VERSE VERSUS CHORUS: STRUCTURE-AWARE FEATURE EXTRACTION FOR LYRICS-BASED GENRE RECOGNITION

Maximilian Mayerl[1] **Stefan Brandl**[2,3] **Günther Specht**[1]
Markus Schedl[2,3] **Eva Zangerle**[1]

[1] Department of Computer Science, Leopold-Franzens-Universität Innsbruck, Austria
[2] Institute of Computational Perception, Johannes Kepler Universität Linz, Austria
[3] Human-centered AI Group, AI Lab, Linz Institute of Technology (LIT), Austria

[1] {firstname.lastname}@uibk.ac.at [2,3] {firstname.lastname}@jku.at

ABSTRACT

Lyrics-based genre recognition aims to automatically determine the genre of a given song based on its lyrics. Previous approaches for this task have commonly used textual features extracted from the entirety of a song's lyrics, neglecting the inherent structure of lyrics consisting of, for instance, verses and choruses. Therefore, we pose the hypothesis that features extracted from different parts of the lyrics can have significantly different predictive power. To test this hypothesis, we perform a series of experiments to determine whether models trained on features taken from verses and choruses perform differently for genre recognition. Our experiments indeed confirm our hypothesis, showing that generally, using features extracted from verses leads to higher performance than features extracted from choruses. Digging deeper, we found that this is especially true for *pop* and *rap* songs. *Rock* songs show the opposite effect, with features extracted from choruses performing better than those taken from verses.

1. INTRODUCTION AND RELATED WORK

In music information retrieval, genre recognition (also known as genre prediction or genre classification) is the task of automatically classifying the genre of a given song by assigning it to one or more predefined genres. This has a variety of applications, including the organization of music collections into easily recognizable categories, or using genre information as a feature in recommender systems. Over the years, many approaches have been developed for this task, most of which focus on audio-based genre recognition, i.e., using information extracted from the song's audio signal [1]. Less, but still substantial, work has also been done on lyrics-based approaches, which use features extracted from a song's lyrics (e.g. [2–5]). Lastly, hybrid approaches combining both audio and lyrics information have also been proposed (e.g., [6,7]), and it has been shown that audio and lyrics features are complementary and that using both together can lead to increased model performance. The approaches making use of lyrics information cover a wide range of feature types as well as underlying machine learning models.

For instance, Neumayer and Rauber [6] explored using lyrics features in conjunction with low-level audio features to detect the genre of songs. For the lyrics, they extracted feature vectors using the popular bag-of-words model and weighted the words using *tf-idf* weighting. They then performed classification via support vector machines. Mayer et al. [2] relied only on lyrics and extracted *tf-idf* weighted bag-of-words features, rhyme, and part-of-speech features as well as general statistical text descriptors for their approach. These features were then used to perform genre recognition using different machine learning models. Ying et al. [3] used information of the parts-of-speech used in a song's lyrics to recognize both genre and mood of a song. They used these features to train and evaluate three different machine learning models, namely support vector machines, k-nearest neighbor, and naive Bayes.

What these, and most other, lyrics-based approaches have in common is that they treat a song's lyrics as a uniform document, extracting features from the entirety of the song's lyrics. However, in reality, song lyrics have a structure. They consist of different parts, which can be divided into categories such as *intro*, *verse*, *chorus*, *bridge*, or *outro*. Leveraging such structural information, Tsaptsinos [4] proposed using a hierarchical neural network model using an attention mechanism. To the best of our knowledge, this is the only work that directly seeks to exploit song structure for genre recognition, making use of the hierarchical structure of song lyrics (words forming lines, lines forming segments, and segments, in turn, forming the song). Their results show that their model, which due to its attention mechanism can automatically learn on which parts of a song it should focus, outperforms existing approaches that extract and use features uniformly across the whole song. This shows that the location of features within a song plays an important role in genre recognition. However, their model applies attention at the word, line, and segment level only, and does not incorporate higher structural elements like *verse* or *chorus*. Fell

and Sporleder [5] used a variety of textual features for three distinct music classification tasks, including genre recognition. The features they employed include features describing information about a song's structure, like the number of repeated segments in a song and whether the song contains a chorus. Their results for genre recognition suggest that including such features can increase genre recognition performance, and therefore that song structure plays a role in determining a song's genre.

