International Journal of Engineering Research and General Science Volume 3, Issue 4, July-August, 2015
ISSN 2091-2730

A Decision Tree Based Record Linkage for Recommendation Systems

MS. N. S. Sheth[1], Asst. Prof. A. R. Deshpande[2].

Department of Computer Engineering,
PICT, Pune, Maharashtra, India.
nikita86.sheth@gmail.com[1], ardeshpande@pict.edu[2].

Abstract— Record linkage merges all the records relating to the same entity from multiple datasets, at the entity level. It is the initial data preparation phase for most of the database projects. Traditionally one to one data linkage is performed among the entities of same type with common unique identifier. The proposed one to many and/or many to many record linkage method is able to link the entities of same or different types with or without availability of common unique identifier. Here a probabilistic record linkage which is based on clustering tree construction that classifies the matching entities by linkage. The tree construction is based on the one of the splitting criterion for the best attribute selection that partitions dataset at each node of the tree. Record Linkage is used in recommender system domain to produce list of recommendations at each leaf of the tree. It is used for matching new user with their product expectations in order to produce list of recommendations. In propose method a decision tree based record linkage is applied to generate book recommendations. This technique is also useful in solving cold start and new user problems.

Keywords— Decision tree, Classification, Clustering, Splitting Criterion, Record Linkage, Model Based, Recommendation System.

I. INTRODUCTION

A record linkage merges all records relating to the same or different entity. The linkage is required for data analysis or mining the information from multiple data sources. A record linkage is either deterministic or probabilistic. Deterministic or rule based linkage generates links based on the common unique identifier among the data sets. Probabilistic linkage is based on the probability for the two given records referring to the same entity. A record linkage is of two types: one-to-one (entity from one data set associates with a single matching entity in another data set) and one-to-many (entity from one data set associate with a group of matching entity in another data set). Different machine learning techniques like classification, clustering are useful to perform record linkage.

A decision tree is a classification algorithm that classifies the labeled data set into predefined classes. It produces a tree structure where internal node represents test on an attribute, each branch represents outcome of the test and each leaf node represents the class label. A path from root to leaf represents classification rules. While in clustering trees each node represents a cluster or concept. Clustering Tree is a decision tree that partitions instances into homogeneous clusters at each node. Decision tree algorithm is applicable in recommendation system to produce recommendations.

Recommender system analyses customer needs and predict items by generating list of recommendations. Recommendation systems rely on ratings for item provided by user. It reduces searching cost. It uses data mining technique to discover useful patterns or recommendations. Recommender system provides proposals to the user if user does not know about existence of item [21]. Recommender system methods can be broadly classified into three types: Content Based, Collaborative Filtering and Hybrid approach. A Content Based approach provides recommendations similar to users past preferred items based on users past preferences. A Collaborative Filtering approach provides recommendations that the user or its peer with similar attributes preferred previously. A Hybrid approach combines collaborative and content based approach to overcome their limitations. Recommender system has to face cold start and new user problem.

New user or user cold start problem where the user has to rate sufficient number of items to gets the accurate recommendations. Item cold start problem is arise if the purchase frequency of particular item is low, then the system can not recommend other items to users who have purchase it.

The proposed Recommendation Using Record Linkage (RURL) method performs data linkage with or without sharing common unique identifier and produces a clustering tree, where each of the leaf contains a cluster of matching or nonmatching instances instead

of a single classification. Clusters are created by selecting the attributes from user table using one of the splitting criteria, while the clustered data is from the item table that is linked to it. It is based on C4.5 decision tree algorithm. It is one class approach as only positive or matching instances are considered to build a recommender system after linkage [1]. This technique handles cold start and new user problems.

The rest of the paper is organized as follows. Section II is survey of related works on record linkage, one class decision tree construction and recommender systems. Section III summarizes system design and algorithm and section IV describes the experimental setup, datasets used, evaluation and results. Finally, the last section V concludes this paper.

