M 2014

U.PORTO

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

TRIBO
Musical venues recommendation system based on the social web

HENRIQUE GRANJA LIMA

DISSERTAÇÃO DE MESTRADO APRESENTADA

À FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO EM 25/07/2014

ÁREA CIENTÍFICA: CIÊNCIAS DA COMUNICAÇÃO

TRIBO

Musical venues recommendation system based on the social web

Henrique Granja Lima

Mestrado em Multimédia da Universidade do Porto

Orientador: Prof. Fabien Gouyon

INESC Porto

Junho, 2014

Ao meu avô Manuel da Silva Granja,

Uma das pessoas mais interessantes que já conheci

E que ainda fazia questão de querer aprender, aos 96 anos de idade.

Acknowledgments

There are countless people that, directly or indirectly, many times without even knowing, helped me in the concretization of this great and difficult personal project. I can’t and I won’t cite all.

First of all, I would like to thank Prof. Fabien Gouyon, for all his support but especially for all the new knowledge perspectives I acquired with him. I met different teachers in this Multimedia course and all of them influenced me and helped me change my point of view about different subjects, so, I would like to express my gratitude to all.

My colleagues in the master degree were also very important in this journey, defying me, criticizing me when necessary, but most all, always giving support. Thank you!

My family and close friends also deserve a special word of gratitude, helping me even in the most complicated moments and encouraging me to go on. Thank you Mom and Dad, Rui, Daniela and Carolina.

Last, but not least, I have to talk about a person who, besides a great friend, was decisive with all his technical support, the guy behind Inov Solutions, João Pires.

Abstract

We live in the web 2.0 generation. Internet is no longer a simple tool and became essential for much we do in a daily basis; it's more and more ubiquitous on our lives, dominating our social interactions. Social networks originated a smaller world and allowed a fast and globalized mechanism of information sharing but, at the same time, limit our privacy, providing the ideal conditions for the creation of a gigantic database about users’ habits. More or less voluntarily, everyone is leaving a considerable digital footprint and this information can be extremely valuable to provide a more customized internet experience.

With the increase of available online information, recommendation systems are getting more relevance, becoming an integrant part of most social platforms. Internet radios made it possible to infer users music tastes, while the proliferation of mobile devices led to the development of location-based systems. In this research, new ways of joining these two domains will be discovered.

The final objective of this project is the development of a geolocalized application with a recommendation system of venues related to music. To achieve this, only free information found on the web will be used, using as main source public APIs made available by different social networks. Venues will be characterized by its customers, so this research will try to prove that people usually go out to places where they can listen to similar music they listen at home. After inferring the user’s musical tastes, a similarity measure will be calculated with respect to each venue, so that a recommendation can be done.

The resulting database of this project, with a musical categorization of venues, can be useful for further researches.

Resumo

Vivemos na geração da web 2.0. A internet já não é só mais uma ferramenta e tornou-se essencial em muito do que fazemos no dia-a-dia; está a tornar-se mais ubíqua nas nossas vidas, dominando as nossas interações sociais. As redes sociais deram origem a um mundo mais pequeno e permitiram a criação de um mecanismo rápido e globalizado de partilha de informação mas, ao mesmo tempo, limitam a nossa privacidade, providenciando as condições ideias para a criação de uma base de dados gigantesca acerca dos hábitos dos utilizadores. Mais ou menos voluntariamente, toda a gente está a deixar uma considerável pegada digital e esta informação pode se tornar extremamente valiosa para providenciar uma experiência na internet mais personalizada.

Com o aumento de informação disponível online, os sistemas de recomendação estão a ganhar maior relevância, tornando-se parte integrante de muitas plataformas sociais. As *internet radios* tornaram possível inferir-se os gostos musicais dos utilizadores, enquanto a proliferação dos dispositivos móveis levou ao desenvolvimento de sistemas baseados na localização. Nesta investigação vão ser descobertos novos caminhos para ligar estes dois domínios.

O objetivo final deste projeto é o desenvolvimento de uma aplicação geolocalizada com um sistema de recomendação de lugares ligados à música. Para isso, vai ser apenas usada informação encontrada livremente na web, usando como principal fonte APIs públicas disponibilizadas por diferentes redes sociais. Os locais vão ser caracterizados pelos seus utilizadores, assim esta investigação vai tentar provar que, quando saem à noite, as pessoas normalmente visitam lugares onde podem ouvir música semelhante à que gostam de ouvir em casa. Depois de se inferir o perfil musical do utilizador, vai ser calculada uma medida de similaridade em relação a cada local, para que se possa realizar uma recomendação.

A base de dados resultante deste projeto, com uma categorização musical dos lugares, pode ser bastante útil em futuras investigações.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AJAX: Asynchronous Javascript and XML

API: Application Programming Interface

CEO: Chief Executive Officer

CORBA: Common Object Request Broker Architecture

DCOM: Distributed Component Object Model

DJ: Disc Jockey

DOM: Document Object Model

GPS: Global Positioning System

HTML: Hyper Text Markup Language

HTTP: Hypertext Transfer Protocol

HTTPS: Hypertext Transfer Protocol Secure

IP: Internet Protocol

IRC: Internet Relay Chat

JSON: JavaScript Object Notation

MIDI: Musical Instrument Digital Interface

MMTD: Million Musical Tweet Dataset

MUCOSA: Music Content Semantic Annotator

ORM: Object Relational Mapping

OSC: Open Sound Control

PHP: Hypertext Preprocessor

REST: Representational State Transfer

SDK: Software Development Kit

SOAP: Simple Object Access Protocol

TCP/IP: Transmission Control Protocol/Internet Protocol

URL: Uniform Resource Locator

XML: Extensible Markup Language

1. INTRODUCTION

I probably belong to the generation mostly influenced by the World Wide Web. Back in the 1990's, the Internet was in its early stage. Cell phones were still too expensive and impractical; people still held their musical collection in a physical form; no one even considered the possibility of shopping or accessing their bank account from a computer.

The first concept of Internet was incubated in the Defense Advanced Research Projects Agency. The Intergalactic Computer Network, seen as the future "main and essential medium of informational interaction for governments, institutions, corporations, and individuals" (Licklider, 1963), was projected well before the launch of the first IBM personal computer (1981) and originated Arpanet, the first sketch of a computer network. Thus, in times of Cold War, the reasons behind the creation of the Internet were based on political and military worries. In a very tactical conflict, it was very important to take advantage of the available information about the enemy, ensuring a viable, secure and continuous process of data sharing and, at the same time, demonstrating a technological advantage over the enemies.

From these initially closed local networks, the growth was exponential, appearing more and more small networks that eventually started to be interconnected through the standard protocol TCP/IP[1], originating a global connection seen as the modern internet. From a knowledge tool, Internet started evolving to a communication tool, with the creation of protocols used in rudimentary chat systems, like IRC. In 2004, the growing complexity of networks originated the term web 2.0, proposed by Tim O'Reilly to designate a new generation of virtual communities and services, which included in its vocabulary the concept of social network.

Today, the Internet is a precious tool in our lives. We can do almost anything from the comfort of our homes, from shopping to gaming with people from all over the world. It's even acceptable in our days to work remotely from home. With increasingly faster and safer connections, and costless storage units, the concept of Cloud Computing[2]

[1] TCP/IP (Transmission Control Protocol/Internet Protocol) is the basic suite of communications protocols to connect computers on the internet

[2] General term to describe the delivery of hosted services over the internet.

came naturally and most of our assets only have a virtual existence. Music tracks, movies or books became easily available, which lead to huge collections of digital material that need to be organized and cataloged.

In our frantic modern world, people feel forced to engage an accelerated lifestyle, with less time for social activities. We live in a globalized world with a vast amount of communication channels, which open the possibility to know more people, however relationships tend to become less meaningful. In this context, people tend to feel kind of isolated and rely on the Internet as a refuge. “Social media is a parallel universe to the real world we all reside in” (Caruso, 2011), where people can share opinions and feelings they can’t express in their real lives. This information, available on social networks, can be used to personalize the user experience on the internet, making it more enjoyable.

At the same time, technology shifted the way we interact with the World through portable devices. It is now possible to produce smaller electronic components for a low price, which can be incorporated in devices like tablets or smartphones. It is also easy to incorporate in these devices geolocation technologies, giving them tracking features.

This new generation of devices was a huge success. “The transition from a PC or notebook to the ‘always on’ smart phone or tablet is not primarily about the smaller, more portable, mobile device. It is rather about the fact that computing services are now available virtually wherever and whenever the user desires them.” (Walters, 2012). It is expected that, by 2015, more than 50% of Internet interactions will come from mobile devices. These interactions can bring new types of information, essentially related to geolocation, which is being more and more used by social platforms.

All this new information users share on the web become much valuable and can be used for better or worse. We can see evidences in new aggressive advertisement style on the internet, in which users are advised about products and services relate do previous interests expressed online. At the same time, online data about interest, habits or places visited can be used to refine recommendation systems. This are the topics in which my investigation will focus.

The objective of this research is to achieve a reliable venues recommendation system, based on the user's musical tastes. The principle to follow is that people with similar musical tastes will probably go the same spots to listen to music. But first, some problems will have to be tackled. First of all, venues will have to be musically classified. For this objective, I will test the principle that the music played on a bar or a pub can be mirrored by the music its customers listen to, obtaining social profiles of customers registered on Foursquare (https://pt.foursquare.com/). Second, the users of the application will also have to be musically classified, so that their music profile can be compared to the venues musical profiles. For that purpose, musical data will be extracted from social networks like Twitter (https://twitter.com) or Facebook (https://www.facebook.com/). The final step will be to compare the profiles using a similarity algorithm. This can be accomplished by using online available services, like the one provided by Last.FM (http://www.lastfm.com/).

In this document, I will start by giving a more detailed description of the problems that are in the foundation of this project: build venues music profiles; infer users' music tastes; calculate similarity between both profiles. Next, I will review the state of the art in the area of social web and recommendation systems. First, a review of academic projects will be done, starting with the possibilities opened by social platforms to the development of recommendation systems; then, I will study the application of these systems to the music domain, with the description of different types of models; finishing with some location-based recommendation projects and how they can be combined with music-based systems. A historic review of the corporate world will be also done, starting with the rise of social networks like Facebook and Twitter, and their important contribution to internet radios and recommendation systems like Musicovery (http://musicovery.com/) or Spotify (https://www.spotify.com) and finishing with the trend of integrating the two features, in platforms like Last.FM and MySpace (https://myspace.com/). I will also explain how my project can extend these contributions. In chapter 4, I will do a short technical insight about the most important concepts for this research, writing about the different data extraction methods, like APIs or web scraping, and about the types of recommendation systems and the mathematical models behind them. In chapter 5, I will describe the adopted methodologies and solutions, from the first solution, where I took advantage of the Million Musical Tweets Dataset to cross information about music and geolocation, to the second solution, where

I used Foursquare API to map venues and used the customers registered in each one when trying to classify the music associated with a specific place. In chapter 6, I will present and evaluate the results, analyzing the regional distribution of the data obtained with the first solution and finding similarity measures between the musical profiles obtained by the application and some information collected personally in a selection of venues located in the city of Porto. Finally, I will present the conclusions of my study, while proposing possible paths for future work.

2. THE RESEARCH PROBLEM

The main objective of this research is to build a venues recommendation system, based on users' listening habits (the emphasis will be on night spots like bars or pubs). The outcome of my research will be a mobile geolocalized application, which can display on a map locations near to the user, with practical information about places he will probably like the most, based on his music tastes. The information about the venues and the users is intended to be entirely based on free content available on the web and continuously updated with the help of the users. To achieve my objective, there are three fundamental problems that need to be tackled.

2.1. Build venues music profiles

After finding places near to the user, the starting problem (and probably the most difficult to solve) will be the musical description of each venue. The objective is to build a system based on very specific user tastes, more in an artist/band level than a musical genre. For this reason, the probable approach will be to find each venue's customers and try to find their favorite artist or bands, using information available on the web, particularly in social networks.

Today, due to the growing exposition of people on the internet, most social platforms have strict privacy policies to protect its users from having their private data compromised. The best solution will always be to obtain this information on a voluntary basis, however it will only be possible to build a significant knowledge database after the application attracts a considerable number of users. Systems based on collaborative and social information, as this is intended to be, always face the so called cold-start problem. When a recommendation system is based on its users information and interactions, it's often difficult to give recommendations when the application is still not popular. To solve this problem, normally content-based techniques are used, in order to characterize items. These techniques describe each item independently, based on inherent properties or characteristics. The descriptors can be automatic (which generally extract low-level characteristics of music like tonality or rhythm) or manual, based on users annotations, which general lead to high-level features (like the music genre). In this case, I could search for keywords found on venues reviews, but this could lead to a more generic music profiling. For this reason, the solution will be to intrusively find the customer's music tastes, using public APIs or scraping methods.

Other problem can be the development of a recommendation system based on two different domains: venues and music. There have been some studies that aim at "integrating and exploiting knowledge on several domains to provide cross-domain item recommendations" (Fernández-Tobías, Cantador, Kaminskas, & Ricci, 2011), however this is a field still not much explored. With the proliferation of social platforms acting in different domains, users just get tired of repeating the registry process over and over again, leading to the so called social network fatigue. "Unless the time required to sign-in, post to, and maintain profiles across each network is reduced, it will be impossible for most users to participate in multiple sites for very long" (O'Hear, 2007). Also, "most of the available social networking sites are isolated amongst each other" (Passant & Raimond, 2008), making it difficult to interconnect data between different platforms. Linked Data is a project that aims to solve this problem, establishing formal connections between items from different sources and integrating different communities like DBpedia[3]. At the same time, major social networks like Facebook or Twitter are following a path that consists in grouping together and integrate different platforms like Foursquare and Spotify, which can make this task easier.

Finally, I raise the hypothesis that is possible to infer the kind of music played in venues like bars and pubs by the musical tastes of the customers. People go out moved by different motivations and it is known "the impact of social interaction in the choice of venues" (Wang, Terrovitis, & Mamoulis, 2013). Also, an individual can choose a particular venue for economic reasons or due to geographical constraints (people tend to visit places near their residence). However, this investigation will try to demonstrate that music is still a great motivation to visit specific venues, particularly those seen as night spots. Even assuming this, an obstacle can be found in the reluctance of users for sharing with the world check-ins in different locations. People are concerned with their privacy and they want to relax when they go out, which can lead to a reduced use of their smartphones to share information through the social networks. At the same time, people tend to register a check-in when they go to a new venue and "omit to report recurring visits to a place" (Wang, Terrovitis, & Mamoulis, 2013).

[3] Crowd-sourced community project aiming at extracting structured information from Wikipedia.

2.2. Infer users' music tastes

The second problem in this research will be to determine the music taste of the users using the application. This is important to make possible the intersection with the venues music profiles, so that the similarities between the two can be quantified. Today, music is integrated in almost every social platform, so the solution for this problem doesn't seem to be difficult. Users often share music videos on Facebook or post tracks listened on Spotify on their Twitter timeline, using connection plugins. One different approach could be to search for music files on the user's device, but this could be a much more intrusive method. In addition, listening habits are changing and people use less physical storage space with music files, normally using streaming services like Spotify or Grooveshark[4]. However, difficulties can be found due to privacy constraints. Usually people don't mind making public the music they listen to, but it is still a kind of information that can be hidden in most social platforms.

[4] Online music streaming service based in the United States and integrating a recommendation system.

2.3. Infer similarity between users and venues

The final step attemps to infer the similarity between the user's and the venue's music profiles. There are different statistical methods to calculate similarities or semantic distances[5] between items. Assuming the collaborative approach followed in this project, a possible approach could be to infer a distance measure between music profiles using algorithms like the Cosine Similarity or Pearson Correlation Coefficients. However, to solve the cold-start problem, this would imply to previously obtain a structured database of user's favorite bands or artists, in the form of a matrix or a social graph. Other approach could be to infer inherit characteristics of the items (in this case, the users and the artists or bands) through content-based methods and then infer similarities using other algorithms. However, this could also be very time consuming. In consequence, the probable solution will be to find similarity methods in open APIs. There already are social platforms that, taking advantage from their large social databases, allow users to use their proprietary algorithms for these kind of tasks, when there is not a commercial intention behind.

[5] Metric defined over a set of items, based on the likeness of their meaning or semantic content.

3. STATE OF THE ART

3.1. Academic

3.1.1. The rise of Recommendation Systems

The dramatic increase of the internet user base brought an open and valuable source of information, constituting a gigantic database about interests and habits, which ultimately led to the second generation of Internet, the so called web 2.0. Since the end of the 20th century, when internet started to be seen as a familiar tool, research has been made to study different ways of extracting and using this kind of information.

Back in 2000, the web browsers were in an early development stage and some studies were being done to infer users interests from their implicit behavior on internet, like a system that “unobtrusively measures normal actions performed by the user on a page” (Goecks & Shavlik, 2000). Goecks and Shavlik (2000) proposed a system that measured a user interest in a particular webpage by inspecting common actions like the number of hyperlinks clicked, the amount of scrolling performed or the addition of the page as a bookmark. A large number of actions on a particular page could indicate a positive level of interest by the user. To implement the system, a method based on learning algorithms was implemented, consisting of two basic steps: record the HTML contents of a visited page, along with the actions performed; build labeled training instances from the information previously recorded, associating with which webpage a set of keywords that can describe it. Once trained, the algorithm could predict measures of actions in a page, based on the semantic content of that page. This research can be viewed as a predecessor of modern tracking agents that use our browsing preferences to provide marketing and publicity strategies specific for each user. It is very common nowadays to be flooded with proposals of flights for a destination that we normally search on the web or particular models of laptops if a search was made when thinking about purchasing a new computer.

With the proliferation of social networks, especially since 2004 (when the term web 2.0 became part of our everyday lives), by 2005 more than a million personal profiles were available across different social networks. With the huge amount of data provided by explicit behavior of internet users, the need of building users’ profiles was dominated by the possibility to “build models of people outside of narrow application domains, by capturing the traces they leave on the Web, and inferring their everyday interests from this" (Liu & Maes, 2005). Starting with the principle that, adding to references about friends and acquaintances, people maintain an explicit profile of their interests and passions on social networks, this investigation aimed at harvesting information, using natural language processing to construct rich models of people. Knowing that these interests are usually categorized under clusters like music, books, movies or sports, an Interest Map was built, following four steps: mine social network profiles; extract a normalized representation of the interests by mapping the keywords into a formal ontology; extend the normalized profile with metadata; use a machine learning algorithm to infer the semantic relationship between pairs of descriptors. Common issues still faced in modern social mining techniques were encountered in this investigation, like the difficulty of mapping free input text into meaningful ontology[6] or the resistance of the users to share information that they know can become public.

