A SURVEY OF TECHNIQUES USED IN RECOMMENDATION SYSTEMS AND AN APPLICATION OF THE COLLABORATIVE FILTERING TECHNIQUE IN A RECOMMENDER SYSTEM

Kushal Dave
University of Regina
kdp799@uregina.ca

Stanley Eze
University of Regina
ees007@uregina.ca

Sibdow Abdul-Jalil Iddrisu
University of Regina
sip099@uregina.ca

ABSTRACT

Recommender systems are one of the most essential technologies out there, with the exponential increase in data and the ever-changing preferences of users. People often use the web for shopping, searching movies, books, etc. and the power of recommender systems makes it easy for the companies who offer these services to easily identify/predict user preferences and make some suggestions of products they may be interested in. Also, from the user point of view, s/he does not have to search hours on end to find what they want as most suggestion typical or rightly give relevant suggestions to users. Recommender systems are typically implemented with one of either three (3) strategies: collaborative technique, content-based technique and a hybrid-based technique which integrates both previous two strategies. In this report, we explore the various techniques used in recommender systems and also present a movie-based recommender system built using the Collaborative Filtering Technique, which is by far the most popular. In the end, we present some of the most popularly used evaluation techniques, and we applied them in the evaluation of our developed system.

Keywords: Recommender System, Collaborative Filtering, User-based filter, Euclidean Distance, Cosine Similarity

I. INTRODUCTION

The advent of the internet, e-commerce, and social media has seen a rapid increase in the data and information being created. It is estimated that 2.5 quintillion bytes of data are created each day and within the last 2 years, 90% of the current world's data was created [7]. With this rapid increase in data created, users usually find it very difficult because it is impossible to easily find relevant information about an item or a product they may be interested in. Cooperate organisations resort to recommender systems to help them help their customers find just the right items relevant to their needs. These recommender systems have been deployed in various aspects of the web including Facebook; where friends are usually suggested based on mutuality, Amazon; where an item is recommended based on similar user activities and/or similar items, Netflix; which is a movie streaming site also deploys a recommender system to help suggest movies to users.

There are three (3) major techniques developed over the years to serve as a framework to help build recommender systems. The first and by far the most popular – collaborative filtering technique which is grounded on the fact that users who share some views of some common items may have similar tastes on other items they may not share at the moment. The main merit of this method is in its simplicity but can be difficult to find recommendations for new users or items added to the database of user and/or items.

Content-based filtering is the other technique that tries to solve the problems encountered when using the collaborative-filtering technique. It uses the concept of matching document/item representation with user preferences and/or profile.

The Hybrid filtering combines the techniques involved in the collaborative filtering technique as well as the content-based technique.

The remainder of the report is ordered as follows. In section II, we discuss the collaborative-filtering technique in detail and provide some of the mathematical theorems behind the technique. In section III, we also look at the content-based filtering, and in section IV, the hybrid filtering technique is discussed. In section V, we present a recommender system developed using the collaborative-filtering technique. We then provide in section VI some system evaluation methods used in the evaluation of recommender systems. Finally, in section VII, we conclude the paper and a discussion for future developments with our system

II. COLLABORATIVE FILTERING

Collaborative filtering is basically people collaborate to help each other to perform filtering by recording their reactions to an item that they have read/seen/watched/reviewed. As an example, Netflix uses data from its users, when user start streaming any of the videos on Netflix for few or more seconds which helps Netflix to get an idea about users likes and/or dislikes. Now if any other user also likes and/or dislike the same video, then that makes it possible to find similarity between users. Now from different users whoever have matching similarities, Netflix starts recommending other videos watched/liked by that user.

Collaborative filtering is the simplest and powerful method for filtering when we have a very large amount of collaborative data from different users. It can be done in mainly two (2) different ways:

1) Memory-based: This technique uses reviewed data to compute the similarity between users or items. This approach uses data from other users or items which are already rated before, which play a relevant role in searching for a neighbour of users [9][11]. By combining preferences from these neighbours and using different algorithms recommendation for the user is generated. Because of the efficiency of this technique, it has a big impact on the real-life application [5].

Memory-based technique classified mainly in two different ways: a) User-based Filtering: Where we look for the similar taste of items between different users and recommends items from those neighbour users which are not reviewed by the user. b) Item-based Filtering: Where it looks for similar items only instead of users and recommends similar those items to the user.

