VOL. 9, NO. 12, DECEMBER 2014 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences


www.arpnjournals.com

DISCOVERING SEQUENTIAL DISEASE PATTERNS IN MEDICAL DATABASES USING FREESPAN MINING AND PREFIKSPAN MINING APPROACH

Silvia Rostianingsih, Gregorius Satia Budhi and Leonita Kumalasari Theresia
Petra Christian University, Informatics Department, Siwalankerto, Surabaya, Indonesia
E-Mail: silvia@petra.ac.id

ABSTRACT

Dr. Soetomo General Hospital had computerized their system to stored inpatient's history. With lots of data to be analysis, one of the needs is a decision support system in order to anticipate the spread of the disease. Therefore the hospital need a system to provide the sequential pattern of disease. One of the sequential pattern mining algorithm is pattern growth based approach. The result is sequential pattern of disease from particular area in a time period based on inpatient's history. Input from user are time period, minimum support, province, and multi-dimensional. The system built with Java Net Beans 6.7 and Oracle 10g. This research showed that FreeSpan and PrefikSpan produce the same output. However, FreeSpan is more appropriate for dr. Soetomo General Hospital because the proccesing time is faster.

Keywords: sequential pattern mining, FreeSpan, prefikSpan.

INTRODUCTION

Dr. Soetomo General Hospital in Surabaya is a national hospital which acts like reference from other hospital. Information system of the patients is stored using Oracle Data and Application [1]. The increasing of civilization in East Java province is increased the patients with various type of disease. The hospital needs tool to monitor this occurrence in order to anticipate the spread of the disease.

Several studies have contributed to the efficient mining of sequential patterns and showed that PrefixSpan in many cases are outperforms the other apriori algorithm [2] [3] [4]. Both FreeSpan and PrefixSpan have similar characteristics which are pattern-growth-based, regular expression constraint, top-down search, and Depth First Search based approach [5] [6] [7] [8].

This research evaluate between FreeSpan and PrefixSpan sequential pattern mining to discover disease pattern from inpatient data (case study dr. Soetomo General Hospital) based on the disease and the occurrence region.

FREESPAN

Sequential pattern mining is a method to discover the relation between items in a dataset [2]. Frequent pattern-projected sequential pattern mining (FreeSpan) uses frequent item in the sequence database to project the projection database. Each projection will be projected database recursively further. Projected database size is usually smaller and easier to work with this database. Here is a FreeSpan algorithm [2]:

Scan DB, find frequent items, and sort into f_list (frequent item list).
(1) Construct a frequent item matrix by scanning DB once.
(2) Generate length-2 sequential patterns.
(3) Generate annotations on item-repeating patterns.
(4) Annotations on projected DB.
(5) Scan DB to generate item-repeating patterns and projected DB.

PREFIXSPAN

Prefix-projected sequential pattern mining (PrefixSpan) is a method to project the sequence databases with the prefix Based only on a frequent / frequent prefixes because any frequent subsequence can be found by growing a frequent prefix PrefikSpan. "PrefixSpan projects growing databases by frequent prefixes" [9]. This research is using bilevel projection calculation [10]:

(1) Scan the sequence to get length-1 item.
(2) Create triangular matrix from length-1 item.
(3) For each length-2 sequential pattern, build a-projected database and count the occurrence item, then build s-matrix.
(4) Each item is put in the end of length-2 sequential pattern.

CHARACTERISTICS OF FREESPAN AND PREFIXSPAN

Characteristics of FreeSpan and PrefixSpan are [5] [6] [7] [8]:

Pattern-growth-based use the divide-and-conquer to create frequent sequences. This algorithm reduce the search space by doing projection on the database.

Regular expression constraint has a property called growth-based anti-monotonic. The sequence must be reachable by growing from any component which matches the part of the regular expression when it satisfies the constraints first.

Top-down search to mine the subset of sequential pattern by constructing the corresponding set of projected databases and mining each recursively from top to bottom.

Depth First Search based approach can very quickly reach large frequent arrangements and therefore, some expansions in the other paths in the tree can be avoided.

