Music genre classification based on song lyrics - comparison between different word embedding techniques and classifiers
Project Proposal for NLP Course, Winter 2022

Bartłomiej Eljasiak
Warsaw University of Technology
bartlomiej.eljasiak.stud@pw.edu.pl

Aleksandra Nawrocka
Warsaw University of Technology
aleksandra.nawrocka.stud@pw.edu.pl

Dominika Umiastowska
Warsaw University of Technology
dominika.umiastowska.stud@pw.edu.pl

supervisor: Anna Wróblewska
Warsaw University of Technology
anna.wroblewska1@pw.edu.pl

Abstract

Music genre classification (MGC), although a well-known task, still remains challenging in the domain of Music Information Retrieval. We tackle the problem of MGC based solely on lyrics and try to solve it using a solution composed of a state-of-the-art word embedding method tuned for this problem and a separate classification model. Our main contribution is the comparison between different word embedding methods, classification techniques and optimization techniques, which in the domain of MGC is currently lacking. The novelty comes with an additional approach in the form of testing the impact of enriching the words with their sentiment obtained using a separate model.

1 Introduction

A music genre is a conventional label on the musical piece which characterizes it as having certain features, conventions, or characteristics. It is quite a complicated problem to say precisely how genres are distinguished. The genre often dictates the style and rhythm of the audio of the song. It seems much harder to define the music genre by lyrics alone, even from a human perspective. Therefore, it is quite an interesting topic to try making such a distinction based on song text. Similar research has already been conducted, but this topic is yet to be fully explored.

The song's lyrics are often related to its melody and rhythm. It is also common for different genres to raise different topics. It was already shown that a combination of audio and text features gets better results than using only audio features [19]. Furthermore, lyrics may be more accessible and easier to process than audio. Therefore, lyrics classification seems to be an interesting field of study both for its own and for its potential connection with audio features.

In this research, we want to explore different methods for lyrics-based genre classification. Our study will include testing different methods of obtaining text embeddings, such as Continuous Bag-of-Words, GloVe, word2vec, BERT, and varying classification models, such as Naive Bayes, Support Vector Machine, XGBoost, and Convolutional Neural Network.

We also want to include in our research sentiment analysis of the text. One of the characteristics of the music genre, though rarer considered, is the emotion that the song conveys. We decided to check how exactly those two relate since there seems to be sparse similar research. Therefore, for the above genre classification task, we will additionally consider the emotion detected in the song lyrics and with the use of fusion techniques check how it influences classification performance.

2 Significance

Music genre classification (MGC) is at this point a well-known research problem and a subdomain of Music Information Retrieval (MIR). Culture and therefore music avoids strict barriers and definitions, nevertheless, each piece of music is usually categorized into one or more genres. MGC enables us to study this categorization, explore similarities and differences between various genres or even construct a taxonomy.

In the past, due to heavy computational limitations, the main focus of MGC was put on finding the best features for classification purposes. In [21] such features were e.g. *AverageSyllablesPerWord* or *SentenceLengthAverage*. Naturally, word embedding played an important role in extract-

ing information from lyrics and the use of simple methods like *bag-of-words* can be found in various papers [8, 17]. With time, an increasing amount of focus was put strictly on embeddings themselves, developing novel and improved representations.

Currently, all state-of-the-art approaches for MGC utilizing lyrics rely heavily on word embeddings. In a recent publication [11] an attempt was made to train the embedding model strictly on lyrics. Unfortunately, the significance of the work is hard to assess due to the lack of usage of this model.

It is also rather common to approach MGC in a multi-modal manner. Usage of the audio itself has to be second if not the most popular source of information with many published articles [2, 32, 24, 21, 20]. Other less trivial data sources are symbolic [34], culture [21], text reviews [24], and cover art [24]. One could say that at this stage researchers experiment with enriching the pieces of music with any meaningful data possible.

To our surprise, we were not able to find any previous research which extracted sentiment from lyrics and used it for purpose of MGC (although a somewhat reversed connection has been studied in [18]). This is a niche exploration which will be a part of this work. It should be noted that the sentiment of the lyrics will be obtained via model from the lyrics themselves. This means that solution proposed will also base solely on lyrics.

