Improving Recommendation Techniques by Deep Learning and Large Scale Graph Partitioning

Gourav Bathla[1], Himanshu Aggarwal[3]
Department of Computer Engineering
Punjabi University
Patiala, India

Rinkle Rani[2]
Department of Computer Science & Engineering
Thapar University
Patiala, India

***Abstract*—Recommendation is very crucial technique for social networking sites and business organizations. It provides suggestions based on users' personalized interest and provide users with movies, books and topics links that would be most suitable for them. It can improve user effectiveness and business revenue by approximately 30%, if analyzed in intelligent manner. Social recommendation systems for traditional datasets are already analyzed by researchers and practitioners in detail. Several researchers have improved recommendation accuracy and throughput by using various innovative approaches. Deep learning has been proven to provide significant improvements in image processing and object recognition. It is machine learning technique where hidden layers are used to improve outcome. In traditional recommendation techniques, sparsity and cold start are limitations which are due to less user-item interactions. This can be removed by using deep learning models which can improve user-item matrix entries by using feature learning. In this paper, various models are explained with their applications. Readers can identify best suitable model from these deep learning models for recommendation based on their needs and incorporate in their techniques. When these recommendation systems are deployed on large scale of data, accuracy degrades significantly. Social big graph is most suitable for large scale social data. Further improvements for recommendations are explained with the use of large scale graph partitioning. MAE (Mean Absolute Error) and RMSE (Root Mean Squared Error) are used as evaluation parameters which are used to prove better recommendation accuracy. Epinions, MovieLens and FilmTrust datasets are also shown as most commonly used datasets for recommendation purpose.**

Keywords—Social big data; social recommendation; deep learning; graph partitioning; social trust

I. Introduction

Social networking sites are used due to a lot of important information available. Large numbers of users interact with each other; share their views on these sites. Business organizations have diverted their business models towards these sites. Researchers, data scientists and innovators are working actively in this area to extract patterns of user behavior. It is assumption that sites which have large numbers of connected users can become source of benefit for other business organizations as well. Analysts have proved that users' personalization is major factor for retaining users on sites. If users are provided with better suggestions for books, friends, topics or products which are of users' interest, it is high probability that effectiveness would be improved significantly. Recommender systems are capable of providing better suggestions to users based on their past history, likings, ratings and trust amongst users. It is dependent on user behavior data and historical data [1]. It is analyzed that 80 percent of movies on Netflix are due to recommendations [2]. 60 percent of YouTube videos from main page are due to recommendation [3]. It reflects significance of recommender systems for users.

Content based and collaborative filtering are categories of recommender systems. In content based systems, users likings is the main factor for recommendation. Ratings based recommendations are provided in collaborative filtering technique. Memory-based and model-based are different methods in collaborative filtering [4]. In memory-based CF, ratings of users are considered for recommendation. In model-based CF, data mining and machine learning techniques are used for recommendation. Models based on clustering [5] and latent semantic [6] are most commonly CF approaches. The drawback of content based approach is users' privacy. Many users do not reveal their like or views for any topic or product. The drawback of collaborative filtering is sparsity and cold start. There exist very few users who provide ratings to products hence user-items ratings matrix values are very less. It is not easy to analyze users similarity based on these few entries. When any user is new to recommender system, there is no ratings information so no similarity with other users can be predicted.

Matrix factorization, large scale graph partitioning, clustering, dimensionality reduction and deep learning are techniques which are provided by recent researches to improve recommendation. There are very few research works which covers all these novel techniques. In this paper, comprehensive analysis of recommendation improvement approaches are available for readers.

Deep learning has been very effective in speech recognition and image processing [7]. Users' likes and similarity with other users are easily concluded from deep learning. AlexNet which is convolutional network model improved classifying images significantly [8]. Feedforward, Recurrent Neural Network, Convolutional Neural Network, Restricted Boltzmann Machine and Deep Belief Network are various models of deep learning. There are models which are beneficial in content based recommendation and others are applicable in collaborative filtering techniques.

The limitation of traditional recommender systems is scalability. When these systems are deployed on large scale data, performance degrades significantly. Some systems are capable enough for recommendation for big social graph, but timings for response are so high that it becomes troublesome for users. There is requirement of novel approach that can handle big social graph. In this paper, solution of scalability issue is also mentioned. Large scale graph partitioning is explained in detail. Random based and trust based partitioning is described in this paper. There are very few research works which covers and integrates sparsity, cold start and scalability solution.

