

Original Research Paper

A Conceptual Model for the E-Commerce Application Recommendation Framework using Exploratory Search

¹Mohammed Mahmudur Rahman, ¹Zulkifly Mohd Zaki,
¹Najwa Hayaati Binti Mohd Alwi and ²Md. Monirul Islam

¹Faculty of Science and Technology, University Sains Islam Malaysia (USIM), 71800 Nilai, Negeri Sembilan, Malaysia

²Faculty of Science and Engineering, International Islamic University Chittagong (IIUC), Kumira, Chittagong-4318, Bangladesh

Article history

Received: 13-06-2020

Revised: 29-06-2020

Accepted: 27-08-2020

Corresponding Author:

Mohammed Mahmudur
Rahman

Faculty of Science and
Technology, University Sains
Islam Malaysia (USIM), 71800
Nilai, Negeri Sembilan,
Malaysia

Email: provaiiuc@raudah.usim.edu.my

Abstract: The users helped by the search engine for online transactions in e-commerce; nonetheless, there is still a lack of search interest from the user and online shopping intentions. To boost the user's search for product recommendations, use a search engine for the quest but not for purchasing purposes. Existing frameworks have some significant problems in the recommendation technology, such as new product problems, fewer evaluation problems, a vast amount of data, etc. Many e-commerce applications lack a better search experience like an Exploratory search system. This research work aims to create a conceptual model for recommendation systems using exploratory search, to study the behavior of users and the efficacy of exploratory search in terms of the quality of the search results produced. Several machine learning algorithms are used in this research to classify e-commerce products and evaluate the performance of these algorithms. The full-text search mechanism is used to implement an exploratory search system. The exploratory search system is evaluated by three criteria called to look up, learn and investigate. After the experiments and evaluation, it is observed that AdaBoost over decision tree performs better than the other classification algorithms implemented. The exploratory search system satisfies the three requirements that are lookup learn and investigate during the search process. Contributions to this work are to build a conceptual model for the recommendation system through the dataset of user events and finding the implementation of an exploratory search mechanism in an e-commerce application.

Keywords: Recommender System, Exploratory Search, Machine Learning, Full Text Search

Introduction

Recommend a better product and improving the search experience on the e-commerce application to the end users, can be more beneficial for the companies. It showed that, in e-commerce applications, there is a lack of a strong recommender program with a better search experience. Choosing a product over the internet is simple, with the aid of E-commerce applications. However, many items coming from many websites, it is difficult to say which would match individual interests and this inquiry usually relates to the recommendation of a product offered to a particular customer. The exhibition of rating based collaboration filtering system of an individual product

might not be sound because of less evaluating recurrence. Also, it is critical to show recently enlisted products having no rating. The less evaluating recurrence problem and the problem of the new product are primary difficulties for a suggestion to a user or customer. In this study, a generalized recommendation system technique is proposed for e-commerce application using the exploratory search. The main contributions to this research is to formulate the product quality measurement by user events dataset and finding implementation of exploratory search process. A Hotel Recommender System website is planned to be built which will recommend hotels by the hotel groups and will facilitates the users to search for the best hotels by the exploratory search engine.

Literature Review

Some essential information about machine learning, classification algorithms that will be implemented in work is incorporated, which will be helpful for detail investigation afterward. Most of the work analyses and describes the different kind of recommender framework and popular recommendation algorithms and its implementations. Methods that give recommendation dependent on the client's taste to find suitable item for them by separating customized data dependent on the client's inclinations from a large amount of data (Das *et al.*, 2017). They use Content-based filtering, Collaborative filtering, Demographic filtering, Knowledge-Based and Hybrid recommender system. In this study (Jiang *et al.*, 2019) focused on the issue of low accuracy of the traditional slope one algorithm and the untreated ratings in recommender frameworks. Bring trust into the recommendation system, calculate the client similarity to characterize trust metrics. A limitation of their work is the new item issue where recommendations are required for things that nobody has yet rated.