The social web opened new perspectives to automatic recommendations, using a large database of items, users and their rating of those items to build collaborative systems. The need for tuning these systems, making them more reliable, led to a research suggesting that "drawing on similarity and familiarity between the user and the persons who have rated the items can aid judgment and decision making" (Bonhard & Sasse, 2006). This study states that most of the recommendation systems proposed in the past were not credible enough, because the users don't know where these choices came from and why some items have been prioritized over others. In opposition, it proposed a user collaborative filtering system that could be closer to the real-world, because it was based on the user rating rather than the item matching. With this principle, similar user groups could be formed, making it possible to recommend items that were not yet classified by the user, but by a close neighbor. There was also the care of testing the system, by deleting part of the real rating set and trying to rebuild it using the prediction algorithms. The study conducted by Bonhard and Sasse (2006) concluded that people prefer recommendations from people they know or, as alternative, wanted to know more information about the recommender's preferences.

Social platforms had a great evolution over time, starting to integrate social tagging

[6] Pillar of the semantic web, representing knowledge as a hierarchy of concepts within a domain.

systems where “users create or upload content (items), annotate it with freely chosen words (tags), and share it with other users” (Cantador, Bellogín, & Vallet, 2010). These tags help the constitution of a classification scheme known as folksonomy, which can help the development of content-based recommendation systems. It is assumed that the tags can describe the interests of the users that provided them, also that a tag can describe better an item content (being attributed as consequence a greater weight) when it is used by a larger number of users. To evaluate these tag-based recommendation models, they were tested against two different social platforms: Delicious[7] (referring a broader domain) and Last.fm (directed specifically to music).

[7] Delicious is a social web service for storing and discovering web bookmarks (URIs saved for later retrieval).

3.1.2. The application on the music field

With the evolution of recommendation systems, it soon became clear how they could be useful in the music industry. In a digital era flooded with new contents, music lost its traditional material support becoming much more available to the common user. The proliferation of music content has brought the necessity of proper classification and categorization that could help the consumers in making their choices.

Content-based tagging systems started to be applied to help in the classification of music. MUCOSA, an "environment for the annotation and generation of music metadata at different levels of abstraction" (Herrera, et al., 2005) was built under this model. Trying to answer the challenge of developing less expensive annotation systems, this system was composed by three main components. An annotation client is responsible for micro-annotations, required to compute predictive models. The process starts with music segmentation and a pre-definition of semantic terms to regulate the automatic descriptors. Automatic training systems compute low-level features like spectral centroids, rhythm or tonality descriptors. A collection tagger deals with macro-annotations across files using three basic strategies: assigning a classification to a subset of available songs; creating a predictive model for a concept, using examples provided by the user; retrieving words used to describe a song or artist and using them as descriptors. A collaborative subsystem is also used to share annotations among different users through the internet.

Collaborative recommendation systems, which analyze users past behaviors and finding relations between items and users, are being more and more used. Originated in the music recommendation system of www.ok.ru, the second most important social platform in Russia, a model was proposed where "all data mined from the history of users' activity, content metadata and social networks are combined in a taste graph" (Bugayvchenko & Dzuba, 2013). In the taste graph, the vertices represent different music entities like users, tracks and artists, while the edges represent relations between the entities (a user likes a track or an artist is similar to another artists). To perform different recommendation tasks, the user travels through the graph, preferring edges with higher weight. To generate recommendations, a random walk[8] is proposed where items with higher steady state[9] probability are most likely to be relevant for the user. Another recommendation task is to select from a set of items the most relevant for a user. This can be achieved by traversing the graph of user preferences, counting all the visits of the vertices, being the ones with higher counts the more relevant for the user. Bugayvchenko and Dzuba (2013) finally suggest some possible improvements to their system. The randomization of recommended sequences is important for avoiding boredom when a user repeatedly visits the same web radio. It is important that the generated sequences remain relevant, coherent and diverse. To introduce this random factor, items can be picked randomly from the weighted list, being subject to a probabilistic rejection algorithm. In order to provide value for the user recommendations, current user context can be used in the models. Items can be filtered by analyzing the paths starting from the context set and ending in the vertices of filtered set. When a new user joins the system, there is not enough information associated, making more difficult the recommendation process. To solve this problem, users are previously segmented by sex, age or region and advises for the new user are given, based on the best matching demographic profile.

Content-based and collaborative-based systems have always their pros and cons, so hybrid models are always more suitable. One example of the crescent need of these mixed models is related to a shift of music consumption on mobile devices, now based on streaming services (people don’t want any more to hold huge collections of music on their device). An Android application called “Beat Commander” was proposed, which is able to stream YouTube videos that are rated by the user, giving suggestions of playlists. “Given that full audio content is not available immediately in a streaming environment”, the authors suggest “a hybrid, dynamic approach to music recommendation” (Schabetsberger & Schedl, 2013). The recommendation task is based on a web service that has access to an internal database holding audio features and user-related data like favorite tracks. The content-based similarity algorithm is based on fluctuation patterns[10]. When the music is played on the smartphone, audio data is analyzed on a pulse-code

[8] Mathematical representation of path formed by a succession of random steps.
[9] A system in a steady state has properties that stay unchanged over time, having null partial derivatives.
[10] Fluctuation Patterns describe rhythm by modelling recurring beats.

modulated[11] format. Then, a Fast Fourier Transform converts the signal into a power spectrum, whose frequency values are grouped into critical bands. The sound pressure values of the power spectrum are finally measured according to the loudness, using a model of fluctuation strength. To perform this analysis, short audio snippets are extracted from music tracks, in order to make the process lighter for the smartphone. The similarity algorithm on the web service uses fluctuation patterns to return a list of similar tracks, by finding the euclidian distance between the seed track and all patterns in the database. To extend the set of tracks known to the user, a social similarity algorithm is used. First, the keyword relevance is determined, using the title, tags and other information, before measuring the popularity of a video with indicators like the click count. To calculate the similarity between tracks, YouTube's related videos function is used to build a graph which nodes are videos and all edges have uniform weight. The seed track marks the starting point to search for tracks within a particular minimum threshold. The final algorithm relies on a vector with both values for the content and social similarity, which helps in the task of sorting the tracks according to the similarity to the seed track. After an evaluation, consisting on user feedback about similarities between tracks, the content algorithm proved to be more successful than the social one, probably because of the simplistic collaborative model proposed.

[11] Pulse-code modulation is a method used to represent on a digital format sampled analog signals.

3.1.3. Location-based recommendations

With the spreading of mobile devices, information about users' locations started to become available on social networks. This originated an increasing number of applications to take advantage of users location. In the location-based social networks, visits are reported explicitly or even implicitly by allowing smartphone applications to report visited locations. It is a very interesting problem the recommendation of new locations, for example, in online publicity.

An interesting investigation was then performed, starting with the principle that "the spatial nature in the past user behavior and also the information about the user social interaction with other users, provide a richer background to build a more accurate and expressive recommendation model" (Wang, Terrovitis, & Mamoulis, 2013). It is proposed the development of a recommendation system based on four factors: visited places by a user in the past; the venues locations; the social relationship between users; the similarity between users. Before proposing algorithms to solve this problem, some studies have been performed in known platforms like Brightkite[12] and Gowalla[13] to prove some important assumptions. Some interesting conclusions about the user's behaviors were found: it was proved that most visits to a place are first-time venues; people usually go to nearby places; friendships and the social factor is very important when a venue choice is made. These served as starting points for the proposed models, particularly the first conclusion. To represent the data from which the system was going to be built, the authors constructed a complex graph, where nodes could represent users or locations and edges could represent friendship links between users or visits of users to locations. Given this model, the problem is to create a list of location recommendations that have not been visited by the user and that are considered successful if the locations are visited within a certain time frame. The investigation of Wang, Terrovitis and Mamoulis (2013) started to propose four basic strategies to solve this problem: a user-based collaborative filtering, based on the idea that "similar users have similar preferences on locations" (Wang, Terrovitis, & Mamoulis, 2013); a location-based collaborative filtering, based on the idea that "people visit similar

[12] Brightkite is a location-based social network disappeared in 2012, where people could check-in at places using text messaging.
[13] Gowalla is a location-based social network similar to Brightkite or Foursquare, that terminated its activity in 2012.

locations” (Wang, Terrovitis, & Mamoulis, 2013); a location nearest neighbor technique, based on “the spatial distance of a location to the locations that have been previously visited by a user” (Wang, Terrovitis, & Mamoulis, 2013); a friend-based collaborative filtering, considering that “people listen to their friends and follow their friends’ recommendations” (Wang, Terrovitis, & Mamoulis, 2013). They later proposed two more refined algorithms. The friendship-based bookmark-coloring algorithm took only into account the friendship edges. Starting with a user, it found the places visited by all the other users within a distance range from the visiting history of the target user. The location-friendship bookmark-coloring algorithm relied on the assumption that “some people are friends online only because they are friends offline or they have other common interests not necessarily related to visiting locations” (Wang, Terrovitis, & Mamoulis, 2013). Knowing this, the algorithm gives relevance to users that behave similarly to the target user. To test this, portions of Gowalla and Brightkite were extracted to create recommendations, which were checked in the real world. The results proved the bigger adequacy of the two last algorithms, which considered the network structure in addition to the past user behavior, and that could cover the totality of users, providing good prospects to a cold-start. It was also concluded that a smaller distance range in the analysis provide better results.

Other research introduced some innovative variables in the venues recommendation problem, projecting a system that “learns user preferences by mining a person’s social network profile. The physical constraints are delimited by a user’s location, and form of transportation, which is automatically detected through the use of a decision tree followed by a discrete Hidden Markov Model” (Savage, Baranski, & Chavez, 2011). The aim of the authors was to “design a ubiquitous location based recommendation system, which by con sidering time geography and similarity measurements, presents a more complete recommendation algorithm”. Personalization was the first problem faced by the application. Instead of obtaining user’s preferences using long questionnaires, boring to the users, the system tries to automatically infer this information, by mining the person’s social network profile on Foursquare. From the GPS coordinates obtained by the mobile device, a list of possible nearby places where the person could be in is returned. By choosing the location, the user helps to reinforce the model with this new check-in. At the same time, the Foursquare API retrieves the history of the user check-ins, adding to each venue contextual information. All this information is also added to the user model. The second challenge was the automatic integration of the user's spatiotemporal constraints. To refine the notion of distance, and taking advantage from the mobile application, the system detects the user's current form of transportation using the accelerometer. It tries to discriminate different movement states that can be associated with a transport: stationary, walking, biking or driving. Markov models[14] are used to filter the results, helping in reducing noise by knowing previous transportation modes and excluding less probable transitions. While the spatiotemporal detection system is located on the client-side, the place recommendation algorithm lies on a server which communicates with the phone. To perform this task, a content-based filtering model was chosen, starting from a set of items that describe the user's preferences to classify an unseen item. The items are the locations visited by the user and have associated information like tags and associated categories. The unseen items are the locations near the user, never visited by him. The list of nearby places retrieved by Foursquare are filtered according to the user's mood. Different moods, linked with a corresponding category, can be selected in the mobile application. After this filtering process, "a set of words containing the intersection between the tags of the user and the tags of the particular place is created" (Savage, Baranski, & Chavez, 2011). The weights for the tags associated with selected places are calculated, being chosen the ones with the higher frequencies. Being a content-based system, it sometimes carries problems when making recommendations for first-time users. To solve this, the application crawls a city's wikitravel[15] page, getting possible landmarks to be suggested to the user. Other issue faced by the researchers was the call limitation of the Foursquare API. For each venue, an API call was needed to find context information, so a set of helper accounts were created, then the API calls could be done from different accounts. A strategy of server-side caching also helped here.

[14] Markov model is a stochastic model which predicts future states not depending on the past.
[15] Project which comprises a free world travel guide.

3.1.4. Joining music and location

The combination of music recommendation systems with location related data available on the internet raised interest on models that could explore simultaneously both dimensions.

One very interesting project that explores this concept on a micro level is COM-Path, a “novel and unique interface for interactive music composition and performance” (Park, Kim, Lee, & Woon, 2010). This application consists of an online map, which can be used as a music-making tool, through the sonification[16] of georeferenced data. Users can select arbitrary routes on the map, by fixing markers, from which local information is obtained (traffic volume, temperature, wind speed and social events) using open APIs. The result is a sonification that can map in an abstract way the regional characteristics of an area. The generated music try to model theoretical concepts with the extracted information. One of this features is the temporal arrangement, which is responsible for organizing the nodes in the sequence they are marked on the map. Parameters of the generated music like pitch, loudness or timbre are controlled by the extracted information related to location, climate, demographics and social events. To produce the actual sound, collected data is transmitted to a source through music digital protocols like OSC or MIDI. Results obtained from different times and locations led to some conclusions. It is possible to distinguish richer sound maps obtained in busy city centers (like Manhattan) from simpler sound maps obtained in quitter towns like Palo Alto. Also, different results can be extracted from the same location: choosing different routes or creating them in different times of the day (the traffic constraint can vary, for example).

A more macro analysis was proposed with the “Million Musical Tweet Dataset (MMTD) – the biggest publicly available source of microblog-based music listening histories that includes geographic, temporal, and other contextual information” (Hauger, Schedl, Kosir, & Tkalcic, 2013). Taking advantage from the huge flow of posts provided by Twitter, this study presents the biggest open source of listening habits with geolocation information extracted from microblogs. The research focused on the correlation between music tastes, day of the week, time of the day and country. It is not

[16] Process which uses non-speech audio to perceptualize data.

easy to extract data from Twitter, which constitutes a relatively noisy source of information, due to its summarized and free syntax content, caused by the 140 characters limitation. This problem could be minimized, thanks to hashtags and automatic posts originated in music players plugins. In order to build the dataset, the Twitter Streaming API was used, providing posts that were filtered according to two principles: having geographic information attached; containing music related hashtags. The extracted artists or bands were then mapped using the MusicBrainz[17] API. To obtain specific locations from the geographic coordinates (latitude and longitude), MapQuest[18] was used; to normalize time information in relation to the different time zones, GeoNames[19] was chosen. Information related to musical genres were also added, considering popular tags from Last.FM. More than one million tweets were analyzed and sequentially grouped by country, day of the week and time of the day. For each group, a music genre count was performed. Some conclusions could be reached: while there were no differences between the various days of the week, big variations were found in the time of the day and country constraints. This information can be visualized in a map with graphics consisting of circles with different sizes related to the number of tweets and different colors related to music genres. These contrasts in the users' habits can be used to develop adaptive systems for contextual music recommendation.

In 2013, Oscar Celma and Alexandre Passant came up with Hellabar, a very interesting and innovative project related to music and venues recommendation. In just 24h, during Hella Hack Oakland (a one-day music and technology hackathon[20] organized by Gracenote and Pandora), the authors developed a prototype of an application that lets users discover venues, based on the artists they like. The first step of this project was to map venues for specific cities chosen for the prototype (New York, Oakland and San Francisco), with the help of the location-based platform Yelp (similar to Foursquare but with a bigger implementation in the United States). To describe the venues musically, their reviews were analyzed, being extracted key words associate with certain music genres. The final product is a JSON structure that lists the venues by genres. The

[17] Open music encyclopedia (https://musicbrainz.org/) that collects music metadata to make it available to the public.
[18] Free online web mapping service (http://www.mapquest.com/) owned by AOL.
[19] Geographical database containing over one million references.
[20] Event where software programmers collaborate intensively to create something in a very short period of time.

frontend application was developed with Angular[21], and connects to Facebook, asking the user to login to his account. Given the required permissions, Facebook API can be invoked to extract the likes with an associated "Musician/Band" label (categorizing it as musical). The extracted artist is then mapped with Gracenote[22] API, which returns the associated genres. For each genre, the previously listed venues are loaded on a map, having the user the competence to filter the locations by chosen genres (within those associated with him).

[21] Open-source web applicaton framework, based on the MVC architecture and requiring JavaScript as programming language.
[22] Gracenote is a large database of music and video metadata available online.

3.2. Corporative

3.2.1. Web 2.0 and the social platforms

The fast growing of information technologies led to a growing necessity of connecting people in networks. These online interconnected systems started gathering small niches of users with something in common, but suffered a fast viral expansion, eventually merging themselves into the biggest network of all: the internet. In 2004, the term web 2.0 appears to designate the new generation of the web as a platform. Social networks mirrored their users' interests, helping in the constitution of a huge online database of their habits.

When it was launched in 2004, Facebook was a closed Harvard campus network, only allowing the registry of users with a university e-mail. This "increased the expectations of validity of the personal information therein provided, as well as the perception of the online space as a closed, trusted, and trustworthy community" (Acquisti & Gross, 2006) and made the website an instant local success. Students just loved the possibility of knowing about their colleagues lives. In less than a year, Facebook already had around one million users, which proved the incredible effect that viral expansion and "mouth-to-mouth" suggestions can have. Initially comprising simplistic profiles, where users could share photos and basic personal information, the website opened itself to the world, becoming a huge online database of people's habits. It incorporated new features like the news feed, tagging, chat and more recently the video chat, being nowadays seen as a general purpose platform. In 2010 Facebook introduced geolocation features, allowing people to share places they visit with Facebook Places. Its universal users' base led naturally to an aggregation and interconnection of varied external services and applications in the platform, like Foursquare or Spotify. But the history of this social network isn't free of problems. With the astonishing growing of users, privacy issues became a central problem, degrading somehow the platform reputation. People started feeling afraid of the major exposition they could have in Facebook, even to companies which wanted their data for commercial purposes, reason why strict privacy settings were introduced. The huge content weight of the platform also posed some speed problems, leading the company to distribute its databases by different servers (or clouds) and to introduce a new technology which decomposed each requested page into small chunks, loaded dynamically by Javascript. The developer ecosystem somehow reflects the complex privacy controls of the platform. The Facebook Graph API is named after the concept of a social graph, an interlinked representation of the information based on nodes (entities like a User or a Page), edges (the connections between the entities, such as Page's Photos) and fields (information about the entities, such as the Birthday of a User). It involves a difficult application registry process, allowing to define several settings, as the included platforms and the server IP whitelist. At the same time, it includes a powerful SDK that allows, for example, to include chunks of HTML in the web pages, as login forms or like lists.

Having a smaller user base compared to Facebook, Twitter's 200 million active users still justify it "has already been used for various information retrieval and data mining tasks, including analyzing the spread of diseases, detecting earthquakes and hot topics, recommendation of information sources and ranking tweets according to the relevance of the user" (Hauger & Schedl, 2012). Created in 2006, Twitter has a big implementation in Indonesia and Brazil and is the most famous microblogging service, allowing users to share short text messages (called tweets) limited to 140 characters. This social network has a slightly different social structure comparing to platforms like Facebook. The connection between people is not bidirectional, comprising two different entities: the friend, which is someone a user is following and the follower, which are the accounts that are following the user. With a very simplistic mechanism, one interesting characteristic of this platform is the use of hashtags to group posts by topic or type. This standardization of format makes it very easy to analyze trending phenomena on the internet, with websites like http://www.hashtags.org presenting detailed and interesting statistics. Twitter is notorious for having famous users, which are followed by millions, making this platform useful to build a good contacts network. Like other platforms, developers can have access to the services using two different APIs: the REST API allows them to get information about users and their timelines, having some endpoints that not require authentication; the Streaming API give developers low latency access to the platform's global stream of data, making it possible to push automatically information as it is provided to the platform.

