Musical Track Popularity Mining Dataset: Extension & Experimentation[☆]

Ioannis Karydis[1], Aggelos Gkiokas[2], Vassilis Katsouros[2], Lazaros Iliadis[3]

[1]*Dept. of Informatics, Ionian University, Kerkyra 49132, Greece, karydis@ionio.gr*
[2]*Institute for Language & Speech Processing, Athena RIC, Athens 15125, Greece agkiokas@ilsp.gr, vsk@ilsp.gr*
[3]*Dept. of Civil Engineering, Democritus University of Thrace, Xanthi 67100, Greece liliadis@civil.duth.gr*

Abstract

Music Information Research (MIR) requires access to real musical content in order to test the efficiency and effectiveness of its methods as well as to compare developed methodologies on common data. Existing datasets do not address the research direction of musical track popularity that has recently received considerate attention. Moreover, sources of musical popularity do not provide easily manageable data and no standardised dataset exists for musical popularity research. To address these issues the Track Popularity Dataset (TPD) was created in a previous work. TPD provided (a) different sources of popularity definition ranging from 2004 to 2014, (b) mapping between different track/ author/ album identification spaces allowing use of different popularity sources, (c) information on the remaining, non popular, tracks of an album with a popular track, (d) contextual similarity between tracks and (e) ready for MIR use extracted features for both popular and non-popular audio tracks. This paper extends the TPD by (a) adding more readily computed features, (b) proposing feature & similarity definitions on popularity trends, (c) formulating common data mining scenarios on tracks' popularity and (d) presenting respective promising results.

Keywords: Music Information Research, Hit Song Science, Dataset, Track

[☆]This work is an extended version of [1]

1. Introduction

One of the most important requirements of Music Information Research (MIR) is access to pertinent musical content. The experimentation on this content mostly aims on the testing of the efficiency and effectiveness of the MIR methods, while providing reference for comparison of new and existing methods in order to show progress. In rare cases, the use of synthetic data can be helpful to the aforementioned use of data in MIR experiments, though music, being highly an artistic form of expression, does not always adhere to a set of deterministic rules that researchers could rely on in order to avoid the requirement for access to real musical content.

Accordingly, in MIR, as in most areas of scientific research, the collection, distribution and use of datasets is of great importance, despite the litany of legal issues [2] that may arise from such practices. Music data for the purposes of MIR usually refer to audio files of recorded performed musical pieces, symbolic representation of a piece, lyrics, metadata as well as contextual to the piece information mainly collected through social networks pertaining to the users' perception of or activities on the pieces. Thus, following the need for such content exchange and its intended use, MIR datasets additionally include commonly used derivative transformations of all the aforementioned musical information in order to avoid legal implications as well as to spare users of time and resources required for these to be produced.

Numerous datasets exist in MIR, as extensively described by Makris et al. [3], that cover a broad area of the domain, though none is immediately applicable for knowledge extraction from the popularity that musical pieces receive. The process of track popularity prediction prior to or during the initial period of a track's release has long been a requirement of the musical industry. Interestingly enough, the gains of such a prediction go far beyond the obvious benefits of allowing musical labels to identify financially interesting clients, as the whole

ecosystem (artists and listeners) also profits. Despite the aforementioned benefits, it was only after the commercial application by Polyphonic HMI [1] that the issue gained significant attention as a research direction, as early as 2005 [4].

1.1. Motivation

Existing commonly used services, such as Spotify[2], Billboard[3], iTunes[4] and Last.fm[5] that provide popularity of musical content do not offer easily manageable data. Spotify's localised charts, although provided an Application Programming Interface (API), have temporarily according to the service's community helpdesk, ceased to function as of approximately March 2015 and are still offline. Billboard's Hot 100 Chart does not offer an API but does provide the most long termed archives, dating back to August 9th, 1958 as well as machine-friendly (rss) methodology of accessing data. iTunes provides localised charts for numerous of the countries iTunes feature localisation as well as a machine-friendly (rss with xml/json) methodology of accessing data. Last.fm's localised charts do not offer an API, though Last.fm does indeed provide the aggregated number of listeners and playcounts for all available tracks.

To add to the difficulties of collecting track popularity information, each of the aforementioned services utilise their respective track identification space making collective use of multiple popularity sources rather difficult. Moreover, collecting just the tracks that exceed the popularity threshold, research cannot deal integrally with the separation of hits from non-hits as no information on non-hits is available, since the collected information only contains the degree of popularity. Finally, having access to the content of the audio files of the popularity chart is, among other parameters, very important in the selection of tracks' representative features that will lead to high quality predictions.

[1] `http://polyphonichmi.blogspot.gr/p/about-company.html`

[2] `https://spotifycharts.com` (previously known as `https://charts.spotify.com`)

[3] `http://www.billboard.com/charts/hot-100`

[4] `http://www.apple.com/itunes/charts/songs/`

[5] `http://www.last.fm/charts`

1.2. Contribution

To address the aforementioned requirements of Section 1.1, our previous work [1] introduced the Track Popularity Dataset (TPD), a collection of data on track popularity for the purposes of MIR, containing:

1. different sources of popularity definition ranging from 2004 to 2014,
2. information on the remaining, non popular, tracks of an album with a popular track,
3. a mapping between different track/author/album identification spaces that allows use of all different sources,
4. contextual similarity information between all popular tracks,
5. ready for MIR use extracted features for both popular and non-popular audio tracks,

Accordingly, the current work extends previous works in three axes: (a) the volume and type of TPD's readily computed features, (b) both the definition of novel feature and similarity for popularity trends as well all as the identification and formulation of interesting popularity related research scenarios, and (c) the experimental verification of the potential of the proposed dataset's descriptive capability as far as inferring musical popularity is concerned. In detail, this work significantly extends [1] by

1. updating related work by including 15 more recent related publications,
2. extending the TPD by adding more computed, ready-to-use and publicly available features,
3. proposing a feature and similarity definition on popularity trends,
4. proposing two interesting popularity related research scenarios,
5. presenting promising experimental results on the aforementioned scenarios.

