Alexandria Engineering Journal (2015) 54, 163–174

HOSTED BY

Alexandria University

Alexandria Engineering Journal

www.elsevier.com/locate/aej
www.sciencedirect.com

ORIGINAL ARTICLE

Efficient mining fuzzy association rules from ubiquitous data streams

Amal Moustafa *, Badr Abuelnasr, Mohamed Said Abougabal

Computer and Systems Engineering Department, Faculty of Engineering, Alexandria University, Egypt

Received 22 January 2015; revised 9 March 2015; accepted 15 March 2015
Available online 11 April 2015

KEYWORDS

Data mining;
Fuzzy association rules;
Fuzzy sets;
Data streams;
Ubiquitous data streams

Abstract Due to the development in technology, a number of applications such as smart mobile phone, sensor networks and GPS devices produce huge amount of ubiquitous data in the form of streams. Different from data in traditional static databases, ubiquitous data streams typically arrive continuously in high speed with huge amount, and changing data distribution. Dealing with and extracting useful information from that data is a real challenge. This raises new issues, that need to be considered when developing association rule mining techniques for these data. It should be noted, that data, in the real world, are not represented in binary and numeric forms only, but it may be represented in quantitative values. Thus, using fuzzy sets will be very suitable to handle these values.

In this paper the problem of mining fuzzy association rules from ubiquitous data streams is studied, and a novel technique FFP_USTREAM (Fuzzy Frequent Pattern Ubiquitous Streams) is developed. This technique integrates fuzzy concepts with ubiquitous data streams, employing sliding window approach, to mine fuzzy association rules. In addition, the complexity and the efficiency of this technique are discussed. Examples of real data sets are used **to test** the proposed technique. Further research issues are also suggested.

1. Introduction

Recent emerging applications, such as network traffic monitoring, sensor network data analysis, web click stream mining, power consumption measurement, and dynamic tracing of stock market fluctuations, call for studying a new kind of data. This is called stream data, which can be continuous, potentially infinite flow of information, as opposed to finite, statically stored data sets. Stream Data Mining is the process of extracting knowledge structures from continuous, and rapid data records.

The dissemination of data streams systems, wireless networks and mobile/handheld devices motivates the need for an efficient data analysis tool capable of gaining insights about these continuous data streams [1]. Ubiquitous data streams mining (UDM) is the process of pattern discovery on mobile, embedded and ubiquitous devices. It represents the next generation of data mining systems, that will support the intelligent and time-critical information needs of mobile users, and will facilitate "anytime, anywhere" data mining.

Fuzzy logic is a type of logic used in artificial intelligence. It is referred to as a multi-valued logic.Instead of having two

* Corresponding author.

Peer review under responsibility of Faculty of Engineering, Alexandria University.

http://dx.doi.org/10.1016/j.aej.2015.03.015

values (true and false), there are a continuum of possible truth values [2]. In fuzzy logic, every proposition is a statement that is assigned a number between 0 (false) and 1 (true), such a statement is called a fuzzy proposition. Fuzzy logic provides a powerful tool to categorize a concept in an abstract way by introducing vagueness.

Many data streams applications exist, that require association rule mining, such as network traffic monitoring and web click streams analysis. These applications' goal is to discover important associations among items as the presence of some items will imply the presence of others.

Fuzzy association rule approach could combine data mining results with human expertise and background knowledge, in the form of rules, to attain labeled classes for classification of data streams. Another advantage of the fuzzy logic approach is that it gives classification results, which include a degree of probability.

This paper demonstrates the effectiveness of Fuzzy Association Rules Mining from Ubiquitous Data Streams. This will be revealed in the coming sections. For this purpose, the remaining part of the paper is organized as follows: the work related to Fuzzy Association rules mining and ubiquitous data streams mining are reviewed and summarized in Section 2. An efficient fuzzy association rules mining technique from ubiquitous data streams is proposed in Section 3, and its complexity is analyzed in Section 4. Moreover,experimental results are discussed in Section 5. The paper is concluded and future research issues are presented, in Section 6.

2. Related work

This paper belongs to different inter-related research fields. The two main related topics of this work are presented: Ubiquitous Data Streams Mining Techniques that can effectively analyze continuously streaming data and Fuzzy association rules mining algorithms. These two fields will be surveyed.