There are numerous datasets available for recommendation which makes readers confused and they are not able to select most suitable dataset as per their requirements. In this paper, Epinions, FilmTrust and MovieLens statistics are described in detail and their usage for particular approaches. MAE (Mean Absolute Error), RMSE(Root Mean Squared Error), Precision , Recall, Diversity, Serendipity and Novelty are used by several researchers as evaluation metrics. The most significant and informative metrics MAE and RMSE are explained in detail in this paper.

The outline of rest of the paper is as follows. Section 2 describes various recommendation techniques and improvements. In Section 3, deep learning application in recommender systems is explained. Section 4 describes standard datasets, metrics used for recommendation performance evaluation Section 5 explains latest tools and systems used for deep learning, recommendation. Section 6 concludes this paper.

II. Recommendation Techniques

Various recommendation techniques are proposed by researchers. Content based, collaborative filtering based and hybrid based are traditional recommendation techniques. When user likes any topic or product, their preferences can be used as important factor to recommend any new topic or product. This type of content based recommendation technique constraint is many users do not provide their preferences. Collaborative filtering is based on users' similarity. When any user A likes any topic or product, rating is provided by user in scale 1-5. In similar manner, other user B also provides rating to same topic or product in the same scale. If these pair of users has provided similar ratings, these users can be considered as similar users. If any new topic or product is liked by user A with some ratings, it is very high probability that other user B will also like this new topic or product with same ratings. The limitation of collaborative filtering is sparsity and cold start. There are very few entries in user- item matrix, so it is difficult to predict recommendation. New users do not provide any ratings hence cold start users cannot be provided with better recommendations. Hybrid approach combines content based and collaborative filtering based approaches.

Fig. 1 clearly demonstrates ratings of products provided by users. User 1 and user 3 have rated Product 1 with same ratings 2 i.e. it can be concluded that these users are similar users. User 4 has provided rating 5 to product 1 hence this user is not similar to user 1 and user3.

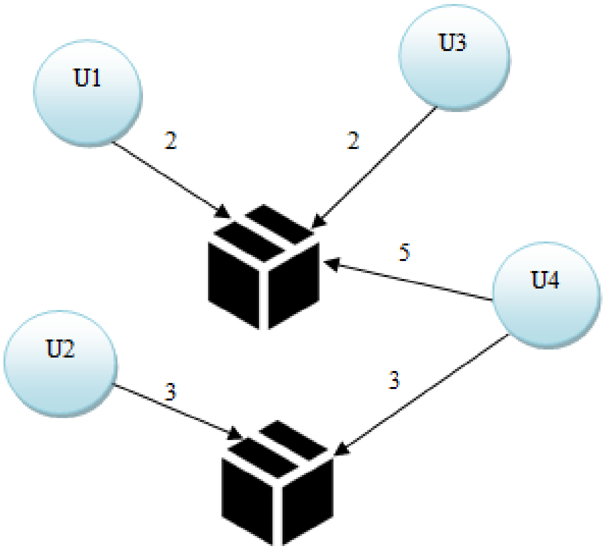


Fig. 1. User – Item Ratings.

TABLE I. USER- ITEM RATINGS MATRIX SPARSE VALUES

Users	*Product 1*	*Product2*
User 1	1	0
User 2	0	1
User 3	1	0
User 4	1	1

TABLE II. USER- ITEM RATINGS COLD START VALUES

Users	*Product 1*	*Product2*
User 1	1	0
User 2	0	1
User 3	1	0
User 4	1	1
User 5	0	0

Table I shows that entries not filled by users in user-item matrix results in sparsity. When numbers of users and items are in large scale, sparsity degrades recommendation accuracy.

When this demo recommender system includes new user 5, there is no entry for this user. Table II shows the cold start issue due to inclusion of new user in recommendation system.

