Using Category-Based Adherence to Cluster Market-Basket Data

Ching-Huang Yun, Kun-Ta Chuang+ and Ming-Syan Chen
Department of Electrical Engineering
Graduate Institute of Communication Engineering+
National Taiwan University
Taipei, Taiwan, ROC
E-mail: chyun@arbor.ee.ntu.edu.tw, doug@arbor.ee.ntu.edu.tw, mschen@cc.ee.ntu.edu.tw

Abstract

In this paper, we devise an efficient algorithm for clustering market-basket data. Different from those of the traditional data, the features of market-basket data are known to be of high dimensionality, sparsity, and with massive outliers. Without explicitly considering the presence of the taxonomy, most prior efforts on clustering market-basket data can be viewed as dealing with items in the leaf level of the taxonomy tree. Clustering transactions across different levels of the taxonomy is of great importance for marketing strategies as well as for the result representation of the clustering techniques for market-basket data. In view of the features of market-basket data, we devise in this paper a novel measurement, called the category-based adherence, *and utilize this measurement to perform the clustering. The distance of an item to a given cluster is defined as the number of links between this item and its nearest large node in the taxonomy tree where a large node is an item (i.e., leaf) or a category (i.e., internal) node whose occurrence count exceeds a given threshold. The category-based adherence of a transaction to a cluster is then defined as the average distance of the items in this transaction to that cluster. With this category-based adherence measurement, we develop an efficient clustering algorithm, called algorithm* CBA *(standing for Category-Based Adherence), for market-basket data with the objective to minimize the category-based adherence. A validation model based on* Information Gain *(IG) is also devised to assess the quality of clustering for market-basket data. As validated by both real and synthetic datasets, it is shown by our experimental results, with the taxonomy information, algorithm CBA devised in this paper significantly outperforms the prior works in both the execution efficiency and the clustering quality for market-basket data.*

1 Introduction

Data clustering is an important technique for exploratory data analysis [16]. Explicitly, data clustering is a well-known capability studied in information retrieval [6], data mining [7], machine learning [9], and statistical pattern recognition [15]. In essence, clustering is meant to divide a set of data items into some proper groups in such a way that items in the same group are as similar to one another as possible. Most clustering techniques utilize a pairwise similarity for measuring the distance of two data points. Recently, there has been a growing emphasis on clustering very large datasets to discover useful patterns and/or correlations among attributes [3][4][10][23]. Note that clustering is an application dependent issue and certain applications may call for their own specific requirements.

Market-basket data (also called transaction data) has been well studied in mining association rules for discovering the set of frequently purchased items [5][13][18]. Different from the traditional data, the features of market-basket data are known to be of high dimensionality, sparsity, and with massive outliers. ROCK is an agglomerative hierarchical clustering algorithm by treating market-basket data as categorical data and using the links between the data points to cluster categorical data [11]. The authors in [17] proposed an EM-based algorithm by using the maximum likelihood estimation method for clustering transaction data. OPOSSUM is a graph-partitioning approach based on a similarity matrix to cluster transaction data [19]. The work in [20] proposed a K-Mean-based algorithm by using large items as the similarity measurement to divide the transactions into clusters such that transactions with similar large items are grouped into the same clusters. OAK in [21] combined hierarchical and partitional clustering techniques. STC in [22] utilized a fixed small to large item ratio to perform the clustering of market-basket data. In market-basket data, the taxonomy of items defines the generalization relationships for the concepts in different ab-

0-7695-1754-4/02 $17.00 © 2002 IEEE

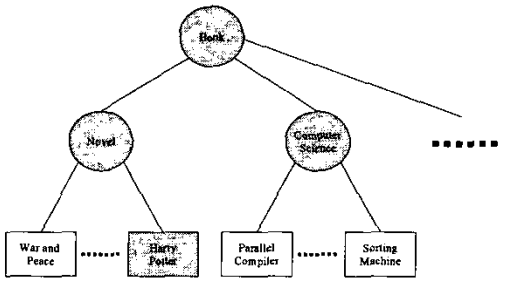


Figure 1. An example taxonomy tree for books.

straction levels. Item taxonomy (i.e., *is-a* hierarchy) is well addressed with respect to its impact to mining association rules of market-basket database [13][18] and can be represented as a tree, called *taxonomy tree.*