The rest of the paper is organised as follows: Section 2 presents background information on Hit Song Science and related work, while Section 3 discuses the extended dataset, its creation processes as well as a detailed analysis of its

content. Next, Section 4 identifies a potentially interesting popularity prediction scenario and presents experimental results obtained using the dataset. Section 5 details future directions concerning the dataset that could ameliorate is usability and further support MIR research. Finally the paper is concluded in Section 6.

2. Background & Related research

This Section details necessary background information on the issue of musical track popularity prediction as well as related existing research.

2.1. Hit Song Science

Hit Song Science (HSS) refers to the MIR direction aiming in predicting the popularity of musical tracks, as presented in top-charts. A number of scenarios' parameters exist as to the prediction's prerequisites, such as the little or no availability of early popularity information, the granularity of the popularity definition, the type of input sources representing the musical tracks and many others.

Similarly, under the auspices of HSS numerous research tasks also take place: popularity pattern modelling, binary (hit/non-hit) or otherwise granulated popularity classification, tracks' future position on the popularity chart prediction given current position, popularity correlation to other activities (i.e. twitter posts, music search/download in peer-to-peer networks, etc), prediction of the popular track subset of an album and many more.

The ability to predict the popularity of musical tracks is of great importance to all parties involved in the musical content life-cycle, with just a few indicative scenarios including the following cases. Some of the creators can work reversely the process of HSS and focus on characteristics that make their songs more probable to be popular in addition to customised characteristics of listeners, markets or distribution channels. The music industry, aiming at maximum profit, could benefit by selecting the most promising of the works for publication as well as, given that popularity predictions can be attributed to specific profile

candidate consumers, modify accordingly its marketing plans. Moreover, music consumers indirectly increase enjoyment of listening by receiving music that the distribution channels have either selected to fit their profile or that is in general more probable to be of high popularity and thus more probable widely liked.

It is widely claimed that the breadth of characteristics that lead to the popularity of a musical piece exceeds the *per se* track's content i.e., the audio and lyrics. Factors such as artist preferential attachment [5], society and culture [4], changing musical tastes leading to evolving popularity pattern[6], psychological parameters on the reasons for preferring a track and listening exposure to tracks [7], associated video clip of the track[8] just to name a few, play also an important role.

Nevertheless, existing research in the area agree that, beyond the very hard to measure characteristics, quantifiable qualities of musical tracks that contribute to a track's popularity do exist [4, 9, 10, 11, 12, 13]. Accordingly, the main burden remains with the transformation representations of musical tracks that need to adhere to tracks' popularity pertinent attributes.

2.2. Existing Research

Existing related literature is mostly focused on methodologies for mining musical track popularity information. Accordingly, the proposal of this work, i.e. the creation of an integrated dataset for the purposes of testing musical track popularity mining information methods, is complementary to the aim of the works presented in the sequel.

In the first work on the area, Dhanaraj and Logan [4], utilise SVN & boosting classifiers on both acoustic and lyric information for the purposes of hit songs' separation from non-hits. Their aim is to determine if such a task is feasible or if hit song science claims are to be deemed as impossible, arriving after experimentation at the former.

Chon et al. [8] research for meaningful patterns within musical data while also attempting to predict both how long an album will stay in chart as well as a new album's position in chart on a certain week in the future based on sales'

data. The results presented therein indicate interesting correlations.

Pachet & Roy, in [14], and Pachet, in [7], describe an experiment aiming at validating the current state-of-the-art methods' capability to predict the popularity of musical titles based on acoustic and/or contextual features. Both these works suggest that the commonly used features for music analysis are not informative enough to offer judgement on notions related to subjective aesthetics.

In [5], Bischoff et al. propose the music pieces' success prediction by exploitation of social interactions and annotations using data mined from the Last.fm [6], reaching promising results.

In a differentiated scenario, the work of Koenigstein et al. [15] compares peer-to-peer file sharing information on songs to their popularity, from Billboard [7] charts, while indicating popularity trends of songs on Billboard having a strong correlation popularity on peer-to-peer network. Accordingly, they propose utilising this correlation to predict a songs' success on the popularity charts.

Similarly to [14], Ni et al. [6] on a slightly alternated research question argue the feasibility of popularity prediction, "given a relevant feature set", while also creating the website "Score a hit" [8].

The work of Kim et al. [16] proposes the collection of users' music listening behaviour from Twitter, based on music-related hashtags, for the purposes of predicting popularity rankings with results showing high correlation between users' music listening behaviour on Twitter and music popularity trend.

Singhi & Brown [17] propose a hit detection model based Bayesian networks on solely lyrics' features.

The work of Herremans et al. [9] focuses on the dance hit song classification problem using hit songs from 1985-2013. The dataset includes musical features and more advanced features from Echo Nest[9]. They propose various classifiers

[6] http://www.last.fm/

[7] http://www.billboard.com/

[8] http://www.scoreahit.com

[9] http://the.echonest.com/

to build and test dance hit prediction models, indicating promising results for the prediction of whether a song is a 'top 10' dance hit versus a lower listed position, thus concluding the capability of learning popularity from signals of musical data.

Nunes & Ordanini [10] focus on the relation between track popularity and the audible instruments to the listener, researching how timbre mixtures expresses listeners' liking. Using data from 1958-2012 Billboard's charts their experimental results show 2-instrument configurations mainly present consistently in top 1 hit songs while 3-instrument configurations leading to less popularity. Moreover, they also suggest that popular songs include vocals and deviation from typical number of instruments leads to popularity.

In [11] Nunes et al. make the case for the repetitive characteristics of the lyrical content of songs as the characteristic leading to consumers' adoption. Their experimentation shows that neither repeated exposure to songs nor features of melodic repetition are the key factors impacting the processing fluency on consumer choice. Moreover, analysing the top 1 hits from Billboard 1958-2012 indicates that songs with repetitive lyrics as having more chances of reaching the top position.

In their work [18] Frieler et al. test melodic features as input in a random forest classification algorithm to receive accuracy slightly better than chance. The dataset utilised therein comprises earworms (hits) while rest were from similar artists and UK chart positions not labelled as earworms. They conclude that for hit song classification the size of intrinsic features is not the most contributing factor in contrast to extra-musical cultural information.