A. Bi-Clustering Approach

Clustering is used to remove the sparsity in user-item matrix. When large scale of data is used in social datasets, there exist a lot of users who have not rated even single item. Processing of large social graph is not an efficient technique. Sparse matrix is optimized to maximum entries filled matrix by using clustering of similar users and items. In uni-clustering approach, only similar users or items are combined. Bi-clustering approach is used to combine users as well as items [9]. Users are combined in one cluster Uc who have similar likings and Items are combined in one cluster Ic which are of similar characteristics. Users and items combinations are not dependent [9]. In bi-clustering technique, many items for which there is no ratings and many users who have not rated any item, are not removed from matrix. These entries are shifted to right side of matrix. Users who are frequent raters and items which are rated frequently are aggregated to one side of area which becomes dense popular. In this approach, empty profiles are removed to provide better recommendation even for large scale datasets.

B. Social Trust Clustering

Social trust is used by several researchers as tool for recommendation. When users are connected on social networking sites, they share their opinion for any topic or product. If any connected user finds this opinion relevant, trust amongst these users increases. These trusted users are strongly connected if represented in social graph. When large numbers of users are connected, large scale graph is not in capability of traditional recommender systems. Trust can be used to cluster strongly connected users. The advantage of clusters is more focus can be given to more trusted users. When recommendation is provided to users, there are very high chances that trusted users are in same cluster.

Clustering has been applied for collaborative filtering, but there are very few clustering techniques based on trust [10]. Similarity measures are used to find similar users in social graph, but several researches prove that it only increases MAE (Mean Absolute Error) i.e. recommendation accuracy degrades. Bayesian model is also used to cluster similar users [11] [5]. Clustering resulted from this approach is not good enough to be applied in recommendation. Graph theoretic methods are also used to cluster users based on their preferences, but is has not been used for recommendation [12].

Trust inference is used to cluster strongly connected users. If users are strongly connected then some extra weight is provided to users for better recommendation. If user A assumes the probability of trusting user B as $p_{a,b}$ and user B assumes the probability of trusting user C as $p_{b,c}$, it is inferred that user A trust on user C should be $p_{a,c}$. Probability of path is large if users are strongly connected. Distance between nodes for clusters is inversely proportional to probability of path amongst them. Several researchers proved that using trust based clustering improves recommendation accuracy by using datasets like Epinions, FilmTrust etc.

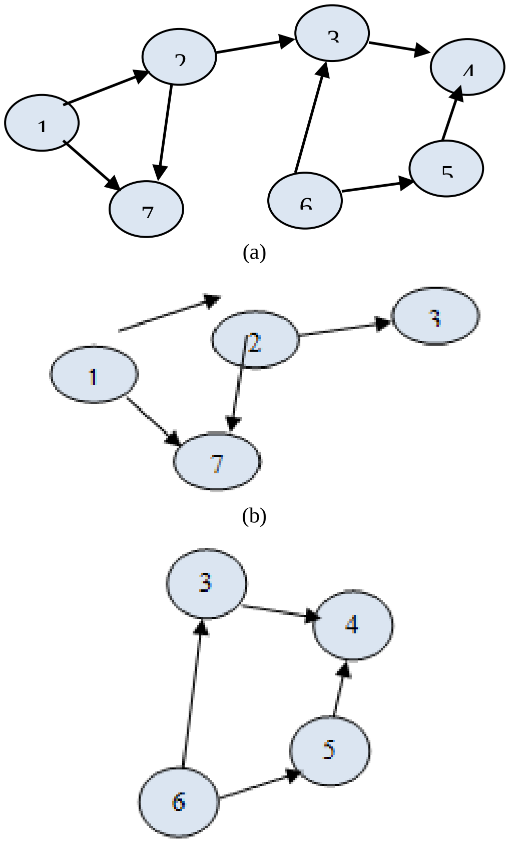
In addition to sparsity and cold start, large scale data recommendation is also issue due to which recommendation accuracy is degraded. Scalability issue is resolved by many researchers by using large scale graph partitioning.

C. Large Scale Graph Partitioning

Social networking sites data can be better represented by using graph. Users are represented by nodes and connections amongst users are represented by edges. Social network analyst process and manipulate social graph to extract important information. Social trust, highest influence node, connection of specific node etc are required for better analysis. Large scale social graph cannot be processed by traditional big data technologies. Centralized systems cannot analyze big social graph. There is need of novel approaches for efficiently dealing with large numbers of nodes and edges. Researchers have proved that big social graph can be easily analyzed by distributing it on different nodes using graph partitioning.