DESIGN

FreeSpan method and PrefixSpan method required the same input which are date and time period, minimum support, province, and multi-dimensional setting. Time period is used to give sequence number. While minimum support is used to filter sequence number apperance. Every data from International Statistical Classification of Diseases and Related Health Problems (ICD) is converted into numbered code. Sequence number is calculated based on time period. Sequence number is used to calculate frequent item matrix based on minimum support. Sequence number is given by patient's ID, check-in date, and, check-out date. Patient with same ID is not having the same sequence number because sequence number based on time between check-out date and the next check-in date. When the next time difference between check-out date and the next check-in date is greater than the time interval, then its sequence number changed. Otherwise, its sequence number is not changed.

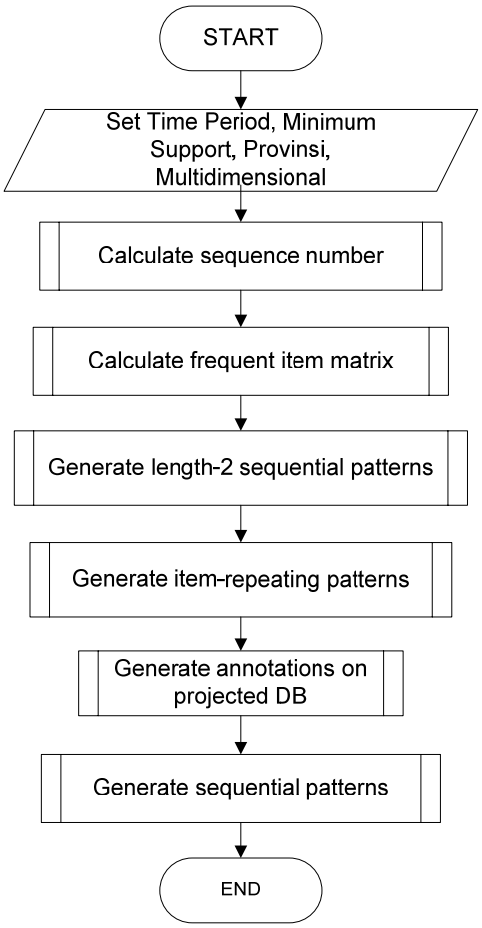

Figure-1. Flowchart of FreeSpan.

Figure-1 showed flowchart of FreeSpan. Frequent item obtained by counting the occurence The process calculate the number of occurance from the sequence pattern for each sequence and filter by minimum support. Generate a length-2 sequential pattern from the frequent item matrix. If the value meets the minimum support, then it becomes length-2 sequential patterns. The results was generated to get annotations on item-repeating patterns and annotations on projected databases. The last process was generate sequential patterns.

Figure-2 showed flowchart of PrefixSpan. Sequence pattern for PrefixSpan obtained by taking an existing item in each sequence. The items were taken from the same sequence there should not be a twin. If the item appears in the same sequence repeatly, then the item is written only once for the sequence.

The process calculate the number of occurance from the sequence pattern for each sequence. Not all items that appear to be frequent length-1 list items, only the items that meet the minimum support are becoming frequent item list length-1. This frequent length-1 is sorted from large to small based on its frequency of occurrence.

Triangular Matrix obtained by building a number of items measuring length x number of items-1 length-1, where each of the matrix box contains three pieces of data, each of which represents a length-2 sequence pattern. Each of data in each box that meets the minimum support will be frequent length-2 list items. Each triangular matrix compared with the minimum support. If the triangular matrix meets the minimum support, then it becomes length-2 sequential patterns. Each length-2 sequential patterns that satisfy minimum support made the projected database, the projected length-1, and the S-Matrix. If the length-1 in the projected database meet the minimum support then S-matrix for length-2 is built.

If the contents of the s-matrix meet the minimum support, then the S-matrix is stored in advance and the process is repeated from the projected database, length-1 for the projected database, and the S-Matrix. Iteration process is complete when projected databases are found to be less than the minimum support. When the iteration process is completed, the results of the S-matrix that satisfies the minimum support is placed behind the x-length and length-1 results from the projected database placed behind the length-x. This process is forming the length-x sequential patterns.