3.2.2. Internet radios and recommendations systems

The exponential growing of available online information makes it easier to categorize users' tastes and behavior. At the same time it arose the necessity of selecting the contents presented to them, overwhelmed with all the information provided in the digital era. Recommendation systems came naturally to solve this problem, with applications in the area of visual contents (MovieLens[23], Youtube or Netflix[24]) or e-commerce (Amazon).

In music, the idea of suggesting artists or bands came in the early days of the Internet. In 1995, Upendra Shardanand and Pattie Maes described a rudimentary recommendation system, which used the e-mail as the sharing platform and constituted one of the first examples of viral expansion, despite the reduced internet user base at the time. It was based on an incremental system, initially building a small profile for each user, with 10 favorite artists. Later, the user would receive by e-mail a list with supposedly similar artists, along with a list of less-known artists, which had to be rated by him. This collaborative process allowed the progressive improvement of user profiles, allowing a similarity classification between them based on mathematical algorithms like the Least Square Method[25] or the Pearson Algorithm[26].

In 2000, just in the turn to the new millennium, Pandora appeared as the first main-stream internet radio. Available only in the United States, Australia and New Zealand, it was a big local success. In the core of its website was a content-based recommendation system based on the Music Genome Project, an effort to "capture the essence of music at the most fundamental level", according to Tim Westergren, one of the founders. 50 musician-analysts came together to listen and analyze music using 400 attributes (called genes). The service asked the users to initially choose a favorite artist, giving then recommendations of similar artists, according to the genes. The user has the possibility of excluding a track that is part of the music station, or, if he really liked it, he can buy the song on one of the online retailers included in the website. Pandora's recommendation system proved to work very well, however, the cumbersome process of

[23] Movie recommendation website, which uses ratings to generate personalized advices.
[24] American provider of on-demand streaming media with an integrated recommendation system.
[25] http://mathworld.wolfram.com/LeastSquaresFitting.html
[26] http://onlinestatbook.com/2/describing_bivariate_data/pearson.html

music classification can slow down the process of inserting new artists into the database.

Another interesting content-based system is the one used in Musicovery, a customized web radio born in 2006 by the know-how combination of Vincent Castaignet, creator of the Mood Pad (a two axis music classification interface), and Frédéric Vavrille, creator of Liveplasma (a relationship map of music). The Mood Pad was a huge success and gave much visibility for the website. Songs were classified according to 40 different acoustic parameters and then placed in a two axis mood interface, with the X-axis ranging from dark to positive and the Y-axis ranging from calm to energetic. This system proved to be very user-friendly by taking off users' shoulders the burden of thinking about specific artists or genres, letting them choose music based on their current mood. In addition, it features weighted faceted browsing features, allowing to "filter and navigate information collections based on the characteristics of its items" (Voigt, Werstler, Polowinski, & Meissner, 2012). Users can limit the mood selection to a specific year or decade, as well as select specific music genres. The weighted nature comes from the possibility of measuring the mood according to the two emotions referenced before (dark/positive, calm/energetic).

An innovative idea emerged in 2007 from the minds of Todd Cronin and Phil Sergi with Bandsintown, a recommender tool that mixed music with location-based information. This live concerts discovery system gained more notoriety when it was acquired by Cellfish, a mobile apps company that released it for iPhone and Android. Bandsintown service is divided into two main areas, which are interconnected to provide a better recommendation system. In the Artist Platform, musicians and bands can introduce their tour dates manually, sync them with their websites or blogs and connect them with external ticket providers or booking agencies. This information lays the ground for the other section devoted to the fans. This section comprises a model that extract the user's musical tastes from connected Pandora, Facebook or Spotify platforms and generates a concert cloud that informs about liked artists that will perform in the user's area soon.

When Spotify was launched on October 2008 as one more music streaming service, no one imagined the changes it would bring for the digital music industry. The service presented a huge music catalogue, providing by the end of 2012 approximately 20 million songs. Despite using a system of digital rights management to prevent commercial use, the Spotify revenue model could allow people to listen full tracks for free. The platform don't charge for his basic services, getting its revenue from the users that pay for the premium service (allowing users, among other things, to download tracks), apart from the publicity income. 70% of its revenue is then distributed by artists, in the form of royalties. Thus, Spotify helped cross the last barrier for the end of music in physical format. But the service also distinguished itself for an extended social media integration, letting users connect their accounts with a variety of platforms like Facebook, Twitter or Last.fm. This helped the platform improving its hybrid recommendation system by using social networks. Initially, the system's recommendations relied on content-based techniques, such as editorial tagging, audio analysis, metadata or natural language processing. However, the company started realizing that could improve its results by combining different models, implementing an implicit collaborative algorithm to infer music similarity. Its algorithm was an extension of Probabilistic Latent Semantic Index, proposed by Hoffman in 1999, a technique based on the factorization of matrixes, composed in this case by users (as rows) and tracks or artists (as columns). The objective of the factorization is to calculate the probability of each co-occurrence, obtaining vectors which are small fingerprints of the musical style or the user's taste. After this step, the artists' similarity could be quantified, using a distance matrix. Apart from the company's recommendation method, users can apply a bunch of other methods to their music library, thanks to the integration of different apps on the Spotify's platform.

3.2.3. Integration of recommendation systems in social platforms

With the increasing amount of data available, social platforms felt the need to filter their content. In this context, we can see recommendation systems in almost all current networks.

Last.FM was the first successful social network entirely dedicated to music, appearing in 2002 as a platform for music lovers. While Pandora was very big on the United States, being created in the United Kingdom, Last.FM started to expand its user base essentially in Europe. It was very important to the music dematerialization movement, however the radio only allowed non-paying users to listen to 30-seconds previews of the tracks. Last.FM distinguishes by its social features, with an automatically generated profile for every user, along with its main functionality, the internet streaming radio. Its huge success helped building a big user base, essential for the efficiency of the collaborative based recommendation system, proving that “better algorithms are nice but better data is nicer” (Krause, 2006). Behind the model used to build the similarity model is a method called audio-scrobbling, responsible for automatically logging all the songs played by the user. Last.FM also provides an optional plugin that monitors the user’s native media-player. This music library creates the conditions for the efficiency of the recommendation system, even without knowing about the songs’ inherent qualities. Although it uses tags associated to the tracks, it is based on the principle that if a user shares some favorite artists with a group of people, he will probably enjoy other popular artists within the group. Having essentially collaborative characteristics, it needs a certain amount of data to work with, taking some time to give recommendations feedback for new users. The Last.FM API allows developers to make use of the platform’s complex algorithms, getting the most popular albums of a certain artist or a list of similar artists to a given band or musician.

When Myspace was launched, in 2003, it was intended to be a generalist social networking service. Created by former Friendster employees, the website took advantage of a complete infrastructure of finance, human resources and technical expertise, which made it possible to be online just 10 days after the company establishment. Myspace rapidly grew to become the world’s most visited social platform, incorporating features like the groups, which allowed users to share a common page and message board, making it easy to disseminate messages through bulletin boards; MySpaceTV, a video sharing platform; moods, small emoticons used to express the mood of the user within a status update. However, in 2008 it was eventually overtaken by Facebook, becoming a network essentially directed to the music niche, used by artists to create their music profiles, taking advantage from the profile HTML customization feature, along with the possibility of uploading tracks and videos. The nature of the website was slowly changing, being now more used by music fans to follow their favorite artists, which eventually led to the creation of MySpace Records, a company's smart move given the quantity of new talents that could be discovered in the platform. In 2009, Myspace entered the recommendation world, presenting Qizmt, the company's MapReduce[27] framework, which was described as a response to the platform millions of users that "consume and produce video, music and other content every minute, which constantly results in very large sets of new data"[28]. Qizmt was conceived to process data generated by the users and the analytics system, transforming it into meaningful real-time recommendations.

Way before the smartphones revolution, in 2004 Yelp appeared as an online urban guide and business review site, laying grounds to the recent location-based recommendation systems. With a big implementation in the United States, it started as a rudimentary email service for exchanging business reviews but started introducing social network features, incorporating a discussion forum and ways of connection users by allowing friends invitations, followers and messages exchange. The company eventually launched a mobile application, which helped in the localization of users, making it easier to recommend nearby venues. The core of the platform is a search tool which can be used by users to access business reviews for companies providing specific types of products and services in their area. The company comprises a business model essentially based on revenues provided by business advertisers which could get a better placement in search results. Agreements between Yelp and the advertisers also introduced features like discounts in products for the platform users. Advertisements proved to be very effective, when a study carried out by Berkeley University correlated restaurants reservations with their rating in the company's website, concluding that an

[27] Programming model used to generate large data sets with a parallel, distributed algorithm.
[28] Statement from an unknown source in MySpace, cited in the article "Facebook Envy? MySpace Open-Sources Recommendation Framework", written by Ben Parr in September 2009 (http://mashable.com/2009/09/15/myspace-qizmt/).

upgrade from 3.5 to 4 stars increased in 20% the bookings. To improve its credibility, Yelp establishes a collaborative recommendation system where each review (or rating) is evaluated by quality, reliability and user activity, disregarding the ones that don't meet these requirements.

Founded in 2009 by Dennis Crowley, Foursquare is a platform similar to Yelp, but its system of rewards make it more appealing for the users. The company implementation substantially grew in the next years, reaching 45 million users, more than a half of them located outside the United States. The platform is particularly directed to mobile devices, being based on check-ins, a mechanism whereby users can post where they are, by selecting from a list of nearby venues. To encourage check-ins, in order to maintain the database always updated, Foursquare comprises a rewards system. In a mechanism similar to a game, users can earn badges in different situations. For example, they can be crowned a superstar if they check-in in more than 50 venues or a player when they check-in with 3 members of the opposite sex. Some prizes are also specific of cities or venues and the users have the possibility of earning discounts (specials). Other Foursquare figure is the mayor, the user with most check-ins in a venue for the past two months. At the same time, the platform is a great mean for discovering new places or meeting friends. Users can leave comments or tips in a venue's page and they can know which of their friends have done check-ins in nearby places, giving the platform social network characteristics. Foursquare also uses a recommendation system, which considers the user's location, check-in history and popular sites near the user, inferring a place popularity by the number of users that have visited it, compared to the total number of visits that all users have made. By following a collaborative method, disregarding personal information provided by the users, this system has the disadvantage of giving recommendations that are "less user tailored and more generic", apart that it "disregards the place's contextual information" (Savage, Baranski, & Chavez, 2011). The platform provides a Core API that let users do basically anything they can do on the application, posting check-ins, and tips or getting information about venues and the friends' location. The Real-Time API can get this data in real time, notifying venue managers when users check-in to their venues or developers when their users check-in anywhere.

Founded in 2012 in California, in the heart of the technological hub of Silicon Valley, Moosify (http://www.moosify.com/) is an interesting approach for joining music and places under the same platform. It is already integrated in different platforms like iOS, Android and Spotify. According to Hans Posch, CEO of Moosify, the application is a response “to the demands and behavior of the youth generation, where music, entertainment and places play an important role when it comes to meeting new people”. After signing up and providing some basic information about his music tastes, along with setting up a basic profile, the user is connected with like-minded people who are living nearby, according to their favorite’s places and most-played music. Seen as a dating platform, Moosify is also focused on the discovery side, by allowing people to share recommendations about new music, new places or a concert that is happening soon.

3.3. Conclusions and proposed novelties

The project proposed in this document is intended to take advantage of two fundamental aspects to develop a venues recommendation application based on users' tastes: make use of the vast amount of information about internet users currently available on the web; use the geolocation features available in modern portable devices.

There is currently a profusion of recommendation systems, many of them addressing music or geolocation, but few trying to join the two worlds. Some researchers have already conducted a "statistical analysis of the correlation between music taste (measured via genre distribution) and temporal as well as geographical properties" (Hauger, Schedl, Kosir, & Tkalcic, 2013), however, their work focus on a macro analysis, trying to find patterns of music tastes by country, while this work uses a micro approach, trying to characterize specific location according to music. Other related approach is the system comprised by Bandsintown, which uses users' musical tastes to recommend concerts; my system will try to extend this approach by suggesting places that people can attend regularly to listen to music, instead of episodic events like concerts. Moosify platform performs a great work by joining people through music, however my application is intended to be more directed to venues and new discoveries. The most similar approach presented in this section is Oscar Celma's and Alexandre Passant's Hellabar application. My system differentiates by tracing bars and pub profiles based on their customer's music tastes, instead of relying in tag words found on reviews.

My work is also innovative on what concerns the recommendations scope. Unlike Foursquare, with a system "less user tailored and more generic" that "disregards the place's contextual information" (Savage, Baranski, & Chavez, 2011), my model will suggest places according to a very specific factor: music. At the same time, I will try to prove that not only "the impact of social interaction in the choice of venues" (Wang, Terrovitis, & Mamoulis, 2013) is relevant, but that music is also an important factor when people choose a place to go out at night, adding to social or location constraints, for example.

Regarding the application's architecture, I will rely in a system with some contact points with I’m feeling LoCo (proposed by Savage, Baranski, Chavez and Hollerer), developing a mobile application that presents a map and detects user’s location, sending this information to a separate server that retrieves nearby locations, music profiles and similarity calculations. Also, this information can be stored in a database and a server-side caching strategy will be used to avoid repetitive API calls when a place had already been analyzed.

Before describing the methodologies of my work, in the next chapter some technical background will be detailed.

4. TECHNICAL FUNDAMENTS

4.1. Data extraction methods

4.1.1. APIs

An application program interface (API) is a set of standards that specifies how certain software components can communicate. It is a software-to-software interface, through which applications can "talk" to each other, without any user knowledge and without knowing details about software implementation. Users can see an API as a black box, in the sense that they know the output to expect given certain input parameters, but they don't know what is happening behind the scenes. For example, when someone buys something online using a credit card, the e-commerce platform probably uses an API to send the credit card information to a remote application that checks if the information is correct and confirms the payment in that case. The user only sees that his purchase was confirmed without knowing nothing processes behind.

An API allows a software application to communicate with a remote application through a series of calls, managed by web services. These are methods by which data is exchanged over the internet and can be accessed by two basic protocols. SOAP is an access protocol created by Microsoft, which relies exclusively on XML and came to solve some problems with technologies that used binary coding to exchange messages (like DCOM or CORBA). The XML structures used in SOAP can become extremely complex, but this problem can get simplified by the use of Web Services Description Language (WSDL). This is a file associated with SOAP, containing a description about the web service functioning and the XML structure, and it can completely automate the process of XML building in some programming languages. REST protocol (Representational State Transfer) came to simplify SOAP protocol, not using XML to make a request. It relies only on HTTP verbs (principally GET) to specify the web services parameters and it normally presents the response in JSON, a format easy to manipulate.

Most social platforms use the REST architecture to make their services available to software developers because of its lower complexity level, not requiring expensive tools for interaction and behaving faster than SOAP. Social networks usually require developers to register an application before they begin using their APIs. By doing so, they obtain the credentials they need to start making the requests. The authentication process is usually based on the OAuth (open authorization) method, which in the last years is replacing the basic HTTP authentication most platforms were using. This new method provides a safer way, so users can authorize an application without sharing the username and the password, sensitive information that in the past was sent over the internet each time an API request was being done. This process by which applications ask user's authorization to access data is based on three simple steps: the client redirects the end user to the resource owner (the social network, in this case) and asks authorization to access specific data (for example, a friend's list); if the user authorizes the application, the client is notified and receives an authorization code; the client provides the resource owner the code, along with the client identifier and gets back an access token. With this token, the client can now make requests on behalf of the user. In spite of being safer, at the same time, this process can be cumbersome to a developer. Without being able to authorize and make requests using only an URL, it becomes difficult to test the APIs outside of the application scope. For this purpose, some platforms have been created to allow developers to discover, test and debug APIs. Apigee, for example, gathers under its console a great collection of APIs, handling for the users time-consuming processes, like the OAuth authorization. The client application also faces some restrictions related to request limits generally imposed by resource owners. There is a certain number of API requests that can be done by a fixed unit of time before these requests are blocked by the provider.

4.1.2. Web scraping

Scraping is a computer software technique for extracting information from websites, simulating a human interaction with the World Wide Web, normally by implementing low-level HTTP or embedding a web browser. It can be a good alternative for APIs in data mining, skirting some limitations imposed by security measures. This technique is widely used by search engines in the task of indexing information on the web, using bots or crawlers (software applications that run automated indexing tasks over the internet). Web scraping focus on transforming unstructured data on the web into structured data that can be stored and analyzed locally.

There are plenty of tools that can be used to help scraping the web. One of them is cURL, a software for data transfer using different protocols. For example, it can be used to simulate a HTTP request to a web page. There are cURL libraries available for almost every web programming language, being PHP's libcurl one of the most famous. With this library, it is easy to set different options for a cURL transfer. There is the possibility of setting the user agent, metadata information that informs the web server about the platform making the request, so that a different version of a web page can be obtained (nowadays, most websites have mobile versions that can be obtained if a mobile user agent is simulated). Cookies, which are small pieces of data stored in a web browser, can also be set when logging in to a website, for example. Also different security protocols can be set, like SSL, when trying to access a protected HTTPS web page.

The output of the scraping process can be queried and analyzed by a different set of tools. After the HTML or XML content retrieved by a HTTP request is converted to a DOM tree representing the structure of the page, it can be queried to find and select different nodes. XPath is a query language based on simple expressions to iterate through the DOM of a web page, trying to find elements using different criteria (*Figure 1*). It can find a node based on its tag value, content or attribute.

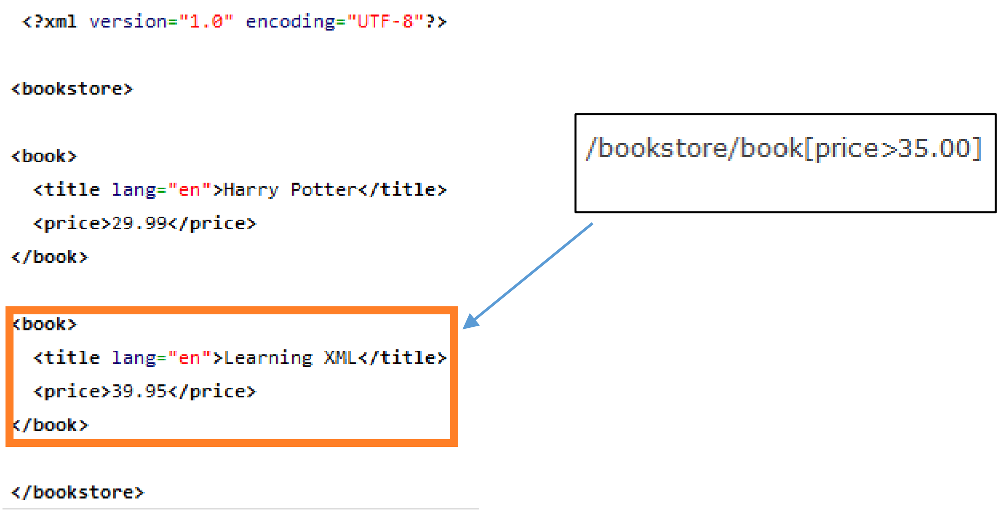


Figure 1 - Result of a XPath expression on a XML document

When the output returns a non-XML compliant document, XPath is of no use. In this case, other approaches can be used. Regular expressions work as sequence of characters forming a search pattern to be matched in a text document (*Figure 2*). Each character in a regular expression can be understood as a metacharacter with its special meaning, or a regular character with its literal meaning (see more in http://www.regexr.com/).