There are various models used in the calculation of this similarities between either user-user or item-item. These include:

- Euclidean Distance
- Pearson Correlation
- Cosine Similarity

2) Model-based: This technique uses model generated from dataset to recommend to users. Instead of using the whole dataset it uses a part of data as a model to recommend. This technique also uses previous reviews of users to improve collaborating filtering technique. The model learning technique is done using data mining or machine learning techniques like clustering, decision tree, regression, Bayesian classifier, matrix technique [1].

Disadvantages: If there is a new item or user introduced then it is hard to find similarity using collaborative filtering which is called a Cold start problem. But there are many different techniques to solve the cold start problem. Process time, complexity is increased with a large amount of data.

III. CONTENT-BASED FILTERING

The collaborative filtering technique has some drawbacks that the content-based filtering technique addresses. We may not be able to recommend items to new users who have not yet rated any item, as we may not be able to compare them to other users based on this rating. This technique only involves the user alone and does not depend on other users. The system recommends an item to a user based on the representation of an item and the user's profile and preferences [3]. This system works more like an Information Retrieval (IR) System, where we have a set of indexed documents/items and what the user wants is matched with the set of documents/items, and then subsequently retrieved.

In this method, items are given a representation, and the type of representation depends on the type of item. Mostly movies are represented with their metadata, i.e., descriptions about the movie, director, genre, actors, etc. Documents are generally represented as a set of indexed term, where certain concepts go into picking the right terms to represent a document. Luhn described the process of document indexing [4] by using the Zipf's law [6]. General stop-words are eliminated, and least frequent words are eliminated as they do not contribute much to uniquely identifying a document, a count of the remaining words are taken and represented as a matrix of indexed terms, commonly known as the term-frequency (TF) matrix. A method called Inverse-Document-Frequency is also used to calculate the number of documents containing a term and this describes a term's rarity in a collection of documents. A term-frequency * Index document frequency (TF-IDF) is then used to assign weights to the terms and stored in the database.

term frequency (d_i, t_j) = number of times terms t_j appears in a document d_i

document frequency (term_j) = $\sum_{i=0}^{n} \ (document\ i)$

IDF = 1/ log (df)

Weight (term, document) = TF * IDF = TF * 1/ log (document frequency)

After the item is represented and stored, a user profile and/or preference is created for every individual user, in this case, no user is dependent on the other, and recommendations can be made to users without them explicitly ranking an item. The user profile can be created using their browsing history, their customised settings on the recommender application, cookies, search history, purchase history, etc. The creation of these user preferences is based on machine learning models such as classification; where user history is divided into binary categories such as “item liked” and “items not liked” [3]. The decision tree models and nearest neighbour methods [3] may also be used to model the user preferences.

After the item is represented and user profile is created/learned, the content-based filtering method suggests relevant items to a user by using a matching function to match the user profile against item. This matching is done using similarity measurement strategies such as the COSINE SIMILARITY and the scores returned are used to rank the potentially relevant items with respect to the user’s preferences [8].

$$\textbf{similarity (userA, userB)} = \cos(\boldsymbol{\theta}) = \frac{\textbf{userA. userB}}{||\textbf{userA}||*||\textbf{userB}||} = \frac{\sum_{i=0}^{n} \boldsymbol{userA\ i * userB\ i}}{\sqrt{\sum_{i=0}^{n} (\boldsymbol{userA\ i})^2}\ \sqrt{\sum_{i=0}^{n} (\boldsymbol{userB\ i})^2}}$$

This technique solves the problem of user dependence in the collaborative-filtering technique and only performs recommendations based on users’ taste. Also, if a new product is added to the system that has not been rated by any user yet can still be recommended to users based on their preference match to that item. One concern with this technique is that a new user’s preferences would have to be learned before relevant recommendations can be made.

IV. HYBRID FILTERING

Both collaborative and content-based methods have their pros and cons. It is nearly impossible to always achieve the best recommendation result using just one of the two recommendation techniques earlier discussed, and that led to the need for a hybrid recommendation technique that combines the two in various ways with fewer drawbacks than any individual method. To avoid some limitations of collaborative and content-based methods and for greater optimization, A hybrid recommendation techniques combine multiple sources of information and various techniques in making a recommendation. The idea within this method is to combine the best features of collaborative and content-based recommender algorithm into one hybrid technique to produce a more satisfactory recommendation than a single technique [1].