Social graph G is partitioned in k partitions - G1, G2, _ _ _ _ _, Gk. These partitions are distributed on nodes so that it can be run in parallel. Random partitioning is used by several researchers to prove the effectiveness of social graph partitioning. It works on different nodes by selecting subgraphs such that every node is part of at least one subgraph and every subgraphs have approximately same numbers of nodes. Balanced partitioning is the main motive of researchers. Social recommendation for large scale graph is not efficient by random partitioning.

Trust is built amongst connected users based on same liking or same ratings for any movie or product. If any user B likes the same product or topic also liked by user A with same ratings, trust values are increased. Trust is asymmetric i.e. if user A trusts user B, it cannot be concluded that user B also have trust on user A. Social graph partitioning with trust assures that most trusted nodes are in same subgraph. It is not assured that partitioning is balanced by using this approach, but recommendation accuracy is improved significantly. Locality is the ratio of number of nodes which are in same subgraph as was in original subgraph and total numbers of nodes. Locality is improved by using trust based partitioning technique. In Fig. 2(a), trusts amongst users are shown as directed edges. In Fig. 2(b) and Fig. 2(c), subgraphs are created after partitioning original subgraph based on trust values.



(a)

(b)

(c)

Fig. 2. (a).User – User Social Trust Graph (b). subgraph1 (c). subgraph2.

Pregel is framework based on Bulk Synchronous Parallel Model. Subgraphs are processed by using vertex centric approach. Supersteps are like iterations, nodes process task in parallel in one superstep and assign its processed task to next superstep. Giraph is open source implementation of Pregel like system. There are many abstract classes and methods available in Giraph. Trust based partitioning can be easily configured using Giraph library. Vertex based and edge based partitioning techniques are used for large scale graph. In vertex based partitioning, subgraphs are created by dividing vertices. Edges are the criteria for dividing large scale graphs in subgraphs.

III. Deep Learning based Recommendation

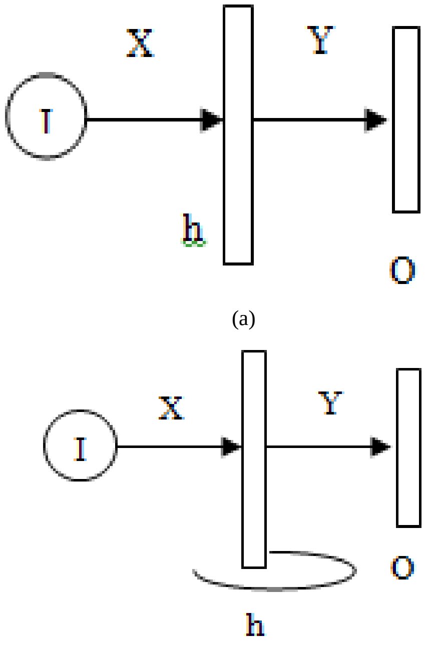
Deep learning is machine learning approach which is applied in many research areas such as Natural Language Processing and computer vision etc [13]. It has provided significant and state-of-the-art improvements in many fields and able to solve complex tasks. It can be used as supervised as well as unsupervised learning. Recommender system is also hot topic for last decade. Several researchers are finding the way to improvise recommender systems. Deep learning is applied by several researchers in recommendation to improve accuracy, precision and recall. Deep learning is applied on recommender systems so that it can improve the performance as it has improved other research areas [13]. Deep learning for recommender systems becomes popular research topic in 2016 during ACM RecSys . The reasons for integrating deep learning with recommender systems are the ability of deep learning to build complex non linear relationship in input and output data, better scalability for large scale of data and better analysing incorrect label data [15]. Researchers have observed that integrating these techniques improves recommendation in tremendous manner. [16] proposed YouTube video recommendation based on deep neural network. In [23], Yahoo news recommender system is implemented using RNN. Traditional recommendation systems are studied extensively and improved, but still sparsity, cold-start and scalability are the issues which degrade recommendation accuracy. Input and output are analysed for verifying improvement in recommendation accuracy. Input in recommendation systems is ratings, clicks or any explicit feedback provided by user and output is ratings prediction for user. Deep learning enhances the improvement in input and output systems. Deep learning can be implemented for content based or collaborative filtering based or combination of different architectures. Deep learning can be used to improve probabilistic matrix factorization. Scalable recommendation is improved by using deep learning as suggested by several researchers.