Jensen & Hebert [19] propose a new feature, the Jensen Chroma Complexity (JCC), for the prediction of the year of origin of hit songs. JCC focuses on how popular music changes across time using harmonic analysis while experimentation reports promising results for predictability in time units of decades as well as classification of years according to harmonic tendencies.

The work [12] by Lee & Lee focuses on predicting popularity of music using the audio signal of songs to extract a feature vector including chroma, rhythm,

timbre and arousal complexity characteristics. Six temporal evolution measurements are defined therein and experimentation utilises multi-layer perceptrons with one hidden layer as classifiers on songs from Billboard with results indicating accuracies with significance level of 5% from random guess.

The authors of [13] propose features from both songs' lyrics and audio content for prediction of hits. Their feature-set comprises of audio features and lyrics-based rhyme, syllable and meter characteristics. Their database is derived from Billboard with varying definitions on the notion of popularity while results indicate that a combination of lyrics and audio features performed better in the identification of hits, though solely lyrics features were more useful in separating hits from non-hits.

Shulman et al. [20] examine various formulations of the use of items' adoption information from virtual social networks, in addition to items' content, for the task popularity prediction of an item, be it song, movie or tweet. The authors therein emphasise that the common practice of focusing on temporal features of early adoption offers no specific explanation on why items that become popular fast are more likely to achieve higher popularity in the end. Their results indicate that predictive models using temporal features achieve higher accuracy on various items types (network structure, early adopters' features and similarity) than all other feature types combined. Additionally, these models also generalised well, to the extent that models trained on any one item type performed with comparable accuracy on items from other types.

Finally, Burgoyne et al. [21] present a close to the theme of musical track popularity work studying musical content's "catchiness", or the "long term musical salience" of a piece. Despite the broader scope of the musical popularity prediction task, the correlation of catchiness to popularity is evident although most probably one directional, since numerous less-memorable top-chart tracks do exist.

The aforementioned existing research, with the exception of [14] and [7], have utilised different datasets to perform experimentation. The diversity of the utilised datasets in terms of size vary greatly as shown in Table 1. Ques-

Research	Music representation	Dataset size	Hit definition	Top charts time span
[4]	content-based audio; lyrics	1700 songs	Billboard top 1	Jan 1956 - Apr 2004
[8]	album sales data	291 albums	Billboard top 1-25	Sep 2002 - Jun 2006
[7], [14]	content-based audio; subjective contextual; objective metadata	32000 songs	HiFind popularity label: low, medium, high	?
[5]	subjective contextual	50555 songs	Billboard top 1, 3, 5, 10, 20, 30, 40, 50	Aug 1958 - Apr 2008
[15]	P2P queries	185598176 p2p queries, 200 songs	Billboard top 10, 20, 20, 30, 40, 50, 100	Jan 2007 - Jul 2007
[6]	content-based audio	5000 songs	Billboard top 5	1962 - 2011
[17]	lyrics	6815 songs	Billboard top 15, 25, 35	2008 - 2013
[16]	objectivecontextual; objective metadata	1806438 tweets, 168 songs	Billboard top 10, 20, 30, 40, 50	Nov 2013 - Jan 2014
[9]	content-based audio; objective metadata	697 OCC + 2755 Billboard songs	OCC & Billboard top 10, 20	Tracks' evolution: 1985-2013, Dance hit prediction: 2009-2013
[10]	content-based audio	2399 songs	Billboard top 1 & positions 90-100	1958 - Aug 2012
[11]	lyrics	1956 songs	Billboard top 1 & positions 90-100	1958 - Dec 2012
[18]	content-based symbolic	266 songs	Availability in Earwormery database	?
[19]	content-based audio	6394 songs	Billboard top 100	1941 - 2014
[12]	content-based audio	867 songs	Billboard 50 Rock Songs Chart (songs $\geq$ 3 weeks on chart)	Jun 2009 - Apr 2014
[13]	lyrics; content-based audio	6815 songs	Billboard Year-End Hot 100 (various definitions)	2008 - 2013
[20]	social network adoption activity	437k Last.fm users	various combinations of (a) threshold of adoptions for items and /or (b) time elapsed since item's introduction	users' start date on last.fm until February 2014

Table 1: Existing HSS research dataset details.

tion marks within Table 1 indicate information that was not available at the respective work, i.e. no time-span was provided for the collection of the dataset.

3. The Dataset

The TPD is a collection of information revolving around the notion of track popularity. Its aim is to provide an easy to use collection of information for the purposes of track popularity data mining research tasks. This Section details the creation process and content of the TPD.

3.1. Creation Process

In order to create the TPD, the potential information sources were separated into three distinct categories: the popularity sources, the metadata/content

sources and the contextual similarity source.

The selection of the popularity periods was made based on the availability of both popularity information from the sources and access to the tracks' content. Thus, from popularity sources Last.fm and Spotify all available popularity charts at the time of collection were amassed, that is from 17 September 2006 up to 28 December 2014 and 28 April 2013 up to 18 January 2015, respectively. From popularity source Billboard the last 10 years were collected, ranging from 03 January 2004 up to 24 January 2015.

Following the collection of the popular tracks from the popularity sources, the metadata/content sources Apple[10], Spotify[11], 7digital[12] were utilised in order to identify and get information for the albums of the collected popular tracks and then to gather information on the remaining, non-popular, tracks of each album.

Access to the content of the collection's tracks was based on the metadata/content sources' (Apple, Spotify and 7digital) 30 second previews clips as all three web services provide an API for the purposes of searching and streaming the audio clips. The collected files where converted to appropriate format in order to undergo feature extraction.

While performing the above mentioned information collection processes, it was confirmed that multiple identification spaces do indeed exist for all track/ author/ album entities. Accordingly, and in order to facilitate the interoperability of the collected information, exact match searches were performed in all sources producing thus a mapping between different track/ author/ album identification spaces. As not all sources engulf information on all collected data, the mapping is not complete, but nevertheless, far from sparse (∼55% of the matrix cells contain values). Content for the mapping was collected from both popularity sources and metadata/content sources.