Hybrid filtering technique has various ways of combining two methods to achieve better performance, and this depends on the users need, and it is specific to some particular application of the recommendation system. A hybrid combination of a collaborative filtering and content-based filtering could be in the form of implementing both methods and selecting the best result after each recommendation, integrating the best characteristics of collaborative filtering into the content-based algorithm, incorporating the best features of the content-based algorithm into collaborative algorithm, or a single model that assimilates both algorithms [2]. These approaches are categorized into various groups which include;

- **Weighted**

In this type of hybrid recommendation, scores of various recommendation technique are joined to produce a single recommendation [2]. That is, scores of various techniques are summed, and the resulting score is equal to the score of the recommended item. These scores are usually combined using some statistical techniques with an additive aggregation for normalization. This method is considered as the most popularly used [12]

- **Switching**

Switching hybrid recommendation system implements both collaborative and content-based in one system. It switches between the two based on the kind of task available. The system has certain criteria for switching, and it is based on the better approach to a particular problem. This kind of recommendation makes the system sensitive to the strength and weaknesses of its constituent recommenders [2].

- **Mixing**

This technique displays recommendations from different techniques at the same time. Multiple rating list are presented into a single rated list [1]. The system makes its final suggestion based on the result of the combination of the recommendation from the two techniques. This is considered as the least used technique since both recommenders that makes up the hybrid are not glued. That is, both recommenders make independent recommendations from each other.

- **Feature Combination**

Feature combination hybrid system combines both techniques in one system. In this case, one of the techniques is the actual recommender while the second is the contributor. The contributor provides data to the actual technique for a recommendation to be made. That is, the actual technique requires some information from its counterpart to make a recommendation [2].

- **Feature Augmentation**

This type of hybrid recommendation system is employed for an improved quality recommendation by the recommender system. The task of recommendation is divided between the two recommender methods where one technique is used for rating or classification of items after which the data or information is incorporated in the next technique for further processing [2]

V. OUR RECOMMENDER SYSTEM

After a thorough survey of the various methods used in the implementation of recommender systems, we designed and developed a recommender system; this makes use of the collaborative filtering technique discussed above. We used the python programming language to write the logic and a **movie/book** dataset to test the system. It mainly uses the user-user similarity measure to make recommendation of items to similar users.

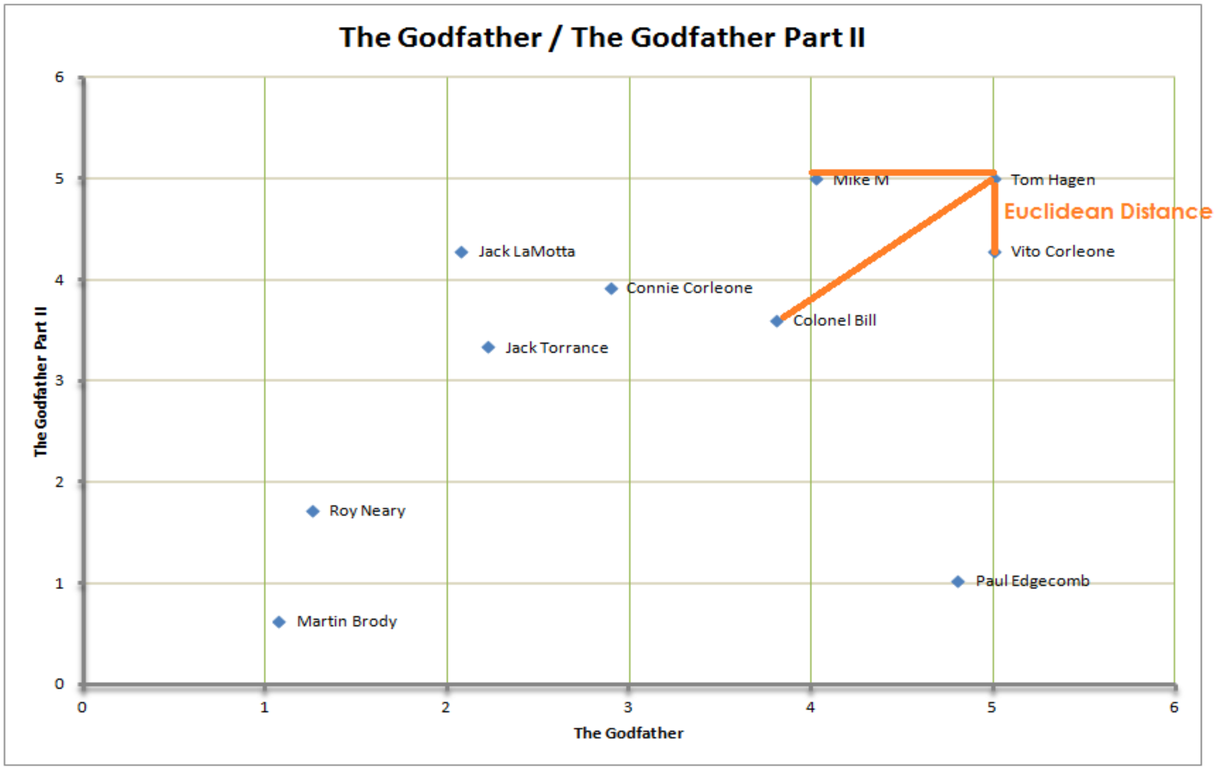
We made use of the **Euclidean Distance** and **Cosine Similarity** measurements of the collaborative-filtering technique. We also came up with a way of merging these two (2) measures to produce a **Hybrid Measure**. For the rest of this section, we discuss in details our implementation.

