|  |
| --- |
| **Classification of Music Lyrics by Genre** |
|  |
| **Shiraz Chakraverty and Dan Price** |
|  |
|  |
|  |
|  |

Abstract

|  |
| --- |
| 000  001  002  003  004  005  006  007  008  009  010  011  012  013  014  015  016  017  018  019  020  021  022  023  024  025  026  027  028  029  030  031  032  033  034  035  036  037  038  039  040  041  042  043  044  045  046  047  048  049 |

In the field of Natural Language Processing (NLP), the machine classification of text into human taxonomies is a formidable task. Machine understanding of language in general is complex. We cannot yet fully understand how the human brain does it but strides have been made in NLP. In the arts, the classification of music into genres inherits some of its nuance from the human cultural context the music originates. Yet, even casual listeners can often identify the assigned genre of a piece of music from written lyrics alone for popular genres. Engendering machines with a similar ability presents with its challenges. With that understanding, newer and emerging algorithms and word embeddings are able to offer classifications useful for organizations doing language tasks at a large scale.

We would like to explore the advantages and disadvantages of various methods in text classification papers we have selected and report our observations, learnings, and a recommendation. With the focus of deep learning, it is clear that neural networks tend to work better than previously used models. We would like to explore this path and provide some comparative analysis. We would also like to offer commentary on the larger issue of algorithmic bias in the setting of music genre classification.

Credits

Introduction

Rolling Stone magazine tells the story of a viral Internet hit song “Old Town Road” by a heretofore-unknown artist Lil Nas X. “Old Town Road” emerged on a social music video sharing site TikTok. It also sparked success on music sharing service SoundCloud. Partly motivated by promotion on social network Instagram from pop sensation Justin Bieber, the song simultaneously debuted on Billboard’s Hot 100 Chart, Hot Country Songs chart, and the Hot R&B/Hip-Hop Songs chart. However, after some time, Billboard elected to remove the song from its Hot Country Songs chart claiming the song “does not merit inclusion on Billboard’s country charts” because “it does not embrace enough elements of today’s country music to chart in its current version” [7]. Controversy over genre labeling ensued. The novelty and notoriety of the hit single ultimately led to famed country artist Billy Ray Cyrus appearing on an “Old Town Road” remix thereby cementing the songs relationship with the country music genre independent of the Billboard decision.

|  |
| --- |
| 050  051  052  053  054  055  056  057  058  059  060  061  062  063  064  065  066  067  068  069  070  071  072  073  074  075  076  077  078  079  080  081  082  083  084  085  086  087  088  089  090  091  092  093  094  095  096  097  098  099 |

In its editorial, Rolling Stone signaled that antiquated definitions of music genres highly correlated to race are to blame for Billboard’s classification controversy. In an age where codification of bias in artificial intelligence systems is under inspection, the conversation surrounding “Old Town Road” can be considered an important litmus test for the role of algorithmic classification in music. As artificial intelligence weaves its way into modern life, it is not unreasonable to consider the contribution machines play in the classification of styles of music. Machines already play a role in the recommendation of music. Genre is an important feature in music recommendation systems.

Text classification is a broadly applied task in the field of natural language processing. Filtering of spam email and sentiment analysis are canonical, highly utilized examples of text classification in application. We seek to evaluate some of NLP’s gains on music lyrics. In this paper, we aim to determine the applicability of machine learning and deep learning algorithms on the task of classifying music lyrics by genre.

Song lyrics exhibit structure distinct from other text documents. Lyrics are often organized in discrete sections of choruses, verses, and bridges. Lyrics often align with a defined musical tempo with regularly occurring patterns. Various academic and industry teams have tried approaches to this space. Efforts to understand music, both sonically and semantically through sound, lyrics, and metadata have coalesced in a subfield known as Music Information Retrieval (MIR).

No single effort has been very successful in finding a stable method that performs significantly well to tackle the lyrical genre classification problem. Algorithms such as Support Vector Machines, k-Nearest Neighbors, and Naive Bayes have all been used in lyrical classification but they all have very low accuracy in comparison to other NLP tasks on other datasets. We explore the application of new and emerging algorithms and models to the lyrics genre classification task. We can then, perhaps, get a machine’s take on how “Old Town Road” should be classified.