II. RELATED WORK

A. Record Linkage

The Record linkage finds the matching records among different datasets which may or may not share common identifier i.e. key. Previously, one to one data linkage was implemented using an SVM classifier algorithm which separate outs matching and nonmatching record pairs relating to the same entity. Probabilistic approaches used to determine the probability of a record pair being match or nonmatch using expectation maximization or maximum-likelihood estimation for complete data [6]. A FS (Fellegi–Sunter) record linkage proposed in [8], is also probabilistic approach, which uses log likelihood ratio for finding similarities between records. An approximate comparator extends the FS method to improve linkage process. A one to many record linkage stated in [9], is based on expectation maximization algorithm also performs the probabilistic linkage by calculating the probability for record being match.

The selection measure i.e. splitting criterion, plays an important role in construction of decision tree as it determines the best split of instances at a given node. Each decision tree induction algorithm uses distinct splitting criteria like Information gain, gain ratio, gini index for finding the best splitting attribute [12].

An automated record linkage method [11] is used to find the matching records. It is based on C4.5 decision tree classification algorithm where tree is constructed using different string comparison methods. Clerical review is required for possible links. C5.0 decision tree algorithm is used to link Genealogical records and performs one to many data linkage [10]. Here decision tree was constructed using Information gain as splitting criteria to classify the records into match or mismatch classes.

Top-Down Induction of Clustering tree (TIC) system stated in [13], is based on construction of first order logical decision tree and clusters. Here, tree node is generated using first order logic and it represents a cluster or a concept. It integrates instance based learning and inductive logic programming to obtain a clustering system. A proposed CLTree (Clustering based on decision Tree) [14] is based on decision tree induction that partitioned data space into clusters and empty region. A simplified tree and meaningful clusters are obtained by cluster tree pruning. A lookahead gain criterion (relative density) is used by this algorithm for better partitioning of clusters and to avoid loss of data points.

With the help of clustering, a complex distribution is divided into simpler one in [15]. Here, for each simplified cluster a model is built using data from a single class to perform one class classification. OcVFDT (One class Very Fast Decision Tree) algorithm proposed in [17] is applied on fully labeled data stream and it is based on VFDT (Very Fast Decision Tree). It requires less memory space as it scans the input only once. OcVFDT is extended to PUVFDT (Positive and Unlabeled Vey Fast Decision Tree) [18] that deals with numeric and discrete attributes of positive and unlabeled data streams using PosLevel parameter.

C4.5 decision tree based OCCT (One Class Clutering tree) [1] algorithm represents a cluster at each leaf of the tree. It performs one-to-many linkage that matches entities of different types. It also explains coarse and fine grained jaccard coefficient, least probable intersections, and maximum likelihood estimation as four different splitting criteria for decision tree induction for only matching pairs.

B. Recommender System

The recommendation system analyzes data from a particular domain like movie, book, music or insurance plan to find the items that user is looking for and to produce a predicted likeliness score or a list of recommended items for a given user. Different data mining algorithms can be used based on each use case.

Decision tree based Movie Recommender System proposed in [19], builds a tree using least probable intersections as a splitting criterion. Here, decision tree construction is based on ID3 algorithms as well as Rating matrix <UserID, ItemID, Rating>. A constructed tree produces list of recommended items at its leaf node with its weighted average by only single traversal of the tree.

A Movie Recommender System proposed in [4], where decision tree is constructed that represent user preference and attempts to solve sparsity, scalability (partially) and transparency problem. It considers both credit as well as candidate preferences.

Collaborative Recommender System [5] is based on RFM (Recency, Frequency, and Monetry) model and decision tree induction. The RFM score improves the accuracy of recommendation and NRS (Normalized Relative Spending) value finds customer's preferences.

A SemTree [3] is an ontology-based decision tree that uses domain ontology to improve the effectiveness of the decision tree. It also uses a reasoner to split instances with more generalized features for better performance. Feature with higher information gain is used to build decision tree node. Recommendations are generated on the basis of user model and rank of an item.

A multi-relational model used in [26] is analyzing multiple Social network information. It is based on random walk with restart algorithm and generic algorithm is used to achieve better recommendation. Five matrices Contact, Rating, Common contact, Book tag, Readers tag are factorized and integrated to solve cold start problem by this model.

A proposed Adaptive Bootstrapping of Recommender System [27] is based on decision tree construction where root mean square error is used as a splitting criterion. It deals with new user and cold start problem. Here, a new user follows the path starting at the root of the tree by asking the questions associated with the nodes along the path and traversing the labeled edges as per answers to get the recommendations at the leaf of the tree.

