Poster Paper

An On-Line Approximation Algorithm for Mining Frequent Closed Itemsets Based on Incremental Intersection

Koji Iwanuma
University of Yamanashi
4-4-11 Takeda, Kofu-shi
Yamanashi, Japan
iwanuma@yamananashi.ac.jp

Yoshitaka Yamamoto
University of Yamanashi
4-4-11 Takeda, Kofu-shi
Yamanashi, Japan
yyamamoto@yamanashi.ac.jp

Shoshi Fukuda
University of Yamanashi
4-4-11 Takeda, Kofu-shi
Yamanashi, Japan

ABSTRACT

We propose a new on-line ϵ-approximation algorithm for mining closed itemsets from a transactional data stream, which is also based on the incremental/cumulative intersection principle. The proposed algorithm, called LC-CloStream, is constructed by integrating CloStream algorithm and Lossy Counting algorithm. We investigate some behaviors of the LC-CloStream algorithm. Firstly we show the incompleteness and the semi-completeness for mining all frequent closed itemsets in a stream. Next, we give the completeness of ϵ-approximation for extracting frequent itemsets.

Keywords

On-line algorithm, approximation, closed itemset, intersection, completeness

1. INTRODUCTION

Intersecting transactions in a data set is an alternative characterization of closed itemsets [1, 3, 4], which naturally leads to an incremental/cumulative computation of closed itemsets in a transaction data stream. CloStream [6] is an exact-computing on-line mining algorithm, which is a direct implementation of the incremental intersecting approach. Such an incremental intersection approach, however, has great difficulties, in practice, for quitting or breaking intersections in early stages, because it is difficult to predict in advance that current intersection operations never produce any frequent closed itemsets[1].

In this paper, we propose a new on-line ϵ-approximation algorithm for mining closed itemsets from a stream, which is also based on the incremental/cumulative intersection principle. The proposed algorithm, called LC-CloStream, is constructed by integrating CloStream [6] algorithm and Lossy Counting algorithm [2]. LC-CloStream succeeded in overcoming the above difficulties using ϵ-approximation [2, 5].

We study fundamental properties of LC-CloStream algorithm. Firstly we show the incompleteness and the semi-completeness for mining all frequent closed itemsets in a stream. Next, we give the completeness of ϵ-approximation for extracting frequent itemsets from a transaction streams.

2. PRELIMINARIES

Let $I = \{e_1, e_2, \ldots, e_r\}$ be a set of items. A non-empty subset A of I is called an *itemset* (or *transaction*). A *transaction stream* of length N is a sequence of N transactions $\langle A_1, A_2, \ldots A_N\rangle$. In this paper, we denote items as a, b, c, $\ldots$, and itemsets as A, B, C, $\ldots$. We also abbreviate an itemset $\{e_1, e_2, \ldots, e_m\}$ as $e_1 e_2 \cdots e_m$, for simplicity.

Let $\mathcal{S}$ be a stream $\langle A_1, \ldots, A_N\rangle$ and B be an itemset. We define a multiset $\mathcal{K}(B,t)$ at time t $(1 \le t \le N)$ as $\mathcal{K}(B,t) = \{A_j \in \mathcal{S} \mid B \subset A_j,\ 1 \le j \le t\}$. The *frequency* of B at time t, denoted as $\sup(B,t)$, is $|\mathcal{K}(B,t)|$. Given a minimal frequency threshold σ $(0 < \sigma < 1)$, B is *frequent* at time t in $\mathcal{S}$ if $\sup(B,t) \ge \sigma \cdot t$. An itemset B is *closed* at time t in $\mathcal{S}$ if there is no itemset C such that $B \ne C$ and $B \subset C$ and $\sup(B,t) = \sup(C,t)$.

The following recursive relation makes it possible to incrementally compute closed itemsets in a stream $\mathcal{S}$. Let $\mathcal{CIS}(\mathcal{S})$ be a set of all closed itemsets in $\mathcal{S}$ and $\circ$ be a well-known concatenation operator of two sequences.

PROPOSITION 1 ([1, 3]). *Let $\mathcal{S}$ be a stream $\langle A_1, \ldots A_N\rangle$. We have:*

$$\begin{aligned} \mathcal{CIS}(\langle A_1\rangle) &= \{A_1\} \\ \mathcal{CIS}(\mathcal{S}_k) \circ \langle A_{k+1}\rangle) &= \mathcal{CIS}(\mathcal{S}_k) \cup \{A_{k+1}\} \cup \\ &\quad \{B \mid \exists C \in \mathcal{CIS}(\mathcal{S}_k) : B = C \cap A_{k+1}\}, \end{aligned}$$

where $\mathcal{S}_k$ is the k element prefix of $\mathcal{S}$, i.e., $\langle A_1, \ldots A_k\rangle$.