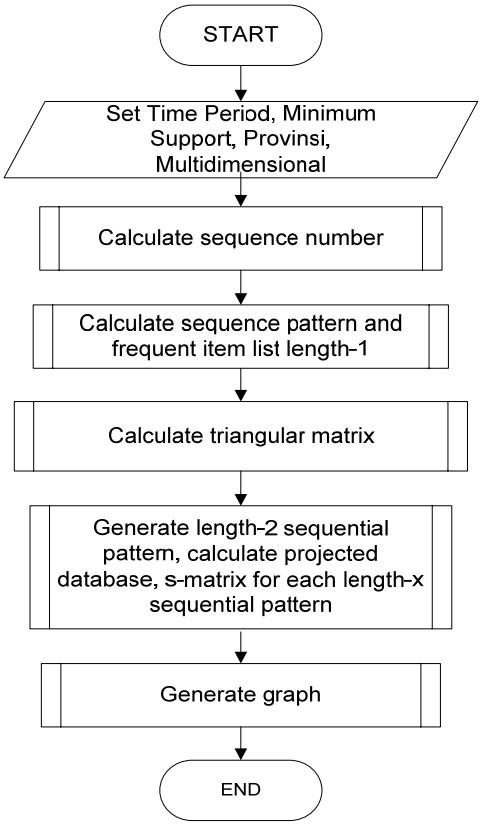

Figure-2. Flowchart of PrefixSpan.

IMPLEMENTATION

FreeSpan and PrefixSpan are testing against data from journal Freespan from Jiawei Han FreeSpan [10]. They also tested against data from journal PrefixSpan from Jiawei Han [9]. The result from the tests can be seen at Table-1 and Table-2. They produce the same sequence, but FreeSpan is more sensitive from producing the sequence.

Table-1. Sequence result from FreeSpan journal.

Sequence of FreeSpan	Sequence of PrefixSpan
{a+ b+} <a+ c> <a a+>	<b, b, a> : 2, <b, b, c> : 2, <b, c, b> : 3, <a, b, a> : 2, <a, b, b> : 2, <b, c, b>:3
{d b+} <d a+>	<(d, b) b> : 2
<e b+>	
{f+ b+} <f f+>	<b, b, f> : 2, <f, b, f> : 2, <(fb) b> : 2, <(fb) f> : 2, <(fb) b, f> : 2
< b b > : 4	< b b > : 4
<b c> : 4	<b c> : 4
<b (c e) > : 2	<b (c e) > : 2
< b c a > : 2	< b c a > : 2
...	...

Table-2. Sequence result from PrefixSpan journal.

Sequence of FreeSpan	Sequence of PrefixSpan
<a a+>	
{a+ b}	
{a+ c+} {b c+} <c c+>	<a, (b, c)>:2, <a, b, a> :2, <a, (b, c) a>:2, <a, c, c> : 3, <a, c, a> : 2, <(b, a) c> : 2, <(c, b) a> : 2
<a+ d> <d c+>	
<a+ f> <f c+>	
<a, a> : 2	<a, a> : 2
<f, b> : 2	<f, b> : 2
<f, b, c> : 2	<f, b, c> : 2
...	...

Testing is done with different amount of data and different minimum support using an interval of 6 days. Figure-3 showed comparison between FreeSpan and PrefixSpan with minimum support 2, 3, and 4 by using 2937 number of data. Figure-4 showed comparison between FreeSpan and PrefixSpan with minimum support 2, 3, and 4 by using 7557 number of data. Figure-5 showed comparison between FreeSpan and PrefixSpan with minimum support 2, 3, and 4 by using 13.982 numbers of data. From these Figures, the graphics showed that FreeSpan processing is faster than PrefikSpan.

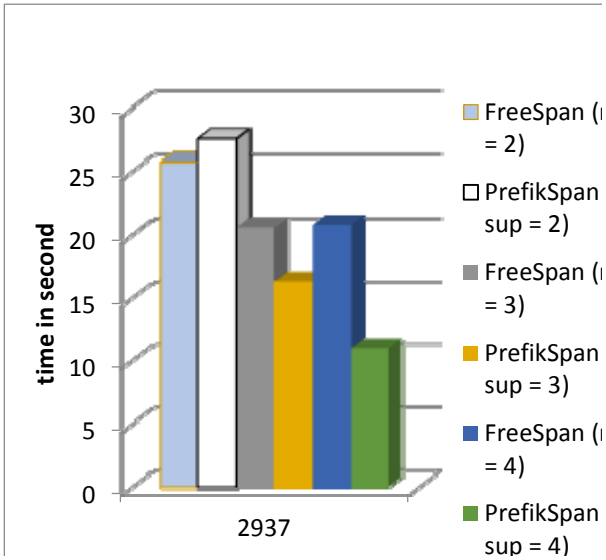

Figure-3. Comparison between FreeSpan and PrefikSpan with 2937 number of data.

