International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:11, 2007

# 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

Manuscript received October 15, 2007. This work was supported in part by the Thailand Research Fund under grant RMU-5080026 and research fund from the National Research Council of Thailand. DEKD research unit is fully supported by Suranaree University of Technology.

Kittisak Kerdprasop is a director of the Data Engineering and Knowledge Discovery (DEKD) research unit, School of Computer Engineering, Suranaree University of Technology, 111 University Avenue, Muang District, Nakhon Ratchasima 30000, Thailand (phone: +66-44-224349; fax: +66-44-224602; e-mail: kerdpras@ sut.ac.th, KittisakThailand@gmail.com).

Nittaya Kerdprasop is a principal researcher of DEKD research unit and an associate professor at the School of Computer Engineering, Suranaree University of Technology, 111 University Ave., Nakhon Ratchasima 30000, Thailand (e-mail: nittaya@sut.ac.th, nittaya.k@gmail.com ).

step. Posing the restriction $Cg(x) \geq f(x)$ for some $C > 1$, we say that $Cg$ envelopes $f$. The validation of this method is the envelope principle. When simulating the point $(x, v)$ where $v = u*Cg(x)$, we produce a uniform simulation over the subgraph of $Cg(x)$. Accepting only points such that $u \leq f(x) / Cg(x)$ then produces points $(x, v)$ 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 $Cg(x) > f(x)$, 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 distributions written as:

$$g(x) = \sum_{i=1}^{K} p_i f(x \mid \theta_i) \quad (1)$$

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 $i = 1, ..., K$, and $p_1 + ... + p_K = 1$. The $f(x|\theta_i)$ 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]:

$$g_{(\mu,\Sigma)}(x) = \frac{1}{(2\pi)^{d/2}\sqrt{\det(\Sigma)}} \exp\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\} \quad (2)$$

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

In stream data a mixture density $p_i f(x|\theta_i)$ has been observed with unknown parameters $\theta_i$ 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 $X = \{ x_1, ..., x_N \}$ drawn from a data generation model $f_{(\mu,\Sigma)}(x_i)$, thus the resulting density for the samples is:

$$\prod_{i=1}^{N} f_{(\mu,\Sigma)}(x_i) = L(\theta \mid x) \quad (3)$$

The likelihood function $L(\theta \mid x)$ is the likelihood of the parameters given the data. In the maximum likelihood problem, the goal is to find $\theta$ that maximizes $L$, that is $\arg\max_\theta L(\theta \mid X)$. In the Gaussian case, the computation of the exponential can be avoided by maximizing $log\ (L(\theta \mid x))$ 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∪Z be full data set. The probability distribution of Z depends on X and the unknown parameter θ. Given an initial parameter $\theta^{(0)}$, The EM algorithm produces a sequence $\{\theta^{(0)}, \theta^{(1)}, \theta^{(2)}, ... \}$ 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.

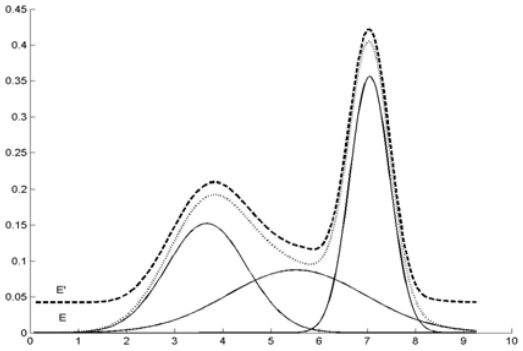



Fig. 1 Rejection sampling with an upper bound E'

The target function is represented as a one-dimensional 3-Gaussian mixtures (the three solid lines at the bottom of Fig. 1) from which we want to draw samples. The density $E(x)$ is estimated with the upper bound requirement that $E(x) > f(x)$ for all $x$. $E'(x)$ is the approximation (shown as a thick dash line in Fig. 1) of the unknown target density. A broad distance of $E$ and $E'$ (e.g., at $x = 1$) represents a rejecting area, whereas a narrow distance (e.g., at $x = 6.5$) is an acceptance one.

