveyed in English song lyrics at the stanza level. The classification leverages Plutchik's framework of eight primary emotions, offering a nuanced understanding of emotional expression in lyrical content.
The selected model architectures are as follows:

- **Random Forest**
- **Support Vector Machine (SVM)**
- **One-Dimensional Convolutional Neural Network (1D-CNN)**
- **Recurrent Neural Network (RNN)**

These models were chosen for their proven effectiveness across various domains and their diverse methodological approaches, providing a comprehensive evaluation of different techniques for emotion classification in text. **DA FINIRE e far stare nella stessa pagina con l'introduzione**

Introduction

Lyrics serve as one of the main foundations of songs, playing a crucial role in expressing feelings in many different ways. The emotional tone of songs can serve various purposes, such as automatized playlist creation or songs' organization, offering an alternative to the more traditional genre-based classification.

The goal of this project is the development of 4 Machine Learning models that perform emotion detection on song lyrics stanzas. To obtain a deeper understanding of emotional fluctuations within the texts, the models assign emotion labels to individual stanzas instead of full songs. The emotion labels are assigned based on Robert Plutchik's eight primary emotions (shown in figure 1), offering a comprehensive range for representing diverse emotional states.

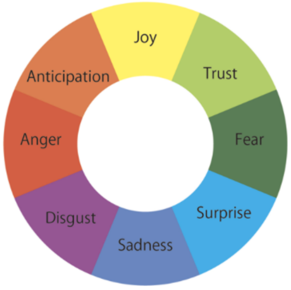

Figure 1: Plutchik's eight primary emotions

This report tries to cover and illustrate clearly various aspects of this work. In the *Method* section, we will provide a detailed explanation of the data and procedures used in the project, in particular describing the pipeline taken to implement the models. Then, the *Results* chapter will provide an overview of the obtained results with the aid of plots and figures, highlighting the significant outcomes. This section will be connected to the last two, i.e. the *Discussion* and *Conclusions* ones, which will explain what the general findings mean, recapping the primary objective of the work and discussing the importance or potential applications of the results.

1. Methods

This section will provide an overview about the data, methods and procedures used in the project.

The dataset used in this project represents a sampled subset of songs derived from the Genius Song Lyrics Dataset[1]. The original dataset (3m records) included songs in many different languages; however, this work focused exclusively on English-language ones. The original dataset contained numerous attributes, but the ones considered relevant for model training are:

- **title:** the song's title;
- **lemmatized_stanzas:** lyrics of the single stanza;
- **stanza_number:** identifies the position of the stanza in the song;
- **is_chorus:** boolean variable that attests whether the stanza is a chorus or not;
- **tag:** represents the genre of the song. For easier handling, this attribute of the original dataset has been one-hot encoded into various boolean variables (is_country, is_pop, is_rap, is_rb, is_rock);
- **label:** represents the emotional classification of the stanza, assigned by Albert Base v2[2] model.

All of these attributes, except for the *title* one, were the result of the preprocessing phase, as will be described later in the *Preprocessing* paragraph.
Due to limited computational power, the labeling process was time-intensive, ultimately resulting in a limited dataset consisting of **(QUANTE? AGGIUNGEREI NUMERO STROFE)**.

The first step in the preprocessing phase of this dataset involved sampling from the original dataset while preserving the proportions of the different genres. This ensured that the genre distribution in the subset remained representative of the full dataset.

The preliminary text cleaning process focused on the *lyrics* attribute, which was the attribute of the original dataset that contained the entire lyrics of each song (in string format). Initially, we built a RegEx to clean the lyrics' strings from noise, specifically targeting words enclosed between square brackets that were irrelevant for the stanza splitting process. Many of the keywords marking different stanzas were written within square brackets, and removing the non-keyword items within brackets was essential to prevent potential issues.

The crucial step was the stanza splitting. After cleaning the strings from the noisy square-bracketed items, we split them based on various keywords used to denote stanzas (such as *chorus, verse, intro, outro, refrain, hook* etc.). The RegEx we developed also accounted for the different formats in which these keywords appeared; between square brackets, parentheses, without brackets, only a double newline character between one stanza and the other. The output of this step was, for each song record, a list of stings, corresponding to a list of stanzas (with the stanza's header as the corresponding keyword).

Next, we removed the resulting strings that were uninformative; such as empty strings or those with fewer than 20 characters, which were too short to provide useful content.

As a result, the output of this preliminary preprocessing phase is a dataset in which the records are not whole songs anymore but single stanzas; each numbered based on its position in the song.

A further and deeper cleaning process on the stanzas involved the creation of the boolean feature *is_chorus*; *true* value for repeated stanzas for the same song or stanzas that had *hook, chorus, refrain, bridge* as a header.

We then removed the stanza headers and the newline characters between verses to obtain cleaner stanzas.

Since choruses, hooks, bridges and refrains often repeat throughout songs, we decided to drop duplicate stanzas in order to avoid redundant data. This resulted in a dataset of cleaned and non-duplicate stanzas: the checkpoint for the labelling step and the starting point for the text lemmatization process. To label the dataset, the Albert Base v2 model has been used; this transformer model is primarily aimed at being fine-tuned on tasks that use the whole sentence to make decisions, such as sequence classification.

The next step involved lemmatizing stanzas using the *spaCy* library. We created a list of lemmatized tokens (filtering punctuation and empty words). We opted for lemmatization over stemming because lemmatization produces more accurate and meaningful results, particularly for tasks requiring semantic understanding, such as in our case.

2. Static Models

2.1 Random Forest

2.2 SVM

3. Neural Networks

3.1 One-Dimensional Convolutional Neural Network

3.2 Recurrent Neural Network

Key findings and conclusions

Bibliography

[1] *Genius Song Lyrics*. URL: https://www.kaggle.com/datasets/carlosgdcj/genius-song-lyrics-with-language-information?select=song_lyrics.csv.

[2] *Albert Base v2*. URL: https://huggingface.co/albert/albert-base-v2.

List of figures