IRA-International Journal of Technology & Engineering
ISSN 2455-4480; Vol.03, Issue 03 (2016)
Institute of Research Advances
http://research-advances.org/index.php/IRAJTE

A Genetic Approach for Enhancing Recommender System's Stability

Immidi Kali Pradeep
Research Scholar- KL University, Guntur, Andhra Pradesh, India.

Dr. M Jaya Bhaskar
Associate Professor- KL University, Andhra Pradesh, India.

Dr. K Hima Bindu
Professor –Vishnu Institute of Technology, Bhimavaram Andhra Pradesh, India.

DOI: http://dx.doi.org/10.21013/jte.v3.n3.p5

How to cite this paper:
Kali Pradeep, I., Jaya Bhaskar, D., & Hima Bindu, D. (2016). A Genetic Approach for Enhancing Recommender System's Stability. *IRA-International Journal of Technology & Engineering (ISSN 2455-4480)*, 3(3).
doi:http://dx.doi.org/10.21013/jte.v3.n3.p5

ABSTRACT

With the growing scope of the E-commerce industry, a recommender system places a critical role in predicting correct entity to the user. Accuracy and degree of trust are most important parameters of a recommender system. Many recommender algorithms are proposed in the literature to enhance the accuracy of recommendations to the user. This paper is focused on enhancing the stability of recommender system using genetic and repetitive smoothening approach. Stability determines the degree of trust by the user to use the items recommended by a recommender system. SVD and slope one recommenders systems are used along with the proposed approach and it has been shown that stability is enhanced using the genetic-repetitive algorithm. Genetic approach has shown significant improvements in the field of research and is considered one of the robust techniques.

Index terms: Recommender systems, repetitive smoothening, E-commerce, genetic algorithm,SVD, Slope one.

1. Introduction

E-Commerce recommendations help the customer to find the products they want to buy. Many E-Commerce websites like Amazon, Flip kart, Snapdeal, Vonnik, Myntra uses recommender algorithms to recommend best items to the customer. Recommender system uses server logs, Navigation maps between web pages, hyperlinks, ratings, sharing information on social networks, reviews and ratings given by users as inputs. Out of these, reviews and ratings are explicitly given by users. The trusts of a recommender system have been an important issue over time. Many researchers have published articles about the accuracy of a recommender system. But, the recommender system's performance is also measured in terms of scalability, novelty, and diversity [2].

The accuracy of a recommender system is measured using root mean square error. It defines the difference between predicted preference and true preference of an item. Whereas, Diversity is defined as average dissimilarity between all pair of items recommended to a given user. The novelty of a recommender system generally refers to how different it is with respect to "what has been previously seen", by a specific user, or by a community as a whole.

Recommender systems having good recommender accuracy may not be stable [1].This paper mainly focuses on stability of the recommender system. Papers published in the related previous literature [3] focused on improving the stability of a recommender system using bagging and iterative smoothening. Stability of a recommender system can be defined as the consistency of a recommender system over new data. For example, a recommender system uses training data and recommends items X and Y to the user. After consuming X, This data is added to the training data. After getting trained by this new training data, if the recommender system doesn't recommends item Y to the user, then the system is said to be unstable.

In this literature, the traditional recommender algorithms are wrapped by genetic – repetitive smoothening approach. And, it has been shown that, the stability of the recommender system has been enhanced.

The paper is organized as followed; initially recommender systems and its enhancements have been described and then, existing techniques for recommendation algorithm stability are presented. Then, the proposed genetic-repetitive smoothening is presented. The experimental studies show that the stability of recommender system has been enhanced over original algorithm.

2. Related work:

2.1 Recommender system

Content-based recommender system is one of the traditional approaches [4]. Items rated by the user both implicitly and explicitly are taken as reference and new items are recommended to the user.

Collaborative filtering approach is another traditional approach. Here, users having similar characteristics are grouped to form a neighborhood. And, the actions of a user like his ratings, etc… are analyzed and recommended to another user having same characteristics. The Hybrid approach has been proposed [15] which takes both item level information and user level information to provide good recommendations to the user.