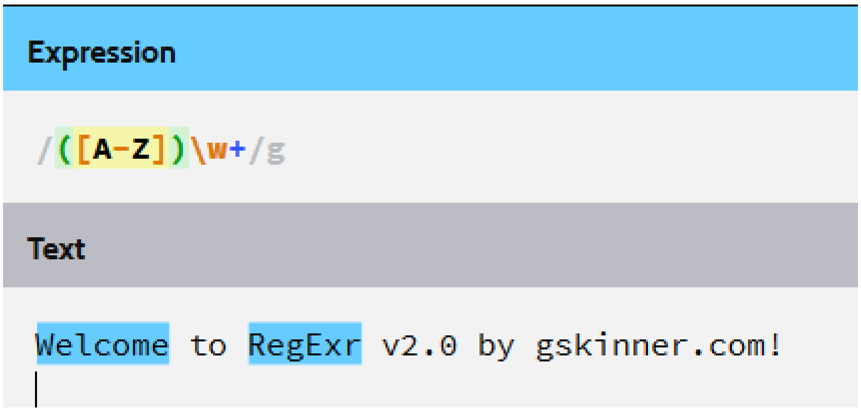


Figure 2 - A regular expression to filter words started by a capital letter

Browser automation tools can also be used with the purpose of web scraping. Tools like Selenium (http://docs.seleniumhq.org/) or CasperJS (http://casperjs.org/) are normally used for functional tests, but they can be used to simulate a browser when scraping items in a DOM environment.

4.2. Recommendation systems

With the evolution of information technologies and the internet in particular, the amount of online digital content has been increasing dramatically, especially after the web 2.0 phenomenon and the appearance of social networks. On the other hand, “the Internet has not only brought us more information and choice, but has also increased the burden of making a choice” (Bonhard & Sasse, 2006). Thus, it was clear the necessity of developing systems to filter the information presented to the user. Additionally, the increasingly competitive online e-commerce market forced the companies to intelligently advise customers about products they could like, based on ratings and previously shown behaviors (starting here the principles of collaborative filtering).

In the music industry, the growth of digital tracks libraries have reinforced the necessity of filtering information. Users have access to many music material that needs to be categorized. In addition, users like to discover new artists and bands, going from the head to the tail of the curve[29] of the listened artists’ distribution (*Figure 3*). Genre clustering and serendipity[30] are then key concepts in recommendation system applied to music.

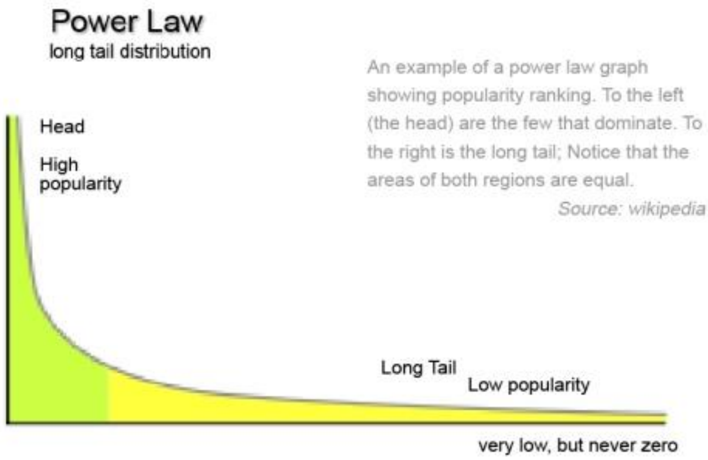


Figure 3 - An explanation of a power law graph

In this chapter, different type of recommendation systems will be detailed, namely collaborative filtering, content-based filtering and hybrid methods.

[29] http://www.longtail.com/about.html
[30] It measures how surprising a recommendation can be.

4.2.1. Collaborative filtering

Collaborative filtering is based on the principle that people who had similar opinions about an item in the past, will probably agree about other items in the future. In the music domain, this recommendation method can be based on explicit evaluation measures, like the ratings given to specific songs and artists, or on implicit feedback, tracking user listening habits, like a track total play count. Given that users are getting lazier and don't want to spend time explicitly evaluating an item, the second method is currently more used, however it only considers positive feedback (i.e. tracks listened by a user). In the base of every collaborative filtering system is a matrix of the user preferences for the item, the rows representing a user profile and the columns representing the items. The objective is to calculate similarities between the users and the items and predict how much a user will enjoy an item.

In the user-based neighborhood technique, as proposed by Celma (2008), the predicted rating given from a user to a particular item is based on the mean of ratings given by the similar users. Knowing that u and v represent two different users, i represents the target item, $R_{u,i}$ represents the known rating given by the user u to the item i and $\overline{R}_u$ represents the average rating of user u, the prediction can be formalized as follow:

$$P_{u,i} = \bar{R}_u + \frac{\sum_{v \in Neighbours(u)}^{k} sim(u,v)(R_{v,i} - \bar{R}_v)}{\sum_{v \in Neighbours(u)}^{k} sim(u,v)}$$

Figure 4 - User-based neighborhood prediction formula

Item-based neighborhood technique proposes a prediction based on the similarity between items. This technique, proposed by Celma (2008), is based on the set of items rated by the user to decide if a target item should be recommended to him. So, to predict the rating of a user on a particular item, it is important to know how the user has rated similar items. To formalize this principle, let $S^k(i;u)$ represent the set of k neighbors of item i rated by the user u, with $j \in S^k(i;u)$:

$$P_{u,i} = \frac{\sum_{j \in S^k(i;u)} sim(i,j) R_{u,j}}{\sum_{j \in S^k(i;u)} sim(i,j)}$$

Figure 5 - Item-based neighborhood prediction formula

In both techniques, similarity measures have to be calculated before the actual predictions can be obtained. In the user-based neighborhood technique, it is necessary to quantify the similarity between users; in the item-based neighborhood technique, the similarity between items needs to be computed. To retrieve these values, different formulas can be used. The Pearson correlation method reflects the degree of relation between two variables:

$$sim(i,j) = \frac{Cov(i,j)}{\sigma_i \sigma_j} = \frac{\sum_{u \in U}(R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u \in U}(R_{u,i} - \bar{R}_i)^2}\sqrt{\sum_{u \in U}(R_{u,j} - \bar{R}_j)^2}}$$

Figure 6 - Pearson correlation formula to calculate similarity

Other method that can be used is the one based on cosine similarity:

$$sim(i,j) = cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{\|i\| * \|j\|} = \frac{\sum_{u \in U} R_{u,i} R_{u,j}}{\sqrt{\sum_{u \in U} R_{u,i}^2}\sqrt{\sum_{u \in U} R_{u,j}^2}}$$

Figure 7 - Cosine similarity formula

Collaborative filtering methods are a very effective way of recommendation in the current social networks, however they also present some issues. Although they require a large dataset to work, the larger the dataset is, the more probable is to find items not rated by users, originating a data sparsity problem. Other issue is the so called cold-start problem. Because collaborative filtering is based on user ratings of items, new users and new items are more difficult to classify because they don’t have associated information yet. The “rich gets richer” is a common paradigm that can be applied to this method, as the most popular items are always related to a large number of items.

4.2.2. Content-based filtering

Content-based filtering relies on the content of the items represented with concrete descriptors and is "based on item-to-item similarity" (Celma, 2008). Before this technique can be applied, items must be described by automatic or manual methods. In the music domain, automatic methods are performed by algorithms that extract low-level (sampling rate of the audio file, spectral centroid of the audio frame), mid-level (tonality, rhythm, harmony, intensity, structure) and high-level (analysis of similar guitar solos) descriptors. Manual methods are provided by users (tagging music tracks) or experts (providing precise information about the production process, artist biography or genre information).

After this information is collected, content-based filtering predicts which items the user will like, based on his preferences, not relying on the other users ratings. To calculate the similarity between two items, an objective distance among items is considered. When the items are described by numeric attributes, simple metrics like the euclidian distance are used (*Figure 8*).

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

Figure 8 - The eucledian distance formula

If the attributes are nominal, a function can be defined to quantify the relations, assuming the value 0 when the two items are equal or 1 otherwise (*Figure 9*).

$$d(x, y) = \omega \sum_{i=1}^{n} \delta(x_i, y_i)$$

Figure 9 - Distance calculation between nominal attributes

Content-based techniques are very effective when there isn't a large dataset of users rating available, however they also pose some problems. The most relevant is the difficulty to recommend new items, assuming that the user will always be advised items too similar to the others in his profile. In addition, these systems still find many problems with extracting high-level descriptors that can be meaningful to the user.

4.2.3. Hybrid methods

It is possible to combine the two previous approaches to “minimize the issues that a solely method can have” (Celma, 2008). One common approach is to apply collaborative filtering first and then refine the results using the semantic distance of social tagging. Burke (2002) propose other methods to integrate different approaches in hybrid methods: a weighted method that uses a linear combination of the results retrieved by different recommendation techniques; a switching system, which uses a unique technique, switching to another one if the results are not good enough.

5. METHODOLOGIES

In order to develop a mobile application which could advise the users about venues, according to their music tastes, it was fundamental to build a database that could store information about users, their music tastes and venues. It was also important to find a way of characterizing the venues musically, building a profile with bands or artists that potentially could be listened in the venue. The chosen path is based on an extrapolation: this research will try to prove that a bar or a pub can be musically described from the kind of music that the customers listen to in their daily lives. Starting with this principle, it was also necessary to find a way of connecting users and venues, trying to discover which places the users often attend.

Considering the difficulty of the problem, two different approaches were tested, one based on the exploration of the Million Musical Twitter Dataset and the other relying entirely on information obtained from APIs, specifically the Foursquare API, the Twitter API and the Facebook Graph API.

One important choice was about the technologies that could help achieving this objective. It was evident from the beginning that the model would have to store a large amount of information, so MySQL was the natural choice for the database. The MySQL database server is highly scalable, successfully storing massive data warehouses without major problems. It is also a multi-platform database engine, with support to Linux, Unix and Windows based operating systems and it is easily embedded into any kind of application, making available plug-in libraries, connectors and drivers, that can integrate in PHP, Perl, Java or .NET. Other advantage is the powerful transactional engine, with high-performance queries mechanism and fast data insert capability. When choosing the programming language, PHP was the option. It seemed an advantage choosing a multi-purpose server-side programming language, which could be used for scripting tasks (such as mining web APIs) or web development. At the same time, PHP has been improving in the last years, with PHP5 transformed this into an object-oriented language. With the growing use of this language in web applications, also many frameworks appeared lately to help developers, along with Object Relation Mapping[31]

[31] Programming technique to convert data between incompatible type systems in object-oriented paradigms, acting as a “virtual object database”.

systems, like Eloquent[32].

[32] ORM included with the Laravel framework that implements an ActiveRecord model for working with the database.

5.1. Tribo 1.0

It was known from the beginning that it would be necessary to obtain a large dataset containing both music and geographical references. The Million Musical Tweet Dataset offers a huge collection of more than one million tweets (dated from September 2011 and April 2013) with music posts and additional geographical (in the form of latitude and longitude coordinates) and temporal references. This dataset was obtained from the continuous tracking of the Twitter Streaming API and comprises posts from more than 200.000 users, from 202 different countries, containing references to more than 100.000 tracks from 25.000 different artists. All the posts containing music related hashtags and geographical information were filtered.

This approach implied the population of the database (*Figure 10*) prior to the first use of the application, so the cold-start problem would not be an issue. However, there were some worries about the time and computation cost necessary to iterate all the dataset records.

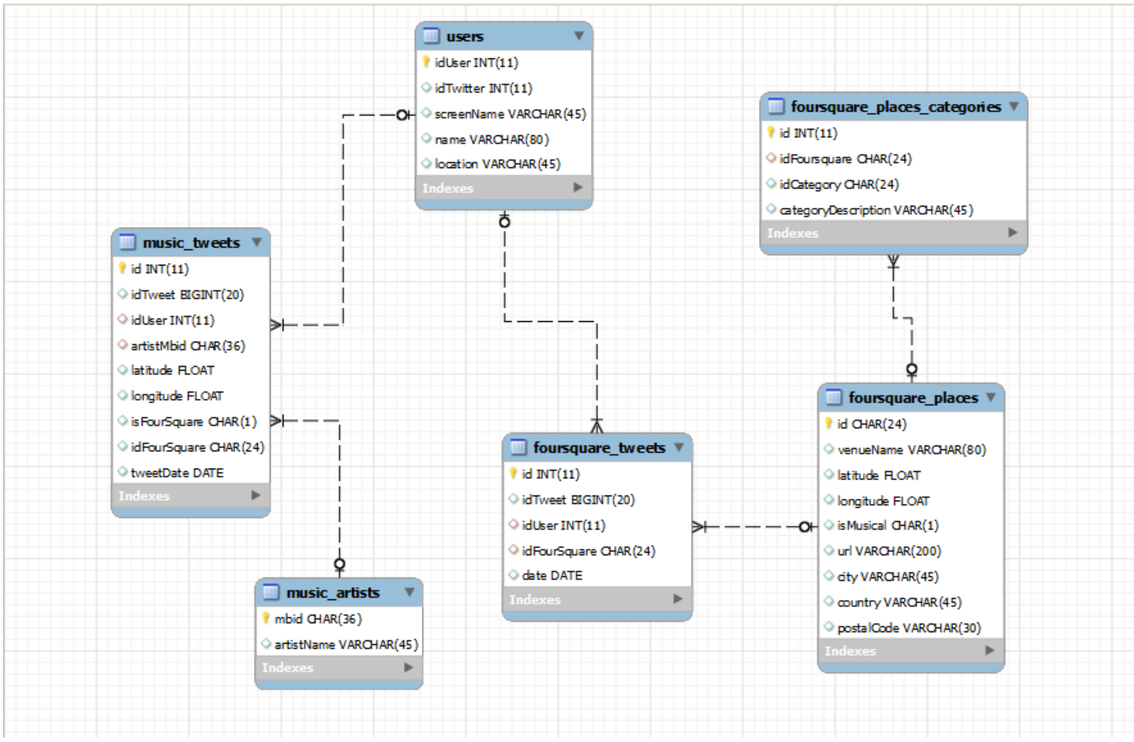


Figure 10 - Tribo 1.0 database diagram

The base mechanism of data acquisition consisted in iterating through all the dataset records (which one corresponding to a different tweet), getting for each new user found his timeline, in order to check if there is a post related to Foursquare. This was the solution found for relating venues with the users and their music tastes. In order to use

APIs to get user timelines (in Twitter) and details about venues (in Foursquare) it was necessary to create developer applications in these platforms. Most of API methods (including these) require an OAuth authentication, so it was necessary to obtain an application ID and secret to authenticate without providing the username and the password. In addition, most providers establish request limits to avoid the commercial use of their APIs. For this reason, and predicting an intensive requests load, a number of helper accounts were created with the objective of making additional API calls. For an easier use of the APIs, PHP open-source wrappers were used: to access Twitter API, it was chosen the one developed by James Mallison (http://github.com/j7mbo/twitter-api-php); to access Foursquare API, it was chosen a client library developed by Stephen Young (https://github.com/hownowstephen/php-foursquare).

To process the tweets of the MMTD, a PHP script was created, having as input parameters the path of the dataset file and the path for the log file (where error information could be stored for debugging). First, a connection to the database have to be set, using the MySQLi[33] extension. For each read line of the dataset, a count of the calls done for each API is maintained, so the script can switch to a different helper account when a certain limit is reached. The record has to be processed, so the important fields can be extracted, namely the tweet ID, the user ID, the geographical coordinates (latitude and longitude), the artist MBID (unique identification established by the MusicBrainz project) and the artist name. If the user is found for the first time (to verify that, a variable is maintained, working as a user's cache), his details are obtained using the user endpoint of the Twitter API and stored in the users database table. Using the user timeline endpoint of the same API, a list of his most recent tweets is obtained. Each tweet has a text part (the main body) and a source, in case it was originated in an external platform. If the source corresponds to Foursquare, the tweet is processed, so the venue where the check-in was registered can be obtained. To do that, venues/search Foursquare API endpoint is used. Given the ambiguity of tweets format, it is difficult to find a pattern to extract the venue's name, so the geographical coordinates are provided as input for the method. The method search for the nearest venue, which works well, since every record in the dataset is geolocalized. The details of the venue are obtained, including the associated categories, which are checked against a list of all the categories

[33] PHP driver used to provide an interface with MySQL databases.

considered as musical, to verify if the place is of interest to the application. The venue is inserted in the **foursquare_places** database table, as well as the associated categories in the **foursquare_places_categories** table. Finally, the actual tweet is stored in the **foursquare_tweets** table.

After this, the musical information of the record is processed. First, the artist is mapped in the MusicBrainz database. This time, there is no need for authentication as this is a completely open API. If it is the first appearance of the artist (a cache variable is also used to keep track of the stored artists), it is saved on the database, in the **music_artists** table. In the end of each iteration, the actual MMTD tweet is stored in the **music_tweets** table. A schema of the whole process can be visualized in *Figure 11*.

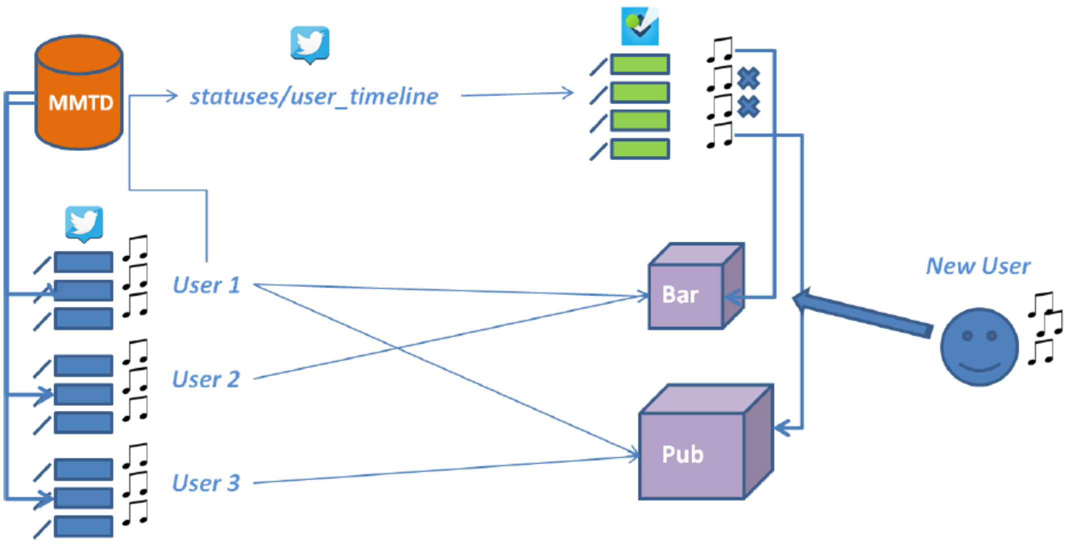


Figure 11 - Tribo 1.0 data extraction method

Hopefully, in the end of this process, an extensive collection of users with related music and foursquare tweets can be obtained, so that tweets can be grouped by venue, which can be musically categorized by music posts found for each user that checked-in in that location.

