International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-1 Issue-1, October 2011

An Efficient Ranking Based Clustering Algorithm

Dipak R.Pardhi, Akhilesh A.Waoo

Abstract: ***There are several databases, which contain large information about research publications in various fields, for examples, DBLP in computer science and PubMed in medical science. US Census data set which contains information with 68 categorical attributes, which is very complex to get the information. Zoo data set which having information with 17 attributes, Plant Cell Signalling data set which describes the interactions of the nodes within the plant signalling network by considering 43 different attributes. Each such database forms an immense size of information network connecting in very complex ways.***

In this work, we are proposing an approach for "information network mining" on such a database. We consider DBLP as an example.The database contains information about research papers, authors, conferences and journals. It also includes the date, year and the place of publication of particular journals and conferences. Various users have very specific personalized search criteria for profiling such patterns and verifying the interest, we are proposed an algorithm RBC_A, so that- (1) In-depth information about research, such as the clustering of conferences due to their sharing of many common authors can be categorized; (2) The reputation of a conference can be evaluated; finally (3) Time relevant information can be inferred. The above have been addressed in the design and development of this work.

Keywords: Information Network, Data Mining, Profiling, Ranking, Clustering, Classifications, Associations, User interface.

I. INTRODUCTION

As we know that, database is a coherent collection of data of one or more related organization. A database management system (DBMS) is mega software designed to assist in managing, maintaining and utilizing large collection of data. Data mining or knowledge discovery in database as it is also known is the non-trivial extraction of implicit, previously unknown and potentially useful information from the data by a number of technical approaches such as clustering, data summarization, finding classifications, finding associations. Classification involves finding rules that partition the data into disjoint groups. An association is an interconnection between two or more types of objects from large database [9]. Now in current studies, data mining on a information network becomes an important issue. Information Network consists of rich information. For Example, DBLP dataset [3] which contains information about research papers, each written by a group of authors, using a set of terms and published in a venue (a conference or a journal). Clustering and ranking on such databases again a very important research area. Clustering is a method of grouping data into different groups, so that the data in each group share similar trends and patterns. A ranking is a relationship between a set of items such that, for any two items, the first is either 'ranked higher than', 'ranked lower than' or 'ranked equal to' the second. Clustering and ranking are often regarded as orthogonal techniques, each of which is applied separately to information network analysis.

Several challenging issues regarding such databases are-

1) Connecting in very complex ways;
2) Contain more than enough information about publications in different fields;
3) Data is associated with multiple hierarchies;
4) In-depth information about research, such as the clustering of conferences due to their shearing of many common authors.

To overcome these problems, we proposed a new algorithm RBC_A with different features.

The main contributions of our paper are as follows-

1. We propose a new algorithm RBC_A of data mining for mining large information.
2. We provide profiling including profiles for authors, conferences, titles on the basis of classification technique.
3. We develop ranking on the basis of association between objects by finding candidate set, support, support count and frequent sets with the help of priori algorithm.
4. We perform clustering-based on ranking and display list of authors in specified cluster size.
5. We provide user-friendly interface for users to browse and comprehend the information derived from the above analyses.
6. We perform comparison of different databases with the help of proposed algorithm.

II. ARCHITETURE OF PROPOSED SYSTEM

Figure 1 shows three layers architecture of the proposed system; Information Extraction and Analysis, Functionality and User Interface.

Manuscript published on 30 October 2011.
* Correspondence Author (s)
Dipak R.Pardhi*, *M.Tech Student, Department of C.S.E, Bansal Institute of Science & Technology, Bhopal (M.P) India, (email: dipakpardhi@yahoo.co.in)*
Akhilesh A.Waoo, *Asst. Prof. M.Tech Department of C.S.E, Bansal Institute of Science & Technology, Bhopal (M.P) India, (email: akhilesh_waoo@rediffmail.com)*

The bottom layer is the information extraction and analysis which takes a DBLP database [3] which is in XML format, extract information from database and analyse that information.
The middle layer is the functionality, which implements the major functions like profiling, ranking and clustering.