Another work (Hwangbo *et al.*, 2018), introduces a suitable recommendation system executed in a considerable manner fashion organization that sells fashion items through both online and offline shopping centres'. Client conduct data and item data to distinguish client inclinations and proactively recommend things that they are probably going to purchase where collaborative filtering is used. The tests are limited in that they can't clarify the impacts of the utilization of the online-offline data and the impacts of inclination decay after some time. Customized hotel suggestions to their clients and construct a machine learning model to anticipate the booking result for a client occasion (Shenoy *et al.*, 2017), because of their search and different characteristics related to that client occasion. The arbitrariness actuated by real-world data makes it challenging to find a shape and need additional improvement.

In this study (Isinkaye *et al.*, 2015), talked about the two traditional recommendation systems and featured their qualities and difficulties with various sort of hybridization procedures used to improve their activities. Information filtering frameworks that manage the issue of data over-burden by filtering imperative data section out of enormous measure of powerfully produced data as per client's inclinations, intrigue, or watched conduct about thing. They use Content-based filtering, Collaborative filtering (Zhang *et al.*, 2015), Hybrid filtering and the limitation is new item problem. A comparative study results of an exploratory laboratory-based study comparing user behaviour in Amazon, which offers non-personalized recommendations and WorldCat.org which does not, Interactive Information Retrieval is used (Wakeling *et al.*, 2014). Hotel suggestion using collaborative filtering and rankboost

algorithm to ignore new item issue. Vast amount of data creating a user-item rating matrix, searched the same interest and isolate them into a cluster using similarity equation. The data burden become considerably more genuine problem, different markers about the amusement of product which ought to be considered in picking group focuses (Huming and Weili, 2010).

Distinguished and characterized the qualities of exploratory search and utilized them as information looking model assessment framework (Palagi *et al.*, 2017). A relative study with respect to qualities and models of exploratory search proposing criteria and techniques for the structure and assessment of exploratory web crawlers. Definition of Exploratory is yet unclear and poorly-defined (Athukorala *et al.*, 2016). The most particular markers that describe exploratory search practices are query length, maximum scroll depth and task fulfilment time. In the event of exploratory search task, used algorithms can expand the level of investigation to retrieve progressively various points and in the lookup task, they can expand the degree of exploitation to retrieve smaller outcomes. An exploratory system (Fafalios and Tzitzikas, 2014), proficient search depends on semantic post-investigation and showing the top results for a keyword, using pproc (post processor) to categorize the keywords and ranks the metadata.

The idea of exploratory search (Marie *et al.*, 2014; Jiang, 2014), showing essential hypothetical establishments, explaining unpredictable significance from the parts of the issue setting and the search procedure. Anticipating the improvement patterns of the exploratory search field by the social idea of data chasing using Hierarchical classification, Faceted classification, Dynamic clustering, social classification. This research on exploratory search has been concentrating on person clients' search operation, overlooking the social contribution to data exploration. This paper (Ruotsalo *et al.*, 2013) implemented Interactive Intent Model to perform the exploratory search. This model takes interactions and feedback, learns from them, predict future search intent and it is not a replacement of the query-typing interaction. A normal idea of exploratory search framework, extracting information, creating an entity-relationship schema (Bozzon *et al.*, 2013). Search parameters are not easy to use. A model framework (Singh *et al.*, 2013; Palagi *et al.*, 2017) creating experiential client information paradigm Queries. A novel description connection paradigm for exploratory web search which permits synchronous and semantically associated description of inquiry results from various semantic points of view. They are relied upon to be of pertinence to a wide class of issues in data retrieval including diverse heterogeneous information and complex data needs. A research tools (Marchionini, 2006; Ariyanto *et al.*, 2019) basic for exploratory search achievement include the formulation of new interfaces that move the process above predictable certainty retrieval.

There are more opportunities for mining client personal conduct standards and applying antagonistic enumerating that attempts to exploit framework and client practices.

Proposed model

In this research, the objective is to create a conceptual model for the e-commerce recommender system based on user interaction data and make an exploratory search engine to search the products by the query to the website. The product contains product group which needs to classify by feeding the user interaction data to the trained algorithm model.

Recommender System Framework

The proposed recommender system is done by three main steps. The first phase includes interaction data collection to the system database. The next phase includes, training the algorithm with the user interaction data and the last phase is, prediction of hotel groups by feeding appropriate interactions data into the classification model.

User Interaction

The user interaction model shows in Fig. 1 the interaction with the system and system response with respect to the user individual request. The user interaction data is collected from the user events. The event may a click or a purchase of the product. Tracking of the user events is necessary for recommendation. The interaction data will be feed into the classification model to predict the product groups. The user interaction contains 11 main features. These key features are necessary for the classification. There could be other feature as well which depends on the e-commerce application domain.