In view of the features of market-basket data, we devise in this paper a novel measurement, called the *category-based adherence*, and utilize this measurement to perform the clustering. The *distance* of an item to a given cluster is defined as the number of links between this item and its nearest large node in the taxonomy tree. In the taxonomy tree, the leaf nodes are called the item nodes and the internal nodes are called the category nodes. For the example shown in Figure 1, "War and Peace" is an item node and "Novel" is a category node. As formally defined in Section 2, a *large item (category)* is basically an item (category) with its occurrence count in transactions exceeding a given threshold. If an item (or category) is large, its corresponding node in the taxonomy tree is called a *large node*. For the example shown in Figure 1, nodes marked gray are assumed to large nodes. The *category-based adherence* of a transaction to a cluster is then defined as the average distance of the items in this transaction to that cluster. With this category-based adherence measurement, we develop an efficient clustering algorithm, called algorithm *CBA* (standing for *Category-Based Adherence*), for market-basket data. Explicitly, CBA employs the category-based adherence as the similarity measurement between transactions and clusters, and allocates each transaction to the cluster with the minimum category-based adherence. To the best of our knowledge, without explicitly considering the presence of the taxonomy, previous efforts on clustering market-basket data. [11][17][19][20][21] unavoidably restricted themselves to deal with the items in the leaf level (also called item level) of the taxonomy tree. However, clustering transactions across different levels of the taxonomy is of great importance for marketing strategies as well as for the result representation of the clustering techniques for market-basket data. Note that in the real market-basket data, there are volume of transactions containing only single items, and many items are purchased infrequently. Hence, without considering the taxonomy tree, one may inappropriately treat a transaction (such as the one containing "parallel compiler" in Figure 1) as an outlier. However, as indicated in Figure 1, purchasing "parallel compiler" is in fact instrumental for the category node "computer science" to become a large node. In contrast, by employing category-based adherence measurement for clustering, many transactions will not be mistakenly treated as outliers if we take categorical relationships of items in the taxonomy tree into consideration, thus leading to better marketing strategies. The details of CBA will be described in Section 3. A validation model based on *Information Gain* (IG) is also devised in this paper for clustering market-basket data. As validated by real and synthetic datasets, it is shown by our experimental results, with the taxonomy information, algorithm CBA devised in this paper significantly outperforms the prior works [14][20] in both the execution efficiency and the clustering quality for market-basket data.

This paper is organized as follows. Preliminaries are given in Section 2. In Section 3, algorithm CBA is devised for clustering market-basket data. Experimental studies are conducted in Section 4. This paper concludes with Section 5.

2 Preliminary

The problem description will be presented in Section 2.1. In Section 2.2, we describe a new validation model, IG validation model, for the assessment to the quality of different clustering algorithms.

2.1 Problem Description

In this paper, the market-basket data is represented by a set of transactions. A database of transactions is denoted by $D = \{t_1, t_2, ..., t_h\}$, where each transaction t_j is represented by a set of items $\{i_1, i_2, ..., i_h\}$. A example database for clustering market-basket data is described in Table 1 where there are twelve transactions, each of which has a transaction identification (abbreviated as TID) and a set of purchased items. For example, transaction ID 40 has items h and item z. A clustering $U =< C_1, C_2, ..., C_k >$ is a partition of transactions into k clusters, where C_j is a cluster consisting of a set of transactions.

Items in the transactions can be generalized to multiple concept level of the taxonomy. An example taxonomy tree is shown in Figure 2. In the taxonomy tree, the leaf nodes are called the *item nodes* and the internal nodes are called the *category nodes*. The root node in the highest level is a virtual concept of the generalization of all categories. In

TID	10	20	30	40	50	60
Items	g, x	m, y	y, z	h, z	g, x, y	g, n
TID	70	80	90	100	110	120
Items	k, m, n	y	g, k, n	m, n	y, z	g, h, n

Table 1. An example database D.

this taxonomy structure, item g *is-a* category B, category B *is-a* category A, and item h *is-a* category B, etc. In this paper, we use the measurement of the occurrence count to determine which items or categories are major features of each cluster.

Definition 1: The count of an item i_k in a cluster C_j, denoted by $Count(i_k, C_j)$, is defined as the number of transactions in cluster C_j that contain this item i_k. An item i_k in a cluster C_j is called a *large item* if $Count(i_k, C_j)$ exceeds a predetermined threshold.