CloStream [6] is an on-line exact counting algorithm for mining closed itemsets in a stream, which uses the above recursive relation in a straightforward way, and thus cannot avoid a combinatorial explosion problem caused by $\mathcal{CIS}(\mathcal{S})$.

3. LC-CLOSTREAM

The LC-CloStream algorithm maintains an internal *frequency table* TS. Formally, TS is a set of tuples $\langle B, f(B), \delta(B)\rangle$, where B is an itemset, $f(B)$ is the number of occurrences of B after the time t_B when B was lastly stored in TS, and $\delta(B)$ is the maximal error count at time t_B. We write the frequency table TS at time t as $TS(t)$, and similarly for $f(B,t)$ and $\delta(B,t)$. Let $\mathcal{SP}(B,t)$ denote the set of supersets of B belonging to the frequency table $TS(t)$, that is, $\mathcal{SP}(B,t) = \{C \in TS(t) \mid B \subset C\}$. We define $\max\mathcal{SP}(B,t)$ as follows:

$$\max\mathcal{SP}(B,t) = \operatorname*{argmax}_{C \in \mathcal{SP}(B,t)} (f(C,t) + \delta(C,t))$$

The former part of LC-CloStream algorithm, i.e., in lines 5 to 18, performs the incremental intersection and the latter

Algorithm 1 LC-CloStream algorithm

Input: a stream $\mathcal{S} = \langle A_1, A_2, \ldots, A_N \rangle$,
a relative minimal frequency threshold σ $(0 < \sigma < 1)$,
a maximal permissible error ratio ϵ $(0 < \epsilon < \sigma)$.
Output: a family $\mathcal{FCS}$ of frequent closed item sets in $\mathcal{S}$
1: $t \leftarrow 1$ ▷ t is a current time
2: Initialize the frequency table TS.
3: **while** $t \leq N$ **do**
4: Read A_t.
5: **for** each $B \in TS$ **do**
6: $C \leftarrow B \cap A_t$
7: **if** $C \neq \emptyset$ **then** ▷ i.e. the case of $\mathcal{SP}(C, t) \neq \emptyset$
8: $D \leftarrow \max\mathcal{SP}(C)$
9: **if** $C \notin TS$ **then** ▷ register C as a new entry
10: $TS \leftarrow TS \cup \{\, \langle C,\ f(D) + 1,\ \delta(D) \rangle \}$
11: **else** ▷ increase the frequency vale of C
12: $TS \leftarrow (TS - \{\, \langle C,\ f(C),\ \delta(C) \rangle \})$
$\cup \{\, \langle C,\ f(D) + 1,\ \delta(D) \rangle \}$
13: **end if**
14: **end if**
15: **end for**
16: **if** $A_t \notin TS$ **then** ▷ register A_t as a new entry
17: $TS \leftarrow TS \cup \{\, \langle A_t,\ 1,\ \epsilon \cdot (t-1) \rangle \,\}$
18: **end if**
19: **for** each $B \in TS$ **do** ▷ ϵ-elimination
20: **if** $f(B) + \delta(B) \leq \epsilon \cdot t$ **then**
21: $TS \leftarrow TS - \{\, \langle B,\ f(B),\ \delta(B) \rangle \}$
22: **end if**
23: **end for**
24: **end while**
25: **return** $\mathcal{FCS}(N) = \{\ B \in TS \mid f(B) + \delta(B) \geq \sigma \cdot N \}$

part in lines 19 to 23 executes the ϵ-elimination operation, which involves an ϵ-approximation computation.

Notice that Algorithm 1 is described declaratively for simplicity, thus has the time complexity $O(k^2)$ where k is the total number of entries in TS, while [6] gave an optimized procedural form of the complexity $O(k)$.

Unfortunately, LC-CloStream algorithm has a counterexample for the completeness, as shown in Example 1. We can, however, give the semi-completeness for LC-CloStream.

Example 1. Let $\mathcal{S}_1$ be a stream $\langle a, b, b, b, b, b, ac, ac, ac \rangle$ of length 9. We suppose $\sigma = 0.3$ and $\epsilon = 0.2$. Then, the frequent closed itemsets in $\mathcal{S}_1$ are three itemsets a, b, ac. At time $t = 1$, LC-CloStream algorithm processes the first transaction a and store the set a, as a new closed itemset, into the frequency table TS. At time $t = 2$, LC-CloStream adds the set b into TS, and so on. At time $t = 6$, LC-CloStream firstly increase the frequency counter $f(B)$ in TS, then the table TS becomes to $\{\langle a, 1, 0 \rangle,\ \langle b, 5, 1 \rangle\}$ at this point. Next LC-CloStream performs the ϵ-elimination rule to TS, and delete the tuple of the closed set a since $f(a, 6) + \delta(a, 6) = 1 < 1.2 = \epsilon \cdot 6$ holds. At time $t = 7$, LC-CloStream registers the set ac to TS as a new closed set, but cannot increase the frequency counter of the set a, because TS has the tuple of a no longer. Thus, LC-CloStream eventually returns the set $\mathcal{FCS}(9) = \{b, ac\}$ and fails to produce the frequent closed itemset a.

