Movie Recommender System

M.Reddi Prasanna[1], Mr J S.Ananda Kumar [2]
[1]*Student, Dept. of MCA, KMMIPS, TIRUPATI*
[2]*Assistant Professor, Dept.of MCA, KMMIPS, TIRUPATI*

Abstract- **Research on recommendation systems has gained a considerable quantity of attention over the past decade because the variety of on-line users and on-line contents still grow at an exponential rate. With the evolution of the social internet, individuals generate and consume data in real time using on-line services like Twitter, Facebook, and internet news portals. With the rapidly growing on-line community, web-based retail systems and social media sites have to process many variant user requests per day. Generating quality recommendations using this large quantity of knowledge is itself a really difficult task. Nonetheless, critical thewweb-based retailers like Amazon and Netflix, the preceding social networking sites got to face further challenge once generating recommendations as their contents are very rapidly ever-changing. Therefore, providing recent info within the least amount of time may be a major objective of such recommender systems. Though cooperative filtering may be a widely used technique in recommendation systems, generating the advice model using this approach may be an expensive task, and often done offline. Hence, it's tough to use cooperative filtering within the presence of dynamically ever-changing contents; per systems need frequent updates to the advice model to keep up the accuracy and therefore the freshness of the recommendations. power of graphic processing units (gpus) will be wont to method massive volumes of knowledge with dynamically ever-changing contents in real time, and accelerate the advice method for social media knowledge streams. During this paper, we address the problem of rapidly changing contents, and propose a parallel on-the-fly cooperative Filtering algorithmic rule victimization gpus to facilitate frequent updates to the recommendations model. We use a hybrid similarity calculation methodology by combining the item–item cooperative filtering with item class info and temporal info. The experimental results on real-world datasets show that the planned algorithm outperformed many existing on-line CF algorithms in terms of accuracy, memory consumption, and runtime. It had been additionally discovered that the planned algorithm scaled well with the information rate and therefore the data volume, and generated recommendations during a timely manner.**

Index Terms- **Recommender system, collaborative filtering, power of graphic processing units.**

I. INTRODUCTION

Generally during decision making process taking opinions from people is a common criterion. In olden days when an individual need to take decision he would probably ask opinions from friends and family.[2] Now,world has been changed[4,5]. E-Commerce sites, on-line communities or groups, forums, discussion teams, web logs, product rating sites, chat rooms are a number of the resources on which individuals will currently share their ideas about something in discussion[1,3]. However, Recommender systems facilitate users by predicting interesting products and services where the amount and complexity of offers increases the user's capability so that they can view the suggestions and make a choice[8]. Recommender systems were introduced as a computer-based intelligent technique to deal with the problem of information and product overload. The system recommends those items to a user that matches his interests.[11] The two basic entities which appear in any Recommender System are the user (sometimes also referred to as customer) and the item (also referred to as product). A user is a person who utilizes the Recommender System providing his opinion about various items and receives recommendations aboutnew items from the system[10]. Collaborative filtering is one of the widely used techniques in recommender systems. But generating the recommendation using collaborative filtering is a costly task and it requires often offline[6]. Therefore it is tough to use collaborative filtering in the presence of dynamically changing contents as such system requires frequent updates to recommender model to maintain accuracy and freshness of the recommendations. In this paper we propose a novel GPU-accelerated, item-based recommender system for social media streams, using a hybrid similarity function computed based on conditional probability, item category information, and temporal information[14]. Experiment results on real-world datasets show that our proposed method outperforms two existing online CF algorithms in terms of accuracy, memory usage, and runtime.[1-15]

II. RELATED WORK

Movie Recommendation Systems based on Collaborative Filtering Recommendation systems (RS), introduced by Tapestry project in 1992, is one of the most successful information management systems [12]. The practical recommender applications help users to filter mass useless information for dealing with the information overloading and providing personalized suggestions. There has been a great success in e-commerce to make the customer access the preferred products, and improve the business profit. In addition, to enhance the ability of personalization, recommendation system is also widely deployed in many multimedia websites for targeting media products to particular customers[8]. Nowadays, Collaborative filtering (CF) is the most effective technique employed by movie recommendation systems, which is on the basis of the nearest-neighbor mechanism[1].

