A Clustering Model for Mining Evolving Web User Patterns in Data Stream Environment

Edmond H. Wu[†], Michael K. Ng[†], Andy M. Yip[‡], and Tony F. Chan[‡]

[†]Department of Mathematics, The University of Hong Kong
Pokfulam Road, Hong Kong.
[†]hcwu@hkusua.hku.hk,mng@maths.hku.hk
[‡]Department of Mathematics, University of California,
405 Hilgard Avenue, Los Angeles, CA 90095-1555, USA.
[‡]mhyip@math.ucla.edu,chan@math.ucla.edu

Abstract. With the fast growing of the Internet and its Web users all over the world, how to manage and discover useful patterns from tremendous and evolving Web information sources becomes new challenges to our data engineering researchers. Nowadays, many current and emerging Web applications require realtime monitoring and analyzing user patterns in the Web. However, most of the existing Web usage analysis and data mining techniques focused on finding patterns (e.g., association rules or clusters) from related static or historical databases, which greatly limits their wide adoption in online environments. Therefore, there is a great demand on designing more flexible data mining algorithms for various time-critical and data-intensive Web applications.

1 Introduction

With the fast growing of our capabilities in data acquisition and storage technologies, tremendous amount of datasets are generated and stored in databases, data warehouses, or other kinds of data repositories such as the World-Wide Web. A new challenge in Web mining is how to manage and discover potential and useful patterns from various types of Web data stored in different databases for particular tasks, such as system performance monitoring and user patterns discovery [1, 2, 4, 5]. Recently, Web applications such as personalization and recommendation have raised the concerns of people because they are crucial to improve customer services from business point of view, particularly for E-commerce Websites. Understanding customer preferences and requirements in time is a premise to optimize these Web services.

In this paper, we purpose a clustering model for generating and maintaining clusters which represent the changing Web user patterns in Websites. The clustering model can be fast updateed to reflect the current user patterns to the Web administrators. The rest of the paper is organized as follows: In section 2, we will introduce the the concept of dense regions discovery. Then, in section 3, we purpose the clustering model for evolving dense clusters discovery. After that, we give the experiment results on real Web data in section 4. Finally, we will address the Web applications and give some conclusion in section 5 and 6.

2 Algorithm for Mining Association Patterns

We first give our definition of dense region: Given a data matrix X, a submatrix $X(R,C)$ of X is called a maximal dense region with respect to v, if $X(R,C)$ is a constant matrix whose entries are v and any proper superset of $X(R,C)$ is a non-constant matrix is non-constant. In many practical applications(e.g., basket analysis from customer transaction databases), the data mining goal is to find association patterns from multidimensional data. For example, if we want to find the groups of customers who will buy the same products. The problem transfers into finding association patterns among all the customers and products. For instance, Yang et al [3] suggested an algorithm for finding error-tolerant frequent itemsets from high-dimensional data, such as binary matrices.

In this research, we use Dense Regions (DRs) to represent association patterns (e.g., subset of matrix with the same value). In practice, the matrices for dense regions discovery are large and sparse. Hence, efficient algortihms are needed for mining dense regions. In [5], we present the algorithm for mining dense regions in large data matrices. Due to the limited length of this paper, we just employ the algorithm in the experiments to find dense regions and omit the detailed introduction of it.

Example 1. Let X be a data 5×5 matrix given by the first matrix below. Using our algorithm for mining dense regions, we first filter out unqualified rows and cloumns (see the second matrix). Then, the dense regions of X with value 1 are returned by the algorithm shown in the four matrices in the brace.

$$\begin{pmatrix} 1&1&1&0&0 \\ 1&1&0&1&0 \\ 0&1&1&0&1 \\ 0&1&1&1&0 \\ 0&0&0&1&0 \end{pmatrix} \rightarrow \begin{pmatrix} 1&1&1&0 \\ 1&1&0&1 \\ 0&1&1&0 \\ 0&1&1&1 \end{pmatrix} \rightarrow \left\{ \begin{pmatrix} 1&1&\ast&\ast \\ 1&1&\ast&\ast \\ \ast&\ast&\ast&\ast \\ \ast&\ast&\ast&\ast \end{pmatrix}, \begin{pmatrix} \ast&1&1&\ast \\ \ast&\ast&\ast&\ast \\ \ast&1&1&\ast \\ \ast&1&1&\ast \end{pmatrix}, \begin{pmatrix} \ast&\ast&\ast&\ast \\ \ast&1&\ast&1 \\ \ast&\ast&\ast&\ast \\ \ast&1&\ast&1 \end{pmatrix}, \begin{pmatrix} \ast&1&\ast&\ast \\ \ast&1&\ast&\ast \\ \ast&1&\ast&\ast \\ \ast&1&\ast&\ast \end{pmatrix} \right\}.$$

3 The Clustering Model for Streaming Data

In this section, we first present a clustering method for mining clusters of dense regions (association patterns) from matrices (multidimensional data) and then purpose some strategies to maintain the evolving clusters in streaming data. Here, we use $|\mathcal{D}|$ to represent the total number of entries in a dense region $\mathcal{D}$.

