ISSN (Online): 2230-7893
www.IJCEM.org

Implementation of an Efficient Matrix based Algorithm for Clustering in Web Usage Mining

Kanika Gupta[1], Kirti Aggarwal[2] and Neha Aggarwal[3]

[1,2,3] Department of computer science, Manav Rachna College of Engineering Faridabad,India
kanikagupta.1987@gmail.com, kirtibansal06@gmail.com, aggarwal.neha83@gmail.com

Abstract

Usage patterns discovered through Web usage mining are effective in capturing item-to-item and user-to-user relationships and similarities at the level of user sessions. However, without the benefit of deeper domain knowledge, such patterns provide little insight into the underlying reasons for which such items or users are grouped together. Furthermore, the inherent and increasing heterogeneity of the Web has required Web-based applications to more effectively integrate a variety of types of data across multiple channels and from different sources. Thus, a focus on techniques and architectures for more effective integration and mining of content, usage, and structure data from different sources is likely to lead to the next generation of more useful and more intelligent applications, and more sophisticated tools for Web usage mining that can derive intelligence from user transactions on the Web. This paper gives an insight into the proposed MABAC Algorithm and the implementation to show how it provides useful information within clusters.

Keywords: *similarity matrix, bond, innerbond, goodness function, web usage, rough set approximation technique.*

1. Introduction

MABAC, a new clustering algorithm give better results than some existing agglomerative hierarchical clustering algorithms discussed [1].MABAC work began with a survey of prominent clustering methods as detailed in a conference paper which provides part of chapter 2's text [8]. Several clustering methods from which we drew inspiration are outlined in the previous section of this chapter. This method involves exploiting features from other methods and merging them into a programmed unit. MABAC itself can be viewed as a hierarchical clustering method with a goodness function based on notions of bond and inner bond that in turn involve direct and indirect link measures [9,8]. The goal in this study is to achieve good clustering performance relative to other clustering methods.

1.1 PROPOSED ARCHITECTURE

MABAC is implemented in web mining for obtaining high quality clusters of user sessions.

It works as follows:-

A user puts up a query and correspondingly a session is maintained in the web log. Various sessions are taken up as input and using ROUGH SET APPROXIMATION TECHNIQUE clusters are maintained for the user sessions. These clusters serve as an input (dataset) to the proposed algorithm (MABAC).

MABAC works in three phases:-

PHASE I: -Construction of Similarity Matrix

It uses a function to convert the Euclidian distance to the similarity. The similarity value should be between 0 and 1 where 0 means no similarity at all and 1 means identical. Two parameters are used here: normalized coefficient alpha and cutoff value gamma.[5,6] Both gamma and alpha are values between 0 and 1. Similarity of a data point to itself is sim[i][i]=1. Similarity of a point to another point is sim[i][j]=alpha*(min/distance[i][j]) where min is the minimum distance. If distance between two points i and j are larger than gamma*max, then sim[i][j]=0, where max is the maximum distance.

PHASE II:-Applying operations (Bond and Inner bond)

1. The bond between two clusters depends not only on the direct link but also on indirect links.
   Bond(c1,c2)=Link(c1,c2)
   $$\div |c1| * |c2|$$
   where $link(c1,c2)=\sum link(p1,p2), p1 \in c1, p2 \in c2$
   i.e., the average link between two clusters
2. The merging criterion depends not only on the bond between two clusters but also on the inner bond of each cluster.

Inner Bond(c) = bond(c,c).

PHASE III: - Applying Goodness Function

A unified function that counts both within cluster information and between cluster information.

Goodness function is defined as the following:

Goodness=link(c1,c2)/ (inner Link(c1) * |c1| / (|c1| + |c2|) +inner Link(c2) * |c2| / (|c1| + |c2|)) .

This goodness function counts not only the similarity between two clusters but also the data property within each individual cluster.

