K-Nearest Neighbor Query Processing using Utility Mining

Ms. Alma Mary Margret[1] Mrs. Kishore Sebastian[2]
[1]Student [2]Assistant Professor
[1,2]Department of Computer Science & Engineering
[1,2]St. Joseph's College of Engineering and Technology Choondacherry, Palai, Kottayam, Kerala, India

Abstract— Mining classification is the process of developing rules to group data tuples together based on certain features. Mining high utility itemsets from a transactional database refers to the process of discovery of items with high utility like profits. A large number of algorithms have been proposed for this problem, but it will generate a large number of itemsets. This will degrade the performance of the system. Inorder to overcome this problem here propose a method of utility mining. To perform the mining process, the system keeps a file called log file which contains the information's regarding the previous searches of a particular user. This will help to identify the browsing behavior of the user. Here mining is done by classifying the items into promising and unpromising factors. A minimum utility value is assigned in order to classify the items. If the percentage count of a particular item chosen by the user is greater than minimum utility, then that item is considered as promising factor. The items with promising factor will be listed in the list. If the percentage count of a particular item chosen by a user is less than minimum utility, then hat item will be filtered out from the list. User will choose a particular item from the list. Selected item and the current location of the user will be transferred as a query. Both current location and selected item is processed inorder to find the nearest neighbor. Current location is found with the help of HTML5 geolocation technology. Nearest neighbors are found by comparing the longitude and latitude of the users current location with the set of items in the database. The nearest neighbors will be listed and user will chose a neighbor as per his requirement. The selected item will be located in the map. The system will also sow the shortest path, distance and time required to reach from source to destination. If the user selects a particular item then it will be considered as successful request and it will be stored in the log file. This will improve the performance of the system. By using his method more accurate results are obtained.

Key words: Itemset, Utility Mining, Log File

I. INTRODUCTION

The explosive growth of location-detection devices (e.g., cellular phones, GPS-like devices, and RFIDs) results in a wide spread of location-based applications. Examples of these applications include location-based store finders, traffic congestion and location-based advertisements. Users who are registered with location-based services continuously send their location information to the location-based database server. Upon requesting to a service, a registered user has to issue a location-based query to server. This query is executed at the server based on the knowledge of the user location.

Data Mining can be defined as an activity that extracts some new nontrivial information contained in large databases. It is concerned with analysis of large volumes of data to automatically discover interesting regularities or relationships which in turn leads to better understanding of the underlying processes. The primary goal is to discover hidden patterns, unexpected trends in the data. Data mining activities uses combination of techniques from database technologies, statistics, artificial intelligence and machine learning. This process helps to mine previously unknown and potentially useful information from large databases. Generating useful patterns hidden in a database plays an essential role in data mining process, such as frequent pattern mining, weighted frequent pattern mining, and high utility pattern mining. Data Mining can be defined as an activity that extracts some new nontrivial information contained in large databases. Traditional data mining techniques have focused largely on detecting the statistical correlations between the items that are more frequent in the transaction databases.

Utility mining emerges as an important topic in data mining field. Mining high utility itemsets from databases refers to finding the itemsets with high profits. Utility of items in a transaction database consists of two aspects one is the importance of distinct items, which is called external utility, and the other one is the importance of items in transactions, which is called internal utility.

Utility of an itemset is the product of its external utility and its internal utility. If the utility of an item is not less than a user-specified minimum utility threshold then the itemset is a high utility itemset. If the utility of an itemset is less than a user-specified minimum utility threshold then it is called a low utility itemset. Mining high utility itemsets from databases has a wide range of applications such as website click stream analysis, business promotion in chain hypermarkets, cross marketing in retail stores online e-commerce management, mobile commerce environment planning and even finding important patterns in biomedical applications.

In this method we are using log file inorder to mine the itemset with high utility.

Log file contain information such as Time Stamp, User Name, Request, Result Status, etc. Every single request by the user will be recorded in the log file. The mined knowledge can be used to find the behaviour patterns of the user.

