International Journal of Emerging Technologies in Engineering Research (IJETER)
Volume 4, Issue 9, September (2016) www.ijeter.everscience.org

Skyline Query Processing for Clustering the Multidimensional Data

Stalin David D
Ph.D Research Scholar, Depart. Of CSE, PSN College of Engineering & Technology,Tirunelveli,Tamilnadu, India.

Dr.A.Jayachandran
Research Supervisor, Department of CSE, PSN College of Engineering & Technology, Tirunelveli,Tamilnadu,India.

Abstract – Data mining is the process of analyzing data from different perspectives and organizing it into useful information - information that can be used to increase turnover and also decrease costs. It is the process of finding correlations or patterns among dozens of fields in large relational databases. Recommender system is a information filtering system that seek to predict the 'rating' or 'preference' that user would give it to an item. Recommender systems are used in applications which need help for users in a decision-making process to choose an item amongst a potentially overwhelming set of alternative products or services. A recommender system estimate the user's profile to some reference characteristics and suggests a personalized recommendation. These recommendation may be from the information item (the content- based approach) or the user's social environment (the collaborative filtering approach) or a combination of both (Hybrid-filtering approach). But the recommender system suffer from the long tail problem, in which the unpopular data that have low number of positive ratings belong to the tail of the item distribution, these types of items should not be discarded or ignored but gainfully utilized in recommendation methods. Hence to overcome the limitation of the existing system, a skyline query processing approach which is known to take into account the multi-dimensional data that has very little popularity will be proposed. Using skylines the users can be recommended with new or unpopular recommendations which solve the long tail problem.

Index Terms – long tail problem, adaptive clustering, and recommender system.

1. INTRODUCTION

Recommendation methods are applied to two popular dataset s; Movie Lens and Book Crossing**.** Movie Lens datasets contains the customer-related attributes such as age and gender and the item- related attributes such as year of the movie and generic. It originally contained 1,000,209 rating on the scale of 1 to 5 from6, 040 customers of 3,900 movie. Book crossing datasets contains the customer-related attributes such as location and age and the item-related variables such as year of the publication, book author and publisher. It originally contained 1,149,780 ratings on the scale of 1to 10 from 278,858 customers of 271,379 books, however there were many blank values in it. We selectively used the items having more than 10 ratings in order to perform 10-fold cross-validation and also eliminated the rating data containing blank values. As a result, 998,539 ratings of 3,260 movies are used for the Movie Lens dataset and 11,990 ratings of552 books are used for the Book Crossing dataset in the experiments.

Many recommender systems ignore unpopular or newly introduced items having only few ratings and focus only on those items having enough ratings to be of real use in the recommendation algorithms. Alternatively, such unpopular or newly introduced items can remain in the system but would require special handling using various cold start methods, such as the ones described.

Unpopular or new items belong to the long tail of the item distribution. Following the spirit of extensive research on the Long Tail Phenomena, these types of items should not be discarded or Ignored but gainfully utilized in recommendation methods. The long tail of recommender systems and Purpose new methods of managing such items from long tail. To split items into the head and tail parts and group items in the tail parts using certain clustering methods. Splitting and grouping improves recommendation performance as compared to some of the alternative non- grouping and fully- grouped methods. Performance improvement by running various experiments on two "real-world" datasets. Head/tail Splitting strategies reducing error rates of recommendations and demonstrate that this partitioning often outperforms clustering of the whole item set. The long Tail problem in the context of recommender systems has been addressed previously. In particular, analyzed the impact of recommender systems on sales concentration and developed an analytical model of consumer purchases that follow product recommendations provided by a recommender system. The recommender system follows a popularity rule, recommending the bestselling product of consumers, and they show that the process tends to increase the concentration of sales. As a result, the treatment is somewhat akin to providing product popularity information. The model in does not account for consumer preferences and their incentives to follow recommendation or not. Also studied the effects of recommender systems on scales concentration and did not address the problem of improving recommendations for the item in the Long Tail, which constitutes the focus of this paper.

In a related question has been studied; to which extent recommender system account for an increase in the long tail of the scales distribution shows that recommender systems increase firm's profits affects scales concentration.

