Designing and Developing a Personalised Recommender System

Austin Jose
Computer Science
Durham University
Durham, United Kingdom

***Abstract*—Recommender systems are used to aid users with personalised recommendations of products or services, improving the relationships among customers. The paper explores and implements well-known recommender systems and combines two of them to create a hybrid recommender system. Collaborative filtering (CB) and content-based (CB) are the two recommender systems that will be combined. Some simple algorithms are used to provide simple recommendations for the target user. The paper reviews the results obtained from the implementations and their corresponding ethical issues and limitations.**

***Index Terms*—collaborative filtering, recommender system, content based, cosine similarity, term frequency–inverse document frequency (tfidf)**

I. Introduction

The main objective of what recommender systems aim to accomplish is to provide the user with the most suitable items depending on user history, service/item history, and history on related items/services. For the current assignment, the domain of application for the implementation is within the food industry, as the set of businesses that I have decided to focus on are bars. The purpose of the recommender system is content recommendation. It aims to do content recommendation by allowing the user to be able to pick a bar and be given recommendations based around their favourite bar depending on certain filtering algorithms.

II. Methods

A. Data description

The data set that is being used within the recommender system contains information of all bars in Toronto along with their corresponding reviews. These reviews that have been selected are only reviews that have been written between 01-01-2015 00:00:00 to 01-01-2019 00:00:00. In total, there are 7444 distinct reviews and 189 distinct bars within the data set.

B. Data preparation and feature selection

To begin the data preparation step, we take the reviews.json data set and business.json data set from the yelp data set [1] that is available for academic use. Firstly, I only took businesses that were currently open as I wanted the recommender system to only use live data and not recommend businesses that were no longer open. I then further cut down the data set by filtering it so that only bars were selected and the region used was Toronto. Columns such as hours, is open, review count, latitude, longitude, hours, postal code and address were removed as this was redundant data that would not aid the recommender system in anyway. To further sample the data, I decided to reduce the time span that the reviews written were from 2015 to 2019. After all this preprocessing, the data was cut down considerably but provided much more suitable data.

For feature selection, in the collaborative filtering section of the implementation, the ratings are the main variables that are focused on with the business name/id accompanying this. This is done to ensure that the collaborative filter focuses on finding correlations in the ratings done by the community and whether or not that bar has any connection to it. Whereas the features selected for the content based is done by using intrinsic feature selection. All user reviews for a bar are combined, as well as all reviews done by each user. This is then ran through a intrinsic feature selection which is then put through vectorisers and cosine correlations to find a recommendation.

C. Hybrid scheme

The hybrid scheme that was selected was a mixed scheme that makes use of both content-based and collaborative filtering in order to recommend a suitable bar for the user. Since my recommender is not that complex, it is more suited to a more mixed approach to the hybrid recommender scheme. This is because both recommenders provide some suitable candidates to be recommended and then finally either uses union or intersection to provide the final list of candidates to be recommended. A feature mixed user and item profile are created to find the duality between the two and any correlations the features have when affecting the recommendations.

D. Recommendation techniques/algorithms

The hybrid recommender system uses the combination of both content based and collaborative filtering to mix down possible candidates, which are then used to determine the final set of candidates to be recommended to the user.

For the content based recommender, we used an item based item technique where a bar is recommended based on a particular bar selected by the user. Within content based filtering, a user profile is meant to be generated, which is done by creating a simple bag of words for the user based on all reviews made by each user.A similar profile is created for each bar in which we can use this to compare and correlate to find similar bars using feature extraction, Term Frequency-Inverse Document Frequency (TFIDF) vectors and cosine similarities.

For collaborative filtering, and even more simple technique is used due to time constraints. The algorithm used was simply finding the mean rating of each business based on all reviewed users. This is then used to determine the correlation between different bars by creating a user-item interaction matrix. The recommended candidates are then found by retrieving the 10 most similar bars to the user selection from the matrix.