[10] https://www.apple.com/itunes/affiliates/resources/documentation/itunes-store-web-service-search-api.html

[11] https://developer.spotify.com

[12] http://developer.7digital.com/

To enrich further the TPD, contextual information as to the similarity of the collection's tracks based on Last.fm's API *track.getSimilar* method were additionally included, providing similarity between tracks, based on listening data.

Finally, for each track of the TPD, four feature-sets extracted directly from the audio content are included in matlab variable MAT-files. The first feature-set, *feature-set A*, is based on jAudio and contains only single overall average and standard deviation values performed on all values of the features over all windows with window size 512 samples and 0% overlap between successive windows. The second feature-set, *feature-set B*, was created with MIRToolbox offering per window feature extraction with window size 1024 samples and 50% overlap between successive windows. The two feature-sets provide different levels of detail on the audio content in order to suit a broad range of applications. The third feature-set, *feature-set C* is based on the periodicity function of the tempo estimation method presented in [22]. The fourth feature-set, *feature-set D* is part of the TPD's extension and is oriented on beat features [23] and is based on the Beat Synchronous Chromas by adopting the method in [24].

3.2. The Content

The TPD contains 23.385 tracks of which, 9.193 are designated as popular by appearing in any of the popularity sources charts, while 14.192 are tracks that appear in one of the 1.843 albums of the popular tracks and are not designated as popular by any of the popularity sources. The popularity ratings records, contain the position of a track for a specific week, collected from Billboard are 57.800, while for Last.fm and Spotify are 43.300 and 6.500, respectively. Of the popular tracks, 1,5% are designated in all three sources of popularity, 5,9% in two sources and 92,6% in just one source. The discrepancy in proportions is due to the range of available data by the popularity sources. As far as the contextual similarity based on Last.fm's API *track.getSimilar* method is concerned, 78% of the popular tracks of the dataset have a degree of contextual similarity to other popular tracks of the dataset. As not all tracks' audio files were possible to be

found, the TPD contains audio derived features for ∼74% of the tracks.

Of the four feature-sets included in the TPD described in Section 3.1, *feature-set A* is meant as a small, less detailed feature set for fast and simple research applications. The features included in *feature-set A* are: overall standard deviation & overall average of spectral centroid (dimension: 1), spectral rolloff point (dim: 1), spectral flux (dim: 1), compactness (dim: 1), spectral variability (dim: 1), root mean square (dim: 1), fraction of low energy windows (dim: 1), zero crossings (dim: 1), strongest beat (dim: 1), beat sum (dim: 1), strength of strongest beat (dim: 1), strongest frequency via zero crossings (dim: 1), strongest frequency via spectral centroid (dim: 1), strongest frequency via fft maximum (dim: 1), MFCCs (dim: 13), LPCs(dim: 10), method of moments (dim: 5), partial based spectral centroid (dim: 1), partial based spectral flux (dim: 1), peak based spectral smoothness (dim: 1), relative difference function (dim: 1), area method of moments (dim: 10). The second feature-set, *feature-set B*, contains windowed MFCCs (dim: 13), rolloff (dim: 1), brightness (dim: 1), flux (dim: 1), zero crossings (dim: 1), inharmonicity (dim: 1), centroid (dim: 1), spread (dim: 1), skewness (dim: 1), kurtosis (dim: 1), flatness (dim: 1), entropy (dim: 1). The third feature-set, *feature-set C*, contains 276 target tempi. For each target tempo this feature-set contains eight energy bands and one chroma (dim: 9). This work extends the TPD by adding a fourth feature-set, *feature-set D*, that for each track contains solely 12 chromas (dim: 12).

In order to provide an aggregated glimpse of the popularity records of the dataset by contrasting the popularity sources, Figure 1 shows the normalised probability density of next week's rank increase/decrease (position change) given current position for all three sources of popularity. Moreover, Figures 2 and 3 show the probability density of rank position when entering and leaving respectively the top-100 popularity chart for all three sources of popularity.

3.3. Format & Usage

The dataset is divided into two separate parts: part a includes the relations/metadata of the tracks and their popularity while part b contains the files

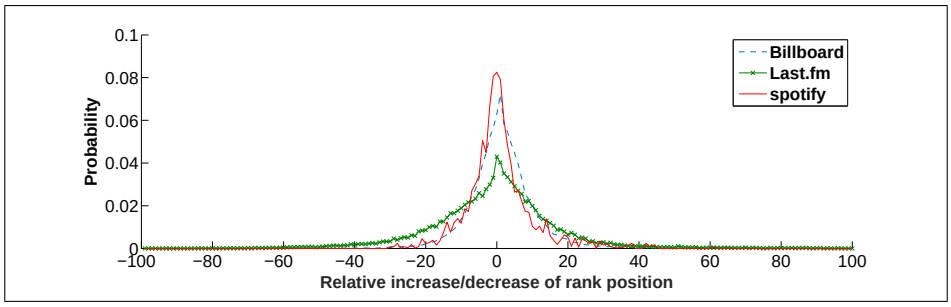

Figure 1: The normalised probability density of position change.

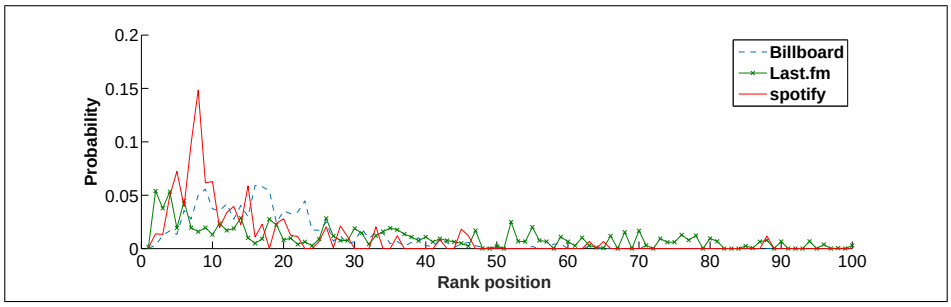

Figure 2: Popularity chart entry position probability density.