The cold start problem for the items in the Long Tail that have only very few ratings. A popular Solution to the cold start problem utilizes content- based methods when two items with no or only few ratings are inferred to be similar based on their content. In our work, we use grouping of items in the long tail, rather than the content-based methods to identify similar items and to leverage their combined ratings to provide better recommendations. Clustering methods used in recommender system. In particular, clusters similar users into the same cluster to overcome the data scarcity problem for collaborative filtering. Also in, item clustering Is used to improve the prediction accuracy of collaborative filtering where items were divided into smaller groups, and existing cf algorithms were applied to each group category separately .we use related clustering ideas but in the context of the Long tail phenomenon to leverage few ratings of the Items in the Long tail.

2. LITERATURE SURVEY

Wen Wu, Liang He, Jing Yang.[1] Recommender systems now tend to gain popularity and significance. The proliferation of many recommender systems leads to the difficulty of locating a good recommender system. The algorithm contained in the recommender system determine the efficiently of the recommender systems. The question now is to find the most appropriate algorithm to meet users needs. So far the research carried out has focused on improving the accuracy of recommender systems.

The recommender system should move beyond the conventional accuracy criteria and take some other criteria into account, such as coverage, diversity, serendipity, scalability, adaptability, risk, novelty and so on. Experimental results with data from VELO indicate the people with different interest degree tend to prefer different algorithms; thus the use of various evaluation criteria to judge the performance of algorithm is meaningful. E-Commerce has proliferated in terms of variety and quantity; the end-users spend considerable time to select the products and services. Recommender system can now be found in many modern applications that expose the user to a huge collection of items. Such systems typically provide the user with a list of recommended items they prefer or supply guesses of how much the user prefer each item. The recommender systems are supported by well founded and incremental algorithms. These algorithms differ considerably with respect to their strengths and weakness. Thus, the users encounter with choices for the selection of the most effective.

Most of the algorithm today focuses on improving the accuracy of the recommender system; however providing accuracy alone is inadequate. The main contribution in this paper is: Five simple recommendation algorithms based on the single item for coupon recommendation. Some special and novel evaluation metric systems are chosen and redefined to evaluate our recommender system.

PunamBedi, Sumit Kr Agarwal.[2] A recommender system compares the user's profile to some reference characteristics. These characteristics may be from the information item (the content-based approach) or the user's social environment or a combination of both (Hybrid-filtering).Recent research shows that collaborative recommender systems are highly vulnerable to profiles injection attacks. Security mechanisms are needed for protecting the recommender systems against these attacks. Aspect Oriented Recommender System (AORS) is a proposal multi agent system that uses the concept of Aspect Oriented Programming (AOP) for building security aspect. Implementing the security in recommender system using a conventional agent oriented approach results not only with the problem of code scattering and code tangling, but also results in weaker enforcement of security concern. In this paper, security crosscutting is handled as aspect in AORS in a modular way to remove scattering and tangling problems. The prototype of AORS has been designed and developed for a book recommender system to increased data security.

Information mounts, it leads to the problem of how to access, navigate through, and select available options. One possible solution to this information overload problem is based on recommender systems or the concept of automatic recommendation generation. A recommender system recommends items to user by predicting items, user information and interactions between users and items .As recommender system is a widely used application, this faces serious privacy and security issues as the personal information collected by the recommender system raising the risk of unwanted exposure of that information.

Malicious user can also enter the bias profiles into the system to manipulate the recommendations or reduce the accuracy of the recommender system for a key business advantages. This has led us to work on how we can avoid these risks or threats to a recommender system and can prevent our system from malicious users. Bedi et al, introduced trust based recommendation algorithm which allows agents in decision making to generate the recommendations based on their trustworthiness. The recommender system should move beyond the conventional accuracy criteria and take some other criteria into account, such as coverage, diversity, serendipity, scalability, adaptability, risk, novelty and so on. Experimental results with data from VELO indicate the people with different interest degree tend to prefer different algorithms; thus the use of various evaluation criteria to judge the performance of algorithm is meaningful. E-Commerce has proliferated in terms of variety and quantity; the end-users spend considerable time to select the products and services. Recommender system can now be found in many modern applications that expose the user to a huge collection of items. Such systems typically provide the user with a list of recommended items they prefer or supply guesses of how much the user prefer each item. The recommender systems are supported by well founded and incremental algorithms. These algorithms differ considerably with respect to their strengths and weakness. Thus, the users encounter with choices for the selection of the most effective.

