Cold-Start Music Recommendation Using a Hybrid Representation

Ben Horsburgh
IDEAS Research Institute
Robert Gordon University
Aberdeen, Scotland
b.horsburgh@rgu.ac.uk

Susan Craw
IDEAS Research Institute
Robert Gordon University
Aberdeen, Scotland
s.craw@rgu.ac.uk

Stewart Massie
IDEAS Research Institute
Robert Gordon University
Aberdeen, Scotland
s.massie@rgu.ac.uk

1. INTRODUCTION

Digital music systems are a new and exciting way to discover, share, and listen to new music. Their success is so great, that digital downloads are now included alongside traditional record sales in many official music charts [10]. In the past listeners would rely on magazine, radio, and friends reviews to decide on the music they listen to and purchase. In the internet age, this style of finding music is being superseded by music recommender systems.

The shift from listening to hard copies of music, such as CDs, to online copies like MP3s, presents the interesting new challenge of how to recommend music to a listener. In such recommender systems, a user will typically provide a track that they like as a query, often implicitly as they listen to the track. The system must then provide a list of further tracks that the user will want to listen to. Many websites exist that provide such recommender systems, and many of the systems provide very good recommendations. However, there are still scenarios that these systems struggle to handle, and where recommendations can be unreliable.

Online music systems allow users to tag any track with a free-text description. A recommender system can then determine the similarity between tracks based on these tags, and make recommendations. However, when a track is new to the system it will have no tags. This means that the track is never recommended, and in turn, the track is very unlikely to be tagged. Turnbull et. al [11] show that social tags tend to be very sparse, and that a huge popularity bias exists. This is further confirmed by data released by Last.fm [7] as part of the million song dataset [3]: from a vocabulary of over 500000 tags, each track, on average, has only 17 tags; 46% of tracks have no tags at all.

This scenario is often referred to as the cold-start problem; the results of which means large volumes of music are excluded from recommendations, even if they may be an excellent recommendation. The aim of our hybrid representation is to reduce the effects of the cold-start problem, therefore increasing the recommendation quality of the overall system.

Digital Economy All Hands Conference '12 Aberdeen, UK
.

2. RELATED WORK

Effects of cold-start can be reduced by learning tags for a track, a task referred to as auto-tagging [2, 9]. Most auto-tagging systems first extract audio features for many short time-frames within a track, and then model this bag-of-frames. A growing trend has been to use Gaussian Mixture Models [1], which describe the important clusters within a tracks' bag-of-frames. Tags are modelled in a similar manner, and correlations between the audio and tag models are used to learn auto-tags.

Hybrid representations are also becoming a prominent way of addressing the cold-start problem [8, 12]. In previous work we defined a representation that incorporates both audio and tag features, and generalises the representation using latent-semantic analysis (LSA) [5]. In the LSA concept space, tracks without tags belonged to concepts with tags. This approach increased the inclusion of cold-start tracks in recommendations.

Hybrid methods are good for cold-start, but they are often harmful when a track is well-tagged. The reason for this is the direct inclusion of content in the representation; content is a much weaker representation than tags.

3. HYBRID TAG REPRESENTATION

We propose that content should be used to strengthen other representations, but should not be integrated into other representations. Our new hybrid representation therefore does not include the content representation directly, but instead uses it to learn a stronger representation when appropriate.

To construct our hybrid representation we first define a content-based recommender system. This is then used to learn a pseudo-tag representation, which must achieve better recommendation quality than the content-based method. Our hybrid representation is a dynamic combination of both tags and pseudo-tags. In this section we will describe each representation.

Content. A content representation for a track is created by extracting our music-inspired texture representation Mel-Frequency Spectrum (MFS) [4] from the track audio. For each track, we first extract one MFS texture-vector for every 186ms of audio, and then compute a mean texture-vector. Finally, LSA is used to generalise all of the mean texture-vectors in our collection. Using the generalised MFS vectors, we are now able to provide content-based recommendations by computing the Euclidean distance between tracks' representations; in our evaluation we will refer to this method as **content**.

Pseudo-tags. Recommendation using the content method provides a list of tracks with similar content to a query track. To generate pseudo-tags, we take advantage of this list of similar tracks, and any tags which represent them. Given a list of N nearest neighbours of a track, we compute a pseudo-tag vector p as a weighted sum of the nearest neighbours tag-vector weights,

$$p_i = \sum_{n=1}^{N} \left(1 - \frac{n-1}{N}\right) t_i \tag{1}$$

where n is the position of the track in the nearest neighbour retrieval, and N is the number of nearest neighbours retrieved. t_i is the term-frequency of the ith tag in the tag vector of the track at position n. The bracketed part of the equation acts as a weighting function, giving a higher weight to tracks that are most similar. Recommendations made using these pseudo-tag vectors will be referred to as the **pseudo-tag** method.