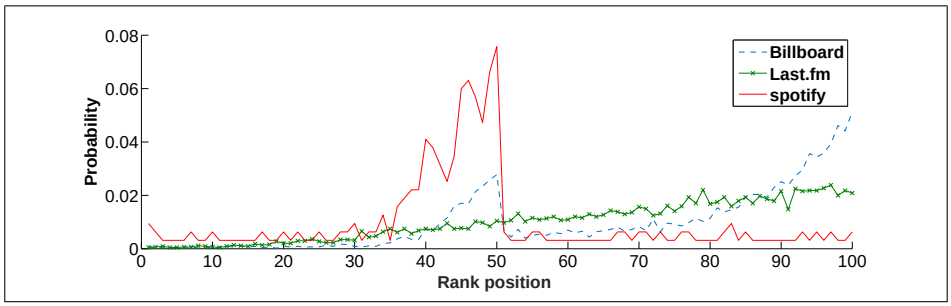

Figure 3: Popularity chart leave position probability density.

of the four feature-sets.

The first part is in the form of a relational database, the compact schema of which is shown in Figure 4. The archive of part a contains the SQL statements that will create the TPD database & tables and subsequently load all the information into the tables of an existing MySQL installation. Moreover, the contents of the first part are also provided in CSV format, in order to support fast use of the data and alleviate the necessity for a relational database. The second part consists of compressed archives of bz2 type that contain the feature-sets in a one file with features per track manner. The complete TPD can be downloaded from `http://mir.ilsp.gr/track_popularity.html`.

Figure 4: Schema for the metadata and the popularity of the tracks.

4. Popularity Prediction

In this Section we present popularity prediction experimentation on the TPD. The aim herein is to experimentally ascertain the potential of the proposed dataset's descriptive capability as far as inferring musical popularity is concerned. To this end, we use off-the-shelf popular machine learning algorithms for classification.

4.1. Popularity Profile & Similarity

The phenomenon of popularity is a rather complex not only in what promotes a track to higher levels, as described in Section 2.1, but also in what are the aspects that describe the popularity level. Accordingly, and bearing in mind that our aim is to provide for data mining from popularity information, we propose

the use of a popularity profile, i.e. a number of characteristics that describe the popularity a musical track has received as well as a similarity measurement for such popularity profiles.

Each under-consideration track k achieves n popularity positions p, where each p_i position is less or equal to a popularity-source defined threshold $thress$, in order for k to be in the chart, occurring every i time-length periods (chart announcements). All consideration takes place within a period of examination e that includes j number of i periods (e.g. for period of examination e equal to 52 weeks and chart-popularity announcements i periods equal to two weeks then j is equal to 26), thus leading to a timeseries of positions $P = (p_1, p_2, \ldots, p_n)$. Accordingly, we retain the following information:

Best position The best position of the track for the period it appears on the top chart list, $bestpos = \min(P)$,

Worst position The worst position of the track for the period it appears on the top chart list, $worstpos = \max(P)$,

Enter position The position of the track when firstly appeared on the top chart list, $enterpos = p_1$,

Weeks on chart The number of weeks the track appears on the top chart list, $weekson = n$,

Weeks off chart The number of weeks, within the period it appears on the top chart list, when the track is above the popularity threshold (off the top chart list), $weeksoff = j - weekson$,

Leave position The position of track the last week before leaving forever the top chart list in the time-frame of e, $leavepos = p_n$,

Average position The average position of the track for all weeks that appears on the top chart list, $averagepos = \frac{\sum_{i=1}^{n} p_i}{weekson}$.

Thus, a track's k vector profile is given by:

$$prof_k = (bestpos_k, worstpos_k, enterpos_k, weekson_k, \\ weeksoff_k, leavepos_k, averagepos_k)$$

It should be noted that sources can operate on different assumptions of $thress$ and i, and indeed Spotify provides weekly and daily top 200 most popular tracks, while most other services commonly use only weekly updates.

To calculate the distance between two tracks' profiles, $t1$ and $t2$ and given the exponential and linear weight vectors for each part of the profile $w1$ and $w2$, respectively, we propose the use of the following generic weighted difference:

$$diff_{t1,t2} = \sum w2 \circ (|w1 \circ (prof_{t1} - prof_{t2})|)$$

wherein the weight vectors are point-wise multiplied to the absolute value of differences between the profiles of the tracks associated.

4.2. Problem Definition

For the purposes of popularity prediction on the data of TPD, we present the following two problem formulations:

1. "Given N albums for which you know the popular track of each, predict the popular track of the unknown album k".
2. "Given the popularity time series of N songs, predict the popularity time series of an unknown song k".

These formulations cover two key areas of the usually performed data mining in track popularity, as described in detail in Section 2.1.

The first problem addresses the issue of identifying characteristics of musical tracks that support the separation of popular from non-popular tracks in a controlled environment. As the true such experimentation would require a, very hard to obtain, exhaustive list of all tracks that did not qualify for the popularity threshold, per each popularity source, we attempt the equivalent by minimising the comparison space in the manageable breadth of the non-popular tracks of

the album, were the popular track under consideration was published in. This pruning, additionally addresses the distinction between the same artist's tracks that where explicitly grouped together by the very same artist.

The second experiment examines the ability to learn from the profile of a popular track in order to predict the popularity of other tracks. The prediction is done with some or none extra information and with varying degree of initial popularity information on the tracks with unknown popularity. The first parameter of having or not extra information on the tracks aims in identifying the ability of the prediction process to map the extra information to the popularity space, and thus represents the core of the HSS process. The interesting twist in this case is the use of contextual information vs. objective features from the audio as extra information. The second parameter, addresses the breadth of initial popularity information availability aiming at simulating the case where a prediction is made for the time to come, after the release of a track.

4.3. Experimental Setup

To address the first problem formulation, we utilised an SVM classifier in order to differentiate between popular and non-popular tracks, for all the tracks that include features in TPD. Tracks are represented using the following alternative feature-sets: (a) the complete *feature-set C*, (b) the first 1000 components of Principal Component Analysis (PCA) on the complete *feature-set C*, (c) the first 100 components of PCA on the complete *feature-set C*, (d) the MFCCs of *feature-set A* and (e) the concatenation of (c) and (d). The experimentation results are based on five-fold cross validation on a subset of 5.000 tracks with 4.000 tracks being used for training the SVM classifier and the rest 1.000 tracks being used as the test set. PCA pre-processing is performed selectively in some of the experimentation sub-scenarios in order to verify the effect of the dimensionality of each feature-set in the results: as numerous of the features of each feature-set could be highly correlated, PCA allows for selection of features' combination that captures the most information possible while also reducing any potential noise. For the implementation, we have employed the libsvm library [25] using a

radial basis kernel and calibrated the parameters γ of the exponent and the cost on a validation set. The selection of SVM algorithm is based on its adoption in a number of related works, as described in Section 2.2, as well as to a number of advantages it exhibits: the lack of necessity for complex tuning of parameters, the increased ability to generalise even on small training corpora, as well as its learning ability in high dimensional spaces [4].

