Measuring the Similarity of Song Artists using Topic Modelling

Erik Calcina
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

Erik Novak
Jožef Stefan International Postgraduate School
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

ABSTRACT

In music streaming platforms, it is necessary a recommendation system to provide users with similar songs of what they already listen and also recommend new artists they might be interested in. In this paper, we present a method to find similarities between artists that uses topic modelling. We have evaluated the method using a data set with music artists and their lyrics. The results show the method finds similar artists, but also is dependant on the quality of the generated topic clusters.

KEYWORDS

song lyrics, topic modelling, clustering, sentence embeddings, language models

1 INTRODUCTION

Nowadays, there are a plenty of music platforms to choose from and listen to music. There, new artists appear every day and many different songs are published. If we take into account all that have been created, we get a large selection of songs which can increase the difficulty of finding suitable songs or artists to listen to.

To find a suitable artist or songs, different aspects can be considered. One such aspect can be the topic of the song; a song topic can be interpreted as the main subject of the song, for example it can be an emotion, an event, a message, or something else. When searching for suitable artists one could decide to search for artists who have songs on similar topics.

In this paper, we propose an topic modeling-based approach for measuring the similarity of the music artists based only on their song lyrics. The approach uses language models for generating song embeddings used to create the topic clusters. These topic clusters are then analyzed to find the similar artists. The experiment was performed on a data set of songs corresponding to fourteen (14) music artists. While the experiment shows that similar artists can be detected using the approach, there is still room for improving its performance.

The main contribution of this paper is a novel approach for detecting similar music artists using topic modelling.

The reminder of the paper is structured as follows: Section 2 contains the overview of the related work on using topic modelling on song data sets. Next, we present the methodology in Section 3, and describe the experiment setting in Section 4. The experiment results are found in Section 5, followed by a discussion in Section 6. Finally, we conclude the paper and provide ideas for future work in Section 7.

2 RELATED WORK

Related works to our topic modeling approach use Latent Dirichlet Allocation (LDA) [1]. One work uses a topic modeling technique for sentiment classification, classifying between happy and sad songs, by using generated topics created with LDA and Heuristic Dirichlet Process [12]. From a data set consisting of 150 lyric they've been able to retrieve the sub-division of two defined sentiment classes [3]. Another work used LDA and Pachinko allocation [7] on a large data set for assessing the quality of the generated topics with applying supervised topic modeling approach. [8]. In our paper we use topic modeling to generate a set of topic clusters used to calculate the similarity between artists.

3 METHODOLOGY

In this section, we present the methodology used in this paper. We present the topic modeling approach used to generate the topic clusters, followed by a description of how the topic clusters are used to measure the similarity between the artists.

3.1 Topic Modeling

To create the topic clusters we use BERTopic [5], a method which uses document embeddings with clustering algorithms to create topic clusters. While BERTopic is described in a separate work, we present a brief description of its workflow. The workflow is also presented in Figure 1.

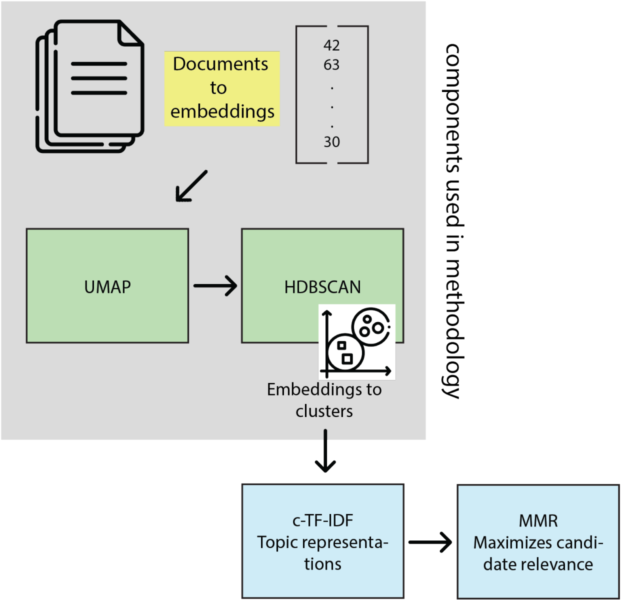
Documents to embeddings
42
63
.
.
.
30
UMAP
HDBSCAN
Embeddings to clusters
components used in methodology
c-TF-IDF Topic representations
MMR Maximizes candidate relevance

Figure 1: The BERTopic methodology workflow. The highlighted part is used in our approach. The image has been designed using resources from Flaticon.com.