3.1 Definition of Dense Clusters

Definition 1 (Dense Region Pairwise Overlap Rate). *Given two dense regions D_i and D_j, Dense Region Pairwise Overlap Rate (DPOR) of D_i is defined as the ratio:*

$$DPOR(D_i, D_j) = \frac{|D_i \bigcap D_j|}{|D_i|} \tag{1}$$

Definition 2 (Dense Region Union Overlap Rate). *Given a set of dense regions $\mathcal{D} = \{D_1, ..., D_n\}$, Dense Region Union Overlap Rate (DUOR) is defined as the ratio:*

$$DUOR(\mathcal{D}) = \frac{|\bigcap_{i=1}^{n} D_i|}{|\bigcup_{i=1}^{n} D_i|} \tag{2}$$

Here, we use $DPOR$ and $DUOR$ to measure the the extent of association (overlap) among different dense regions. Based on them, we give the definition of dense cluster as follows:

Definition 3 (Dense Clusters). *Given a set of dense regions $\mathcal{D} = \{D_1, ..., D_n\}$, a Dense Cluster $\mathcal{DC} = \bigcup_{i=1}^{k} D_i$ is defined as a subset of $\mathcal{D}$ with k DRs such that:*

- *For any $D_i \in \mathcal{DC}$, $DPOR(D_i, \mathcal{DC}) \geq MinDPOR$ and for any $D_j \notin \mathcal{DC}$ but $D_j \in \mathcal{D}$, $DPOR(D_j, \mathcal{DC}) < MinDPOR$, where $MinDPOR$ is the minimal threshold of $DPOR$.*
- *For $\mathcal{DC}$, $DUOR(\mathcal{D}_\mathcal{C}) \geq MinDUOR$, where $MinDUOR$ is the minimal threshold of $DUOR$.*

Example 2. In Example 1, dense regions D2 and D4 have common entries. In this case, $DPOR(D_2, D_4) = 3/6 = 50\%$, $DPOR(D_4, D_2) = 3/4 = 75\%$, $DUOR(D_2 \cup D_4) = 3/7 = 43\%$. If we set $MinDPOR = 50\%$ and $MinDUOR = 40\%$, then $D_2 \cup D_4$ is a dense cluster (DC) in matrix X.

3.2 Dense Cluster Generation and Maintenance

With the definition and data structure of dense cluster, we propose an algorithm for mining dense clusters from evolving data patterns. Because dense clusters denote a set of overlapping dense regions, in the inital stage, we use the algorithm in [5] to find dense regions in the given data matrices.

We propose a data model for mining dynamic dense regions and dense clusters in data stream environment. The main attributes of a dense region DR include: Dense Region ID, Timestamps(starting time T_s and ending time T_e of the DR), Dense Region Indexes(row and column indexes of D in matrix X). The main attributes of a dense cluster DC contain: Dense Cluster ID, Timestamps(T_s and T_e of the DC), Dense Cluster Indexes ($DPOR$, $DUOR$ and IDs of its DRs).

Using the indexing scheme for DRs and DCs above, we can employ greedy method to find all the dense clusters satisfying preset conditions. Besides MinDPOR and MinDUOR, we also set a threshold MinDC(The minimal size a dense cluster) to restrain the size of the dense clusters found by the algorithm. It means that for any DC, the total number of entries $|DC| = |\bigcup_{i=1}^{n} D_i| \geq MinDC$. The benefit of setting MinDC is that we can filter out trivial clusters which are not so userful to analyze data patterns. What's more, we can do some pruning on the dense regions to improve the effciency of the algorithm.