Several deep learning models are used – Multilayer Perceptron, AutoEncoder, Convolutional Neural Network, Recurrent Neural Network, Deep Semantic Similarity Model and Restricted Boltzmann Machine. Some models use only single deep learning techniques while some models use composite deep learning techniques [7]. It is proved that Restricted Boltzmann Machine performs better that traditional matrix factorization. Several research works have been proposed for applying deep learning in recommendation systems, but very few are able to succeed in improving recommendation accuracy. Moreover, scalability is the main concern for researchers using deep learning for social recommendation. In [4], It is clearly mentioned that neural network can not work properly due to very few entries in user-item matrix. It is necessary to normalize the values so that neural networks are trained effectively.

Several models in deep learning are proposed but feedforward neural network is the most commonly used model [13]. In this model, input layer submits data to hidden layer and after some processing functions, it is submitted to output layer. Constraint is that model can only use numerical data, but natural languages are used for recommendation. Recurrent model is applied for using different data and after processing submitted to output layer.

It is clear from Fig. 3(a) that input is submitted to hidden layers and after rectifier function, it is submitted to output function. In Fig. 3(b), recurrent model is demonstrated which used recurring hidden layers for sending information and finally it is submitted to output layer. Natural language processing can be easily applied using recurrent model.

Multilayer perceptron can be applied on user-item ratings to improve recommender systems. It is the simplest model [14] .It can approximate any function [15]. The advantage of using this model is that data need not be input separately as it is directly be used in multilayer neural network model. Multiple layers are present in this model. This model can be used for any degree of accuracy based on measurable function. Neural collaborative filtering uses matrix factorization linear approach and MultiLayer perceptron to improve recommendation accuracy. MLP is applied for improving YouTube video recommendation [16]. Multi-layer neural networks work on its function and perform better than state-of-the-art algorithms for traditional recommender systems. Complex linear and non-linear links can be better predicted by deep neural networks. It can also learn on large scale of data.



(a)

(b)

Fig. 3. (a). Feedforward Model (b) Recurrent Model.

Deep neural network works on at least one input layer and one output layer. Many hidden layers can be present in input and output layers depending upon the complexity of data. There is no standard to define that how many hidden layers can be concluded as deep [17]. Neural network with two hidden layers can be considered as deep neural network [18] [19]. In deep neural network, activation function decides the communication of data amongst different layers. Rectifier function is example of activation function.

$$f(z) = \max(0, z) \tag{1}$$

It is observed in experiment evaluation of several research works that using deep learning, MAE and RMSE improves significantly hence improves recommendation accuracy. It is also analysed that DNN model performs better with two hidden layers as compared to model where more numbers of layers are added.

Artificial neural networks work on at least one input and one output. Rectifier function enhances the ratings prediction hence improves recommendation accuracy. It is observed in research works that simplest rectifier function can improve ratings predictions significantly. Hidden layers numbers selection is tedious task in deep neural network. Researchers have provided the solution that when MAE stops improving, no more hidden layers should be included. User-item and user-user matrix values are combined to be deployed on hidden layers in deep neural network. For Example, music recommendation is very popular amongst users. It provides users choice amongst large numbers of songs. Existing music recommendation techniques are based on content of songs but it is very difficult to recommend songs to user just based on their emotions or solely content of songs. Deep learning is used by several researchers to improve recommendation accuracy.

In [20], novel approach is proposed which combines deep belief network with content to improve recommendation. In this approach, feature learning is based on songs content directly and automatically. In deep belief network, many hidden layers and one observation layer exist. It is combination of Restricted Boltzmann Machines and feed-forward model [14].

The summary of deep learning models used for recommendations are as follows:

- Multilayer perceptron model is type of feedforward model. In this model, there are various hidden layers that functions exactly like feed forward technique.
- Deep Belief Network is combination of Restricted Boltzmann machines and feed forward model. Restricted Boltzmann machines are used in collaborative filtering as user-item matrix can be improved by using this model.
- Convolution Neural Network model is based on signals. It is proven to be efficient in image recognition. Content based recommendation is improved by this model.
- Recurrent Neural Network is used in applications where hidden layers are to be processed many times. In content based recommendation, this model is widely used as it can provide better user's preferences for any topic or product.

