Intl. J. Bioinformatics and Biological Sci.: (V. 7 n.1&2, p. 29-34): June & December 2019

DOI: 10.30954/2319-5169.01.2019.5

The Closed Sequence Patterns for DNA Data without Candidate Generation

S. Jawahar[1*] and P. Sumathi[2]

[1]Assistant Professor, Department of Computer Science and Applications, Sri Krishna Arts and Science College, Coimbatore-8, T.N., India
[2]Assistant Professor, PG & Research Department of Computer Science, Government Arts College, Coimbatore-18, T.N., India

*Corresponding author: shivamjawahar@gmail.com

Abstract

Sequential pattern mining is a technique which efficiently determines the frequent patterns from small datasets. The traditional sequential pattern mining algorithms can mine short-term sequences efficiently, but mining long sequence patterns are in efficient for these algorithms. The traditional mining algorithms use candidate generation method which leads to more search space and greater running time. The biological DNA sequences have long sequences with small alphabets. These biological data can be mined for finding the co-occurring biological sequence. These co-occurring sequences are important for biological data analysis and data mining. Closed sequential pattern mining is used for mining long sequences. The mined patterns have less number of closed sequences. This paper proposes an efficient Closed Sequential Pattern Mining (CSPAM) algorithm for efficiently mining closed sequential patterns. The CSPAM algorithm mines closed patterns without candidate generation. This algorithm uses two pruning methods namely, BackScan pruning, and frequent occurrence check methods. The former method prunes the search space and latter detects the closed sequential pattern in early run time. The proposed algorithm is compared with PrefixSpan algorithm, better scalability and interpretability is achieved for proposed algorithm. The experimental results are based on sample DNA datasets which outperform the other algorithms in efficiency, memory and running time.

Keywords: Sequential pattern mining(SPM), DNA, closed sequential patterns, prefixspan, CSPAM

Sequential pattern mining (SPM) is a technique which identifies the interesting complete set of sub-sequences from huge dataset [7]. The SPM is a sequence of item sets that occurs frequently in a specific order. The sequence items in the item sets are based on two factors either on time value operation or within a time gap. The ordered list is represented in large sequence, where every event is a collection of item-set occurring simultaneously. The entire timestamps associated with the events is the ordering of the list of events[8]. In customer transaction, the events are viewed together as an interesting sequence known as customer sequence. Each customer transaction is expressed as individual item set in the sequence and all the customer transactions are listed in an ordered list with respect to the transaction-time[6].

Sequential pattern mining was first introduced by R. Agrawal and R. Srikanth in[1], and it has become an essential data mining task. In past years more sequential algorithms were proposed for mining from which SPADE[3], PrefixSpan[4] and SPAM[5] was efficient. In SPADE algorithm, the breadth-first search process is used whereas in PrefixSpan and SPAM algorithms uses depth-first search method for mining process. The vertical data format method

is implemented in SPADE algorithm and mines the sequential patterns using simple join. The horizontal data format method is implemented in PrefixSpan algorithm which mines sequential patterns using the pattern growth method. The SPAM algorithm uses vertical bitmap representation method for mining sequential patterns. This algorithm outperforms PrefixSpan and SPADE algorithms in mining large datasets. However, this algorithm requires more memory space when compared to PrefixSpan and SPADE mining algorithms.

The closed sequential pattern mining was first proposed by X.Yan *et al*.[2] to overcome the limitations of sequential pattern mining algorithms. This method can mine more useful information than the sequential pattern. The closed sequential patterns can be mined in two steps,

1. To find final closed sequential patterns and ,
2. To find the closed sequential pattern set and post-prune it.

The author's contributions can be summarized as follows:

- Introduced the problem of mining closed sequential patterns in biological sequence.
- The CSPAM algorithm is proposed which mines closed sequential patterns efficiently without candidate generation.
- Proposed two pruning methods namely BackScan method and frequent occurrences method to prune search space.
- The varied minimum support threshold increased the efficiency.
- The efficiency of CSPAM algorithm is evaluated with two pruning methods which outperform in terms of memory and running time.

Related Works

Recently many sequential mining algorithms have been proposed and these approaches cover various data mining problems. In general two research problems are concerned in mining sequential patterns.

1. Improve the efficiency of the mining process[9]. This mining process mainly focuses on improving the efficiency of sequential patterns.
2. Extracting the time-related patterns using the mining process[10]. This pattern extraction method can find other time-related patterns from various databases such as weblog patterns, cyclic patterns etc for finding frequent patterns in time-related databases.