Inspired by those results, we formulate the following hypothesis: The structure of a song's lyrics plays a substantial role in genre recognition, and extracting features from different parts of the lyrics can lead to a significantly different performance of genre recognition models. To this end, we make the following contributions: (1) We construct a dataset of lyrics that contains the required information about lyrics' structure; (2) We perform a series of experiments, covering both feature sets as well as machine learning algorithms used in the past for genre recognition, and show that there indeed exists a significant difference in predictive performance between features extract from the verses of a song as opposed to the same features extracted from the choruses; and (3) We examine how that difference in performance depends on the concrete machine learning algorithm and feature set used, as well as on the genre.

The remainder of this paper is structured as follows. Section 2 explains how we created the dataset required for our experiments. In Section 3, we provide a detailed overview of our experiments. Following that, we discuss our results in Section 4 and finally provide a summary and outlook to future work in Section 5.

2. DATASET

To investigate the impact on classification performance of features from different structural elements of lyrics, we require a dataset that contains both lyrics data with structure information (i.e., information on which structural part of the song— verse, chorus, etc.—any particular line in the lyrics belong to) as well as genre tags. Since no such dataset is publicly available, we describe the creation of such a dataset in the following and subsequently, provide a statistical description of its contents.

2.1 Dataset Creation

We used the popular LFM-2b dataset [8] to obtain an initial list of songs. Using this list of songs, we then obtained lyrics data from *genius.com*. We chose this source because it not only provides lyrics for a large number of songs, but it also has an active community of users who annotate lyrics with structure information. In addition to lyrics, they also provide tags for many of the songs in their database. We use the primary tag as given by *genius.com* as our genre tag, and then filtered the list of songs such that only songs with one of the top five most occurring genre tags — *pop*, *rock*, *country*, *rap*, and *r&b* — remained. These steps provided us with an initial dataset of 2,135,504 songs with both genre and lyrics information.

Property	Value
Number of songs	295,416
Number of artists	39,357
Number of tokens in all choruses	193,696,032
Number of tokens in all verses	195,567,571
Average number of choruses per song	3.46
Average number of verses per song	2.37

Table 1: Summary statistics of our dataset.

We then performed a series of cleaning and transformation steps on this initial dataset. First, we expanded repeating parts of the lyrics that were given in short form; i.e., lyrics on *genius.com* frequently use annotations like *[x2]* to indicate that a given part (paragraph or line) in the lyrics should be repeated. We removed those repeating annotations and duplicated the corresponding parts in the lyrics text accordingly. Following that, we removed all other annotations that do not correspond to structural parts of the lyrics; for instance, instrument annotations or singer information. In the next step, we used `langdetect` version `1.0.9` as well as `polyglot` version `16.7.4` to remove all songs with non-English lyrics from the dataset. Finally, since the dataset at that point contained duplicates—i.e., songs with identical lyrics—we removed those. These duplicates exist because LFM-2b sometimes contains multiple versions of the same song, like covers by a different artist or versions recorded during live events. To decide which song among a given set of duplicates to keep, we used the number of listening events according to LFM-2b and kept the copy with the highest listening count. We chose this approach since the copy with the highest listening count is also most likely to have the most complete set of genre tags.

Following these cleaning and transformation steps, we split the lyrics of each song into its constituent structure parts. For this, we used the structure annotations provided by *genius.com*. These annotations specify which part of the song the subsequent text belongs to, with that part extending until the next structure annotation or until the end of the song. They often also give some additional information, like who sings the given part or which number the part has in the song. Examples of such annotations include *[Chorus]* or *[Verse 1: Austin Brown]*. For our splitting, we considered the following set of structural annotations: *verse*, *chorus*, *intro*, *outro*, *bridge*, *hook*, *refrain*, *interlude*, and *drop*. We also combined *chorus* and *refrain* and mapped both of those annotations to *chorus*. For our subsequent experiments, we considered only the *verse* and *chorus* parts of the songs in the dataset, but we retained the other parts for potential future work.

Subsequently, we removed all songs that did not consist of at least two parts. This gave us an intermediate dataset consisting of 416,945 songs. Finally, since our experiments focus on *verse* and *chorus*, we created a final dataset containing only those songs which have at least one verse and at least one chorus, yielding a total of 295,416 songs.