Definition 2: The count of a category c_k in a cluster C_j, denoted by $Count(c_k, C_j)$, is defined as the number of transactions containing items under this category c_k in cluster C_j. A category c_k in a cluster C_j is called a *large category* if $Count(c_k, C_j)$ exceeds a predetermined threshold.

Note that one transaction may include more than one item from the same category, in which case the count contributed by this transaction to that category is still one. In this paper, the minimum support percentage S_p is a given parameter for determining the large nodes of the taxonomy tree in the cluster. For a cluster C_j, the minimum support count $S_c(C_j)$ is defined as follows.

Definition 3: For cluster C_j, the minimum support count $S_c(C_j)$ is defined as:

$$S_c(C_j) = S_p * |C_j|.$$

where $|C_j|$ denotes the number of transactions in cluster C_j.

Consider the example database in Table 1 as an initial cluster C_0 with the corresponding taxonomy tree recording the counts of the items/categories shown in Figure 2. Then, $Count(g, C_0) = 5$ and $Count(E, C_0) = 7$. With $S_p = 50\%$, we have $S_c(C_0) = 6$. In this example, all categories are large but all items are not.

2.2 Information Gain Validation Model

To evaluate the quality of clustering results, some experimental models were proposed [8][12]. In general, *square error criterion* is widely employed in evaluating the efficiency of numerical data clustering algorithms [8]. Note that the nature feature of numeric data is quantitative (e.g., weight or length), whereas that of categorical data is qualitative (e.g., color or gender) [16]. Thus, validation schemes using the concept of variance are thus not applicable to assessing the clustering result of categorical data. To remedy this problem, some real data with good classified labels, e.g., mushroom data, congressional votes data, soybean disease [2] and Reuters news collection [1], were taken as the experimental data for categorical clustering algorithms [11][20][21]. In view of the feature of market-basket data, we propose in this paper a validation model based on Information Gain (IG) to assess the qualities of the clustering results.

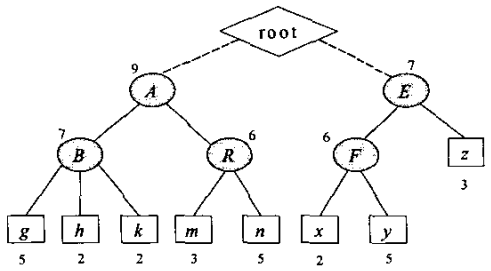


Figure 2. An illustrative taxonomy example whose transactions are shown in Table 2 ($S_p = 0.5$).

The definitions required for deriving the information gain of a clustering result are given below.

Definition 4: The entropy of an attribute J_a in the database D is defined as:

$$I(J_a, D) = -\sum_{i=1}^{n} \frac{|J_a^i|}{|D|} * log_2 \frac{|J_a^i|}{|D|}.$$

where $|D|$ is the number of transactions in the database D and $|J_a^i|$ denotes the number of the transactions whose attribute J_a is classified as the value J_a^i in the database D.

Definition 5: The entropy of an attribute J_a in a cluster C_j is defined as:

$$I(J_a, C_j) = -\sum_{i=1}^{n} \frac{|J_{a,c_j}^i|}{|C_j|} * log_2 \frac{|J_{a,c_j}^i|}{|C_j|}.$$

where $|C_j|$ is the number of transactions in cluster C_j, and $|J_{a,c_j}^i|$ denotes the number of the transactions whose

attribute J_a is classified as the value J_a^i in C_j.

Definition 6: Let a clustering U contain C_1, C_2, ..., C_m clusters. Thus, the entropy of an attribute J_a in the clustering U is defined as:

$$E(J_a, U) = \sum_{C_j \in U} \frac{|C_j|}{|D|} I(J_a, C_j).$$

Definition 7: The information gain obtained by separating J_a into the clusters of the clustering U is defined as:

$$Gain(J_a, U) = I(J_a, D) - E(J_a, U).$$

Definition 8: The information gain of the clustering U is defined as:

$$IG(U) = \sum_{J_a \in I} Gain(J_a, U).$$

where I is the data set of the total items purchased in the whole market-basket data records.