Information Society 2022, 10–14 October 2022, Ljubljana, Slovenia

Document Embeddings. Document vector representations are generated using a sentence-transformer [11] model. The model creates a semantic representation of the documents, which allows measuring the semantic similarity. The available models support creation of both monolingual and multilingual vectors. Since the embeddings will be used as an input of a clustering algorithm, dimensionality reduction is performed to improve the clustering results. The dimensionality reduction algorithm used is UMAP [10].

Document Clustering. Once the document embeddings are prepared, they are input into a clustering algorithm to create the topic clusters. The algorithm used is HDBSCAN [9], an optimized extension of the DBSCAN [4] algorithm. The chosen algorithm creates clusters based on the density of the document embedding space, which allows the documents to not be assigned to a cluster if it's not similar to any of the neighbouring documents.

Topic Word Description. Once the topic clusters are created, a topic word description is generated using the document's text. For each cluster the TF-IDF score is calculated for each word found in any of the cluster's documents; the scores are called cluster TF-IDF (c-TF-IDF). The words with the highest c-TF-IDF score are then chosen as the topic word description. Furthermore, maximal marginal relevance (MMR) is performed to diversify the selected words by measuring both the words relevance to the documents, and its similarity to the other selected words. Note that the topic word description were used only for the preliminary analysis of our work, but not for measuring artists similarity.

3.2 Artists' Similarity using Topic Clusters

Once the topic clusters are created, the similarity between artists can be measured. First, for each topic we count the songs that corresponds to a particular artist. This gives us the number of songs an artist has in a particular topic. To ensure that the presence is strong enough, we decide to remove the artists from a topic if the number of their associated songs is below some threshold. The threshold is set to five (5) in order to ensure that the songs were not assigned to a cluster by coincidence. Afterwards, for each pair of artists we calculate their similarity using the following equation:

$$\mathrm{sim}(a, b) = \frac{|A \cap B|}{|A|}, \quad (1)$$

where A is the set of topics of artist a, and B is the set of topics of artist b.

4 EXPERIMENT

We now present the experiment setting. First, we introduce the data set used and its pre-processing steps. Next, we describe the implementation details.

4.1 Dataset

To test our approach, we use a dataset with raw lyrics data [2]. The dataset consists of 218,210 rows containing the following attributes:

- *Song name.* The name of the song.
- *Release year.* The year when the song was released.
- *Song artist.* The name of the artist.
- *Artist genre.* The genre of the song.
- *Song lyrics.* The lyrics text of the song.

The attributes used in our analysis are song name, artist and lyrics.

Data Processing. For our experiment we took fourteen (14) artists of various degrees of similarity. This reduces the data set to 4,470 rows which is 2.05% of the whole data set.

After reviewing the lyrics, we realized that the data set has many song variations by the same artist, which can be seen as duplicates. To find and remove the duplicates, we created the TF-IDF representations for the songs, and calculated the cosine similarity with all other songs of the same artist; if the similarity is greater than 50% it was labeled as a duplicate and removed from the data set. This resulted in a smaller data set containing 3,455 song lyrics.

The final data set statistics used for our experiments is shown in Table 1.

Table 1: The experiment data set statistics. For each artist we denote the music genre of the artist (genre), the number of their songs in the data set (songs), and the average number of words in the song's lyrics (avg. length).

Artist	genre	songs	avg. length
black-sabbath	Rock	160	184
bon-jovi	Rock	320	266
dio	Rock	127	203
aerosmith	Rock	208	226
ac-dc	Rock	171	193
coldplay	Rock	138	174
50-cent	Hip-Hop	318	502
2pac	Hip-Hop	259	648
eminem	Hip-Hop	369	640
black-eyed-peas	Hip-Hop	119	463
celine-dion	Pop	182	230
britney-spears	Pop	225	313
frank-sinatra	Jazz	356	133
ella-fitzgerald	Jazz	503	156
Together	-	3,455	319

4.2 Implementation details

In this section, we present the details of how the approach is developed.

Language model. The method uses the pre-trained Sentence Transformer model, more precisely the `all-mpnet-base-v2` model[1], available via the HuggingFace's transformer library [13]. It can take up to 384 tokens as one input, which is more than the average number of words in our data set, and returns a 768 dimensional dense vectors. The vectors have been shown to be appropriate for task such as clustering and semantic search.

Dimensionality reduction. To perform dimensionality reduction, we set the UMAP parameters as follows: Fist, the number of neighboring sample points used when making the manifold approximation is set to five (5), to make the algorithm use the local proximity of the documents. Second, we set the dimensionality of the embeddings to one (1). This values were selected using hyper-parameter tuning.