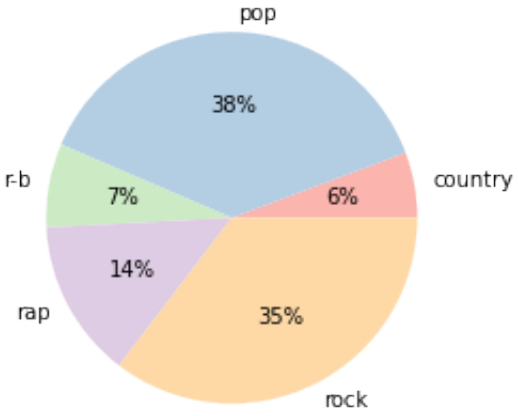

Figure 1: Genre distribution for primary genre.

2.2 Dataset Statistics

The final dataset consists of 295,416 songs created by 39,357 different artists. A summary of the dataset contents is given in Table 1. The total number of tokens (i.e., unigrams) contained in the choruses and verses for all songs is 193,696,032 and 195,567,571, respectively. This means that the amount of textual content for both chorus and verse is mostly balanced, ensuring that any difference in performance we may observe between using only chorus or verse does not stem from imbalanced training data. The distribution of primary genre labels in the dataset is given in Figure 1.

3. EXPERIMENTAL SETUP

The goal of this work is to determine whether there exists a difference in the predictive power between features extracted from different structural parts of a song's lyrics. For this, we carried out a series of experiments using different sets of textual features and different classification algorithms, each trained and evaluated once on only the *verse* and once only on the *chorus* parts of the songs in our dataset.

As explained in Section 2, we limited our final dataset to only songs which contain both at least one *chorus* and one *verse* part. This was done because, to get comparable results, we need to perform all our experiments on the same set of songs. As *chorus* and *verse* are the most common parts of a song, limiting our experiments to these two parts ensures that we have a dataset of sufficient size left to train and evaluate our models.

3.1 Feature Sets

Our experiments were performed using ten different sets of textual features commonly used in the literature (e.g., [2, 5–7, 9]). Those features together capture a wide variety of information about the lyrics, including content, grammatical structure, sound structure as well as complexity. Before using them in our experiments, we standardized them by scaling them to unit variance. We describe the feature sets used in the following:

- **Bag-of-words**: Classic bag-of-words features, counting the occurrences of word sequences of a given length, also known as n-grams, in the lyrics. In our experiments, we use unigrams, bigrams, trigrams, as well as (1,3)-grams, which is a combination of all word sequences of lengths one to three. To limit memory consumption of the resulting feature vectors, we only used the top 20,000 most common sequences appearing in the data. We also only consider sequences that appeared in at most 80% of all songs. This was done to filter out very common terms like function words. Such bag-of-words features have been frequently used in lyrics-based music classification, including for genre recognition, before (e.g. [2, 5–7, 9]).

- **Rhyme features**: A set of nine features describing the rhymes and alliterations present in the lyrics. The rhyme features (numbers of couplets, clerihews, alternating rhymes, and nested rhymes, percentage of rhymes in the text, and the number of unique rhyme words) were originally used by Mayer et al. [2] for lyrics-based genre recognition. Since alliterations play an important role in rap lyrics [10], we additionally added features describing alliterations (numbers of alliterations of length two, three, and four or longer).

- **Readability features**: A set of 13 readability metrics: Flesch reading easy, SMOG index, Flesch-Kincaid grade, automated readability index, Coleman-Liau index, Dale-Chall readability score, Linsear Write score, Gunning Fog index, Fernandez-Huerta score, Szigriszt-Pazos score, Gutierrez-Polini score, Crawford score, and the number of difficult words (i.e., words that are not on the Dale-Chall list of easy words). Readability features have been used for genre classification on normal texts, e.g. by Falkenjack et al. [11]. We included them in our experiments since we expect that they also carry useful information for lyrics.

- **Lexical features**: A set of 32 general lexical features, including token count, character count, repeated token ratio, number of unique tokens per line, average token length, average number of tokens per line, line count, unique line count, blank line count, blank line ratio, repeated line ratio, counts for specific symbols (exclamation mark, question mark, colon, etc.), count of digits, ratio of punctuation to the whole text, stop word count, stop word ratio, and hapax/dis/tris legomenon ratios. Different combinations of such lexical features have frequently been used for genre recognition (e.g., [2, 7, 9]).