2.1. Ubiquitous data streams mining

The approach, based on finite statically stored data sets, is not satisfactory in several applications. These include wireless network analysis, intrusion detection, stock market analysis, sensor network data analysis, and, in general, any setting in which every information available should be used to make an immediate decision. Such situations demand new algorithms, that are able to cope with evolutions of data as shown in Table 1 [3].

Ubiquitous Data Mining (UDM) is the time-critical process of pattern discovery in data streams in a wireless environment [4]. The widespread use of mobile devices, with increasing computational capacity, is leading to the emergence of the ubiquitous computing paradigm. This paradigm facilitates continuous access to data and information by mobile users with handheld devices [5]. UDM is the process of analyzing data from distributed and heterogeneous sources with mobile devices or within sensor networks, where the data is continuously streamed to the device, and where there are temporal constraints, that necessitate analysis ''anytime, anywhere'' [6].

In the following subsections preprocessing required for data streams is presented.

...

2.1.1. Data streams techniques

Research problems and challenges that appeared in mining data streams have their solutions using well established statistical and computational approaches. These solutions could be categorized to data-based and task-based ones. This classification is depicted in Fig. 1 [7]. In data-based solutions, the idea is to examine only a subset of the whole dataset or to transform the data vertically or horizontally to an approximate smaller size data representation. On the other hand, in task-based solutions, techniques from computational theory have been adopted to achieve time and space efficient solutions.

2.1.1.1. Data-based techniques. Data-based techniques refer to summarizing the whole dataset or choosing a subset of the

Figure 1 Classification of data streams preprocessing methods [7].

incoming stream to be analyzed. Sampling, load shedding and sketching techniques are most commonly used [7].

2.1.1.2. Task-based techniques. Task-based techniques are those methods that modify existing techniques,or invent new ones in order to address the computational challenges of data streams processing. Approximation algorithms, window models (landmark window model, damped window model, or sliding window model), and algorithm output granularity represent this category [7].

2.1.2. Data streams association rules

A number of algorithms have been proposed for extracting knowledge from streaming information. These include clustering, classification, frequency counting and time series analysis techniques. As the number of applications on mining data streams grows rapidly, there is an increasing need to perform association rule mining on stream data. An example application of data streams association rule mining is to estimate missing data in sensor networks [8]. Another example is to predict frequency estimation of Internet packet streams [9].

In the MAIDS project [10], this technique is used to find alarming incidents from data streams. Association rule mining can also be applied to monitor manufacturing flow to predict failure, or generate reports based on web log streams [11].

Several frequent pattern algorithms [12–18] were suggested in the literature. Most of these approaches either produce approximate results, discover special subsets (closed, maximal, and constraint-based) or provide exact results. These are compared in Table 2 [19].

2.2. Fuzzy association rules mining

Most studies have shown how binary valued transaction data may be handled. However, transaction data in real-world applications, usually consist of fuzzy and quantitative values. Thus, designing sophisticated data-mining algorithms, able to deal with various types of data presents a challenge to workers in this research field. Three Fuzzy Association rules algorithms [20–22] were recently proposed in the literature. These are compared in Table 3.

2.3. Need to extend existing work

Close study of Tables 2 and 3 reveals the need to develop a novel technique for mining fuzzy association rules from ubiquitous data streams.This technique should enjoy the following features:

- the ability of handling the continuous flow of data streams,
- the ability of handling data concept drift over time,
- the ability of handling memory bounded size, suitable for ubiquitous applications,
- facilitates data analysis and quick decision support for users, by scanning the flow of data streams only once,
- could determine fuzzy sets & membership functions,
- could determine user-defined min support, and
- gives accurate results.

In the following sections, an efficient Fuzzy Frequent Pattern Ubiquitous Streams technique (FFP_USTREAM), that satisfies the above features is proposed.

3. Proposed technique: FFP- USTREAM

The proposed technique consists of four steps, as indicated in Algorithm A.1. These are detailed below.

