Recommender System Based on User’s Preference Transition

Harshada Bharude[1], Ankeeta Girap[2], Pratik Patil[3], Prof.Nidhi R.Sharma[4]
Department of Computer Engineering
[1,2,3,4] Mumbai University

Abstract- *Recommender systems have become an important tool for users to identify interesting items and also for businesses to promote their products to the right users. With development in social networks, travellers have begun to seek recommendations and advise from websites like Trip Advisor. While travellers are willing to share their opinions on social networks, which provides an opportunity for hospitality businesses to learn their customers’ preferences. Given these preferences data, recent advances in machine learning research has made it possible to build automatic recommender systems that can generate hotel recommendations tailored for each traveller.*

Keywords- Recommender Systems, Collaborative Filtering, Hotel Recommendation.

I. INTRODUCTION

Recommendation Systems aim to suggest items like hotels, books, movies, tourism attractions, etc. that are potentially to be liked by users. To identify the appropriate items, recommendation systems use various sources of information like the content of the items and the historical ratings given by the users. These systems were originally designed for users with insufficient personal experience or with limited knowledge on the items. However, with the rapid expansion of Web and e-commerce, overwhelming number of items is offered, and every user can be benefited from recommender systems.Chen and Chuang well-studied the topic Hotel recommendation in a hospitality research (2016). Many of the travellers receive similar recommendations through static methods, like newspapers and television. Due to the advancements in internet hotel recommendation has acquired an interactive form, where travellers can now read recommendations and reviews shared by other travellers on social network, such as Twitter, Trip Advisor. However, in all of these recommendation scenarios, travellers receive the same recommendation without personalization. For example, a traveller with limited budget may still be recommended with an expensive hotel because of its high average rating. Considering there are thousands of hotels in a popular destination, it is impractical for travellers to find out the hotel they really need by simply sorting the hotels via a criterion. Consequently, personalized hotel recommendation is needed to identify a small set of hotels what are potentially to be liked by travellers. Over the last decade there have been rapid advances in RecSys, from both academia and industry (Bennett and Lanning, 2007) numerous recommendation techniques have been proposed to achieve personalized recommendation.

This paper aims in identifying issues presented in hotel recommendation and review its techniques in the context of hospitality.

II. PERSONALIZED HOTEL RECOMMENDATION FOR INDIVIDUALS

Hotel recommendation is not at all a new thing, and is overlapped with hotel selection. Traditionally, the preferences of travellers are unknown or known to a limited extend, thus all travellers receive similar recommendation lists by measuring the overall quality of hotels.

Fortunately, social network has made it possible to get a better understanding of travellers by analysing information they shared on social networks, such as reviews, ratings, profiles, and social connections. With this rich information available the personalized hotel recommendation becomes possible. In this section, we will be reviewing how personalized hotel recommender systems can be built using information shared over social networks.

2.1 Recommendation using Explicit Feedback

Social network websites such as TripAdvisor provide travellers a virtual place to share their opinions on hotels. While statement reviews are possible, ratings are the most preferred format of review. For example, TripAdvisor allows travellers to rate a 1-5 star on the hotel, and optionally to different dimensions of the hotel, such as cleanness, location, and service. Despite of popular star ratings, some websites tend to use formats, such as thumbs up and thumbs down in Facebook.

These kinds of feedback provided by travellers are calling Explicit Feedback, where the travellers explicitly tell us whether they like or dislike the hotel. Generally the explicit feedback-based recommender systems can be categorized into content-based filtering and collaborative filtering.

2.1.1 Content-based Filtering

Content-based methods (Pazzani and Billsus, 2007) which generate recommendations by exploiting regularities in the item content. For example, actors and directors as well as genres can be extracted as content of movies. In the case of hotel recommendation, the content could be location, price, star rating, etc. To make recommendations for a traveller u, we just need to find out which hotels are similar to the hotels the traveller liked before, i.e., highly rated by traveller u. The similarity between two hotels t and t can be computed by popular measures such as Pearson Correlation Coefficient (PCC) and Vector Space Similarity. Despite of its simplicity, content-based methods have limitations. Primarily, it can be difficult to define features or extract content from some hotels. Secondly, travellers will always be recommended with hotel that are highly similar to the hotels he/she liked, which leads to the lacking of diversity (Bradley and Smyth, 2001) and a potentially better hotel may never be recommended.