Recommendation algorithm can prevent the recommender system from malicious user's profile injection attacks because even if biased profiles will be injected into the system, these can't be part of recommendation generation. In our proposed approach we have explored the various security relevant crosscutting concerns, their side effects on MAS modularity and build a security aspect. This security aspect is the integration of type-based and policy-based security strategies. Finally this security aspect is woven with multiple agents of AORS to eliminate the scattering and tangling problem. These agents coordinate and communicate with each other in the system to update their trust values on others for threat avoidance. Prior work is to introduce the concept of monitoring agent which monitors ongoing activities of system agents. Whenever some new agent is introduced or killed or influenced by the malicious code in the system then this agent immediately shows the appropriate message to the system administrator.

3. EVALUATION OF RECOMMENDER SYSTEM TO DETECT THE LONG TAIL PROBLEM

The Each Item (EI) recommendation method builds data mining models for each individual item I in I to estimate unspecified ratings. In other words, the EI method does not group an item with the other similar items at all and builds predictive models only by using the data in each item.

The EI method builds a predictive model for each of the 3260 movies using the ratings of each particular movie. if the movie toy story had 272 ratings we can built a linear regression model to predict the unknown ratings for that movie, use RMSE to measure the performance of the model and apply 10- fold cross- validation to compute RMSE for That movie the process was repeated3260 times for movie in Movie in Movie Lens. The main problem with the EI recommendation, method is that only a few ratings are available in the long tail. So the predictive model for the tail items are learned from only a few training examples using the EI method.

The LRTP problem is caused by a lack of data to build good predictive models in the tail, and therefore, clustering items can be a reasonable solution. The TOTAL CLUSTERING (TC) Recommendation method clusters the whole item set I into different groups by applying Conventional clustering methods such as k-means clustering and building rating-predictive models for each resulting group.

The TC method clusters 3260 movies into 100 groups using k-means clustering method and builds predictive models. If we want to predict unknown rating of the movie the other Boleyn girl for customer C, then the TC method first determines into which of these 100 groups that movie belongs. If the movie belongs to group G5, which Includes 30 other moving having 10,000 transactions among them, then the TC method applies the SVM method to group G5 and computes RMSE error rates using 10-foldcross-validation on these 10,000 ratings. This process was a repeated 10 times on the Movie Lens data for each cluster is shown in figure 1.

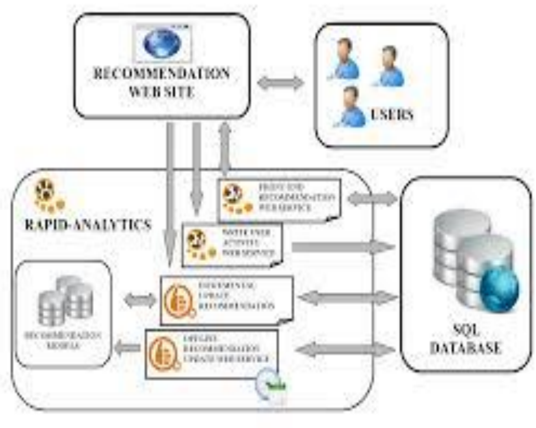

Fig 1: RMSE model

Adaptive clustering method clusters the item with other similar items when it has only a small amount of data, but groups to a lesser extent or does not group at all when it has a considerable amount of data. In the case of the Movie Lens dataset all movies from that dataset are ordered based on the popularity for each movie. Then, the popularity each movie is compared with the criterion number of ratings a. If it is larger than a, then the AC method does not apply any clustering method; instead, it keeps basic EI approach. However, if it is smaller then a, then the AC method clusters the movie with other similar movies one by one until the resulting group size reaches a. After that, the AC method builds rating predictive models using the resulting group for each item is shown in figure 2.

The movie secret sunshine only has 50 ratings, and then the AC method groups it with the most similar movie Beijing Bicycle, which has 35 ratings. However, the group size is still less than the criterion 100. Thus, the AC method finds the next similar movie poetry, which has 40 ratings, and then the group size becomes 125(50+35+40), which is large then the criterion number 100.Next, the AC method builds the predictive model for the movie secret sunshine using the 125 ratings in cluster group. In this way, it is been build predictive models for 3260 movies in the Movie Lens dataset. If it is needed to predict the unknown rating of the movie secret sunshine for customer C, then the AC method finds the predictive model build only for the movie and estimates the rating using it.