Next, we show a semi-completeness theorem which partially overcomes the deficit shown above in LC-CloStream. Furthermore, we give completeness theorem of LC-CloStream for frequent itemsets mining based on ϵ-approximation.

Definition 1. Let S be a stream of length N, B be a closed itemset and ϵ be a maximal error ratio. We say, B is *ϵ-extendable* on $\mathcal{S}$ if there is a closed itemset C such that $B \subset C$, $B \neq C$ and $\sup(B) - \sup(C) \leq \epsilon N$

THEOREM 1 (SEMI-COMPLETNESS FOR CLOSED ITEMSETS). *Let $\mathcal{S}$ be a stream of length N and B be a frequent closed itemset in $\mathcal{S}$. If B is NOT ϵ-extendable, then $B \in \mathcal{FCS}(N)$.*

Definition 2. Let $\mathcal{S}$ be a stream of length N, σ be a minimal frequency threshold and $\mathcal{FCS}(N)$ be a output produced from $\mathcal{S}$ by LC-CloStream algorithm. Then we define $\mathcal{RS}(N)$ as follows:

$$\mathcal{RS}(N) = \mathcal{FCS}(N) \cup \{C \mid \exists B \in \mathcal{FCS}(N) : C \subset B, C \neq \emptyset\}$$

THEOREM 2 (COMPLETENESS FOR ITEMSETS). *Let $\mathcal{S}$ be a stream of length N and B be a frequent itemset in $\mathcal{S}$. Then $B \in \mathcal{RS}(N)$.*

Definition 3. Let $\mathcal{S}$ be a stream of length N and ϵ be a maximal error ratio. For any itemset B at time t $(1 \leq t \leq N)$, we define $F(B, t)$ and $\Delta(B, t)$ as follows:

1. if $\mathcal{SP}(B, t) = \emptyset$, then $F(B, t) = 0,\ \ \Delta(B, t) = \epsilon \cdot t$
2. if $\mathcal{SP}(B, t) \neq \emptyset$, then
$F(B, t) = f(\max\mathcal{SP}(B, t), t),\ \Delta(B, t) = \delta(\max\mathcal{SP}(B, t), t)$.

We call $F(B, t) + \Delta(B, t)$ the *estimated frequency* of B at time t.

Notice the estimated frequency $F(B, t) + \Delta(B, t)$ is defined based on $TS(t)$ of time t, while the counting frequency $f(B, t) + \delta(B, t)$ depends just on $TS(t-1)$ of the previous time $t - 1$.

THEOREM 3 (ϵ-APPROXIMATON OF FREQUENCY). *Let $\mathcal{S}$ be a stream of length N and ϵ be a maximal error ratio. For any itemset B, we have*

$$F(B, N) \leq sup(B, N) \leq F(B, N) + \epsilon \cdot N$$

4. CONCLUSIONS

LC-CloStream can avoid a part of combinational explosion problems in a bursty transactional data stream [5]. In the future, we will study an efficient implementation using a sophisticated data structure, and also have a plan to investigate a more advanced framework where the frequency table has a fixed constant size [5].

5. ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Number 25730133 and 25330256, and also supported by JST PRESTO (Sakigake).

6. REFERENCES

[1] C. Borgelt, X. Yang, R. Nogales-Cadenas, P. CarmonaSaez and A. Pascual-Montano: Finding Closed Frequent Item Sets by Intersecting Transactions. *Proc. EDBT 2011*, pp.367–376 (2011)

[2] G. S Manku and R. Motwani: Approximate Frequency Counts over Data Streams. *VLDM'02*, pp.346–357 (2002)

[3] T. Mielikäinen: Intersecting Data to Closed Sets with Constraints. *Proc. FIMI 2003*, CEUR WS. Proc. 90 (2003)

[4] F. Pan, G. Cong, A. Tung, J. Yang and M. Zaki: Carpenter: Finding Closed Patterns in Long Biological Datasets. *ACM SIGKDD 2003*, pp.637–642 (2003)

[5] Y. Yamamoto, K. Iwanuma, S. Fukuda: Resource-oriented Approximation for Frequent Itemset Mining from Bursty Data Streams. *ACM SIGMOD 2014*, pp.205–216 (2014).

[6] S. Yen, C. Wu, Y. Lee, V.S. Tseng, C. Hsieh: A Fast Algorithm for Mining Frequent Closed Itemsets over Stream Sliding Window. *IEEE Int'l Conf. on Fuzzy Systems*, pp.27–30, Taipei (2011).