



Literature Review submitted to
The Department of Computer Engineering of Çankaya University
in fulfillment of the requirements for
CENG 407 Innovative System Design and Development I

LITERATURE REVIEW FOR A PLATFORM FOR MUSIC, BOOK AND MOVIE RECOMMENDATION

By

Ege Varolgüneş	201611062
Enzel Ege Sarı	201612057
Yaren İpek	201711038
Pelinsu Serimer	201711058

Group No: 07

Supervisor: Assist. Prof. Abdül Kadir Görür

Çankaya University
Department of Computer Engineering
Fall 2020

STATEMENT OF NON-PLAGIARISM

I hereby declare that all information in this assignment has been obtained and presented in accordance with academic rules and ethical conduct and the work I am submitting in this document, except where I have indicated, is my own work.

Date:

Group Members' Name and Surname

Signatures

Ege Varolgüneş

Enzel Ege Sarı

Yaren İpek

Pelinsu Serimer

TABLE OF CONTENTS

ABSTRACT	4
1. INTRODUCTION	4
2. MAIN FINDINGS	5
2.1 Recommender Systems	5
2.2 Web Development	6
2.3 Mobile Development	7
2.4 Deep Learning	8
3. DECISION	10
4. CONCLUSION	11
REFERENCES	12

ABSTRACT

Several modern recommender systems are developed to help users to devote more time to themselves by proposing relevant suggestions in the requested context. This project is designed to improve present recommender systems, by bringing multiple recommender systems which focus on different areas together in one platform. Conducting a literature review provides an opportunity to learn how to design an innovative system and understand the challenges along the way. This literature review aims to find the best methodologies to create a well-functioning recommendation platform for music, books, and movies. Hence, the previous works regarding this field are reviewed. The findings of this literature review, the decisions made upon this research, and the challenges that might be faced further in the development stages are acknowledged and presented in this report.

KEYWORDS: Recommender System, Software Development, Deep Learning.

1. INTRODUCTION

The popularization of different types of web services leads the recommender system concept to become popular and varied. The concept of a recommender system aims to suggest relevant items to users. Various item examples can be given depending on industries. (Rocca, 2019) The scope of this project is to develop a platform that involves multiple recommender systems with various suggestion items. The platform will focus on recommendations of the following items: movies to watch, books to read, music to listen to. Diverseness of the items will not only offer variety to the options proposed to the users but also reduce the item search period required by the users to find the right option. The platform developed will be more flexible for the user's decision changes. In addition to these advantages, it is expected that the platform will be more appealing to users having different requests in context. Within the scope of this work, in the Main Findings (Chapter 2) section, firstly we review the recommender system concept (2.1). And, since we plan to develop the platform both as a web and a mobile platform, development methodologies will be examined focusing on web development (2.2) and mobile development (2.3) concepts consecutively. Then, the neural networks (2.4) concept will be discussed. In the last section of the paper, references are specified.

2. MAIN FINDINGS

2.1 Recommender Systems

The recommender systems are more valuable and important attributes to our daily lives. The rise of services such as Netflix, Spotify, Soundwave or even shopping pages like Amazon or Trendyol made the customized recommendations more and more popular. Recommender systems help companies and customers tremendously. Recommender systems are algorithms that makes suggestions to the users according to their preferences. (Rocca, 2019)

The recommender systems use special algorithms to figure out what the customer likes and provides relevant suggestions. Few real-life examples for such systems are Soundwave, GoodReads, and MovieLens. Soundwave makes suggestions based on the location of the user and gives the music trends that the user is currently at. (O'Hear, 2013) GoodReads gives out recommendations according to genres of books the user enjoyed previously. (Strickland, 2009) MovieLens gives the user lots of movie options to rate and from the user's ratings it understands the user's behavior and recommends the user other movies they will enjoy watching.

These recommender systems use different approaches such as, a collaborative filtering-based system, content-based system, hybrid system, and Popularity based system.

Collaborative Filtering (CF) based methods uses the ratings given by the users as the main source of the behaviors to make a recommendation. One downside is that the rating may be sparse which causes the CF based methods to decrease their performance. The CF based methods gather information on different users and make a well calculated recommendation with the other user's behaviors included. (Wang, Wang, & Yeung, 2015)

Content-Based Filtering method uses the previous behaviors of the user such as purchases or consumptions and makes recommendations based on those historical data. Content-based filtering methods match up the attributes of user behavior, in other word users interests, with the attributes of the items and the system makes the recommendations regarding these comparisons. (Lops, de Gemmis, & Semeraro, 2010)

Hybrid methods usually combine the content-based methods with collaboration-based methods to come up with a recommendation for the user. This method gives much more accurate recommendations and is currently used by most industries. There are two approaches one way is to create two different recommender systems and combine the suggestions or combine the methods into one single recommender system. (Rocca, 2019)

2.2 Web Development

Web development is the task of developing and maintaining websites for the Internet or an intranet. Since developing a website for this project is one of the main goals, research is conducted for some necessary terms and concepts regarding web development. This section of the literature review has related research that helped us to choose the related architecture and protocol.