In chapter 6, the data sample will be analyzed using different distribution parameters, with an evaluation of the number of cities, venues, users and artists found by country. The objective is to obtain a sample with venues with an evenly geographic distribution (it is important here to analyze the number of venues per city rate), as well as a reasonable number of artists associated with each venue (with the analysis of the number of artists per venue).

5.2. Tribo 2.0

Given the problems found in the first solution and the difficulties in intersecting music taste information with visited venues, a different approach was considered. Instead of getting music posts and geolocalized posts separately, crossing this information through the users, the venues are mapped upfront and the customers are obtained for each one, using Foursquare API. Then, their musical tastes are inferred through their Facebook or Twitter profiles.

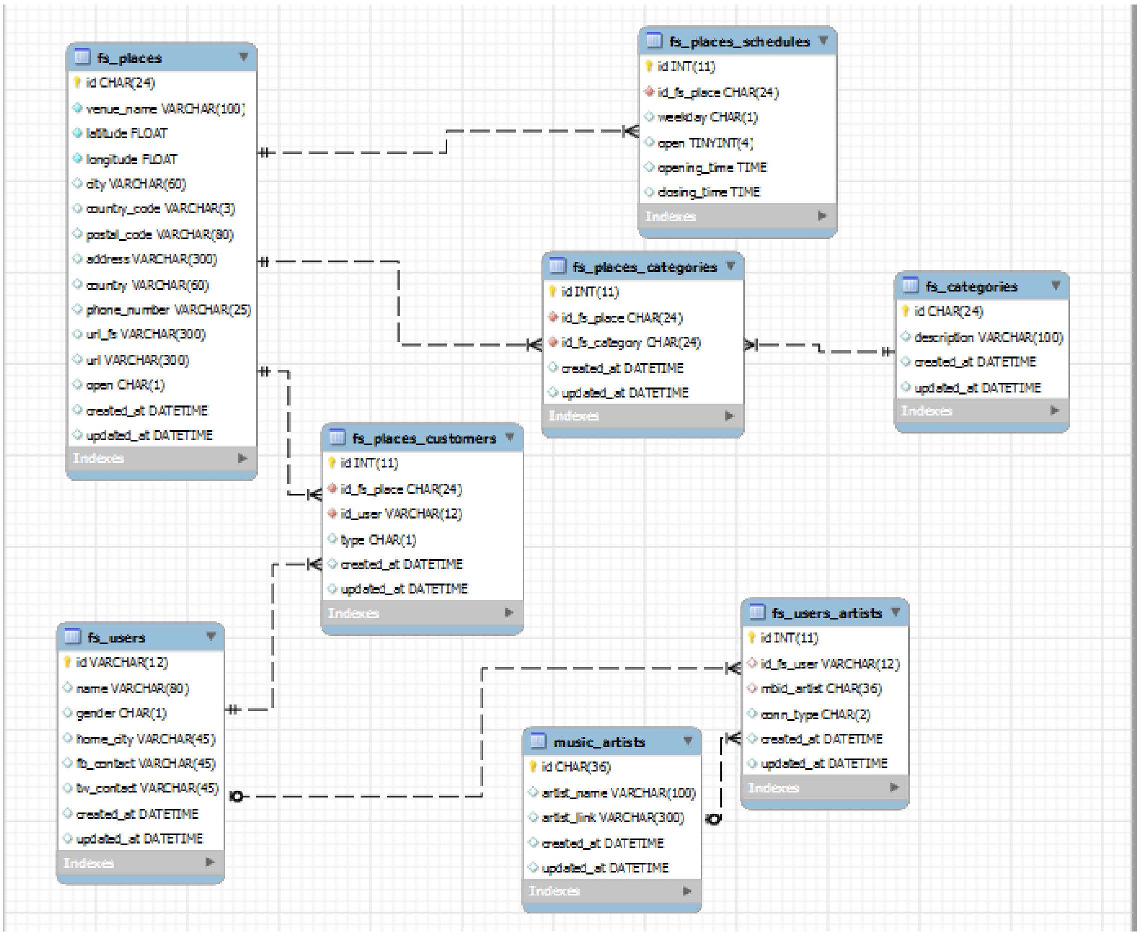


Figure 12 - Tribo 2.0 database diagram

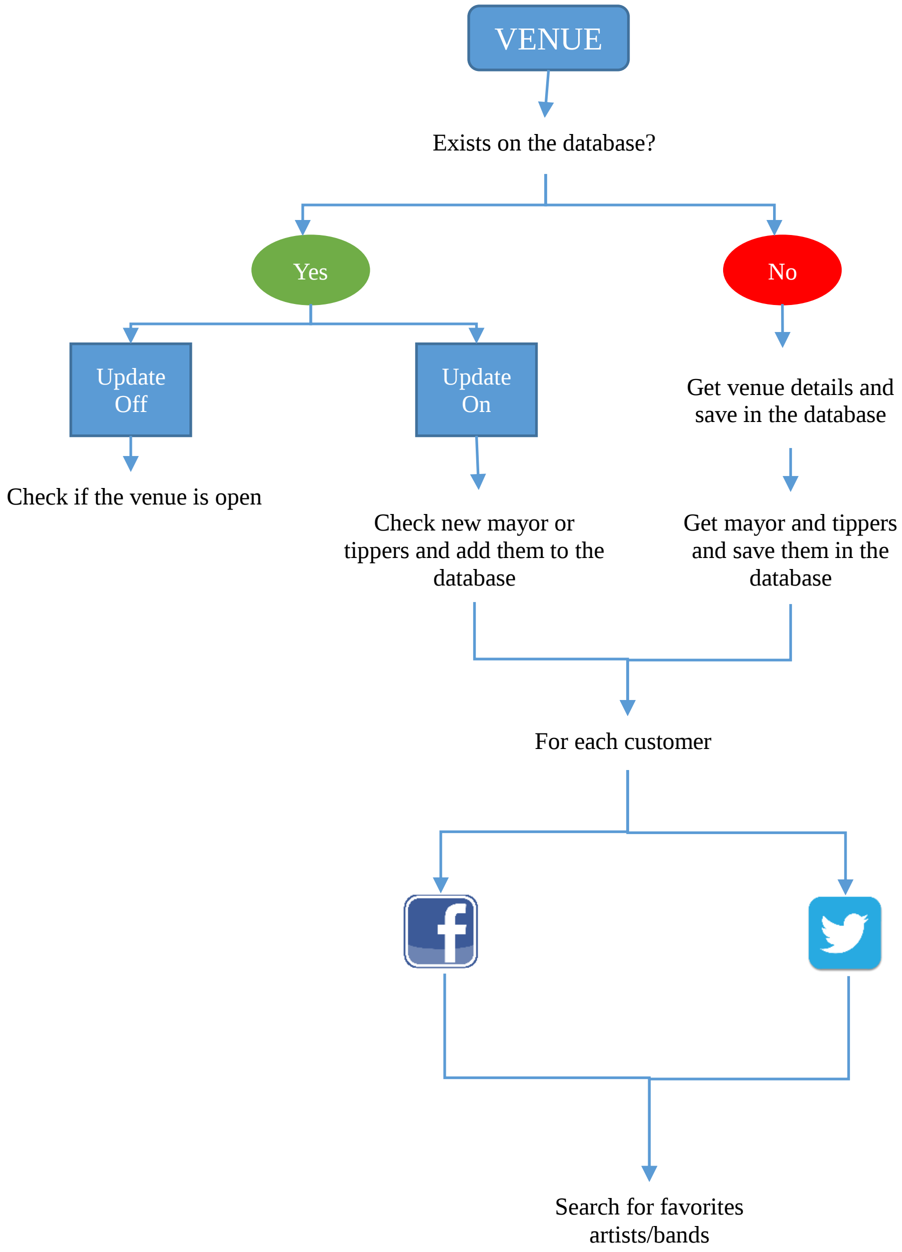
With this approach, the database (*Figure 12*) was expected to be populated after the application started to be used, retrieving the information about the venues and the customers only when it is needed. For this reason, the logic was divided by three different web services, located on a remote server and developed according to a RESTful architecture. To improve the database operations and to make it easier the mapping between entities and tables, this time the Eloquent ORM was used, which could provide a simple ActiveRecord[34] implementation for interacting with MySQL.

The first web service, **GetVenues**, relies fully on the Foursquare API to retrieve venues near given coordinates (latitude and longitude), specified as **get** parameters. To obtain this information, **venues/search** endpoint was used. Apart from the coordinates, other important parameters have to be considered in this method. There is a radius parameter that specifies the range of the venues search from the specified point. The default corresponds to a city-wide area, but in this case a smaller value was set (500 meters), because the intention is to advise venues in a particular neighborhood. Other possibility is to filter the results from category, providing an array of foursquare category ids. In Foursquare, categories are distributed hierarchically and can be obtained with the **venues/categories** endpoint. When a parent category is specified in the venues search service, there is no need for discriminating all the children. The categories to be filtered were chosen manually with the concern of including diverse types of venues, but all somehow connected to music or night life (*Appendix B*). The objective was to compare the predictions of the system given venues where music plays a central part and others where people go for social or other diverse reasons. The first web service returns few information about each filtered venue, like the city and country where is located and the associated categories. This data can be detailed in the following service.

To get details for a particular venue, including customers and their music tastes, **GetVenueDetail** web service can be used. This service uses a caching mechanism so that a venue doesn't have to be retrieved by the Foursquare API if it has already been saved on the database. To control this, apart from the ID of the venue, a boolean[35] parameter is provided. If **update** is set to false (or 0), the information about the venue (including its customers) is only retrieved from the API if it doesn't exist on the database. Still, it is always checked if the venue is currently open. If **update** is set to true (or 1), the customers of the venue are always updated, even if the venue exists. The database distinguishes between two types of foursquare users. A mayor is seen as the best client of a venue, having the highest number of check-ins in the last two months. The tippers are users that left some kind of feedback on the venue's page.

[34] Padrão de interface de objeto em modelos que usam bases de dados relacionais.
[35] Data type with only two possible values.

The method to get a venue's detail return, for each customer, its associated Facebook ID or Twitter username, in case these accounts are connected to Foursquare. In order to infer the users music tastes (and, as consequence, the music profile of the venues they visit), each of these accounts will be queried for data related to music.



Facebook user profiles are getting more and more detailed, allowing the users to share different information connected to sports, books, movies or music. To get favorite bands, the music page will be used. In spite of constituting a huge habits database, the privacy settings in this platform are getting stricter, making it more difficult to retrieve any kind of information. In addition, Facebook introduced with the second version of its Graph API even more limiting rules to developers, that can now access most information only if a the user is already using their application and has given permissions to it. For this reason, the first version of the API was used, together with a more intrusive method called scraping. This method simulates web page requests, so it can access the source code of a page, whose DOM is searched for certain elements.

To perform the task of getting data using the HTTP protocol, cURL software tool was used. After getting the username associated with the Facebook ID, provided by Foursquare, cURL has to simulate a login to a page using a valid username/password combination. This is necessary because Facebook doesn't allow the access to its contents for non-authenticated users. To achieve this, a first login is attempted, with the username/password set as post parameters. This attempt fails, but it allows to get the cookies needed for the second login attempt. With the correct cookies, cURL can authenticate a user. After this, the next page to be requested is the music page of the user. With the source code retrieved, the artist names can be obtained, knowing that Facebook pages follow a fixed pattern in HTML formatting. However, a second problem arises. In order to improve the content loading, Facebook break each page into small chunks of HTML that are loaded asynchronously (*Figure 14*). For this reason, it is impossible to get a well formed HTML DOM, essential to use an efficient query method like XPath. The solution to get the artists names will be based on regular expressions. Previously to every "Musician/Band" label, each artist has a HTML link element, having as title the artist name and as href the artist website.

```
<a class="_gx7" href="https://www.facebook.com/rodrigoleaomusic?ref=profile" title="Rodrigo Le&#xe3;o" target="_blank">Rodrigo Leão<span data-hover="tooltip" data-tooltip-position="right" class="_56_f _5dzy _5dzz" id="u_0_4f"></span></a><div class="_1fs8 fsm fwn fcg">Musician/Band
```

Figure 14 - Chunk of HTML code containing an artist link

There are other href elements in the artists' area, so the regular expression has to extract all innermost link elements, i.e. all the href elements not containing a similar element within, next to the "Musician/Band" label (*Figure 15*).

```
/<a(?:.(?!<a))*?(M|m)usician\/(B|b)and/
```

Figure 15 - Regular expression to extract the artist link

After the artists are extracted, the final step is to map them using the MusicBrainz API and store the MBID, artist name and the artist link in the database.

The process for retrieving artists or bands from the users' Twitter profile is simpler and is based in the one proposed by Hauger and Schedl in "Exploring Geospatial Music Listening Patterns in Microblog Data". The Twitter API allows developers to obtain users timelines through the **user_timeline** endpoint. The next step is to filter the tweets with a hashtag related to music (see *Table 1*). However, the ambiguity in the tweet's format makes it difficult to extract the artist name. Normally, users post music (or plugins in their behalf) in the format artist/track, using different patterns like **song title – artist** or **song title by artist**, but they can also post list of their favorite artists. To get better results, the potential artists are again mapped using the MusicBrainz API before being saved in the database.

#nowplaying	**used by Hauger/Schedl**
#np	used by Hauger/Schedl
#itunes	used by Hauger/Schedl
#musicmonday	used by Hauger/Schedl
#mm	
#thisismyjam	used by Hauger/Schedl
#spotify	
#onheadphones	
#grooveshark	
#lastfm	
#stereomood	

Table 1 - List of music-related hashtags

Finally, the **GetArtistSimilarity** web service is used to compare the user's music profile with the venues' music profiles (built by the artists found on their customer's profiles). In order to facilitate the task, it will be done with the help of Last.fm API, wrapped with a PHP library developed by Felix Bruns (https://github.com/fxb/php-last.fm-api). In the venues artists list, the used algorithm takes in account the origin of the found artists, valuing more an artist favored by a mayor then by a tipper. There is also the preoccupation of normalizing the similarities to a percentage scale. The algorithm is summarized on *Figure 16*.

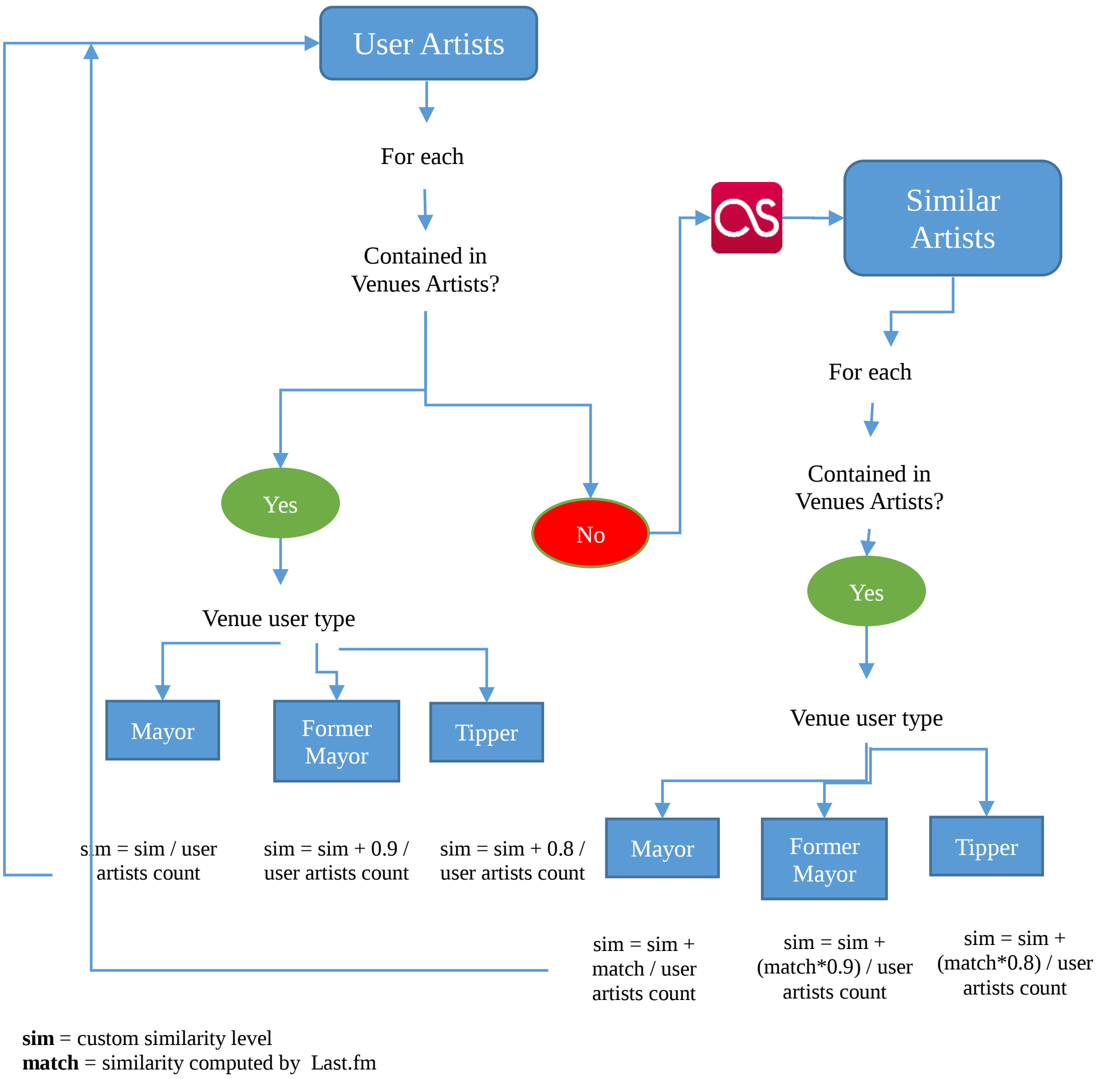


Figure 16 - Artist similarity algorithm

The algorithm starts iterating through the user artists list, mapping each artist with the MusicBrainz API. If the artist also exists in the venue's profile, a parcel is added to the similarity total, with a value weighted by the type of customer originating the artist: if the artist comes from a mayor, the value is only divided by the number of users (to normalize the total); if the artist comes from a formal mayor (a user that has been overtaken by the current mayor), the value is multiplied by a 90% factor; if the artist comes from a tipper, the value is multiplied by a 80% factor. If there isn't a direct correspondence, the similar artists list is obtained as a fallback with Last.FM. The list is built based on the Last.FM collaborative engine (for example, if an artist appears paired with other artist in different users lists, they can be attributed a certain amount of similarity). This list is iterated and it is verified if each artist exists in the venue's list. If so, a parcel is added to the total, with the same weighing as in the previous case, but this time taking also in account the matching level found between the two different artists. In the end, a similarity measure is returned, along with a list containing the similar artists that matched a venue's artist.