Reading through the papers approaching MGC in different ways, it is striking that in some cases crucial elements of the proposed solution, are presented without proper justification. In the case of [30] a 100-dimensional GloVe model was used. It was stated that it is better than another technique called word2vec, but no proof or reference was provided. No other methods were used therefore it is impossible to say what was the value gained from using the GloVe and not e.g. bag-of-words. In another work [1] authors used word embeddings obtained from BERT and DistilBERT, with build-on classifiers, then compared their accuracy to BILSTM [28], which as input received text embedded in an unspecified manner. Numerous simplifications, lack of details and often incomparable results should raise concerns among researchers. How can one declare improvement over some method or even guarantee the value of the proposed work, when provided context for the work is insufficient? Those concerns motivated us to create such context as a result of this project. We want to declare with detail the conditions under which one word embedding method can be described as better for purpose of MGC and test which classification method works best on created lyrics representations.

3 Concept and work plan

Our project is divided into two parts. The main part consists of the comparison between different word embedding techniques, classifiers and optimization methods for MGC. In the second, smaller part we are going to test a modified approach to the above problem.

3.1 Work plan

In accordance with overall deadlines, we plan to:

- by November 18th have the architecture of the main part of the project prepared and at least one word embedding technique tested,
- by November 25th have the main part of the project finished and the architecture of the modified approach prepared,
- by December 9th have both parts of the project finished, the presentation and plan for the second project prepared,
- by December 16th have the report of the first project prepared.

3.2 Risk analysis

While planning the project we have identified multiple risks, which are presented in the table 1.

Risk	**Consequence**	**Mitigation**
not enough time for performing the project because of other obligations	late submission of the project or not finishing it	reducing the project scope by testing smaller numbers of models/techniques
not big enough computational resources to conduct proper experiments	lower quality of conducted experiments	-

Table 1: Risk analysis

Regarding the second part of the project, as we define it as a hypothesis rather than a thesis, we do not consider unsatisfactory results of the modified approach as a failure. We are interested in the outcome of conducted experiments but we do not have high expectations when it comes to accuracy.

4 Approach and research methodology

4.1 Datasets

While trying to find possible datasets for our project it was important to us for the song lyrics to be in their raw form. That means that they should not be transformed into e.g. bag-of-words model. The reason for this decision was that if we would like to use prepared models in a real-world scenario new song lyrics could be used with minimal preparation. What is more, as the language evolves all the time, with this approach there is still a possibility of further training of the models on song lyrics with the presence of not previously known words. And last but not least, we also wanted to minimize the risk of worse performance of prepared models which could be caused by simplifying the assumptions.

The effect of making this decision is the fact that we could not choose e.g. the musiXmatch dataset (the official lyrics collection of the Million Song Dataset [4]) as the dataset for conducting the experiments. This very well-known dataset consists of an enormous number of song lyrics which are unfortunately kept in a bag-of-words model form. But as we will not use this dataset, we also will not focus on its description.

We have found two datasets that meet our expectations:

- *Song lyrics from 79 musical genres* dataset from Kaggle website [23],
- *MetroLyrics* dataset processed and put in a GitHub repository [6].

In the description of the first dataset, we can find the information that the dataset consists of 379 893 song lyrics from 4239 artists. Around 50% of the song lyrics are in English and we will probably test our models on them. Information about the artists is kept in a separate file and contains a list of music genres each artist is connected with. As we plan to predict only one music genre for each song we will have to preprocess this dataset by reducing these lists to individual genres and assigning them to song lyrics of appropriate artists. Furthermore, song lyrics also need some preprocessing as they contain punctuation and span across multiple lines.

By contrast, the second dataset requires minimal work on our site. It was initially published on Kaggle website and consisted of 362 237 song lyrics from 18231 artists. The majority of song lyrics (probably around 60%) were in English. Unfortunately, this dataset was removed from Kaggle website and we were not able to find it in its original form anywhere else. We have found a preprocessed version of it in a GitHub repository of a students' project performed by University of California students in 2018. This version's song lyrics have punctuation removed and contain only one genre for each entry. Based on descriptions of the original dataset we expect 11 genres and an unbalanced dataset with highly frequent *Rock* label.

Finally, in case of problems connected to data we are considering creating our own dataset. This would be possible with the usage of Spotify API [29] and Genius API [12], which are well documented. We would use Spotify API to get recommendations of songs for chosen genres and Genius API for lyrics extraction. In comparison with both found datasets, our dataset would be a lot smaller but definitely balanced.