2.1.2 Collaborative Filtering

Collaborative Filtering methods generate recommendations by analysing preferences provided by travellers, e.g., ratings. One of the most popular and accurate CF method is Matrix Factorization (MF) (Koren, 2009). This approach discovers the latent factor spaces shared between travellers and hotels, where the latent factors can be used to describe both the taste of travellers and the characteristics of hotels. The attractiveness of a hotel to a traveller is then measured by the inner product of their latent feature vector.

2.2 Recommendation using Implicit Feedback

All users are not willing to submit their preferences, where collecting feedbacks inherently delivers a more user-friendly recommender systems. Examples of inherent feedback include the time a user remained on a webpage, location information of users, and the number of clicks a user rendered on an item. The importance of implicit feedback has been accepted recently, and it gives an opportunity to use the vast amount of implicit data that were collected over the years. In this section, we analyse implicit feedback-based recommender systems in the discourse of hotel recommendation.

2.2.1 Relative Preference-based Filtering

A preference relation (PR) denotes user preferences in form of pairwise ordering between items, i.e., are item X is better than item Y? This kind of representation is a useful alternative to explicit ratings as it can be inferred from implicit data. For example, the PR over two Web pages can be inferred by the stayed time, and consequently applies to the displayed hotels.Once the user-wise preferences are computed from implicit feedback, they can be set as input for model-based collaborative filtering methods (Brun, 2010; Desarkar, 2012; Liu, 2015).

2.2.2 Text-based Filtering

Now the first step is to determine the topics from text. Consider an example, a review comments may include a number of sentences, and a method is required to categorise which topic the sentence belongs to. This can be done using easy keywords matching method (Liu, 2013) or improved techniques such as topic models (Mei, 2007).

Once the topics are determined, the second task is to get positivity, negativity, subjectivity opinions from associated text. One of the methods is to look up words or phrases into sentiment dictionaries, like the SentiWordNet. Having the opinions extracted, missing ratings can be filled and a denser dataset is derived for better recommendation performance.

2.3 Evaluation of Hotel Recommender Systems

The evaluation metrics are necessary for building successful recommender systems. Struggle is done to determine the proper way of measuring the quality of recommendations. This section reviews common evaluation metrics for recommender systems.

2.3.1 Accuracy Metrics

Two popular metrics are Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE), which measure the differences between the true preferences and predicted preferences. Let N be the number of unrated items by user, and be the predicted rating of item.

2.3.2 Diversity

Traditionally, the evaluation of recommender systems is mainly supported by accuracy metrics such as RMSE.The accuracy metrics fail to evaluate some properties of the hotels other than the preferences, such as Serendipity (Ge, 2010) and Diversity (Zhou, 2008). For example, a hotel recommendation list should contain both luxury hotels and budget hotels even if a traveller prefers budget hotels in most cases.

2.3.3 Coverage

Coverage refers to the percentage of hotels out of all hotels a recommender systems can recommend. This metric is based on the observation that some hotels may not have the chance to be recommended to any traveller if it is not popular i.e. a new hotel.

A low coverage means the Recommendation System can only make recommendations on a small number of distinct hotels, in other words, it always recommends the popular hotels. Note that Recommendation System with high coverage implies higher diversity (Lüand Liu, 2011).

2.3.4 Stability

Stability measures consistency of recommendations for the same traveller (Adomavicius and Zhang, 2012). The recommendations generated by a stable Recommendation System should be similar after some new preferences are added. For example, the first recommendation of an unstable Recommendation System predicts hotel X as 5-star and hotel Y as 1-star. Then the traveller stayed in hotel X and rated it as 5-star. With this new preference added to the preferences data, an unstable Recommendation System may generate the second recommendation that predicts hotel Y as 5-star. The 5-star hotel Y which was1-star, may lead to user confusion and lower the trust of the Recommendation System. The property of stability was studied in detail in (Adomavicius and Zhang, 2012).