Profiling is the process of getting relevant information about the particular object present in the database. Ranking is the process of assigning priority to object according to some classifications present in the database. Clustering is the process of grouping particular objects on the basis of ranking.

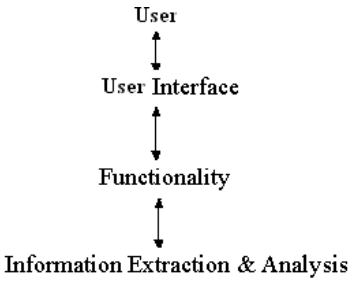


Figure 1: Three Layer Architecture of Proposed System

The top layer is a user interface, which interacts with user and responds to their request.

A. Information Extraction and Analysis

This is the bottom layer in which extraction of information and analysis of that information takes place. It takes a DBLP database which is XML data and extracts information. Finally we get total information in table form.
Table contains different columns like number of authors is given with their titles of paper, conferences and journals published by author, venue or the place of publication, the conference or journal details, finally year and date of publication.

B. Functionality (Major Functional Modules)

This is middle layer in which the major modules such as profiling, ranking & clustering are present. This is done when a user is browsing the DBLP data set [3]. This layer is carried out with the help of classification rules and association rule algorithm.

C. Profiling

Profiling is the process of getting relevant information of objects present in the database. For example, if we consider profile of an author then we get details of that author including title of paper published by that author, venue or the place of publication, conference or journal selection, year and date of that particular paper and the list of co-authors which are working with that author.
In this work, for getting profiling of particular object from the database, we use Naïve Bayes Classification technique [10], in which different classifications are carried out. Some examples of such classifications by considering DBLP database are getting titles published in particular year, getting titles by particular author, getting titles at particular venue, getting titles on particular date etc. Naive Bayes Classification technique does not use rules, a decision tree or any other explicit representation of the classifier. Rather, it uses the branch of Mathematics known as probability theory to find the most likely of the possible classifications.

D. Ranking

Ranking aims at giving conferences or authors with higher authority higher rank. It's very helpful for users to quickly navigate to important objects. For example, Authors having more number of papers are always given higher priority to be displayed than less number of papers published author.
For ranking purpose, we find association or interconnection between two or more types of objects then find the support, candidate set, frequent set and frequent count of each and every object present in the database. From frequent count of each object, apply ordering we get ranking easily. For candidate set generation, frequent set generation and frequent count, we use Priori Algorithm [9].Finally we get frequent count of each and every object present in the database. Apply ordering on this frequent count set, we get ranking in which authors published many papers having higher rank.

E. *Clustering*

Clustering is a useful data mining technique for the discovery of data distribution and patterns in the underlying data. It is a method of grouping data into different groups, so that the data in each group share similar trends and patterns [9].
In this work, Clustering is takes place with the help of ranking. In previous section, we perform ranking in which we get the maximum number of authors published many papers. We store that maximum number, then after giving the cluster size, calculate first the cluster size is zero or not. If value is zero or other then adjust the size according to condition with the help of maximum count number. Finally display the list of authors by initializing start count and end count.

F. User Interface

User-friendly interface and visualization packages from the top-layer of the system and will play an essential role in its usability. The interface allows users to conveniently show the details.The design encourages user interaction and provides various facilities for explorative and multidimensional analysis of the data, including profiling, ranking and clustering. The system provides user-friendly interface so that a user can interact with the system directly. The design makes the user's interaction as simple and efficient as possible.

III. PROPOSED WORK

A. Previous Work with respect to RBC_A data mining

In previous work, a BibNetMiner system has been designed for sophisticated information network mining on databases. In that work, they have taken a DBLP database as an example and demonstrate several attractive functions of BibNetMiner, including clustering; ranking and profiling of conferences and authors based on the research sub-fields [1].
In LinkClus system,

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

the similarities between two objects is measured based on the similarities between the objects linked with them. Linkclus is an efficient clustering technique via heterogeneous semantic links. In that work, a SimTree is structured by taking a publication database (PubDB). Links contain rich semantic information that may indicate important relationship between objects [2].