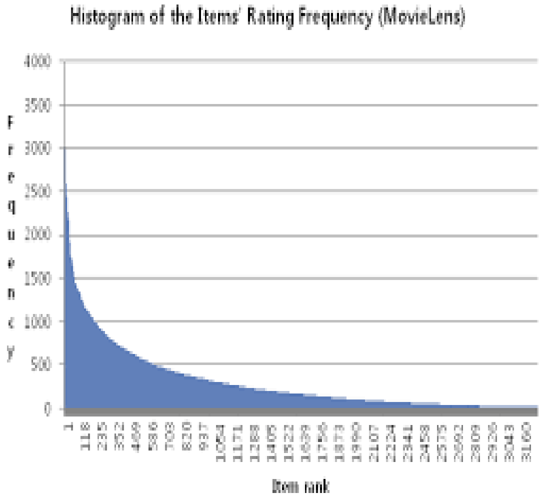

Fig 2: Histogram of the Item Rating Frequency

4. SKYLINE PROCESSING

Skyline Query Processing solves the limitation of the Adaptive clustering method for the long tail problem. Advanced query operators, such as skyline queries are necessary in order to help users to handle huge amount of data from a large by identifying set of non-dominated points. Skyline queries means it retrieve set non-dominated points or better points from the given data points. It provides an interactive environment for information retrieval that help user to get answer for the given preference based query. using skylines the user can now be recommended with new or unpopular recommendations which solves the long tail problem and hence detect the scarcity problem by providing sufficient amount data to built the good predictive model and it improves scalability and accuracy of the information retrieval by the user . Aspect Oriented Recommender System (AORS) is a proposal multi agent system that uses the concept of Aspect Oriented Programming (AOP) for building security aspect. Implementing the security in recommender system using a conventional agent oriented approach results not only with the problem of code scattering and code tangling, but also results in weaker enforcement of security concern. In this paper, security crosscutting is handled as aspect in AORS in a modular way to remove scattering and tangling problems. The prototype of AORS has been designed and developed for a book recommender system to increased data security.

The long tail brings dual benefits for increasing companies' profit: Compared with popular items, long tail items embrace relatively large marginal profit, which means the endeavor to expand the long tail market, can bring much more profit. if a product sells a lot, its profit margin usually low since all competitors have to sell the same product for the same price; if non-popular products are brought to the interest of right buyers with a successful mechanism, profitability increases drastically. Availabilities to the tail items can also boost the sales on the head due to the so called "one-stop shopping convenience" effect. According to results of many analyses, companies like Amazon that apply the Long Tail effect successfully make most of their profit not from the bestselling products, but from the long tail. Hence the skyline process solves the LTRP and makes the recommender system to be more robustness and diversity. .As recommender system is a widely used application; this faces serious privacy and security issues as the personal information collected by the recommender system raising the risk of unwanted exposure of that information. The recommender systems are supported by well founded and incremental algorithms. These algorithms differ considerably with respect to their strengths and weakness. Thus, the users encounter with choices for the selection of the most effective.

5. RESULT

The skyline query processing is used to cluster item according to the popularities which makes the products with low rating to be available for the choice of customer in head items, thus solves the LTRP problem.

6. CONCLUSION

Many recommender systems ignore unpopular or newly introduced items which is having only few ratings and focusing on those items with enough ratings to be of real use in the recommendation algorithms such unpopular or newly introduced items can remain in the system bus require special handling thus the total sale of large number of non-hit items is called "the long tail".

To solve this issue, the skyline query processing is been identify to evaluate every dimensions of the database and to handle d-dimension and select points which are not dominated by any other points in the database. Skyline can support queries that have specific interest in different subsets of dimensions. Whenever some new agent is introduced or killed or influenced by the malicious code in the system then this agent immediately shows the appropriate message to the system administrator and allows agents in decision making to generate the recommendations based on their trustworthiness.

REFERENCES

[1] Yoon-JooPark, "The Adaptive clustering method for the long tail problem of recommender systems" IEEE Transcation on knowledge and Data Engineering, 2013.
[2] Adomavicius.G and Tuzhilin.A, "Expert-driven Validation of Rule-based User Models In Personalisation Application "IEEE Trans Data Mining and Knowledge Discovery, 2005.
[3] Adomavicius.G and Tuzhilin.A, "Toward the Next Generation of Recommender Systems: A Survey of the State of the Art and possible Extensions" IEEE Transaction on Knowledge and Data Engineering, 2006.
[4] Anderson.C, "The Long Tail.New York: Hyperion Press", 2007.

