HOW DOES THE SPOTIFY API COMPARE TO THE MUSIC EMOTION RECOGNITION STATE-OF-THE-ART?

Renato PANDA (panda@dei.uc.pt)[1, 2], **Hugo REDINHO** (redinho@student.dei.uc.pt)[1], **Carolina GONÇALVES** (mcarolina@dei.uc.pt)[1], **Ricardo MALHEIRO** (rsmal@dei.uc.pt)[1, 3] and **Rui Pedro PAIVA** (ruipedro@dei.uc.pt)[1]

[1]*CISUC*, *DEI*, **University of Coimbra**, Portugal
[2]*Ci2 – Smart Cities Research Center*, **Instituto Politécnico de Tomar**, Tomar, Portugal
[3]**Miguel Torga Higher Institute**, Coimbra, Portugal

ABSTRACT

Features are arguably the key factor to any machine learning problem. Over the decades, myriads of audio features and recently feature-learning approaches have been tested in Music Emotion Recognition (MER) with scarce improvements. Here, we shed some light on the suitability of the audio features provided by the Spotify API, the leading music streaming service, when applied to MER. To this end, 12 Spotify API features were obtained for 704 of our 900-song dataset, annotated in terms of Russell's quadrants. These are compared to emotionally-relevant features obtained previously, using feature ranking and emotion classification experiments. We verified that energy, valence and acousticness features from Spotify are highly relevant to MER. However, the 12-feature set is unable to meet the performance of the features available in the state-of-the-art (58.5% vs. 74.7% F1-measure). Combining Spotify and state-of-the-art sets leads to small improvements with fewer features (top5: +2.3%, top10: +1.1%), while not improving the highest results (100 features). From this we conclude that Spotify provides some higher-level emotionally-relevant features. Such extractors are desirable, since they are closer to human concepts and allow for interpretable rules to be extracted (harder with hundreds of abstract features). Still, additional emotionally-relevant features are needed to improve MER.

1. INTRODUCTION

During the beginning of the 21st century notorious changes have occurred on the way people access and consume music, movies and other media. Before technical advances such as ubiquitous internet access, digital compact audio formats or the massification of mobile devices, music consumption would swirl around physical media such as tapes, optical discs or vinyl. These were normally sold at local stores, which provided limited and normally region-specific catalogs. In the course of the last decade this paradigm has changed, with music access, available through streaming services, taking over music ownership.

The first internet-based music services, centered on illegal distribution, contributed to the decline of music revenues, which achieved its lowest point in 2014. Since then, music revenues have risen for 6 consecutive years[1] led by streaming services such as Spotify, Deezer, or Pandora [1].

Nowadays these services provide easy access to millions of songs with unprecedented convenience (e.g., Spotify offers over 70 million tracks as of December 31, 2020[2]). However, such massive amount of data requires better search and discovery mechanisms than simply searching by artist, title or genre. Spotify mitigates these issues by using several data-driven personalization methods, and manually curated playlists. These are mostly based on users' listen history, while less focus has been given to the audio content due to the complexity of the task.

Meanwhile, we know that music is a language to express emotions, with some considering it to be its primary function [2]. Thus, music emotion recognition (MER) researchers have been proposing computational models to uncover and exploit these relations automatically. It is known that Spotify and other industry players are interested in various music information retrieval (MIR) topics, e.g., Spotify is said to be planning to use the user tone to detect his/her mood and personalize music suggestions[3].

So, how does the MER current state-of-the-art research compare with the solutions of the aforementioned services? In this paper we explore this question and shed some light on possible paths for future research. To this end, we assess how the audio features provided by the Spotify API[4] compare with the features used in MER. First, 12 Spotify audio features were gathered for a subset of 704 songs from our 900-song dataset [3]. Several audio feature ranking and emotion classification experiments are then run using these, as well as the top 100 features identified experimentally by our team as emotionally-relevant in [3], to understand how these compare and complement.

Among others, we verified that the 12 Spotify features are worse at discriminating Russell's quadrants, although four of them proved relevant. While not being a magic

[1] With an expected drop in revenues due to COVID-19 [1]
[2] https://newsroom.spotify.com/company-info/
[3] https://www.bbc.com/news/entertainment-arts-55839655
[4] https://developer.spotify.com/documentation/web-api/

wand, the inclusion of extra audio features by Spotify might help improving their recommendation data, something that may even be already in place internally, while not available thought their public API.

This paper is organized as follows. Section 1 introduces the problem, motivation and objectives. Section 2 briefly describes the related work relevant to the topic. Next, the experimental setup, including dataset, audio features and classification strategies are presented in Section 3, while Section 4 discusses the observed results. Finally, Section 5 draws the conclusions and suggests future work.