TABLE I: COMPARISONS BETWEEN DIFFERENT SPLITTING CRITERIA

Sr. No.	Method Used	Splitting Criteria	Improvements	Limitations	Dataset	Ref. No.
1.	Binary Decision Tree using Rating matrix (Hybrid Recommendation system)	Least probable intersection size.	Requires only single traversal of DT for producing recommendation list.	Need to Construct rating matrix separately.	MovieLens	[2]
2.	Ontology based Reasoning	Information Gain	Uses a reasoner and ontology concept.	Model is based on overall ratings of item.	Netflix Prize Movie	[3]
3.	C4.5 Decision Tree Algorithm (Content based Recommendation system)	Information Gain	Solve sparsity, scalability (partially) and transparency problem.	For item evaluation recommendation list need to examine each time.	MovieLens	[4]
4.	C4.5 Decision Tree using NRS value and Clustering using RFM score (Hybrid Recommendation system)	Transaction Matrix	It is based on RFM model and decision tree induction.	Classification is based on customer transaction matrix.	Retail Business	[5]
5.	Adaptive Bootstrapping (Collaborative Recommendation system)	Root mean Square Error	It deals with cold start problem and based on dynamic interview questions.	Large computations are required for unknown users.	Netflix	[6]

III. RECOMMENDATIONS USING RECORD LINKAGE MODEL INDUCTION

A. System Design

This section gives outline of the Recommendations Using Record Linkage (RURL) technique. Linkage based recommendation system consists of three modules: one class decision tree induction module, clustering at the leaf model and recommendation list generation module as shown in Fig.1.

One class decision tree construction is based on the splitting criteria to select best splitting attribute at each level of the tree. Splitting criteria is heuristic for attribute selection to choose the best split of the dataset at each internal node and measures the similarity between two record sets. The attribute with highest/lowest score is used as the next split of the tree [12].

Decision tree induction for record linkage and recommendation generation is based on C4.5 decision tree algorithm. It uses coarse grained jaccard coefficient, average gain or normalized gain as a splitting criterion to choose the best splitting attribute at each level of the tree. Coarse Grained Jaccard is the ratio of records belonging to the both subsets to the total number of records. Average Gain is the ratio of the information gain to the number of attribute values. Normalized Gain is the ratio of the information gain to the log of number of partitions created due to split [20]. Let S be a set of instances with p number of instances of class P and n number of instances of class N, A be the set of attributes, Av be the attribute values, T_{Av} be the subset of instances having attribute value Av, m be the number of partitions created due to split. Coarse Grained Jaccard coefficient (CGJ), Average Gain Ratio (AGR) and Normalized gain Ratio (NGR) measure is defined as,

$$\text{CoarseGrainedJaccard}(S, A) = \frac{|T_{A_v} \cap T_{A_{!v}}|}{|T_{A_v} \cup T_{A_{!v}}|} \quad (1)$$

$$\text{AverageGain}(S, A) = \frac{\text{Gain}(S,A)}{|A_v|} \quad (2)$$

$$\text{NormalizedGain}(S,A) = \frac{\text{Gain}(S,A)}{\log_2 m}, m \geq 2 \quad (3)$$

where,

$$\text{Entropy}(S) = -\frac{p}{p+n}\log_2\frac{p}{p+n} - \frac{n}{p+n}\log_2\frac{n}{p+n}$$

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{i=1}^{v} \frac{|S_i|}{|S|} \text{Entropy}(S_i)$$