2. IMPLEMENTATION RESULTS

A simulated example shows how the algorithm works:

ISSN (Online): 2230-7893
www.IJCEM.org

Table 1: user sessions

	A1	A2	A3
S1	2	1	3
S2	3	2	1
S3	2	1	3
S4	2	2	3
S5	1	1	4
S6	1	1	2
S7	3	2	1
S8	1	1	4
S9	2	1	3
S10	3	2	1

With reference to Table 1
Information System= (U, A)

U={s1,s2,s3,s4,s5,s6,s7,s8,s9,s10}
where U is the universe (a finite set of objects, U={S1,S2….., Sm}
A={a1,a2,a3}
where A is the set of attributes (features, variables)
where V(a) is the set of values a, called the domain of attribute a.
V1={1,2,3}
V2={1,2}
V3={1,2,3,4}
The notation U/A means that we are considering elementary sets of the universe U in space A.

Table 2: sets of universe in space A

U/A	A1	A2	A3
{s1,s3,s9}	2	1	3
{s2,s7,s10}	3	2	1
{s4}	2	2	3
{s5,s8}	1	1	4
{s6}	1	1	2

With reference to table 2
Target set={s1,s3,s4,s5,s9}
Lower approximation is given by the following set of objects:

LX= {s1, s3, s9} U {s4}
= {s1, s3, s9, s4}

Upper approximation is given by the following set of objects:

BX= {s1, s3, s9} U {s4} U {s5, s8}
= {s1, s3, s9, s4, s5, s8}

The boundary of S in U, defined as the difference between the upper and lower approximations, contains elements which are in the upper but not in the lower approximation:

BNX= {s1, s3, s9, s4, s5, s8} – {s1, s3, s9, s4}
= {s5, s8}

So the clusters formed are :

C1= {s1, s3, s4, s9}
C2= {s1, s3, s4, s5, s8, s9}
C3= {s1, s3, s4, s5, s9}

2.1 Applying MBAC Algorithm

I phase:- The first phase is to get a similarity matrix. Here our sample data in table 2 is converted into clusters and then the similarity is calculated by a function.

Sim(c1,c2)= No. of elements in c1/No. of elements in (c1+c2)=4\10 = 0.40
Sim(c1,c3)= No. of elements in c1/No. of elements in (c1+c3)=4\9 = 0.44
Sim(c2,c3)= No. of elements in c2/No. of elements in (c2+c3)=5\11=0.45

So the similarity matrix formed is:-

$$M = \begin{bmatrix} 1 & 0.4 & 0.44 \\ 0.4 & 1 & 0 \\ 0.4 & 0 & 1 \end{bmatrix}$$

The graph for corresponding similarity matrix is obtained as:-

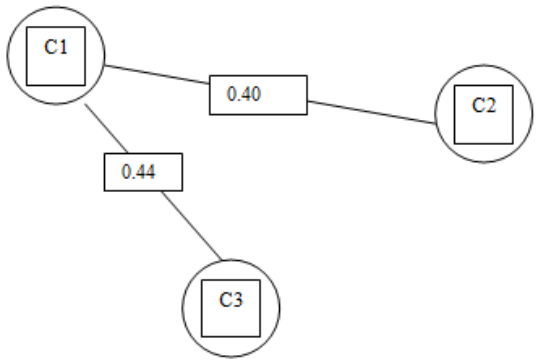

Fig1. Graph for similarity matrix

II Phase:-Applying Operations
In the second phase, the functions Bond and Inner bond are applied. This function counts both within cluster information and between cluster information.

1. The bond between two clusters depends not only on the direct link but also on indirect links.
2. The merging criterion depends not only on the bond between two clusters but also on the inner bond of each cluster.

Bond(c1,c2) =0.40+0.2*0.1
=0.40+0.02
=0.42
Bond(c1,c3)=0.44+0.3*0.15
= 0.44+0.045
= 0.485
Innerbond (c1) = Bond(c1,c1) =1\(c21* c1)
=1\(4*4)
=1\16=0.062
Innerbond (c2) = Bond(c2,c2) =1\(c2 * c2)
=1\(6*6)
=1\36=0.027
Innerbond (c3)=Bond(c3,c3) =1\(c3 * c3)
=1\(5*5)
=1\25=0.040

III phase:-Applying Goodness Function

In the third phase general hierarchical clustering algorithm is applied with a specified goodness function. The goodness function is very important for the clustering quality. A unique goodness function that measures both between-group and within-group property. It works very well for all test data sets here.