Sparsity, cold start and scalability of user-item matrix can be reduced by using deep model and large scale graph partitioning. User-item interaction is improved significantly by using deep learning. For example, in Restricted Boltzmann Machines item features and implicit feedback is considered in addition to collaborative filtering ratings, hence improves sparsity and cold start. In this paper, solution for this concern is provided by following steps:

- Improving trust and rating matrix using indirect trust by hyperedge and transitive closure.
- Improving recommendation accuracy by using deep learning.
- Improving scalability by partitioning large scale social graph.

IV. Recommendation Dataset and Evaluation Metrics

A. Datasets

Several datasets are used by researchers for proving their approach better as compared to previous approaches. It is confusing for readers to select best suitable datasets. In this section, most commonly used datasets are explained with their usage so that readers can use proper dataset in their experiment evaluation based on approach.

1) Epinions: Epinions is site that maintains ratings by users in numerical scale. It also stores trust and distrust amongst users. Several research works use trust as an important factor for proving better recommendation accuracy. This is the reason Epinions is used in a lot of research works. It maintains who trust who and assumed as recommendation standard dataset. Trust data and ratings data are collected in this dataset. Trust data format is source_user_id, target_user_id, trust_binary_value. For example, if user 1000 trusts user 2000, format for storage is 1000,2000,1, where 1 denotes true value of trust amongst these users. In this dataset, only true trust value is stored and no distrust value is stored. Rating data format is user,item,ratings where ratings in scale 1-5. For example user 10 provides rating 3 to item 20, format for storage is 10,20,3. Table III clearly shows the statistics of user-item ratings and user-user trust values. These values are large enough to train the data and test remaining data for social recommendation. The motive of researchers is to improve trusters in this dataset.

TABLE III. Epnions Dataset Statistics

Dataset statistics	
Users	49290
Items	139738
Ratings	664824
Social Trust statistics	
Trusts	487181

2) FilmTrust: FilmTrust is small dataset used for recommendation. In this dataset, co-purchased products record is also maintained. The advantage of this dataset is combination of user-item ratings as well as user trust values. Detailed statistics is mentioned in Table IV.

TABLE IV. FILMTRUST DATASET STATISTICS

Dataset statistics	
Users	1508
Items	2071
Ratings	35497
Social Trust statistics	
Trusts	1853

3) MovieLens: MovieLens is data collected by GroupLens research. It is specifically used for movie recommendation. This dataset is most suitable for large scale recommendation experiment evaluation as shown in Table V.

TABLE V. MOVIELENS DATASET STATISTICS

Dataset statistics	
Users	138493
Movies	27278
Ratings	20000263

B. Evaluation Metrics

Several metrics are used to evaluate recommendation approaches performances. The most commonly used metrics are introduced here so that readers can apply these in their techniques and evaluate performance comparing it with existing techniques. It is verified by several researchers that small improvement in these metrics concluded significant improvements in recommendation accuracy. Precision, Recall, MAE and RMSE are most commonly used evaluation metrics.

1) Mean Absolute Error (MAE): It is sum of difference in predicted ratings by proposed approach and ratings that exist in original and divided by number of observations.

$$MAE = \sum_{i=0}^{n} \left(P1(u,i) - P2(u,i)\right)/n \qquad (2)$$

Where P1(u,i) is predicted rating for user u of product i and P2(u,i) is original ratings for user u of product i. Researchers prove their better approaches by reducing this error as much as possible. In **[21]**, it is mentioned that approaches which achieve even small improvement in MAE, is significant contribution.

2) Root Mean Squared Error (RMSE): This metrics is calculated as square root of sum of difference of predicted rating and original rating square and divided by number of observations. Researchers observed that RMSE is more accurate evaluation metric.

$$RMSE = \sqrt{\sum_{i=0}^{n} \left(P1(u,i) - P2(u,i)\right)^2/n} \qquad (3)$$

Where P1(u,i) is predicted rating for user u of product i and P2(u,i) is original ratings for user u of product i. RMSE and MAE more difference signifies degraded recommendation performance.