Adding to traditional approaches, some modern recommender systems take the present context of the user and give recommendations [5]. The contextual information about a user may include time, day, location, and mood of the user. This information may be collected using explicit surveys or implicit surveys using timestamp and GPS. For example, a trolley bag is always recommended to the user during a long weekend and during holidays than normal days. Semantic-based recommender system is advancements in recommender systems [6]. Item descriptions and reviews are represented in texts. Many times, the keywords representing an item may not be sufficient for a recommender system to recommend an item to a customer. And, reviews explicitly written by the users are in textual techniques. Text mining techniques like document classification and clustering are helpful to determine the structure of the text. There is a huge literature on text mining; the ontological approach is one of the most known techniques. The stability of a recommender system defines as the resistance of an algorithm's prediction with respect new data. It is measured using root mean square shift [7].

$$\text{RMSS} = \frac{\sqrt{\sum_{(u,i)\,\epsilon P1 \cap P2}(P1\,(u,i) - P2(u,i))^2}}{|\text{P1} \cap \text{P2}|}$$

In the above equation, P1 is the prediction made on original rating data set and P2 is the prediction made on combination of original rating set and predictions from P1. (u,i) denotes user item pair.

Bagging and iterative smoothening techniques were proposed in previous literature [3]. Bagging takes known ratings as training data D with and unknown sample (whose rating has to be predicted) as S/D. The training data is then divided into batches. And, the recommender algorithm is applied on each batch of known rating data.

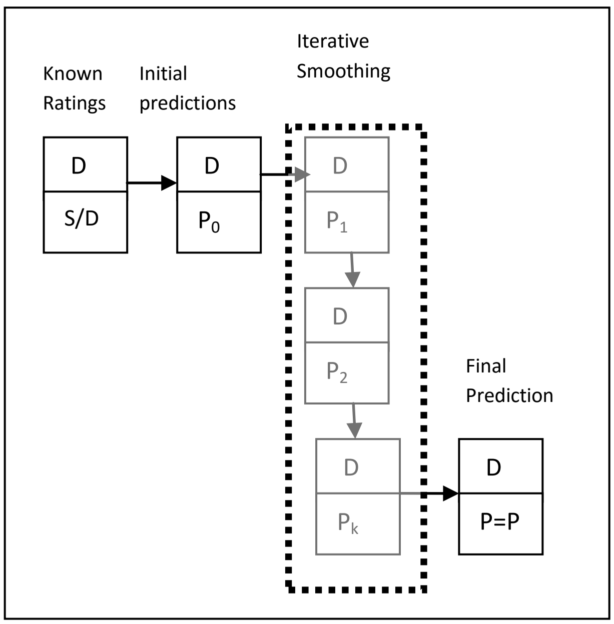

Fig 1: The Iterative smoothening approach for increasing the stability of a recommender system.

The model learned from each batch is applied on unknown rating S/D. And the average predictions of all batches are considered as the final prediction. The bagging can be mathematically written as

$$P(u,i) = \frac{1}{m}\sum_{m=1}^{M} fm(u,i)$$

Here, M is the total number of batches, fm (u,i) is the recommender function, and P(u,i) is the prediction of a new user item pair.

Iterative smoothening is another approach. The Initial model is built on Known training data D using a recommender algorithm. This model is used to predict unknown data S/D. In the second iteration, predicted training data is added to initial training data and a new model is built to predict unknown samples. This process repeats till accurate predictions are obtained.

2.2 Genetic approach

The genetic algorithmic approach was developed by John Holland in early 1960's [8]. A Genetic algorithm works with a concept of fitness evaluation of a chromosome. Genetic algorithm is a form of machine learning technique. Unlike classic algorithms, genetic algorithms produce offsprings. A chromosome is considered as a collection of gene.