A complete numerical example on the use of these definitions will be given in Section 3.3. For clustering market-basket data, the larger an IG value, the better the clustering quality is. In market-basket data, with the taxonomy tree structure, there are three kinds of IG values, i.e., $IG_{item}(U)$, $IG_{cat}(U)$, and $IG_{total}(U)$, for representing the quality of a clustering result. Specifically, $IG_{item}(U)$ is the information gain obtained on items and $IG_{cat}(U)$ is the information gain obtained on categories. $IG_{total}(U)$ is the total information gain, i.e., $IG_{total}(U) = IG_{item}(U) + IG_{cat}(U)$. In general, market-basket data set is typically represented by a 2-dimensional table, in which each entry is either 1 or 0 to denote purchased or non-purchased items, respectively. In IG validation model, we treat each item in market-basket data as an attribute J_a with two classified label, 1 or 0.

3 Design of Algorithm CBA

The similarity measurement of CBA, called category-based adherence, will be described in Section 3.1. The procedure of CBA is devised in Section 3.2 and an illustrative example is given in Section 3.3.

3.1 Similarity Measurement: Category-Based Adherence

The similarity measurement employed by algorithm CBA, called category-based adherence, is defined as follows. In the taxonomy tree, the *nearest large node* of an item i_k is itself if i_k is large and is its nearest large ancestor node otherwise. Then, the distance of an item to a cluster is defined below.

Definition 9: (Distance of an item to a cluster): For an item i_k of a transaction, the *distance* of i_k to a given cluster C_j, denoted by $d(i_k, C_j)$, is defined as the number of links between i_k and the nearest large nodes of i_k. If i_k is a large node in cluster C_j, then $d(i_k, C_j) = 0$. Otherwise, the nearest large node is the category node which is the lowest generalized concept level node among all large ancestors of item i_k. Note that if an item or category node is identified as large node, all its high level category nodes will also be large nodes.

Definition 10: (Adherence of a transaction to a cluster):
For a transaction $t = \{i_1, i_2, ..., i_p\}$, the adherence of t to a given cluster C_j, denoted by $H(t, C_j)$, is defined as the average distance of the items in t to C_j and shown below.

$$H(t, C_j) = \frac{1}{p} \sum_{k=1}^{p} d(i_k, C_j).$$

where $d(i_k, C_j)$ is the distance of i_k in cluster C_j.

3.2 Procedure of Algorithm CBA

The overall procedure of algorithm CBA is outlined as follows.

Procedure of Algorithm CBA

Step 1. Randomly select k transactions as the seed transactions of the k clusters from the database D.

Step 2. Read each transaction sequentially and allocates it to the cluster with the minimum category-based adherence. For each moved transaction, the counts of items and their ancestors are increased by one.

Step 3. Repeat Step 2 until no transaction is moved between clusters.

Step 4. Output the taxonomy tree for each cluster as the visual representation of the clustering result.

In Step 1, algorithm CBA randomly selects k transactions as the seed transactions of the k clusters from the database D. For each cluster, the items of the seed transaction are counted once in the taxonomy tree. In each cluster, the items and their ancestors are all large in the very beginning because their count is one (which means 100% in the only seed transaction), larger than the minimum support threshold. For each cluster, these large nodes represent the hot sale topics in this cluster. In Step 2, algorithm CBA reads each transaction sequentially and allocates it to the cluster with the minimum category-based adherence. After one transaction is inserted into a cluster C_j, the counts of the items and

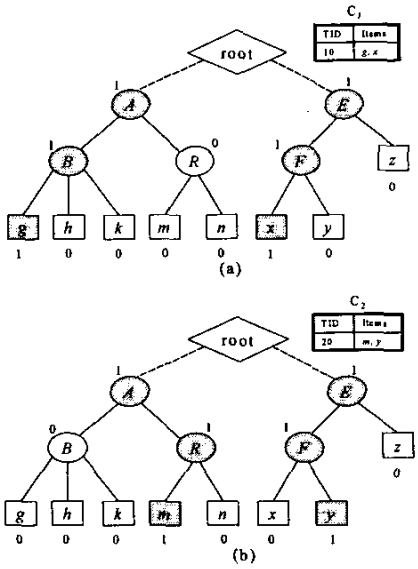


Figure 3. In Step 1, algorithm CBA randomly chooses the seed transaction for each cluster.