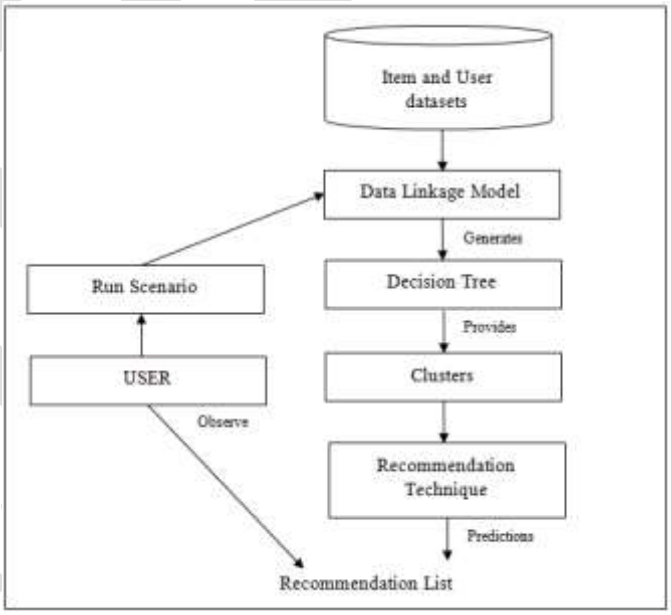

Fig. 1: Block Diagram

A decision tree is constructed in an incremental way where the inner nodes of the tree consist of attributes from user table and a leaf node consists of a set or a group of matching records from another table. Each leaf is represented by MLE of the records from the item table. It takes records from two datasets (Cartesian product) as input. A record linkage tests for each possible pair of new or test records against the linkage model to determine that a pair is a match or not. Tree Based Record Linkage for Recommendation System considers only positive or matching instances for recursive decision tree induction and recommendation generations.

This technique requires only single tree traversal to obtain the list of recommended items at leaf nodes of the tree. It is a model based hybrid recommendation approach as only single tree is constructed by choosing one of the best attribute from user table and leaf

nodes represent matching instances from the item table that is the content of an item. It also handles cold start and new user problems as the new user is provided with the list of recommendations without rating for sufficient items.

B. Algorithm:

Algorithm for Recommendations Using Record Linkage is as follows,

Input: T_{AB} , set of records (r_i) from user and item table,

a_i , Set of attributes from user table.

b_i , Set of attributes from item table.

r_a , Set of records from user table.

r_b , Set of records from item table.

Output: T, Clustering Tree with recommendations.

Method:

1. Tree, T= { };
2. If $a_i = Ø$
3. Find matching models by performing record linkage between table A and B;
4. else
5. For all $a_i \in A$, calculate information split criteria value on each a_i;
6. a'= a best splitting attribute by using step 5;
7. For all $v_i \in a'$
8. Build decision tree T_{vi}, by applying splitting criteria at each node, sub node of the tree.
9. For each leaf of the tree, create models M using MLE for r_b that provides clusters of records from table B.
10. a_i = accept the input from user.
11. Traversing the Tree from root to leaf to provide the list of recommendations at the leaf of the decision tree.

IV. EXPERIMENTAL SETUP AND RESULTS

A Tree Based Record Linkage for Recommendation System is implemented by using machine learning open source weka libraries in java. An implementation is carried out with the help of java Eclipse IDE, apache tomcat and mysql database.

This system generates list of recommendations of books to the new user that are expected to be liked. The experiment is carried out using Book-Crossing dataset which is collected by Cai-Nicolas Ziegler. It contains ratings provided by users about the books. A user rated for a fixed number of books and a book rated by fixed number of users is filtered out as a dataset. Book-Crossing dataset provides a rating for each book on the scale of 0 to 10 (dislike to like) which is converted to binary scale by calculating the average rating for each book and also age is categories into ten intervals A to J .

The training phase trained the labeled examples by performing data linkage using decision tree induction and cluster model formations at each leaf of the tree. Only positive or matching instances are considered further to construct a decision tree for recommendation system. The testing phase evaluates the model by applying it on the untrained or new records.

The system is evaluated by calculating precision, recall and tree. Tree size is based on number of nodes of the tree. The different splitting criteria such as average gain ratio (AGR), normalized gain ratio (NGR) and coarse grained jaccard coefficient (CGJ) are compared using precision, recall and tree size with respect different data folds or database sizes. Fig. 2 shows that AGR and NGR produces tree with same number of nodes. Size of tree for AGR is smaller and for J48 greater.

Fig. 3 and Fig. 4 shows that precision and recall curve for AGR is greater while precision and recall for CGJ and NGR is smaller as well as same.