4.2 Embeddings

One of the key problems in the domain of natural language processing has to be the question of how to use words in a model which only understands numbers. This question sparked numerous attempts of representing language in a mathematical way. One can always assign each unique word a different number and in this way encode any language into the computer, but this is insufficient when it comes to using this encoded representation. It was rather clear, that in order for such transformation to be in any way useful, the original meaning of the word should be embedded into this numeric representation itself. This word embedding ought to be treated as a vector in a given, high-dimensional space. For a given model dimensionality is fixed, therefore each word is represented by a vector of a set length, typically a hundred or so numeric values.

There is still a task of creating a model capable of such transformations. It has a couple of possible

approaches, mainly prediction-based and count-based. The second one, although simpler, will not be described here, since all methods described further make use of the first approach. Prediction-based word embedding models share a common trait, which unsurprisingly is that the embedding for a word was learned by performing the task of predicting given word [3]. This definition does not set any requirements for the prediction itself, and, in fact, different approaches have been used successfully, such as the continuous Bag-of-Words Model (CBOW) and continuous Skip-gram Model [22].

There is a single more distinction for the different techniques used, which is significant for this work. The meaning of some words is not dependent on their structure or origin, but on the context in which they are used. In the case of homonyms, it is impossible to state the singular meaning of the word without context, e.g. bank as a financial institution vs. side of a river or play as in theatre vs. as in sport. More traditional embeddings do not incorporate the context of a word in determining its embedding. These models are called static or non-contextualized. The ones that do generate differentiable vectors depending on the context are subcategorized as contextualized word embeddings. Despite the described advantage of the latter method, prior has proven to be successful in multiple cases, such as GloVe [25] or fastText [5].

Other techniques which prove promising are ELMo [26], a deep bidirectional language model, which is pre-trained on a large text corpus and, extracting contextualized word embedding form, pre-trained Google's Bidirectional Encoder Representations from Transformers (BERT) [10].

Little has been said about using word embedding in the context of music lyrics. [11] describes the process of training the word embedding model strictly on music lyrics, but lacks proper evaluation methods to be comparable to other works. This means that in order to reach state-of-the-art we are bound to testing various methods of word embedding, retraining them whenever it is possible.

4.3 Classification models

We want to test a few varying classification models for this specific task. We will consider Naive Bayes classifier, Support Vector Machine, XGBoost, and Convolutional Neural Network.

Naive Bayes is a classifier based on Bayes' theorem known for good performance on real-world tasks despite being a simple model. It is fast and good at dealing with unbalanced data. It has also widespread applications on text classification tasks [27].

Support Vector Machine tries to find a hyperplane that best separates samples of different classes. It is often used for text classification tasks and historically achieved great results [14].

XGBoost [7] is a decision-tree based algorithm that uses a gradient boosting method. It shows great performance on large-scale tasks and is a very flexible and versatile tool.

Convolutional Neural Networks are one of the primarily used types of neural networks used commonly in both image and text classification [15]. Their main feature is using layers with convolution filters that are applied to feature vectors.

As for CNN architecture, we want to test a few different optimizers, such as Adam optimizer [16], Stochastic Gradient Descent and AdaDelta optimizer [33].

4.4 Sentiment analysis

It is a well-known fact that music can convey deep emotions to listeners. These emotions are present in both melody and lyrics. In this research, we decided to study the connection between those emotions contained in lyrics and the song genre. To do this we want to include sentiment analysis when classifying song lyrics to the song's genre. Since emotions in song lyrics are often non-binary we want to consider a model that can recognize more varying emotions.

We decided to use an already pre-trained model for emotion recognition called Emotion English DistilRoBERTa-base [13]. The model was created by fine-tuning DistilRoBERTa-base and training it on the balanced dataset of 2800 observations for each emotion summing up to 20k observations in total.

The model predicts Ekman's six basic emotions, that is anger, disgust, fear, joy, sadness, surprise, and an additional neutral class - summing up to seven labels.

4.5 Modified approach using sentiment analysis model and data fusion technique

As mentioned above, besides comparing different word embedding techniques and classifiers,

we want to test another, significantly modified approach. We want to take advantage of already existing, pre-trained sentiment analysis models and see if they can help to improve the accuracy of prepared architectures. To do that we will make use of methods known from multi-modal machine learning (of course we only have one modality).