For the second problem formulation, we utilised Non-linear Auto Regressive (NAR) and Non-linear Auto Regressive with eXternal input (NARX) dynamic Neural Networks (NNs) for the prediction of a track's popularity. The NAR case covers for the scenario that aside to the popularity position timeseries, no extra information is provided for each track. The NARX case on the other hand deals with the scenario of providing to the prediction process extraneous information on the tracks considered. This extra information in our experimentation is (a) the features of each track as described by *feature-set A* (NARX-f) and (b) the popularity position timeseries of the contextually most similar track (NARX-c). The latter is based on the intuition that tracks deemed similar by users may receive similar popularity and thus, by providing the popularity position timeseries of the contextually most similar track as an input to the NARX NN, the mapping process is assisted.

In both NAR & NARX cases, a feed-forward back-propagation NN with one hidden layer was utilised containing a varying number of hidden neurons (ranging in $[1, 10]$) and delays (ranging in $[1, 30]$) in order to test the effect of the neuron and size of initial information availability, while the network performed one-step-ahead prediction. The experimentation also included the division of the dataset into training, validation of generality, and testing subsets in different sizes. In all experiments with the NN presented herein evaluation of the performance was only based on the testing subset. The learning function used was the LevenbergMarquardt back-propagation function, the output layer transfer function was the hyperbolic tangent sigmoid transfer function while the performance function was the Mean Squared Error (MSE), between the outputs and targets, performance function. As this is essentially a regression experi-

ment, we also calculated the Regression (R) values indicating the correlation between the outputs and targets of the NN.

4.4. Experimental Results

For the first problem formulation, the results obtained can be seen in the confusion matrices of Table 2. Table's 2 combined confusion matrices allow fast visualisation of the proposed algorithm's performance for all varieties of configurations (i.e. feature set scenario and/or use of PCA). Accordingly, each row of the matrix represents the instances in the predicted class, each column represents the instances in the actual class (ground-truth) while the columns are also grouped together according to the variety of configuration. In Table 3 we present the resulting measures, such as accuracy, specificity, etc. from the confusion matrices of Table 2. One may observe that in terms of overall accuracy the best method feature set is the combination of the rhythm features with PCA with the MFCCs that gives an accuracy of 55.22%. However, this is mainly due to the high specificity, i.e. 68.71%, which means the potential of classifying correctly the non-popular tracks rather than the precision of classifying correctly the popular tracks which is 28,53%. On the other hand, MFCCs demonstrate the highest precision rate with 45.92%, with the respective specificity showing the poorest result.

	Rhythm no PCA		Rhythm with PCA 100		Rhythm with PCA 100		MFCC		Rhythm with PCA 100 & MFCC	
Predicted → Ground truth ↓	**Popular**	**Non-popular**	**Popular**	**Non-popular**	**Popular**	**Non-popular**	**Popular**	**Non-popular**	**Popular**	**Non-popular**
Popular	13,22%	20,36%	11,86%	21,72%	11,44%	22,14%	15,42%	18,16%	9,58%	24,00%
Non-popular	28,00%	38,42%	24,78%	41,64%	23,86%	42,56%	29,58%	36,84%	20,78%	45,64%
Total	41,22%	58,78%	36,64%	63,36%	35,30%	64,70%	45,00%	55,00%	30,36%	69,64%

Table 2: Confusion matrices for the first problem formulation.

For the second problem formulation, Figure 5 shows the resulting ratio of R/MSE for all four approaches used, wherein regression R measures the correlation between outputs and targets (with R values of 1 meaning a close relationship, 0 a random relationship) and MSE being the Mean Squared Error, i.e. the average squared difference between outputs and targets (with lower values

	Rhythm no PCA	Rhythm with PCA 1000	Rhythm with PCA 100	MFCC	Rhythm with PCA 100 & MFCC
Accuracy	51,64%	53,50%	54,00%	52,26%	55,22%
Precision	39,37%	35,32%	34,07%	45,92%	28,53%
Sensitivity	32,07%	32,37%	32,41%	34,27%	31,55%
F1-measure	35,35%	33,78%	33,22%	39,25%	29,97%
Specificity	57,84%	62,69%	64,08%	55,47%	68,71%

Table 3: Performance measures resulting from the confusion matrices of Table 2 for the first problem formulation.

indicating better results and zero value no error). The x-axis is not continuous and designates the parameters used for each value of the methods' experimentation. Due to space requirements, these parameters are only shown in detail in Table 4 for the best result.

Approaches NARX-f MFCCs & NARX-f all use as extra information from *feature-set A* only the MFCCs and all features, respectively. The ratio of R/MSE was chosen in order to present, using a single value, the requirement for minimisation of the MSE value and maximisation of the R value. The results indicate the superiority of the NARX method utilising extraneous information for the prediction over the NAR approach that used solely the past values of the popularity of the track under examination. Of the alternatives of the NARX method, NARX-c, the contextual extraneous information, performs clearly better followed by the solutions that utilised solely the MFCC and all features, respectively.

Table 4 presents all the parameters of the NN experimentation for the best ratio R/MSE result of all approaches.

	MSE	R	Delays	Hidden neurons	Dataset division (training, validation, testing)
NAR	796.2	0.70	4	3	60%, 20%, 20%
NARX-c	82.2	0.99	5	2	60%, 20%, 20%
NARX-f MFCCs	169.8	0.86	5	1	20%, 60%, 20%
NARX-f all	293.9	0.94	5	2	40%, 20%, 40%

Table 4: Detailed parameters for the best ratio R/MSE result of all approaches.