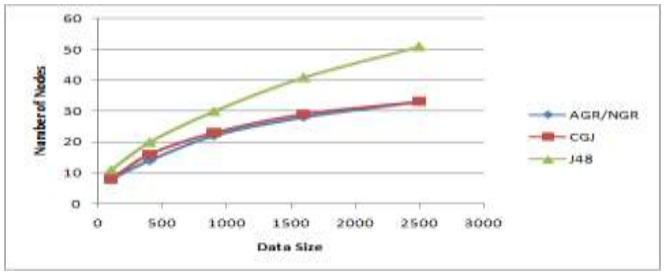

Fig. 2 Curve for Total Number of Tree Nodes

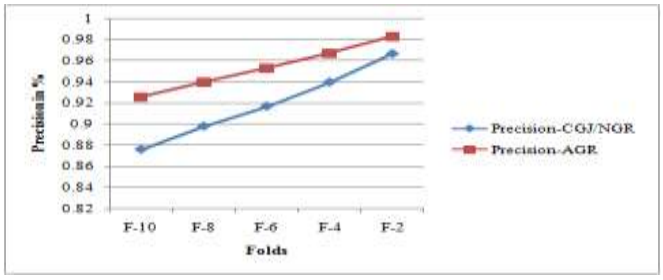

Fig. 3 Precision Curve for k-Folds

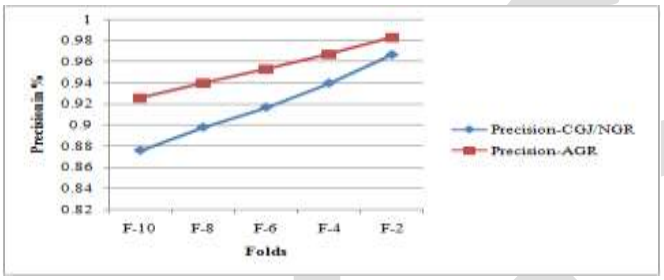

Fig. 4 Recall Curve for k-Folds

V. CONCLUSION

One class decision tree based record linkage model links the records and classifies them into matching and non matching group of instances, by adding class attribute if it is not available. It is guided by one of the splitting criteria from average gain, normalized gain or course gain jaccard correlation coefficient for attribute selection at each internal tree node. Average gain ratio criteria gives better results for precision, recall and tree size as compared to normalized gain or course gain jaccard correlation coefficient.

Instead of computing similarities between users and items, linkage based recommender system constructs a tree model. A decision tree algorithm is used construct a tree model so that new user follows a path starting at root node and traversing the edges labeled by user attributes values to generate recommendations. A list of recommended items is obtained at tree's leaf node by only single tree traversal. The relationship between user features and recommended items is transparent as tree model itself represents them. It reduces amount of search required by the user to search the expected book.

REFERENCES:

[1] Ma'ayan Dror, Asaf Shabtai, Lior Rokach, and Yuval Elovici, "OCCT: A One -Clustering Tree for Implementing One-To-Many Data Linkage," IEEE Transactions On Knowledge And Data Engineering, vol. 26, pp. 682 - 697, March 2014.
[2] P. Christen and K. Goiser, "Quality and Complexity Measures for Data Linkage and Deduplication," Quality Measures in Data Mining, vol. 43, pp. 127 - 151, 2007.
[3] J. Han, M. Kamber, J. Pei, "Data Mining, Concepts and Techniques," Third Edition, Elsevier.
[4] M. H. Dunham, "Data Mining, Introductory and Advanced Topics," Pearson Education.
[5] S. B. Kotsiantis, "Decision trees: a recent overview," Artificial Intelligence Review, Springer, vol. 39, pp. 261 - 283, April 2013.
[6] A. Franco-Arcega, J. A. Carrasco-Ochoa, G. Sanchez Diaz, J. Fco. Martínez-Trinidad, "Decision tree induction using a fast splitting attribute selection for large datasets," Expert Systems with Applications, vol. 38, pp.14290 - 14300 October 2011.
[7] V. Torra and J. Domingo-Ferrer, "Record Linkage Methods for Multidatabase Data Mining," vol. 123, pp. 101-132, 2003.
[8] Scott L. DuVall, Rich ard A. Kerber , Alun Thomas , "Extending the Fellegi–Sunter probabilistic record linkage method for approximate field comparators," Journal of Biomedical Informatics 43,pp. 24–30, 2010.
[9] A. J. Storkey, C.K.I. Williams, E. Taylor, and R.G. Mann, "An Expectation Maximization Algorithm for One-to-Many Record Linkage," University of Edinburgh Informatics Research Report, 2005.
[10] S. Ivie, G. Henry, H. Gatrell, and C. Giraud-Carrier, "A Metric-Based Machine Learning Approach to Genealogical Record Linkage," Proceeding Seventh Annual Workshop Technology for Family History and Genealogical Research, 2007.