Algorithm

To predict the product groups by the user interaction data, first, the classification algorithm needs to be trained. Some classification algorithms are chosen to classify the product groups. The best classification algorithm will be chosen after the classification performance evaluation.

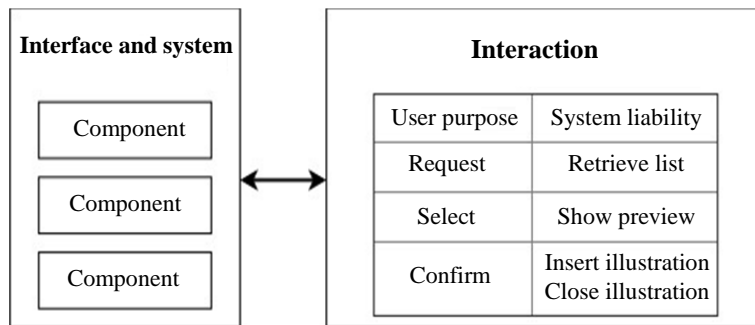


Fig. 1: User interaction model

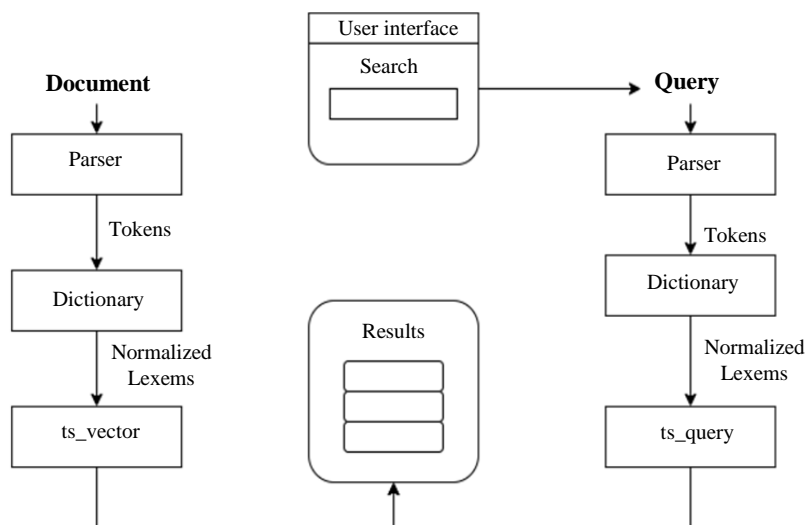


Fig. 2: Full text search

Product Group Prediction

The product group prediction will be done by feeding the correct user interaction data into the trained model. Choosing the correct interaction data is important. There are two ways to retrieve the data from the user interaction database. The first way is when search occurs on the website. In that case, the user interaction data retrieval will be based on the search results. The search result products features will have impact on the recommendation of the products. The another one is, when the click event occurs. In that case, the recommendation will be based on the clicked item category or other users' same interactions on the same product or based on the nearby location where the click event occurred. These decisions will be varying from different e-commerce application domain. The decisions will be decided by the e-commerce application decision makers.

Exploratory Search Framework

In this research work, the exploratory search is implemented by full text search mechanism as shown in Fig. 2. For the implementation, the PostgreSQL database has been used which supports full text search mechanism. Full text search facilitates two major functions for searching on the database.

The functions are *ts vector* and *ts query*:

- 1) *ts vector*: Creates a list of tokens of the query text.
- 2) *ts query*: Query the vector for the occurrence of certain words or phrases.