In the existing systems GPS devices are used to find the current location which is an online process. But in our proposed system is an offline system. Here it is not using the GPS device in order to find the current location of a user. Using HTML5 core technology, can find the current location without a GPS device.

II. RELATED WORK

In paper [8] presents middleware architecture and algorithms that can be used by a centralized location based broker service. The algorithms adjust the resolution of location information into spatial or temporal dimensions. It is to meet specified anonymity constraints based on the

entities whomay be using location services within a given area. This paper investigates a complimentary approach that concentrates on the principle of minimal collection. In this approach location-based services collect and use only de-personalized data—that is, practically anonymous data [14]. This approach promises benefits for both parties. For the service provider, practically anonymous data causes less overhead. It can be collected, processed, and distributed to third parties without user consent. For data subjects, it removes the need to evaluate potentially complex service provider privacy policies. Practical anonymity requires that the subject cannot be reidentified (with reasonable efforts) from the location data. Consider a message to a road map service that comprises a network address, a user ID, and coordinates of the current location. Identifiers like the user ID and the network address are obvious candidates for reidentification attempts. For anonymous service usage, the user ID can be omitted and the network address problem is addressed by mechanisms such as Crowds [12] or Onion Routing [13]. It provides sender anonymity.

A. Fast Algorithms for Mining Association Rule:

We consider the problem of discovering association rules between items in a large database of sales transactions. We present two new algorithms for solving this problem that are fundamentally different from the known algorithms. Empirical evaluation shows that these algorithms outperform the known algorithms by factors ranging from three for small problems to more than an order of magnitude for large problems. We also show how the best features of the two proposed algorithms can be combined into a hybrid algorithm, called Apriori Hybrid. Scale-up experiments show that Apriori Hybrid scales linearly with the number of transactions. Apriori Hybrid also has excellent scale-up properties with respect to the transaction size and the number of items in the database.

Given a set of transactions D, the problem of mining association rules is to generate all association rules that have support and confidence greater than the user-specified minimum support and minimum confidence respectively. AIS algorithm is used to find all association rules. Another algorithm to perform this task is called the SETM algorithm. In this paper, it presents two new algorithms, Apriori and AprioriTid that differ fundamentally from the previous algorithms.

Experimental results show that the proposed algorithms always outperform the earlier algorithms. The performance increases with problem size, and ranges from a factor of three for small problems to more than an order of magnitude for large problems. The best features of Apriori and AprioriTid are be combined into a hybrid algorithm, called Apriori Hybrid. Experiments show that the Apriori Hybrid has excellent scale-up properties, opening up the feasibility of mining association rules over very large databases.

In order to discover large itemsets the algorithm makes multiple passes over the data. In the first pass, it counts the support of individual items and determines which items are larger, that is the items having minimum support. For each subsequent pass, the process starts with a seed set of itemsets which is found large in the previous pass. This seed set is used for generating new potentially large itemsets, called candidate itemsets. Then it counts the actual support for these candidate itemsets during the pass over the data. At the end of the pass, it determines which of the candidate itemsets are actually large, and they become the seed for the next pass. This process continues until no new large itemsets are found.

B. Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases

Incremental and interactive data mining provide the ability to use previous data structures and mining results in order to reduce unnecessary calculations when a database is updated. Frequent pattern mining plays an important role in data mining applications, its two limitations are, first, it treats all items with the same importance/weight/price and, second, in one transaction, each item appears in a binary (0/1) form, i.e., either present or absent. The importance of an itemset can be measured by the concept of utility. It can handle the dataset with nonbinary frequency values of each item in transactions, and also with different profit values of each item. Therefore, utility mining represents real world market data. By utility mining, several important business area decisions like maximizing revenue or minimizing marketing or inventory costs can be considered and knowledge about itemsets / customers contributing to the majority of the profit can be discovered. The previous works are based on a fixed database and did not consider that one or more transactions could be deleted, inserted, or modified in the database. By using incremental and interactive high utility pattern (HUP) mining, can use the previous data structures and mining results, and avoid unnecessary calculations when the database is updated or the mining threshold is changed.

