A Comprehensive Study of Retail Based Recommendation System

Nirupma Singh[1], Siddharth Nanda[2]
Faculty - IT, iNurture Education Solutions, Bengaluru, India[1,2]

Abstract - At present, various communication channels are booming. Almost all are connected with internet. In this age of connectivity and globalization various retail platforms are also growing. We have various online retail platforms available currently like Amazon, Flipkart etc. So all these platforms are trying their best to improve their sales. Improvement of sales can be achieved by provide the customer with good experience and proper product satisfaction. Every customer wants to have with multiple choices and sometimes it's a boon or issue. This problem can be sorted based on various recommendation systems being used today. Here we have illustrated about different algorithms used for recommendation such as collaborative filtering system, Matrix Factorization and KNN. We have a compared and analysed the use of these algorithms to present a descriptive view. The recommendation systems will help the customer ease their shopping pattern and save time for the customer also. The prime target is to improve customer experience for the various shopping portals.

Keywords - Recommendation System, k-nearest Neighbours (KNN), Collaborative Filtering, Matrix Factorization.

I. INTRODUCTION

The tremendous growth of the world-wide-web and the emergence of e-commerce has led to the development of recommender systems. [1] To successfully make a decision customers need suggestions of recommendation system. Based on suggestions of recommendation systems which were retrieved from other individuals or authorities. Relying on these derivations choices can be made even without adequate first-hand information or idea of all available options.

[2] Humans manage and govern this information overload through their own efforts, help from others and certain amount of good fortune. First, most elements and information are removed from the transmission simply because they are inaccessible or invisible to the user. Second, a large amount of filtering is done for us. Newspaper publishers select which articles their readers want to read. The bookstores decide which books to carry. However, with the beginning of the era of electronic information, this barrier will become less and less a factor. Finally, we trust friends and other people we trust to make recommendations. We need technology to help us analyze all the information which can help us sort items we really want and need, and to unsubscribe from the things we do not want bothered with.

As per machine learning techniques, a recommendation system makes predictions based on the historical behavior of users. Specifically, it is to predict the user's preference for a set of elements based on past experiences. To create a recommendation system, the two most popular approaches are content-based and collaborative filtering.

The content-based approach requires a good amount of information about the characteristics of the elements, instead of using the user's interactions and comments. For example, they can be attributes of movies such as genre, year, director, actor, etc., or textual content of articles that can be extracted by applying Natural Language Processing. Collaborative Filtering, on the other hand, does not need anything else, except the historical preference of the users in a set of elements. Because it is based on historical data, the basic assumption here is that users who have agreed in the past also tend to agree in the future. In terms of user preference, it is usually expressed in two categories. Explicit Classification is a rate given by a user to an item on a sliding scale, such as 5 stars for Titanic. This is the most direct response from users to show how much they like an article. Implicit classification, suggests the preference of users indirectly, such as visits to pages, clicks, purchase records, listen or not listen to a music track, etc. In this article, I will closely analyze the collaborative filtering that is a traditional and powerful tool for recommendation systems.

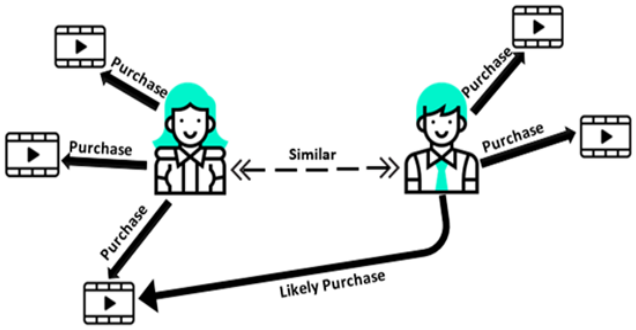

Figure 1: Similarity Evaluation

The standard method of collaborative filtering is known as the closest neighbourhood algorithm. There are user-based FCs and article-based CFs. Let's first look at the CF based on the user. We have a n × m notation matrix, with the user u_i, i = 1, ... n and the item p_j, j = 1, ... m. Now, we want to predict the note r_{ij} if the target user I have not looked at / noted an element j. The process consists of calculating the similarities between the target users i and all the other users, selecting the first X similar users and taking the weighted

average of the scores of these X users having similarities as weight.