2. RELATED WORK

Music Emotion Recognition is a subfield of MIR which aims to automatically extract emotional information present in music. The field bridges knowledge from areas as diverse as music theory, machine learning, digital signal processing and psychology. In very broad terms, a typical MER solution uses a source of musical data (e.g., audio signals, lyrics, scores) to understand the governing relations between its properties (i.e., features) and emotional cues (e.g., annotations, in supervised learning), deriving rules to classify the emotion in newer examples (i.e., unknown musical data) [4].

The relations between music and emotions have been a matter of study for long by psychologists, with many relations documented [5]. As an example, consonant harmonies and major modes are usually associated with positive emotions, while the opposite with negative ones. However, the mechanisms governing these are not fully understood yet, and some studies present contradictory results [5].

In the field of computer science, several digital signal processing algorithms have been developed to capture sound and music characteristics from audio signals. These have been originally proposed to solve specific problems (e.g., speech recognition) but soon were employed in other MIR subfields, with MER being no exception [6].

In this journey of recognizing emotions with computers, different emotion paradigms (e.g., categorical or dimensional) and related taxonomies (e.g., Hevner [7], Russell [8]) have been used. These, intertwined with one of the datasets proposed in the field (e.g., [9]–[11]) served as foundations to many works. From emotion classification using raw audio signals [3], [12], [13], symbolic notations [14], or lyrics [9], to multi-label classification (i.e., several emotions per clip) [15], [16], dimensional MER [10], music emotion variation detection [11], [17] or multi-modal approaches [13], [14].

The majority of these works follow a typical machine learning approach, with handcrafted features, testing different datasets and machine learning strategies. However, results have stagnated over time, with authors suggesting better solutions "should perhaps be more musical knowledge-intensive" [18] to narrow the so-called semantic gap [19], a view also supported by us [3].

An alternative (or even complementary) path to the approach with handcrafted features is deep learning (DL). DL has been gaining momentum due to the ever increasing computational power and big data. There, AI-powered feature engineering is used, by feeding the neural network directly with the dataset (e.g., in the form of spectrograms). Several MER studies have tested techniques such as convolutional and recurrent neural networks [17]. However, such solutions fell short of expectations (so far), in part due to the lack of massive high quality datasets, which are complex to obtain [3] – one of the open problems in MER.

2.1 Spotify

Spotify is the major music streaming service, with 345 million monthly active users and 155 million subscribers in 93 markets as of December 31, 2020[5]. The service is responsible for reshaping the way music listeners experience music nowadays, exchanging ownership for easier access to large catalogs and personalized recommendations. Nowadays the service revolves around playlists, with 4 billion of them interconnecting 70+ million songs.

With such massive amounts of data, the company was able to expand from a simple music streaming player to a data-driven personalization service that drives discovery and engagement, increasing in value with each new user. The effect it now has on new songs, artists and their earnings is immense, e.g., "being added to Today's Top Hits, a list with 18.5 million followers (…), raises streams by almost 20 million and is worth between $116,000 and $163,000" in additional revenue from Spotify alone [20]. Still, the question remains: how exactly does Spotify recommend personalized content to each user?

Although not fully documented to the public, several details are known. Traditionally, recommendations relied on collaborative filtering [21], a technique used to understand a specific user's music taste based on historical listening data from all users, for instance using implicit matrix factorization [22]. Such techniques are content-agnostic (i.e., do not use the audio signals), relying only on users' consumption patterns. This fact is also its Achilles heel: they are unable to recommend new and unpopular songs given the lack of listening data – known as the *cold-start problem*, relevant when ~40,000 tracks are added daily[6].

To overcome this, Spotify includes additional sources of information, driven by the acquisition of The Echo Nest in 2014 – a music intelligence service providing automatic data extraction from songs by web crawling (e.g., metadata, lyrics, reviews), and digital signal processing techniques on the audio signal itself. Among others, it is able to estimate the danceability of a song or its valence.

Moreover, Spotify is known to use deep learning techniques to crunch metadata and audio signals for better recommendations. This began with Dieleman's work with deep convolutional neural networks [23] and has been evolving since then. Nowadays Spotify provides datasets and contributes with research in diverse areas from recommendation to user modeling or music creation[7]. Despite their advances, one major hurdle persists regarding MER

[5] https://newsroom.spotify.com/company-info/

[6] https://haulixdaily.com/2019/05/spotify-40000-tracks-per-day

[7] https://research.atspotify.com/

– the absence of quality emotion annotations that can help to better understand and predict what makes a happy or sad song.

3. EXPERIMENTS

The system provided by Spotify has been evolving over the years, fusing different sources of data (i.e., historical usage data, music metadata, web crawling, high-level features from The Echo Nest and DL). Even so, it is unclear if the emotional content present in the audio signal is being captured. To shed some light on this, we tested the high-level features provided by the Spotify API in a typical MER problem and compared them with the state-of-the-art.