Here are some features of Full Text Search:

- 1) Stemming
- 2) Ranking
- 3) Stop-words removal
- 4) Multiple language support
- 5) Accent support
- 6) Indexing
- 7) Phrase search

The named entities of the product data help to acquire the search results based on the search query. For this research implementation, the named entity columns of hotel data are used to obtain search results:

- 1) Accommodation type: Accommodation type of the Hotel
- 2) Hotel name: Hotel/Resort name
- 3) Address: Hotel detailed address
- 4) District: Hotel district
- 5) Region: Hotel region
- 6) Country: Hotel country
- 7) Star rating: Rating of the hotel
- 8) Guest rating: Guest Rating of the hotel

9) Review badge: Review badge of the hotel

The full text search helps to acquire the exploratory search mechanism to this research implementation. The Asynchronous JavaScript and XML (AJAX) functionality gives instant search result to the application for every word input and without reloading the application. The search results then help the users to obtain a clearer idea about the query. From these result the user can interpret and compare that what he is looking for. Also, By the instant search result the user will have a clear idea about the search query. Thanks to the AJAX for this features which helps the user to learn about the products and gives the user an idea about the query. There is three main phase of exploratory search. These are: Lookup, Learn and Investigate. The lookup phase will be handled by the full text search. Then, the instant search result will help the user learn and investigate about the products and the user will have an idea about the query.

System Workflow

The system workflow indicates how the overall recommender system and exploratory search work together in an implemented system. From the Fig. 3, there are two individual interactions from the system. One is by the search query which is referred by blue arrow signs and the other is the events by the users which is referred by the black arrow signs.

When a user searches on the system, the search results of the products come with the product recommendations based on the following search results which help the users to choose the product by the recommendation also. The user interactions data retrieval from the search result may vary depending on the search or the events that occurred by the user.

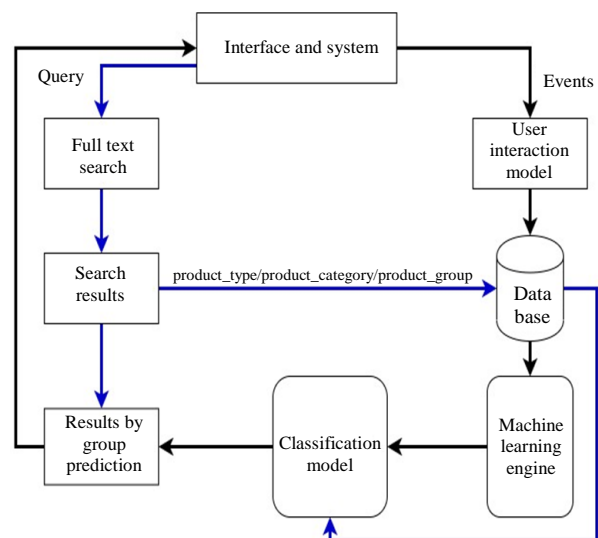


Fig. 3: System Workflow

Experimental Analysis

According to this research, the category of the e-commerce application product data is divided into two parts. One is the product data which has the necessary details about products including product name, product price etc. Another one is the user events data. The user events data consist of some features which represents how the user interacted to the website. There should be some features of the product which helps to formulate the product group. The product group plays an important rule to recommend products to the consumers by the product group classification. This research implementation will be based on the hotel booking application. The Expedia provides the user interaction data which helps us to do the experiment and performance evaluation on those data.

Dataset Collection

For this research purpose, the user interaction data are used. For the experiment of this research, the dataset should contain user events. The user interaction dataset acquired from the Kaggle website which is provided by the Expedia. Expedia provided logs of customer behaviour. These include what customers searched for, how they interacted to the Expedia website. The train dataset contains around 300 million record with 24 different columns and the test dataset contains around 75 million records. These data represent the user events to the website. For this research implementation and experiment purpose, all of the 24 columns are not necessary. After removing some of the columns, the experiment accuracies and the performance of the classification grow. Also, after removing some of the columns the dataset still represents as the user events data in Table 1. The Expedia dataset consist of null data

and the garbage values. For this reason, the dataset needs to be cleaned before the experiment. The proposed dataset is based on the concept of the Expedia user events dataset. The product dataset may vary depends on the different e-commerce application domain. The product dataset should have the seller and consumer rating for the product group formulation.

Data Preprocessing

The data preprocessing is all about transform the raw data in a useful and efficient format. There are some columns in Expedia dataset which will not use in this experiment. These data columns are, site name, posa continent, is mobile, is package, channel, srch destination type id, hotel continent and cnt. Without these columns, the dataset still represents user events well. The null and the garbage value will be removed from the dataset. The duplicate columns will be removed. The columns which contains the timestamp as the data these columns divided by the month and the year. After cleaning, there is 1.9 million interactions data found. The dataset columns are mentioned below after cleaning.