V. RECOMMENDATION TECHNOLOGIES

Several technologies are invented by researchers and practitioners to provide recommendation to users. Standard packages can be applied to applications like statistical analysis as there are fixed and formulated concepts. There is no standard package for recommendation due to its dynamic behaviour. Different types of technologies are available for content based, collaborative filtering based, hybrid based, social recommendation and trust based recommendation. Traditional Big data technologies cannot be used to implement data which is in social graph. Hadoop and MapReduce are not easily deployed for large scale graph processing. Large scale graph where users are represented as nodes and connections amongst them are represented as edges, can be manipulated by graph processing technologies like Pregel, Giraph, GraphLib etc. These libraries are designed specifically for graph algorithms like clustering, partitioning, shortest path or finding maximum weight etc. In this section, most commonly used technologies are described with application examples.

A. Pregel

Pregel uses Bulk Synchronous parallel model. It is based on vertex message passing approach. Supersteps are defined for each iteration and every vertex is running job in parallel in one superstep. When this superstep completes, vertices states is passed to next superstep. This can be most suitable for large scale social recommendation. When recommendation is to be applied for Big data, centralized systems can not process large scale user-item and user-user trust matrix. There is need of novel technique which can distribute large scale data and run jobs in synchronous. Pregel is most suitable model for this issue. Large scale graph algorithms are predefined and also many processing techniques can be customized on Pregel.

B. Giraph

Apache Giraph is API library which is open source implementation of Pregel. Vertex class is already available to form graph of vertices. When any vertex needs to communicate with other vertex, there are methods for accomplish it. VoteToHalt can be used by vertex after completion of superstep. Many graph processing algorithms are defined in Giraph API. Social graph algorithms such as shortest path and global popularity techniques are also available in Giraph API.

C. SNAP

Stanford Network Analysis Platform (SNAP)library is written in C++ and social recommendation can be configured using this library functions. Large network and graph can be easily processed by using this library. During computing on nodes, values can be changed dynamically which is most significant advantage. It was released in 2009 as general purpose STL(Standard Template Library).

D. TensorFlow

TensorFlow **[22]** is library which is open source and deep learning, machine learning algorithms are implemented. It can run on multiple CPUs and GPUs and machine learning, neural networks are used in this library. It can be deployed for large scale of data easily. This library is implemented in C++. TensorFlow is very much efficient for object recognition in image. In TensorFlow, dataflow graph permits user to compute independent tasks in parallel. Data is represented in the form of tensors (n-dimensional arrays) and mathematical computations can be easily modelled using n-d arrays. Large scale of data can be trained using this library efficiently. This library is not only suitable for large scale machine learning implementation but also for small scale. It is very simple to build computational graph and running sessions on graphs. API for C++, MATLAB and Python are defined in TensorFlow. Linear regression model which is used to predict dependent variable from the known set of independent variables, can be implemented using TensorFlow.

VI. Conclusion

Recommendation is very crucial tool for social networking sites and business organization. Large scale of unstructured data results in information overload issue which is very confusing for users to select best suitable topic, news, product, movies or music. Recommendation assists users to provide suggestions based on their likings. In this paper content based, collaborative filtering based and hybrid based techniques for recommendations are explained in detail. It was also mentioned that sparsity, cold start and scalability are the issues in recommendation techniques. Several research works have been carried out to improve recommendation accuracy. In this paper, bi-clustering, social trust clustering, deep learning and large scale graph partitioning are elaborated so that reader can understand different methodologies to improve recommendation. Deep learning based recommender systems is core theme of this paper. Convolutional neural network, deep feedforward model, recurrent neural network model and deep belief model are described with their relevance. Large scale graph partitioning is also categorized as random based or trust based partitioning. Very few research works cover comprehensive analysis of sparsity, cold start and scalability issues in single research work. In this paper, these issues are shown to be resolved by deep learning and large scale graph partitioning. Recommendation experiment evaluation is very much necessary to prove accuracy. Standard datasets such as MovieLens, Epinions and FilmTrust is described in detail. Also, MAE (Mean Absolute Error) and RMSE (Root Mean Squared Error) are mentioned as most commonly used metrics for proving recommendation accuracy. Latest technologies such as Pregel, Giraph and SNAP are also elaborated in this paper.