In brief, we built on our previous work [3], where a 900-clip dataset was used to predict Russell's quadrants [8], identifying and proposing novel emotionally-relevant features. Here, we adopt a similar strategy, adding the 12 audio features provided by the Spotify API[8]. Each step in this direction is described in the following subsections.

3.1 Dataset

The original dataset contains 900 audio clips (up to 30 sec) annotated with Russell's quadrants (i.e., Q1 to Q4, representing respectively happiness, tension/aggression, sadness, and calmness). Both samples and metadata were sourced from AllMusic[9] and balanced (quadrants and genres). The AllMusic mood tags were matched against Warriner's norms of valence and arousal for English words and transformed into quadrant annotations, following a manual validation by volunteers. Further details in [3].

Crawling the Spotify service returned 704 of the 900 songs, for which audio features were obtained and used.

3.2 Audio Features

In general terms, a feature describes a characteristic part of something, it may be the composer, genre, its tempo, duration, or even more abstract statistics of the signal itself [24]. In this study we use computational audio features proposed in the literature or provided directly by Spotify.

3.2.1 Literature Features

From the audio clips, a total of 2719 features were initially extracted using MIR Toolbox, Marsyas and PsySound3 audio frameworks, as well as a set of novel features proposed in our previous work [3]. This high number is caused by the duplication of features across frameworks, as well as the summarization of time series into several statistics. These were then reduced by excluding features whose values had zero variance, as well as pairs of features with correlation higher than 0.9, as detailed in [3].

Next, the ReliefF algorithm [25] was used to identify and rank features according to their emotional relevance. Then, emotion classification experiments with the top 100 of these features achieved an F1-measure of 76.4% [3].

In this work we use these 100 features for the subset of 704 songs[10].

3.2.2 Features provided by the Spotify API

Spotify provides 12 audio features through its API[11]:

- Acousticness – whether the track is acoustic
- Danceability – how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity
- Energy – a perceptual measure of intensity and activity based on dynamic range, perceived loudness, timbre, onset rate, and general entropy
- Instrumentalness – whether a track contains no vocals (> 0.5 indicate instrumental tracks, with confidence increasing as it approaches 1.0)
- Key – the key the track is in according to standard Pitch Class notation
- Liveness – indicates presence of an audience in the recording (> 0.8 provides strong likelihood that the track is live)
- Loudness – the overall loudness in decibels (dB)
- Mode – the modality (major or minor) of a track
- Speechiness – presence of spoken words
- Tempo – overall estimated tempo in beats per minute (BPM)
- Time signature – estimated overall time signature (meter) of a track
- Valence – describes the musical positiveness conveyed (higher valence sounds more positive, e.g. happy, cheerful)

These 12 high-level audio features were obtained for our 704 tracks and used in the experiments described next.

3.3 Feature Selection and Emotion Classification

Having the audio features, our next step was to understand which of these were more suited to our quadrants classification problem by using the ReliefF [25] algorithm. Several reasons weighted in favor of this particular algorithm, namely, not being as CPU intensive as forward feature selection (which performs exhaustive classification tests) and the fact that it provides a rank / weight for each feature. At each iteration of the algorithm, a random song is selected and the K closest songs of the same class (i.e., quadrant) and of different classes are picked (using the Euclidean distance between feature vectors). The weight of each feature is then adjusted based on this distance between same vs. different class instances. Three different rankings were computed:

1. Ranking of our 100 features
2. Ranking of the 12 Spotify API audio features
3. Ranking of all 112 features combined

[8] https://developer.spotify.com/documentation/web-api/

[9] https://www.allmusic.com/

[10] https://github.com/renatopanda/TAFFC2018

[11] https://developer.spotify.com/get-audio-features/

The main goal with the first one was to re-rank and understand the changes in our top 100 features, since the original order was computed with the complete dataset (900 songs), against 704 now. The second ranking helps understanding which Spotify API features are more emotionally-relevant and better discriminate our dataset. Finally, ranking the 112 features together gives us a better understanding on how they compare and work together.

Using these rankings, several classification tests were run. Here, Support Vector Machines (SVM) was selected given its better results in the MER field [4] and our prior experience [3]. To this end, John Platt's implementation of sequential minimal optimization (SMO) for training SVMs, provided by the Weka[12] framework, was used with 10 repetitions of 10-fold cross-validation.

4. DISCUSSION

This section discusses the outcomes of the experiments described previously, regarding the dataset reduction, feature ranking and emotion classification[13].

4.1 Dataset Analysis

Our original dataset contains 900 audio clips, balanced across quadrants (225 clips each) and genres. However, 196 of these were not found in Spotify. The missing songs are spread across all quadrants, as shown in Figure 1, with Q2 (tense/anxious) affected the most (58 clips eliminated, now 167) and Q4 (calm/relaxed) on the other end of the spectrum (lost 37, now with 188).