IBM defines the web services as a generic term for an interoperable machine-to-machine software function that is hosted at a network addressable location. (IBM, 2020) When creating a web service there is a need for an architecture and a protocol. An architecture describes how the databases, applications and middleware systems work with each other. (Banga, 2020) A protocol is an agreement on how the web services will be exchanging data.

Service-Oriented Architecture (SOA) is an architectural style that uses services of the network such as the web for building software applications. Applications in SOA, as the name suggests, are based on services. SOA enables the users to use the existing assets of the already existing IT infrastructures. (Mahmoud, 2005) (Bean, 2009) OA has 6 main components. These are service, policies, endpoints, contracts, messages, and service consumers. Services are the most important component of the SOA. Each service offers different functionality. Policies define the restrictions and terms such as security, auditing, etc. An endpoint is a URL that identifies the location on the built-in HTTP service where the web services listener listens for incoming requests. (IBM) Messages are the communication unit of the SOA. There are various kinds of Messages such as REST, SOAP, JMS. Consumers communicate with services by messaging. A consumer can be another application or a service.

RedHat defines API (Application Programming Interface) as a set of definitions and protocols for building and integrating application software. (Red Hat) The main purpose of an API is to let an application's methods to other applications and respond to data requests easily and quickly. It provides saving time and money. APIs give us flexibility and provide opportunities for innovation.

REST (Representational State Transfer) is an architectural style that provides fast client-server communication. It is not a standard, so it supports different data formats such as JSON, XML, HTML, etc. REST is stateless; it provides scalability and visibility. It is cacheable. REST has a layered system because of the client-server architecture. REST lets developers to use HTTP methods such as GET, POST, HEAD, PUT, DELETE, and PATCH.

SOAP is a protocol for the exchange of information in a distributed environment. (IBM, 2020) SOAP is an XML based messaging protocol. It only supports XML format and these messages are human-readable. For this reason, it requires more bandwidth and computation power. SOAP can ride on different transfer protocols such as HTTP, SMTP, UDP, etc.

2.3 Mobile Development

Mobile development is the process of creating software applications that run on a mobile device. (Amazon) This section of the literature review focuses on mobile development strategies. These strategies are native development, hybrid development, cross-platform development, and progressive web applications approach.

Native development aims to develop software applications for use on a particular target platform or device. Major mobile development platforms are iOS by Apple, Android by Google, and others, such as Microsoft Windows phones or RIM Blackberry. (Latif, Lakhrissi, Nfaoui, & Es-Sbai, 2016). In this approach of mobile development, native mobile developers choose the platform to be worked on by deciding how they link the application with the underlying operating system. Native mobile developers use a specific Software Development Kit (SDK) and programming language for each target platform. For example, to develop a native platform for the Android operating system, developers use Android Studio for SDK, and Java as a programming language, while another developer uses XCode for SDK, and Objective-C/Swift as a programming language to develop a native platform for iOS.

Hybrid development is the blend solution that consists of both native and web approaches. Hybrid platform developers embed the code written in HTML, CSS, and JavaScript into a native application. (Kidecha, 2020)

Cross-platform development aims to develop software applications for use on different target platforms or devices. Cross-platform approach has been motivated by the difficulty that occurred in the native approach. The development phase is easier and more efficient in the cross-platform approach, compared to the native approach. Because in the cross-platform approach, developers create a code base, then only make small changes on the code, if it is required by the platform. They do not need to use different SDK's, platforms or programming languages for multiple platform development. (Latif, Lakhrissi, Nfaoui, & Es-Sbai, 2016)

There is another option called “progressive web applications”. This approach consists of browser-based HTML, CSS, and JavaScript codes and is accessed over the Internet. (Summerfield, -) Since the development of a web application is one of the goals of the project, the development of a progressive web application would be redundant. Hence, this approach will not be reviewed any further.

2.4 Deep Learning

While implementing a recommender system, usage of deep learning algorithms are more common to use simply because deep learning can efficiently learn the underlying behaviors of the input data and give related outputs. Deep learning has a high level of flexibility and is really powerful for sequentially modeling tasks or decisions. Also, Deep learning is a really good option to model the interactions of the non-linear data in which a person's behaviors can be explained by non-linear equations most of the time. (Le, 2019)

Through in-depth research on the techniques of deep learning that can be used in this project 5 suitable techniques are found. Multi-Layer Perceptron based recommendation, autoencoder based recommendation, Convolutional Neural Networks Based Recommendation, Recurrent Neural Networks Based Recommendation, and Restricted Boltzmann Machines Based Recommendation are techniques that are going to be discussed in this section.