3.1. Step 1: Specifying sliding window

To handle continuously generated ubiquitous data streams, a sliding window model is used as shown in Algorithm A.2, to find exact recent fuzzy frequent patterns. The first step is to determine a sliding window size. Old transactions are expired,

once the new transactions arrive into the current window. The size of the window depends on the application and the system resources [19]. In this work, the size of window is considered constant.Variable window size is left for future work.

An example of sliding window in fuzzy data streams is depicted in Fig. 2. In this example, the window size (number of panes) and pane size (number of transactions) are set to 2. The notation (A:5) indicates the number of purchased units of item A.

3.2. Step 2: Fuzzification

Fuzzification of quantative values of data streams attributes involves two processes. Firstly, derive the fuzzy sets for variables and represent them with linguistic terms [23]. Secondly, estimate the membership functions [24]. In practice, membership functions can have multiple different types, such as triangular waveform,trapezoidal waveform, Gaussian waveform, bell-shaped waveform, sigmoidal waveform, and S-curve waveform. In this work triangular and trapezoidal waveforms are used.

Figure 2 Sliding window-based data streams.

3.3. Step3: Tree structure construction

In order to infer Fuzzy Association Rules from ubiquitous data streams, Dynamic Fuzzy Frequent Pattern tree (DFFP-tree) is constructed. The basic structure of DFFP-tree is similar to Frequent Pattern tree (FP-tree) [25]. However, a membership value is added to each node as shown in Algorithm A.3.

In order to render this technique suitable for limited resources ubiquitous devices, an Algorithm A.4 is added to delete old panes, and insert new panes (i.e. restructuring trees), as windows are changed.

Step by step construction procedure is shown in Fig. 3. In addition a comparative analysis between DFFP-tree and previous tree structures is illustrated in Table 4.

3.4. Step 4: Extracting fuzzy association rules

Once the DFFP-tree is constructed, Algorithm A.5 is used to generate the complete set of exact (not approximate) fuzzy frequent patterns from the current window. Hence, fuzzy

Figure 3 Dynamic Fuzzy Frequent Pattern tree (DFFP-tree) construction.

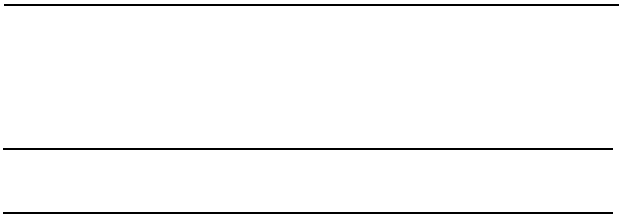

Figure 4 Membership functions with three Regions Fuzzy Sets.

association rules are deduced according to a certain confidence using Algorithm A.6.

4. Discussion of the complexity of the Algorithms

In this section, the complexity of FFP-USTREAM is analyzed. The following notations are defined for this discussion.

P, is the number of transactions in each window.
S, is the number of fuzzy sets.
I, is the number of distinct attributes relative to the study.

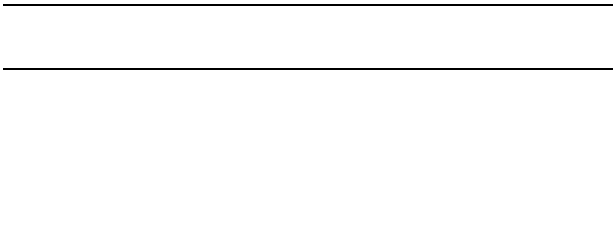

Figure 5 Membership functions with five Regions Fuzzy Sets.

The discussion is separated into two parts: runtime complexity, and memory complexity.

4.1. Runtime complexity

It may be shown that the run time complexity is divided into three major steps.

- Fuzzification process, which is the mapping of attributes quantitative values to fuzzy sets. The number of mapping will be of order $O(|S|*|P|*|I|)$ maps.
- DFFP-tree construction, it is the core of the proposed technique. The complexity will be of order $O(|P|*|I|)$, representing updating node value, or creating new node.
- Extracting the fuzzy association rules, its complexity will be of order $O(|P|*|I|)$, representing number of comparison.

4.2. Memory complexity

The memory complexity varies with the window size and the order of items in data streams transactions. Assuming worst-case scenario, number of nodes in DFFP-tree may be derived to be of order $O(|P|*|I|)$.