Exploratory Search Evaluation

The full text search is implemented in the website called <http://www.recsys.xyz> to acquire the exploratory search mechanism to this research implementation. As it was explained before that the Asynchronous JavaScript and XML (AJAX) functionality gives instant search result to the application for every word input and without reloading the application. When a user searches on the system, the search results of the products come with the product recommendations based on the following search results which help the users to choose the product by the recommendation also.

Table 1: Data fields after cleaning of Expedia user events dataset

| Field name | Field description | Field data-type |
|---------------------------|--|-----------------|
| Date time year | The year | int |
| Data time month | The month | int |
| Hotel country | Hotel country | int |
| Is booking | 1 if a booking, 0 if a click | int |
| Orig destination distance | Physical distance between a hotel and float a customer at the time of the search | float |
| Srch adults cnt | The number of adults specified in the hotel room | int |
| Srch children cnt | The number of children specified in the hotel room | int |
| Srch ci year | Check-in date | int |
| Srch ci month | Check-in date | int |
| Srch co year | Check-out date | int |
| Srch co month | Check-out date | int |
| Srch destination id | ID of the destination where the hotel search was performed | int |
| Srch rm cnt | The number of hotel rooms spec-ified | int |
| User id | The number of hotel rooms spec-ified | int |
| User location city | The ID of the city the customer is located | int |
| User location country | The ID of the country the customer is located | int |
| User location region | The ID of the region the customer is located | int |
| Hotel cluster | ID of a hotel cluster | int |

Lookup:

1. Known item search: The hotel data exist in the database. The search results are fetched from the database
2. Navigation: The search results are dependent on the search query. The search results navigate the user to surf specific products
3. Transaction: The transaction happens whenever a user types letters in the search box

Learn:

1. Knowledge acquisition: After getting the idea from the ajax search results, the user gets about what to search next to get the desired result. The recommendations are based on the search results interaction from the database
2. Interpretation: The interpretation of search results recommendations based on similar search interactions occurred before

Investigation:

1. Analysis: The analysis will be done by the user when the ajax search happens.
2. Evaluation: The evaluation will be done by sorting the highest recommended products.
3. Discovery: The discovery of the desired product will be chosen by the user based on the recommendation and the search results.

Performance Evaluation

In this research, the classification algorithms are chosen for the experiment and evaluation. The user interaction data consist of 100 hotel group. So, to classify the hotel groups the classification algorithms will be suitable. The main goal is to measure different algorithms performance and finally select the best one. Each of the algorithm performance evaluation includes Accuracy measurement, graph plot by the accuracies found by running the algorithm, classification report and ROC curve. These performance evaluation techniques help us to choose the best classifier. For the implementation of classification algorithm, the scikit-learn library is chosen. This library provides the machine learning classification algorithms, performance evaluation methods. To run the algorithms, perform analysis and capture the results the Kaggle kernel is used. The Kaggle kernel specifications consist of 4 CPU Cores, 16 Gigabytes of RAM and 5 Gigabytes of auto saved disk space. For the implementation and analysis of this research, 5 product groups of size(s) 5, 10, 20, 50 and 100 are created out of the original dataset.

Table 2: Tree depth vs accuracy comparison of Adaboost over decision tree classifier

| Tree depth | Accuracy |
|---------------|----------|
| Group size 5 | |
| 3 | 46.40% |
| 5 | 51.97% |
| 10 | 65.17% |
| Group size 10 | |
| 3 | 28.71% |
| 5 | 32.48% |
| 10 | 43.86% |

AdaBoost Over Decision Tree Classifier

Description

Adaptive Boosting with the Decision Tree Classifier would help to acquire better accuracy level. Essentially, it sets the weights of the classifiers and train the data sample in each iteration to ensure accurate predictions of unusual observations. The results after applying AdaBoost over Decision Tree Classifier are mentioned in Table 2.

Graph Plot

After boosting with the AdaBoost over the Decision Tree Classifier the accuracy grows in Fig. 4. It is because, the AdaBoost Classifier set weights of the classifiers and train the data sample in each iteration to ensure accurate predictions. The AdaBoost over the classifiers algorithm enhance the accuracy level and gives better result.