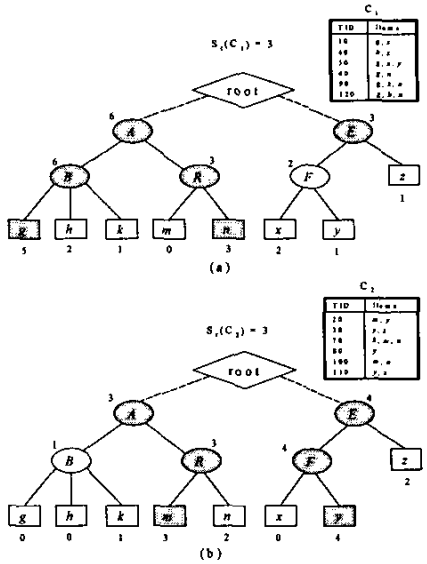


Figure 4. In Step 2, algorithm CBA reads each transaction sequentially and allocates it to the cluster with the minimum category-based adherence.

their ancestors are increased by one in the corresponding nodes in the taxonomy tree of C_j. In addition, the minimum support count of C_j is updated. In Step 3, algorithm CBA repeats Step 2 until no transaction is moved between clusters. In Step 4, algorithm CBA outputs the taxonomy tree of the final clustering result for each cluster, where the items, categories, and their corresponding counts are presented.

3.3 An Illustrative Example

For the example database D shown in Table 1, we set $k = 2$ and $S_p = 50\%$. In Step 1, algorithm CBA randomly chooses TID 10 and TID 20 as the seed transaction of the cluster C_1 and C_2, respectively. Then, for cluster C_1 shown in Figure 3(a), nodes marked gray are the purchased items of TID 10 and the corresponding categories in the taxonomy tree, and they are identified as large nodes. Similarly, for cluster C_1, shown in Figure 3(b), nodes marked gray are large. In Figure 3, the count of each node is illustrated nearby. For example, $Count(g, C_1)$ is 1 and $Count(g, C_2)$ is 0. In Step 2, algorithm CBA first allocates TID 30 to cluster C_2 because $H(30, C_2) = \frac{1}{2}(1 + 0) = \frac{1}{2}$ (i.e., the link number of item y to category F plus the link number of item z to category E) is smaller than $H(30, C_1) = \frac{1}{2}(1+1) = 1$. Similarly, TIDs 40, 50, 60, 90, and 120 are allocated to cluster C_1 which is shown in Figure 4(a). TIDs 30, 70, 80, 100, and 110 are allocated to cluster C_2 which is shown in Figure 4(b). Then, algorithm CBA derives $S_c(C_1) = 3$ and $S_c(C_2) = 3$ by $S_p * |C_1| = 0.5 * 6 = 3$ and $S_p * |C_2| = 0.5*6 = 3$, respectively. Because $Count(g, C_1) > S_c(C_1)$, category A is identified as a large node in cluster C_1 and marked gray. In Step 3, algorithm CBA proceeds to iteration 2. In iteration 2, two transactions, TID 50 and TID 70 are moved. TID 50 is moved from cluster C_1 to cluster C_2 because $H(50, C_1) = \frac{1}{3}(0 + 2 + 2) = \frac{4}{3} > H(50, C_2) = \frac{1}{3}(2 + 1 + 0) = 1$, and TID 70 is moved from cluster C_2 to cluster C_1 due the $H(70, C_1) = \frac{1}{3}(1 + 1 + 0) = \frac{2}{3} < H(70, C_2) = \frac{1}{3}(2+0+1) = 1$. Then, algorithm CBA identifies the large nodes again. In iteration 3, only one transaction TID 100 is moved from cluster C_2 to cluster C_1. In iteration 4, there is no movement and thus algorithm CBA proceeds to Step 4. The final feature trees for the clusters are shown in Figure 5.

Note that a transaction at item level may not be similar to any cluster. For example, TID 10 $\{g, x\}$ and TID 40 $\{h, z\}$ have no common items, but item g and item h have common category B and item x and item z have common category E. Thus, TID 10 is similar to TID 40 in the high level concept. By taking category-based adherence mea-

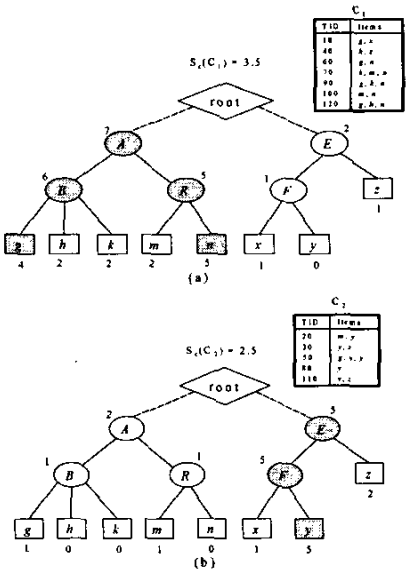


Figure 5. In Step 4, algorithm CBA generates the clustering U_1.