III. Implementation

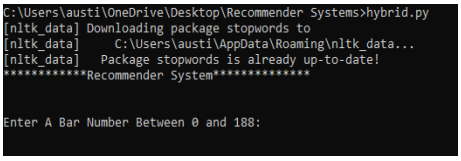

Fig. 1. An image of the input interface

As seen in (1), the user interface is simple and meant to make it so that it is not overly complicated. The user data that is gathered is explicit data that targets the user's interests. This would be asking the user to enter a number that indexes a particular bar name. This would however mean that the user itself cannot be tracked and the user himself does not know how the data is being manipulated in the background to provide the recommendations.

A. Recommendation algorithm

For the first recommender system, a content based recommender system is implemented based on [2] and [3]. Within this implementation, bars are suggested depending on a particular bar which is based on the metadata that is being used. In this case, the reviews are the metadata that is being used to judge similar bars and their correlations. Using the processed data for this section which is the combined data set containing all reviews for each person and the combined reviews for each place, we use word vectorisation to solve the natural language processing problem. This is used to extract features from the reviews that carry a semantic meaning which can be then used to check for correlations with other bars. For this vectorisation, we compute the Term Frequency-Inverse Document Frequency (TFIDF) vectors for each business and their reviews. This results in a matrix where each column represents a word in the overview vocabulary as well as each column represents a bar business. The TFIDF score in overview is the score of the frequency of a word in a document in this case, it would be the frequency that word appears in the combination of all reviews for each business. This is then down weighed by the number of times this word occurs in the other businesses. This is used to reduce the importance of frequently occurring words which was somewhat cleaned up using nltk's stop words and the algorithm provided in [3]. Cosine correlation is then used to determine the the candidates for recommendation.

For collaborative filtering, an item based recommender system is implemented based on [4]which uses a simple memory based approach as opposed to a model based approach. The implementation begins by creating a new data frame by grouping up each business with their corresponding average rating done by the users. The number of ratings that have been made for each business is then calculated and added to the data frame. A matrix is then created that holds the user-item interactions by producing a pivot table holding the user id, business name and the rating made by the user. The selected bar made by the user is then compared with all other bars to check for correlations with ratings. This correlation matrix displayed the bars that were most in common with the selected bar by looking at the overall ratings made by the entire community. The 10 bars that have the highest correlation then become candidates for intersection or union at the end.

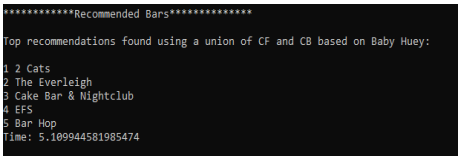

Fig. 2. An image of the output interface

The output (2) is similar in regard to the input in how simple the user interface looks. From research it has been highlighted that 72% of recommenders only show between 3 and 5 recommendations. This is due to the fact that displaying more recommenders may lead to overwhelming the user with lots of information, whereas putting too few recommendations would not keep the user interested enough to explore the other recommendations [5]. Since the recommender is quite generic, it does not take into consideration the geographic location of the user, how old the user is or their environment. Instead it provides a generic recommendation for all users who like a specific bar.

The specific set of recommendations that are outputted is due to either the union or intersection of the candidates present from the content based and collaborative filtering results. If the set of intersection has a length of 5 or more, then the intersection set is presented to the user. Whereas if the number of bars that are similar between both candidate sets are smaller than 5, then the union of the candidate sets are used to ensure that there is always 5 recommendations presented to the user in the interface.

IV. Evaluation results

The evaluation of the overall recommender was not able to be done, but instead, I have looked into the matrices and distributions of the data itself.

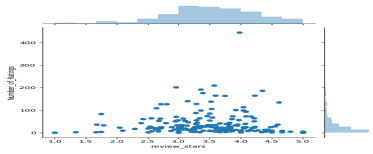

Fig. 3. A graph showing the distribution between the number of ratings and the ratings

n (3), the distribution of the number of ratings and ratings are shown that were used within collaborative filtering. From the figure we can see that a lot of bars have not been rated at all. As the number of ratings increase, the fewer number of of people are rating the bars. The majority of ratings seem to be positive, as seen in the histogram at the top of the chart. The density seems to lie around 3-4 rating while there are only a few ratings that have been given 1. Less number of users have rated these particular bars.

In general, it can said that comparing this implemented hybrid recommender system compared to a well known base recommender system such as the collaborative filtering recommender system, mine would perform quite poorly as it does not fully make use of user-profiles nor does it train or test any data sets to predict ratings.

