Finding Music Fads by Clustering Online Radio Data with Emergent Self Organizing Maps

Florian Meyer and Alfred Ultsch

Abstract Music charts provide a simple statistic of sold records. Web 2.0 provides social networks, where detailed information from listeners is available. In particular, there are keywords, so called tags, that are given by the network members to classify songs into genres.

An important topic are music fads, i.e., small time intervals of a few weeks with a strong presence of similar music genres. We introduce a distance on the weekly music charts to uncover music fads. Fads are visualized using Emergent Self Organizing Maps (ESOM). They are automatically found by analysing the progress of the impact of music genres. This algorithm does not rely on an estimation of the number of fads. Dominant genres of the fads were found to characterize them.

Keywords Clustering · Music fads · Self organizing maps.

1 Introduction

To find the right time for placing a song in the market is very important. Music genres (like colours for clothes, shoe brands, …) come in and out of fashion. Such fashions last often only for a few weeks. To use these brief fashions, so called fads, we developed an easily usable method to visualize and analyse the behaviour of fashions by observing online radio data. A great benefit of using online data is, that they are both free and up to date. It is an easy and cheap way to follow the fads.

Tagging is often referred to as the process of assigning keywords to a special group of objects and is an important feature of community based social networks like Flickr, YouTube or Last.fm. We used the user-generated descriptions of Last.fm to generate features that describe songs. Tagging is already used by many users classifying items, being controlled by the creator and consumer of the content. For our study we chose to analyse the data provided by the music community Last.fm,

Florian Meyer (✉)
Data Bionics Research Group, University of Marburg, 35032 Marburg, Germany.
e-mail: meyer@mathematik.uni-marburg.de

A. Fink et al., (eds.), Advances in Data Analysis, Data Handling and Business Intelligence, Studies in Classification, Data Analysis, and Knowledge Organization, DOI 10.1007/978-3-642-01044-6_38, © Springer-Verlag Berlin Heidelberg 2010
an internet radio broadcaster featuring a music recommendation system. The users can assign tags to songs. Tags make it possible to organize the media songs in a semantic way and makes a useful base for discovering new music trends. Because of the huge amount of songs offered by the online radio station, it is necessary to reduce the online data. One possible way is to concentrate on the most important (meaning most often heard) songs, but this way much information is lost. Another problem with using song charts is that hit songs usually stay only for a few weeks at the top of the charts – but that does not mean, that after a song has gone there is a new music style. A better way is to transform the song charts into genre vectors. We use tags to assign genres to the songs and look at which genre becomes popular instead of analysing songs directly. Another advantage of examining genres instead of songs is the possibility to adapt the model. If one is interested only in certain genres, he or she can easily select them.

An intuitive user interface is required to avoid losing an overview. We propose the Emergent-Self-Organizing-Map (ESOM) (Ultsch, 2003) to visualize the genre vectors. It is topology preserving and combined with the U-Map it provides a visually appealing user interface and an intuitive way of exploring new content.

2 Related Works

There has been some work on enhancing the user interface based on tags and we will briefly mention some here. Flickr uses “Flickr clusters” which can provide related tags to a popular tag, grouped into clusters. Begelman, Keller, and Smadja (2006) used clustering algorithms to find strongly related tags visualizing them as a graph. Hassan-Montcro and Herrero-Solana (2006) proposed a method for an improved tag cloud and a technique to display these tags with clustering based layout.

The ESOM has already been used successfully to visualize collections of music and photos and on clustering documents. Most of these works have in common that they cluster the data based on features extracted directly from the media. An example is MusicMiner (Mörchen, Ultsch, Nöcker, & Stam, 2005) which uses the timbre distance, a measure based on frequency analysis of audio data. The WEBSOM project (Kaski, Honkela, Lagus, & Kohonen, 1998) is an ESOM based approach in free text mining. Here each document is encoded as a histogram of word categories which are formed by the ESOM algorithm based on the similarities in the contexts of the words. Our approach is different however information is not used that can be extracted from the objects raw data itself but instead user generated content. The works mentioned above show that the ESOM is a powerful tool in visualizing high dimensional data. In Lehvak, Risi, and Ultsch (2007) music was clustered by using tag information and it was shown how an ESOM can help to navigate through the music in an intuitive way.
3 Data

The data that is used in this article are taken from the online radio Last.fm. We used data from 110 weeks, starting 2005 and ending 2007. All together, our statistics are based on more than 6 million songs and more than 75,000 tags. Tags are short symbolic descriptions, e.g., "heavy metal", "favorite song", etc. The users of Last.fm assign tags to songs and browse the content via tags allowing them to only listen to songs tagged in a certain way. The tagcount \( t_{ij} \) is the number of users who assigned a tag \( i \) to a song \( j \). For each week, we have the number how frequently a song was played by users. These numbers are fixed for every week \( i \in W_{all} = \{1, \ldots, 110\} \) in a 6 million dimensional vector called song vector denoted as \( s_i \). An even larger matrix (6 million \( \times \) 75,000) is required to save how often a tag is assigned to a song.