DBLP is the computer science database in XML file format. DBLP contains a more than enough information about research publications in different fields.

Each such database forms an immense size of information network, connecting in very complex ways research papers, authors, conferences/journals, and possibly citation Again it gives the date, year & the place of particular journals or conferences are published [3].

An algorithm TruthFinder have been implemented for the truth discovery with multiple conflicting information providers on the web, which concentrate on problem called *Veracity*, *i.e.*, *conformity to truth*, which studies how to find true facts from a large amount of conflicting information on many subjects that is provided by various web sites. In that work, TruthFinder successfully finds true facts among conflicting information, and identifies trustworthy web sites better than the popular search engines by taking an example of authors of books from various websites [4].

The ranking cube approach has been implemented by some authors which is use for efficient answering top-k queries with multidimensional selections [5].

A Rankclus algorithm has been implemented by some authors in which integrating clustering with ranking for heterogeneous information network analysis takes place. RankClus address the problem of generating clusters for a specified type of objects, as well as ranking information for all types of objects. In that work, a bi-typed heterogeneous network is used for clustering and ranking [6].

NetClus algorithm has been implemented which gives ranking-based clustering of Heterogeneous Information Networks with the help of Star Network Schema [7].

RankSQL is used for supporting ranking queries in relational database management system. RankSQL provides a seamless support of *top*-k queries as a first-class query type and integrating this type of queries in the existing SQL query engines [8].

Above we have discussed some reference papers which are directly or indirectly connected to our proposed algorithm RBC_A.

B. Problem Description for RBC_A data mining

We present an efficient mining algorithm for mining large information in this paper.To address this issue; we explore a new model of mining explained in section II (figure 1). We use two algorithms to address this work-

1. Naïve Bayes Classification algorithm used for carrying out different classifications
2. Priori algorithm used for ranking in which different steps are carried out like
   - Find candidate set.
   - Find frequent set.
   - Find frequent count or support count.
   - Apply ordering ascending or descending on the support values.

We take a DBLP dataset. Consider K is a set of mutually exclusive and exhaustive classifications $C_1,C_2,\ldots C_k$ having probabilities values $P(C_1)$, $P(C_2)$…….$P(C_k)$ and n attributes $a_1,a_2,\ldots..a_n$ which for given instance values $v_1,v_2\ldots.v_n$ respectively.

Algorithm 1: Naive Bayes Classification Algorithm

In this work, for getting profiling of particular object from the database, we use Naïve Bayes Classification technique, in which different classifications are carried out. Some examples of such classifications are getting titles published in particular year, getting titles by particular author, getting titles at particular venue, getting titles on particular date etc.

Definition 1: Prior Probability: The probability of an event, e.g. getting titles in particular year is the number of titles in that year divided by total number of titles in the database. This is called as prior probability.

Definition 2: Conditional Probability (Posterior Probability): The probability of an event occurring if we know that an attribute has a particular value (or that several variables have particular values) is called the conditional probability of the event occurring and is written as, P (class = titles in particular year | author = A_1). The vertical bar can be read as 'given that', so the whole term can be read as 'the probability that the class is getting title in particular year given that the author is 'A_1'. It is also called a posterior probability.

Algorithm 2: Priori Algorithm

Priori algorithm simply counts item occurrences to determine the frequent item sets. A subsequent pass, say z, consist of two phases. First frequent item sets L_{z-1} found in the $(z-1)^{th}$ pass are used to generate the candidate item sets $C_{cz,}$ using the candidate generation procedure. Next, the database is scanned and support of candidates in $C_{cz,}$ is counted. For the fast counting, we need to efficiently determine the candidates in C_{cz} contained in the given transaction t. The set of candidate item sets is subjected to a pruning process to ensure that all the subsets of the candidate sets are already known to be frequent item sets.

Definition 1: Association: An association is one of the data mining technique in which an association or interconnection between two or more types of objects from large database takes place.