- **Lexical diversity features**: A set of five lexical diversity metrics: measure of textual lexical diversity (MTLD), Herdan's C, Summer's S, Dugast's U and Maas' a.

- **Part-of-speech features**: A set of five features measuring the frequency of specific parts-of-speech within the lyrics: pronoun, adjective, adverb, noun, and verb frequency. Features describing the distribution of specific parts-of-speech within lyrics have been used for genre recognition for example by Mayer et al. [2], Mayer and Rauber [7], and Fell and Sporleder [5].

- **Morphological features**: A set of six features describing morphological properties of the lyrics: past tense ratio, -ing form ratio, comparative and superlative ratios for adjectives, and comparative and superlative ratios for adverbs. The ratio of verbs in past tense has been used for genre recognition before by Fell and Sporleder [5]. Since this was shown to be a valuable feature, we added the other five features to capture additional information about the morphological properties of a song's lyrics.

Since our goal is to determine potential differences between *verse* and *chorus* for individual feature types, we did not include a combination of all features in our experiments. It has already been shown before that different textual features carry orthogonal information, and combining them leads to increased performance [9].

3.2 Classification Algorithms

We repeat our experiments with different classification models to ensure that any difference in performance we measure between features extracted from different parts of a song is not only due to a specific property of any of the classification algorithms. Concretely, we chose the following algorithms: random forests, support vector machines, and feed-forward neural networks. All of these models are widely used in the machine learning community and have been shown to work well on many different tasks, including specifically genre recognition (e.g. [2–6, 9]).

For all three algorithms, we used the implementations provided by scikit-learn [1] version `1.0.2`. For support vector machines, we chose scikit-learn's `LinearSVC`, which uses LIBLINEAR [12], and followed the recommendation in the sklearn documentation to set `dual=False`, since in our case the number of training samples is significantly higher than the number of features. All other parameters for the used algorithms were left at their default values (100 estimators and Gini impurity for `RandomForstClassifier` and one hidden layer with 100 neurons and ReLU activation for `MLPClassifier`). Note that we did not perform a grid search to find the best hyperparameters, since our goal is only the determine the difference between features extracted from verses and choruses, not to get the best possible performance from the models.

[1] `https://scikit-learn.org/`

4. RESULTS AND DISCUSSION

We investigated every combination of song part (chorus or verse), feature set, and machine learning algorithm. Training and evaluation were done using 5-fold cross-validation. For the cross-validation, the data was shuffled before splitting it into folds. Feature extraction was done completely separately for the different song parts. For models trained and evaluated on the songs' verses, feature extraction only considered text located within the verses of the songs in the dataset, and equivalently for models trained and evaluated on the songs' choruses.

For evaluation, we chose the F_1 score to quantify the performance of each model. Since our dataset is imbalanced in terms of genres, we will focus our discussion on the macro-averaged F_1 scores, so as to not over-weigh the importance of the most common genres. However, for the sake of full transparency, we also report the micro-averaged F_1 scores for each model. Since the goal of our experiments is to determine whether there exists a performance difference between models trained on choruses and verses, we also performed statistical significance tests whenever our results indicated that the model trained on choruses performed better than the corresponding model trained on verses, or vice versa. For this, we employed a one-sided t-test ($p = .01$) on the scores of the individual cross-validation folds of both models, using the alternative hypothesis that the better overall score was indeed greater than the lower score.

We provide the results obtained in Table 2. We first take a look at the general performance of the models and feature sets. As is expected, we observe a substantial variance in performance for different machine learning models and features. The overall best performance in terms of the macro-averaged F_1 score is achieved by the support vector machine model trained on (1,3)-gram features extracted from the songs' verses, with an F_1 score of 0.5108. The best performance of the random forest and neural network model, respectively, was 0.4296 and 0.5082, both also for (1,3)-gram features trained on verses. The ten feature sets also exhibit substantial variability in performance across different machine learning models. This is especially true for part-of-speech features as well as morphological features. These two feature sets show a significantly lower performance when used with a support vector machine as opposed to random forests or neural networks. Looking at the higher score, obtained for the models trained and evaluated on the songs' verses, part-of-speech features achieved a score of 0.1872 against 0.2970 and 0.2888, and morphological features achieved a score of 0.1694 against 0.3154 and 0.2914.