Moreover, if the algorithms presented in the previous works want to calculate which patterns cover 20 percent of the total profit, then their internal data structures are designed in such a way that they can only calculate the asked amount. If the amount is 15 percent of the total profit, then previous algorithms have to build their data structures again. They cannot take any advantages from their previous design. However, in the real world, the businessmen need to repeatedly change the minimum threshold according to their business requirements. Thus, the "build once mine many" property (by building the data structure only once, several mining operations can be done) is required to solve this interactive mining problem.

C. High Utility Itemset Mining

Data Mining can be defined as an activity that extracts some new nontrivial information contained in large databases. Traditional data mining techniques have focused largely on detecting the statistical correlations between the items that are more frequent in the transaction databases. Also termed as frequent itemset mining, these techniques were based on the rationale that itemsets which appear more frequently must be of more importance to the user from the business perspective .In this paper throw light upon an emerging area called Utility Mining which not only considers the frequency of the itemsets but also considers the utility associated with the itemsets. The term utility refers to the importance or the usefulness of the appearance of the itemset in transactions quantified in terms like profit, sales or any other user preferences. In High Utility Itemset

Mining the objective is to identify itemsets that have utility values above a given utility threshold.

The primary goal is to discover hidden patterns, unexpected trends in the data. Data mining activities uses combination of techniques from database technologies, statistics, artificial intelligence and machine learning. An itemset can be defined as a non-empty set of items. An itemset with k different items is termed as a k-itemset. For e.g. {bread, butter, milk} may denote a 3-itemset in a supermarket transaction. Frequent itemsets are the itemsets that appear frequently in the transactions. The goal of frequent itemset mining is to identify all the itemsets in a transaction dataset. Frequent itemset mining plays an essential role in the theory and practice of many important data mining tasks, such as mining association rules, long patterns, emerging patterns, and dependency rules.

In utility based mining the term utility refers to the quantitative representation of user preference i.e. the utility value of an itemset is the measurement of the importance of that itemset in the user's perspective. Here note that the sales analyst is not interested in the number of transactions that contain the itemset but he or she is only concerned about the revenue generated collectively by all the transactions containing the itemset. In practice the utility value of an itemset can be profit, popularity, page-rank, measure of some aesthetic aspect such as beauty or design or some other measures of user's preference. Formally an itemset S is useful to a user if it satisfies a utility constraint i.e. any constraint in the form u(S) >= minutil, where u(S) is the utility value of the itemset. A minutil is a utility threshold defined by the user. In our example if it take utility of an itemset as the unit profit associated with the sale of that itemset then with utility threshold minutil = 500 then the itemset ABC has a utility value of 555 which means that this itemset is of interest to the user even though its support value is just 20%.Since while considering the total utility of an itemset S multiply the utility values of the individual items consisting the itemset S with the corresponding frequencies of the individual items of S in the transactions that contain S, so the utility based mining approach can be said to be measuring the significance of an itemset from two dimensions. The first dimension being the support value of the itemset i.e the frequency of the itemset and the second dimension is the semantic significance of the itemset as measured by the user.

D. Efficient Data Mining for Path Traversal Patterns

In this paper, shall explore a new data mining capability which involves mining access patterns in a distributed information-providing environment where documents or objects are linked together to facilitate interactive access. it is important to note that, since users are traveling along the information-providing services to search for the desired information, some objects are visited because of their locations rather than their content, showing the very difference between the traversal pattern problem and others which are mainly based on customer transactions. This unique feature of the traversal pattern problem unavoidably increases the difficulty of extracting meaningful information from a sequence of traversal data. However, as these information-providing services are becoming increasingly popular nowadays, there is a growing demand for capturing user behavior and improving the quality of such services. As a result, the problem of mining traversal patterns has become too important not to address immediately

E. Efficient Mining of Partial Periodic Patterns in Time Series Database

Partial periodicity search, i.e., search for partial periodic patterns in time-series databases, is an interesting data mining problem. Previous studies on periodicity search mainly consider finding full periodic patterns, where every point in time contributes (precisely or approximately) to the periodicity. However, partial periodicity is very common in practice since it is more likely that only some of the time episodes may exhibit periodic patterns. Here present several algorithms for efficient mining of partial periodic patterns, by exploring some interesting properties related to partial periodicity, such as the Apriori property and the max-sub pattern hit set property, and by shared mining of multiple periods. The max-sub pattern hit set property is a vital new property which allows us to derive the counts of all frequent patterns from a relatively small subset of patterns existing in the time series. Show that mining partial periodicity needs only two scans over the time series database, even for mining multiple periods.