$$r_{ij} = \sum_k Similaries\,(u_i, u_k)\,.\,r_{kj} \div number\ of\ ratings$$

Although different people may have different reference levels in the assessment, some people tend to assign high scores in general, but some are quite strict even if they are satisfied with the elements. To avoid this bias, we can subtract the average rating of all items from each user when calculating the weighted average and add it for the target user, as shown below.

$$r_{ij} = \bar{r}_i + \left[\sum_k Similaries\,(u_i, u_k)\,.\,(r_{kj} - \bar{r}_k) \div number\ of\ ratings\right]$$

Two ways to calculate similarity are Pearson Correlation and Cosine Similarity.

Pearson Correlation: $Sim(u_i,u_k) = \sum_j (r_{ij} - r_i)\,.\,(r_{kj} - r_k) \div \sqrt{\sum_j (r_{ij} - r_i)^2 \sum_j (r_{kj} - r_k)^2}$

Cosine Similarity: $Sim(u_i,u_j) = (r_i \times r_k) / |r_i|\,|r_i| = (\sum_{j=1}^{m} r_{ij}\ r_{kj}) \div \sqrt{\sum_{j=1}^{m} r_{ij}^2 \sum_{j=1}^{m} r_{kj}^2}$

Basically, the idea is to find the users most similar to your target user (closest neighbours) and to weight their assessments of an element as a prediction of the evaluation of that element for the target user.

Without knowing anything about the elements and the users themselves, we think that two users are similar when they assign similar evaluations to the same element. Similarly, for item-based CF, we say that two items are similar when they have received similar ratings from the same user. Next, we will predict for a target user an item by calculating the weighted average of the scores assigned to most X similar items of that user. One of the main advantages of article-based CF is stability, ie the ratings assigned to a given item will not significantly change overtime, contrary to human tastes.

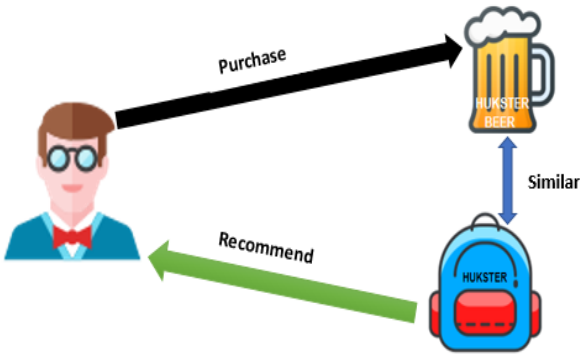

Figure 2: Nearest Neighbour Recommendation

Matrix Factorization: Since scarcity and scalability are the two main challenges of the standard CF method, it is a more advanced method that breaks down the initial dispersed matrix into small matrices with latent factors / characteristics and dispersion. less. This is the matrix factorization.

In addition to solving scarcity and scalability issues, there is an intuitive explanation of why we need multidimensional arrays to represent user preferences. A user gave good ratings to the movie Avatar, Gravity and Inception. These are not necessarily 3 separate opinions, but they show that these users may be in favour of sci-fi movies and that there could be many more sci-fi movies that this user would like. Unlike specific movies, latent functions are expressed by higher-level attributes, and the Science Fiction category is one of the latent functions in this case. The factorization of the matrix finally gives us how much a user is aligned on a set of latent characteristics and how much a movie fits into this set of latent characteristics. The advantage of this compared to the nearest standard neighbourhood is that, although two users have not ranked any of the same movies, it is still possible to find a similarity between them if they share similar underlying tastes, again latent characteristics.

Collaborative filtering gives predictive systems great predictive power and requires the least information at the same time. However, it has certain limitations in certain particular situations.

First, the underlying tastes expressed by latent characteristics are not really interpretable because there are no content-related metadata properties. In the example of the film, it is not necessarily a genre like that of science fiction in my example. It can be the motivation of the soundtrack, the quality of the plot, etc. Collaborative filtering is the lack of transparency and explicability of this level of information.

On the other hand, collaborative filtering is facing a cold start. When a new article arrives, until a significant number of users qualify, the template cannot make personalized recommendations. Similarly, for those items in the queue that have not received too much data, the model tends to give them less importance and favour popularity when they recommend popular items.