Next, we examine the difference in performance between models trained and evaluated on choruses and verses, respectively. The results draw a clear picture: models trained and evaluated on verses consistently outperform the equivalent models trained and evaluated on choruses. We observe a difference in performance for every single combination of machine learning algorithm and feature set, at least in terms of the macro-averaged F_1 score. The

Feature Set	Random Forest			Support Vector Machine			Neural Network		
	Chorus	Verse	Diff.	Chorus	Verse	Diff.	Chorus	Verse	Diff.
F_1 macro									
unigrams	.377 (±.002)	.428 (±.002)	.0506†	.414 (±.002)	.502 (±.002)	.0880†	.419 (±.003)	.505 (±.003)	.0858†
bigrams	.364 (±.000)	.416 (±.003)	.0522†	.396 (±.002)	.485 (±.001)	.0888†	.396 (±.003)	.479 (±.005)	.0824†
trigrams	.328 (±.002)	.385 (±.001)	.0562†	.333 (±.002)	.422 (±.001)	.0894†	.342 (±.003)	.419 (±.003)	.0772†
(1,3)-grams	.379 (±.002)	.430 (±.002)	.0502†	.432 (±.002)	.511 (±.002)	.0792†	.424 (±.002)	.508 (±.003)	.0846†
rhyme	.254 (±.002)	.318 (±.002)	.0638†	.200 (±.001)	.280 (±.001)	.0796†	.230 (±.004)	.307 (±.009)	.0776†
readability	.263 (±.001)	.371 (±.002)	.1078†	.224 (±.001)	.355 (±.001)	**.1308**†	.264 (±.006)	.360 (±.002)	.0964†
lexical	.359 (±.002)	.400 (±.002)	.0412†	.291 (±.002)	.363 (±.001)	.0720†	.360 (±.004)	.403 (±.001)	.0430†
lexical diversity	.253 (±.002)	.351 (±.001)	.0980†	.195 (±.000)	.319 (±.000)	.1242†	.245 (±.002)	.333 (±.001)	.0878†
part-of-speech	.223 (±.001)	.297 (±.001)	.0738†	.180 (±.000)	.187 (±.001)	.0068†	.207 (±.005)	.289 (±.002)	.0816†
morphological	.201 (±.002)	.315 (±.001)	**.1146**†	.164 (±.002)	.169 (±.000)	.0056†	.181 (±.003)	.291 (±.005)	**.1100**†
F_1 micro									
unigrams	.538 (±.002)	.577 (±.003)	.0396†	.526 (±.002)	.566 (±.002)	.0402†	.484 (±.003)	.548 (±.003)	.0644†
bigrams	.516 (±.000)	.559 (±.003)	.0432†	.504 (±.001)	.548 (±.000)	.0444†	.472 (±.003)	.532 (±.004)	.0602†
trigrams	.462 (±.001)	.516 (±.003)	.0534†	.460 (±.001)	.509 (±.001)	.0486†	.434 (±.004)	.490 (±.002)	.0566†
(1,3)-grams	.541 (±.002)	.581 (±.001)	.0402†	.525 (±.002)	.567 (±.002)	.0420†	.492 (±.003)	.552 (±.003)	.0598†
rhyme	.409 (±.002)	.449 (±.002)	.0404†	.425 (±.001)	.456 (±.002)	.0308†	.433 (±.002)	.461 (±.003)	.0286†
readability	.422 (±.001)	.499 (±.001)	**.0772**†	.439 (±.001)	.511 (±.002)	**.0716**†	.453 (±.001)	.513 (±.002)	.0596†
lexical	.486 (±.003)	.525 (±.001)	.0388†	.473 (±.003)	.519 (±.001)	.0464†	.493 (±.002)	.526 (±.001)	.0332†
lexical diversity	.363 (±.001)	.434 (±.002)	.0712†	.425 (±.002)	.478 (±.001)	.0522†	.376 (±.002)	.482 (±.002)	**.1068**†
part-of-speech	.392 (±.001)	.440 (±.001)	.0482†	.400 (±.001)	.406 (±.001)	.0054†	.408 (±.002)	.448 (±.002)	.0400†
morphological	.385 (±.001)	.435 (±.001)	.0502†	.388 (±.002)	.388 (±.001)	.0000	.396 (±.002)	.443 (±.002)	.0468†