III. PROPOSED MODEL OR ALGORITHM

Following is a brief description of the methods which are used in the algorithm.

Collaborative filtering

Collaborative filtering also referred to as social filtering, filters information by using the recommendations of other people. It is based on the idea that people who agreed in their evaluation of certain items in the past are likely to agree again in the future. A person who wants to see a movie for example, might ask for recommendations from friends. The recommendations of some friends who have similar interests are trusted more than recommendations from others. This information is used in the decision on which movie to see.

Many collaborative filtering systems have to be able to handle a large number of users. Making a prediction based on the ratings of thousands of people has serious implications for run-time performance. Therefore, when the number of users reaches a certain amount a selection of the best neighbors has to be made. Two techniques, correlation-thresholding and best-n-neighbor, can be used to determine which neighbors to select. The first technique selects only those neighbors who's correlation is greater than a given threshold. The second technique selects the best *n* neighbors with the highest correlation.

In this approach, the data of users along with that of items is used to make the prediction. The underlying assumption of the collaborative filtering approach is that if a person *A* has the same opinion as a person *B* of an item, A is likely to have B's opinion on a different item as well.

Following is a brief description of the methods which are used in the algorithm.

Three main techniques are used mainly to create a recommendation system –

A.Content-Based System - The prediction is generated using the meta data of the movie or user. For e.g - Identifying the genres of Movie that the target user prefers and then recommending a similar movie from the data base[10].

B.Collaborative-Filtering System – Many users have similar tastes, and hence rate movies similar ratings to movie. Also many movies are rated similarly by different users. Thus by collaborating these similarities predictions can be generated[5].

C.Mixed Approach – Using both the Meta Data and the ratings data to predict the ratings.

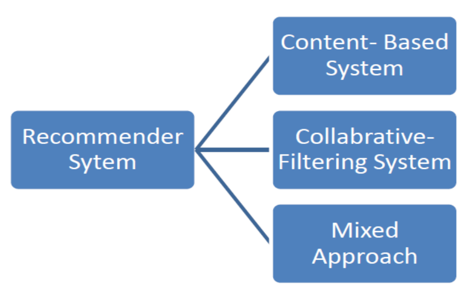

Figure 1. The Collaborative system algorithm [25]

1.Descriptions of Collaborative Preprocessing

The basic idea, again, is to scan through the user-item matrix and replace missing values by using some naive non-collaborative methods that provide a simple rating.

1.1.User Average Scheme

In this scheme, for each user, ui, we compute the average user rating, $r\bar{i}$ which is expressed by the average of the corresponding row in the user-item matrix. The user average is then used to replace any missing ri,j value. This approach is based on the idea that a user's rating for a new item could be simply predicted if we take into account the same user's past ratings. It can be formally stated as:

rij={$r\bar{i}$, $ifuseruihasnotrateditemijr$,$ifuseruihasrateditemij$

1.2.Item Average Scheme

In the item average scheme, we utilize the item average $r\bar{j}$ of each item, ij, as a fill-in for missing values ri,j in the matrix. Specifically, we calculate the column average of each column in the user-item matrix, and fill all slots of the same column that have no value, using that average:

rij={$r\bar{i}$, $ifuseruihasnotrateditemijr$,$ifuseruihasrateditemij$.