[5] Bell.R.M and Yehuda.k, "Improved Neighbood – based collaborative Filtering" KDD Cup07, Calofornia, 2007.
[6] Fleder.D.M and Hosanagar.k, "Blockbuster Cultures Next Rise or Fall; The Impact of Recommender Systems on Sales Diversity", NET Institute Working Paper No.07-10,2008.
[7] Heli,JaesooYoo, "An Efficient Scheme for Continous Skyline Query Processing OverDynamic Data Set" , IEEE Transaction, 2014.
[8] Hervas-Drane.A, "Word of Mouth and Recommender Systems: A Theory of the Long Tail"NET Institute Working Paper No.07-41,2007.
[9] Linden.G, Smith.B, and York.J, "Amazon.com Recommendations: Item-to-Item collaborative filtering" , IEEE Internet Computing, 2003.
[10] Park,Y.J,Tuzhilin.A, " The Long Tail of Recommender systems and how to leverage It" , Proceeding of the 2008 ACM conference on Recommender Systems, Lausanne Switzerland, PP.no 11-18, 2008.
[11] PunamBedi, Sumit Kr Agarwal, "Managing Security in Aspect-oriented Recommender System", IEEE Transaction, 2011.
[12] Smola.A.J, Schoelkopf.B, "A tutorial on support vector regression, Statistics and computing", 2006.
[13] Truong.K.Q.Ishikawa.F, and Honiden.S, Improving Accuracy of Recommender Systems by Item Clustering", IEICE Transaction on Information and Systems, 2007.
[14] wittenL.H and Frank.E, "Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations" . Burlington, MA: Morgan Kaufmann, 2005.
[15] Xiao Yong Li, Yijie Wang,Xiaoling Li, Guangdong Wang "Skyline Query Processing on Interval Uncertain Data" , IEEE Transaction on Knowledge and Data Engineering, 2012.
[16] Haifeng Yu, Phillip B. Gibbons, Member, IEEE, Michael Kaminsky, and Feng xiao, [2010] "SybilLimit: A Near-Optimal Social Network Defense Against Sybil Attacks" Wei Wei, Fengyuan Xu, Chiu C. Tan, Qun Li The College of William and Mary, Temple University wwei, fxu, liqun @cs.wm.edu, cctan@temple.edu,[2013] "Sybil Defender: A Defense Mechanism for Sybil Attacks in Large Social Networks"
[17] Manuel Uruen´a, Member, IEEE, Rube´n Cuevas, Member, IEEE, A´ngel Cuevas, Member, IEEE, and Albert Banchs, Member, IEEE, [2013] "A Model to Quantify the Success of a Sybil Attack Targeting RELOAD/Chord Resources"
[18] Xiaohui Liang, Student Member, IEEE, Xiaodong Lin, Member, IEEE, and Xuemin (Sherman) Shen, Fellow, IEEE, [2013] "Enabling Trustworthy Service Evaluation in Service-oriented Mobile Social Networks"
[19] Yingying Chen, Member, IEEE, Jie Yang, Student Member, IEEE, Wade Trappe, Member, IEEE, and Richard P. Martin, Member, IEEE,[2010] " Detecting and Localizing Identity-Based Attacks understanding trust evolution in an online world," in KDD. ACM, 2012, pp. 253–261.
[20] Y. Chen, J. Yang, W. Trappe, and R. P. Martin, "Detecting and localizing identity-based attacks in wireless communication and sensor networks," IEEE Trans. Veh. Technol., vol. 59, no. 5, pp. 2418–2434, Jun. 2010.
[21] S. Capkun, J. P. Hubaux, and L. Buttyan, "Mobility helps peer-to-peer security," IEEE Trans. Mobile Comput., vol. 5, no. 1, pp. 43–51, Jan. 2006.
[22] C. Bettstetter, G. Resta, and P. Santi, "The node distribution of the random waypoint mobility model for wireless ad hoc networks," IEEE Trans. Mobile Comput., vol. 2, no. 3, pp. 257–269, Jul.–Sep. 2003.
[23] U. Kuter and J. Golbeck, "Sunny: A new algorithm for trust inference in social networks using probabilistic confidence models," in AAAI, 2007, pp. 1377–1382.
[24] K. Nordheimer, T. Schulze, and D. Veit, "Trustworthiness in networks: A simulation approach for approximating local trust and distrust values," in IFIPTM, vol. 321. Springer-Verlag, 2010, pp.157–171.
[25] G. Wang and J. Wu, "Multi-dimensional evidence-based trust management with multi-trusted paths," Future Generation Computer Systems, vol. 27, no. 5, pp. 529–538, 2011.
[26] J. Sabater and C. Sierra, "Reputation and social network analysis in multi-agent systems," in AAMAS. ACM, 2002, pp. 475–482.