surement, many transactions may not be taken as outliers if we take categorical relationships of items into consideration. In addition, transactions at the item level may have the same similarities in different clusters. However, by summarizing the similarities of every item across their category levels, algorithm CBA allocates each transaction to a proper cluster. For example, TID 50 has three items: g, x, and y. Item g is large in cluster C_1 and item y is large in cluster C_2. Thus, TID 50 has the same similarities in both C_1 and C_2. However, item x is a category F which is a large node in C_2. Thus, TID 50 is allocated to C_2.

To provide more insight into the quality of CBA, we calculate the IG values of the clustering U_1 shown in Figure 5. Note that for an item i_k, $I_{i_k}^{Yes}, I_{i_k}^{No}$ are the two classified labels of item i_k for representing purchased and non-purchased values. For item g, the information gain $Gain(g, U_1) = I(g, D) - E(g, U_1) = (-\frac{5}{12}log_2\frac{5}{12} - \frac{7}{12}log_2\frac{7}{12}) - [\frac{7}{12}(-\frac{4}{7}log_2\frac{4}{7} - \frac{3}{7}log_2\frac{3}{7}) + \frac{5}{12}(-\frac{1}{5}log_2\frac{1}{5} - \frac{4}{5}log_2\frac{4}{5})] = 0.10$. Similarly, IG values of other items are $Gain(h, U_1) = 0.15$, $Gain(k, U_1) = 0.48$, $Gain(m, U_1) = 0.31$, $Gain(n, U_1) = 0.48$, $Gain(x, U_1) = 0$, $Gain(y, U_1) = 0.98$, and $Gain(z, U_1) = 0.39$. Hence, $IG_{item}(U_1) = \sum_{J_a \subset I} Gain(J_a, U_1) = 2.89$, where I is the set of items $\{g,\ h,\ k,\ m,\ n,\ x,\ y,\ z\}$. Similarly, $Gain(B, U_1) = 0.33$, $Gain(R, U_1) = 0.2$, $Gain(A, U_1) = 0.41$, $Gain(F, U_1) = 0.65$, $Gain(E, U_1) = 0.48$, and $IG_{cat}(U_1) = \sum_{J_a \subset C} Gain(J_a, U_1) = 2.07$, where C is the set of categories $\{A,\ B,\ E,\ F,\ R\}$. Then, $IG_{total}(U_1) = IG_{item}(U_1) + IG_{cat}(U_1) = 4.96$.

4 Experimental Results

To assess the efficiency of CBA, we conducted experiments to compare CBA with a traditional hierarchical clustering algorithm, called *CL* (standing for *Complete Link*) [14] and another algorithm proposed in [20] (for the convenience, the algorithm is named as *Basic* in this paper). By extending both previous approaches with taxonomy consideration in market-basket data, we also implement algorithm *CLT* (standing for *Completed Link with Taxonomy*) and algorithm *BasicT* (standing for *Basic with Taxonomy*) for comparison purposes. The details of data generation are described in Section 4.1. The experimental results are shown in Section 4.2

4.1 Data Generation

We take the real market-basket data from a large bookstore company for performance study. In this real data set, there are $|D| = 100K$ transactions and $N^I = 21807$ items. Note that in this real data, there are volume of transactions containing only single items, and many items are purchased infrequently. In addition, the number of the taxonomy level in this real data set is 3. In addition, to provide more insight into this study, we use a well-known market-basket synthetic data generated by the IBM Quest Synthetic Data Generation Code [5], as the synthetic data for performance evaluation. This code will generate volumes of transaction data over a large range of data characteristics. These transactions mimic the transactions in the real world retailing environment. This generation code also assumes that people will tend to buy sets of items together, and each such set is potentially a maximal large itemset. The average size of the transactions is denoted by $|T|$. The average size of the maximal potentially large itemsets is denoted by $|I|$. The number of maximal potential large itemsets is denoted by $|L|$. The number of items in database is denoted by N^I. The number of roots is denoted by N^R and the number of the taxonomy level is denoted by N^L.