In general, it makes sense to have common algorithms for building a more complete machine learning model, for example combining content-based filtering by adding explainable keyword dimensions, but we should always consider the Balance model / compute the complexity and efficiency of performance improvement.

II. LITERATURE SURVEY

Recommender Systems in E-Commerce [3] Online sales of various portals can be improved in various ways. Firstly it can help window shoppers to actually buyers. People often visit websites, surf the site and end up buying nothing. The recommendation system can help these customers to find commodities of their interest and need hence making a successful purchase. Also recommendation systems helps in improving the sales of any online shopping porta by helping the consumer buy a complementary product hence improving the sales. There are various online shopping

portals at the dispersal of the mankind. So in this critical commercial competitive environment gaining customer support and loyalty is a key factor. Recommendation system helps in improving the relationship between the portal and the customer. The recommendation system learns the habits and the operations of working of the customer and each time the customer visits the portal comes up with best matching options as per the customer's requirement. This best helps and improving the sales as a satisfied customer is the best gift.

Content Based, Collaborative Recommendation. [4] The proposition of collaborative method for recommending is not very usual. This technique doesn't recommend you options or commodities based on the users past purchasing habits but it makes suggestions based on the items liked by similar users. The comparative analysis of the users is done here rather than computing the similarity of commodities. So here they have clearly deployed the concept of nearest neighbor based on the comments and ratings on certain products from which we can draw a conclusive or say very strong correlation.

Combining Content-Based and Collaborative Recommendation[5] Here, a content-based approach is used to recommend. Content-based recommendation systems work primarily on methods that use information retrieval or similar methods. While the collaborative approach to advocacy is very different: rather than recommending items because they look like items a user has enjoyed in the past, we recommend items that other similar users have liked . Rather than calculating the similarity of the elements, we calculate the similarity of the users. Here, the method used combines the best of both techniques. Here, user profiles are managed based on content analysis. All of these well-maintained user profiles are benchmarked to determine similar users. Users receive suggestions both when they score high on their own profile and when they are rated by a user with a similar profile. This hybrid approach avoids the mentioned limitations for collaborative and content-based systems, while adding significant benefits to improve sales.

Recommendation as classification: Using social and content-based information in recommendation.[6]In this paper the authors are trying to create a method where the users can get better results by exploiting the best of both ratings and content based data. They have taken an approach which deals with fulfilling the gaps and drawbacks of older frameworks like social filtering and content based filtering. Here the authors have formalized the movie suggestion problem as a learning problem specifically, the problem of learning a function that takes as its input a user and a movie and produces as output a label indicating whether the movie would be liked (and therefore recommended) or disliked:

f(user, moviel) → {liked, disliked}

Here as a problem in classification, the authors have also tried predicting whether a movie viewer has liked or disliked and not an exact rating. Their approach has an output as a set of movies whose predictions will be liked by the viewer. They are able to generalize the inputs to the problem to other information describing both viewers and movies.

An agglomerative clustering of a search engine query log.[7] Here universal methodologies like hierarchical agglomerative clustering (HAC) is being discussed. In this method a least quadratic in 'n' always repeatedly finds the closest two documents and merges them together. They have explained in the form of an example let us consider the far are the two documents from each other can be calculated without examining the contents of those documents. This type of characteristic is referred to as "content - ignorance". This is completely different to older techniques like content - aware clustering algorithms, which typically uses as a distance between documents some function of the fraction of points which they have in common. Clustering web pages by content may require storing and manipulating a staggeringly large amount of data.

Item-based top-n recommendation algorithms.[8] Here the authors have focused on a particular technique called as Top - n recommendation algorithms which analyze the similarities between various searches of commodities and based on this similar options or related commodities will be suggested to the user. Top-N recommendation algorithms have been deployed in different formats since the early days of CF-based recommender systems and are known to be computationally scalable and implementable (both in terms of model construction and model application). But due lack of user's purchasing habit related data it can produce less accurate recommendations when compared to user-based algorithms.