Table 2: Summary of the results of our experiments. For all numbers, leading zeros were omitted for space reasons. *Diff.* is the change in F_1 score between verse and chorus. Numbers in parentheses are the standard deviation between the five cross-validation folds. Bold numbers indicate the highest and underlined numbers to lowest amount of change for each combination of feature set and machine learning algorithm. † indicates that the difference in performance is statistically significant.

micro-averaged F_1 scores show almost the same picture, with one notable exception: morphological features used with a support vector machine achieved the same score for *chorus* and *verse*. The exact difference in performance again varies depending on the machine learning algorithm and feature set. The biggest absolute difference in terms of macro-averaged scores is seen with support vector machines using readability features. Trained and evaluated on choruses, this model achieves an F_1 score of 0.2244, as opposed to a score of 0.3552 for verses. This constitutes an absolute change in score of 0.1308. The biggest relative difference is also shown by support vector machines, but for lexical diversity features. For this model, the score for verses is 0.3188 as opposed to 0.1946 for choruses, for a change of 0.1242, which is a relative change of 63.82%. The lowest difference in terms of macro-averaged F_1 scores is observed for morphological features used with support vector machines. The difference between *chorus* and *verse* for this model is 0.0056, which is also the lowest relative difference of 3.42%. For micro-averaged F_1 scores, the lowest change is 0, as mentioned before, also for morphological features using support vector machines.

To get a better sense of how the difference in performance depends on the used machine learning algorithm or feature set, we computed the average difference in macro-averaged F_1 scores between *chorus* and *verse* along both of these dimensions. The results for this are given in Table 3. We can observe from these that the performance difference

Average over ...	Average Difference
Machine Learning Algorithms	
Random Forest	0.0708
Support Vector Machine	0.0764
Neural Network	**0.0826**
Feature Sets	
unigrams	0.0748
bigrams	0.0745
trigrams	0.0743
(1,3)-grams	0.0713
rhyme	0.0737
readability	**0.1117**
lexical	0.0521
lexical diversity	0.1033
part-of-speech	0.0541
morphological	0.0767

Table 3: Average amount of change in the macro-averaged F_1 score between *chorus* and *verse*, averaged over machine learning algorithm and feature set. Bold numbers indicate the highest average differences.

Genre	readability			lexical		
	RF	SVM	NN	RF	SVM	NN
country	0.0054	0.0000	0.0011	-0.0448†	0.0001	-0.0076
pop	0.0298†	0.0367†	0.0360†	0.0110†	0.0271†	0.0193
r&b	0.0048	0.0000	-0.0021†	0.0271†	-0.0113†	-0.0583†
rap	0.4995†	0.6587†	0.4878†	0.3009†	0.3820†	0.2950†
rock	-0.0001	-0.0402†	-0.0412†	-0.0154†	-0.0383†	-0.0383

Table 4: Performance differences for the feature sets *readability* and *lexical* for individual genres in terms of F_1 scores. The values in the table are computed as the differences between chorus and verse, with positive numbers indicating that models trained on verses performed better, and negative numbers indicating that models trained on choruses performed better. † indicates that the difference in performance is statistically significant. Abbreviations: RF = random forest, SVM = support vector machine, NN = neural network.

does not seem to vary much with the machine learning algorithm. The biggest difference here is obtained by neural networks, with an average of 0.0826, while the smallest change is seen for random forests, with an average of 0.0708. In contrast, performance differences between feature sets are more substantial. The biggest average score difference there is produced by *readability*, with an average of 0.1117, while the smallest difference is seen for *lexical*, with an average of 0.0521.

Those results confirm our initial hypothesis: The predictive performance of genre recognition models is indeed significantly different depending on which parts of a song we extract the features from. Concretely, results show that features extracted from the verses of songs perform significantly better than those extracted from the choruses. As mentioned in Section 2, the amount of textual content in choruses and verses is almost identical for our dataset. We can therefore rule out that this difference is caused by different amounts of training data for the model. We also conducted our experiments with various machine learning algorithms and feature sets, showing that the changes in performance are also not due to any particular properties of specific features or algorithms. We can therefore conclude that these differences are caused by a more fundamental difference in information content between choruses and verses.