Lemma 1 (Pruning). *Given a set of dense regions $\mathcal{D} = \{D_1, ..., D_n\}$, for any two dense region D_i, $D_j \in \mathcal{D}$, if $DPOR(D_i, D_j) < MinDUOR$, then $(D_i \cup D_j)$ and any of its superset containing D_i and D_j cannot be a Dense Cluster.*

Proof. From the definition of $DPOR(D_i, D_j) = \frac{|D_i \bigcap D_j|}{|D_i|}$, we have:

$$DUOR(\mathcal{D}) = \frac{|\bigcap_{i=1}^{n} D_i|}{|\bigcup_{i=1}^{n} D_i|} \leq \frac{|D_i \bigcap D_j|}{|D_i \bigcup D_j|} \leq DPOR(D_i, D_j) < MinDUOP$$

Hence, D cannot be a dense cluster by the definition of DC.□

Lemma 2 (Pruning). *Given a set of dense regions $\mathcal{D} = \{D_1..., D_n\}$, for any two dense region D_i, $D_j \in \mathcal{D}$, if $|D_i| < MinDUOR \times |D_j|$, then $(D_i \cup D_j)$ and any of its superset containing D_i and D_j cannot be a Dense Cluster.*

Proof. From the definition of DUOR and Dense Cluster, we have:

$$DUOR(\mathcal{D}) \leq DUOR(D_i \cup D_j) = \frac{|D_i \bigcap D_j|}{|D_i \bigcup D_j|} \leq \frac{|D_i|}{|D_j|} < MinDUOR$$

Hence, D cannot be a dense cluster by the definition of DC.□

Lemma 3 (Pruning). *Given a set of dense regions $\mathcal{D} = \{D_1..., D_n\}$, for any dense region $D_i \in \mathcal{D}$, if $|D_i| < MinDC \times MinDUOR$, then any subset of $\mathcal{D}$ containing D_i cannot be a Dense Cluster.*

Proof. From the definition of DUOR and Dense Cluster, we have:

$$DUOR(\mathcal{D}) = \frac{|\bigcap_{i=1}^{n} D_i|}{|\bigcup_{i=1}^{n} D_i|} \leq \frac{|D_i|}{MinDC} < MinDUOR$$

Hence, D cannot be a dense cluster by the definition of DC.□

Therefore, given a set of candidate dense regions $\mathcal{D} = \{D_1, D_2..., D_m\}$, we can use Lemma 1, 2 and 3 to eliminate unqualified dense regions and finally find the qualifying dense clusters. The benefit of adopting pruning process is that it can greatly improve the efficiency of the algorithm so that the clustering model can be applied in online clustering of evolving association patterns (dense regions). We summarize the dynamic clustering algorithm as follows:

```
Begin
1. Use DRIFT algorithm (refer to [5]) to mine dense regions from streaming data
2. Set the clustering model thresholds (e.g.,MinDPOR, MinDUOR, MinDC)
3. Prune out unqualified dense regions (DRs)
4. For each qualifying DR, search the set of DRs to form a DC in a greedy manner
5. Indexing and storing all the dense clusters (DCs) found
6. If a new DR generates, test whether this DR can be merged in any existing DC
7. If a old DR eliminates, test whether any DC needs to be eliminated or updated
8. Maintain and Update the Dense Clusters by the changing Dense Regions
9. Output the clustering results at certain time point if any query arrives
End
```

4 Experiments

We used the Web usage data from ESPNSTAR.com.cn, a sports Website to test and evaluate the performance and effectiveness of our clustering model proposed. Table 1 shows the statistics of the Website pages accessed by Web users during two months in 2003. The three columns on the right denote the dense regions found from these datasets by using our dense regions discovery algorithm. (In the data matrices, rows represent visitor (Web users), columns denote Web pages, here, we set the minial size of the dense regions to find is 10×10.)

Dataset	No.Accesses	No.Sessions	No.Visitors	No.Pages	No. DRs	Average Size	Maximal Size
ES1	583,386	54,300	2,000	790	104	13 × 15	47 × 32
ES2	2,534,282	198,230	42,473	1,320	350	15 × 14	29 × 46
ES3	6,260,840	517,360	50,374	1,450	978	16 × 21	23 × 42
ES4	78,236	5,000	120	236	56	12 × 14	34 × 25
ES5	7,691,105	669,110	51,158	1,609	1,231	17 × 13	39 × 51

Table 1. Real Web Datasets and the corresponding Dense Regions found

We proposed several clustering experiments below to evaluate the performance of the clustering algorithm using different model thresholds.