[1] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

Clustering algorithm. In the HDBSCAN algorithm, the minimum number of documents in a cluster is set to five (5).

5 RESULTS

In this section, we present the experiment results. We analyze the topic clusters, followed by the description of the finding on artist's similarity.

Topic Cluster Analysis. The experiment has generates 215 topic clusters, out of which only 107 have at least one artist with more than five (5) songs in it. The cluster containing songs that are deemed as outliers is not included in the analysis.

The statistics of the topic clustering is shown in Table 2. Evidently, artists with a larger number of songs are spread over several topic clusters than those with less songs.

Table 2: Topic clustering results. For each artist we show the number of different topics the artist is associated with (topics), and the average number of their songs in the associated topics (avg. songs).

Artist	topics	#avg. songs
black-sabbath	6	5
bon-jovi	10	6
dio	4	7
aerosmith	9	6
ac-dc	7	5
coldplay	2	5
50-cent	17	9
2pac	13	9
eminem	18	9
black-eyed-peas	3	12
celine-dion	8	6
britney-spears	12	6
frank-sinatra	16	8
ella-fitzgerald	28	8

Artists' Similarity Analysis. The artists' similarity is shown in Figures 2 and 3, which show the heatmaps of the absolute and relative co-occurrence of artists in topic clusters, respectively.

By looking at rows of Figure 2, we see the number of common topics with other artists. For example, by taking 50-cent with his 17 topics, we see that he shares five (5) of them with 2pac, one (1) with black-eyed-peas, one (1) with ac-dc, and six (6) with eminem. From this we conclude that 50-cent, 2pac and eminem have more topics in common than the rest of the artists. In other words, 50-cent is more similar to the 2pac and eminem than to the rest of the artists.

Figure 3 shows the similarities calculated using Equation 1. The similarities become more visible, but at the same time can be also misleading. Artists with smaller number of topics can result in higher similarity with other artists with higher number of topics. For example, Coldplay have two (2) topics, one of which is shared with Bon Jovi. Despite the fact that only one topic is in common, it is unlikely they have a similarity of 50%.

6 DISCUSSION

In this section we discuss the advantages and disadvantages of the proposed methodology, and its possible improvements.

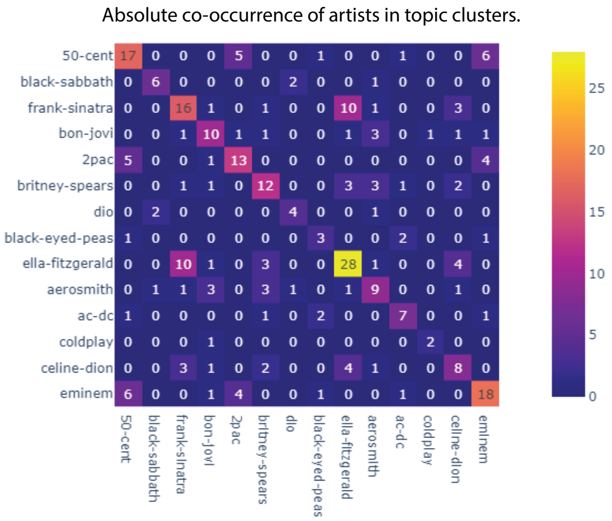

Figure 2: The absolute co-occurrence of artists in topic clusters.

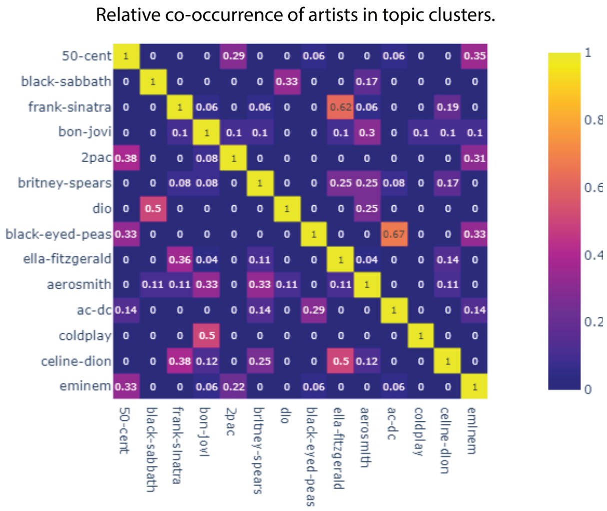

Figure 3: The relative co-occurrence of artists in topic clusters. Artists with smaller number of topics can result in higher similarity with other artists.