Finally, we aimed to investigate how this observed difference in performance depends on the concrete genre. To this end, we took the feature sets with the biggest and smallest difference — *readability* and *lexical* — and computed per-genre F_1 scores for them, again for the same three machine learning algorithms. We then, as before, computed the difference in performance between verse and chorus. These per-genre differences are given in Table 4. In this table, positive numbers indicate that the model trained on verses performed better, while negative numbers indicate that the models trained on choruses performed better.

Looking at these results, we can see a varied picture of the different genres. For *country*, we observe that the differences change between positive and negative, which indicates no clear tendency for whether verses or chorus perform better for this genre. The differences for countries are also not statistically significant, with one exception (random forests using *lexical* features). For *r&b*, there is also no clear tendency, with differences likewise varying between positive and negative. For *pop*, we consistently see better performance when using verse features, ranging from 0.011 to 0.0367, with five of the six observed differences being statistically significant. *Rap* is the genre for which we observe by far the largest difference in performance. Features extracted from verses clearly outperform those extracted from choruses for this genre, with differences between 0.295 and 0.6587, all of which are significant. Finally, for *rock*, we find the opposite behavior, with features extracted from choruses outperforming those extracted from verses. The statistically significant differences here range from -0.0154 to -0.0412.

5. CONCLUSION AND OUTLOOK

Building on earlier results that the structure of lyrics may play a role in genre recognition, we formulated the hypothesis that features extracted from different structural parts of a song lead to significant differences in predictive performance for genre recognition models. Through a series of experiments, we confirmed this hypothesis and showed that, depending on genre, features extracted from songs' verses can perform better than those extracted from choruses. Looking closer at how performance varies with the concrete genre, we found that this is especially true for the genres *pop* and *rap*. *Rock*, on the other hand, exhibits the opposite behavior, with classifiers using features extracted from choruses achieving better accuracy than those using features extracted from verses.

In future work, we will investigate whether we can find similar differences for other lyrics-based MIR tasks, like mood recognition or popularity prediction. Additionally, the observed differences might be used to construct better features or classification models which exploit the structure of song lyrics. Finally, we plan to incorporate audio-based features extracted from verses and choruses, respectively, and investigate whether those show similar behavior and whether they can complement lyrics-based features in a multi-modal classification setup.

6. REFERENCES

[1] B. L. Sturm, "A survey of evaluation in music genre recognition," in *International Workshop on Adaptive Multimedia Retrieval*. Springer, 2012, pp. 29–66.

[2] R. Mayer, R. Neumayer, and A. Rauber, "Rhyme and style features for musical genre classification by song lyrics." in *Proceedings of the International Conference on Music Information Retrieval*, 2008, pp. 337–342.

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

[4] A. Tsaptsinos, "Lyrics-Based Music Genre Classification Using a Hierarchical Attention Network." in *Proceedings of the 18th International Society for Music Information Retrieval Conference*, Oct. 2017, pp. 694–701.

[5] M. Fell and C. Sporleder, "Lyrics-based analysis and classification of music," in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 2014, pp. 620–631.

[6] R. Neumayer and A. Rauber, "Integration of text and audio features for genre classification in music information retrieval," in *European Conference on Information Retrieval*. Springer, 2007, pp. 724–727.

[7] R. Mayer and A. Rauber, "Musical genre classification by ensembles of audio and lyrics features," in *Proceedings of the International Conference on Music Information Retrieval*, 2011, pp. 675–680.

[8] A. B. Melchiorre, N. Rekabsaz, E. Parada-Cabaleiro, S. Brandl, O. Lesota, and M. Schedl, "Investigating gender fairness of recommendation algorithms in the music domain," *Information Processing & Management*, vol. 58, no. 5, p. 102666, 2021.

[9] M. Mayerl, M. Vötter, M. Moosleitner, and E. Zangerle, "Comparing lyrics features for genre recognition," in *Proceedings of the 1st Workshop on NLP for Music and Audio (NLP4MusA)*, 2020, pp. 73–77.

[10] C. Coscarello, "The word out: A stylistic analysis of rap music," Master's thesis, 2003.

[11] J. Falkenjack, M. Santini, and A. Jönsson, "An exploratory study on genre classification using readability features," in *Proceedings of the Sixth Swedish Language Technology Conference (SLTC 2016), Umeå, Sweden*, 2016.

[12] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *The Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.