Classification Report

The AdaBoost over the Decision Tree Classifier classification report was measured for the 10 hotel groups in Fig. 5. Here, for the Group 2 has the precision score 0.50 out of 1.00 which indicates that the proportion of the correctly predicted of Group 2 interactions out of all predicted Group 2 interactions by the classification. The recall score of Group 2 is 0.48 out of 1.00 which indicates that the proportion of the correctly predicted of Group 2 interactions out of the number of Group 2 interactions by the classification. The F1 score for the Group 2 is 0.49 out of 1.00. After boosting, the precision, recall and F1 score improves a lot compared to the other previously evaluated algorithms. The lowest precision, recall and F1-score is 0.33, 0.32 and 0.32 which is considerable. The more the F1 score will be, the classification model will be assumed better. The macro average for the precision, recall and F1 score is 0.44, 0.44 and 0.44 out of 1.00. The weighted average for the precision, recall and F1 score is also the same. Here, the support value for each hotel group indicates the number of samples of the true response of that class.

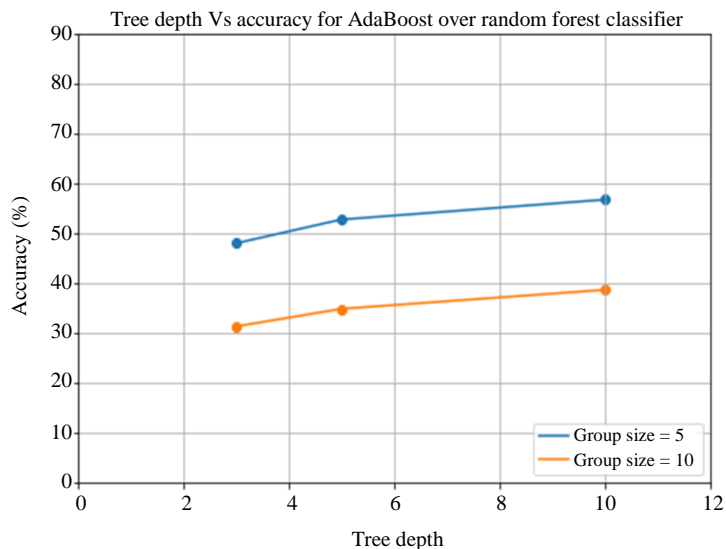


Fig. 4: Tree depth vs Accuracy of AdaBoost over Decision Tree Classifier

| | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 2 | 0.50 | 0.48 | 0.49 | 4892 |
| 4 | 0.44 | 0.45 | 0.45 | 5367 |
| 5 | 0.41 | 0.39 | 0.40 | 4809 |
| 7 | 0.47 | 0.58 | 0.52 | 5217 |
| 15 | 0.51 | 0.46 | 0.49 | 5094 |
| 19 | 0.42 | 0.43 | 0.43 | 4927 |
| 51 | 0.39 | 0.35 | 0.37 | 4603 |
| 55 | 0.42 | 0.39 | 0.40 | 5209 |
| 70 | 0.47 | 0.51 | 0.49 | 5417 |
| 83 | 0.33 | 0.32 | 0.32 | 4805 |
| accuracy | | | 0.44 | 50340 |
| macro avg | 0.44 | 0.44 | 0.44 | 50340 |
| weighted avg | 0.44 | 0.44 | 0.44 | 50340 |