In genetic algorithms, cross operator and mutation operator is generally applied on the chromosome. Crossover operator is applied on chromosomes with a good fitness function. It involves swapping genes in two chromosomes to form successor chromosome(s) with better fitness function as compared to its predecessors. Mutation is another important operator in genetic approach. It involves random shuffling of genes within the chromosomes. A Fitness function is used to evaluate the eligibility of a chromosome to survive. It changes from domain to domain. The generalized algorithm for the genetic algorithm is given below.

1) Consider a population of chromosome X.
2) Calculate fitness of each chromosome x1,x2….xn.
3) Select chromosome with good fitness function xk, xm.
4) From the above set of population, pick two chromosoms with good fitness.
5) Apply crossover- Shift some elements from first chromosome to second and vice-versa
6) Apply mutation- Shift the position of elements within the chromosome
7) Repeat step 2
8) Until population with good fitness function is generated.

3. Proposed Approach

Input: Known Rating Data 'R' , Recommender algorithm 'T' , min_acc , min_chrom, Unknown data 'P'.

Algorithm:

Step1: Known Rating data is divided into two parts, training data 'D' and testing data 'C'.
Step2: Use a recommender system 'T' and create a model M using training data 'D'.
Step3: The accuracy of the model 'M' is tested on 'C'.
Step 4: If, model accuracy > min_acc
Step 5: Apply Model 'M' on 'P' (M->P)
Step 6: Divide P ϵ p_i ,n equal batches (p1, p2,…….,pn) and consider each 'p_i' as a chromosome.
Step 7: Repeat for each 'p_i'
Step 8: Consider a chromosome
Step 9: Apply crossover and mutation on chromosomes 'D' and 'p_i' to form 'p_{ic}'.

Step 10: Create a model using recommender algorithm on 'p_{ic}' and check the predictions on C.
Step 11: If, Model accuracy> min_acc- Fitness function
Step 12: count=count+1
Step 13: End If count> min_chrom,
Step 14: Increase classifier accuracy
Step 15: End if
Step 16: End

In Genetic repetitive smoothening technique, the available rating data is divided into training data 'D' and test data 'C'. This training data is taken as input to a recommender algorithm and a model is built. After testing the accuracy of the model on test data, this model is applied on Unknown data 'P'. The predicted rating data 'P' is divided into batches (p_1,p_2,………,p_n).

Consider each 'p_i' as a chromosome and each predicted rating in the chromosome 'p_i' as a gene in the chromosome. In the same way, consider training data D as another chromosome.

Apply crossover and mutation over predicted data and training data. Built a model over this data and apply on testing data 'C'. If accuracy of the model is greater than min_acc defined by the user, the p_i is said to be stable. Repeat the same process for each value of p_i.

If the number of batches crossing min_acc is less than min-chrom, Increase the classifier accuracy by using techniques like cross-validation , bootstrap , random sampling[9] etc… and apply from step 7 again.

4. Experimental study

In our experimental study, Standard Movie Lens 100K Dataset is taken as input. The data is divided into training data(75%) and test data(25%). Two recommender systems are considered, and stability is compared before and after applying our proposed technique.

4.1 Recommender algorithms

Singular value decomposition and weighted slope one approach is used as reference algorithm to show improvements is stability.

4.1.1 Singular value decomposition

In matrix factorization, the matrix is divided into parts in away when combined results in the original matrix. It is used to discover the inherent relationship between two entities in the matrix. For example, two users rate a movie with five rating if both like the hero, heroin or genres etc… these latent features are analyzed by matrix factorization technique.
The singular value decomposition is a matrix factorization technique. In SVD, a m×n matrix is split into 3 parts as follows.

$$\begin{bmatrix} x11 & \cdots & x1n \\ \vdots & \ddots & \vdots \\ xm1 & \cdots & xmn \end{bmatrix}$$
$$= \begin{bmatrix} u11 & \cdots & u1r \\ \vdots & \ddots & \vdots \\ um1 & \cdots & umr \end{bmatrix} \times \begin{bmatrix} s11 & \cdots & s1r \\ \vdots & \ddots & \vdots \\ sr1 & \cdots & srr \end{bmatrix} \times \begin{bmatrix} v11 & \cdots & v1n \\ \vdots & \ddots & \vdots \\ vn1 & \cdots & vrn \end{bmatrix}$$

Here, 'S' is the singular value matrix 'X'. 'R' is the rank of matrix 'X'. Rank is directly proportional to linear independent rows or columns in a matrix. In the present experiment, SVD learns the known rating matrix and unknown rating is estimated as product of user and item matrix. SVD++ is an advanced function of SVD, where it considers both implicit and explicit feedback by the user [12]. Both the algorithms use stochastic gradient descent approach for training purpose. The stability of matrix factorization technique is affected by the sequence of training data fed for learning process.

