Approximate Frequent Pattern Discovery Over Data Stream

Kittisak Kerdprasop, and Nittaya Kerdprasop

Abstract—Frequent pattern discovery over data stream is a hard problem because a continuously generated nature of stream does not allow a revisit on each data element. Furthermore, pattern discovery process must be fast to produce timely results. Based on these requirements, we propose an approximate approach to tackle the problem of discovering frequent patterns over continuous stream. Our approximation algorithm is intended to be applied to process a stream prior to the pattern discovery process. The results of approximate frequent pattern discovery have been reported in the paper.

Keywords—Frequent pattern discovery, Approximate algorithm, Data stream analysis.

I. INTRODUCTION

DATA stream is defined as massive amounts of data continuously generated at a rapid rate, possibly time-varying and unpredictable [1], [4], [10]. Major characteristics of data streams are the continuously online arrival of data elements, uncontrolled order of such elements upon arrival, variable sizes, and a one-time processing of an element before it is discarded or archived due to the massive size of data that far exceeds the storage capacity. The requirements of timely analysis and efficient memory usage constrain most data stream mining algorithms to sacrifice accuracy of the analysis results for the fast and feasible processing.

Development of approximation algorithms [6], [13] is a direct solution to the problem of data stream analysis. However, the large volumes of data continuously arriving in a stream could eventually make the algorithms inefficient. A more practical solution is to apply a data reduction technique along with the approximation algorithms. Data summarization techniques, such as wavelet analysis [11] and histogram [4], have been proposed as synopsis data structures to provide a summary presentation of data. The issue of dynamic space allocation as the underlying data distribution changes over time is a fundamental problem of these approaches.

Data stream analysis by choosing a subset of the incoming stream is another class of techniques for producing approximate results. Sampling is a statistical-based technique widely used to scale up the algorithms [8]. Nevertheless, in the context of data stream in which the data size is unknown, simply applying a sampling method cannot give reliable approximation.

We, therefore, propose a novel approximation method to draw representatives from data stream. To produce a good approximation to the true value or quantity of underlying stream, we apply the expectation-maximization technique to get a good guess of data characteristics. Our algorithm has been designed to produce data elements from which the approximate analysis is close to the exact one. We then perform frequent pattern discovery over the sample data. Frequent pattern analyses on several data sets to verify the reliability of the method have been conducted.

The paper is organized as follows. Section 2 presents the theoretical background of our method. Section 3 is the proposed algorithm. Section 4 presents some of the experimental results from frequent pattern analyses over the reduced data stream. We conclude in Section 5 with a discussion for future work.

II. DATA STREAM DENSITY ESTIMATION

When the number of data is overwhelming and the exact data distribution is unknown, the characteristics of stream have to be estimated before data sampling can be performed. We concentrate on the sampling problem because the efficiency of frequent pattern discovery depends largely on the ability to draw samples effectively.

For a particular domain of stream data, we consider the rejection sampling method. Rejection sampling, or acceptance-rejection sampling, is a sampling method first introduce by Von Neumann [15]. This method is used in cases where a target distribution, \( f(x) \), is too complicate for us to sample from it directly.

Suppose we have a simpler distribution, \( g(x) \), which we can evaluate and generate samples from, then the difficult sampling problem can be avoided by sampling from \( g(x) \) instead. By generating a uniform random variable \( u \) from the interval \([0,1]\), we accept \( x \) if the condition \( u \leq f(x) / Cg(x) \) holds; otherwise reject the value of \( x \) and repeat the sampling
step. Posing the restriction \( C_g(x) \geq f(x) \) for some \( C > 1 \), we say that \( C_g \) envelopes \( f \). The validation of this method is the envelope principle. When simulating the point \((x, \nu)\) where \( \nu = u^* C_g(x) \), we produce a uniform simulation over the subgraph of \( C_g(x) \). Accepting only points such that \( u \leq f(x) / C_g(x) \) then produces points \((x, \nu)\) uniformly distributed over the subgraph of \( f(x) \) and thus, marginally, a simulation from \( f(x) \).

Rejection sampling will work best if \( g \) is a good approximation to \( f \). However, in a high-dimensional problem the value of \( C \) needs to be chosen very large to ensure the requirement \( C_g(x) > f(x) \), for all \( x \). The result is an enormous rejection rate.

The difficulty of applying rejection sampling method directly to the problem of data stream analysis is that we do not know beforehand where the modes of \( f \) are located or how high they are. In other words, we do not know the exact characteristics of the target density. We thus propose to apply the Expectation-Maximization (EM) technique \[7\], \[12\], \[14\] to approximate the density \( f(x) \).