In our final recommendation approach, we do not want to use only pseudo-tags, since these ignore tags already present in the collection. We therefore integrate the pseudo-tags we learn with the existing tag-vector of each track. However, we also do not want to decrease the effectiveness of existing track representations that do not suffer from cold-start. When our recommender system is using a tag-only representation, we will refer to this as the **tag** method.

Hybrid. Effective use of pseudo-tags can be achieved by creating a tag + pseudo-tag hybrid, which balances the influence of both tags and pseudo-tags. We therefore determine the number and weighting of pseudo-tags that we use based on how well-tagged a track is. If a track suffers from cold-start, many pseudo-tags will be used, and they will be given a high weighting. Conversely, if a track is well-tagged, very few pseudo-tags will be used, and they will be given a low weighting. Our final hybrid representation h is computed as

$$h_i = \alpha p_i + \beta t_i \tag{2}$$

where α is the weight of pseudo-tag vector p, and β is the weight of tag-vector t. i is the index of the tag being computed. When limiting the number of pseudo-tags being used, we set the weights of unwanted pseudo-tags to 0. The exact thresholds and weightings that we used were learned from a training set in our collection. We will refer to our hybrid representation as **hybrid**.

4. EVALUATION

The objective of our evaluation is to measure the performance of hybrid recommendation, and to examine the effect our representation has on cold-start tracks. A single track is provided as a query, and we measure the quality of recommendations made by our recommender system.

The dataset we are using contains 3174 tracks by 764 artists, split across 12 distinct super-genres. We collect the tag representations of each track using the Last.fm Audioscrobbler API [6]. This API provides a normalised tag-frequency vector, where the most popular tag for any given track has a weight of 100. The average number of tags assigned to a track in our collection is 34, with a standard deviation of 24.4. The maximum number of tags a track has is 100. Cold-start is also present in our collection, with 25% of tracks having fewer than 10 tags, and 3% of tracks having no tags.

We use 10-fold cross validation, and each track occurs in exactly one fold. For each query, our recommender system provides 10 recommendations. To measure the quality of these recommendations we use the association score measure, which we developed in previous work [5]. The data used to calculate the association score was collected for $175,000$ Last.fm users over a period of 2 months. This score estimates the relationship between tracks based on the proportion of listeners who liked both tracks, computed as

$$\text{association}(s_i, s_j) = \frac{\text{likes}(s_i, s_j)}{\text{listeners}(s_i, s_j)} \tag{3}$$

where s_i and s_j are the query and recommended tracks. $\text{listeners}(s_i, s_j)$ is the number of people who have listened to both track s_i and track s_j, and $\text{likes}(s_i, s_j)$ is the number of people who have liked both s_i and s_j. A score of 0 indicates no evidence of users liking a recommendation, and a score of 1 indicates that all listeners of the query and recommendation also liked both the query and the recommendation, that is, the recommendation is of a high quality.

5. RESULTS

We first evaluate the performance of our pseudo-tag representation, to validate that it is meaningful. Figure 1 compares the recommendation quality of pseudo-tags and content. The horizontal axis shows the number of recommendations made, and the vertical axis shows the association score. The values shown are the mean average association score at the given number of recommendations, and error bars show 95% confidence.

Content-based recommendation is illustrated by the grey dashed line, and pseudo-tags by the black dashed line. When 1 recommendation is made, both methods are comparable; when 2 or more recommendations are made, pseudo-tags generate significantly better recommendations than content. While pseudo-tags are learned using content, they do not necessarily always describe content, and so tags, which describe other features, will also be present.

Figure 2 shows recommendation quality using tags and our hybrid representation, in the same format as the previous figure. For all numbers of recommendation, the hybrid-based approach offers significantly better recommendations than the tag-based method, and much better than either content or pseudo-tags alone. Increased quality is achieved because cold-start tracks now have a meaningful representation, and non-cold start tracks are still allowed to use the full descriptive power of their tag representation.

6. CONCLUSIONS

The motivation for this work was to increase the performance of a music recommender system by reducing the effects of the cold-start problem. We have shown that recommendations made using our pseudo-tag representation are of higher quality than when a state-of-the-art content-based representation is used. But, integrating these pseudo-tags with tags, to create our hybrid representation, provides significantly better quality recommendations than when using a tag-based representation. In future work we aim to examine in closer detail the effects our hybrid representation has on cold-start tracks.