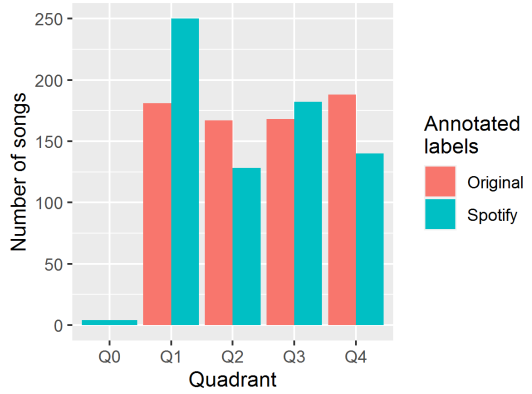

Figure 1. Distribution of the 704 clips across quadrants (Q0 are songs with neutral energy and/or valence).

Spotify estimates both energy and valence (EV) of each song. These are the two dimensions that define Russell's circumplex model of emotion [8]. In reality, energy is not exactly arousal, but serves as its surrogate. By transforming them into quadrants we can understand song distribution across quadrants (Figure 1). As illustrated, Spotify seems skewed towards Q1, with 250 of 704 songs considered happy. On the other hand, only 128 end up in Q2 (tense, anxious).

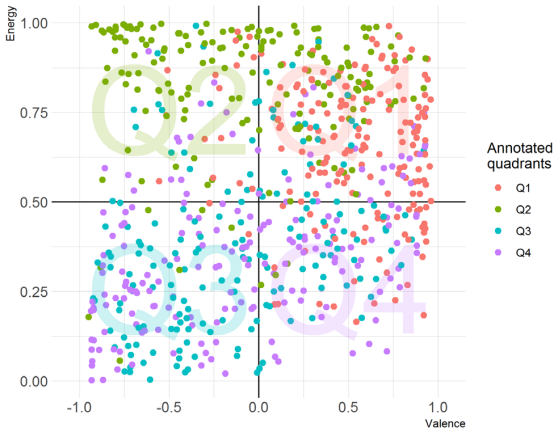

Figure 2. Songs according to Spotify's EV values (colored according to the original dataset annotations).

To better grasp these differences, Figure 2 presents the 704 songs placed in the Russell's plane, colored according to the original annotations. A perfect scenario would be each color (original quadrant) contained inside a single plane quadrant (Spotify quadrant), which is far from observed. Despite the lack of accuracy, this visualization uncovers interesting tendencies. Namely, the energy metric is more accurate, since most songs originally tagged with Q1 and Q2 (red and green) are placed in the top half of the plot, while the remaining are in the opposite end. The same cannot be said about valence, since, for instance, many Q2 songs (anxious/tense, in green) have positive valence, and Q4 songs (calm/relaxed, in purple) negative valence.

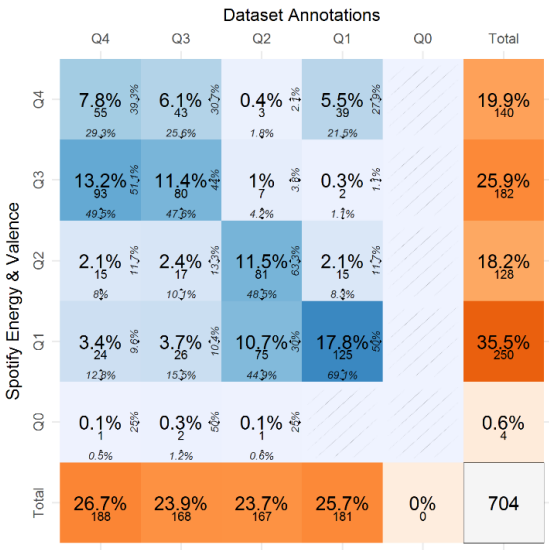

Figure 3. Original quadrants (annotations) vs. Spotify energy-valence based quadrants confusion matrix (each tile showing counts, overall percentage, row percentage and column percentage).

[12] https://www.cs.waikato.ac.nz/ml/weka/

[13] Data available at: https://github.com/renatopanda/SMC2021

These differences were somewhat expected given that: i. Spotify values (i.e., energy and valence) are computed from the audio signal, while the original dataset was validated by humans, and ii. predicting valence from audio is an harder problem, still to be fully addressed in MER [3]. In other words, it demonstrates that existing audio feature extractors still need to be further studied and improved.

To conclude, Figure 3 provides a more analytical view of this. Here we can see that Q1 is where we find more agreement (125 clips), while Q2 songs (originally 167) are mostly spread by Spotify between Q1 (75) and Q2 (81). A similar situation happens with Q3 and Q4, with the majority of these placed in Q3 by Spotify (93 + 80).