We consider multi-dimensional stream data as mixtures of Gaussian, or normal, probability density functions (pdf). Gaussian mixtures \[9\], \[12\] are combinations of Gaussian \((\mu, \Sigma)\) densities written as:

\[
g(x) = \sum_{i=1}^{K} p_i f(x \mid \theta_i) \tag{1}
\]

A random variable \( x \) denotes independent observation in \( K \) mixture components. The \( p_i \)’s are the mixing proportions, \( 0 < p_i < 1 \) for all \( i = 1, \ldots, K \), and \( p_1 + \ldots + p_K = 1 \). The \( f(x \mid \theta) \) denotes the density of a \( d \)-dimensional Gaussian distribution with mean vector \( \mu \) and covariance matrix \( \Sigma \), that is \( \theta = (\mu, \Sigma) \), and the Gaussian pdf is given by \[5\], \[14\]:

\[
g_{(\mu, \Sigma)}(x) = \frac{1}{(2\pi)^{d/2}\sqrt{\det(\Sigma)}} \exp\left\{ -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right\} \tag{2}
\]

By varying the number of Gaussians \( K \), the mixing proportions \( p_i \), and the parameter \( \theta \), of each Gaussian density function, Gaussian mixtures can be used to describe any complex pdfs.

In stream data a mixture density \( p_i f(x \mid \theta) \) has been observed with unknown parameters \( \theta \), and \( p_i \). To find these parameters to optimally fit a mixture model for a given set of data, the EM algorithm \[7\], \[12\], \[14\] can be used. The EM algorithm is a broadly applicable approach to the iterative computation of maximum likelihood estimates. For a set of \( iid \) samples \( X = \{ x_1, \ldots, x_N \} \) drawn from a data generation model \( f_{(\mu, \Sigma)}(x) \), thus the resulting density for the samples is:

\[
L(\theta \mid X) = \prod_{i=1}^{N} f_{(\mu, \Sigma)}(x_i) = L(\theta \mid x) \tag{3}
\]

The likelihood function \( L(\theta \mid x) \) is the likelihood of the parameters given to find \( \theta \) that maximizes \( L \), that is:

\[
\arg \max_{\theta} L(\theta \mid X). \quad \text{(In the Gaussian case, the computation of the exponential can be avoided by maximizing } \log \left( L(\theta \mid x) \right) \text{ instead of } L(\theta \mid x) \).
\]

The EM algorithm is an approach to find the maximum of likelihood functions in incomplete data problems. Let \( X \) be observed data, \( Z \) be unobserved data, and \( Y = X \cup Z \) be full data set. The probability distribution of \( Z \) depends on \( X \) and the unknown parameter \( \theta \). Given an initial parameter \( \theta^{(0)} \), The EM algorithm produces a sequence \( \{\theta^{(0)}, \theta^{(1)}, \theta^{(2)}, \ldots \} \) that converges to a stationary point of the likelihood function.

### III. A NOVEL ALGORITHM ON STREAM SAMPLING

For the problem of frequent pattern discovery over data stream, we assume that the observed data distribute normally. The central idea of our approach is the bounded estimation of stream data characteristics. Given a specific number of models, the EM method is applied to estimate the mean value of each model. Then these means are scaled up to get an upper bound \( E' \) for the underlying partially observed target density. The proposed idea can be graphically displayed as in Fig. 1.
we scale up this distribution to obtain an approximate $E'$, which is a simpler distribution that we can evaluate and generate samples from. The outline of our approximate sampling algorithm is illustrated in Fig. 2.

**Input:** a $d$-dimensional data set $D$ with $N$ points an integer $K$ to specify the number of models a sample size $SS$ 

**Output:** a sample set $S$ drawn from the mixture models

---

**Algorithm:**

1. If $d > 0$ then
   - Apply PCA to obtain 1st and 2nd components
2. Transform $D$ to a two-dimensional data set $X$
3. Set $\text{max\_iteration} = \text{max}(50, d \ast K)$
4. Initialize parameter $\theta = (\mu, \Sigma)$ for each of $K$ Gaussian models
5. Initialize the prior probabilities $P(m_k)$ of each model
6. Repeat
   - Set $\text{count} = 0$
   - Get a sample set $D$
   - Transform $D$ to a two-dimensional data set $X$
   - Apply PCA to obtain 1st and 2nd components
   - Transform $D$ to a two-dimensional data set $X$
   - Repeat
     - Transform $D$ to a two-dimensional data set $X$
   - End
7. Compute the probability
   - $P(m_k | x_n, \theta^{(i)}) = \frac{P(m_k | \theta^{(i)}) \cdot P(x_n | \mu_k, \Sigma_k)}{\sum_j P(m_j | \theta^{(i)}) \cdot P(x_n | \mu_j, \Sigma_j)}$
8. Update means $\mu_k$, variances $\Sigma_k$, and priors $P$
   - $\mu_k^{(i+1)} = \frac{\sum_{n=1}^{N} x_n P(m_k | x_n, \theta^{(i)})}{\sum_{n=1}^{N} P(m_k | x_n, \theta^{(i)})}$
   - $\Sigma_k^{(i+1)} = \sum_{n=1}^{N} P(m_k | x_n, \theta^{(i)}) (x_n - \mu_k^{(i+1)}) (x_n - \mu_k^{(i+1)})^T$
   - $\sum_{n=1}^{N} P(m_k | x_n, \theta^{(i)})$
   - $P(m_k | \theta^{(i+1)}) = \frac{1}{N} \sum_{n=1}^{N} P(m_k | x_n, \theta^{(i)})$
9. Until the $\text{max\_iteration}$ has been reached or the joint likelihood of all data with respect to all the models is greater than the lower boundary criterion $CL(\theta)$
   - $L(\theta) \geq CL(\theta) = \sum_{k=1}^{N} \sum_{n=1}^{N} P(m_k | x_n, \theta) \log p(x_n | \theta)$
10. Get $E(\theta)$ as $\theta = (\mu_k, \Sigma_k)$ for $k = 1, ..., K$
11. Get $E'(X)$ as a rough $\theta' = (\mu_k', \Sigma_k')$ from $r$ iterations, $r < 10$