Finding periodic patterns in time series databases is an important data mining task with many applications. Many methods have been developed for searching periodicity patterns in large data sets. However, most previous methods on periodicity search are on mining full periodic patterns, where every point in time contributes (precisely or approximately) to the cyclic behavior of the time series. For example, all the days in the year approximately contribute to the season cycle of the year. A useful related type of periodic patterns, called partial periodic patterns, which specify the behavior of the time series at some but not all points in time, have not received enough attention. Partial periodicity is a looser kind of periodicity than full periodicity, and it exists ubiquitously in the real world. The purpose of the current paper is to fill the gap by considering the efficient mining of partial periodic patterns. Most methods for finding full periodic patterns are either inapplicable to or prohibitively expensive for the mining of partial periodic patterns, because of the mixture of periodic events and non-periodic events in the same period

III. METHODOLOGY

The proposed system work based on the concept of utility mining. The system will store the details of items selected by the user in each login. Each successful request send by the user will be stored in the log file. This log file is mined inorder to find the promising factor and unpromising factor. A minimum utility is considered. If the percentage of a particular item selected by the user is higher than minimum utility, then it is considered as promising factor. If the percentage of a particular item is less than minimum utility then it is considered as unpromising factor. All the items with promising factor will be listed. User will choose a particular item from the list based on this needs.

1) Client:

In order to perform the action user need to register. Data will be provided only to the registered users. The browsing behaviour of the user will be stored in the log file. This will

help to find the most chosen itemset of a particular user. This will help to reduce the search time for finding the neighbour. So the user can view the nearest facilities in short span of time.

2) Find the Current Location of the client :

After the successful login the current location is to be found. Current location is used to find the nearest facilities. Current location is also used to find the itemset a particular user has chosen more in a particular user. HTML5 geolocation method is used to find the current location of the user. Using this method the longitude and latitude of the user is obtained. This longitude and latitude is used to find the nearest neighbour from the database. The current location will be stored in the log file along with the browsing behavior of user.

3) Mining the Log File :

Every successful request send by the user will be recorded in Log file. So every search made by the user will be in the log file. This log file is mined in order to find the itemset associated to a particular user. By looking in to this file, the search pattern of the user can be found easily. Only the request with appropriate selection will be stored in the log file. Promising and unpromising items are categorised from this file.

4) Send Query with Current Location :

After mining the log file the request will be passed to find the nearest neighbour. The request will contain the current location and the item selected by the user. The itemset are obtained by mining the log file. The items having promising factor will be listed. User has to choose a particular item from this list

5) List the nearest neighbours related to the selected item:

Based on the current location and item chosen by the user the system will list the nearest neighbours. Here we are comparing the longitude and latitude of user's current location and the neighbours. Items having matching longitude and latitude with the user's current location will be listed as set of nearest neighbors.

6) Choose a neighbour from the list :

From the list of items user has to select a particular item. If a user selects a particular item from that list, then it is considered as a successful request and it will be stored in the log file. The selected item will be displayed in the map. Also the user is provided with the distance and time required to reach the destination from the users current location.