- **EUCLIDEAN DISTANCE**

The Euclidean distance measures the length of the segment connecting two distinct points in a space. In our implementation, we used it to calculate the distances between a user we would like to make recommendations of items to with other users of the system; the lower the distance measure between the former and the later, the more similar they are and we can make relevant recommendations based on that but the higher the distance, the less similar the two users are.

$$\text{Euclidean Distance} = \text{dist(userA, userB)} = \sqrt{(userA\ -\ userB)^2 + \ (userA\ -\ userB)^2}$$

$$= \sqrt{\sum(userAi\ -\ userBi)^2}$$



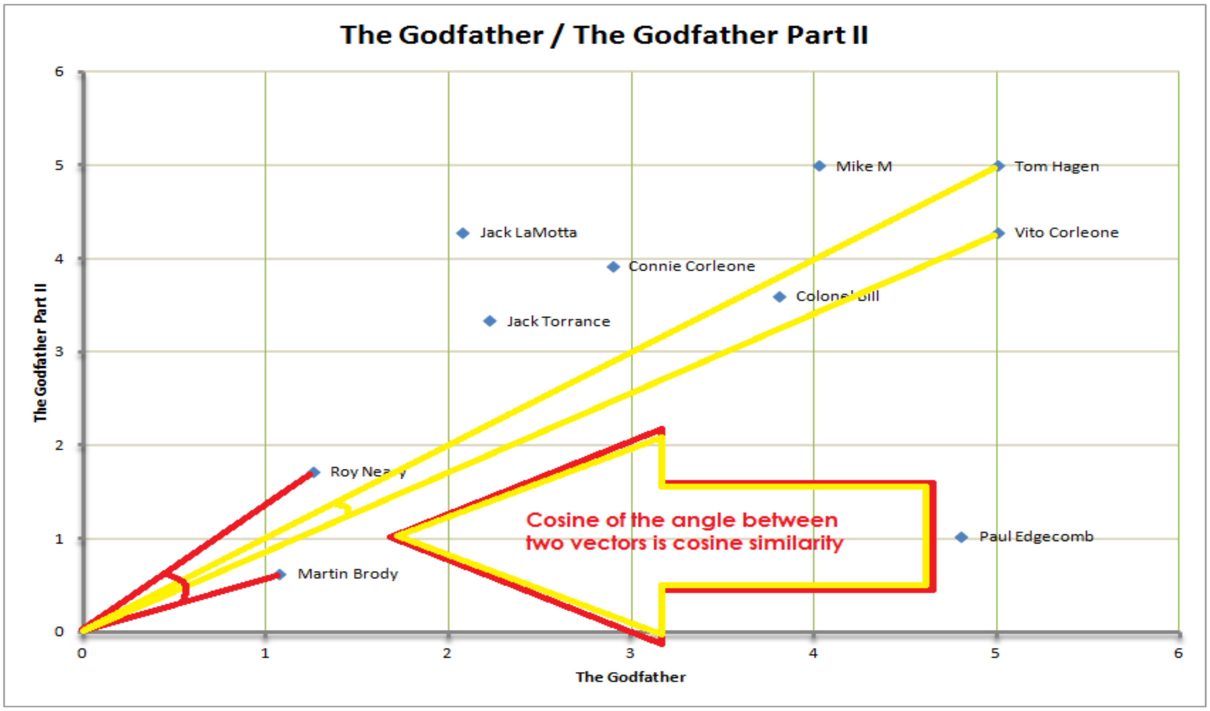
- **COSINE SIMILARITY**

This measure the angle between two non-zero vectors in a plane. We used this formula to calculate the distance angle between two system users, and based on this we are able to determine which user is more similar to the other. Based on this, we are able to make relevant recommendations for similar users. The cosine can be formulated by using the Euclidean dot product formula:

$$\mathbf{userA.userB} = \mathbf{||userA||\ ||userB||\ cos}\boldsymbol{\theta}$$

Given two system users, say USER1 and USER2, the cosine similarity which is a measure of the distance angle between them can be represented as:

$$\text{Similarity} = \mathbf{cos}\ (\boldsymbol{\theta}) = \frac{\boldsymbol{USER1\ .\ USER2}}{\boldsymbol{||USER1||\ ||USER2||}} = \frac{\sum_{i=0}^{n} \boldsymbol{USER1\ i * USER2\ i}}{\sqrt{\sum_{i=0}^{n}(\boldsymbol{USER1\ i})^2}\ \sqrt{\sum_{i=0}^{n}(\boldsymbol{USER2\ i})^2}}$$



- **HYBRID MEASURE**

The hybrid measure combines both the Euclidean measure and cosine measure; then it calculates the average. This measure provides us with a harmonic mean of the two measures and gives us a better way of making relevant recommendations to users.

Formula: $$\textbf{Hybrid Measure} = \frac{\textbf{Euclidean Distance + Cosine Measure}}{\textbf{2}}$$

VI. SYSTEM EVALUATION

Every user-centric system has a way of verifying whether the system has delivered to the user, to some extent, exactly what they want, and the recommender system is no exception. There are various ways of measuring a recommender system's effectiveness, and this section covers the most widely used ones. Information Retrieval has some intersection with Recommender Systems, and it is no surprise that some of its system evaluation techniques are applied to Recommender systems. There are two major measurements, namely:

- **Precision**: This measure shows how useful a recommended item is to the user. It can be defined as the ratio of recommended and relevant items to all the recommended items returned.

$$\textbf{Precision} = \frac{|\boldsymbol{Relevant} \cap \boldsymbol{Recom}|}{|\boldsymbol{Recom}|} = \frac{|\boldsymbol{Relevant} \cap \boldsymbol{Recom}|}{|\boldsymbol{Relevant} \cap \boldsymbol{Recom}| + |\overline{\boldsymbol{Relevant}} \cap \boldsymbol{Recom}|}$$

$$\mathbf{0 \leq Precision \leq 1}$$

- **Recall**: This measure shows how complete a search result is. It can be defined as the ratio of recommended and relevant documents to the ratio of all the relevant items in the system (both recommended and not recommended)

$$\textbf{Recall} = \frac{|\boldsymbol{Relevant} \cap \boldsymbol{Recom}|}{|\boldsymbol{Relevant}|} = \frac{|\boldsymbol{Relevant} \cap \boldsymbol{Recom}|}{|\boldsymbol{Relevant} \cap \boldsymbol{Recom}| + |\boldsymbol{Relevant} \cap \overline{\boldsymbol{Recom}}|}$$

$$\mathbf{0 \leq Recall \leq 1}$$

There is a kind of inverse relationship between precision and recall. A higher precision means we get a small number of items returned but contains a lot of relevant items whereas a higher recall means we get a large number of retrieved items, but most of them are non-relevant to the user needs. In practise, a compromise is made between these two measures to achieve the best recommended items relevant to a user need. These measures evaluate the capability of a recommender system to provide an ordered list of items that a user likes [10].

Another system evaluation method used for recommender systems are based on the ability of the method to predict a user's taste for an item(s). They are:

- **MEAN ABSOLUTE ERROR (MAE)**: Given T = {u, i, r} where T depicts (user$_i$, item$_i$, rating$_i$). The formula for this measure can be given as:

$$\mathrm{MAE} = \frac{1}{|T|} \sum_{(u,i,r) \in T} |p(ui) - r| \qquad [10]$$

  p(ui): the probability of prediction of an item to a user

- **ROOT MEAN SQUARED ERROR (RMSE)**: Using the same definition above; the formula for this measure can be given as:

$$\text{RMSE} = \sqrt{\frac{1}{|T|} \sum_{(u,i,r)\, \in T} (p(ui) - r)^2} \qquad [10]$$

p(ui): is the probability of prediction of an item to a user

Both measures based on Information Retrieval and Prediction can be used to evaluate a Recommender System so as to be able to know how good the system is performing and ways to improve the system.

