ISSN:0975-9646

Survey Paper on Analysis of Various Recommendation Algorithms

Dolly Sigroha,Chhavi Rana
Department of Computer Science and Engg
U.I.E.T, MDU,Rohtak

Abstract-Recommender system apply various techniques and prediction algorithm to predict user interest on information, items and services from the tremendous amount of available data on the internet. The paper studies various algorithm in weka and the metrices used to evaluate algorithm performance. The basic algorithm or predictive model we use are – simple linear regression, k-nearest neighbours(kNN), naives bayes, support vector machine. We also review the pearson correlation coefficient algorithm and an associative analysis-based heuristic. The algorithms themselves were implemented from abstract class recommender, which was extended from weka distribution classifier class. The abstract class adds prediction method to the classifier. In addition to introducing these techniques we survey their use in recommender system. The paper also analyze the algorithm of user based and item based techniques and some modern recommendation approaches such as context-aware approach, Semantic-based approaches, cross-domain based approaches, peer-to-peer approaches and cross-lingual approaches.

Keywords-Recommender system, naives bayes,kNN, support vector machine.

INTRODUCTION

Recommender systems, recommendation systems, recommendation engines, recommendation frameworks, recommendation platforms or simply recommender form or work from a specific type of information filtering system technique that attempts to recommend information items (movies, TV program/show/episode, video on demand, music, books, news, images, web pages, scientific literature etc.) or social elements (e.g. people, events or groups) that are likely to be of interest to the user. Typically, a recommender system compares a user profile to some reference characteristics, and seeks to predict the 'rating' or 'preference' that a user would give to an item they had not yet considered. These characteristics may be from the information item (the content-based approach) or the user's social environment (the collaborative filtering).[6]

Researchers at Xerox PARC developed Tapestry, the first recommendation support system [7]. Tapestry was an electronic messaging system that allowed users to either rate messages (“good” or “bad”). . Although Tapestry provided good recommendations, it had one major drawback; the user was required to write complicated queries [4]. The first system to generate automated recommendations was the GroupLens system . The GroupLens system provided users with personalized recommendation on Usenet postings. It recommended articles found interesting by users similar to the target user .

The technique we use for recommendation is data mining. Data mining is defined as the process of discovering patterns in data. The process must be automatic or (more usually) semiautomatic Data mining is the analysis of data and the use of software techniques for finding patterns and regularities in sets of data .Data mining provides a number of algorithms to obtain profiles of users based on historical data, which are used to predict the preferences of new users. The process of applying data mining techniques on web data in order to obtain customer usage patterns is known as web mining The process of data mining typically consists of 3 steps, carried out in succession: Data Preprocessing , Data Analysis, and Result Interpretation.

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)

The input to a Recommender System depends on the type of the employed filtering algorithm. Generally, the input belongs to one of the following categories: Ratings (also called votes), which express the opinion of users on items; Demographic data, which refer to information such as the age, the gender and the education of the users; Content data, which are based on a textual analysis of documents related to the items rated by the user.

The goal of Recommender Systems is to generate suggestions about new items or to predict the utility of a specific item for a particular user.

The output of a Recommender System can be either a Prediction or a Recommendation.

- A Prediction is expressed as a numerical value, ra,j , which represents the anticipated opinion of active user ua for item ij . This predicted value should necessarily be within the same numerical scale (example: 1-bad to 5-excellent) as the input referring to the opinions provided initially by active user ua
- A Recommendation is expressed as a list of N items, where N <= n, which the active user is expected to like the most. The usual approach in that case requires this list to include only items that the active user has not already purchased, viewed or rated.

BACKGROUND

Recommender system involves various algorithms and techniques:

Collaborative filtering algorithms (CF) are algorithms that require the recommendation seekers to express their preferences by rating items. In this algorithm, the roles of recommendation seeker (a user) and preference provider are merged; the more users rate items (or categories), the more accurate the recommendation becomes.

Content based algorithms are algorithms that attempt to recommend items that are similar to items the user liked in the past. They treat the recommendation's problem as a search for related items. Information about each item is stored and used for the recommendations. Items selected for recommendation are items that content correlates the most with the user's preferences [9]. Content based algorithms analyze item descriptions to identify items that are of particular interest

Hybrid approach

Some recommender systems combine different techniques of collaborative approaches and content based approaches. The combination of approaches can proceed in different ways [18]:

1) Seperate implementation of algorithms and joining the results.

2) Utilize some rules of content-based filtering in collaborative approach.