Fig. 5: Classification report of AdaBoost over decision tree classifier

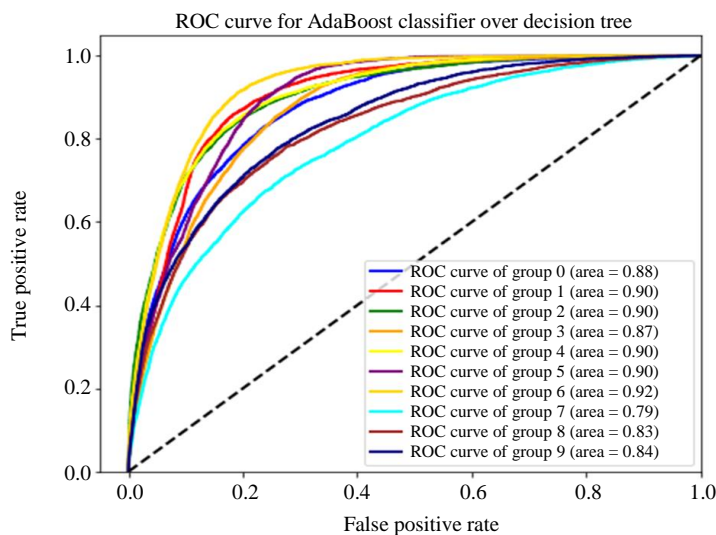


Fig. 6: ROC curve of AdaBoost over decision tree classifier

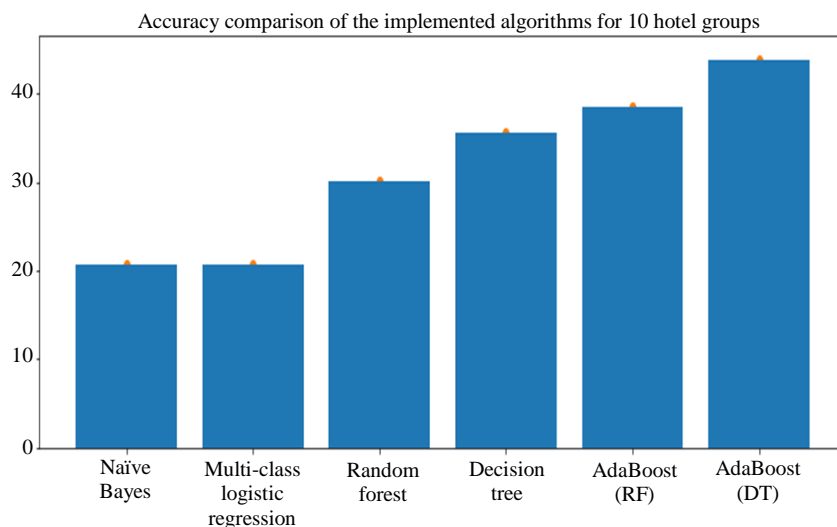


Fig. 7: Accuracy comparison of the implemented algorithms for 10 hotel groups

ROC Curve

Now, from the above ROC curve observation in Fig. 6, the lowest Area under Curve (AUC) of the ROC curve is 79 and the highest AUC of the ROC curve is 92. It seems that, after applying AdaBoost over the Decision Tree Classifier the model performs better and ensure accurate predictions of different groups. It is assumed that the more AUC of the ROC, the model is assumed to be better.

Analysis

The AdaBoost over the Decision Tree Classifier performs better. The classification of different classes is good. So, from the above observations, the AdaBoost over Decision Tree fits best with the current dataset. The recommender system will be implemented based on the AdaBoost over Decision Tree Classification model.

The chart mentioned in the Fig. 7 represents the accuracy difference of the implemented algorithms. These are the highest accuracies found for the 10 hotel groups. The bar chart shows that the AdaBoost over Decision Tree Classifier performs better than other implemented algorithms because of the error are minimized by the boosting approach. The misclassified product is classified rightly by AdaBoost classifier. For this reason, the accuracy is high compared to other implemented algorithms.

Conclusion

In this study machine learning algorithm is implemented and evaluated. It is noticed that the accuracy is higher when the tree depth is high. Decision tree classifier is boosted using AdaBoost classifier. For exploratory search, look up, learn and investigate these three criteria is fulfilled successfully during the exploratory search process. In this

research, a conceptual model for recommender system and exploratory search is given. The product group formulation has been done which has an important role to recommend products. There is a huge chance of future research on recommender system and exploratory search model. In this research work, the implementation has been done using user interaction data with different features. Extracting more features from the interactions data will contribute to improve the recommendation system. Also, the more the real user data can be gathered the classification model will be good to classify the product groups. Finding more features and adding them in the dataset will bring a significant change in the results of the product groups prediction. For the exploratory search analysis, other efficient search frameworks can be used to give the better search result. Also, giving more effective search result to the users can be beneficial. Beside this, answering query questions is another important feature to implement. In this research work, the dataset is trained with several machine learning algorithms. In future new algorithms can be implementing for better accuracy and results. For exploratory search process there is a feature can be included which is how to answer the question about the search query.

Acknowledgment

All the authors contributed to conception, drafting and critical proof reading of this manuscript.

Author's Contributions

Mohammed Mahmudur Rahman: Performed the computations, designed the model, the computational framework and analysed the data, drafted the manuscript.