4.2 Performance Study

We conduct two experiments in this section for performance study and the clustering quality is evaluated by the IG values. For algorithms CBA, Basic, and BasicT, the minimum support percentage S_p is set to 0.5%. Recall that

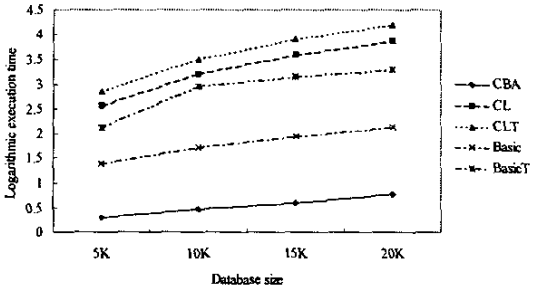


Figure 6. Execution time in logarithm for CBA, CL, CLT, Basic, and BasicT when the database size $|D|$ varies.

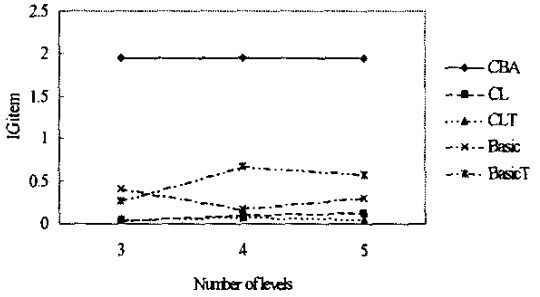
there are three kinds of IG values, i.e., IG_{item}, IG_{cat}, and IG_{total}, for evaluating the quality of the clustering result. IG_{item} is the information gain obtained on items and IG_{cat} is the information gain obtained on categories. $IG_{total} = IG_{item} + IG_{cat}$.

4.2.1 Experiment One: When the database size $|D|$ varies

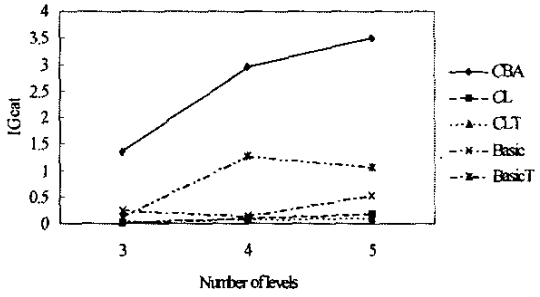
In this experiment, the scalability of CBA is evaluated by both the real data. By varying the real database size $|D|$ from $5K$ to $20K$, it is shown in Figure 6 that CBA significantly outperforms other algorithms in execution efficiency. Note that the logarithmic scale with base 10 is used in the y-axis of Figure 6 since the execution time of CBA is significantly shorter than those of other algorithms and the execution times of CBA increase linearly as the database size increases, indicating the good scale-up feature of algorithm CBA.

4.2.2 Experiment Two: When the number of taxonomy levels N^L varies in synthetic data

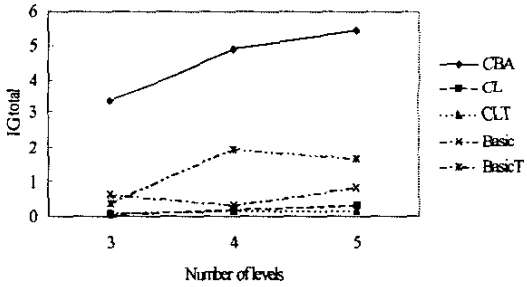
In the synthetic data experiment shown in Figure 7, we set $|D| = 100K$, $|T| = 5$, $|I| = 2$, $|L| = 2000$, $N^I = 5000$, $N^R = 100$, and N^L varies from 3 to 5. When the number of taxonomy levels increases, the number of internal (i.e., category) nodes also increases. Thus, the IG_{cat} increases so that CBA can obtain more information gain on categories than on items, indicating the advantage of CBA by employing the category-based adherence as the measurement.



(a) Information gain on items



(b) Information gain on categories



(c) Information gain in total

Figure 7. The IG values when the number of taxonomy levels N^L varies.

5 Conclusion

In view of the features of market-basket data, we devised in this paper a novel measurement, called the category-based adherence, and utilize this measurement to perform the clustering. With this category-based adherence measurement, we developed algorithm CBA for market-basket data with the objective to minimize the category-based adherence. A validation model based on Information Gain (IG) was also devised in this paper to assess the quality of clustering for market-basket data. As validated by both real and synthetic datasets, it was shown by our experimental results, with the taxonomy information, algorithm CBA devised in this paper significantly outperforms the prior works in both the execution efficiency and the clustering quality for market-basket data.