3) Utilize some rules of collaborative filtering in contentbased approach.

4) Create a unified recommender system, that brings together both approaches.

Robin Burke worked out a taxonomy of hybrid recommender systems categorizing them. [19]

Nearest neighbor classifier (kNN) . Given a point to be classified, the *k*NN classifier finds the *k* closest points (nearest neighbors) from the training records. It then assigns the class label according to the class labels of its nearest-neighbors. The underlying idea is that if a record falls in a particular neighborhood where a class label is predominant it is because the record is likely to belong to that very same class.

Naive Bayes

Classifier assumes the probabilistic independence of the attributes – i.e. the presence or absence of a particular attribute is unrelated to the presence or absence of any other.

Support vector machine

The goal of a Support Vector Machine (SVM) classifier [10] is to find a linear hyperplane (decision boundary) that separates the data in such a way that the margin is maximized

We also have various modern recommendation approaches such as context-aware approaches, Semantic-based approaches[20], cross-domain based approaches, peer-to-peer approaches and
cross-lingual approaches.[17]

LITERATURE REVIEW

A lot of researchers have carried out their work in the area of recommender system in the past decade. A comprehensive survey of the rating classifier as well as the prediction algorithm together has not been done in the past few years. This paper presents here a couple of such techniques and enlist the major research work in this area.

Nearest neighbor classifier (kNN) [8] Given a point to be classified, the kNN classifier finds the k closest points (nearest neighbors) from the training records. It then assigns the class label according to the class labels of its nearest-neighbors. The underlying idea is that if a record falls in a particular neighborhood where a class label is predominant it is because the record is likely to belong to that very same class.

Given a query point q for which we want to know its class l, and a training set X = {{x1, l1}...{xn}}, where x j is the j-th element and l j is its class label, the k-nearest neighbors will find a subset Y = {{y1, l1}...{yk}} such that Y □ X and Σk1 d(q,yk) is minimal. Y contains the k points in X which are closest to the query point q. Then, the class label of q is l = f ({l1...lk}). The implementation for k-nearest-neighbor was adapted from some information from Paul Perry[4]. The first step in the algorithm is to compute the mean square difference between each user using ratings they had in common.

Next, the differences are translated into weights.The predicted rating is a weighted sum of the k-nearest neighbors who rated that item.Naïve bayes recommender is a model-based rather than memory-based recommender. The logic for the algorithm was taken from course material and from Witten[1]. Smoothed counts are used. The format of our Weka record and the problem do not line up as well as some of the work done in our class. Every record in the ratings dataset contains a user, an item, and a rating, with the rating taking the role of the class attribute. Generally, naïve Bayes predicts the posterior probability of a class, given the prior probabilities of other attributes.

Ghani and Fano [12], for instance, use a Naive Bayes classifier to implement a content-based RS.

Miyahara and Pazzani [13] implement a RS based on a Naive Bayes classifier, they define two classes: like and don't like. Pronk et al. [14] use a Bayesian Naive Classifier as the base for incorporating user control and improving performance, especially in cold-start situations. Breese et al. [5] implement a Bayesian Network where each node corresponds to each item.

Support Vector Machine (SVM) is primarily a classier method that performs classification tasks by constructing hyperplanes in a multidimensional space that separates cases of different class labels. SVM supports both regression and classification tasks and can handle multiple continuous and categorical variables. To separate the two classes we have many possible boundary lines. Each boundary has an margin. Larger the margin we are less likely to misclassify unknown items.

Xu and Araki [16] used SVM to build a TV program RS. They used information from the Electronic Program Guide (EPG) as features Xia et al.[15] present different approaches to using SVM's for RS in a CF setting. They explore the use of Smoothing Support Vector Machines (SSVM). They also introduce a SSVM-based heuristic (SSVMBH) to iteratively estimate missing elements in the user-item matrix. Oku et al. [11] propose the use of Context-Aware Vector Machines (C-SVM) for context-aware RS. They compare the use of standard SVM, C-SVM and an extension that uses CF as well as C-SVM.

The Pearson algorithm was adapted from Resnick[2]. It uses the Pearson correlation as a similarity metric between vectors of ratings. Typically, memory-based algorithms such as this one use some sort of ratings-based measure to describe the association between users. The measure acts as a weight when predicting a user's rating on an unseen item. This is described by Breese et al[3].