Dependency networks for inference, collaborative filtering, and data visualization. [9] In this article, the authors illustrated a graphical method for the probabilistic relation. Here, they compared the technique mentioned above with the Bayesian network, also called dependence network. These types of charts are more likely to be cyclical. The probability factor of a dependency network, like a Bayesian network, is a set of conditional distributions, one for each node, according to its parents. We identify several basic properties of this representation and describe a computationally efficient procedure for learning graph and probability components from data. The application of this representation to probabilistic inference, collaborative filtering (the task of predicting preferences) and the visualization of causal predictive relations.

Clustering navigation patterns on a website using a sequence alignment method.[10] Here, the discussion focuses on the sequence alignment method which is a non-Euclidean distance measure that reflects the order of the elements and is deployed in various domains. The sequence alignment method is also called string editing distance. He also used the health sector. It is used for molecular biology by sequence comparison and speech recognition. In general, the similarity between the sequences is reflected by the amount of work to be done to convert one sequence into another. As a result, the SAM distance measurement or

similarity can be scaled and represented by a score. On the basis of the score that can be high or low, the effort needed to equalize the sequences is greater or less. In addition, SAM marks for the following operations during the equalization process: insert, delete, and reorder. Insert and delete operations are applied to single items. the reordering operation is applied to the common elements. This evolutionary data makes it possible to obtain data allowing to improvise the recommendation system.

Empirical Analysis of Predictive Algorithms for Collaborative Filtering[11] The recommendation system uses a database configuration to store information about the user's preferences. This database keeps records of user details and helps to predict new products or products that a user might like. Automated search in a corpus of elements is based on a query that identifies the intrinsic characteristics of the searched elements. Searching for text documents (for example, web pages) uses queries containing desired words or concepts in the returned documents. Searching for CD titles, for example, requires identification of the artist, genre, or desired period. Most content extraction methodologies use a similarity score type to match a query describing the content with titles or individual items, and then present the user with a classified list of suggestions. A complementary method of identifying potentially interesting content uses data on the preferences of a set of users. Normally, these systems do not use any information about the actual content (eg, words, author, and description) of the elements, but are based on usage patterns or preference of other users. The so-called collaboration filters or recommendation filters are based on the assumption that a good way to find interesting content is to look for like-minded people and then recommend titles that are liked by similar users.

Recommender Systems in E-Commerce[12] In this article, they made five contributions to the understanding of recommendation systems in e-commerce. First, we provide a series of examples of referral systems covering the range of different applications of e-commerce referral systems. Second, we analyze how each of the examples uses the referral system to improve revenue on the site. Third, we describe an application allocation from recommendation systems to a taxonomy of application implementation forms. Fourth, we examine the efforts required by users to find recommendations. Fifth, we describe a set of suggestions for new recommendation system applications based on parts of our taxonomy that have not been explored by existing applications. The document is useful for two groups: academics who study referral systems in e-commerce and implementers who plan to implement referral systems on their site.

III. EXISTING SYSTEM APPROACH

In ancient techniques, which use the gamma-Gaussian distribution, we see the theory of probability and statistics, the normal gamma distribution (or gamma-Gaussian distribution) is a bivariate family of four parameters of continuous probability distribution. This is the previous conjugate of a normal distribution with unknown mean and precision. The normal distribution is by far the most important probability distribution. One of the main reasons for this is the Central Limit Theorem (CLT), which will be discussed later in the book. To give you an idea, the CLT indicates that if you add a large number of random variables, the distribution of the sum will be roughly normal under certain conditions. The importance of this result comes from the fact that many real-life random variables can be expressed as the sum of a large number of random variables and that, through the CLT, we can argue that the distribution of the sum must be normal. CLT is one of the most important results in probability and we will discuss it later. Here we will introduce normal random variables.

We first define the standard normal random variable. We will then see that we can obtain other normal random variables by scaling and shifting a normal random variable.

A continuous random variable Z is said to be a normal normal random variable (Gaussian standard), represented by Z~N (0,1), if its PDF is given by

$$f_z(z) = 1/\sqrt{2}\pi \, .e^{-z^2/2}, \forall z \in R$$

IV. PROPOSED SYSTEM APPROACH

From a commercial point of view, a subscription-based service template that offers personalized recommendations to help you find programs or any video that may be of interest to you. To do this, we need to create a system of recommendations based on search and user suggestion. These user-based surveys, in turn, will help the system estimate the likelihood of attending a particular item or show the program based on a number of factors, including: (a) user interactions with e-commerce portals covering viewing history and how the user rates various items, (b) other users with similar tastes and preferences, and (c) information about items such as genres, categories, actors, year of release, etc. In addition to knowing what you have seen or navigated, the portal can provide personalized recommendations for the best user experience. These things need more records to keep as (a) the time of day you watch, (b) the devices you are using to access the services, and (c) how long you watch or browse.