It should be noted that EM requires a pre-specified number of $K$ components to be incorporated into the mixture models. According to our proposed method, a suitable number should be selected by a user. To cope with multi-dimensional problem, we propose to use a statistical method – principal component analysis (PCA) – to reduce the complicated problem to a simpler two-dimensional problem. That is, we take into account only the first and second major components of the data set. The two-dimensional data are used to train the EM algorithm to estimate parameters μ and Σ of the Gaussian mixture models. The estimated Gaussian pdf is a distribution E (as shown in Fig. 1). To sample from the estimated density

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

*// Data preprocessing steps //*
1. If $d > 0$ then
Apply PCA to obtain 1st and 2nd components
2. Transform *D* to a two-dimensional data set *X*

*// Density estimation with EM to get a rough pdf $E'(X)$ //*
3. Set max_iteration = max{50, $d * K$}
4. Initialize parameter $\theta = (\mu, \Sigma)$ for each of *K* Gaussian models
5. Initialize the prior probabilities $P(m_k)$ of each model *m* to $1/K$, $k = 1,..., K$
6. Repeat
7. Compute the probability

$$P(m_k^{(i)} \mid x_n, \theta^{(i)}) = \frac{P(m_k^{(i)} \mid \theta^{(i)}) \bullet p(x_n \mid \mu_k^{(i)}, \Sigma_k^{(i)})}{\sum_j P(m_j^{(i)} \mid \theta^{(i)}) \bullet p(x_n \mid \mu_j^{(i)}, \Sigma_j^{(i)})}$$

8. Update means $\mu_k$, variances $\Sigma_k$, and priors *P*

$$\mu_k^{(i+1)} = \frac{\sum_{n=1}^{N} x_n P(m_k^{(i)} \mid x_n, \theta^{(i)})}{\sum_{n=1}^{N} P(m_k^{(i)} \mid x_n, \theta^{(i)})}$$

$$\Sigma_k^{(i+1)} = \frac{\sum_{n=1}^{N} P(m_k^{(i)} \mid x_n, \theta^{(i)}) (x_n - \mu_k^{(i+1)})(x_n - \mu_k^{(i+1)})^T}{\sum_{n=1}^{N} P(m_k^{(i)} \mid x_n, \theta^{(i)})}$$

$$P(m_k^{(i+1)} \mid \theta^{(i+1)}) = \frac{1}{N} \sum_{n=1}^{N} P(m_k^{(i)} \mid x_n, \theta^{(i)})$$

9. Until the *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 $CL(\theta)$

$$L(\theta) \geq CL(\theta) = \sum_{k=1}^{K} \sum_{n=1}^{N} P(m_k \mid x_n, \theta) \log p(x_n \mid \theta)$$

10. Get $E(X)$ as $\theta_i = (\mu_k, \Sigma_k)$ for $k = 1, ..., K$,
11. Get $E'(X)$ as a rough $\theta_i' = (\mu^r{}_k, \Sigma^r{}_k)$ from *r* iterations, $r < 10$

*// Sampling steps //*
12. Set count = 0
13. While count < *SS*
14. Sample *x* from $E(X)$
15. Generate *u* from U(0,1)
16. If $u \leq E(x)/(\sqrt{d} * E'(x))$
then Accept *x*, add it to *S*, and increment count
17. Return *S*

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
Dataset Characteristics

| Dataset | File size | # Transactions | # Items |
|---|---|---|---|
| Breast cancer | 21.1 KB | 191 | 10 |
| Chess | 237 KB | 2130 | 37 |
| DNA | 252 KB | 2,000 | 61 |
| Audiology | 41.1 KB | 150 | 70 |