7. REFERENCES

[1] L. Barrington, D. Turnbull, and G. Lanckriet. Auto-tagging music content with semantic multinomials. In *Proc. of Int. Conference on Music Information Retrieval*, 2008.

[2] T. Bertin-Mahieux, D. Eck, and M. Mandel. Automatic tagging of audio: The state-of-the-art. *Machine Audition: Principles, Algorithms and Systems. IGI Publishing*, 2010.

[3] T. Bertin-Mahieux, D. Ellis, B. Whitman, and P. Lamere. The million song dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*, 2011.

[4] B. Horsburgh, S. Craw, and S. Massie. Music-inspired texture representation. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 52–58, 2012.

[5] B. Horsburgh, S. Craw, S. Massie, and R. Boswell. Finding the hidden gems: Recommending untagged music. In *Twenty-Second International Joint Conference on Artificial Intelligence*, pages 2256–2261, 2011.

[6] Last.fm API. http://www.last.fm/api, May 2012.

[7] Last.fm dataset, the official song tags and song similarity collection for the Million Song Dataset. http://labrosa.ee.columbia.edu/millionsong/lastfm, may 2012.

[8] M. Levy and M. Sandler. Music information retrieval using social tags and audio. *IEEE Transaction on Multimedia*, 11(3):383–395, 2009.

[9] R. Miotto, L. Barrington, and G. Lanckriet. Improving auto-tagging by modeling semantic co-occurrences. In *Proc. ISMIR*, pages 297–302. 2010.

[10] Official Charts Company. http://www.officialcharts.com/chart-rules/, 2009.

[11] D. Turnbull, L. Barrington, and G. Lanckriet. Five approaches to collecting tags for music. In *International Society for Music Information Retrieval 2008*, pages 225–230, 2008.

[12] K. Yoshii and M. Goto. Continuous plsi and smoothing techniques for hybrid music recommendation. In *Proceedings of the 10th International Society for Music Information Retrieval Conference*, 2009.

APPENDIX

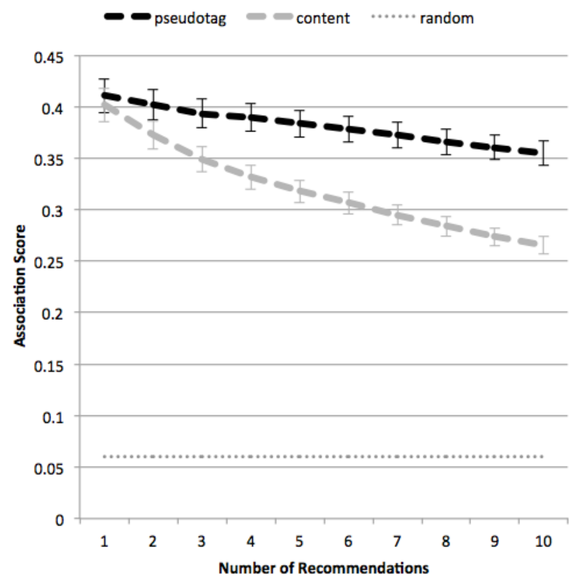

Figure 1: Pseudo-tag vs. Content Quality

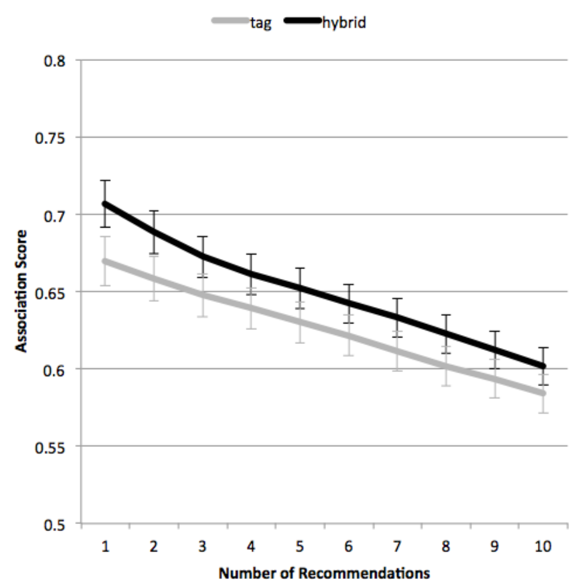

Figure 2: Hybrid vs. Tag Quality