4.2 Standard Features' Ranking and Classification

The next step was to understand how the elimination of songs from the dataset (196 songs dropped) influenced both features' relevance and emotion classification.

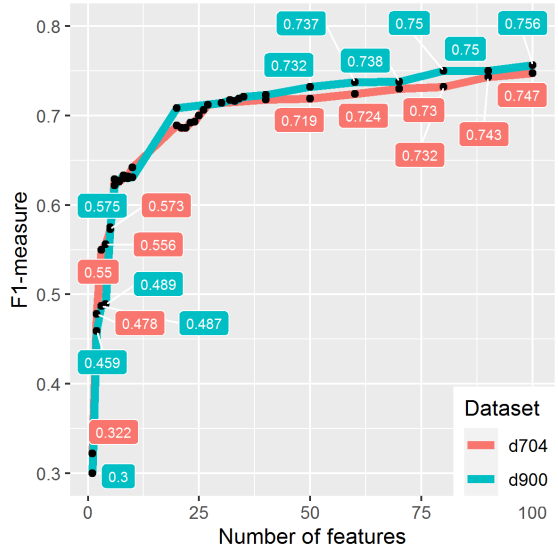

Figure 4. Classification results for 900 vs. 704 instances using the obtained feature ranking.

To this end we used the Weka ReliefF implementation to re-rank the top 100 features obtained previously using the 704 songs subset. Given the nature of ReliefF, several differences in the feature order were expected, and several reasons contribute to this. First, one-fifth of the songs were removed and ReliefF estimates the weight of each feature based on the Euclidean distance between features of randomly picked instances' assigned to distinct classes. Removing instances from the experiment will lead to different distances and thus different weights. Moreover, a different number of instances or even the seed (random) will lead to slight weight variations and ranking fluctuations (e.g., the top 10 features of each ranking might be different). Thus, the new ranking is useful to assess how the classification changes by using the best N features (e.g., top 10 to top 100), more than to assess which feature comes first.

Using this ranking, quadrants classification for both datasets (900 and 704) was tested using SVMs and is shown in Figure 4 ($C = 8$, $\gamma = 0.08$ for 704, $C = 7$, $\gamma = 0.1$ for 900).

As illustrated, the results are very close between the two sets, with the highest F1-measures separated by less than 1% (75.6% to 74.7%). The slightly lower result of 704 subset (with 21.8% fewer songs) might be indicative of the impact that the size of the dataset has in machine learning problems – the more quality data available, the better, more generalizable and robust are the identified patterns.

4.3 Spotify API Features' Ranking and Classification

Although only 12 features are provided, these are of much higher level than most features found in audio frameworks. As an example, Spotify's danceability is derived by combining information about tempo, rhythm, beat strength and regularity. Results of the ReliefF algorithm in our dataset (704) and the 12 features, in Figure 5, sheds some light on their contribution to discriminate across quadrants.

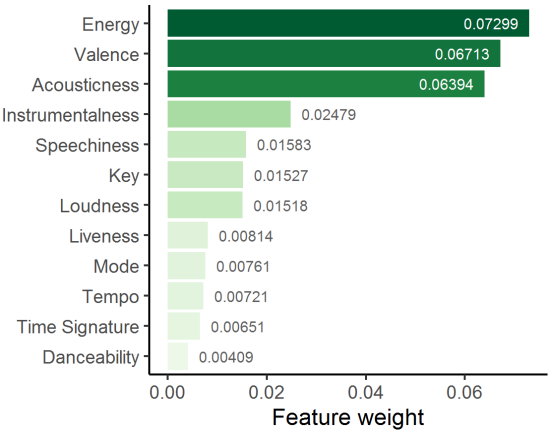

Figure 5. Influence of Spotify API features to separate songs among Russell's quadrants, according to ReliefF.

As expected, energy and valence audio features have the highest weight to the problem. These features are highly correlated with the quadrants, since they define the Russell's plane. If the algorithms that extract them from audio signals were perfect, they would probably be enough. Still, using only these in classification experiments achieves an F1-measure of 47.8%. In addition, the acousticness feature was also highly rated and thus required further inspection, as illustrated in Figure 6. Originally developed by The Echo Nest[14], it distinguishes between natural acoustic sounds (e.g., acoustic guitar, piano, unprocessed human voice – high acousticness) and mostly electric sounds (e.g., electric guitars, synthesizers, auto-tuned vocals – low).

Further analysis of the acousticness feature in our dataset uncovered two interesting facts. First, low acousticness,

[14] https://blog.echonest.com/post/53511313353/new-audio-attribute-acousticness

indicating mostly electric sounds, is more prevalent in Q1 (i.e., happy) and especially Q2 (i.e., tense/angry) songs (Figure 6-A). Secondly and as a result of the first, a high (negative) correlation was found between energy and acousticness (Figure 6-B).