Zulkifly Mohd Zaki: Developed the theory, conceived and planned the experiments, to investigate and supervised the findings of this work.

Najwa Hayaati Binti Mohd Alwi: Performed the analysis, to investigate and supervised the findings of this work.

Md. Monirul Islam: Verified the analytical methods, conceived the idea and supervised.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

- Ariyanto, D., Nugroho, L. E., & Permatasari, A. E. (2019, December). The framework accommodation of systems recommendation via social media. In *Journal of Physics: Conference Series* (Vol. 1339, No. 1, p. 012097). IOP Publishing.
- Athukorala, K., Głowacka, D., Jacucci, G., Oulasvirta, A., & Vreeken, J. (2016). Is exploratory search different? A comparison of information search behavior for exploratory and lookup tasks. *Journal of the Association for Information Science and Technology*, 67(11), 2635-2651.
- Bozzon, A., Brambilla, M., Ceri, S., & Mazza, D. (2013). Exploratory search framework for Web data sources. *The VLDB Journal*, 22(5), 641-663.
- Das, D., Sahoo, L., & Datta, S. (2017). A survey on recommendation system. *International Journal of Computer Applications*, 160(7).
- Fafalios, P., & Tzitzikas, Y. (2014). Exploratory professional search through semantic post-analysis of search results. In *Professional search in the modern world* (pp. 166-192). Springer, Cham.
- Huming, G., & Weili, L. (2010, April). A hotel recommendation system based on collaborative filtering and rankboost algorithm. In *2010 Second international conference on multimedia and information technology* (Vol. 1, pp. 317-320). IEEE.
- Hwangbo, H., Kim, Y. S., & Cha, K. J. (2018). Recommendation system development for fashion retail e-commerce. *Electronic Commerce Research and Applications*, 28, 94-101.
- Isinkaye, F. O., Folajimi, Y. O., & Ojokoh, B. A. (2015). Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal*, 16(3), 261-273..
- Jiang, L., Cheng, Y., Yang, L., Li, J., Yan, H., & Wang, X. (2019). A trust-based collaborative filtering algorithm for E-commerce recommendation system. *Journal of Ambient Intelligence and Humanized Computing*, 10(8), 3023-3034.
- Jiang, T. (2014). Exploratory search: a critical analysis of the theoretical foundations, system features and research trends. In *Library and Information Sciences* (pp. 79-103). Springer, Berlin, Heidelberg.
- Marchionini, G. (2006). Exploratory search: from finding to understanding. *Communications of the ACM*, 49(4), 41-46.
- Marie, N., Gandon, F., Giboin, A., & Palagi, É. (2014, September). Exploratory search on topics through different perspectives with DBpedia. In *Proceedings of the 10th International Conference on Semantic Systems* (pp. 45-52).
- Palagi, E., Gandon, F., Giboin, A., & Troncy, R. (2017, March). A survey of definitions and models of exploratory search. In *Proceedings of the 2017 ACM Workshop on Exploratory Search and Interactive Data Analytics* (pp. 3-8).
- Ruotsalo, T., Peltonen, J., Eugster, M., Głowacka, D., Konyushkova, K., Athukorala, K., ..., & Kaski, S. (2013, October). Directing exploratory search with interactive intent modeling. In *Proceedings of the 22nd ACM international conference on Information and Knowledge Management* (pp. 1759-1764).
- Shenoy, G. G., Wagle, M. A., & Shaikh, A. (2017). Kaggle competition: Expedia hotel recommendations. arXiv preprint arXiv:1703.02915.
- Singh, R., Hsu, Y. W., & Moon, N. (2013). Multiple perspective interactive search: a paradigm for exploratory search and information retrieval on the web. *Multimedia tools and applications*, 62(2), 507-543.
- Wakeling, S., Clough, P., & Sen, B. (2014, August). Investigating the potential impact of non-personalized recommendations in the OPAC: Amazon vs. WorldCat. org. In *Proceedings of the 5th Information Interaction in Context Symposium* (pp. 96-105).
- Zhang, K., Wang, K., Wang, X., Jin, C., & Zhou, A. (2015, April). Hotel recommendation based on user preference analysis. In *2015 31st IEEE International Conference on Data Engineering Workshops* (pp. 134-138). IEEE.