

Literature Review

Joshua Bridge

14032908

joshua.m.bridge@stu.mmu.ac.uk

November 23, 2017

Contents

0.1	Introduction	2
1	Recommendation systems	2
1.1	Collaborative recommendation	3
1.2	Content-based recommendation	3
1.3	Knowledge-based recommendation	4
1.4	Hybrid recommendation	4
1.4.1	Spotify - Discover Weekly	4
1.5	Machine learning recommendation	5
1.6	Image classification in recommendations	5
2	Image-filter generation	5
3	Web-based user interface	5
4	Conclusion	5

List of Figures

1	System component architecture of Spotify's 'Discover Weekly'	5
---	--	---

0.1 Introduction

In order to understand the background of the technology involved in this project, it is necessary to complete a significant review into the current and past literature. This will help form a basis of knowledge from which the future development and analysis of the proposed system will utilise and build upon. This research will be vital in order to make use of the most optimal technology for any given problem.

The following review will be an investigation into the current knowledge of the three major components of the proposed system, which are as follows:

- The recommendation system.
- The image-filter generation system.
- The web-based user interface.

1 Recommendation systems

As defined in Ricci et al. (2011), recommending content is a problem which involves attempting to predict what a user may desire at any given time when using a system. In the ‘internet era’ applications of this are far-reaching, such as recommending products on Amazon.com Linden et al. (2003). Recommendation systems are a popular topic due to the “abundance of practical applications that help users to deal with information overload and provide personalized recommendations, content, and services to them” (Adomavicius & Tuzhilin 2005).

Jannach et al. (2010) describe three different methods for giving recommendations: Collaborative, Collaborative, Content-based, Knowledge-based, and Hybrid.

1.1 Collaborative recommendation

When using an online streaming platform that serves video content such as Netflix or YouTube, a user will visit their site looking for something to watch. The problem faced by these sites is trying to find content that the user hasn't seen before, but will align with the user's interests enough to make them want to watch it. The information needed to carry out such a prediction can include the users history of interaction with the system, from which preference can be extrapolated (a.k.a. 'metadata'). If there is not enough metadata about the user from which to extrapolate preference with reasonable certainty, then preference can be inferred from other users of the system - especially users with a similar predicted preference.

1.2 Content-based recommendation

There can sometimes be barriers to using collaborative recommendation, such as when the 'cold-start problem' is present. This problem describes when a user has just signed up to a website, it will not yet have enough information / user history to recommend them content. In this situation content-based recommendation could be used more effectively to build up some more reliable recommendations. There are different kinds of content-based recommendation, as it can be applied to many different kinds of content. For example

a system may work differently if the content is user-generated as opposed to internally or computer generated. In a system with user-generated content, the information about that content must be inferred, or provided by the user.

1.3 Knowledge-based recommendation

1.4 Hybrid recommendation

1.4.1 Spotify - Discover Weekly

Discover Weekly is a playlist made every week for every user of Spotify. Instead of getting humans to curate a playlist for each and every user, an algorithm is used to try and find a set of songs which it predicts the user will like, but hasn't listened to before. As described by Popper (2015) it does this by mixing collaborative and content-based recommendation. The first is done by analysing every playlist ever made on spotify and determining their similarity to the playlists made by the user. The next step is looking for songs in those playlists that the user in question hasn't heard yet.

The second, more complicated part of this process is the content-based recommendation. Spotify's algorithms actually analyse the music data in several different ways to determine a users 'taste'. The first is by defining the genres a user listens to, which it does by scanning music blogs for what genre a certain piece of music is described as being. Using this as a search, spotify can then look for other music in that genre to recommend. The arguably more complicated part of this process is the auditory analysis of raw music data. As shown in figure 1 the raw audio is converted into 'batch audio models' where they can be used as a basis for recommendations.

Discover Weekly Data Flow

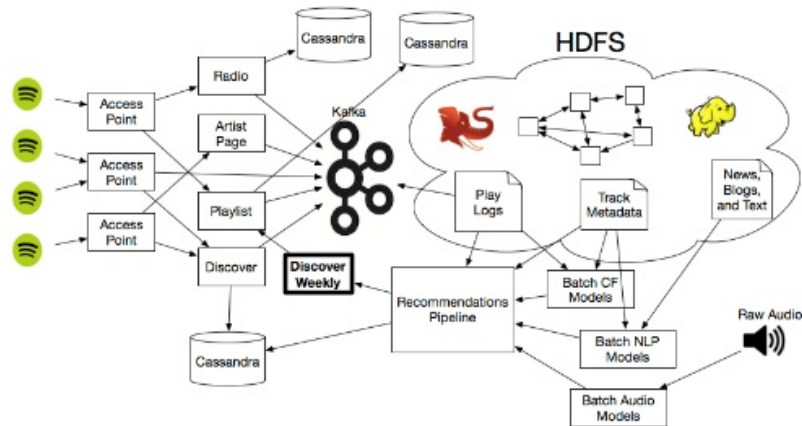


Figure 1: System component architecture of Spotify’s ‘Discover Weekly’ (Johnson 2015)

1.5 Machine learning recommendation

1.6 Image classification in recommendations

2 Image-filter generation

3 Web-based user interface

4 Conclusion

References

- Adomavicius, G. & Tuzhilin, A. (2005), ‘Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions’, *IEEE transactions on knowledge and data engineering* **17**(6), 734–749.
- Jannach, D., Zanker, M., Felfernig, A. & Friedrich, G. (2010), *Recommender systems: an introduction*, Cambridge University Press.
- Johnson, C. (2015), ‘From idea to execution: Spotify’s discover weekly’.
URL: <https://www.slideshare.net/MrChrisJohnson/from-idea-to-execution-spotifys-discover-weekly/>
- Linden, G., Smith, B. & York, J. (2003), ‘Amazon. com recommendations: Item-to-item collaborative filtering’, *IEEE Internet computing* **7**(1), 76–80.
- Popper, B. (2015), ‘Tastemaker: How spotify’s discover weekly cracked human curation at internet scale’.
URL: <https://www.theverge.com/2015/9/30/9416579/spotify-discover-weekly-online-music-curation-interview>
- Ricci, F., Rokach, L. & Shapira, B. (2011), Introduction to recommender systems handbook, *in* ‘Recommender systems handbook’, Springer, pp. 1–35.