All of these data are used as inputs that help to process the algorithms of the recommendation system. (An algorithm is a process or set of rules followed in a troubleshooting operation.) The recommendations system does not include demographic information (such as age or gender) as part of the decision-making process.

V. FUTURE WORK AND DISCUSSION

Perhaps the biggest problem faced by referral systems is that they need a lot of data to make recommendations effectively. It is not by chance that the most identified companies with excellent recommendations are those with lots of user data: Google, Amazon, Netflix, Last.fm. As illustrated in the presentation slide of Strands in Recked, a

good recommendation system first needs element data (from a catalog or other form), then it must capture and analyze user data (behavioral events) and then the Magical algorithm does the job. The more elements and user data you have to work with a referral system, the better the chances of getting good recommendations. But it can be a chicken and egg problem: to get good recommendations, you need lots of users, to get lots of data for recommendations.

VI. CONCLUTION

Recommendation systems are a critical way to automate mass customization of e-commerce sites. They will become increasingly important in the future as modern companies increasingly focus on the long-term value of customers for the company. E-commerce sites will work hard to maximize the value of the customer to your site, providing exactly the price and service they believe will create the most valuable relationship with the customer. Because customer retention will be very important to sites, this relationship will generally benefit both the client and the site, but not always. Important ethical challenges will come to balance the value of recommendations to the site and to the client. There are many different techniques for implementing recommendation systems, and different techniques can be used almost regardless of how the recommendation system is intended to increase revenue for the site. Ecommerce sites can first choose a way to increase revenue, then choose the degree of persistence and automation they want, and finally choose a recommendation system technique that fits that profile.

VII. REFERENCES

[1]. Resnick, P., & Varian, H. R. (1997). Recommender systems. Communications of the ACM, 40(3), 56-59.

[2]. Shardanand, U., & Maes, P. (1995, May). Social information filtering: Algorithms for automating" word of mouth". In *Chi*(Vol. 95, pp. 210-217).

[3]. Schafer, J. B., Konstan, J., & Riedl, J. (1999, November). Recommender systems in e-commerce. In *Proceedings of the 1st ACM conference on Electronic commerce* (pp. 158-166). ACM.

[4]. Balabanović, M., & Shoham, Y. (1997). Fab: content-based, collaborative recommendation. *Communications of the ACM*, *40*(3), 66-72.

[5]. Shoham, Y. (1997). Combining content-based and collaborative recommendation. Communications of the ACM.

[6]. Basu, C., Hirsh, H., & Cohen, W. (1998, July). Recommendation as classification: Using social and content-based information in recommendation. In Aaai/iaai (pp. 714-720).

[7]. Beeferman, D., & Berger, A. (2000, August). Agglomerative clustering of a search engine query log. In *KDD* (Vol. 2000, pp. 407-416).

[8]. Deshpande, M., & Karypis, G. (2004). Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)*, *22*(1), 143-177.

[9]. Heckerman, D., Chickering, D. M., Meek, C., Rounthwaite, R., & Kadie, C. (2000). Dependency networks for inference, collaborative filtering, and data visualization. *Journal of Machine Learning Research*, *1*(Oct), 49-75.

[10]. Hay, B., Wets, G., & Vanhoof, K. (2001). Clustering navigation patterns on a website using a sequence alignment method. *Intelligent Techniques for Web Personalization: IJCAI*, 1-6.

[11]. Breese, J. S., Heckerman, D., & Kadie, C. (1998, July). Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence* (pp. 43-52). Morgan Kaufmann Publishers Inc.

[12]. Schafer, J. B., Konstan, J., & Riedl, J. (1999, November). Recommender systems in e-commerce. In Proceedings of the 1st ACM conference on Electronic commerce (pp. 158-166). ACM.

[13]. https://towardsdatascience.com/limitations-of-collaborative-recommender-systems-9801036941b3 accessed on 14th May 2019