A. System Model:

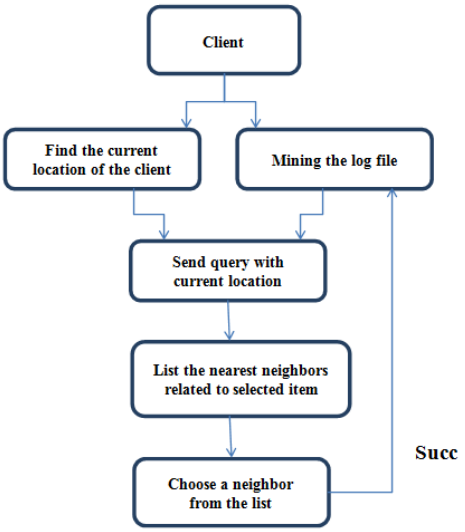


Fig. 1: Model

B. Data Owner

Due to the rapid advancements in network technology, the cost of transmitting a terabyte of data over long distances has decreased significantly in the past five years. In addition, the total cost of data management is five to ten times higher than the initial acquisition costs and it is likely that computing solution costs will be dominated by people costs in the future. Consequently, there is a growing interest in outsourcing database management tasks to third parties that can provide these tasks for a much lower cost due to the economy of scale. This new outsourcing model has the apparent benefits of reducing the costs for running DBMSs independently and enabling enterprises to concentrate on their main businesses. On the other hand, there are two new concerns with this model.

In order to respond for a client's query it needs the necessary information about the data owners. Data owners are the nearest neighbors for a client's query. So the data owner's wants to provide necessary information about their organization. To list the information of a data owner in nearest neighbor list he/she should be registered with necessary information. The information about this organization will be stored in the data base. And this data base will be supplied to the service provider. After receiving the client request, the query will be processed. The processing will be performed with the help of data provided by the data owner's. The data owners should be very careful while providing the information about his/her organization. Because this information will be provided to the clients as there query result.

C. Service Provider

1) Step 1: Data owner will send their necessary information and there location information to the service provider. And this information's are stored in the data base for further examination. Only the information of registered user will be available to the client as his/her request result.
2) Step 2: Client sends his/her query point to the service provider. The query point will contain the information about the client's current location. The current locations are obtained with the help of HTML 5 technology.
3) Step 3: Server perform the processing of the query from the client. The processing is done with the help of information provided by the data owners. With the help of information from the data owner the server will generate the neighbors. And the neighbors will be listed.
4) Step 4: From list of nearest neighbors, the user will choose one according to his choice. If the user makes a choice, then it is considered as successful request. This path will be stored in the log file in order to make the further search easier.

D. Geolocation

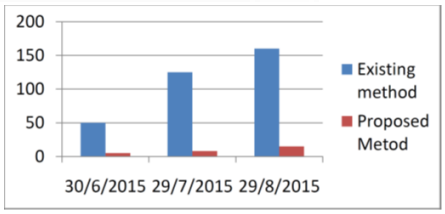
Geolocation is the identification of the real-world geographic location of an object, such as a mobile phone, Internet-connected computer terminal, etc. Geolocation may refer to the practice of assessing the location, or to the actual assessed location. Geolocation is closely related to the use of positioning systems but may be distinguished from it by a

greater emphasis on determining a meaningful location rather than just a set of geographic coordinates. HTML5 added set of new APIs under title Geolocation that helps the applications to identify location related information from visiting users. The Geolocation API defines a high-level interface to location information associated only with the device hosting the implementation, such as latitude and longitude. The API itself is agnostic of the underlying location information sources.

In order to utilize Geolocation API in a web application, it must understand the API structures. The browser that supports Geolocation API provides navigator object with geolocation object. This object can be used to retrieve location info. The geolocation API is now supported in the majority of browsers, but it is good practice to always check for support before you do anything. You can easily check for compatibility by testing for the presence of the geolocation object. The geolocation API offers a simple method to obtain the user's current location. The method to find the current location will return the latitude and longitude if it is successful. To display the result in a map, you need access to a map service that can use latitude and longitude

IV. Experiment and Result

As per the performance of the existing system more accurate results are obtained. There is great variation in the number of items being selected by the user in the existing system and proposed system. In the existing system the count is incremented even if it is a valid or invalid request. So the utility of the item will cross the minimum utility level with in short interval. This will degrade the performance of the system. As per the proposed system item is stored in the log file if it is a valid request. So the utility of the item will be incremented only if the item selection is a valid one. This will improve the performance of the system and more accurate result will be obtained.