III. PERSONALIZED HOTEL RECOMMENDATION FOR GROUPS

In real-world applications, there are many scenarios where recommendations are made for a group of travellers, such as holiday packages (McCarthy et al, 2007) and touris.promotions (Garcia et al, 2009). Group Recommender Systems (G- Recommendation Systems) focuses on making recommendations that fit the needs of a group of travellers, instead of individuals. In classic Recommendation Systems, the goal is to maximize the satisfaction of a single traveller. However, G- Recommendation Systems need to make trade-off among travellers in the group, where the optimal recommendations that satisfy everyone often do not exist. Recent developments in interactive media and social networks (e.g. interactive TV) have further linked users into groups. (Gartrell, 2010; Vasuki, 2010; Yu,2006;Jameson and Smyth,2007; Masthoff, 2011), and therefore heightened the need for G-RecSys. However, personalized G-RecSys have only been discussed in limited literature comparing to classic Recommendation Systems, and this is particularly true in the context of hospitality. A few survey papers have tried to summarize related works. For example:

(1) The influential survey by Jameson and Smyth (2007) divided group recommendation into four sub-tasks: Group Preference Specification, Group Recommendation Generation, Explaining Recommendations, and Achieving Consensus. Descriptions are given on how existing G- Recommendation Systems handle these tasks.
(2) Carta and Boratto (2010) classified user groups into four types: Established Group, Occasional Group, Random Group, and Automatically Identified Group. Existing G-Recommendation Systems are examined with focuses on how the type of group affects the design of G- Recommendation Systems.
(3) Recently, Masthoff (2011) surveyed techniques used in the Group Recommendation Generation sub-task. Eleven aggregation strategies inspired by Social Choice Theory are summarized with discussions on existing G- Recommendation Systems.

Current G- Recommendation Systems research mainly focus on answering the following three questions:

1) How to collect and represent preferences?
2) How to obtain recommendations by aggregating preferences of individuals?
3) How to explain the recommendations?

3.1 Group Recommendation Generation

Group Recommendation Generation is defined as the process of aggregating group users' preferences and making recommendations based on the aggregated preferences. In spite of preference specification, individual users' preferences have to be aggregated in some way, and identifying the proper aggregation approach has been the main focus in literature (Jameson and Smyth, 2007; Arrow 2012). Generally, there are three approaches to generate group recommendations, and all require preference aggregations (Jameson and Smyth, 2007):

(1) Combining Recommendations for Individuals: In this approach, the classic RecSys will be applied to make recommendations for individuals. The recommendation for a group is then computed by merging the recommended items for each individual in the group. The combining is controlled by a selected aggregation function and in the simplest case the items with highest predicted ratings for individuals are selected.

(2) Assembling Preferences of Individuals: This approach also relies on the individuals' ratings predicted by the classic RecSys. The only difference is that instead of making a list of recommendations for each individual, the ratings for each item is aggregated from preferences of all group users. The group recommendations are made by selecting the items with highest ratings.

(3) Constructing Group Preference Models: This approach does not require predictions of ratings for individual users.The known preferences of individual users are aggregated into a single profile for the whole group. After the completetion of aggregation process, the group looks no different from a normal user, and recommendations are made for this group using classic RecSys.

Basically, G-Recommendation Systems either aggregate preferences of individuals or construct a group preference model. The important advantage of Group Preference Models over preference aggregations is the privacy benefits. When users' preferences are accumulated into a group preference model, the individual user's preferences are hid- den. Afterall, preference aggregation methods can make better recommendations in some cases. Consider an example, items recommended by preference aggregation approaches won't be disliked by all group users, where it is possible, though unlikely, that no group user likes the items recommended by Group Preference Models. No matter which approach is selected, the main task is how to perform aggregation. Most aggregation methods which were discussed in existing surveys are inspired by strategies from Social Choice Theory (Arrow 2012). For example, the Maximizing Average strategy will recommend item that can achieve the highest average rating from group members.Whereas on the other hand, the Minimizing Misery will discard items that are avoided by any group member even if the average rating is high. These kinds of strategies are very intuitive but selecting which one to use is a manual process. The choice of aggregation methods is often left as an open question or very basic ones are used (Amer-Yahia, 2009). However, a lot of established aggregation methods have been developed in communities other than Recommendation Systems and Social Choice Theory, such as Fuzzy Integrals (Beliakov, 2007). These techniques are few of the strong tools to aggregate data,and are often less context dependent.

3.2 Explaining Recommendations

Explaining Recommendations (McSherry, 2005; Knijnenburg et al, 2012) is the responsibility of making the recommendation process more transparent to the users, i.e. why these items are recommended? how confident the recommendations will be liked? For example, a RecSys could make the following explanation (O'Donovan and Smyth, 2005): "the items are recommended to you because they have been successfully recommended to users A, B, and C who are similar to you.Also, we have made X, Y, and Z times recommendations to them in the past, which received P, Q, and R likes". In case of group recommendation, the Explaining Recommendations task refers to make group users fully understand the recommendations. However, the main goal of explanation is not to convince the users about the proposed recommendations, but helping the users to understand other group users' feelings about the recommendations. This process will surely help the group users to adjust the proposed recommendations to arrive a final decision. Unlike classic RSs, debates and negotiations are necessary for group users, and this calls for understanding of not only the pros but also the cons of the proposed recommendations. While existing explanation approaches focus on determining how good the recommendation is for the user, it is now desirable to know how bad the recommendation is for each group user.

