Genre Musical Lyrics Classification

Andrei-Eusebiu Blahovici*
University of Bucharest, Romania
blahoviciandrei1@gmail.com

Bogdan-Ioan Popa*
University of Bucharest, Romania
popabogdanpopa@gmail.com

Alexandru-Florentin Dabu*
University of Bucharest, Romania
adabu34@gmail.com

***Abstract*—There has been extensive research recently for tasks that involve lyrics processing. Our aim is to reproduce results from state of the art literature, whilst also bringing our own contribution to the existing results by trying different preprocessing steps and machine learning and deep learning algorithms. We are going to make an analysis only at word level, ignoring possible lyrical structure.**

I. Introduction and related work

a) Results based on traditional machine learning algorithms: Preliminary research in the field of genre classification shows that traditional machine learning algorithms achieve pretty good results. For example, in Teh Chao Ying, Doraisamy et al. [1], they used K-NN, Naive Bayes and SVM, but they used a dataset of 600 songs chosen at random after some predetermined factors. Because our dataset is significantly larger, we opted for Logistic Regression, Naive Bayes and AdaBoost Classifiers. As an improvement to prior machine learning research, this paper presents POS tagging as a form of pre-processing that yields better results.

b) Deep learning for genre classification using lyrics and audio: In the paper published by Kawisorn Kamtue, Kasina Euchukanonchai, et al. [2] they classify Lukthung songs from other genres based on the lyrics and the audio of those songs. At first, the words that are longer than 20 characters and have a frequency less than 10 are deleted from the dataset vocabulary. Afterward, the sentences are transformed into BoWs, which are going to be fed through a MLP network.

c) Recurrent neural network approaches: In the paper published by Alexandros Tsaptsinos [3], he brings together multiple state of the art results and shows how to achieve better results using the HAN network presented in Fig. 1. He uses both the word attention and sentence attention modules of the network. In his approach, Glove embeddings [4] are used for the representation of the words which are retrained.

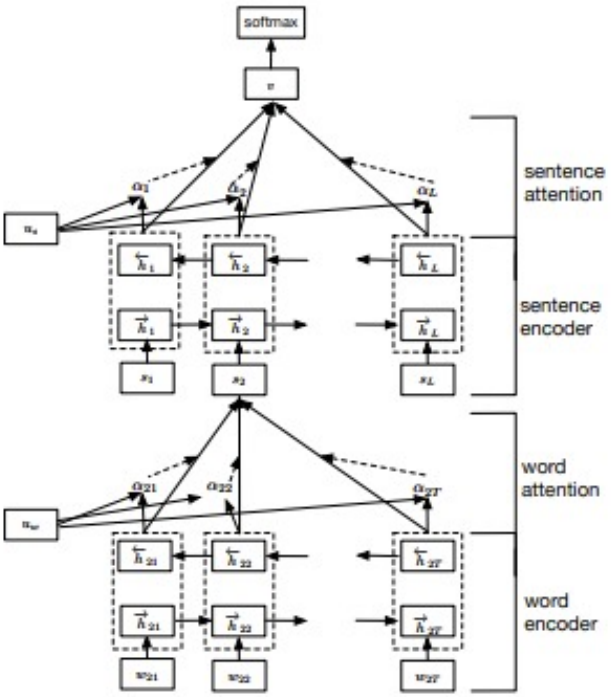

Fig. 1. HAN network architecture [5]

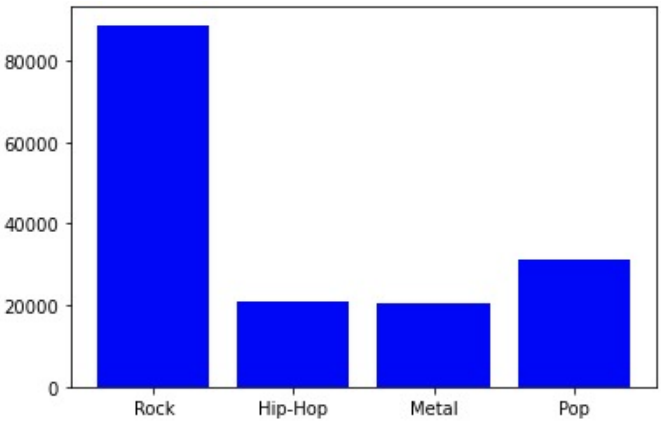

Fig. 2. Dataset distribution

II. Implementation details and results

A. Dataset and data processing

For our experiments, we are going to employ a version of the Kaggle's 380,000+ lyrics from MetroLyrics which only takes into consideration the English lyrics. We are going to use just the lyrics, which lack line representation in this version of the dataset, and the genre they are from for our analysis. As there are a lot of genres that have few samples, we decided to go just with the genres Rock, Hip-Hop, Metal, and Pop. The final distribution of the dataset is shown in Fig. 2.