[11] P. Christen and K. Goiser, "Towards Automated Data Linkage and Deduplication," Technical report, Australian National University, 2005.
[12] N. S. Sheth, A. R. Deshpande, "A Review of Splitting Criteria for Decision Tree Induction," CIIT, 2015, in press.
[13] H. Blockeel, L. D. Raedt, and J. Ramon, "Top-Down Induction of Clustering Trees," Proceedings International Conference on Machine Learning, pp. 55 - 63, July 1998.
[14] Bing Liu, Yiyuan Xia, Philip S. Yu, "Clustering Through Decision Tree Construction," Proceedings ACM International Conference on Information and Knowledge Management, pp. 20 - 29, 2000.
[15] Shiven Sharma, Colin Bellinger, and Nathalie Japkowicz, "Clustering Based One-Class Classification for Compliance Verification of the Comprehensive Nuclear-Test-Ban Treaty," Proceeding Conference on Artificial Intelligence, vol. 7310, pp. 181 - 193, May 2012.
[16] Francois Denis, Remi Gilleron, Fabien Letouzey, "Learning from positive and unlabeled examples," Theoretical Computer Science, pp.70 – 83, 2005.
[17] C. Li, Y. Zhang, and X. Li, "OcVFDT: One-class Very Fast Decision Tree for One-class Classification of Data Streams," Proceeding Third International Workshop Knowledge Discovery from Sensor Data, pp. 79 - 86, January 2009.
[18] Xiangju Qin, Yang Zhang, Chen Li, Xue Li, "Learning from data streams with only positive and unlabeled data," Journal of Intelligent Information Systems, vol. 40, pp. 405 - 430, June 2013.
[19] W. Dianhong, J. Liangxiao, "An improved attribute selection measure for decision tree induction," Fourth International Conference Proceedings on Fuzzy Systems and Knowledge Discovery, vol. 4, pp. 654 - 658, August 2007.
[20] Xavier Amatriain, Alejandro Jaimes, Nuria Oliver, and Josep M. Pujol, "Data Mining Methods for Recommender Systems."Recommender Systems Handbook, Springer, 2011.
[21] Deuk Hee Park, Hyea Kyeong Kim, Il Young Choi, Jae Kyeong Kim, "A literature review and classification of recommender systems research." Expert Systems with Applications, 2012.
[22] Amir Meisels, Karl Luke, Lior Rokach, "A Decision Tree Based Recommender System." Innovative Internet Commodity services, 2010.
[23] A. Bouza, G. Reif, A. Bernstein, and H. Gall, "Semtree: Ontology-Based Decision Tree Algorithm for Recommender Systems," Proceeding International Semantic Web Conference, 2008.
[24] P. Li and S. Yamada, "A Movie Recommender System Based on Inductive Learning," Proceeding. IEEE Conference on Cybernetics and Intelligent Systems, vol. 1, pp. 318 - 323, December 2004.
[25] S. L. Lee, "Commodity Recommendations of Retail Business Based on Decision Tree Induction," Expert Systems with Applications, vol. 37, pp. 3685 - 3694, May 2010.
[26] Qiuzi Shangguan, Liang Hu, Jian Cao, Guandong Xu, "Book Recommendation Based On Joint Multi-Relational Model," Proceeding IEEE Conference Cloud and Green Computing, vol. 37, pp. 523 - 530, November 2012.
[27] N. Golbandi, Y. Koren, and R. Lempel, "Adaptive Bootstrapping of Recommender Systems Using Decision Trees," Proceeding Fourth ACM International Conference on Web Search and Data Mining, pp. 595 - 604, 2011.
[28] Vassilios S. Verykios, Mohamed G. Elfeky, Ahmed K. Elmagarmid, Munir Cochinwala, Sid Dalal, "On the Accuracy and Completeness of the Record Matching Process," Proceedings of the 2000 Conference on Information Quality, 2000