4.1.2 Slope one approach

Slope one recommender algorithm is both user based and system based [13]. Two parameters should be considered for slope one approach. 1) The Neighborhood of users who rated the same item. 2) The rating information of the user on other items. Slope one approach combines both parameters for recommendations. It is denoted as, $R'_{uj}(j) = R_{uj} + d_{ij}$, Here, 'u' is the user, 'i' is the item (whose rating is to be predicted) , j is item's the rating by user 'u'. R_{uj} is the rating of user 'u' on item 'j'. d_{ij} is the average difference between ratings of item 'i' and 'j' by other users. Another variation of slope one approach is weighted slope one approach. Where, 'R'_{uj} (j)' is proportional to the total number of users having rated both items 'i' and 'j'.

4.2 Dataset

Movie lens 100K dataset [10] is considered for evaluation of proposed system. The dataset has 100000 ratings by 943 users on 1682 items. Each user has rated at least 20 movies. Every movie has a movie id , movie title release data, IMDB url and genre. The user information has information like user id, age, Gender, occupation and zip code.

4.3 Parameters for genetic repetitive smoothing approach

The model trained on the known rating data is applied on unknown rating data to predict the ratings. Here, model accuracy is ensured to be more than 0.9. In case of less accuracy, the classifier accuracy has been enhanced using techniques in [9]. The predictions made by the model on unknown data are considered. These predicted ratings have been divided into n batches and each batch is considered as a chromosome. Based on analysis it has been observed to divide the predicted ratings into 10-30 parts. As the number of batches increases, the number of iterations increases and hence, time complexity increases. Initially, the crossover is applied on a set of

chromosomes from known chromosomes 'D' and predicted chromosome 'pi'. Stratified sampling [14] is used to select d∊D. The advantage of stratified sampling is sample proportion of data is selected from each class in original data. It has been observed that mutation doesn't result much change in stability and is optional. A model is prepared based on p_i U d and its accuracy is checked. If accuracy crosses min_acc, it means the model is satisfying fitness function. In the present experiment, the min_acc value is considered to be 0.85. If in any case, a chromosome is not satisfying min_acc, the fitness is increased by using more samples from 'D' and increasing the classifier accuracy.
The stability of matrix factorization approach is very high at the initial stage and decreases as the number of iterations increases. But, after certain iterations, stability is again increased. Stability of slope one approach remained constant for few iterations and increased as the number of iterations increased.

Stability

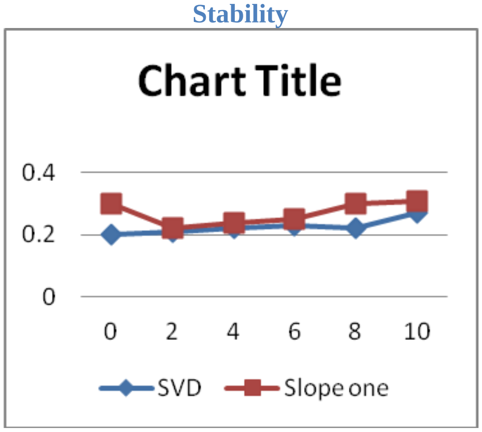

Fig 2: Stability of recommender system over iterations

X axis denotes number of iterations and Y axis is the stability of recommender system over a number of iterations. The recommender system's stability is compared with traditional approach and results are demonstrated below.