Acknowledgement

The authors are supported in part by the National Science Council, Project No. NSC 91-2213-E-002-034 and NSC 91-2213-E-002-045, Taiwan, Republic of China.

References

[1] Reuters-21578 news collection, http://www.research.att.com/ lewis/reuters21578.html.

[2] UCI Machine Learning Repository. *http://www.ics.uci.edu/~mlearn/MLRepository.html.*

[3] C. C. Aggarwal, C. M. Procopiuc, J. L. Wolf, P. S. Yu, and J.-S. Park. Fast Algorithms for Projected Clustering. *ACM SIGMOD International Conference on Management of Data*, pages 61–72, June 1999.

[4] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications. *ACM SIGMOD International Conference on Management of Data*, 27(2):94–105, June 1998.

[5] R. Agrawal and R. Srikant. Fast Algorithms for Mining Association Rules in Large Databases. *Proceedings of the 20th International Conference on Very Large Data Bases*, pages 478–499, September 1994.

[6] M. Charikar, C. Chekuri, T. Feder, and R. Motwani. Incremental Clustering and Dynamic Information Retrieval. *Proceedings of the 29th ACM Symposium on Theory of Computing*, 1997.

[7] M.-S. Chen, J. Han, and P. S. Yu. Data Mining: An Overview from a Database Perspective. *IEEE Transactions on Knowledge and Data Engineering*, 8(6):866–833, 1996.

[8] R. Duda and P. Hart. Pattern Classification and Scene Analysis. *Wiley, New York*, 1973.

[9] D. H. Fisher. Knowledge Acquisition via Incremental Conceptual Clustering. *Machine Learning*, 1987.

[10] S. Guha, R. Rastogi, and K. Shim. CURE: An Efficient Clustering Algorithm for Large Databases. *ACM SIGMOD International Conference on Management of Data*, 27(2):73–84, June 1998.

[11] S. Guha, R. Rastogi, and K. Shim. ROCK: A Robust Clustering Algorithm for Categorical Attributes. *Proceedings of the 15th International Conference on Data Engineering*, 1999.

[12] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. On Clustering Validation Techniques. *Journal of Intelligent Information Systems*, 2001.

[13] J. Han and Y. Fu. Discovery of Multiple-Level Association Rules from Large Databases. *Proceedings of the 21th International Conference on Very Large Data Bases*, pages 420–431, September 1995.

[14] A. K. Jain and R. C. Dubes. Algorithms for Clustering Data. *Prentice Hall*, 1988.

[15] A. K. Jain, R. P. Duin, and J. Mao. Statistical Pattern Recognition: A Review. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, pages 4–37, Jan. 2000.

[16] A. K. Jain, M. N. Murty, and P. J. Flynn. Data Clustering: A Review. *ACM Computer Surveys*, 31(3), Sept. 1999.

[17] C. Ordonez and E. Omiecinski. A Fast Algorithm to Cluster High Dimensional Basket Data. *Proceedings of the 1st IEEE International Conference on Data Mining (ICDM 2001)*, Nov./Dec. 2001.

[18] R. Srikant and R. Agrawal. Mining Generalized Association Rules. *Proceedings of the 21th International Conference on Very Large Data Bases*, pages 407–419, September 1995.

[19] A. Strehl and J. Ghosh. A Scalable Approach to Balanced, High-dimensional Clustering of Market-baskets. *Proceedings of the 7th International Conference on High Performance Computing*, December 2000.

[20] K. Wang, C. Xu, and B. Liu. Clustering Transactions Using Large Items. *Proceedings of ACM CIKM International Conference on Information and Knowledge Management*, 1999.

[21] Y. Xiao and M. H. Dunham. Interactive Clustering for Transaction Data. *Proceedings of the 3rd International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2001)*, Sept. 2001.

[22] C.-H. Yun, K.-T. Chuang, and M.-S. Chen. Self-Tuning Clustering: An Adaptive Clustering Method for Transaction Data. *Proc. of the 4th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2002)*, Sep. 2002.

[23] T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH: An Efficient Data Clustering Method for Very Large Databases. *ACM SIGMOD International Conference on Management of Data*, 25(2):103–114, June 1996.