Dataset

Data for the lyrical classification problem is hard to come by due to copyright and other original content protection requirements. Artists and music labels do not usually publish lyrics with audio. With the rise of digital music and streaming, websites have emerged that build the infrastructure to publish crowd-sourced lyrics for ad revenue. Despite the public sourcing of the lyrics, public access to the entire dataset remains limited. Fortunately, a Kaggle user published a dataset of over 300,000 lyrics from a crawl of lyric website Metrolyrics.com.

The initial 98 megabyte dataset is downloadable as a comma delimited file. The columns in the file include an index, song title, release year, artist, genre, and lyrics. We are particularly interested in the text in the lyrics column for features and the genre column for labels. The lyrics are a string with carriage returns denoting an end of line. Statistics specific to the overall structure of the lyrics, such as line length, add value to the classification task. Lyrics with less than 100 words are filtered from the final dataset.

The original dataset includes disproportionate representation of genres. The Rock genre comprises nearly one-third of the dataset with over 100,000 records. The least populated genre is folk with just over 2000 records. Due to the skewed distribution of classes, we sampled 1000 records from seven genres. The genres include Jazz, Other, Hip-Hop, Not Available, Rock, Pop, and Country. We utilized Sci-Kit learns “train\_test\_split” method to create an 80/20 split of the train and test data.

Algorithms

While linear and kernel models rely on good hand selected features, deep learning architectures attempt to prevent this by letting the model learn important features themselves. However, not much research has looked into the performance of these deep learning methods with respect to the genre classification task on lyrics. Here, we attempt to understand this situation by extending the deep learning ideas on text classification to the particular case of lyrics. Previous non-neural lyrical classifiers struggled to achieve a classification accuracy any higher than 50%. We can see evidence of this here.

Because genre classification is a language classification task, we utilize high dimensional word embeddings as language representation in neural networks. Due to the expressive and poetic nature of music lyrics, we apply BERT (Bidirectional Encoder Representations from Transformers) to the task of lyrics genre classification.

An important idea in NLP is the use of dense vectors to represent words. To learn these word vectors a variety of methods have been proposed. A successful methodology proposes that similar words have similar context and thus that these vectors should be learnt through their context, such as in the word2vec model propose the GloVe method which combines global matrix factorisation and local context window. We tried using this method and did not find a major improvement.

Results

Discussion

Acknowledgments

The acknowledgements should go immediately before the references. Do not number the acknowledgments section. Do not include this section when submitting your paper for review.

References

Alfred. V. Aho and Jeffrey D. Ullman. 1972. *The Theory of Parsing, Translation and Compiling, volume 1*. Prentice-Hall, Englewood Cliffs, NJ.

American Psychological Association. 1983. *Publications Manual.* American Psychological Association, Washington, DC.

Ashok K. Chandra, Dexter C. Kozen, and Larry J. Stockmeyer. 1981. [Alternation](http://dl.acm.org/citation.cfm?doid=322234.322243). *Journal of the Association for Computing Machinery*, 28(1):114-133. https://doi.org/10.1145/322234.32224.

Association for Computing Machinery. 1983. *Computing Reviews*, 24(11):503-512.

James Goodman, Andreas Vlachos, and Jason Naradowsky. 2016. [Noise reduction and targeted exploration in imitation learning for abstract meaning representation parsing](http://aclweb.org/anthology/P16-1001). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, pages 1–11. <https://doi.org/10.18653/v1/P16-1001>.

|  |
| --- |
| 600  601  602  603  604  605  606  607  608  609  610  611  612  613  614  615  616  617  618  619  620  621  622  623  624  625  626  627  628  629  630  631  632  633  634  635  636  637  638  639  640  641  642  643  644  645  646  647  648  649 |

Dan Gusfield. 1997. *Algorithms on Strings, Trees and Sequences*. Cambridge University Press, Cambridge, UK.

Mary Harper. 2014. [Learning from 26 languages: Pro- gram management and science in the babel program](http://aclweb.org/anthology/C14-1001). In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*. Dublin City University and Association for Computational Linguistics, page 1. <http://aclweb.org/anthology/C14-1001>.

Alexander V. Mamishev and Murray Sargent. 2013. *Creating Research and Scientific Documents Using Microsoft Word*. Microsoft Press, Redmond, WA.

Alexander V. Mamishev and Sean D. Williams. 2010. *Technical Writing for Teams: The STREAM Tools Handbook*. Wiley-IEEE Press, Hoboken, NJ.