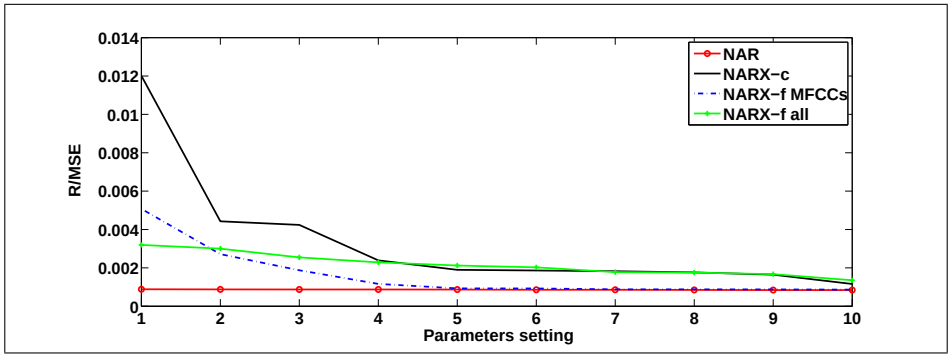

Figure 5: R/MSE ratio for all approaches.

4.5. Results' discussion

The use of SVM on rhythmic (with or without PCA) and/or MFCC features addressing the first problem formulation indicates some interesting results. The performance of this method, as aggregately measured by F1-measure, is slightly but consistently adversely affected by the use of PCA on rhythmic features while between rhythmic and spectral (MFCC) features, MFCCs present a clear amelioration in performance. The former is attributed to the already low dimensionality of the feature-sets and their diminished correlation, while the latter to the attribute-rich characteristics encompassed by the MFCC features in contrast to the solely rhythmic features.

The results obtained from the second problem formulation indicate two interesting take-aways. The first relates to the superiority of the NARX versus the NAR approaches for the current problem formulation. The existence of external / exogenous information is shown to significantly influence the predicting capability of the methods experimented with. The choice of the external information is also shown to be important and relates to the second take-away. Of the three approaches experimented herein, the NARX-c, i.e. the approach using as external information the popularity of contextually-based similar tracks, is shown to outperform the other two that use external information based on spectral and a variety of audio based features. The superiority is attributed to the contextual information deriving from human activities that, when avail-

able, are of great importance to MIR purposes, as shown in various experiments [26, 27] and eloquently put by D. Byrd: "*Music is created by humans for other humans, and humans can bring a tremendous amount of contextual knowledge to bear on anything they do; [...]. But (as of early 2008) computers can never bring much contextual knowledge to bear, often none at all, and never without being specifically programmed to do so. Therefore doing almost anything with music by computers is very difficult; many problems are essentially intractable. For the foreseeable future, the only way to make significant progress is by doing as well as possible with very little context, thereby sidestepping the intractable problems*"[13].

All in all, the effectiveness and efficiency of results is, as described in Section 2.1, based on not only the capability of each proposed method and the parameters utilised therein, but also on both the identification of quantifiable qualities of musical tracks that contribute to a track's popularity as well as the availability of transformation representations for musical tracks that closely adhere to tracks' popularity pertinent attributes. Moreover, for both problem formulations results indicate that the features selected herein did indeed provide a degree of adherence to tracks' popularity pertinent attributes. This is mostly evident for the approach used in second problem formulation presented herein, where for the NARX-c option, R values where as closely as possible to indicate the a relationship between outputs and targets while the MSE was significantly low.

Finally, the approaches utilised in both problem formulations are not without limitations. The SVM classifier approach used in the first problem formulation is heavily reliant on both the selection of kernel and respective parameters [28] while its complexity is significantly high for large-scale tasks when using a non-linear kernel [29]. For the second problem formulation, the use of dynamic Neural Networks for timeseries prediction maybe highly suitable, though is sensitive

[13] `http://www.informatics.indiana.edu/donbyrd/Teach/I545Site-Spring08/SyllabusI545.html`

to the size of past values of the series to be predicted used for the prediction. This is especially true for the best performing NARX option used herein that additionally utilises past values of the external series for the prediction.

5. Future Direction of the Dataset

The TPD is not without issues that can be ameliorated in future versions. One of these issues pertains to the automatic selection of album including each popular track: as more than one such albums may exist (hit collections, re-publication of the same artist, etc), there is no easy way to select the appropriate other than manual filtering. Moreover, the requirement of having access to the content of both popular and non-popular tracks elevated the complexity and timely conclusion of the collection process, which in order to remain within limits affected the size of the popularity records collected from the only source, Billboard, containing information not included in the TPD.

Some of the future actions that would greatly ameliorate the TPD are:

API A documented API for the purposes of accessing from a single point, aggregated, integrated and fully up-to-date popularity information.

Automated updates The design and implementation of a fully automated collection and integration web-based service that will update the dataset by harvesting the sources using event-driven or periodical triggers.

Popularity sources The addition of more popularity sources mostly oriented to social networks, such as twitter based hash-tags (e.g. *#nowplaying* with mention of track's metadata) as well as directly collecting tracks' airtime from e-radios using common protocols (e.g. *Shoutcast*, *Icecast*, etc).

6. Conclusion

This work extends the Track Popularity Dataset while also presents experimentation with the dataset. The dataset is, to the best of the authors' knowledge, the first complete attempt to create an integrated dataset for the purposes

of mining information from musical track popularity. It includes three different sources of popularity definition with records ranging from 2004 to 2014, a mapping between different track/ author/ album identification spaces in order to facilitate the use and comparison of the different sources, information pertaining to the remaining, non popular, tracks of an album with popular track(s), contextual similarity between tracks based on social networks and ready for MIR use extracted features for both popular and non-popular audio tracks. Moreover, using the proposed dataset we formulate common data mining scenarios on tracks' popularity and present respective promising results.

Despite the inherent difficulty of popularity prediction prior to or during the initial period of a track's release, such a process has long been a requirement of the musical industry, while interestingly enough, the gains of such a prediction also profit artists and listeners. Thus, the availability of datasets that will allow music information researchers to experiment and compare their methods would greatly support the advancement of the research direction.