The plan is to divide song lyrics in half. The first part will serve as an input to the word embedding model and the second to the sentiment analysis model. As outputs we will get an embedding of passed words and a vector of probabilities of different emotional states. These outputs, even though take the same form of numerical vectors, have different meanings. What is more, the output of the sentiment analysis model can be seen as a final decision as it contains information understandable to humans in contrast to the received embedding. However, the origins of these outputs may have common parts, e.g. reoccurring choruses, and because of that not as much information may be introduced in the second part of the song lyrics as would in the case of a new modality. Regardless of these considerations, we need to make use of data fusion techniques.

There are a lot of data fusion techniques. They can be divided into groups such as early fusion or late fusion. Early fusion is the process of merging data from multiple sources before conducting the analysis. This data can be described as input data, even though it is often already preprocessed, e.g. in the form of embeddings. An example of early fusion is simple concatenation of input data in the form of numerical vectors, where the resulting vector can be passed as input to the next module in the model's architecture. Late fusion is a concept similar to an ensemble. Classifiers are trained for all modalities and their outputs are merged to obtain a single decision e.g. by using weighted voting.

We are going to use an early fusion method even though part of our data will be in the final form. The method is called cross-modal attention [9] and is based on the usage of multi-head scaled dot product attention [31]. In contrast to self-attention, query matrix comes from a different modality than key and value matrices. This can help to capture relationships between different modalities. In our case, we will apply this method to only one modality by using the output of the sentiment analysis model for the query matrix and embedding of the first part of song lyrics for key and value matrices. This will lead to sentiment-guided lyrics feature output, which will be introduced as input data to a classifier.

We want to test if this approach will obtain better accuracy for the music genre classification problem. By using the sentiment analysis model and cross-modal attention mechanism we make our model more complex and introduce more trainable parameters. We also make use of transfer learning. These arguments weigh in favour of this approach. However, we only use the first halves of song lyrics to obtain embeddings and we do not introduce the second modality, even though we use a multi-modal data fusion technique, so predicted gains are lower than in the case of multiple modalities. As there exist arguments from both sides it is hard to tell what will be the outcome of the experiment.

References

[1] Hasan Akalp et al. "Language Representation Models for Music Genre Classification Using Lyrics". In: *2021 International Symposium on Electrical, Electronics and Information Engineering*. ISEEIE 2021. Seoul, Republic of Korea: Association for Computing Machinery, 2021, pp. 408–414. ISBN: 9781450389839. DOI: 10.1145/3459104.3459171. URL: https://doi.org/10.1145/3459104.3459171.

[2] Safaa Allamy and Alessandro Lameiras Koerich. "1D CNN Architectures for Music Genre Classification". In: *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*. 2021, pp. 01–07. DOI: 10.1109/SSCI50451.2021.9659979.

[3] Marco Baroni, Georgiana Dinu, and Germán Kruszewski. "Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors". In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 238–247. DOI: 10.3115/v1/P14-1023. URL: https://aclanthology.org/P14-1023.

[4] Thierry Bertin-Mahieux et al. “The Million Song Dataset”. In: *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*. 2011.

[5] Piotr Bojanowski et al. “Enriching Word Vectors with Subword Information”. In: *CoRR* abs/1607.04606 (2016). arXiv: 1607.04606. URL: http://arxiv.org/abs/1607.04606.

[6] Connor Brennan et al. *SongGenreClassification*. 2018. URL: https://github.com/hiteshyalamanchili/SongGenreClassification/tree/master/dataset (visited on 10/31/2022).

[7] Tianqi Chen and Carlos Guestrin. “XGBoost: A Scalable Tree Boosting System”. In: (2016).

[8] Önder Çoban and Işıl Karabey. “Music genre classification with word and document vectors”. In: *2017 25th Signal Processing and Communications Applications Conference (SIU)*. 2017, pp. 1–4. DOI: 10.1109/SIU.2017.7960145.

[9] Krishna D. N. and Ankita Patil. “Multimodal Emotion Recognition Using Cross-Modal Attention and 1D Convolutional Neural Networks”. In: Oct. 2020, pp. 4243–4247. DOI: 10.21437/Interspeech.2020-1190.

[10] Jacob Devlin et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *CoRR* abs/1810.04805 (2018). arXiv: 1810.04805. URL: http://arxiv.org/abs/1810.04805.