4 Frequential Genre Integration

There are many more tags than genres that can be considered so we have to remove the ones that do not stand for a certain kind of music genre, such as "seen-live", "favourite albums", etc.

The tf–idf (term frequency – inverse document frequency) technique is often used in documents to find words in the text which are able to characterise the document. These words should not appear too often (like the articles the, a, ... ) but should also not be very rare in the text. The tf–idf algorithm gives every word a weight which shows how well they can be used to characterise the text.

Last.fm provides the number of people \( t_{ij} = \text{tagcount}_{ij} \) that have used a specific tag for an song \( j \). The \( t_{ij} \) were scaled to the range of \([0, 1]\). Then the term frequency \( f_{ij} \) can be calculated as

\[
f_{ij} = \frac{t_{ij}}{\sum_k t_{kj}}
\]

with the denominator being the accumulated frequencies of the other tags used for a specific song. The term frequency (in this case a tag frequency) indicates how specific a tag for a certain song is. The inverse document frequency \( \hat{f}_i \) is then defined as follows:

\[
\hat{f}_i = \log \frac{N}{n_i}
\]

with \( N \) being the total number of songs in the collection and \( n_i \) the number of songs that have been assigned to the tag \( i \). The inverse document frequency indicates how specific a tag is. Tags like "favorite song" are assigned to nearly every song. So they are not very useful in describing the character of the songs.
The weights

\[ w_{ij} = f_{ij} \hat{f}_i \]

were used to detect the important tags.

How much impact the genres have on certain weeks can now be calculated by applying the weights matrix \( W = \{w_{ij}\} \) to the weekly song vector \( s_i \).

\[ g_i = W s_i \]

In this way, we receive 110 weekly genre vectors \( g_i \) and have a strong reduce the dimension. The song vectors \( s_i \) have a dimension of over 6 million, in the genre vectors the dimension in cut down to 600. Every component of the vector \( g_i \) describes the impact of a genre in the week \( i \). The genre vectors are not only smaller than the song vectors, they are also more appropriate for analysing music fads because the influence of solitary songs is marginal.

5 Visualisation of Music Fads

An ESOM is an artificial neural network that performs a nonlinear and discontinuous projection which is able to preserve topographic structures such as clusters. The genre vectors are mapped onto a two-dimensional grid of neurons. The grid is toroid to avoid boundary effects. In contrast to the K-mean SOM, the ESOM has significantly more neurons than there are expected clusters.

The unsupervised training process is partly motivated by how visual information is handled in the cerebral cortex of the mammalian brain and equals a regression of an ordered set of model vectors \( m_i \in \mathbb{R}^n \) into the space of observation vectors \( x \in \mathbb{R}^n \) by performing the following process:

\[ m_i(t + 1) = m_i(t) + \delta_{c(t), i} (x(t) - m_i(t)). \]

where \( t \) is the sample index of the regression step, whereby the regression is performed recursively for each presentation of a sample of \( x \). Index \( c \), the best matching unit (BMU) or winner, is defined by the condition

\[ \| x(t) - m_c(t) \| \leq \| x(t) - m_i(t) \| \quad \forall i. \]

The neighbourhood function \( h \) is the Gaussian

\[ h_{c(t), i} = \alpha(t) \exp \left( -\frac{\| r_i - r_c \|^2}{2\sigma^2(t)} \right), \]

where \( 0 < \alpha(t) < 1 \) is the learning-rate factor, which decreases monotonically with the regression steps, \( r_i \) and \( r_c \) are the vectorial locations in the display grid and \( \sigma(t) \) corresponds to the width of the neighbourhood function, which is also decreasing.
monotonically with the regression steps. For a more detailed discussion of the SOM see Kaski et al. (1998).

The U-Map (Ultsch, 2003) is constructed on top of the map of ESOM. The U-Height for each neuron \( n_i \) equals the accumulated distances of \( n_i \) to its immediate neighbors \( N(i) \):

\[
U_{\text{Height}}(n_i) = \sum_{j \in N(i)} d(m_i, m_j),
\]

where \( d(x, y) \) is the distance function used in the SOM algorithm to construct the map and \( N(i) \) denotes the indices of the immediate neighbours of neuron \( i \).

A single U-Height shows the local distance structure of the corresponding neuron. The overall structure of distance emerges, if a global view of a U-Map is regarded. A U-Map is usually displayed as a three-dimensional landscape and has become a standard tool to display the distance structures of the ESOM. The U-Map delivers a "landscape" of the distance relationships of the input data in the data space. It has the property that weight vectors of neurons with large U-Heights are very distant from other vectors in the data space and that weight vectors of neurons with small U-Heights are surrounded by other vectors in the data space. Outliers and other possible cluster structures can easily be recognized. U-Maps have been used in a number of applications to detect new and meaningful information in data sets.