Association Rule-based Recommendation. Assuming that there are n items, I = {i1, i2,..., in}, in the initial user-item matrix, R. A transaction T subset of I is defined as a set of items that are rated or purchased together. An association rule between two sets of items, IX and IY , such that IX, IY subset of I and IX ∩IY = □, states that if items from set IX are present in transaction T, then there is a strong probability that items from set IY would also be present in T. An association rule of that form is often denoted by IX →IY . The quality of association rules is usually evaluated by calculating their support and confidence. The support, s, of a rule measures the occurrence frequency of the rule's pattern IX →IY . Rules with high support are important since they describe a sufficiently large population of items.

The confidence, c, of a rule is a measure of the strength of implication IX → IY . Rules with high confidence are important because their prediction of the outcome is normally sufficiently accurate.

CONCLUSIONS

This paper presented the various techniques and algorithm to build the recommender system and to improve the performance and accuracy of the recommender system. We reviewed various algorithms such as nearest neighbor, support vector machine, naïve bayes. We also presented the Pearson correlation coefficient algorithm and an associative analysis-based heuristic. We also introduce various modern recommendation approaches such as context-aware approaches, Semantic-based approaches, cross-domain based approaches, peer-to-peer approaches and cross-lingual approaches. We have also uncovered areas that are open to many further improvements, and where there is still much exciting and relevant research to be done in coming years.

REFERENCES

[1] Witten I. H. and Frank I. *Data Mining*, Morgan Kaufman Publishers, San Francisco, 2000.

[2] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, John Riedl, GroupLens: an open architecture for collaborative filtering of netnews, Proceedings of the 1994 ACM conference on Computer supported cooperative work, p.175-186, October 22-26, 1994, Chapel Hill, North Carolina, United States

[3] John S. Breese, David Heckerman and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence, pages 43-52, July 1998

[4] Deshpande, M., and Karypis, G. Item-based top-<i>n</i> recommendation algorithms. ACM Trans. Inf. Syst. 22, 1 (2004), 143-177.

[5]. Breese, J., Heckerman, D., and Kadie, C., Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence*, page 4352, 1998.

[6] http://en.wikipedia.org/wiki/Recommender_system

[7]. Goldberg, D., Nichols, D., Oki, B.M. and Terry, D. Using Collaborative Filtering to Weave an Information Tapestry. *Communications of the ACM*, 35, 12 (December 1992), 51-60.

[8]. Cover, T.,and Hart, P., Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, 13(1):21–27, 1967

[9] van Meteren, R., and van Someren, M. Using content-based _ltering for recommendation.

[10]. Cristianini, N.,and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, March 2000.

[11]. K. O. et al. Context-aware svm for context-dependent information recommendation. In *International Conference On Mobile Data Management*, 2006.

[12]. Ghani, R.,and Fano, A., Building recommender systems using a knowledge base of product semantics. In *In 2nd International Conference on Adaptive Hypermedia and Adaptive Web Based Systems*, 2002

[13]. Miyahara, K.,and Pazzani, M.J., Collaborative filtering with the simple bayesian classifier. In *Pacific Rim International Conference on Artificial Intelligence*, 2000.

[14]. Pronk, V., Verhaegh, W., Proidl, A., and Tiemann, M., Incorporating user control into recommender systems based on naive bayesian classification. In *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*, pages 73–80, 2007

[15]. Xia, Z., Dong, Y., and Xing, G., Support vector machines for collaborative filtering. In *ACMSE 44: Proceedings of the 44th annual Southeast regional conference*, pages 169–174, New York, NY, USA, 2006. ACM.

[16]. Xu, J.,and Araki, K., A svm-based personal recommendation system for tv programs. In *Multi-Media Modelling Conference Proceedings*, 2006.

[17].Algorithms and Methods in Recommender Systems,Daniar Asanov Berlin Institute of Technology Berlin, Germany.

[18] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," IEEE Trans. on Knowl. and Data Eng., vol. 17, pp. 734–749, June 2005. [Online]. Available http://dx.doi.org/10.1109/TKDE.2005.99

[19] R. Burke, "The adaptive web," P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds. Berlin, Heidelberg: Springer-Verlag, 2007, ch. Hybrid web recommender systems, pp. 377–408. [Online]. Available: http://portal.acm.org/citation.cfm?id=1768197.1768211.

[20]. A. Elgohary, H. Nomir, I. Sabek, M. Samir, M. Badawy, and N. A. Yousri, "Wiki-rec: A semantic-based recommendation system using wikipedia as an ontology," in Intelligent Systems Design and Applications (ISDA), 2010 10th International Conference on, 29 2010