Example 1: We use all the DRs from these datasets for dense cluster discovery. The result in Fig 1 showed that when increasing MinDPOR, the running time will decrease. It can be explained that the many unqualfied dense regions are eliminated during the pruning process. (Here, MinDUOR=0.5, MinDC=200)

Example 2: Similarly, the result in Fig 2 showed that it revealed a linear relationship when varying MinDUOR. (Here, MinDPOR=0.6, MinDC=200)

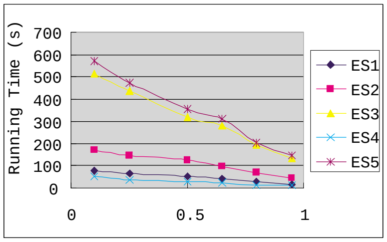

Fig. 1. Varying MinDPOR

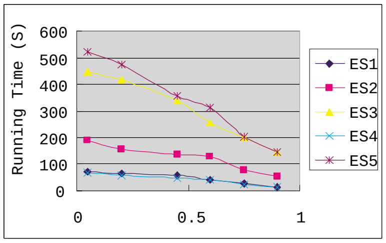

Fig. 2. Varying MinDUOR

Example 3: We also test the sensitiveness of the clustering algorithm by varying preset minimal size of dense cluster. The result in Fig 3 showed that the setting of MinDC will effect the total clustering time. In practice, larger size of dense clusters is more interesting. (Here, MinDPOR=0.6 and MinDUOR=0.5)

Example 4: We further test the scalability of the clustering algorithm by increasing the changing DRs for dense cluster discovery. The result in Fig 4 showed that when increasing DRs, the running time also increases linearly. It showed that it is feasible to apply this algorithm for online clustering of DRs (Here, we use ES3 and ES5 for testing, MinDPOR=0.6, MinDUOR=0.5, MinMinDC=200).

Above experiments results also evaluate the effectiveness of pruning process and the feasibility of the clustering model for data mining.

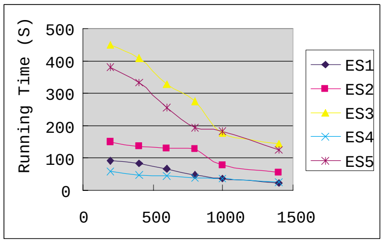

Fig. 3. Varying MinDC

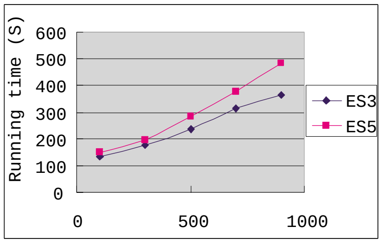

Fig. 4. Varying DRs

5 Web Applications

In this section, we will address how to apply the clustering model in practical Web mining applications. In our previous work [1, 2, 5], we proposed some novel methods to discover potential users patterns from multidimensional Web datasets for effective Web mining, such as associatin rules [2], dense regions [5].

Web administrators can use the clustering model for evolving analysis of Web usage. For example, in the experiments, we use the sports Website's datasets which contain the Web accesses information during some periods. The clustering model can help us identify groups of users with common interest which are in the same cluster, or separate different Web users to promote different Web services(e.g., invite football fans to subscribe new football member service).

What's more, we can reorganize the Web pages and content so that it can meet the need of more customers. For instance, if some Web user cluster or Web page cluster become larger, it means these Web user are more interested in the Website or the Web pages are getting more popular among visitors at particular time. Acquiring such information, the Website dministrators can timely response to such pattern changes and then optimize their Web services provided. In practice, the clustering model can be used to online monitor Web usage, Web personalization, recommendation, and system perforamnce analysis etc.

6 Conclusions

In this paper, we propose a new clustering model for dense clusters discovery. The experiments showed that it is effective and efficient, no matter for offline or online clustering applications. It can be employed in different Web mining applications, such as Web user patterns discovery and evolving anlaysis. In the future, we will extend the clustering model for other data mining applications.

References

1. E. H. Wu, M. K. Ng, and J. Z. Huang, *On improving website connectivity by using web-log data streams*, Proc. of the 9th International Conference on Database Systems for Advanced Applications (DASFAA 2004), Jeju, Korea, 2004.
2. E. H. Wu, M. K. Ng, *A graph-based optimization algorithm for Website topology using interesting association rules*, Proc. of the Seventh Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2003), Seoul, Korea, 2003.
3. C. Yang, U. Fayyad, and P. S. Bradley, *Efficient discovery of error-tolerant frequent itemsets in high dimensions*, Proceedings of the Seventh ACM SIGKDD Conference, San Francisco, California, pp. 194–203, 2001.
4. Q. Yang, J. Z. Huang and M. K. Ng,*A data cube model for prediction-based Web prefetching*, Journal of Intelligent Information Systems, 20:11-30, 2003.
5. Andy M. Yip, Edmond H.Wu, Michael K.Ng, Tony F. Chan, *An efficient algorithm for dense regions discovery from large-scale data stream*, Proc. of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD2004), 2004.