Still regarding feature weights, on the other end of the spectrum we have features such as danceability, time signature, tempo and mode. While a lower weight for time signature (and also key) is not a surprise, one would expect features like mode [6] and danceability to have a different result. After all, major modes are usually associated with happiness and, intuitively, one would expect danceability to be too. In this case, it may happen that the algorithms used are not robust enough yet or caused by specificities of this dataset. In addition, it may be that these features work better in the presence of others currently missing.

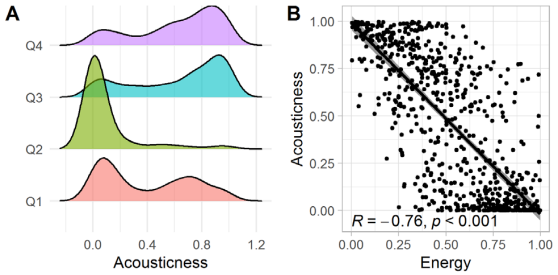

Figure 6. Distribution of acousticness values per quadrant (A), and correlation between energy and acousticness (B).

Figure 7. Classification results using the 12 Spotify API features (our top10 features added for comparison).

The classification results obtained using the 12 Spotify API features is presented in Figure 7, with the results from our best 10 features added for comparison. One notable fact is that with only one feature – energy, the classifier is able to correctly identify more than half of the songs, achieving an F1-measure of 44.2% (and accuracy of 50.9%), while our best feature is way behind (32.2%).

However, given the abovementioned findings (i.e., less relevant features and high correlations), the classification results using Spotify API features scarcely increase from the fourth to the twelfth feature.

4.4 Combining both sets of Features

After testing each set of features individually – Spotify API and ours, the logical step was to verify if the combination of both would improve the classification results. To this end, we combined both into a 112 features set and applied the ReliefF feature selection algorithm to assess the weight of each feature to the problem. Remarkably, four out of the 12 Spotify API features were among the top 5, while another four were placed in the bottom 5, as illustrated in Figure 8. The remaining were ranked in positions 52nd, 74th, 92nd, and 99th.

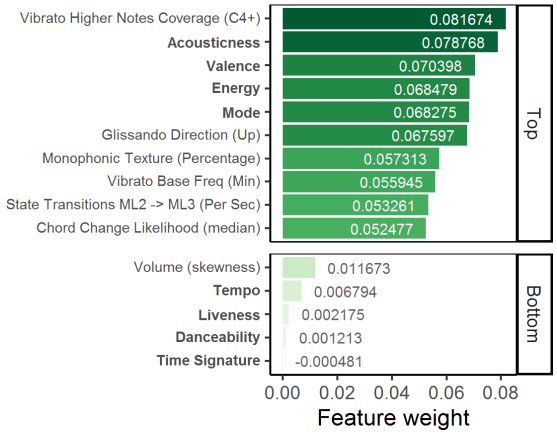

Figure 8. Best and worst rated features according to ReliefF (Spotify API features in **bold**).

In addition, the top 10 features included three features related to expressive techniques, two related to musical texture (proposed by us in [3]) and one to melody (from PsySound3). As expected, the three most important Spotify API features were again energy, valence and acousticness. Though, this time their order was reversed and each had slightly different weights. Moreover, mode was ranked fifth amongst the 112 features, obtaining a much higher weight than previously obtained when only 12 features were ranked (in Section 4.3).

To understand these differences we need to remember that ReliefF is a filter model – uses statistics extracted from the training data (i.e., Euclidean distance) and correlates them with the associated labels (i.e., quadrants). In brief, in each iteration a random instance (song) is selected and the two songs closer to it (one for each class) are also selected. This is based on the Euclidean distance between feature vectors, as if they were placed in an N-dimensional space. Then, the weight of each feature is readjusted using the differences between the feature values of the random song and the other two (of same and different class). This has several implications, two of those are: 1) a different number of features (dimensions) will influence the distance between instances and 2) the random selection of

songs to compute the weights (non-exhaustive) will lead to slight variations in feature weights over each run of the algorithm (with different random seeds). Both points shed some light on the slight order and weight differences, and might explain the increase in the importance of mode. Since mode is binary (major or minor), its relevance was lower when combined with other 11 features – especially considering that only 3 were of high relevance. Once the dimensionality increases and the number of relevant features increases, its own (i.e., mode) discrimination power is increased (interaction between features).

The next step was to use the combined ranking and repeat the classification experiments with SVMs (C = 0.07, γ = 4). As summarized in Table 1, no substantial differences were observed between results of our set of 100 features and the combined set of 112 features. Here, the more noteworthy were: 1) the higher F1-measure from Spotify API when only one feature is used; 2) the fact that with few features (less than 10), the combined set performs slightly better; and 3) the convergence in results as the amount of features increases (illustrated in Figure 9).