Multi-Layer Perceptron (MLP) based recommendation method is a multiple hidden layer feedforward neural network. The hidden layers are placed between the input layer and the output layer. MLP is an effective and concise network model that can make an approximation

of any measurable function to any needed accuracy. The feature representation of MLP is straightforward even if it is not expressive as the other recommendation methods.

Autoencoder based recommendation method is one of the unsupervised models. Autoencoder attempts to reconstruct input data in the output layer. For the important feature representation of the input data, the bottleneck layer is used. autoencoders used to learn the lower-dimensional feature representations at the bottleneck layer.

Convolutional Neural Networks (CNN) based recommendation is also a feed-forward neural network like the MLPs but there are convolution layers and pooling operations included. CNN can capture global features as well as the local features. This increases the model's efficiency and accuracy remarkably. CNN's mostly used to extract features from texts, audios, videos and images. The usage of CNN is more common than the other methods. (Ying , et al., 2018)

Recurrent Neural Networks (RNN) based recommendation method is suitable for sequential data modeling. It has memory and loop attributes to remember the previous computations. Therefore, they can be used for temporal dynamics of interactions and sequential patterns of user behaviors. But this method can be used for session-based recommendations most of the time. (Hidasi, Baltrunas, Karatzoglu, & Tikk, 2015)

Restricted Boltzmann Machines (RBM) based recommendation method is a two-layered neural network model. It has one visible and one hidden layer. It can easily be loaded to a deep network. The visible layer of the RBM consists of binary values, therefore the ratings are shown in a vector to fulfill this restriction. In a recommendation system, each user has a unique RBM with a shared parameter. This means that each user will tell what they want and rate those needs and the RBM will gather the behavior information. (Ali, 2019)

3. DECISION

Deciding upon the methodologies that will be used in the development phase of the project has a great value. The correct decision of the methodologies will lead to a correctly working recommendation platform, hence a correctly completed project.

The first decision is to decide on the recommender system algorithms from the options that are mentioned in the previous section. Each algorithm has its perks and drawbacks but using a content-based filtering system will benefit the platform better than the other options since conducting a questionnaire to retrieve the customers behaviors and then giving out the recommendations from the outcome will be much faster in receiving the recommendations from the platform.

Deciding on the web services were challenging because the perks of each method were too many. In the end the decision is made upon the usage of REST API. An API enables us to use the full functionality of another application while the web services only allow us to use specific tasks. Therefore, APIs are a better option. REST APIs use fewer resources and support various formats of messages. For these reasons, the development of web services will be done by using REST API.

To decide which mobile development approach to be used, options that are mentioned in the previous section are evaluated. Since native development for each platform is a costly and time-consuming process, most popular mobile development companies use the cross-platform approach. They avoid rewriting the code for different platforms. Even the hybrid approach has the same code-shareability as the cross-platform approach, it is mostly used when user experience and performance is not a priority for the application, hence it will not be a great option to consider in our project. However, the native approach is more appropriate in our case. In the scope of mobile development, our platform has been planned as only an Android application to limit the scope of the project and reduce the cost of training. Also, it has been seen that the native approach has richer libraries, higher performance, and higher user experience.

Decision of the deep learning technique to be used is more crucial than the decisions of the other components of the project. Hence, the decision for the deep learning method needed more thinking and trial and error to make sure of the performance. Even though the decision upon the technique for deep learning is to be decided, some of the techniques are eliminated down to two possible techniques. Further in the development phase the decision for the algorithm will be made between CNN and RBM methods.

4. CONCLUSION

Since the recommender systems are in demand and gained importance through the industries, recommender system's performance become also important. Currently the users can get recommendations for books, movies and music from different platforms which makes it hard for users to get fast and reliable suggestions. By combining these three components of recommendations the time spent to achieve recommendations will be deducted significantly and by getting the recommendation from the same type of algorithm will make better predictions for the users even more than getting those recommendations from three different platforms and three different algorithms. It has been decided that the implementation of the project will be a web application and a mobile application. For web development there were 2 main possibilities of web services rest apis and SOAs. rest apis come up to be a better solution with both the amount of resources and also the functionality. Since there is development of a web application as one of the goals of the project, developing an addition of mobile application would improve the outcomes of the project further. For mobile development the decision is made on using the native development approach since this approach gives higher performance. The decision for deep learning algorithms has not been made due to the fact that there is more than one good option for developing a recommender system. The performance of both of the options can only be decided upon the development phase but all the possibilities are limited to two options being CNN method and RBM method.