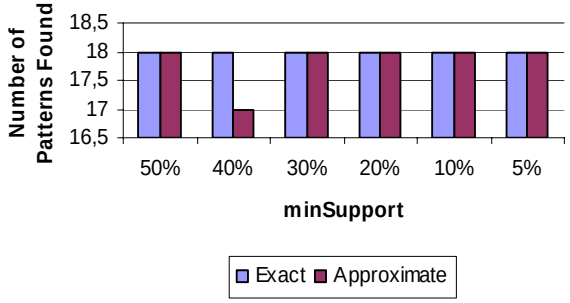
We comparatively study the performance of frequent pattern discovery on the whole dataset (and call it an exact analysis) against the sample data prepared as explained in the algorithm (Fig. 2). We compare the speed of discovery process, including sampling time for an approximate analysis, as well as 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.

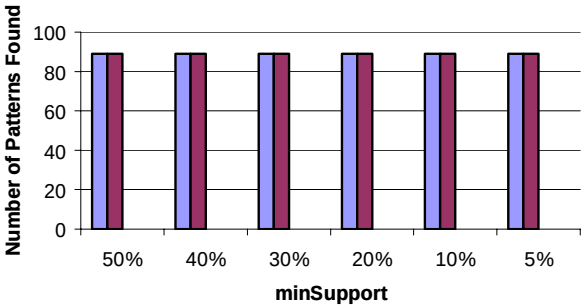
For the experimental results on speed comparison, our proposed sampling method can actually help reducing processing time of frequent pattern discovery. This gain is quite obvious in the case of low support value (minimum support threshold is less than 10%).

## V. Conclusion and Discussion

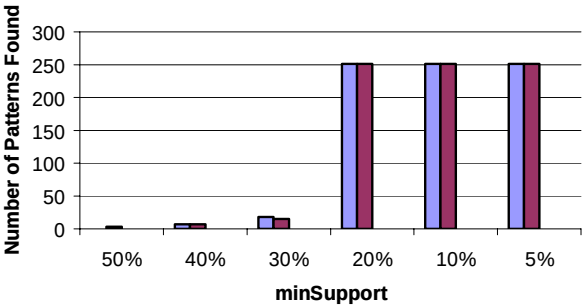
Frequent pattern discovery over data stream is a challenge problem due to the limitation on time and space. We propose to tackle the problem by means of sampling. Instead of applying simple random sampling, we argue that blindly taking sample from the stream in which we do not know the size of data in 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].