The sequential pattern mining methods are classified into two types:

1. **Apriori Algorithms**
2. **Pattern Growth Algorithms**

2.1 Apriori Algorithms

The Apriori [Agrawal and Srikant 1994] and AprioriAll [Agrawal and Srikant 1995] are the algorithms which were implemented for frequent item set mining. The apriori property is used in this algorithm and generates candidate sequences using apriori-generate join procedure. All the non-empty subsets of a frequent item set must also be frequent which belongs to a category of properties. This property is knows as ant-monotonic property (or) downward-closed property. This algorithm reduces the search space of the algorithm. It scans the data item set for generating candidate item and generates frequent item set by removing infrequent data item set. In this algorithm two steps are involved, first it joins two data item sets and in the second step, the algorithm calculates the occurrence of each candidate set and the search space is reduces by pruning the infrequent data item set.

2.2 Pattern-Growth Algorithms

The pattern growth method is the solution to the problem of generate-and-test which is on based sequential pattern mining algorithms. By using this method candidate generation step is avoided and it focuses on the search space of the database. In this

algorithm 3 steps are involved namely, building the database for mining, dividing the database search space and finally generating candidate sequences by frequent growth method.

PROBLEM STATEMENT

The closed sequential pattern mining problem is formalized and explained here.

Let A={a_1,a_2,….,a_m} be a set of all alphabets. The subset of A is called an alphabet data item set. A sequence S=(S_1,S_2, …, S_n) ($S_i \subseteq$ A) is an ordered list of sequence data item sets. The data items in each sequence item set are sorted in ordered list. The sequence data item set length is defined as the total number of the data item set in the given sequence. The sequence alphabet data item set SI_1 =(X_1,X_2,…..,X_m) is a subsequence of another sequence data item set SI_2 =(Y_1,Y2,….,Y_n), denoted as $SI_1 \subseteq SI_2$, if there exists integers $1 \leq a_1 < a_2 < \ldots < a_m \leq n$ and $X_1 \subseteq Y_{i1}$, $X_2 \subseteq Y_{i2}$, . . . , and $X_m \subseteq Y_{im}$. SI_2 represent a super-sequence of SI_1 and SI_2 contains SI_1.

A sequence database, SDB={SI_1,SI_2,…,SI_n}, is a set of sequences and each sequence has an ID. The size, |SDB|, of the sequence database SDB is the total number of sequences in the SDB. The support of a sequence X in a sequence database SDB is the no of sequences in SDB which contain X item sets.

Definition 1 (Sequential Patterns): A sequence is an ordered list of data item sets. Given a minimum support threshold min_sup, a sequence α is a sequential pattern on sequence database only if support (α) is greater than min_sup.

Definition 2 (Closed Sequential Patterns): A sequential pattern α is a closed sequential pattern if there does not exist a sequential pattern β, such that support (α) = support (β) and $\alpha \sqsubset \beta$.

The closed sequential pattern mining is used to mine the item set of closed sequential patterns which satisfies the minimum support value min_sup for a given input sequence data item set. The table 4.1 represents the sample sequence database with sequence ID and sequence data.

PROPOSED METHOD

The CSPAM algorithm is proposed for mining closed sequential patterns. This algorithm uses the depth-first search method for mining closed sequential patterns. The breadth-first search technique is mostly used but they are inefficient for mining closed sequence. The CSPAM algorithm is used to overcome the limitations of sequential pattern mining. The closed sequential patterns can be mined in two different ways namely:

1. Finding the closed sequential patterns without verifying the discovered patterns.
2. Finding closed sequential pattern and post-pruning it.

Definition: The CSPAM is used to solve the problem of mining the item set of closed sequential patterns which satisfies the minimum support threshold, min_sup for an input sequence from database SDB.

***Algorithm*:** Closed Sequential Pattern Mining (CSPAM)

Input:

(a) An input Sequence database, SDB and

(b) Minimum support Threshold, min_sup.

Output:

(a) Closed sequential patterns without candidate generation.

Algorithm Steps:

- The database is scanned to eliminate empty subsets.
- The segment tree is constructed by scanning the database SDB.
- Find the frequent 1-sequence which is greater than the minimum support threshold, min_sup using theorem sequences <s′ and s $\supset$ s′, and the total number of items in SDBs equals to $SDB_{s'}$.
- The item-extension and sequence extension are used to find n-frequent sequence.