REFERENCES

- Ali, A. (2019, May 26). *Restricted Boltzmann Machine (RBM) with Practical Implementation*. Retrieved from Medium: <https://medium.com/machine-learning-researcher/boltzmann-machine-c2ce76d94da5>
- Amazon. (n.d.). *What is Mobile Application Development*. Retrieved from AWS: <https://aws.amazon.com/tr/mobile/mobile-application-development/>
- Banga, S. (2020, June 16). *What is Web Application Architecture? Components, Models and Types*. Retrieved from Hackr.io: <https://hackr.io/blog/web-application-architecture-definition-models-types-and-more>
- Bean, J. (2009). *SOA and Web Services Interface Design: Principles, Techniques and Standards*. Morgan Kaufmann.
- Hidasi, B., Baltrunas, L., Karatzoglu, A., & Tikk, D. (2015). SESSION-BASED RECOMMENDATIONS WITH RECURRENT NEURAL NETWORKS. *ICLR 2016*.
- IBM. (2020, July 16). *SOAP*. Retrieved from IBM: https://www.ibm.com/support/knowledgecenter/SSGMCP_5.2.0/com.ibm.cics.ts.webservices.doc/concepts/soap/dfhws_overview.html
- IBM. (2020, July 16). *What is a Web Service?* Retrieved from IBM: https://www.ibm.com/support/knowledgecenter/SSGMCP_5.2.0/com.ibm.cics.ts.webservices.doc/concepts/dfhws_definition.html
- IBM. (n.d.). *SOAP Endpoint*. Retrieved from IBM: https://www.ibm.com/support/knowledgecenter/SSSHYH_7.1.0/com.ibm.netcoolimpact.doc/dsa/imdsa_web_netcool_impact_soap_endpoint_c.html
- Kidecha, S. (2020, May 27). *Native vs. Hybrid vs. Cross-Platform: How and What to Choose?* Retrieved from DZone: <https://dzone.com/articles/native-vs-hybrid-vs-cross-platform-how-and-what-to-choose#:~:text=Hybrid%20has%20limited%20access%20to,match%20the%20Native%20app%20development>
- Latif, M., Lakhri, Y., Nfaoui, E., & Es-Sbai, N. (2016). Cross platform approach for mobile application development: A survey. *2016 International Conference on Information Technology for Organizations Development (IT4OD)* (pp. 1-5). Fez: IEEE.
- Le, J. (2019, October 31). *Recommendation System Series Part 2: The 10 Categories of Deep Recommendation Systems That Academic Researchers Should Pay Attention To*. Retrieved from Towards Data Science: <https://towardsdatascience.com/recommendation-system-series-part-2-the-10-categories-of-deep-recommendation-systems-that-189d60287b58>
- Lops, P., de Gemmis, M., & Semeraro, G. (2010). Content-based Recommender Systems: State of the Art and Trends. In *Recommender Systems Handbook* (pp. 73-105). Boston: Springer.

- Mahmoud, Q. H. (2005, April). *Service-Oriented Architecture (SOA) and Web Services: The Road to Enterprise Application Integration (EAI)*. Retrieved from Oracle:
<https://www.oracle.com/technical-resources/articles/javase/soa.html>
- O'Hear, S. (2013, July 9). *Soundwave, The Music Discovery App Backed By Mark Cuban, Is Placing A Big Bet On Big Data*. Retrieved from Techcrunch:
<https://techcrunch.com/2013/07/09/soundwave/>
- Red Hat. (n.d.). *What is an API*. Retrieved from Red Hat:
<https://www.redhat.com/en/topics/api/what-are-application-programming-interfaces>
- Rocca, B. (2019, June 3). *Introduction to Recommender Systems*. Retrieved from Towards Data Science: <https://towardsdatascience.com/introduction-to-recommender-systems-6c66cf15ada>
- Strickland, J. (2009, May 19). *How Goodreads Works*. Retrieved from How Stuff Works:
<https://computer.howstuffworks.com/internet/social-networking/networks/goodreads.htm>
- Summerfield, J. (-). *Mobile Website vs. Mobile App: Which is Best for Your Organization?* Retrieved from Human Service Solutions:
<https://www.hswsolutions.com/services/mobile-web-development/mobile-website-vs-apps/>
- Wang, H., Wang, N., & Yeung, D.-Y. (2015). Collaborative Deep Learning for Recommender Systems. *KDD '15: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1235-1244). New York, NY: Association for Computing Machinery.
- Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W. L., & Laskovec, J. (2018). Graph Convolutional Neural Networks for Web-Scale Recommender Systems. *KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 974-983). London: Association for Computing Machinery.