In chapter 6, the outcome of these web services will be analyzed. The objective is to test if the artists obtained by the web service **GetVenueDetail** can somehow characterize the venue musically. At the same time, we want to know if, given the artists that integrate the venue profile and the user's favorite artists, **GetArtistSimilarity** can provide an accurate measure of the social similarity between the two. To do this, a field study was conducted in the city of Porto, where some venues were visited to ask their customers which bands are their current favorites and which bands they expected to listen in the visited venue.

Finally, the client side, which will be using these web services, is meant to be a mobile application, available for iOS and Android, but that can be also accessed from a web browser in a desktop computer. PhoneGap framework was chosen for allowing software programmers to build multi-platform applications, using JavaScript, HTML5 and CSS3, which reduces the development time by not requiring specific languages like Ojective-C[36]. The native devices of the mobile terminals (like the acceloremeter[37] and the camera) are still included, using plugins that communicate directly with the core of the mobile operating systems. This technology is conjugated with jQuery Mobile, to enable a responsive and attractive design, with smooth page transitions and customized buttons and menus.

The application is based on a map, where music related venues near user's current location are marked. There is also an option to type a specific location in a text box. Google Maps API helps providing a way to display this information in a standardized and accurate manner. It takes advantage from HTML5 persistent storage abilities, using a caching mechanism which maintains a list of venues whose details are retrieved, so the loading gets faster. To get the user's music tastes, Facebook Graph API is

[36] General-purpose programming language used by Apple for its operating system.
[37] Device normally included in smartphones, used to measure the proper acceleration.

implemented on the client-side, with the help of a library developed by Cristoph Coenraets (https://github.com/ccoenraets/OpenFB), which allows the integration of this API in PhoneGap apps with no need for a plugin. When the application is launched, the **onDeviceReady** event is fired, asking for the user to login to his Facebook account. This is the platform used to infer his music tastes. In case of a successful login, the favorite artists are retrieved through his music page and are stored persistently in HTML5 localStorage[38], after the list is converted to a string, using JSON's stringify function (HTML5 localStorage only can save string in the key/value form).

After the login, the user is prompted to allow the recognition of his position, using the best available geolocation device. In case the user is using a mobile device, the GPS method can be used, finding the user location in relation for at least 4 satellites. If the application is ran through a desktop, probably an approximate location will be retrieved using IP address geolocation, based on information about nearby wireless points and the computer's IP address. Obtained the geographic coordinates, the map is loaded *(Figure 17*) and the first web service is called. **GetVenues** obtains the places near the user's location and mark them in the map. The user can also enter a different address in a text box, shifting the center of the map and loading venues for the given location. Each venue's marker has a tooltip that let the user visualize its name and its categories when touched.

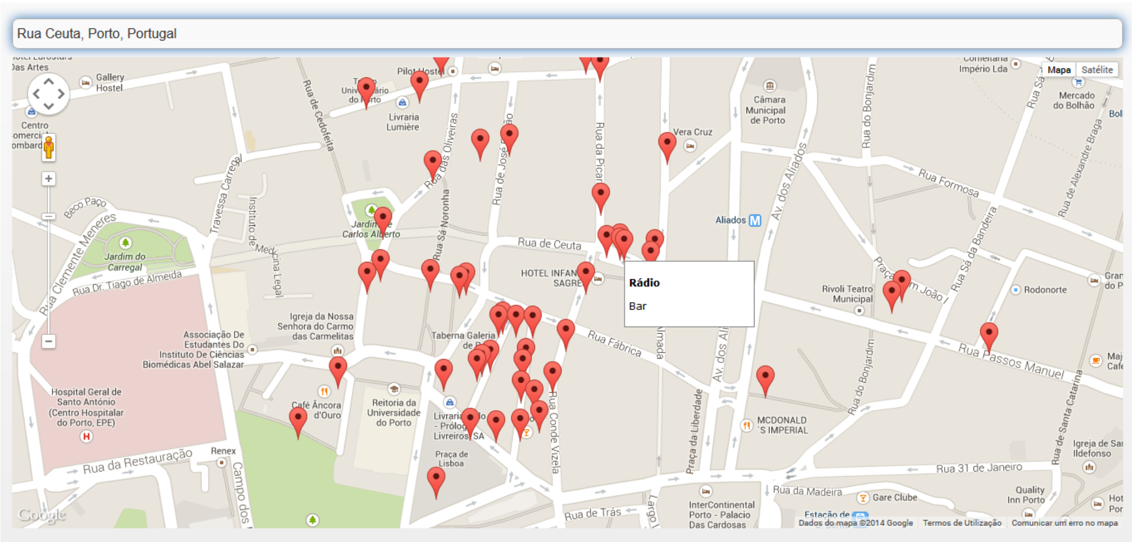


Figure 17 - Tribo mobile application

When a venue's marker is double-touched, **GetVenueDetail** is called, with the update

[38] The localStorage HTML5 object stores information persistently with no expiration date.

parameter set to true in case this is the first click on the venue. A new dialog is opened, showing information about the venue's name and webpage, the opening hours of the place, including the current status and a list of bands that are related to the place and can characterize the bar or the pub musically. **GetArtistSimilarity** is called asynchronously using AJAX to compare the user's music profile (obtained from his Facebook account) and the venue's music profile, obtained from the **GetVenueDetail** web service. When a response is returned, the success callback is invoked and the matching score is displayed on the dialog box.

6. EVALUATION RESULTS

6.1. Tribo 1.0

The first solution for the research problem was based on MMTD, the largest known dataset combining musical posts with geographic information. With 1.090.727 records, corresponding to the same number of tweets, it takes a huge machine load and much optimized algorithms to process all this data. Besides, the adopted process involved at least two API calls and one database insertion per record (when a new user is found, this number increases), which increases the process time. It was estimated that a single record takes at least 3 seconds to be processed; given the approximately one million tweets, the whole process could take more than one month to be completed. Due to this reason, and also because of some problems caused in the remote server, only 532.808 records were processed (approximately half of the total), so the results presented here are just partial and we'll try to extrapolate them for the whole sample. In *Table 2*, *Table 3*, *Table 4* and *Table 5*, a general analysis of the number of cities, venues, users and artists (associated with a venue) found by country is done.

COUNTRY	NUMBER OF CITIES
United States	639
Brazil	333
Indonesia	249
Malaysia	203
Mexico	160
United Kingdom	131
Russia	105
Turkey	81
Philippines	63
Japan	62

Table 2 - Top number of cities per country

COUNTRY	NUMBER OF VENUES
Brazil	3349
United States	2081
Indonesia	2036
Malaysia	1519
Mexico	1126
Russia	875
Turkey	590
Chile	361
United Kingdom	313
Philippines	308

Table 3 - Top number of venues per country

COUNTRY	NUMBER OF USERS
Brazil	1117
Indonesia	732
United States	657
Malaysia	589
Mexico	381
Russia	279
Turkey	199
Chile	138
United Kingdom	111
Philippines	96

Table 4 - Top number o users per country

COUNTRY	NUMBER OF ARTISTS
Brazil	1237
United States	1168
Malaysia	720
Mexico	590
Indonesia	549
Russia	438
Chile	384
United Kingdom	298
Turkey	250
Japan	188

Table 5 - Top number of artists per country

To be more accurate about the descriptions, some rate measures were also calculated (*Table 6*, *Table 7*, *Table 8* ,*Table 9*, *Table 10* and *Table 11*).

COUNTRY	VENUES/CITY
Singapore	20,71429
The Netherlands	17,00000
Brazil	10,05706
Ukraine	9,00000
Russia	8,33333
Indonesia	8,17671
Belarus	7,71429
Malaysia	7,48276
Turkey	7,28395
Mexico	7,03750

Table 6 - Top ratings venues per city

COUNTRY	VENUES/CITY
Bangladesh	1,00000
Iceland	1,00000
Cyprus	1,00000
Nicaragua	1,00000
Norway	1,00000
Bahamas	1,00000
Honduras	1,00000
Croatia	1,00000
Georgia	1,00000
Jersey	1,00000

Table 8 - Bottom ratings venues per city

COUNTRY	USERS/VENUE
Nicaragua	1,00000
Norway	1,00000
Bahamas	1,00000
Honduras	1,00000
Croatia	1,00000
Togo	1,00000
Trinidad and Tobago	1,00000
Andorra	1,00000
Armenia	1,00000
Guyana	1,00000

Table 7 - Top rating users per venue

COUNTRY	USERS/VENUE
Bulgaria	0,09091
New Zealand	0,12121
Maldives	0,16667
Lebanon	0,17647
Argentina	0,20513
Israel	0,21429
Belgium	0,22222
Costa Rica	0,22340
Portugal	0,23810
Greece	0,24194

Table 9 - Bottom ratings users per venue

COUNTRY	ARTISTS/VENUE
Sri Lanka	8,92857
Ghana	7,75000
Bangladesh	6,00000
Nicaragua	4,00000
Norway	4,00000
Puerto Rico	3,03704
Ireland	3,00000
Honduras	3,00000
Sweden	2,58824
Kazakhstan	2,25000

Table 10 - Top ratings artists per venue

COUNTRY	ARTISTS/VENUE
Lebanon	0,17647
Bulgaria	0,18182
Indonesia	0,26965
Israel	0,28571
Martinique	0,30000
Tunisia	0,33333
Mozambique	0,33333
Qatar	0,33333
Brazil	0,36936
Austria	0,40000

Table 11 - Bottom ratings artists per venue

Finally, to better visualize the asymmetric distributions of these ratings, the respective boxplots are presented (*Table 11*, *Table 12*, *Table 13*).

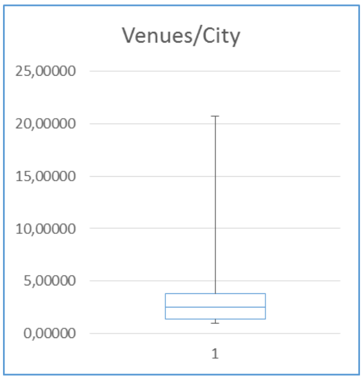


Table 11 - Boxplot venues per city

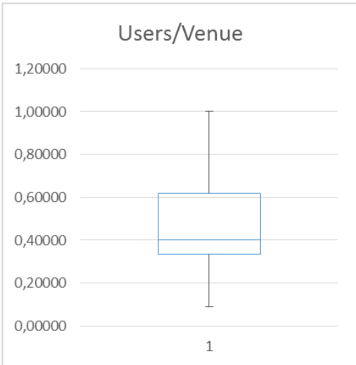


Table 12 - Boxplot users per venue

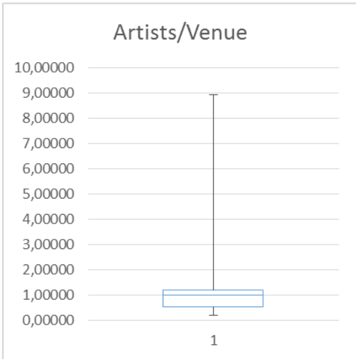


Table 13 - Boxplot artists per venue

6.2. Tribo 2.0

Because of the many questions raised about the liability of describing a venue musically by its customers' tastes, we thought it was important to perform a field validation, asking customers of some of the most known bars, cafes and pubs of Porto about their listening habits. The study was conducted on the 13^{th} and 14^{th} of June 2014 (Friday and Saturday), days usually with a good affluence of people to the city's downtown. The chosen area was the vicinity of Rua de Ceuta, with the use of the client application to retrieve venues in a vicinity range of 500m.

In the questionnaire, the customers were asked to fill two columns, one with their current favorite bands or artists and the other with a list of bands or artists they expected to hear in the venue they were visiting. Other important question was the reason why they chose that particular venue, with possible options being musical, social, geographic, economic or other. In addition, they were asked to fill some personal information, like the name, age and, most important, their Facebook and Twitter ids, so their tastes could be validated against the two social networks chosen in this research to extract musical tastes. However, most people didn't give this information, maybe due to privacy issues or simply because they didn't know it.

The questions were made to 72 persons of all ages, predominantly between 20 and 40 years old. We cannot give an accurate statistic of this variable because most people didn't give their age. Among these people, a great percentage (51.4%) also didn't specified the reason why they were in that particular venue. Among the people that answered this question, almost a half (48.6%) answered they were moved by social reasons, 20% answered other reasons, 14.3% said the music was the important reason, having the same percentage the geographical factor. 2.8% talked about economic reasons.

In the following tables, the customers' musical tastes will be specified, being calculated similarity measures about the venues profiles paired with the users tastes, the users listening expectations and a sample playlist that were registered in the venues, sometimes with the help of the Soundhound[39] mobile application. The customer's artists or bands were joined in a unique list. The customers were distributed by more than 15 places, however the studies are concentrated in 3 bars where it was possible to extract feedback from more people, apart for having, in theory, 3 different profiles: Rádio, a bar located in Praça Dona Filipa de Lancastre (*Table 12*); Casa de Ló, other bar, located in Travessa de Cedofeita (*Table 13*); Aduela, a bar located in Rua das Oliveiras (*Table 14*).

VENUE	Rádio		
Venue artists	**Customers favorite artists**	**Customers expected artists**	**Sample playlist**
Music for Deep Meditation	David Bowie	White Stripes	Cypress Hill
Kasabian	The Velvet Underground	Lemonheads	Gabriel o Pensador
Kings Of Leon	Joy Division	The Strokes	Tom Jones
Lana Del Rey	The Clash	Nada Surf	James Brown
Arcade Fire	Echo & The Bunnymen	Deus	Plan B
Lizy Rize	Radiohead	Morphine	Amy Winehouse
TESA	Mogwai	Arcade Fire	The Strokes
Sonante Mixed Choir	Arcade Fire	The Rolling Stones	Blur
Beirut	The National	Blur	
DIE ANTWOORD	Gregory Porter	Arcade Fire	
FRITZ KALKBRENNER	Rolling Stones	Black Keys	
Efterklang	Otis Redding	Queens of the Stone Age	
BrainStorm	The Cult	Radiohead	
Charles Bradley	The Beatles	Pearl Jam	
PZ	Led Zepellin	Pixies	
Vampire Weekend	U2	Audioslave	
Justin Timberlake	Stone Temple Pilots	Ornatos Violeta	
Iggy Pop	Pearl Jam	The Cure	
Lizard King (the doors tribute)	Soundgarden	Muse	
Chevelle	Mad Season	David Bowie	
Chris Cornell	Alice In Chains	Bruce Springsteen	
Tom Morello	Queens of the Stone Age	Ramones	
	The Black Keys	Walkmen	

[39] Mobile application used to automatically recognize music playing around.

VENUE	Rádio		
Venue artists	**Customers favorite artists**	**Customers expected artists**	**Sample playlist**
	Pink Floyd	The Cult	
	BB King	The Doors	
	The Doors	Arctic Monkeys	
	Pixies		
	Audioslave		
	Ornatos Violeta		
	Sways		
	The Cure		
	The Prodigy		
	Blasted Mechanism		
	Vibe Tribe		
	Talamasca		
	Offspring		
	System of a Down		
	Pennywise		
	Comme Restus		
	Mata-Ratos		
SIMILARITY TO VENUE	**0,19803**	**0,24738**	**0,09815**

Table 12 - Music profiles of venue and customers for Rádio

VENUE	Casa de Ló		
Venue artists	**Customers favorite artists**	**Customers expected artists**	**Sample playlist**
Fado em Si Bemol	Queen	The Beatles	Sonic Youth
Lux Exterior	Muse	Nina Simone	Pavement
Taisag Porto	U2	Chuck	The Rolling Stones
Luiger Lima	Love of Lesbian	LCD Soundsytem	The Locust
Tallowate	Guns and Roses	Black Keys	Vacationer
Andre Indiana	Red Hot Chilli Peppers	Arcade Fire	Los Campesinos! Pulled Apart by Horses
Ace Produktionz	Joy Division	Pixies	
Joana Barra Vaz	The Editors	The Kills	The Vaccines
Expensive Soul	Los Planetas	She Wants Revenge	
Feel Smooth	Pixies	Sonic Youth	
Mao Morta	The Smiths	Elvis Presley	
THE PIANO BATTLE	System of a Down	The Phantom	
Dream Theater	Jimmy Hendrix		
Rage Against The Machine	Led Zepellin		
IRIS - Oficial Webpage	The Doors		
Ana Moura	Jethro Tull		
Peter Murphy	Deep Purple		
DJ LOOPS	Black Sabbath		
Pickle Puss	Bob Marley		
Brahms-Composer	Chuck Berry		

VENUE	Casa de Ló		
Venue artists	**Customers favorite artists**	**Customers expected artists**	**Sample playlist**
JOSE MALHOA	Little Richard		
Adriana Calcanhotto	The Animals		
Gabriel o Pensador	Creed		
Brahms-Composer	Radiohead		
Projecto Sem Nome	Portishead		
Helena Sarmento	Massive Attack		
Robert Schumann	Moulinex		
Alceu Valença	Yo La Tengo		
Chico Buarque	Belle and Sebastian		
Antonio Carlos Jobim - Verve Records	Sonic Youth		
Luiz Gonzaga	Deus		
Sixto Rodriguez - The Sugar Man	Manic Streat Preachers		
Genesis	Pavement		
Official Violent Femmes	Spiritualized		
MPB4	Ornatos Violeta		
Presto Duo	Metallica		
Capicua	Craig David		
Martin Garrix	Elvis Presley		
Positive Vibes	Morphine		
Black Coffee	Tindersticks		
Distant Ship	Timber Timbre		
Justice	Django Django		
Zoo kid	Nick Cave		
Ian Curtis			
Bauhaus			
Yuksek			
Blind Zero			
Dead Combo			
TERRAKOTA			
SIMILARITY TO VENUE	**0,05413**	**0,08422**	**0,01680**

Table 13 - Music profiles of venue and customers for Casa do Ló

VENUE	Aduela		
Venue artists	**Customers favorite artists**	**Customers expected artists**	**Sample playlist**
FOLLAKZOID	Parliament-Funkadelic	The National	Sérgio Godinho
The Handsome Family	Sly & The Family Stone	The Strokes	Zero7
Vampire Weekend	Manu Chao	Morphine	Capitão Fausto
Bright Eyes	C2C	Nick Cave	Tame Impala
Conor Oberst	Erykah Badu	Muse	Black Keys
Morto Coltese	Ornatos Violeta	Queens of the Stone Age	Nicola Conte
Guadalupe Plata	Arthur H	Tame Impala	Billie Holiday
Will Stratton	Grant Lee Buffalo	Radiohead	Chet Baker
Antes Cowboy que Toureiro	Tegan and Sara	Nick Cave	
Throes + The Shine	Queens of the Stone Age	Joy Division	
courtney barnett	Radiohead	Depeche Mode	
Noiserv	Queen	Jefferson Airplane	
Blaya	Black Keys	Vampire Weekend	
Ena Pa 2000	Tom Waits	Arcade Fire	
Mao Morta	Robert Wyatt	Pixies	
YO-LANDI VI$$ER	Nick Cave	The Cure	
Xutos & Pontapes	Pavement	Peter Murphy	
Dead Combo	Giant Sand		
The Black Keys	Arcade Fire		
Silence 4	Django Django		
Artic Monkeys	At the Drive-In		
Teofilo Sonnemberg - Berg	The Talking Heads		
	Mogwai		
	Michael Jackson		
SIMILARITY TO VENUE	**0,11470**	**0,16856**	**0,26353**

Table 14 - Music profiles of venue and customers for Aduela

6.3. Discussion

Even knowing that the records analyzed in MMTD didn't cover the all dataset (approximately half of the more than 1 million tweets were processed), it is possible to reach some conclusions from the available data and try to explain why this solution was discontinued.