B. Experiments

a) Classical machine learning algorithms:

Pre-processing

For this experiment, we used two distinct approaches to process the data. Firstly, we used the following: convert numbers to words, remove punctuation, and apply stemming. Secondly, we encoded each word by their POS tagging, by linking the lemma of the word with its' corresponding tag. To generate the Bag of Words we used CountVectorizer with 1000 features.

Used algorithms

We used the following models from sklearn: **LogisticRegression**, **AdaBoostClasifier**, and three **Naive Bayes** variants (**GaussianNB, MultinomialNB, BernoulliNB**). Throughout our experiment, the models that used the first pre-processing method performed better overall, but the second method scored the highest using BernoulliNB. The accuracies for every class obtained with this approach are shown in the first bar plot in Fig. 4. Also, we have provided the confusion matrix in Fig. 3.

b) First deep learning-based approach: The model used by us is similar to the one presented in the paper, but has two hidden layers of 1024 neurons and an output layer of four neurons with softmax activation, while the paper presents a MLP with 2 hidden layers of 100 neurons.

As we can see from Fig. 2, the rock class has significantly more songs and feeding our model with a training dataset which has this much imbalance showed to be very ineffective so what we did was to take as much data from the train dataset as we could such that we had an equal amount of samples from every class. This shows a great improvement and places this approach first across all of our experiments.

c) Improving the accuracies using recurrent neural networks: These experiments are inspired by Alexandros Tsaptsinos [3] and we are going to try to reproduce them and see if we obtain similar results. For all approaches, we are using Glove embeddings to represent each word and we let them train during inference. Also, we are using gradient clipping after each batch of data and early stopping as a regularization method.

LSTM-based approach

This network takes in the embeddings of each word of a song's lyrics and runs them through a unidirectional LSTM layer. The outputs of the LSTM layer are then passed through a MLP layer of size 4, in contrast with the number of classes we are trying to classify. Finally, a softmax is applied to the outputs of the entire architecture.

Hierarchical Attention Network

We are going to employ a simplified version of the HAN network that only makes use of the word attention layer, ignoring possible information that it might extract at line level. Even though this is not the original architecture, it still achieves better results than the simple LSTM-based approach.