Goodness(c1,c2)=Bond(c1,c2)/ (Innerbond(c1) * |c1| / (|c1| + |c2|) +InnerBond(c2) * |c2| / (|c1| + |c2|))
= 0.42 / ((0.062*4/10)+(0.027*6/10))
= 0.42/(0.248+0.0162)
` =10.24

Goodness(c1,c3)=Bond(c1,c3)/ (Innerbond(c1) * |c1| / (|c1| + |c3|) +InnerBond(c3) * |c3| / (|c1| + |c3|))
= 0.0485 / ((0.062*4/9)+(0.040*5/9))
=0.485/(0.075+0.0222)
=9.7

Goodness(c1,c2)>Goodness(c1,c3)

So cluster 2 merges with cluster 1 instead of cluster 3
In this way the clustering is performed using the proposed MABAC Algorithm.

3. CONCLUSION

In this paper, we have presented a novel hierarchical algorithm MABAC that involves not only direct similarity but also the indirect similarity, not only between-group information but also within-group information. A byproduct of this algorithm is that it identifies some outliers by putting them in extremely small clusters. It may be a useful part in an outlier detection algorithm.

References

[1] kanika Gupta, kirti Aggarwal,Neha aggarwal: An Efficient Matrix based Algorithm for Clustering in Web Usage Mining, international journal of computer application,'unpublished".
[2] Jaideep Srivastava, Robert Cooley, Mukund Deshpande and PangNing Tan, "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data " Department of Computer Science and Engineering
[3] H.Hannah Inbarani, K.Thangavel,A. Pethalakshmi," Rough set based Feature Selection for Web Usage Mining", International Conference on Computational Intelligence and Multimedia Applications 2007
[4] R.Cooley, AND J.Srivastava 1999b, "WebSift: The web site information filter system. In Proceedings of the Web Usage Analysis and User Profiling" Workshop (WEBKDD'99), Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Boston, August).
[5] R.Cooley, B. Mobasher, and J.Srivastava 1999a, " Data preparation for mining world wide web browsing patterns", Knowl. Inf. Syst., 1, 1 (Feb)
[6] M. Eirinaki and M. Vazirgiannis, "Web mining for web personalization," ACM Trans. Inter. Tech., Vol. 3, No. 1, pp. 1-27, 2003.
[7] S. Jespersen, T. B. Pedersen, and J. Thorhauge, "Evaluating the markov assumption for web usage mining," in WIDM '03: Proceedings of the 5th ACM international workshop on Web information and data management. New York, NY, USA: ACM Press, 2003.
[8] Miha Grčar," User Profiling:Web Usage Mining", Department of Knowledge Technologies,Jozef Stefan Institute Jamova 39, 1000 Ljubljana, Slovenia.
[9] P. Giudici and C. Tarantola," Web mining pattern discovery", Department of Economics and Quantitative Methods,University of Pavia, Italy September 12, 2003..
[10]Faten Khalil, Jiuyong Li and Hua Wang," A Framework of Combining Markov Model with Association Rules for Predicting Web Page Accesses", Department of Mathematics & Computing University of Southern Queensland Toowoomba, Australia, 4350,..

Kanika Gupta received M.Tech (Information Technology) degrees with Hons. from Maharishi Dayanand University in 2011. Presently, she is working as an Assistant Professor in Computer Science and Engineering Department in Manav Rachna College of Engineering, Faridabad. Her areas of interest are web mining, clustering.
Kirti Aggarwal received M.Tech (Computer Science and Engineering) degrees with Hons. from Maharishi Dayanand University in 2009. Presently, she is working as an Assistant Professor in Computer Science and Engineering Department in Manav Rachna College of Engineering, Faridabad. Her areas of interest are Data Mining, Clustering
Neha Aggarwal is pursuing her M.Tech, final year(computer Science and Engineering) degree from Manav Rachna College of Engineering, Maharishi Dayanand University . Her areas of interest are data mining, clustering