1.3.Composite Scheme

When a missing entry regarding user's ui opinion on item ij is located, we first compute the user average, $r\bar{i}$ calculated as the average of the corresponding user-item matrix row. Then we look for existing ratings in the column which corresponds to item ij . Assuming that a set of l users, L = {u1, u2, ..., ul}, has provided a rating for item ij , we can compute a correction term for each user up 2 L equal to δp = rp,j − $r\bar{p}$ After the corrections for all users in L are computed, the composite rating can be calculated: rij= {$r\bar{i}$ + $\Sigma\delta plp$=1l, $ifuseruihasnotrateditemijr$, $ifuseruihasrateditemij$.

IV. CONCLUSION

In this paper, we proposed a GPU-accelerated collaborative filtering algorithm for social media data streams. To address the dynamically changing contents and high user/item churn present in such data streams, we propose a method to gradually discard the obsolete user/item information and update the recommendation model in a timely manner using GPUs. Parallel processing power of GPUs speeds up the recommendation process and keeps the recommendation model up-to-date with respect to the rapidly changing contents. Using conditional probability-based similarity calculation along with item category information and temporal information, our system was able to bring the newly added items to users' attention at an early stage. The accuracy, memory usage, and runtime comparisons with two existing online CF algorithms showed that our proposed approach produced better recommendations using less memory and less execution time. We also showed that the proposed parallel GPU implementation outperformed the respective multi-core CPU implementation of the same algorithm and achieved up to 12X speedup. The proposed algorithm scaled well with the increasing number of users, number of items, and the data rate, and produced quality results in a timely manner, making it applicable for large-scale social stream recommendation systems.

REFERENCES

[1] Su X, Khoshgoftaar TM (2009) A survey of collaborative filtering techniques. Adv Artif Intell 2009:4

[2] Deshpande M, Karypis G (2004) Item based top-N recommendation algorithms. ACM Trans Inf Syst 22(1):143–177

[3] Bobadilla J, Fernando O, Hernando A et al (2013) Recommender systems survey. Knowl Based Syst 46:109–132

[4] Pan R, Zhou Y, Cao B et al (2008) One-class collaborative filtering. In: Proceedings of the IEEE international conference on data mining, pp 502–511

[5] Kitts B, Freed D, Vrieze M (2000) Cross-sell: a fast promotion-tunable customer-item recommendation method based on conditional independent probabilities. In: Proceedings of the ACMSIGKDD conference on knowledge discovery and data mining, pp 437–441

[6] Linden G, Smith B, York J (2003) Amazon.com recommendations: item-to-item collaborative filtering. IEEE Internet Comput 7(1):76–80

[7] Sarwar B, Karypis G, Konstan J et al (2001) Item-based collaborative filtering recommendation algorithms. In: Proceedings of the ACM international conference on world wide web, pp 285–295

[8] Das AS, Datar M, GargAet al (2007) Google news personalization: scalable online collaborative filtering. In: Proceedings of the 16th international conference on world wide web, pp 271–280
[9] Davidson J, Liebald B, Liu J et al (2010) The YouTube video recommendation system. In: Proceedings of the ACM RecSys, pp 293–296
[10] Hofmann T (2004) Latent semantic models for collaborative filtering. ACMTrans Inf Syst 22(1):89–115
[11] Gemulla R, Nijkamp E, Haas PJ et al (2011) Large-scale matrix factorization with distributed stochastic gradient descent. In: Proceedings of the ACM SIGKDD conference on knowledge discovery and data mining, pp 69–77
[12] Han S, Yang Y, LiuW(2011) Incremental learning for dynamic collaborative filtering. J Softw 6(6):969–976
[13] Narang A, Srivastava A, Katta NPK (2011) Distributed scalable collaborative filtering algorithm. In: Proceedings of Euro-Par conference, pp 353–365
[14] Shang Y, Li Z, Qu W et al (2014) Scalable collaborative filtering recommendation algorithm with MapReduce. In: Proceedings of the IEEE international conference on dependable, autonomic and secure computing, pp 103–108
[15] Wang Y, Liao X, Wu H et al (2012) Incremental collaborative filtering considering temporal effects.