Definition 2: Association Rule: An association rule is an expression of the form X→Y, where X & Y are the sets of items. The intuitive meaning of such a rule is that the transaction of the database which contains X tends to contain Y. For a given transaction database T, an association rule is an expression of the form X→Y, where X and Y are subsets of A and X→Y holds with confidence τ, if τ% of the transactions that support X also support Y. The rule X→Y has support σ in the transaction set T if σ% of translations in T supports XUY.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Definition 3: Support: Support means how often X and Y occurs together as a percentage of the total transactions.

Definition 4: Confidence: Confidence measures how much a particular item is dependent on another.

Definition 5: Frequent Set: Let T be the transaction database and σ be the user-specified minimum support. An item set X is a subset of A is said to be a frequent set in T with respect to σ, if $s(X)_T >= \sigma$.

Consider $C_c \epsilon C_k$ be set of items ,having probability $P(C_c)$ and s no. of items $b_1,b_2,,\ldots b_s$ for given transaction set T having values $t_1,t_2,\ldots t_r$ respectively.

From DBLP dataset, if we consider, a transaction T of finding number of authors for particular article key. We have C_c, which is set of authors A_1, A_2, A_3, set T is the set of transactions t_1,t_2,t_3,t_4,t_5 gives us set of authors present in that particular article set. σ=user specified minimum support= 20%. Since T contains 5 records, it means that an itemset that is supported by atleast (5*20/100) = 1 transaction is a frequent set.

Finally we get frequent count of each and every object present in the database. Apply ordering on this frequent count set, we get ranking in which authors published many papers having higher rank.

Clustering is takes place with the help of ranking. In ranking, we get the maximum number of authors published many papers. We store that maximum number, then after giving the cluster size, calculate first the cluster size is zero or not. If value is zero or other then adjust the size according to condition with the help of maximum count number. Display the list of authors by initializing start count and end count.

C. RBC_A for mining large information

The problems regarding databases are solved in new proposed algorithm RBC_A. Initillay database is partition into number of parts. By considering various classifications using classification technique, we find relevant information of each and every object or item present in the database. After finding information we find the candidate set, support and support count of frequent sets using priori algorithm. Apply ordering on frequent count set for ranking purpose. Finally prepare clusters with the help of ranking as per specified cluster size.
The main idea behind this algorithm is to discovery of hidden and useful patterns as well as needful information from the large amount of database.

Algorithm of RBC_A

Initially, a DBLP database D is segmented into number of parts say Q_1, Q_2,..Q_m. Apply Naive Bayes Classification Algorithm to classify the database by taking some classifications.

Algorithm – RBC_A
(Ranking Based Clustering Algorithm)

Take set of K classifications= $(C_1,C_2,\ldots C_k)$ having n attributes $a_1,a_2,\ldots a_n$ for given instance value $v_1,v_2,\ldots v_n$ respectively.

1) For all classes C_i
2) $P(C_i)=N(R_i)/n$
   // $P(C_i)$=Prior Probability of classification.
   // $N(R_i)$= No.of records in instance R_i
   // n= Total No.of records in given dataset.
3) For all classes C_i
4) For all attribute with their values of C_i
5) $P(C_i) * P(a_1=v_1$ and $a_2=v_2\ldots\ldots$and $a_n=v_n \mid C_i)$
   // $P(C_i)$= conditional probability of class C_i occurring for the specified instance
   // Apply Priori Algorithm to calculate frequent set and frequent count, to perform ranking
   Consider $C_c \epsilon C_k$ be set of items ,having probability $P(C_c)$ and s no. of items $b_1,b_2,,\ldots b_s$ for given transaction set T having values $t_1,t_2,\ldots t_r$ respectively.
   σ=user specified minimum support= 20%
6) Initialize z=1, C_c=all the 1-itemsets.
7) // Read the database to count the support of C_c to determine L_1
   L_1={ frequent 1-itemsets}
8) z=2 ; // z represent pass number.
9) While $(L_{z-1} \neq \varphi)$
10) do
11) Begin
   // C_{cz}=generation of candidate itemsets with given L_{z-1}
12) $C_{cz}= \varphi$
13) For all itemsets $b_1 \epsilon L_{z-1}$
14) do
15) For all itemsets $b_2 \epsilon L_{z-1}$
16) do
17) If $b_1[1]=b_2[1] \wedge b_1[2]=b_2[2] \wedge \ldots b_1[z-1]< b_2[z-1]$
18) Then $O=b_1[1],b_2[2],\ldots b_1[z-1],b_2[z-1]$
19) $C_{cz}=C_{cz} \cup \{O\}$
   // Prune (C_{cz})
20) For all $O \epsilon C_{cz}$
21) For all (z-1) subsets y of O
22) do
23) if $y \notin L_{z-1}$
24) then $C_{cz}=C_{cz}/\{O\}$
   // Display list of frequent sets.
25) For all transactions $t \epsilon T$
26) do
27) Increment the count of all candidates in Ccz that are contained in t.
28) L_z= All candidates in C_{cz} with minimum support.
29) Z=z+1
30) End
   // Answer= $U_z L_z$
   // Ranking
   Consider $C_c \epsilon C_k$ be set of items ,having probability $P(C_c)$ and s is the no. of items $b_1,b_2,,\ldots bs$ for given transaction set T having values $t_1,t_2,\ldots t_r$ respectively.
   // Read the database to count the support of C_c