3.3 Achieving Consensus

The proposed recommendations can be a good solution but may eventually be rejected by the group. Making the final decision is a complicated process that may involve extensive debate and negotiations. Typical G-Recommendation Systems assume group users are independent and consider each user equally. Technically, G-RecSys is able to identify the recommendations that maximize the overall satisfaction of the group, however, the true maximized satisfaction may not be achieved when interactions exist among group users. For example, while recommending travel destination for a family, the recommended destination may maximize the average satisfaction of all family members.

However, the parents may prefer another destination over their favourites because they care about the children' satisfaction, but on the other hand, the children may not consider their parents' satisfaction too much. In this case, one of the children' favourite destinations that not disliked by the parents may be the final decision.

Ideally, G-Recommendation Systems should take such in-group interactions into consideration, either prior the recommendation generation or make adjustment after received feedback of proposed recommendations. Taking into account the user interactions in recommendation generation has been studied by Amer-Yahia (2009), where a consensus function is defined to maximizing item relevance and minimizing disagreements between group users.

However, modelling complex user interactions remain an unsolved research problem. Alternative way to consider user interactions is to make adjustment by evaluating feedback of proposed recommendations. This type of process is called Reinforcement Learning, and has been applied in context of classic RecSys (Taghipour, 2007; Mahmood and Ricci, 2009)

IV. CONCLUSIONS

This chapter mainly aims to present the start of the art in recommendation systems for the purpose of hotel recommendation. This paper also included recommendation techniques using explicit feedback, such as ratings. We also reviewed some recommendation techniques using implicit feedback, such as clicks and page views, which is gaining popularity in recent years. To figure out recommender systems, we reviewed commonly used metrics, including accuracy metrics, diversity, coverage, and stability. Also we provide a list of free and open source software packages for practitioners to create their own recommender systems.

REFERENCES

[1] Adomavicius, G., & Zhang, J. (2012). Stability of recommendation algorithms. ACM Transactions on Information Systems (TOIS), 30(4), 23.

[2] Amer-Yahia, S., Roy, S. B., Chawlat, A., Das, G., & Yu, C. (2009). Group recommendation: Semantics and efficiency. Proceedings of the VLDB Endowment, 2(1), 754-765.

[3] Arrow, K. J. (2012). Social choice and individual values (Vol. 12). Yale university press.

[4] Baccianella, S., Esuli, A., & Sebastiani, F. (2010, May). SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In LREC (Vol. 10, pp. 2200-2204).

[5] Bennett, J., & Lanning, S. (2007, August). The netflix prize. In Proceedings of KDD cup and workshop (Vol. 2007, p. 35).

[6] Bradley, K., & Smyth, B. (2001). Improving recommendation diversity. In Proceedings of the Twelfth Irish Conference on Artificial Intelligence and Cognitive Science, Maynooth, Ireland (pp. 85-94).

[7] Brun, A., Hamad, A., Buffet, O., & Boyer, A. (2010, September). Towards preference relations in recommender systems. In Preference Learning (PL 2010) ECML/PKDD 2010 Workshop.

[8] Chen, T., & Chuang, Y. H. (2016). Fuzzy and nonlinear programming approach for optimizing the performance of ubiquitous hotel recommendation. Journal of Ambient Intelligence and Humanized Computing, 1-10.

[9] Desarkar, M. S., Saxena, R., & Sarkar, S. (2012, July). Preference relation based matrix factorization for recommender systems. In International Conference on User Modeling, Adaptation, and Personalization (pp. 63-75). Springer Berlin Heidelberg.

[10] Garbers, J., Niemann, M., & Mochol, M. (2006). A personalized hotel selection engine. In Proceedings of the third European Semantic Web Conference.

[11] Garcia, I., Sebastia, L., Onaindia, E., & Guzman, C. (2009, September). A group recommender system for tourist activities. In International Conference on Electronic Commerce and Web Technologies (pp. 26-37). Springer Berlin Heidelberg.