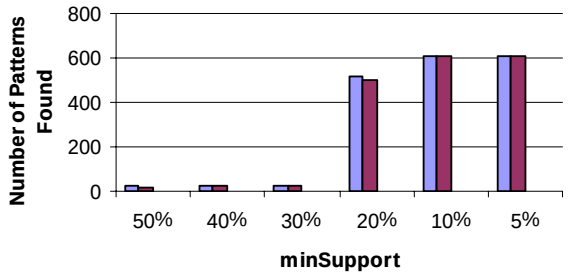

(a) Breast cancer data




(b) Chess data

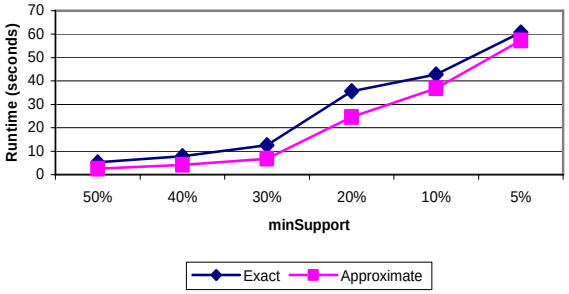



(c) DNA data

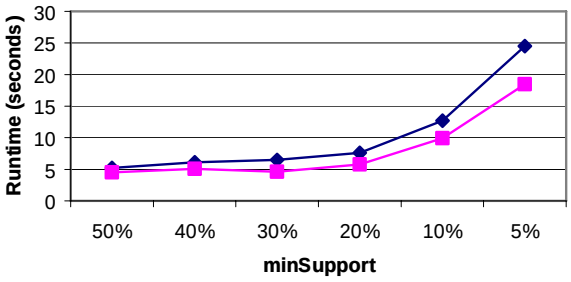



(d) Audiology data

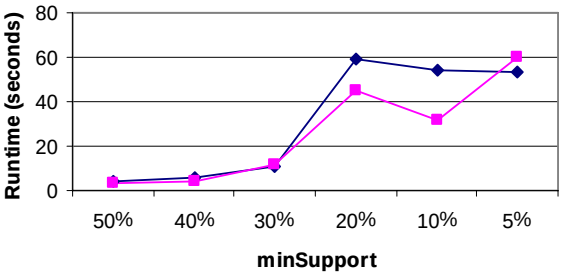
Fig. 3 an accuracy comparison of exact and approximate frequent pattern discovery



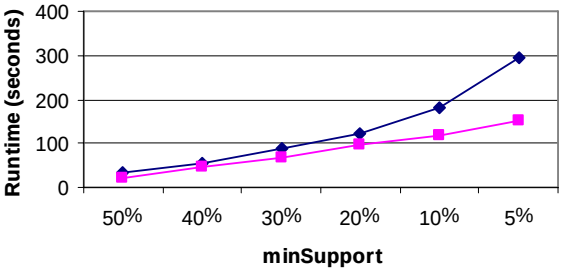

(a) Breast cancer data




(b) Chess data




(c) DNA data




(d) Audiology data

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 interest in skewed data in which distributions are not uniformly distributed.

## REFERENCES

[1] C. Aggarwal, J. Han, J. Wang, and P. Yu, "A framework for clustering evolving data streams," in *Pro. Very Large Data Bases*, 2003.

[2] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, 1993, pp. 207–216.

[3] R. Agrawal and R. Srikant, "Fast algorithm for mining association rules," in *Proc. Int. Conf. Very Large Data Bases*, 1994, pp. 487–499.

[4] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom, "Model and issues in data stream systems," in *Pro. ACM PODS*, 2002.

[5] J. Bilmes, "A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models," Dept. Electrical Engineering and Computer Science, University of California Berkeley, Technical Report TR-97-021, 1998.

[6] G. Coremode and S. Muthukrishnan, "What's hot and what's not: Tracking most frequent items dynamically," in *Pro. ACM PODS*, 2003.

[7] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society B*, vol. 39, pp. 1-22, 1977.

[8] P. Domingos and G. Hulten, "A general method to scaling up machine learning algorithms and its application to clustering," in *Pro. ICML*, 2001.

[9] M. A. T. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, pp. 381–396, 2002.

[10] M. Gaber, A. Zaslavsky, and S. Krishnaswamy, "Mining data stream: A review," *SIGMOD Record*, vol. 34, pp. 18–26, 2005.

[11] A. Gilbert, Y. Kotidis, S. Muthukrishnan, and M. Strauss, "One-pass wavelet decompositions of data streams," *IEEE Trans. Knowledge and Data Engineering*, vol. 15, pp. 541–554, 2003.

[12] J. M. Marin, K. Mengersen, and C. Robert, "Bayesian modelling and inference on mixtures of distributions," in *Handbook of Statistics*, vol. 25, Elsevier-Science, 2005.

[13] S. Muthukrishnan, "Data streams: Algorithms and applications," in *Proc. ACM-SIAM Symposium on Discrete Algorithm*, 2003.

[14] B. Resch, "A tutorial for the course computational intelligence," Available: http://www.igi.tugraz.at/lehre/CI

[15] J. von Neumann, "Various techniques used in connection with random digits," *Applied Mathematics Series*, vol. 12, National Bureau of Standards, Washington, D.C., 1951.

**Kittisak Kerdprasop** is an associate professor at the school of computer engineering, and a director of DEKD research unit, 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.