From *Table 1*, *Table 2*, *Table 3* and *Table 4* we can see the geographic distribution of the dataset. With no surprise, most of the countries scoring better in the four indicatives are also countries with a great Twitter implementation, with the United States and Brazil appearing in the top two places of number of cities, venues, users and artists represented in our database. The only surprise here can be the fact that Brazil always outcomes United Kingdom, the second placed country in the percentage of users in 2012, but this can be explained by the fact that more than 500.000 records were not analyzed. We can see in the tops also countries with many Foursquare users, like Indonesia, Malaysia and the Philippines, fact that confirm that our system found some users using simultaneously the two social platforms.

However, these results confirm that Twitter was not a good starting point, with a great implementation outside Europe (excepting the United Kingdom) as we can see in *Table 15*. This fact could make it difficult to test the application in a real environment, since the research were based in Portugal. Being a small country it's natural to represent a small percentage of the tweets, however Portugal just scored 35º in the number of cities (11), the same place in the number of venues (42, a very reduced number), 45º in the number of users (10) and 26º in the number of artists (79). This last indicative could give some good perspectives in the building of music profiles of the users, however, the low intersection with tweets related to Foursquare would be a problem.

COUNTRY	% OF USERS
United States	50.99
United Kingdom	17.09
Australia	4.09
Brazil	3.44
Canada	2.92
India	2.87
France	1.76
Indonesia	1.43
Iran	0.88
Ireland	0.85

Table 15 - Twitter worldwide percentage of users by 2012

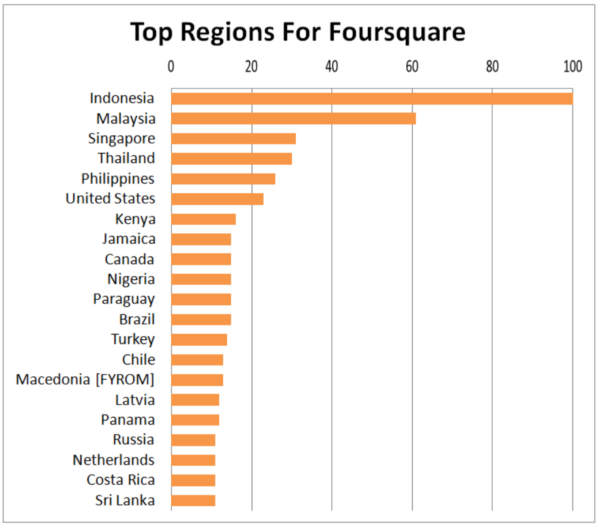


Table 16 - Foursquare worldwide percentage of users by 2012

A more interesting analysis was the one involving rates instead of absolute values. When calculating the number of venues per city, the number of users per venue and the number of artists per venue, it's easy to see that there isn't a significant number of intersected music and check-in posts through the users. From *Table 11*, we can see a very asymmetric distribution, the boxplot showing that 75% of the countries have less than 5 listed venues per city. Results from *Table 12* advice even more against this solution. The distribution presents a negative skew, with every country showing that there is not more than one user listed per venue. *Table 13* showed the same asymmetric distribution and clearly concentrated in a low rating level of artists per venue.

These results, and the low efficiency of the algorithm processing the data, proved that it was a good decision changing the approach towards the problem. The second solution, loading the venues dynamically according to the user's location, didn't allow the same kind of analysis, but it's known beforehand that all the Foursquare venues have some users posting some tips and that most of them have an associated Twitter or Facebook account, so it will not be so difficult to relate users and artists to the places.

The question imposed in this matter is whether the artists found in the profiles of the customers could somehow characterize the venues musically and here there is a lot of uncertainty. To have a better insight, a field study was performed, which confirmed some of our bigger worries about the answers given to the problem. To start, we see that a very small percentage of individuals answering the questionnaire pointed out music as a factor to elect a particular place to go out. This can point to the direction that more and more people go out just to be with friends and drink cheap, making it difficult to advice venues based on music. But this assumption can be better explored with the next few results, which are calculations of similarity levels between venues' music profiles, obtained through open information on the internet and real profiles of real people. It is assumed beforehand that the algorithm calculating the similarities, being based on the Last.FM accurate social system, works well.

The similarity results confirmed somehow our negative perspectives, showing values below 25%, but suggested at the same time the different nature of different venues. As expected, Rádio was the bar with the highest scores, showing a 20% overlap between its music profile and the customers' tastes, a 25% similarity to what people expect to listen, but only a 10% value with the sample playlist. This last variable is not very reliable, since there are few songs listed and they could have been played in a specific period of the night or by a specific DJ. People who know the nightlife of Porto see Rádio as a bar frequented by music lovers, knowing what to expect regarding music and these results confirm it.

Aduela was specifically chosen by being a bar where people go to hang out, eat some Portuguese delicacies and drink good wine. The lower similarity results confirm it, however one could expect even lower results. Particularly, the high similarity level found between the music profile extracted by our system and the sample playlist obtained in site is kind of surprising. This, however, can be in part explained by the not so varied playlist, making the songs played more expectable.

The bigger surprise came with the results of Casa do Ló. This bar, in spite of having many daytime customers, who go there to study or have a tea, is known by giving much importance to the quality of music and DJs that hire to play there. At the same time, the bar has some thematic nights, when the playlist is controlled by people having varied music tastes and this fact can explain the low correspondence between customers and the music profile. People just don't know what to expect in a particular night.

It could be also interesting to do this calculations with individual customers, instead of building a large customers' profile, particularly trying to relate their motivations with the similarity levels found, however there were very few people stating that music was their motivation to go out and this can be done in a future research.

7. CONCLUSIONS

The research realized in this project was one more valuable contribution to the evolving area of the recommendation systems. The main objective was to advice venues taking in account the musical tastes of the customers, trying to go further than most of the location-based systems, which recommend places based on ratings and popularity. This system is more capable of being directed to a specific niche of people who love music.

Still, this investigation started with a not proven assumption that the music played on a bar or a pub could be predicted by the favorite artists of the customers. Although we had very strong indications that the social factors are getting even more importance when it comes to go out and that people tend to devalue music, given the abundance and trivialization brought by the digital evolution, it was confirmed that the very nature of venues is much varied and that the traditional bar environment, with people talking about music and suggesting tracks to the DJ still exists.

Some technical problems were also found, being proved that, even with the increasingly generic character of social networks like Facebook, where people can nowadays share their interests about any domain, and no matter the efforts done by some projects like DBPedia, it is still very difficult to cross multi-domain data. Facebook is taking strong steps to solve this problem, having connections with most of the social networks, however people still face the problem of the so-called “social network fatigue”, don’t wasting their time filling repetitive forms whenever they want to join a new platform.

Other recurring problem is the way to extract data from the internet, with the APIs still posing many restrictions to the developers. Most of them are publicly available but impose request limits to the users, which can make the scripts that process this information much slower. With an application to be used by mobile clients, the web services lie on a remote server and one of the biggest issues still found is the loading time of venue details and the slow calculation of similarity levels.

In the end, even with its utility and liability to be proven, it was possible to develop a much valid system that objectively can link venues and places to people and their music tastes, which, in conjunction with the built database, can constitute a very interesting starting point for future investigations. This project also implements a hybrid data extraction method, combining APIs with scraping, offering different solutions to the problem of getting information from the internet.

7.1. Future work

After finished the first stage of the project, having a recommendations system more efficient and refined, the next step is to go social. Tribo has the potential of becoming a night life social platform. There is the idea of incorporating new features, like the possibility of registering in real time the tracks or artists played in a specific venue, connecting applications like Soundhound (https://www.soundhound.com/) or Shazam (http://www.shazam.com/). This would also increase the reliability of the venues music profiles, improving the not precise cold-start database. Other approach to solve the cold-start problem could be the inclusion of content-based descriptors, which could attribute high-level music characteristics to a bar as a fallback.

It would be also interesting to implement a platform of social interaction, allowing to leave feedback or arrange meetings in the venue's page. With a Foursquare permission by the user, it would also be possible to know in real time where our music friends are. This could be also lucrative to the bars and pubs, which could use this platform to promote their spaces or propose real-time offers or discounts.

The most important contribution of this research is to lay grounds for new approaches, using the database of venues and music it helped building.

References

Acquisti, A., & Gross, R. (2006). Imagined communities: awareness, information sharing, and privacy on the facebook. *PET'06 Proceedings of the 6th international conference on Privacy Enhancing Technologies* (pp. 36-58). Berlin, Germany: Springer-Verlag.

Bonhard, P., & Sasse, M. A. (2006). 'Knowing me, knowing you' - Using profiles and social networking to improve recommender systems. *BT Technology Journal*, 84-98.

Bugayvchenko, D., & Dzuba, A. (2013). Musical recommendations and personalization in a social network. *RecSys '13 Proceedings of the 7th ACM conference on Recommender systems* (pp. 367-370). NY, USA: ACM.

Burke, R. (2002). Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, 331-370.

Cantador, I., Bellogín, A., & Vallet, D. (2010). Content-based recommendation in social tagging systems. *RecSys '10 Proceedings of the fourth ACM conference on Recommender systems* (pp. 237-240). NY, USA: ACM.

Caruso, R. (2011, April 15). *The Social Media Parallel Universe – Science Meets Social*. Retrieved from Bundle Post: http://bundlepost.wordpress.com/2011/04/15/the-social-media-parallel-universe-science-meets-social/

Celma, O. (2008). *Music Recommendation and Discovery in the Long Tail.* Barcelona, Spain.

Fernández-Tobías, I., Cantador, I., Kaminskas, M., & Ricci, F. (2011). A Generic Semantic-based Framework for Cross-domain Recommendation. *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems* (pp. 25-32). NY, USA: ACM.

Goecks, J., & Shavlik, J. (2000). Learning users' interests by unobtrusively observing their normal behavior. *Proceedings of the 5th international conference on Intelligent user interfaces* (pp. 129-132). NY, USA: ACM.

Hauger, D., & Schedl, M. (2012). Exploring Geospatial Music Listening Patterns in Microblog Data. *AMR 2012 - 10th international workshop on Adaptive Multimedia Retrieval.* Copenhagen, Denmark.

Hauger, D., Schedl, M., Kosir, A., & Tkalcic, M. (2013). The Million Musical Tweets Dataset: What can we learn from Microblogs. *ISMIR*, 189-194.

Herrera, P., Celma, O., Massaguer, J., Cano, P., Gómez, E., Gouyon, F., . . . Wack, N. (2005). MUCOSA: A Music Content Semantic Annotator. *MUCOSA: A Music Content Semantic Annotator.* London, UK: DBLP.

Krause, S. (2006, January 30). *Pandora and Last.fm: Nature vs. Nurture in Music Recommenders* . Retrieved from Words & Numbers: http://blog.stevekrause.org/2006/01/pandora-and-lastfm-nature-vs-nurture-in.html

Licklider, J. C. (1963). *Memorandum For Members and Affiliates of the Intergalactic Computer Network.* Washington, D.C.: Advanced Research Projects Agency.

Liu, H., & Maes, P. (2005). Interestmap: Harvesting social network profiles for recommendations. *Beyond Personalization - IUI 2005.* San Diego, USA.

O'Hear, S. (2007, January 2). *Could 2007 be the year of social network fatigue?* Retrieved from ZDNet: http://www.zdnet.com/blog/social/could-2007-be-the-year-of-social-network-fatigue/53

Park, S., Kim, S., Lee, S., & Woon, Y. (2010). Online Map Interface for Creative and Interactive Music-Making. *Proceedings of the 2010 Conference on New Interfaces for Musical Expression.* Sydney, Australia.

Passant, A., & Raimond, Y. (2008). Combining Social Music and Semantic Web for music-related recommender systems.

Savage, N., Baranski, M., & Chavez, N. H. (2011). I’m feeling LoCo: A Location Based Context Aware Recommendation System. In G. Gartner, & F. Ortag, *Advances in Location-Based Services: 8th International Symposium on Location-Based Services* (pp. 37-54). Vienna: Springer.

Schabetsberger, C., & Schedl, M. (2013). Personalized Music Recommendation in a Mobile. *MoMM '13 Proceedings of International Conference on Advances in Mobile Computing & Multimedia* (p. 63). NY, USA: ACM.

Shardanand, U., & Maes, P. (1995). Social Information Filtering: Algorithms for Automating "Word of Mouth". *CHI '95 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 210-217). NY, USA: ACM Press/Addison-Wesley Publishing Co.

Voigt, M., Werstler, A., Polowinski, J., & Meissner, K. (2012). Weighted faceted browsing for characteristics-based visualization selection through end users. *EICS '12 Proceedings of the 4th ACM SIGCHI symposium on Engineering interactive computing systems* (pp. 151-156). NY, USA: ACM.

Walters, T. (2012, November 26). Understanding the “Mobile Shift”: Obsession with the Mobile Channel Obscures the Shift to Ubiquitous Computing. 2. Digital Clarity Group.

Wang, H., Terrovitis, M., & Mamoulis, N. (2013). Location Recommendation in Location-based Social Networks using User Check-in Data. *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (pp. 374-383). NY, USA: ACM.

APPENDIX A

Tribo 1.0 Geographic Distribution

The following is a table that summarizes Tribo 1.0 results, listing the number of cities, places, users and artists found per country as well as the venues per city rate, users per venue rate and artists per venues rate.

COUNTRY	Nº CITIES	Nº VENUES	Nº USERS	Nº ARTISTS	VENUES/CITY	USERS/VENUES	ARTISTS/VENUES
Andorra	1	1	1	1	1,00000	1,00000	1,00000
Argentina	15	39	8	41	2,60000	0,20513	1,05128
Armenia	1	1	1	1	1,00000	1,00000	1,00000
Australia	48	80	21	68	1,66667	0,26250	0,85000
Austria	3	5	2	2	1,66667	0,40000	0,40000
Azerbaijan	2	7	3	4	3,50000	0,42857	0,57143
Bahamas	2	2	2	3	1,00000	1,00000	1,50000
Bahrain	1	2	1	3	2,00000	0,50000	1,50000
Bangladesh	2	2	1	12	1,00000	0,50000	6,00000
Belarus	7	54	23	66	7,71429	0,42593	1,22222
Belgium	53	126	28	61	2,37736	0,22222	0,48413
Brazil	333	3349	1117	1237	10,05706	0,33353	0,36936
Bulgaria	3	11	1	2	3,66667	0,09091	0,18182
Canada	26	81	30	83	3,11538	0,37037	1,02469
Chile	61	361	138	384	5,91803	0,38227	1,06371
China	14	48	12	21	3,42857	0,25000	0,43750
Colombia	15	71	23	86	4,73333	0,32394	1,21127
Costa Rica	35	94	21	48	2,68571	0,22340	0,51064
Croatia	2	2	2	2	1,00000	1,00000	1,00000
Cyprus	2	2	1	4	1,00000	0,50000	2,00000
Czech Republic	1	3	2	3	3,00000	0,66667	1,00000
Denmark	2	3	2	3	1,50000	0,66667	1,00000
Dominican Republic	16	66	17	39	4,12500	0,25758	0,59091
Ecuador	6	21	11	23	3,50000	0,52381	1,09524
Egypt	12	33	12	15	2,75000	0,36364	0,45455
El Salvador	6	13	6	10	2,16667	0,46154	0,76923
Estonia	4	5	2	3	1,25000	0,40000	0,60000
Fiji	1	4	1	2	4,00000	0,25000	0,50000
Finland	7	13	5	7	1,85714	0,38462	0,53846
France	57	193	72	168	3,38596	0,37306	0,87047
Georgia	4	4	1	2	1,00000	0,25000	0,50000
Germany	31	95	43	105	3,06452	0,45263	1,10526
Ghana	3	4	2	31	1,33333	0,50000	7,75000