[11] Seungheon Doh et al. “Musical Word Embedding: Bridging the Gap between Listening Contexts and Music”. In: *CoRR* abs/2008.01190 (2020). arXiv: 2008.01190. URL: https://arxiv.org/abs/2008.01190.

[12] Genius. *Genius API*. URL: https://docs.genius.com/ (visited on 10/31/2022).

[13] Jochen Hartmann. *Emotion English DistilRoBERTa-base*. 2021. URL: https://huggingface.co/j-hartmann/emotion-english-distilroberta-base/ (visited on 11/06/2022).

[14] Thorsten Joachims. “Text categorization with Support Vector Machines: learning with mny relevant features”. In: (1998).

[15] Yoon Kim. “Convolutional Neural Networks for Sentence Classification”. In: (2014).

[16] Diederik P. Kingma and Jimmy Lei Ba. “Adam: A Method for Stochastic Optimization”. In: (2015).

[17] Dawen Liang, Haijie Gu, and Brendan O’Connor. “Music genre classification with the million song dataset”. In: *Machine Learning Department, CMU* (2011).

[18] Yu-Ching Lin et al. “Exploiting genre for music emotion classification”. In: Aug. 2009, pp. 618–621. DOI: 10.1109/ICME.2009.5202572.

[19] Rudolf Mayer and Andreas Rauber. “Music genre classification by ensembles of audio and lyrics features”. In: (2011).

[20] Rudolf Mayer and Andreas Rauber. “Music Genre Classification by Ensembles of Audio and Lyrics Features.” In: Jan. 2011, pp. 675–680.

[21] Cory McKay et al. “Evaluating the Genre Classification Performance of Lyrical Features Relative to Audio, Symbolic and Cultural Features.” In: *Proceedings of the 11th International Society for Music Information Retrieval Conference, ISMIR 2010* (Jan. 2010), pp. 213–218.

[22] Tomas Mikolov et al. “Efficient estimation of word representations in vector space”. In: *arXiv preprint arXiv:1301.3781* (2013).

[23] Anderson Neisse. *Song lyrics from 79 musical genres*. 2022. URL: https://www.kaggle.com/datasets/neisse/scrapped-lyrics-from-6-genres/versions/3 (visited on 10/31/2022).

[24] Sergio Oramas et al. “Multimodal deep learning for music genre classification”. In: *Transactions of the International Society for Music Information Retrieval. 2018; 1 (1): 4-21.* (2018).

[25] Jeffrey Pennington, Richard Socher, and Christopher Manning. “GloVe: Global Vectors for Word Representation”. In: *Pro-*

ceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543. DOI: 10.3115/v1/D14-1162. URL: https://aclanthology.org/D14-1162.

[26] Matthew E. Peters et al. "Deep contextualized word representations". In: *CoRR* abs/1802.05365 (2018). arXiv: 1802.05365. URL: http://arxiv.org/abs/1802.05365.

[27] Sebastian Raschka. "Naive bayes and text classification i-introduction and theory". In: (2014).

[28] M. Schuster and K.K. Paliwal. "Bidirectional recurrent neural networks". In: *IEEE Transactions on Signal Processing* 45.11 (1997), pp. 2673–2681. DOI: 10.1109/78.650093.

[29] Spotify. *Spotify Web API*. URL: https://developer.spotify.com/documentation/web-api/ (visited on 10/31/2022).

[30] Alexandros Tsaptsinos. "Lyrics-Based Music Genre Classification Using a Hierarchical Attention Network". In: *CoRR* abs/1707.04678 (2017). arXiv: 1707.04678. URL: http://arxiv.org/abs/1707.04678.

[31] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[32] Yang Yu et al. "Deep attention based music genre classification". In: *Neurocomputing* 372 (2020), pp. 84–91. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2019.09.054. URL: https://www.sciencedirect.com/science/article/pii/S0925231219313220.

[33] Matthew D. Zeiler. "ADADELTA: An Adaptive Learning Rate Method". In: (2012).

[34] Mingliang Zeng et al. "MusicBERT: Symbolic Music Understanding with Large-Scale Pre-Training". In: *CoRR* abs/2106.05630 (2021). arXiv: 2106.05630. URL: https://arxiv.org/abs/2106.05630.