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

```
31) Initialize j=1, passes=p, comparisons=comp, temp,
                    w=0, x=0,u=1
32) For all items bi ϵ Cc do
33) S[j] = Support (bi)
34) j=j+1
35) Set comp =s-1; p= s-1;
36) For all passes p from w
37) For all comparisons comp from x
38) do
39) if(S[x]<=S[x+1])
40) then temp=S[x];
41) S[x]=S[x+1];
42) S[x+1]=temp;
43) x=x+1
44) w=w+1
      // Display List
45) For all values u upto s
46) do
47) Return S[u]
48) u=u+1
49) end
      //Clustering
      // On the basis of ranking of authors, perform
        clustering.
      //Create a cluster of authors on the basis of number of
        titles published.
      // Initialize maximum number in the list, which is at
        the top of S[u]
50) Initialize  maxCount=S[1],f=1, startCount=0,
      endCount=0
51) clusters=N    // N represent Cluster number
      // Calculate whether the cluster size is zero or not.
52) If(maxCount%clusters=0)
53) clusterSize=maxCount/clusters
54) else
55) clusterSize=(maxCount/clusters)+1
      // Display Author List with specified clusterSize
56) For all clusters from f
57) do
58) startCount=(f-1)* clusterSize+1
59) endCount=(f) * clusterSize;
60) f=f+1
61) For all u
62) do
63) If(u>=startCount&&u<=endCount)
64) Return author List.
65) end
```

IV. RESULT ANALYSIS

In result analysis, we compare working of our proposed algorithm on different types of databases. We consider different databases which are given below-

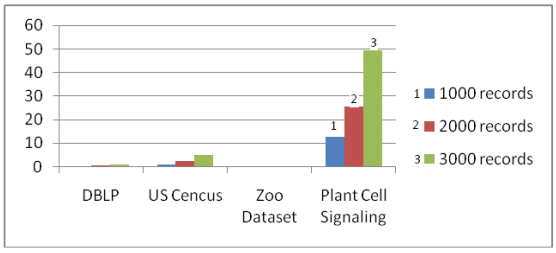
1. DBLP database which contains large information about research publications with their authors.
2. US Census data set which contains information with 68 categorical attributes.
3. Zoo data set having information with 17 attributes.
4. Plant Cell Signalling Data Set which describes the interactions of the nodes within the plant signaling network with 43 attributes.

Grpah 1- Different databases with different number of records with respect to time.

X-axis- Type of database
Y-axis- Time required in second.