Recommender algorithm	Using repetitive smoothening approach (Average)	Without using the approach (Average)
Slope one Approach	0.27	0.53
Singular value decomposition	0.22	0.66

Table 1: Average increase in stability after applying Genetic- recommender smoothening approach

From the above table it has been observed that singular value decomposition has more positive effect with the present approach than slope one approach.

5. Conclusion and Future scope

The stability of recommender system using Genetic-Repetitive smoothening approach is demonstrated. It has been observed that the stability of a recommender system has significant improvement in SVD as compared to slope one approach. In future, this technique can also be applied to other recommender algorithms. This technique is not tested on big databases which may have memory issues. But, at its initial stage, it shows good improvement in enhancing recommender system's stability.

References

1. Adomavicius, G., and Zhang, J. 2010. "On the Stability of Recommendation Algorithms," ACM Conference on Recommender systems, Barcelona, Spain: ACM New York, NY, USA 47-54.
2. G. Shani and A. Gunawardana, "Evaluating recommendation systems," in Recommender Systems Handbook, F. Ricci, et al. Eds. New York, NY, USA: Springer, 2011., pp. 257–294.
3. Adomavicius and Jingjing Zhang ,"Improving Stability of Recommender Systems: A Meta-Algorithmic Approach" IEEE Transactions on knowledge and data engineering, Vol 27, No. 6, June 2015, pp.1573-1587.
4.Michael J. Pazzani and Daniel Billsus, (2007). Content-based Recommendation Systems. In "The Adaptive Web: Methods and Strategies of Web Personalization" (Springer, LNCS #4321), February 1, 2007 pp 325-341.
5. Yize Li, Jiazhong Nie, Yi Zhang ,"Contextual Recommendation based on Text Mining", Coling 2010: Poster Volume, Beijing, August 2010, pp 692–700.
6. Marco de Gemmis, Pasquale Lops,"Semantics-aware Content-based Recommender Systems", recommender system hand book, Springer, pp 119-159.
7. Gediminas Adomavicius and Jingjing Zhang , "Iterative Smoothing Technique for Improving Stability of Recommender Systems",ACM RecSys 2012. September 2012, Dublin, Ireland pp 3-
8. Genetic Algorithm TOOLBOX For Use with MATLAB, Andrew Chipperfield User's Guide.
9. Ping Ning Tan, Introduction to data mining, http://www-users.cs.umn.edu/~kumar /dmbook/ch4.pdf
10. Group lens data, http://grouplens.org/datasets/movielens/100k/
11. Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl," Incremental Singular Value Decomposition Algorithms for Highly Scalable Recommender Systems", Department of Computer Science and Engineering, University of Minnesota
12. Yehuda Koren, Robert Bell and Chris Volinsky, "Matrix factorization techniques for recommender systems" IEEE Computer Society AUGUST 2009, pp 30-37.
13. D Lemire and A maclachlan " slope one predictors for online rating based collaborative filtering", presented at the SIAM DATA MINING, (SDM'05) NEWPORT BEACH, CA, USA, 2005

14. Stratified random sampling http://dissertation.laerd.com/stratified-random-sampling.php

15. Prem Melville ,Raymond J. Mooney and Ramadass Nagarajan, “Content-Boosted Collaborative Filtering for Improved Recommendations” , Proceedings of the Eighteenth National Conference on Artificial Intelligence(AAAI-2002), Edmonton, Canada, July 2002 pp. 187-192.

Author’s Profile

I Kali Pradeep is persuing Ph.D in KL University, Guntur, Andhra Pradesh. He is also working as Assistant Professor in Vishnu Institute of Technology Bhimavaram, Andhra Pradesh. His Research area includes Data mining and Genetic approaches.

Dr. M Jaya Bhaskar is working as Associate Professor in KL University Guntur, Andhra Pradesh. He completed his Ph.d from Krishna University Andhra Pradesh. His research area includes Network Security.

Dr. K Hima Bindu is working as Professor in Vishnu Institute of Technology, Bhimvaram, Andhra Pradesh. She completed her Phd from Central University of Hyderabad. She has a vast experience in decision making systems. Her Research area includes Data Mining.