An ethical issue that is present within the developed recommender system is that there is a privacy risk when inferring data. This is because in the background, user and item profiles are being created, but the recommender does not tell the user themselves. There is no consent from the user to allow the recommender to take in data made by the user. In order to combat this, the recommender could clearly state that the user's data will be used and if the user consents then it continues otherwise the program can be exited and no user profile gets created. This is especially the case when new users are being brought into the system.

A. Ethical issues

Another ethical issue that can be taken into consideration is the fact that personal information can be leaked and a person may be able to be pinpointed. Since the data set that has been used is within a time span and in one location, a malicious attacker could use this to figure out where a user might be. A way in which we combat this is by letting the user control the filters used within the recommender. An example would be that the user can determine which cities are present in the data set and the time span of the reviews as well. If need be, it might be worth having an option to allow the user to hide certain reviews. To minimise the risk of leaks and exploitation, user data could be encrypted and stored in a decentralised database. This leads to better transparency between the system and the user; however, drawbacks are present which include, reducing the accuracy of the recommendations or even shifting the blame more onto the user rather than the system.

Lastly, biases and behaviour manipulation is an ethical concern. Since the RS uses reviews made by the community, there could be significant bias when recommending certain bars. In addition to this, the recommender system itself is not very accurate and could lead to a negative experience if the user goes to a bar that has been falsely recommended. A way this could be fixed would be to try and expand the user profile more by tracking cookies if done on a website or even looking at their social networks to provide a more accurate recommendation based on previous interests and history.

V. Conclusion

The sparsity problem is one of the main limitations present within CF as there is a large amount of bars present within the data, but not all users review and rate each and every bar. There might be some bars that have many ratings but others that barely have any. This can be seen in (3) when looking at the density distribution of the ratings. This problem is also present in the hybrid model, as CB also doesn't have many reviews to base the correlation off of. Another common limitation within CF is the scaling problem, a huge amount of data is needed to predict and display accurate recommendation, but this requires significant computational power and large data sets. With my current computational power, it is impossible to process large amounts of data and therefore have needed to cut down the data set considerably.

Future improvements that could be made to the recommender system is to employ a user based CF rather than a item based as it is found to be much better [7]. To further improve the CF, an algorithm based on the Markov decision processes model (MDP), which is briefly mentioned in [8] can be used. Not only this but [9] explores the advantages of using SVD and demographic data to further improve the system. You could take into consideration the user's geolocation to make more accurate predictions. It allows the system to have a strong initial model as well as a type of memory less model which saves memory in the long term. To further improve the CD side of the hybrid recommender, it would be an improvement to further consider the user profile and not only the item profile. This allows more accurate recommendations to take place depending on user interest and history.

References

[1] Yelp dataset
Source: https://www.yelp.com/dataset
Visited: 07/03/2021

[2] Content based recommenders
Source: https://www.datacamp.com/community/tutorials/recommend
Visited: 07/03/2021

[3] Restaurant recommender
Source: https://github.com/gann0001/Restaurant-Recommendation-
Visited: 07/03/2021

[4] Item based collaborative filtering
Source: https://www.youtube.com/watch?v=$_2nES58GEHM$
$Visited 06/03/2021$

[5] Felix Beierle, Akiko Aizawa, and Joeran Beel. Exploring choice overload in related-article recommendations in digital libraries. arXiv preprint arXiv:1704.00393, 2017

[6] Schafer, J.B., Frankowski, D., Herlocker, J. and Sen, S., 2007. Collaborative filtering recommender systems. In The adaptive web (pp. 291-324). Springer, Berlin, Heidelberg.

[7] Peter Bostr¨om and Melker Filipsson. Comparison of user based and item based collaborative filtering recommendation services, 2017.

[8] Thorat, P.B., Goudar, R.M. and Barve, S., 2015. Survey on collaborative filtering, content-based filtering and hybrid recommendation system. International Journal of Computer Applications, 110(4), pp.31-36.

[9] Polat, H. and Du, W., 2005, March. SVD-based collaborative filtering with privacy. In Proceedings of the 2005 ACM symposium on Applied computing (pp. 791-795).