[Table 1: Different databases with different number of records]

Number of Records	1000	5000	10000
DBLP	0.118	0.578	0.937
US Cencus	0.766	2.391	5.078
Zoo Dataset	0.125	0.312	0.485
Plant Cell Signalling	12.704	25.422	49.234

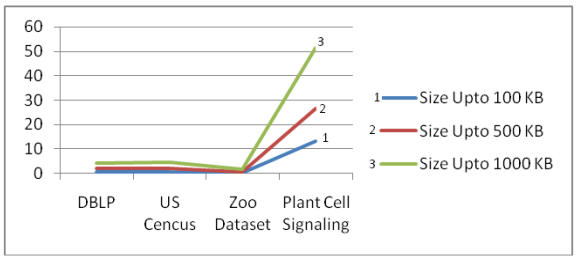


Grpah 1- Different databases with different number of records w.r.to time

Grpah 2- Different databases with different sizes with respect to time.
X-axis- Type of database
Y-axis- Time required in second.

Size of Database	Upto 100	Upto 500	Upto 1000
DBLP	0.578	2.12	4.23
US Cencus	0.677	2.24	4.5
Zoo Dataset	0.234	0.678	1.56
Plant Cell Signalling	13.123	26.4	51.34

Table 2: Different databases with different sizes



Grpah 2- Different databases with different sizes with respect to time

V. ABOUT THE DEMONSTRATION

In this demo, we will showcase the functionalities of database DBLP. The most updated DBLP data set will be used as the demonstration data set. We will demonstrate how profiling, ranking and clustering takes place.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

In the demonstration example, we take a small database of only 10 records but we can take thousands of records.We hope to interact with other re-searchers and practitioners to see how the interfaces and interactions should be improved with additional features.

VI. CONCLUSION

Finally after developing algorithm RBC_A, we conclude following points:-

1. It provides multidimensional profiling, workout profiles for authors, conferences, titles on the basis of classification technique.
2. It develops a ranking on the basis of association between objects by finding candidate set, support, support count and frequent sets with the help of priori algorithm. For example we get ranking on the basis of authors published more number of papers having higher rank.
3. It gives clustering-based on ranking and display list of authors in specified cluster size.
4. It also provides a user-friendly interface and a multi-resolution visualization tool for users to browse and comprehend the information derived from the above analyses.
5. Using DBLP data set, we can work with heterogeneous information network (Bi-type Information network (Information network).
6. Comparison of different databases takes place with the help of proposed algorithm.

REFERENCES

1. Yizhou Sun, Tianyi Wu, Zhijun Yin, Hong Chen Jiawei Han, Xiaoxin Yin, Peixiang Zhao, BibNetMiner: Mining Bibliographic Information Networks Canada.June9–12, 2008 in SIGMOD'08, Vancouver, BC, Canad
2. X.Yin, J.Han, and P.S.Yu.Linkclus: Efficient clustering via heterogeneous semantic links. In VLDB'06, Seoul, Korea, Sept. 2006
3. DBLP. http://www.informatik.uni-trier.de/ ley/db/.The dblp computer science bibliography
4. Xi. Yin, Ji. Han, and Ph. S. Yu. Truth discovery with multiple conflicting information providers on the web. In KDD'07, San Jose, CA, Aug. 2007
5. Xin, J. Han, H. Cheng, and X. Li. Answering top-k queries with multi-dimensional selection: The ranking cube approach. In VLDB'06 Seoul, Korea, Sept. 2006.
6. Y. Sun, J. Han, P. Zhao, Z. Yin, H. Cheng, and T. Wu. Rankclus: Integrating clustering with ranking for heterogeneous information network analysis. In EDBT'09, 2009.
7. Yizhou,Sun Yintao,Yu Jiawei Han: Ranking-Based Clustering of Heterogeneous Information Networks with Star Network Schema. KDD'09, June 28–July 1, 2009, Paris, France.
8. Chengkai Li, Mohamed A. Soliman, Kevin Chen-Chuan Chang, Ihab F. Ilyas: RankSQL: Supporting Ranking Queries in Relational Database Management Systems
9. Arun K. Pujari: Data Mining Techniques, Universities Press, Computer Science Book.
10. Max Bramer: Principles of Data Mining Book , Springer.
11. Jiawei Han : Data Mining Concepts and Techniques Book . University of Illinois at Urbana-Champaign , Micheline Kamber