Language Models Limitations. The chosen language model `all-mpnet-base-v2` supports a maximum sequence length of 384 tokens which is the downside of this model for our experiment. Although the average number of words in the song lyrics is below the input limit, some artist have songs that are longer than that. However, songs have repeating sections, e.g. chorus, which is most likely inside the first 384 words. Therefore, the language models may not create a representation out of the whole song's lyrics, but it might capture the majority because of the song's repeated text.

Clustering Algorithm Selection. The clustering algorithm HDBSCAN can create a cluster consisting of examples, which do not fall into any of the topic clusters. It is convenient when instead of forcing songs into clusters, it labels them as outliers. The downside is when the majority of songs are labeled as outliers. To avoid this, other clustering algorithms that assign a cluster to every document can be used, for example K-means clustering [6].

6.1 Topic Cluster Discussion

Some artists with a small number of songs have a lower number of topics assigned, which is a problem for finding similarities. On the other side artists with higher number of songs tend to have more topics. Additionally, to avoid taking into account small number of artist co-occurrances, which can be a product of data noise, a filter threshold can be considered to remove them from the final analysis.

7 CONCLUSION

In this paper we present a way to measure similarity between music artists using topic modeling. We cluster lyrics and compare artists based on the generated topic clusters. The results have shown that the approach finds similar artists. However, it is heavily dependent on the number and quality of the topic clusters.

In the future, we intend to apply the methodology on a larger data set of song lyrics and artists. In addition, we intend to use all of the topic cluster information (including topic word descriptions) in order to improve the methodology's performance.

ACKNOWLEDGMENTS

This work was supported by the Slovenian Research Agency and the Slovene AI observatory under proposal no. V2-2146.

REFERENCES

[1] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation". In: *J. Mach. Learn. Res.* 3 (2003), pp. 993–1022. ISSN: 1532-4435. DOI: http://dx.doi.org/10.1162/jmlr.2003.3.4-5.993. URL: http://portal.acm.org/citation.cfm?id=944937.

[2] Connor Brennan, Sayan Paul, Hitesh Yalamanchili, Justin Yum. *Classifying Song Genres Using Raw Lyric Data with Deep Learning*. Accessed August 30, 2022. https://github.com/hiteshyalamanchili/SongGenreClassification. 2018.

[3] Maibam Debina Devi and Navanath Saharia. "Exploiting Topic Modelling to Classify Sentiment from Lyrics". In: *Machine Learning, Image Processing, Network Security and Data Sciences*. Ed. by Arup Bhattacharjee et al. Singapore: Springer Singapore, 2020, pp. 411–423. ISBN: 978-981-15-6318-8.

[4] Martin Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise". In: AAAI Press, 1996, pp. 226–231.

[5] Maarten Grootendorst. "BERTopic: Neural topic modeling with a class-based TF-IDF procedure". In: *arXiv preprint arXiv:2203.05794* (2022).

[6] Xin Jin and Jiawei Han. "K-Means Clustering". In: *Encyclopedia of Machine Learning*. Ed. by Claude Sammut and Geoffrey I. Webb. Boston, MA: Springer US, 2010, pp. 563–564. ISBN: 978-0-387-30164-8. DOI: 10.1007/978-0-387-30164-8_425. URL: https://doi.org/10.1007/978-0-387-30164-8_425.

[7] Wei Li and Andrew McCallum. "Pachinko allocation: DAG-structured mixture models of topic correlations". In: *ICML '06: Proceedings of the 23rd international conference on Machine learning*. New York, NY, USA: ACM, 2006, pp. 577–584. ISBN: 1595933832. DOI: 10.1145/1143844.1143917. URL: http://portal.acm.org/citation.cfm?id=1143917.

[8] Alen Lukic. *A comparison of topic modeling approaches for a comprehensive corpus of song lyrics*. Tech. rep. Tech report, Language Technologies Institute, School of Computer Science ..., 2015.

[9] Leland McInnes and John Healy. "Accelerated Hierarchical Density Based Clustering". In: *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*. 2017, pp. 33–42. DOI: 10.1109/ICDMW.2017.12.

[10] Leland McInnes, John Healy, and James Melville. *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*. 2018. DOI: 10.48550/ARXIV.1802.03426. URL: https://arxiv.org/abs/1802.03426.

[11] Nils Reimers and Iryna Gurevych. "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Nov. 2019. URL: https://arxiv.org/abs/1908.10084.

[12] Chong Wang, John Paisley, and David Blei. "Online variational inference for the hierarchical Dirichlet process". In: *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings. 2011, pp. 752–760.

[13] Thomas Wolf et al. "Transformers: State-of-the-Art Natural Language Processing". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6. URL: https://aclanthology.org/2020.emnlp-demos.6.