VII. CONCLUSION

There have been several techniques introduced to help in the implementation of these system which includes: Collaborative-filtering, Content-based filtering and a hybrid method. The collaborative filtering being the most popular and easiest to implement.

“Linking this system to AI and what is learnt in class, the task of making recommendation to users can be seen as a learning problem that makes use of past knowledge about users to make these predictions” [8].

We implemented the collaborative-filtering technique with the Euclidean distance measure, Cosine similarity and a hybrid method that combines the former two methods. There was also a presentation of techniques to evaluate a system based on Prediction and Information Retrieval. Our system currently executes in the command prompt, and we intend to add a graphical user interface (GUI) in the future. Also, other techniques of recommender systems are planned to be developed so as to make it more robust to handle different user scenarios.

It is inevitable that recommender systems play a vital role in today’s information age where we are overwhelmed with massive amounts of data to deal with on a daily basis. A recommender system ensures that users get what they want within the shortest possible time without having to search the web for long periods of time.

REFERENCES

[1] B, P., P, D. and U, P. (2017). Methods of Recommender System: A Review. *2017 International Conference on Innovations in information Embedded and Communication Systems (ICIIECS).*

[2] Kunal, S., Akshaykumar, S., Saurabh, D. and Antala, K. (2019). Recommender Systems: An overview of different approaches to recommendations. *International Conference on Innovations in information Embedded and Communication Systems (ICIIECS).*

[3] M. Pazzani and D. Billsus, "Content-Based Recommendation Systems", *The Adaptive Web*, pp. 325-341. Available: https://link.springer.com/chapter/10.1007/978-3-540-72079-9_10. [Accessed 22 March 2019].

[4] H. P. Luhn, "The Automatic Creation of Literature Abstracts," in *IBM Journal of Research and Development*, vol. 2, no. 2, pp. 159-165, Apr. 1958.
doi: 10.1147/rd.22.0159

[5] F. Isinkaye, Y. Folajimi and B. Ojokoh, "Recommendation systems: Principles, methods and evaluation", *Egyptian Informatics Journal*, vol. 16, no. 3, pp. 261-273, 2015. Available: 10.1016/j.eij.2015.06.005 [Accessed 20 March 2019].

[6] C. J. Van Rijsbergen, "Information Retrieval", Butterworth-Heinemann Newton, MA, USA 1979.

[7] B. Marr, "How Much Data Do We Create Every Day? The Mind-Blowing Stats Everyone Should Read", *Forbes.com*, 2018. [Online]. Available: https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/#5c4689c260ba. [Accessed: 20- Mar- 2019].

[8] Lops P., de Gemmis M., Semeraro G. (2011) Content-based Recommender Systems: State of the Art and Trends. In: Ricci F., Rokach L., Shapira B., Kantor P. (eds) Recommender Systems Handbook. Springer, Boston, MA; doi: https://doi.org/10.1007/978-0-387-85820-3_3

[9] Zhi-Dan Zhao and Ming-Sheng Shang, "User-Based Collaborative-Filtering Recommendation Algorithms on Hadoop", *2010 Third International Conference on Knowledge Discovery and Data Mining*, 2010. Available: 10.1109/wkdd.2010.54 [Accessed 22 March 2019].

[10] L. Candillier, F. Meyer and M. Boullé, "Comparing State-of-the-Art Collaborative Filtering Systems", *Machine Learning and Data Mining in Pattern Recognition*, pp. 548-562. Available: http://lcandillier.free.fr/publis/MLDM07.pdf. [Accessed 24 March 2019].

[11] XiaoMing Zhu, HongWu Ye, and SongJie Gong. 2009. A personalized recommendation system combining case-based reasoning and user-based collaborative filtering. In *Proceedings of the 21st annual international conference on Chinese control and decision conference* (CCDC'09). IEEE Press, Piscataway, NJ, USA, 4062-4064.

[12] Çano, E. and Morisio, M. (2017). Hybrid recommender systems: A systematic literature review. *Intelligent Data Analysis*, 21(6), pp.1487-1524.