Future directions of the dataset include its manual filtering in order to enhance its content, the creation of an API for the dissemination of the dataset's information, an automated collection of up-to-date popularity information process and the expansion of the sources by addition of social networks and e-radios.

References

[1] I. Karydis, A. Gkiokas, V. Katsouros, Musical track popularity mining dataset, in: Artificial Intelligence Applications and Innovations: 12th IFIP WG 12.5 International Conference and Workshops, AIAI 2016, Thessaloniki, Greece, September 16-18, 2016, Proceedings, 2016, pp. 562–572.

[2] D. Karydi, I. Karydis, I. Deliyannis, Legal issues in using musical content from itunes and youtube for music information retrieval, in: International Conference on Information Law, 2012.

[3] D. Makris, K. Kermanidis, I. Karydis, The greek audio dataset, in: Artificial Intelligence Applications and Innovations, Vol. 437 of IFIP Advances in

Information and Communication Technology, Springer Berlin Heidelberg, 2014, pp. 165–173.

[4] R. Dhanaraj, B. Logan, Automatic prediction of hit songs., in: International Society for Music Information Retrieval Conference, 2005, pp. 488–491.

[5] K. Bischoff, C. S. Firan, M. Georgescu, W. Nejdl, R. Paiu, Social knowledge-driven music hit prediction, in: International Conference on Advanced Data Mining and Applications, 2009, pp. 43–54.

[6] Y. Ni, R. Santos-Rodriguez, M. McVicar, T. De Bie, Hit song science once again a science?, in: International Workshop on Machine Learning and Music, ACM, 2011.

[7] F. Pachet, Hit song science, in: T. . O. Tao (Ed.), Music Data Mining, Chapman & Hall / CRC Press, 2011, Ch. 10, pp. 305–326.

[8] S. H. Chon, M. Slaney, J. Berger, Predicting success from music sales data: A statistical and adaptive approach, in: ACM Workshop on Audio and Music Computing Multimedia, 2006, pp. 83–88.

[9] D. Herremans, D. Martens, K. Srensen, Dance hit song prediction, Journal of New Music Research 43 (3) (2014) 291–302.

[10] J. C. Nunes, A. Ordanini, I like the way it sounds: The influence of instrumentation on a pop songs place in the charts, Musicae Scientiae`doi: 10.1177/1029864914548528`.

[11] J. C. Nunes, A. Ordanini, F. Valsesia, The power of repetition: repetitive lyrics in a song increase processing fluency and drive market success, Journal of Consumer Psychology 25 (2) (2015) 187 – 199. `doi:http://dx.doi.org/10.1016/j.jcps.2014.12.004`.

[12] J. Lee, J.-S. Lee, Predicting music popularity patterns based on musical complexity and early stage popularity, in: Proceedings of the Third Edition

Workshop on Speech, Language & Audio in Multimedia, SLAM '15, ACM, 2015, pp. 3–6. doi:10.1145/2802558.2814645.

[13] A. Singhi, D. Brown, Can song lyrics predict hits?, in: 11th International Symposium on Computer Music Multidisciplinary Research, 2015, pp. 457–471.

[14] F. Pachet, P. Roy, Hit song science is not yet a science, in: International Society for Music Information Retrieval Conference, 2008, pp. 355–360.

[15] N. Koenigstein, Y. Shavitt, N. Zilberman, Predicting billboard success using data-mining in p2p networks, in: International Symposium on Multimedia, 2009, pp. 466–470.

[16] Y. Kim, B. Suh, K. Lee, #nowplaying the future billboard: Mining music listening behaviors of twitter users for hit song prediction, in: International Workshop on Social Media Retrieval and Analysis, 2014, pp. 51–56.

[17] A. Singhi, D. G. Brown., Hit song detection using lyric features alone, in: International Society for Music Information Retrieval Conference, 2014.

[18] K. Frieler, K. Jakubowski, D. Mllensiefen, Yearbook of the German Society for Music Psychology, Gttingen: Hogrefe-Verlag, 2015, Ch. Is it the Song and Not the Singer? Hit Song Prediction Using Structural Features of Melodies, pp. 41–54.

[19] K. Jensen, D. Hebert, Predictability of Harmonic Complexity Across 75 Years of Popular Music Hits, The Laboratory of Mechanics and Acoustics, 2015, pp. 198–212.

[20] B. Shulman, A. Sharma, D. Cosley, Predictability of popularity: Gaps between prediction and understanding, CoRR abs/1603.09436.

[21] J. A. Burgoyne, D. Bountouridis, J. V. Balen, H. Honing, Hooked: A game for discovering what makes music catchy, in: International Society for Music Information Retrieval Conference, 2013, pp. 245–250.

[22] A. Gkiokas, V. Katsouros, G. Carayannis, Towards multi-purpose spectral rhythm features: An application to dance style, meter and tempo estimation, IEEE/ACM Transactions on Audio, Speech, and Language Processing 24 (11) (2016) 1885–1896. doi:10.1109/TASLP.2016.2554283.

[23] S. Bck, M. Schedl, Enhanced beat tracking with context-aware neural networks, in: 14th International Conference on Digital Audio Effects, 2011, pp. 135–139.

[24] T. Fujishima, Realtime chord recognition of musical sound: a system using common lisp music, in: International Computer Music Conference, 1999.

[25] C.-C. Chang, C.-J. Lin, Libsvm: A library for support vector machines, ACM Transactions on Intelligent Systems and Technology 2 (3) (2011) 27:1–27:27.

[26] P. Lamere, Social tagging and music information retrieval, Journal of new music research 37 (2) (2008) 101–114.

[27] I. Karydis, K. L. Kermanidis, S. Sioutas, L. Iliadis, Comparing content and context based similarity for musical data, Neurocomputing 107 (0) (2013) 69 – 76.

[28] G. C. Cawley, N. L. Talbot, On over-fitting in model selection and subsequent selection bias in performance evaluation, Journal of Machine Learning Research 11 (Jul) (2010) 2079–2107.

[29] M. Nandan, P. P. Khargonekar, S. S. Talathi, Fast svm training using approximate extreme points., Journal of Machine Learning Research 15 (1) (2014) 59–98.