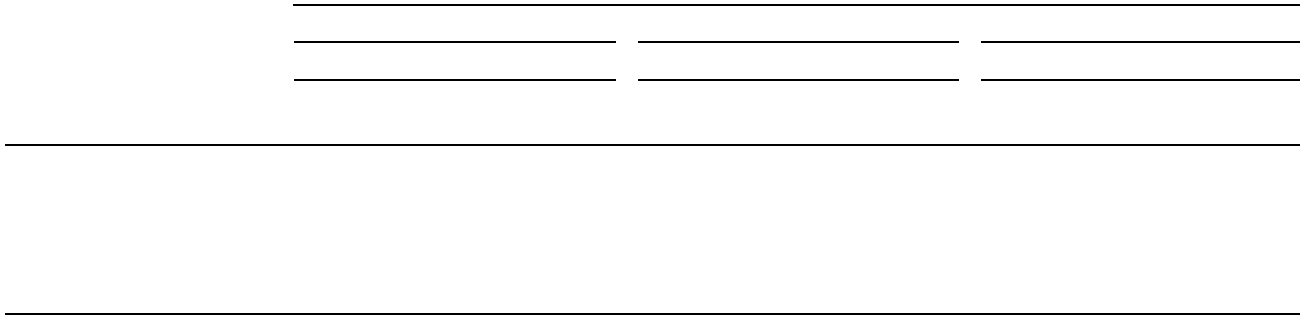

Figure 6a "Accidents" runtime distribution on variation of fuzzy regions.

Figure 6b "Retail" runtime distribution on variation of fuzzy regions.

5. Experimental results

To assess the validity of the proposed technique, experiments are conducted using real data sets, in Microsoft Visual Studio 2010 C#. Two different data sets were chosen, "Accidents" and "Retail" data sets. These were downloaded from fimi.cs.helsinki.fi/data/ and summarized in Table 5.

5.1. Required parameters

Two main parameters have great effect on the experimental results performance, the selection of window size & pane size, and the selection of an appropriate membership function with fuzzy sets regions.

5.1.1. Window & pane size

DFFP-tree dynamically restructures itself after each window slide. The runtime and memory complexity vary depending on the window parameters w (window size) and p (pane size), as mentioned in Section 4 above and illustrated in Section 5.2.

5.1.2. Membership functions & fuzzy sets

An important step in FFP_USTREAM, was the selection of the membership functions and the number of fuzzy sets. Thus, an appropriate membership function must be selected carefully depending on the nature of the data sets and the user application.In this work, Fuzzy Logic Tool Box in Matlab 7 was used.

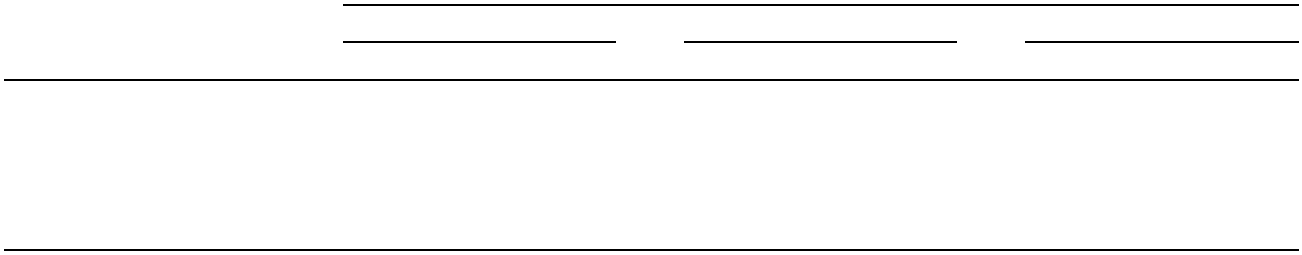

Figure 7a "Accidents" node distribution.

Figure 7b "Retail" node distribution.

In Fig. 4, the membership function with three fuzzy regions ('LOW', 'MIDDLE', 'HIGH') is presented. The membership function with five fuzzy regions ('VERYLOW', 'LOW', 'MIDDLE', 'HIGH', 'VERY HIGH') is also presented in Fig. 5.