- The BackScan search pruning technique is applied to frequent n-sequence as prefix for pruning the search space.
- The forward directional item and backward directional item are calculated and apply occurrence check method.
- Eliminate the non-closed sequential pattern items.
- If both forward directional item and the backward directional item is NULL, then closed sequential pattern is given as output.

The flow diagram Fig. 1 represents the proposed CSPAM algorithm. The method includes various steps such as constructing segment tree, verifying the minimum threshold value, mining frequent n-sequences, applying pruning methods namely BackScan search method and occurrence check method, identifying the closed sequential patterns.

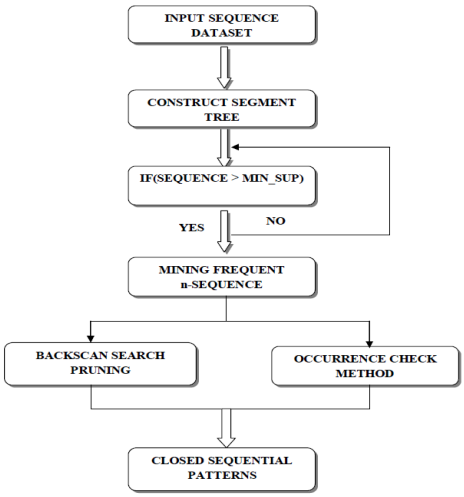

Fig. 1: CSPAM algorithm

4.1 BackScan search pruning technique

The pruning methods employed in various mining algorithms are based on the newly identified frequent patterns and some already mined closed sequential patterns or candidate set values. When compared to other pruning methods BackScan pruning method is more dynamic and more efficient. According to the given theorem, the prefix sequence can be pruned safely and this method directly mines the frequently closed sequential patterns with respect to prefix sequence.

Theorem: Let the input sequence for CSPAM be n-sequence, $S_p=S_1,S_2,\ldots..S_n$. If i (1>i) and there exists an data item set S′ which appears in each item of the i^{th} prefix S_p in sequence database SDB, the prefix S_p is safely terminated.

Proof: The item S′ appears in each sequence of the i^{th} the prefix S_p in sequence database SDB, new prefix can be obtained as $S'_p=S_1,S_2\ldots\ldots S_{i-1}S'S_i\ldots..S_n$ $(1 < i)$ or $S'_p=S'S_1S_2\ldots. S_n$ $(i = 1)$, and both $(S_pS'_p)$ and $(sup^{SDB}(S_p) = sup^{SDB}(S'_p))$. The frequent item e″ w.r.t. prefix S_p is frequent item set w.r.t. $S'_{p,}$ in the meantime $(<S_p, e'')$ $<S'_p, e''>)$ and $(sup^{SDB}(<S_p, e''>) = sup^{SDB}(<S'_{p,} e''>)$. This means that no further closed sequences can be mined with prefix S_p.

4.2 Checking Occurrence Method

The new pruning method called occurrence checking is used which detects the closed patterns in the early mining process.

Lemma 1. Occurrence checking: A sequential pattern X is not closed sequence if a frequent item Y exists such that (1) Y appears in every sequence of X's projected database and (2) the distance between X item and Y item is identical in every closed sequence of X's projected database.

Proof. If a frequent item Y appears in every sequence of X's in projected database and the distance between X and Y item is identical in every sequence of X's projected database, then it always discovers another frequent sequence containing X and Y whose support is equivalent to X's support. Therefore, X cannot be closed.

Advantages

The proposed closed sequential pattern mining (CSPAM) algorithm mines less number of closed sequences when compared to sequential pattern mining algorithms. The closed sequential mining

method is more scalable and achieves more interpretability than the sequential pattern mining.

EXPERIMENTAL RESULTS

In this section, the experimental reports of proposed CSPAM and PrefixSpan algorithms are verified on the following steps:

1. Finding the closed sequential patterns,
2. Proposed CSPAM algorithm shows better efficiency by finding lesser closed frequent patterns, and
3. CSPAM algorithm has better scalability for biological sequence databases in terms of efficiency, memory and running time.

Table 1: Sequence Database

Sequence ID	Sequence
1	CGAAC
2	TGCGA
3	CGAC
4	ACGGA

Table 1 represents sequence database example which contains sequenceID and sequence data. To evaluate the various aspects of the algorithm CSPAM an extensive performance study is performed. In the experimental results CSPAM and PrefixSpan algorithms are compared for various parameters.