No Features	Spotify	Ours	Combined
1	44.2%	32.2%	31.4%
5	56.2%	57.3%	59.6%
10	56.1%	64.2%	65.3%
20	-	68.9%	68.9%
50	-	71.9%	71.8%
Best result (# features)	58.5% (11)	74.7% (100)	74.2% (100)

Table 1. Summary of the emotion classification results.

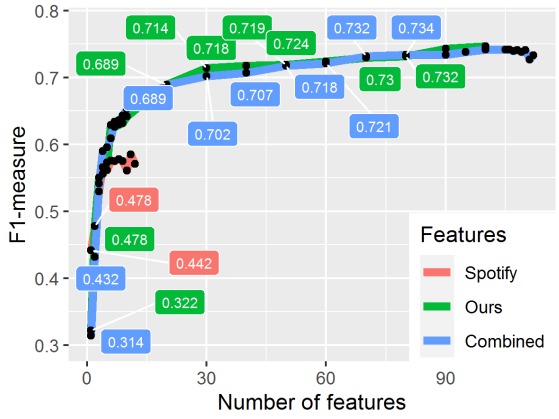

Figure 9. Classification results for each feature set.

The first is a consequence of the selected ranking algorithm, since the same feature is available in both Spotify and Combined sets (energy) but was only selected as first in the former. Although much more resource-intensive, one approach to mitigate this is by combining ReliefF (filter model type of feature selection) with a wrapper model (e.g., forward feature selection). Still, this is dissipated when more features are added. When the number of features in use increases (e.g., 5 to 10), we see the advantages of combining both sets.

As the number of features in use increases, the performance differences become negligible. One possible explanation is that the combination of some of the (lower-level) features in our top 100 may be able to capture similar emotional cues to the few higher-level proposed by Spotify API and thus their relevance drops in the combined set. As an example, the most relevant Spotify API feature (energy) combines dynamic range, perceived loudness, timbre, onset rate, and general entropy. Such characteristics were previously extracted separately to obtain our top 100 feature set [3].

5. CONCLUSIONS

This paper offered an analysis on the suitability of the Spotify API audio features to the music emotion recognition field. As part of this, its features were compared to audio features previously proposed in the scientific literature that are known to be emotionally-relevant.

From the experiments, several conclusions were drawn. First, three of the 12 Spotify API features were identified as highly relevant to emotion classification – energy, valence and acousticness. While the first 2 were expected, given their relation to the Russell's plane, acousticness – which was found to be highly correlated with energy, was somewhat new. Moreover, tense/anxious songs were almost exclusively low on acousticness (i.e., non-natural, electric sounds), information that complements our previous survey on emotionally-relevant audio features [6].

Secondly, the 12 features provided by the Spotify API are subpar to the problem in analysis, achieving only 58.5% F1-measure, when compared to the state-of-the-art (74.7%). While Spotify's goal is music recommendation and user taste modeling, we believe that the addition of emotionally-relevant audio features may improve the system, after all some argue that music's primary function is to express emotions [2]. Such idea might even be already in use, but just not exposed in their public API.

Finally, we believe that novel audio feature extractors, are needed to improve this as well as other MIR problems, since most MIR solutions are generic, "without relying on musically meaningful features" [18]. These novel features should be higher-level (i.e., closer to human knowledge), providing ways to uncover interpretable rules between emotions and an handful of audio cues, after all that is science's main goal – to explain and understand.

Acknowledgments

This work was supported by CISUC (Center for Informatics and Systems of the University of Coimbra). Renato Panda was supported by Ci2 - FCT UIDP/05567/2020.

6. REFERENCES

[1] IFPI, "Global Music Report 2020: The Industry in 2019," 2020. [Online]. Available:

http://www.idf.org/node/26452?language=es.

[2] A. Pannese, M.-A. Rappaz, and D. Grandjean, "Metaphor and music emotion: Ancient views and future directions," *Conscious. Cogn.*, vol. 44, pp. 61–71, Aug. 2016, doi: 10.1016/j.concog.2016.06.015.

[3] R. Panda, R. Malheiro, and R. P. Paiva, "Novel Audio Features for Music Emotion Recognition," *IEEE Trans. Affect. Comput.*, vol. 11, no. 4, pp. 614–626, Oct. 2020, doi: 10.1109/TAFFC.2018.2820691.

[4] X. Yang, Y. Dong, and J. Li, "Review of data features-based music emotion recognition methods," *Multimed. Syst.*, pp. 1–25, 2017, doi: 10.1007/s00530-017-0559-4.

[5] A. Gabrielsson and E. Lindström, "The Role of Structure in the Musical Expression of Emotions," in *Handbook of Music and Emotion: Theory, Research, Applications*, P. N. Juslin and J. A. Sloboda, Eds. Oxford University Press, 2011, pp. 367–400.