5.2. Analysis of results

In the following sections, the results of DFFP-tree construction are presented. Extensive experimental analyses show that DFFP-tree is highly efficient in terms of memory storage when finding exact fuzzy frequent patterns from a high-speed ubiquitous data streams.

5.2.1. Runtime trend

For each data sets, the overall runtime,of DFFP-tree, based on changes in the fuzzy set regions was evaluated. The results are depicted in Table 6 and Fig. 6. In the graphs, *y*-axes represent the total time including DFFP-tree construction time and tree restructuring time. The *x*-axes show the variation of the product of number of window size and pane size.

A close study of Fig. 6 reveals the following trends.

- Larger pane and window sizes imply longer total tree construction & restructuring times for the two types of data sets (Dense & Sparse).
- The number of fuzzy sets regions affects the tree construction runtime. A long runtime will be required with large fuzzy sets regions.
- Higher runtime will be required for dense data sets as compared to sparse data sets.

5.2.2. Memory efficiency

Experiments have been conducted to verify the memory requirements for DFFP-tree construction on different data sets by varying the window size. FFP_USTREAM always captures the window contents in full (i.e., not depending on support

Table 8 Fuzzy frequent patterns.

threshold) in order to minimize the database scan to only once. Thus, the support threshold values do not influence the required memory in this approach. The number of nodes in DFFP-tree with the window size and pane size variations, is depicted in Table 7. Table 7 is represented by Fig. 7.

It should be notated that the number of nodes is constant, with the variation of window size for different data sets.

5.2.3. Fuzzy frequent patterns list

After the construction of DFFP-tree, fuzzy association rules algorithm has been applied to extract exact fuzzy frequent patterns. Data sets mentioned in Table 5 and membership functions with five regions fuzzy sets (''Very Low'', ''Low'', ''Medium'', ''High'', ''Very High'') mentioned in Fig. 5 are used. Table 8 represents a sample of the final fuzzy frequent 1-itemsets and 2-itemsets with their total scale cardinality in all transactions calculated as ''count'' value for pane size = 20. The complete sets are not presented for lack of space. Predefined minimum support for ''Accidents'' data sets is selected = 5% where it is = 2.5% for ''Retail'' data sets. Hence, fuzzy association rules could be deduced as shown in Table 9.

Note that the numbers written before Linguistic terms represent weather conditions (sunny, cloudy, etc.) for ''Accidents'' data sets, where,they represent purchased items (milk, bread, eggs,...) for ''Retail'' data sets.

6. Conclusion & future work

To the authors' knowledge,there are no existing techniques in the literature, that mine fuzzy association rules from a high-speed ubiquitous data streams. In this paper, an efficient technique suitable for ubiquitous applications, and satisfying the features mentioned in Section 2.3, was proposed. Also, a novel tree structure Dynamic Fuzzy Frequent Pattern tree (DFFP-tree) that combines fuzzy frequent pattern tree with the concept of dynamic tree restructuring (i.e. deleting old panes, and inserting new ones) at runtime was introduced. The proposed FFP_USTREAM technique could be very helpful in many practical situations for managers to make more significant and flexible decisions such as.

- Determining stock required in retail applications.
- Determining methods of treatment in medical applications.
- Determining methods of precaution in road safety applications.

The following are some suggestions for possible future work.

- The current FFP_USTREAM is based on the assumption that fuzzy sets and membership function are generated in triangular and trapezoidal waveforms only. Other

waveforms such as Gaussian, bell-shaped, sigmoidal, and S-curve should also be considered depending on the applications.
- In this paper, window size & pane size were assumed constant. It is worthwhile to consider variable window and pane size.
- The proposed FFP_USTREAM handles distributed homogenous data streams. Thus, there is a need to investigate the cases where the distributed flow of data streams are heterogeneous, and the data sets are incompatible.
- The problem of mining fuzzy association rule from ubiquitous data streams can be introduced using other window techniques like tilted-window or land mark window.

Appendix A. Algorithms used in the proposed technique

Algorithm A.1. FFP_USTREAM.

Algorithm A.2. Create sliding window module.

Algorithm A.3. Construction of a dynamic fuzzy frequent pattern tree DFFP-tree.