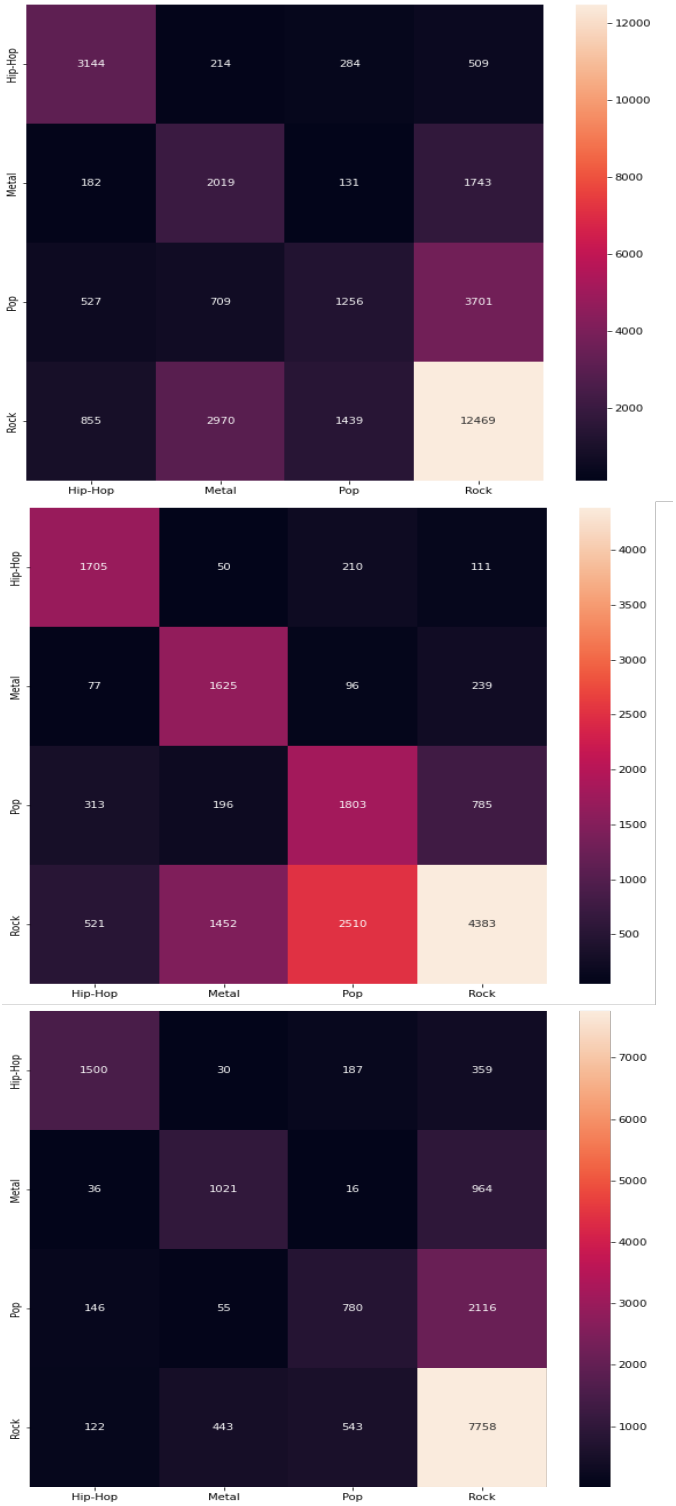

Fig. 3. This figure shows the confusion matrices for our 3 best experiments i.e. the first confusion matrix corresponds to the BernoulliNB classifier with POS tagging, the second confusion matrix is the one for the MLP and lastly, the third one is the one for the HAN network

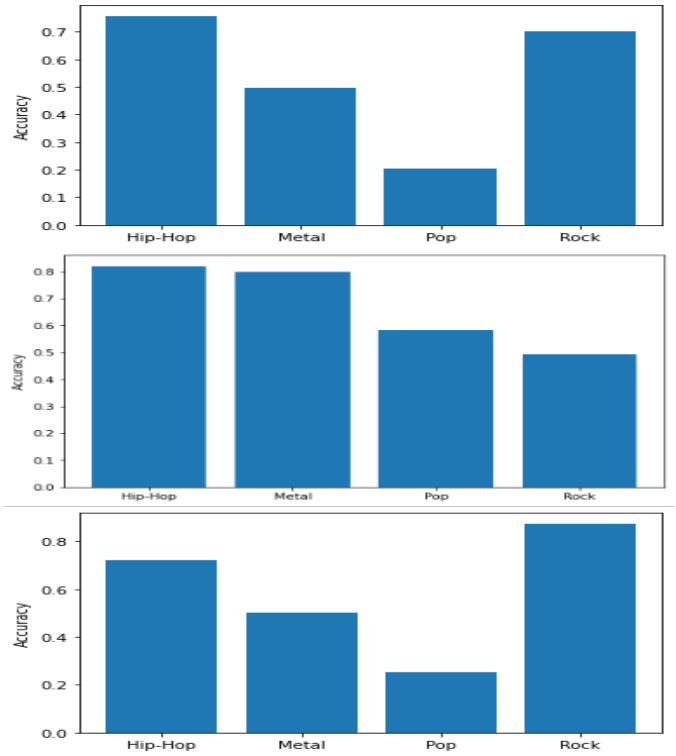

Fig. 4. This figure shows the accuracy per class for our 3 best experiments i.e. the first bar plot corresponds to the BernoulliNB classifier with POS tagging, the second plot is the one for the MLP and lastly, the third one is the one for the HAN network

III. Conclusions and future work

As we have noticed from our experiments, then simple MLP seems to bring the best results out of all the experiments we have run so far since it takes advantage of a more balanced version of the dataset. This shows that the other approaches, even though they bring fairly good results, are strongly biased towards the Rock genre as it is the predominant one.

An approach we have not covered in our experiments is using the sentence attention module of the HAN network, which might potentially bring so much better results in combination with a more balanced version of the dataset. Also, it is worth it to try freezing the Glove embeddings during training. Furthermore, we have not tested if, for example, using additional preprocessing steps, such as removing the stopwords, the punctuation etc. brings any benefit to the classification. For our future experiments, we would also like to try different types of language representations, like the ones that can be obtained from a network trained on huge datasets, some popular examples being BERT [6] and XLM-RoBERTa [7].

* Equal contribution.
* Github link: https://github.com/Dawlau/lyrics-classification-songs

References

[1] T. C. Ying, S. Doraisamy, and L. N. Abdullah, "Genre and mood classification using lyric features," in *2012 International Conference on Information Retrieval Knowledge Management*, 2012, pp. 260–263.

[2] K. Kamtue, K. Euchukanonchai, D. Wanvarie, and N. Pratanwanich, "Lukthung classification using neural networks on lyrics and audios," *CoRR*, vol. abs/1908.08769, 2019. [Online]. Available: http://arxiv.org/abs/1908.08769

[3] A. Tsaptsinos, "Lyrics-based music genre classification using a hierarchical attention network," *CoRR*, vol. abs/1707.04678, 2017. [Online]. Available: http://arxiv.org/abs/1707.04678

[4] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," vol. 14, 01 2014, pp. 1532–1543.

[5] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," 01 2016, pp. 1480–1489.

[6] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. [Online]. Available: http://arxiv.org/abs/1810.04805

[7] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," *CoRR*, vol. abs/1911.02116, 2019. [Online]. Available: http://arxiv.org/abs/1911.02116