---

**Fig. 2** An algorithm to obtain approximate samples

---

**IV. EXPERIMENTATIONS**

To verify the utility of the proposed method on the real-world data we test our algorithm on four data sets: Wisconsin diagnostic breast cancer, Chess, DNA, and Audiology. These data are taken from from UC Irvine Machine Learning Database Repository (http://www.ics.uci.edu/~mlearn/MLRepository.html). The details of selected datasets are summarized in Table I. After the sampling step, the sample data are tested for accuracy and efficiency on the discovery of frequent patterns. We adopt the Apriori algorithm [2], [3] as a method to discover frequent patterns.

**TABLE I**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>File size</th>
<th># Transactions</th>
<th># Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>Breast cancer</td>
<td>21.1 KB</td>
<td>191</td>
<td>10</td>
</tr>
<tr>
<td>Chess</td>
<td>237 KB</td>
<td>2,130</td>
<td>37</td>
</tr>
<tr>
<td>DNA</td>
<td>252 KB</td>
<td>2,000</td>
<td>61</td>
</tr>
<tr>
<td>Audiology</td>
<td>41.1 KB</td>
<td>150</td>
<td>70</td>
</tr>
</tbody>
</table>

We comparatively study the performance of frequent pattern discovery on the whole dataset (and call it an exact analysis) against the sample data prepared as explained in the algorithm (Fig. 2). We compare the speed of discovery process, including sampling time for an approximate analysis, as well as the accuracy of patterns found. All experimentations have been performed on a 796 MHz AMD Athlon notebook with 512 MB RAM and 40 GB HD.

The comparison results of accuracy and run time are shown in Figs. 3 and 4, respectively. An accuracy comparison has been done on the basis of number of frequent patterns discovered on each value of minimum support. The accuracy obtained from sampled data is almost as good as the accuracy of pattern discovery from the original dataset. It is also worth noticing that the lower minimum support value, the greater the number of patterns found.

For the experimental results on speed comparison, our proposed sampling method can actually help reducing processing time of frequent pattern discovery. This gain is quite obvious in the case of low support value (minimum support threshold is less than 10%).

---

**V. CONCLUSION AND DISCUSSION**

Frequent pattern discovery over data stream is a challenge problem due to the limitation on time and space. We propose to tackle the problem by means of sampling. Instead of applying simple random sampling, we argue that blindly taking sample from the stream in which we do not know the size of data in advance is incorrect. We, thus, propose a better solution by introducing the concept of guessing an upper bound $E'$ and lower bound $E$ of stream distribution. The distance of $E$ and $E'$ at each sampling point is a decision criteria for either sample acceptance or rejection. A narrow distance among the two estimated densities tends to the acceptance case if the distance ratio is greater than the generated uniform random variable from the interval $[0, 1]$. 

---
Fig. 3 an accuracy comparison of exact and approximate frequent pattern discovery

Fig. 4 a run-time comparison of exact and approximate frequent pattern discovery
The proposed idea of rejection sampling from the bounded density functions is intended to be a data preparation step prior to the frequent pattern discovery process. The experimental results confirm the accuracy and efficiency of our proposed method. We plan to investigate the problem of frequent pattern discovery from stream data further on the issues of data estimation. That is, we are interested in skewed data in which distributions are not uniformly distributed.

REFERENCES


Kittisak Kerdprasop is an associate professor at the school of computer engineering, Suranaree University of Technology, Thailand. He received his bachelor degree in Mathematics from Srinakarinwirot University, Thailand, in 1986, master degree in computer science from the Prince of Songkla University, Thailand, in 1991 and doctoral degree in computer science from Nova Southeastern University, USA, in 1999. His current research includes Data mining, Artificial Intelligence, Functional Programming, Computational Statistics.

Nittaya Kerdprasop is an associate professor at the school of computer engineering, Suranaree University of Technology, Thailand. She received her B.S. from Mahidol University, Thailand, in 1985, M.S. in computer science from the Prince of Songkla University, Thailand, in 1991 and Ph.D. in computer science from Nova Southeastern University, USA, in 1999. She is a member of ACM and IEEE Computer Society. Her research of interest includes Knowledge Discovery in Databases, AI, Logic Programming, Deductive and Active Databases.