V. Conclusion

The emergence of mobile devices with fast Internet connectivity and geo - positioning capabilities has led to a revolution in customized location-based services (LBS), where users are enabled to access information about points of interest (POI) that are relevant to their interests and are also close to their geographical coordinates. Here we are using a log file inorder to find the nearest neighbor. Utility mining technique is used here. With the help of log file the items are categorized into promising factor and unpromising factor. Only the items with promising factor will be listed. Other items will be filtered out. The search made by the user will be stored in the log file only if it is a successful request. The processing of the query send by the client is processed in the service provider. The service provider will generate nearest neighbors based on the client's current location and item chosen by the user. The item will be chosen from the mined data. This will helps to improve the performance of the system. And it will help to generate most accurate results.

Data owners are the one who act as the nearest neighbor for clients query. The information needed for the processing of the query is provided by the data owner. This information will also help the client to get more details about that organization. The data owners will enter their information during the registration process. Client will send the request for the nearest neighbor. This query will contain the current location of the user as well as the selected category. With the help of this current location the query is processed by the service provider and list of nearest neighbor is displayed in the map. The current location will be in the form of longitude and latitude.HTML5 Geolocation method is use to find the current location of the user. The data owners also use this method in order to register theirorganization. So the organizations location coordinates will help to find the query send by the client. HTML5 Geolocation is a java script to find the latitude and longitude of the user. The nearest neighbors obtained as a result of client query is displayed in the map. The longitude and longitude will help to display the nearest neighbor in the map.

References

[1] Agrawal, Rakesh, and RamakrishnanSrikant. "Fast algorithms for mining association rules."In Proc. 20th int. conf. very large data bases, VLDB, vol. 1215, pp. 487-499. 1994.
[2] Agrawal, Rakesh, and RamakrishnanSrikant. "Mining sequential patterns."In Data Engineering, 1995. Proceedings of the Eleventh International Conference on, pp. 3-14. IEEE, 1995.
[3] Ahmed, ChowdhuryFarhan, Syed KhairuzzamanTanbeer, Byeong-SooJeong, and Young-Koo Lee. "Efficient tree structures for high utility pattern mining in incremental databases." Knowledge and Data Engineering, IEEE Transactions on 21, no. 12 (2009): 1708-1721.
[4] Chang, JoongHyuk. "Mining weighted sequential patterns in a sequence database with a time-interval weight." Knowledge-Based Systems 24, no. 1 (2011): 1-9.
[5] Cai, Chun Hing, Ada WC Fu, C. H. Cheng, and W. W. Kwong. "Mining association rules with weighted items." In Database Engineering and Applications Symposium,1998.Proceedings.IDEAS'98. International, pp. 68-77. IEEE, 1998.
[6] Chen, Ming-Syan, Jong Soo Park, and Philip S. Yu. "Efficient data mining for path traversal patterns." Knowledge and Data Engineering, IEEE Transactions on 10, no. 2 (1998): 209-221.
[7] Chan, Raymond Chan, Qiang Yang, and Yi-Dang Shen. "Mining high utility itemsets."In Data Mining, 2003.ICDM 2003. Third IEEE International Conference on, pp. 19-26. IEEE, 2003.
[8] Erwin, Alva, Raj P. Gopalan, and N. R. Achuthan. "Efficient mining of high utility itemsets from large datasets."In Advances in Knowledge Discovery and Data Mining, pp. 554-561.Springer Berlin Heidelberg, 2008.