Table 2: Comparison of two forms of frequent patterns

Sequence Form	Frequent Patterns	Pattern Length
Prefix Span	A:4, C:4, G:4, AA:2, AC:3, CA:4, CC:3, CG:3, GA:4, GC:3, CAC:2, CGA:3, CGC:2, GAC:2, CGAC:2	15
CSPAM	T:1, AA:2, AC:3, CA:4, CC:4, GA:4, GC:3, CGA:3, CGAC:2	9

The above experiments were conducted on a machine with Intel Core i3 2.0 Ghz CPU, 4GB memory and Windows 7 system implemented in net beans IDE 8.2. In the experiment we compared PrefixSpan and CSPAM algorithms for given sequence database. The number of patterns in PrefixSpan is 15 in length whereas CSPAM mines 9 closed frequent patterns for the given dataset. The frequent patterns and closed sequential patterns are shown in the table 2. The pattern length for CSPAM is lesser than PrefixSpan algorithm which also decreases the memory space and increases the efficiency.

CONCLUSION

The problem of mining closed sequential pattern in the biological sequence is introduced and studied in detail with different experimental results. An efficient algorithm named CSPAM is implemented for closed sequences. The CSPAM algorithm has following features:

1. It mines closed sequential patterns without candidate generation which greatly reduces the search space, and
2. Two pruning techniques are used which is very efficient in mining time.

The experimental study includes sequence dataset for the performance study of CSPAM algorithm. The proposed algorithm is more efficient than PrefixSpan in terms of efficiency, memory and running time which includes various pruning techniques.

REFERENCES

1. A Agrawal, R., & Srikant. 1995. Mining Sequential Pattern. *Eleventh International Conference on Data Engineering, Taipei, Taiwan,* 3-14.
2. X. Yan, J. Han, & R, Afshar. 2003. CloSpan: Mining closed sequential patterns in large databases. *Proc. SIAM Int'l Conf. Data Mining,* 166-177.
3. M. Zaki. 2001. SPADE: An efficient algorithm for mining frequent sequences. *Machine Learning, 42*, 31-60.
4. J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu, 2001. "PrefixSpan : Mining sequential patterns efficiently by prefixprojected pattern growth," *Proc. Int'l Conf. Data Engineering (ICDE '01), pp. 215-224.*

5. J. Ayres, J. Gehrke, T. Yiu, and J. Flannick, 2002. "Sequential pattern mining using a bitmap representation," *Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (SIGKDD '02), pp. 429-435.*
6. Qiankun Zhao and Sourav S. Bhowmick, "Sequential Pattern Mining: A Survey". (2003). *Technical Report, CAIS, Nanyang Technological University, Singapore, 118*.
7. R. Agrawal and R. Srikant, 1995. "Mining Sequential Patterns", *In Proceedings of the 11th International Conference on Data Engineering, pp. 3-14, Taipei, Taiwan.*
8. Salvatore Orlando, Raffaele Perego and Claudio Silvestri, A new algorithm for gap constrained sequence mining. 2004. *In Proceedings of the ACM Symposium on Applied Computing,* 540-547.
9. Manan Parikh, Bharat Chaudhari and Chetna Chand, 2013. A Comparative Study of Sequential Pattern Mining Algorithms, Volume 2, Issue 2, February 2013, *International Journal of Application or Innovation in Engineering & Management (IJAIEM).*
10. Thabet Slimani, and Amor Lazzez, Sequential Mining: Patterns And Algorithms Analysis.
11. J. Pei, J. Han, and R. Mao, CLOSET: An efficient algorithm for mining frequent closed itemsets. (may 2000). *Proc. ACM SIGMOD Workshop Research Issues in Data Mining and Knowledge Discovery (DMKD '00),* 21-30.
12. M. Zaki and C. Hsiao, CHARM: An efficient algorithm for closed itemset mining. (april 2002). *Proc. SIAM Int'l Conf. Data Mining (SDM '02),* 457-473.
13. J. Pei, J. Han, and R. Mao, "CLOSET: An efficient algorithm for mining frequent closed itemsets," *Proc. ACM SIGMOD Workshop Research Issues in Data Mining and Knowledge Discovery (DMKD '00), pp. 21-30, May 2000.*
14. J. Wang, J. Han, and J. Pei, CLOSET : Searching for the best strategies for mining frequent closed itemsets. (aug 2003). *Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (SIGKDD '03),* 236-245.
15. M. Zaki and C. Hsiao, "CHARM: An efficient algorithm for closed itemset mining," *Proc. SIAM Int'l Conf. Data Mining (SDM '02), pp. 457-473, Apr. 2002.*