Figure 1 shows a toroid ESOM with 82 \( \times \) 50 neurons that was trained with the 110 weekly genres vector using the Databionics ESOM Tools. The left picture shows a line drawn form one BMU to the next. The right picture shows the resulting U-Map.

To get a plain island map with a unique representation for every neuron we have to cut the toroid map along the highest hills.

6 Identification of Fads

In the last chapter an U-Map with a unique representation for the genre charts was created. On this map, it can be seen that the genre vectors follow a great valley, but, from time to time they "jump" over a small hill.
Fig. 2 The left picture shows the Euclidian distances of the genre vectors from one week to the next. On the right the logarithmic distribution of the distances is shown.

Obviously, the genre vectors are not randomly distributed. Chronologically neighbouring vectors are represented nearby on the U-map. This shows the strong conjunction between the time and the genre vectors. On the other hand there are some relative highly hills between some genre vectors and their followers. So there is a steady low movement for some weeks but then the genre vector shifts strongly from one week to the next. The time intervals with the low moments of the genre vectors are called music fads.

To find the music fads means to find those gaps in the data. In Fig. 2 the Euclidian distances of the genre vectors between one week and the following week are plotted. Some extreme values show that the genre vectors do not move continuously. They show, when a “jump” happens and when a new fad begins.

There are 110 weeks and so there are 109 distances between them. The distribution of the log transformed distances is also shown in Fig. 2 (on the right side).

This distribution can be approximated by a mixture model of three normal distributions. The two largest stand for normal weekly distances, the third contains the “jumps”. To find a boundary value to decide how large a weekly distance must be if a new fad has started the two large gaussian were separated from the small one by a Bayes decision. The boundary that was found this way is about 1.15. According to this we have six music fads.

7 Fads Characterisation

To make the music fads into useful information it is necessary to characterise the fads. For this purpose we calculate the fad genre vector $F_k$ as the average over the weekly genre vectors in a fad

$$F_k = \frac{\sum_{i \in W_k} g_i}{|W_k|}.$$
where \( W_k \) are the weeks of the fad number \( k \) and \( g_i \) are the genre vectors of the week \( i \). We compare this cluster genre chart to the average of the genre vectors of the whole time period

\[
D_k = F_k - \frac{\sum_{i \in W_{all}} g_i}{|W_{all}|},
\]

where \( W_{all} \) is the set of all 110 weeks.

\( D_k \) is a vector which contains the displacement of the impact a genre has during a fad from its average impact. This gives us a metric ranking of the importance of the genres.

8 Results

As a result of this analysis 6 music fads have been found. The music fads vary strongly in their length. While the shortest last only for 2 weeks, the longest fad lasted for 50 weeks. The changes in the music style are easily found and the fads can be characterised by the dominant music genres. For each music fad there is a displacement vector which shows the impact of a genre for this fad. Those genres which have the most influence on the fad have the highest numbers. You can see the most important genres assigned to their music fads in Fig. 3.

9 Discussion

It could be objected that online radio users are not a representative sample of the population. That is certainly true. Though, it is an interesting community for music research. A music manager usually is not interested in the whole market but only in his field of activity. It is easy to adapt this method for personal interests by selecting a special set of genres.

This method does not make predictions of music fads, but it is possible to recognise a new fad very quickly. Of course it would be better to identify music fads before they started, but that is hardly possible.

The tf–idf technique shows the impact of a genre. It does not state how many genres are optimal for the analysis. In this paper we took 600 genres which produced good results. Of course, the selection of genres also depends on the aim of the analysis. Though the tf–idf technique is a useful tool, the genre selection requires external knowledge and experience.

10 Summary

Visualization of music fads using user-generated tags was demonstrated to work well. We use the Emergent-Self-Organizing-Map (ESOM) to visualize the genre vectors. It is topology preserving and combined with topographical maps it provides a visually appealing user interface and an intuitive way to understand fashions in the complex music market. The temporal fashion development is shown
Fig. 3 The characteristic genres for the music fads are (1) 90s, stoner rock, industrial; (2) hip-hop, britpop, rap; (3) German, industrial metal, industrial; (4) metal, indie rock, indie; (5) progressive metal, progressive rock, metal; (6) funk, progressive metal, funk rock.

as a path on the U-map in valleys surrounded by mountains. In these valleys there exist hills that separate the music fads. There is an easy way to find the characteristics of a fad by comparing the average genre vector with the fad genre vector. This way, we get the genres that have the highest impact of the fad. This information is important for every music manager who decides when to place a song on the market.

References