[6] R. Panda, R. M. Malheiro, and R. P. Paiva, "Audio Features for Music Emotion Recognition: a Survey," *IEEE Trans. Affect. Comput.*, pp. 1–1, 2020, doi: 10.1109/TAFFC.2020.3032373.

[7] K. Hevner, "Experimental Studies of the Elements of Expression in Music," *Am. J. Psychol.*, vol. 48, no. 2, p. 246, Apr. 1936, doi: 10.2307/1415746.

[8] J. A. Russell, "A circumplex model of affect," *J. Pers. Soc. Psychol.*, vol. 39, no. 6, pp. 1161–1178, 1980, doi: 10.1037/h0077714.

[9] R. Malheiro, R. Panda, P. Gomes, and R. P. Paiva, "Emotionally-Relevant Features for Classification and Regression of Music Lyrics," *IEEE Trans. Affect. Comput. – TAFFC*, vol. 9, no. 2, pp. 240–254, Apr. 2018, doi: 10.1109/TAFFC.2016.2598569.

[10] Y.-H. Yang, Y.-C. Lin, Y.-F. Su, and H. H. Chen, "A Regression Approach to Music Emotion Recognition," *IEEE Trans. Audio, Speech, Lang. Process. – TASLP*, vol. 16, no. 2, pp. 448–457, 2008, doi: 10.1109/TASL.2007.911513.

[11] A. Aljanaki, Y.-H. Yang, and M. Soleymani, "Developing a benchmark for emotional analysis of music," *PLoS One*, vol. 12, no. 3, Mar. 2017, doi: 10.1371/journal.pone.0173392.

[12] Y. Feng, Y. Zhuang, and Y. Pan, "Popular Music Retrieval by Detecting Mood," in *26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval – SIGIR 2003*, 2003, vol. 2, no. 2, pp. 375–376, doi: 10.1145/860435.860508.

[13] C. Laurier, "Automatic Classification of Musical Mood by Content-Based Analysis," Universitat Pompeu Fabra, 2011.

[14] R. Panda, R. Malheiro, B. Rocha, A. P. Oliveira, and R. P. Paiva, "Multi-Modal Music Emotion Recognition: A New Dataset, Methodology and Comparative Analysis," in *10th International Symposium on Computer Music Multidisciplinary Research – CMMR 2013*, 2013, pp. 570–582.

[15] T. Li and M. Ogihara, "Detecting emotion in music," in *4th International Symposium on Music Information Retrieval – ISMIR 2003*, 2003, pp. 239–240.

[16] B. Wu, E. Zhong, A. Horner, and Q. Yang, "Music Emotion Recognition by Multi-label Multi-layer Multi-instance Multi-view Learning," in *22th ACM International Conference on Multimedia – ACM MM 2014*, 2014, pp. 117–126, doi: 10.1145/2647868.2654904.

[17] M. Malik, S. Adavanne, K. Drossos, T. Virtanen, D. Ticha, and R. Jarina, "Stacked Convolutional and Recurrent Neural Networks for Music Emotion Recognition," in *14th Sound & Music Computing Conference – SMC 2017*, 2017, pp. 208–213.

[18] R. Scholz, G. Ramalho, and G. Cabral, "Cross task study on mirex recent results: An index for evolution measurement and some stagnation hypotheses," 2016.

[19] Ò. Celma, P. Herrera, and X. Serra, "Bridging the Music Semantic Gap," in *Workshop on Mastering the Gap: From Information Extraction to Semantic Representation, held in conjunction with the European Semantic Web Conference*, 2006, vol. 187, no. 2, pp. 177–190.

[20] L. Aguiar and J. Waldfogel, "Platforms, Promotion, and Product Discovery: Evidence from Spotify Playlists," Seville, 2018–04, 2018. doi: 10.3386/w24713.

[21] C. Johnson, "Algorithmic Music Recommendations at Spotify," 2014, [Online]. Available: https://www.slideshare.net/MrChrisJohnson/algorithmic-music-recommendations-at-spotify.

[22] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *IEEE International Conference on Data Mining*, 2008, pp. 263–272, doi: 10.1109/ICDM.2008.22.

[23] A. van den Oord, S. Dieleman, and B. Schrauwen, "Deep content-based music recommendation," in *Neural Information Processing Systems Conference (NIPS 2013)*, 2013, pp. 2643–2651.

[24] D. Huron, "What is a Musical Feature? Forte's Analysis of Brahms's Opus 51, No. 1, Revisited," *Online J. Soc. Music Theory*, vol. 7, no. 4, 2001.

[25] I. Kononenko, E. Šimec, and M. Robnik-Šikonja, "Overcoming the Myopia of Inductive Learning Algorithms with RELIEFF," *Appl. Intell.*, vol. 7, no. 1, pp. 39–55, 1997, doi: 10.1023/A:1008280620621.