References

[1] P. Gopalan, J.M. Hofman, and D.M. Blei. "Scalable recommendation with poisson factorization." arXiv preprint arXiv:1311.1704, 2013.

[2] C.Gomez-Uribe and N.Hunt, “The Netflix recommender system: Algorithms, business value, and innovation”, ACM Transactions on Management Information Systems (TMIS) 6, 4 , 13, 2016.

[3] J. Davidson, B. Liebald, J. Liu, P. Nandy, T. Vleet, U. Gargi, S. Gupta, Y. He, M. Lambert, B. Livingston, and D. Sampath, “The YouTube Video Recommendation System”, In Proceedings of the Fourth ACM Conference on Recommender Systems (RecSys ’10). ACM, New York, NY, USA, 293–296, 2010.

[4] H. Lee and J. Lee, “Scalable deep learning-based recommendation systems”, ICT Express,2018.

[5] L.H. Ungar and D.P. Foster, “Clustering methods for collaborative filtering” , In AAAI Workshop on Recommendation Systems, pp.112–125, 1998.

[6] T. Hofmann, “Latent semantic models for collaborative filtering”, ACM Trans. Inf. Syst. 22 (1), pp. 89–115, 2014.

[7] S. Zhang, L. Yao and A. Sun, “Deep learning based recommender system: A survey and new perspectives”, arXiv preprint arXiv:1707.07435, 2017.

[8] A. Krizhevsky, I. Sutskever and G. Hinton, “Imagenet classification with deep convolutional neural networks”, Adv. Neural Inf. Process. Syst., 2012.

[9] D. Zhang, C.H. Hsu, M. Chen, Q. Che, N. Xiong, and J. Lloret , “Cold-start recommendation using bi-clustering and fusion for large-scale social recommender systems”, IEEE Transactions on Emerging Topics in Computing, 2(2), 239-250, 2014.

[10] T. DuBois, J. Golbeck, J. Kleint, and A. Srinivasan , “Improving recommendation accuracy by clustering social networks with trust”, Recommender Systems & the Social Web, 532, pp. 1-8,2009.

[11] J.S. Breese, D. Heckerman, and C. Kadie , “Empirical analysis of predictive algorithms for collaborative filtering”, pp. 43–52, 1998.

[12] B.J. Mirza, B.J. Keller, and N. Ramakrishnan, “Studying recommendation algorithms by graph Analysis”,Journal of Intelligent Information Systems,20(2), pp. 131–160, 2003.

[13] J. Liu and C. Wu, “Deep Learning Based Recommendation: A Survey”, pp. 451-458. 10.1007/978-981-10-4154-9_52, 2017.

[14] R. Fakhfakh , A.B. Ammar and C.B. Amar , “Deep Learning-Based Recommendation: Current Issues and Challenges”, International Journal of Advanced Computer Science and Applications (IJACSA), 8, 12,2017.

[15] Q. Dang and C. Ignat, “dTrust: a simple deep learning approach for social recommendation”, The 3rd IEEE International Conference on Collaboration and Internet Computing (CIC-17), 2017.

[16] P. Covington, J. Adams, and E. Sargin, “Deep neural networks for youtube recommendations”, In Proceedings of the 10th ACM Conference on Recommender Systems, ACM, 191–198, 2016.

[17] J. Schmidhuber, “Deep learning in neural networks: An overview,”, Neural Networks, vol. 61, pp. 85–117, 2015.

[18] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 436–444, 2015.

[19] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” ICLR, 2017.

[20] X. Wang and Y. Wang, “Improving content-based and hybrid music recommendation using deep learning”, In Proceedings of the 22nd ACM international conference on Multimedia , pp. 627-636, ACM, 2014.

[21] R. Bell, Y. Koren and C. Volinsky, “ Modeling relationships at multiple scales to improveaccuracy of large recommender systems” in Proceedings of International SIGKDD Conference onKnowledge Discovery and Data Mining pp. 95–104,2007.

[22] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis , J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard and M. Kudlur, “Tensorflow: a system for large-scale machine learning”, In OSDI ,Vol. 16, pp. 265-283,2016.

[23] S. Okura, Y. Tagami, S. Ono and A. Tajima, “Embedding-based news recommendation for millions of users”, In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining , pp. 1933-1942,2017.