[9] Creighton, Chad, and Samir Hanash. "Mining gene expression databases for association rules." Bioinformatics 19, no. 1 (2003): 79-86.
[10] Eltabakh, Mohamed Y., MouradOuzzani, Mohamed A. Khalil, Walid G. Aref, and Ahmed K. Elmagarmid. "Incremental Mining for Frequent Patterns in Evolving Time Series Datatabases." (2008).
[11] Georgii, Elisabeth, Lothar Richter, Ulrich Rückert, and Stefan Kramer. "Analyzing microarray data using quantitative association rules." Bioinformatics 21, no. suppl 2 (2005): ii123-ii129.
[12] Han, Jiawei, Guozhu Dong, and Yiwen Yin. "Efficient mining of partial periodic patterns in time series database."In Data Engineering, 1999.Proceedings., 15th International Conference on, pp. 106-115. IEEE, 1999.
[13] Han, Jiawei, and Yongjian Fu. "Discovery of multiple-level association rules from large databases." In VLDB, vol. 95, pp. 420-431. 1995.
[14] Han, Jiawei, Jian Pei, and Yiwen Yin. "Mining frequent patterns without candidate generation."In ACM SIGMOD Record, vol. 29, no. 2, pp. 1-12.ACM, 2000.
[15] Lee, Seung-Cheol, Juryon Paik, Jeewoong Ok, Insang Song, and Ung Mo Kim. "Efficient mining of user behaviors by temporal mobile access patterns." Int'l J. Computer Science Security 7, no. 2 (2007): 285-291.
[16] Li, Hua-Fu, Hsin-Yun Huang, Yi-Cheng Chen, Yu-Jiun Liu, and Suh-Yin Lee. "Fast and memory efficient mining of high utility itemsets in data streams."In Data Mining, 2008.ICDM'08. Eighth IEEE International Conference on, pp. 881-886. IEEE, 2008.
[17] Li, Yu-Chiang, Jieh-Shan Yeh, and Chin-Chen Chang. "Isolated items discarding strategy for discovering high utility itemsets." Data & Knowledge Engineering 64, no. 1 (2008): 198-217.
[18] Lin, Chih-Hsiang, Ding-Ying Chiu, Yi-Hung Wu, and Arbee LP Chen. "Mining Frequent Itemsets from Data Streams with a Time-Sensitive Sliding Window."In SDM, pp. 68-79. 2005.
[19] Liu, Ying, Wei-keng Liao, and AlokChoudhary. "A fast high utility itemsets mining algorithm."In Proceedings of the 1st international workshop on Utility-based data mining, pp. 90-99.ACM, 2005.
[20] Martinez, Ricardo, Nicolas Pasquier, and Claude Pasquier. "GenMiner: mining non-redundant association rules from integrated gene expression data and annotations." Bioinformatics 24, no. 22 (2008): 2643-2644.
[21] Pei, Jian, Jiawei Han, Hongjun Lu†, ShojiroNishio, Shiwei Tang, and Dongqing Yang. "H-Mine: Fast and space-preserving frequent pattern mining in large databases." IIE transactions 39, no. 6 (2007): 593-605.
[22] Pei, Jian, Jiawei Han, BehzadMortazavi-Asl, Jianyong Wang, Helen Pinto, Qiming Chen, UmeshwarDayal, and Mei-Chun Hsu. "Mining sequential patterns by pattern-growth: The prefixspan approach." Knowledge and Data Engineering, IEEE Transactions on 16, no. 11 (2004): 1424-1440.
[23] Shie, Bai-En, Hui-Fang Hsiao, Vincent S. Tseng, and S. Yu Philip. "Mining high utility mobile sequential patterns in mobile commerce environments."In Database Systems for Advanced Applications, pp. 224-238.Springer Berlin Heidelberg, 2011.
[24] Shie, Bai-En, Vincent S. Tseng, and Philip S. Yu. "Online mining of temporal maximal utility itemsets from data streams." In Proceedings of the 2010 ACM Symposium on Applied Computing, pp. 1622-1626.ACM, 2010.
[25] Sun, Ke, and FengshanBai. "Mining weighted association rules without preassigned weights." Knowledge and Data Engineering, IEEE Transactions on 20, no. 4 (2008): 489-495.
[26] Tanbeer, Syed Khairuzzaman, ChowdhuryFarhan Ahmed, Byeong-SooJeong, and Young-Koo Lee. "Efficient frequent pattern mining over data streams." In Proceedings of the 17th ACM conference on Information and knowledge management, pp. 1447-1448.ACM, 2008.