Gibraltar	1	2	2	2	2,00000	1,00000	1,00000
Greece	28	62	15	28	2,21429	0,24194	0,45161
Guadeloupe	5	7	6	13	1,40000	0,85714	1,85714
Guatemala	10	16	6	13	1,60000	0,37500	0,81250
Guyana	1	1	1	1	1,00000	1,00000	1,00000
Haiti	1	1	1	1	1,00000	1,00000	1,00000
Honduras	1	1	1	3	1,00000	1,00000	3,00000
Hong Kong	2	5	2	5	2,50000	0,40000	1,00000
Hungary	5	21	9	24	4,20000	0,42857	1,14286
Iceland	3	3	2	4	1,00000	0,66667	1,33333
India	25	81	32	103	3,24000	0,39506	1,27160
Indonesia	249	2036	732	549	8,17671	0,35953	0,26965
Iran	2	2	1	1	1,00000	0,50000	0,50000
Ireland	3	7	4	21	2,33333	0,57143	3,00000
Israel	3	14	3	4	4,66667	0,21429	0,28571
Italy	42	130	44	87	3,09524	0,33846	0,66923
Ivory Coast	1	1	1	1	1,00000	1,00000	1,00000
Japan	62	142	46	188	2,29032	0,32394	1,32394
Jersey	2	2	1	2	1,00000	0,50000	1,00000
Jordan	2	7	3	7	3,50000	0,42857	1,00000
Kazakhstan	1	4	2	9	4,00000	0,50000	2,25000
Kenya	7	32	12	22	4,57143	0,37500	0,68750
Kosovo	3	11	6	12	3,66667	0,54545	1,09091
Kuwait	8	25	12	15	3,12500	0,48000	0,60000
Latvia	13	56	20	62	4,30769	0,35714	1,10714
Lebanon	8	17	3	3	2,12500	0,17647	0,17647
Liechtenstein	2	2	1	1	1,00000	0,50000	0,50000
Lithuania	2	5	3	5	2,50000	0,60000	1,00000
Macedonia	1	1	1	1	1,00000	1,00000	1,00000
Malaysia	203	1519	589	720	7,48276	0,38776	0,47400
Maldives	5	18	3	12	3,60000	0,16667	0,66667
Malta	1	1	1	1	1,00000	1,00000	1,00000
Martinique	6	10	3	3	1,66667	0,30000	0,30000
Mauritius	1	1	1	1	1,00000	1,00000	1,00000
Mexico	160	1126	381	590	7,03750	0,33837	0,52398
Morocco	9	34	11	35	3,77778	0,32353	1,02941
Mozambique	2	3	1	1	1,50000	0,33333	0,33333
Nepal	1	2	2	2	2,00000	1,00000	1,00000
Netherlands	45	112	43	134	2,48889	0,38393	1,19643
New Zealand	11	33	4	14	3,00000	0,12121	0,42424
Nicaragua	1	1	1	4	1,00000	1,00000	4,00000
Nigeria	1	2	2	4	2,00000	1,00000	2,00000
Norway	1	1	1	4	1,00000	1,00000	4,00000
Oman	1	4	2	2	4,00000	0,50000	0,50000
Panama	5	20	8	14	4,00000	0,40000	0,70000

Paraguay	9	39	17	69	4,33333	0,43590	1,76923
Peru	24	91	25	70	3,79167	0,27473	0,76923
Philippines	63	308	96	124	4,88889	0,31169	0,40260
Poland	5	8	5	15	1,60000	0,62500	1,87500
Portugal	11	42	10	79	3,81818	0,23810	1,88095
Puerto Rico	9	27	7	82	3,00000	0,25926	3,03704
Qatar	2	9	3	3	4,50000	0,33333	0,33333
Russia	105	875	279	438	8,33333	0,31886	0,50057
Saint Martin	1	1	1	1	1,00000	1,00000	1,00000
Saudi Arabia	7	41	13	20	5,85714	0,31707	0,48780
Serbia	4	6	4	10	1,50000	0,66667	1,66667
Singapore	7	145	75	168	20,71429	0,51724	1,15862
South Africa	27	68	27	47	2,51852	0,39706	0,69118
South Korea	11	47	19	26	4,27273	0,40426	0,55319
Spain	38	119	46	92	3,13158	0,38655	0,77311
Sri Lanka	6	14	5	125	2,33333	0,35714	8,92857
Sudan	1	2	1	1	2,00000	0,50000	0,50000
Suriname	1	1	1	1	1,00000	1,00000	1,00000
Sweden	14	34	11	88	2,42857	0,32353	2,58824
Switzerland	11	16	9	20	1,45455	0,56250	1,25000
Taiwan	11	18	5	12	1,63636	0,27778	0,66667
Thailand	49	103	35	61	2,10204	0,33981	0,59223
The Netherlands	1	17	12	26	17,00000	0,70588	1,52941
Togo	1	1	1	2	1,00000	1,00000	2,00000
Trinidad and Tobago	1	1	1	2	1,00000	1,00000	2,00000
Tunisia	3	3	1	1	1,00000	0,33333	0,33333
Turkey	81	590	199	250	7,28395	0,33729	0,42373
Turks and Caicos Islands	1	1	1	1	1,00000	1,00000	1,00000
Uganda	4	15	4	7	3,75000	0,26667	0,46667
Ukraine	11	99	36	99	9,00000	0,36364	1,00000
United Arab Emirates	6	30	12	28	5,00000	0,40000	0,93333
United Kingdom	131	313	111	298	2,38931	0,35463	0,95208
United States	639	2081	657	1168	3,25665	0,31571	0,56127
Venezuela	7	25	10	17	3,57143	0,40000	0,68000
Vietnam	1	1	1	1	1,00000	1,00000	1,00000
Zimbabwe	2	5	2	2	2,50000	0,40000	0,40000

APPENDIX B

Music Categories List

In the following table are listed the Foursquare music categories considered in the filtering of places.

Name	ID
Arts & Entertainment	4d4b7104d754a06370d81259
Concert Hall	5032792091d4c4b30a586d5c
Country Dance Club	52e81612bcbc57f1066b79ef
Music Venue	4bf58dd8d48988d1e5931735
Salsa Club	52e81612bcbc57f1066b79ec
Event	4d4b7105d754a06373d81259
Music Festival	5267e4d9e4b0ec79466e48d1
Food	4d4b7105d754a06374d81259
Cafeteria	4bf58dd8d48988d128941735
Café	4bf58dd8d48988d16d941735
Coffee Shop	4bf58dd8d48988d1e0931735
Irish Pub	52e81612bcbc57f1066b7a06
Tea Room	4bf58dd8d48988d1dc931735
Nightlife Spot	4d4b7105d754a06376d81259
Bar	4bf58dd8d48988d116941735
Beach Bar	52e81612bcbc57f1066b7a0d
Beer Garden	4bf58dd8d48988d117941735
Brewery	50327c8591d4c4b30a586d5d
Champagne Bar	52e81612bcbc57f1066b7a0e
Cocktail Bar	4bf58dd8d48988d11e941735
Dive Bar	4bf58dd8d48988d118941735
Gay Bar	4bf58dd8d48988d1d8941735
Hookah Bar	4bf58dd8d48988d119941735
Hotel Bar	4bf58dd8d48988d1d5941735
Karaoke Bar	4bf58dd8d48988d120941735
Lounge	4bf58dd8d48988d121941735
Nightclub	4bf58dd8d48988d11f941735
Other Nightlife	4bf58dd8d48988d11a941735
Pub	4bf58dd8d48988d11b941735
Sake Bar	4bf58dd8d48988d11c941735
Speakeasy	4bf58dd8d48988d1d4941735
Sports Bar	4bf58dd8d48988d11d941735
Strip Club	4bf58dd8d48988d1d6941735
Whisky Bar	4bf58dd8d48988d122941735
Wine Bar	4bf58dd8d48988d123941735

	Parent category
	Child category

APPENDIX C

Enquiries List

These are the results of all the field enquiries made on the door of various venues in Porto, Portugal.

Venue: Aduela **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Tom Waits	Nick Cave
Robert Wyatt	Pulp
Nick Cave	Morphine
Pavement	
Giant Sad	

Venue: Aduela **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Arcade Fire	Nick Cave
Django Django	Rage Against the Machine
At the Drive-In	Joy Division
The Talking Heads	Depeche Mode
Mogwai	Jefferson Airplane

Venue: Aduela **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Tom Waits	Joy Division
Nick Cave	Vampire Weekend
Arcade Fire	
Michael Jackson	

Venue: Aduela **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Parliament Funkadelic	The National
Sly and The Family Stone	The Strokes
Manu Chao	Morphine
C2C	Nick Cave
Erykah Badu	Muse

Venue: Aduela **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Ornatos Violeta	Morphine
Artur H	
Grant Lee Bufallo	
Gisela João	

Venue: Aduela **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Yeah Yeah Yeahs	Morphine
Tegan and Sara	Arcade Fire
Faith No More	Pixies
Ornatos Violeta	The Cure
Portico Quartet	Peter Murphy

Venue: Aduela **Visit motivation:** Geographic

Favourite Bands/Artists	Bands/Artists expected to listen
Queens of the Stone Age	Queens of the Stone Age
Radiohead	Tame Impala
Queen	Radiohead
Ornatos Violeta	The Hives
Black Keys	Black Keys

Venue: Moonshine Pub **Visit motivation:** Musical

Favourite Bands/Artists	Bands/Artists expected to listen
Metallica	Led Zepellin
Manowar	Metallica
Machine Head	Offspring
Blind Guardian	

Venue: Moonshine Pub **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Queen	Rage Against the Machine
Dire Straits	Metallica
The Prodigy	System of a Down
Radiohead	Depeche Mode
Rui Veloso	Korn

Venue: Moonshine Pub **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Queen	
HIM	
Ornatos Violeta	
Radiohead	
Rui Veloso	

Venue: Moonshine Pub **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Portishead	
Rolling Stones	
Resistência	

Venue: Moonshine Pub **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Rage Against the Machine	Rage Against the Machine
The Prodigy	The Prodigy
Max Cooper	Max Cooper
Daft Punk	Daft Punk

Venue: Moonshine Pub **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Placebo	Muse
Muse	
The Killers	
Ben Harpers	

Venue: Casa de Ló **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Morphine	Billie Holliday
Tindersticks	Tame Impala
Timber Timbre	Dead Man's Bones
Django Django	
Nick Cave	

Venue: Casa de Ló **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Love of Lesbian	The Beatles
Guns and Roses	Elvis Presley
Red Hot Chili Peppers	Nina Simone
Extremoduro	The Phantom

Venue: Casa de Ló **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Metallica	
Children of Bodom	
Craig David	
Elvis Presley	

Venue: Casa de Ló **Visit motivation:** Musical

Favourite Bands/Artists	Bands/Artists expected to listen
Yo La Tengo	LCD Soundsystem
Belle and Sebastian	Black Keys
Sonic Youth	Arcade Fire

Venue: Casa de Ló **Visit motivation:** Musical

Favourite Bands/Artists	Bands/Artists expected to listen
dEUS	Pixies
Manic Street Preachers	The Kills
Pavemente	She Wants Revenge
Spiritualized	Sonic Youth
Ornatos Violeta	

Venue: Casa de Ló **Visit motivation:** Geographic

Favourite Bands/Artists	Bands/Artists expected to listen
Radiohead	Joy Division
Portishead	Vampire Weekend
Massive Attack	
Moulinex	
Muse	

Venue: Casa de Ló **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Chuck Berry	
Litlle Richard	
The Animals	
Creed	
Judas Priest	

Venue: Casa de Ló **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
The Smiths	
Jethro Tull	
System of a Down	
Deep Purple	
Jimmy Hendrix	

Venue: Casa de Ló **Visit motivation:** Other

Favourite Bands/Artists	Bands/Artists expected to listen
Joy Division	Nina Simone
The Editors	The Beatles
Los Planetas	Chuck
Pixies	

Venue: Casa de Ló **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Queen	The Beatles
Muse	
U2	

Venue: Rádio **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
The National	David Bowie
Rolling Stones	Bruce Springsteen
Otis Redding	The Strokes
The Cult	Ramones
The Beatles	Walkmen

Venue: Rádio **Visit motivation:** Geographic

Favourite Bands/Artists	Bands/Artists expected to listen
Led Zepellin	Cult
The Beatles	The Doors
Rolling Stones	Rolling Stones
U2	The Strokes
Cult	Arctic Monkeys

Venue: Rádio **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Stone Temple Pilots	Blur
Pearl Jam	
Mad Season	
Soundgarden	
Alice in Chains	

Venue: Rádio **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Arcade Fire	Arcade Fire
Queens of the Stone Age	Black Keys
Black Keys	

Venue: Rádio **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
David Bowie	White Stripes
Velvet Underground	Lemonheads
Joy Division	The Strokes
The Clash	Nada Surf
Echo & the Bunnymen	The Clash

Venue: Rádio **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Radiohead	dEUS
Mogwai	Morphine
Arcade Fire	Arcade Fire
The National	Rolling Stones
Gregory Porter	White Stripes

Venue: Rádio **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
The Prodigy	Muse
Blasted Mechanism	Pixies
Vibe Tribe	
Talamasca	

Venue: Rádio **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Offspring	
System of a Down	
Pennywise	
Come Restus	
Mata Ratos	

Venue: Rádio **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Pink Floyd	Queens of the Stone Age
Led Zepellin	
BB King	
The Doors	

Venue: Rádio **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Radiohead	Radiohead
Pearl Jam	Pearl Jam
Pixies	Pixies
Audioslave	Audioslave
Ornatos Violeta	Ornatos Violeta

Venue: Rádio **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Swans	The Cure
Joy Division	Radiohead
The Cure	Pearl Jam

Venue: Lusitano **Visit motivation:** Musical

Favourite Bands/Artists	Bands/Artists expected to listen
Bill Evans	Hot Chip
Nujabes	Ornatos Violeta
Michel Petrucciani	Peaches
Ornatos Violeta	New Order

Venue: Baixaria **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Jeff Buckley	
Chick Corea	
Sex Pistols	
Mutemath	

Venue: Baixaria **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Pearl Jam	
Metallica	
Arctic Monkeys	
Spice Girls	
Limp Bizkit	

Venue: Baixaria **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Rolling Stones	Bamba Social
Keb-Mo	Ornatos Violeta
João Bosco	Blame Zeus
Spock Freud Orchestra	Orquestra Jazz Matosinhos

Venue: Baixaria **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Bob Marley	Primitive Reason
Carlos Paredes	Led Zepellin
António Variações	Bob Marley
Ana Moura	Guns and Roses
	UB40

Venue: Era Uma vez no Porto **Visit motivation:** Other

Favourite Bands/Artists	Bands/Artists expected to listen
Xutos e Pontapés	Violent Femmes
Black Keys	System of a Down
Natiruts	Ornatos Violeta
Limp Bizkit	Pixies
System of a Down	Pearl Jam

Venue: Era Uma Vez no Porto **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Pixies	
Nirvana	
Nina Simone	
The National	

Venue: Era Uma Vez no Porto **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Nina Simone	Police
Ornatos Violeta	Joy Division
Chet Baker	Two Doors Cinema Club
Foo Fighters	Primal Scream
Tom Jobim	

Venue: Era Uma Vez no Porto **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Pixies	Metronomy
The Knife	Primal Scream
Hot Chips	Pixies
Faith No More	David Bowie
Mão Morta	Police

Venue: Era Uma Vez no Porto **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Radiohead	Phoenix
Queens of the Stone Age	Police
Pixies	
Portishead	
Little Annie	

Venue: Livraria da Baixa **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Queen	Seu Jorge
The Beatles	Xutos e Pontapés
Radiohead	Queen
Jorge Ben	The Beatles
Amália	Ornatos Violeta

Venue: Gare **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Josh Wink	Hardfloor
Dub Fire	Trentmoller
Pig & Dan	
Paco Osuna	
Steve Parker	

Venue: Pitch **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Beethoven	Jamiroquai
Dire Straits	The Beatles
Radiohead	Chemical Brothers
Jeff Buckley	Freak Power
Keith Jarrett	Proppeller Heads

Venue: Pitch **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Arctic Monkeys	Arcade Fire
Led Zepellin	Smoke Jonic
Pink Floyd	Silencio (Prins Tomas Remix)
Black Keys	
Arcade Fire	

Venue: Pitch **Visit motivation:** Geographic

Favourite Bands/Artists	Bands/Artists expected to listen
Kasabian	
Erlend Oye	
The Withest Boy Alive	
The Smiths	
The Doors	

Venue: Pitch **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Paul Kalkbrener	
Freshkitos	

Venue: Café Vitória **Visit motivation:** Other

Favourite Bands/Artists	Bands/Artists expected to listen
The XX	The XX
Arcade Fire	Haim
Disclosure	Disclosure
Young the Giant	London Grammar
The National	

Venue: Café Vitória **Visit motivation:** Other

Favourite Bands/Artists	Bands/Artists expected to listen
Zeca Afonso	
Fausto	
José Mário Branco	
Elis Regina	
Sigur Rós	

Venue: Café Vitória **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Tom Waits	
Divine Comedy	
Beirut	
Nina Simone	
Joanna Newson	

Venue: Café Vitória **Visit motivation:** Musical

Favourite Bands/Artists	Bands/Artists expected to listen
Mindy Smith	
Dixie Chicks	
Ella Fitzgerald	
Cristina Branco	
Sétima Legião	

Venue: Café Vitória **Visit motivation:** Other

Favourite Bands/Artists	Bands/Artists expected to listen
Sky Ferreira	
António Zambujo	
Nick Cave	
Dear Telephone	
Justin Timberlake	

Venue: Café Vitória **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Bjork	
Radiohead	
M.I.A.	

Venue: Café Vitória **Visit motivation:** Other

Favourite Bands/Artists	Bands/Artists expected to listen
Bon Iver	Pharrell
SBTRK	Rihanna
M.I.A.	Peixe Avião
Rihanna	Best Youth
Lykke Li	Betthoven

Venue: Café Vitória **Visit motivation:** Social

System of a Down	Bands/Artists expected to listen
Coldplay	
Britney Spears	
Toranja	
Katy Perry	
Shania Twain	

Venue: Café Vitória **Visit motivation:** Other

Favourite Bands/Artists	Bands/Artists expected to listen
Tops	AfroBeat
Clipping	Technobreaga
Sophie	
Phédre	
Fatima	

Venue: Café Vitória **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Smashing Pumpkins	
Hole	
Crazy Town	
Guano Apes	
Garbage	

Venue: Café Vitória **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Heavenwood	Linda Martini
Linda Martini	PAUS
Balkan Beat Box	Portishead
Portishead	Balkan Beat Box

Venue: Café Vitória **Visit motivation:** Social

Favourite Bands/Artists	Bands/Artists expected to listen
Two Doors Cinema Club	
The Vaccines	
M.I.A.	
Massive Attack	
Eminem	

Venue: Fé **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Beyoncé	Beyoncé
Eminem	Rhianna
Rhianna	Justin Timberlake
Justin Timberlake	Anselmo Ralph
Valete	Seu Jorge

Venue: Rendez-vous **Visit motivation:** Musical

Favourite Bands/Artists	Bands/Artists expected to listen
Pink Floyd	Dire Straits
Dire Straits	Rolling Stones
Rolling Stones	
The Doors	
Jimmy Hendrix	

Venue: Piolho **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
The XX	
Mumford & Sons	
Coldplay	
The Gift	
Radiohead	

Venue: Piolho **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Black Eyed Peas	
Natiruts	
The Gift	
Coldplay	
Sorriso Maroto	

Venue: Piolho **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Skillet	
3DG	
Evanescence	
Linkin Park	
AFI	

Venue: Piolho **Visit motivation:** Other

Favourite Bands/Artists	Bands/Artists expected to listen
Azeitonas	
Silence 4	
Nelson Freitas	
Anselmo Ralph	

Venue: Piolho **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Muse	
Silence 4	
Lykke Li	
Spice Girls	
Evanescence	

Venue: Gin Tónico **Visit motivation:** Musical

Favourite Bands/Artists	Bands/Artists expected to listen
U2	
Pixies	
David Bowie	
Pearl Jam	

Venue: Gin Tónico **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
U2	
Pearl Jam	
Coldplay	
Muse	
Police	

Venue: Gin Tónico **Visit motivation:**

Favourite Bands/Artists	Bands/Artists expected to listen
Pearl Jam	U2
U2	Pearl Jam
José Cid	