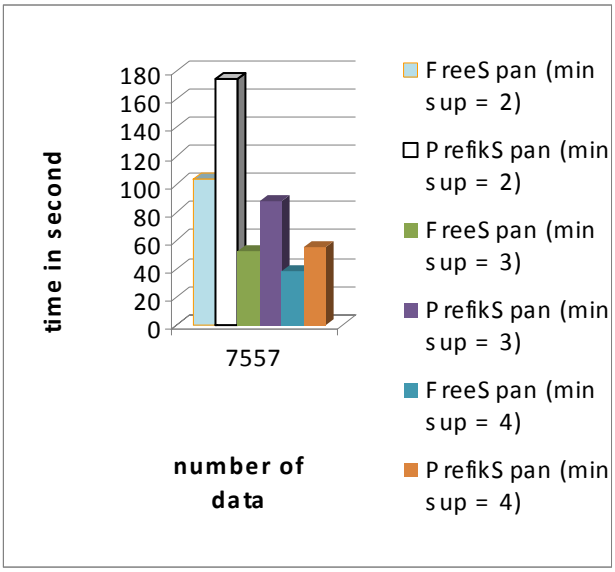

Figure-4. Comparison between FreeSpan and PrefikSpan with 7557 number of data.

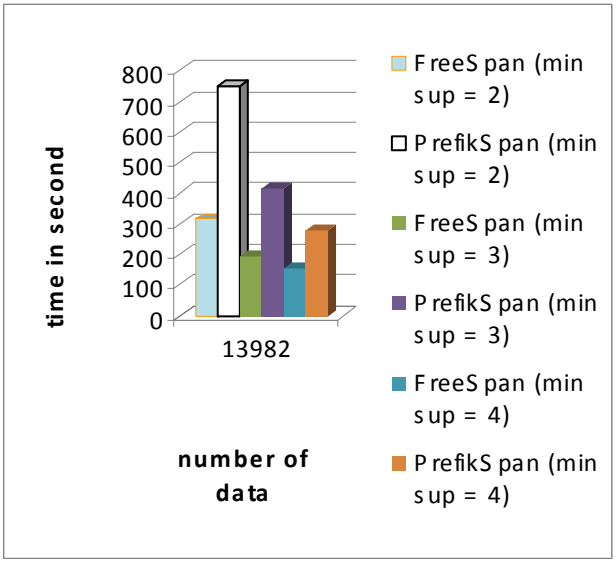

Figure-5. Comparison between FreeSpan and PrefikSpan with 13.892 number of data.

Testing is performed with data from 1 January 2004 until 31 July 2005, with minimum support = 5, time period = 12 days, using East Java province and additional multidimensional district and sex. The result in ICD code can be seen in Figure-6.

Sequential Patterns			Number of Sequence: 820
No	Prefix	Rule in ICD	Number
215	D63.0	<D63.0, Z50.5>	15
216		<D63.0, (Z50.5, Surabaya City)>	6
217		<D63.0, Z50.5, Z50.5>	5
218		<D63.0, D63.0>	13
219		<D63.0, (D63.0, Z50.6>)	5
220		<D63.0, Z50.6>	11
221		<D63.0, (Z50.6, Surabaya City)>	5
222	I08.3	<(I08.3, L.J70.0, N16.8) N36.0>	2
223		<(I08.3, L.J70.0, N16.8) (N36.0, N12.1, Surabaya City)>	2
224		<(I08.3, I08.3>	10

Figure-6. Sequential pattern multidimensional.

The rule can be read in graph as seen in Figure-7. Rule 216 showed that 'anemia in neoplastic disease' is repeatly occurs in Surabaya. Rule 217 showed that 'anemia in neoplastic disease' is followed by 'speech theraphy'.

Figure-7. Graph multidimensional result.

Second testing is performed with data from 1 January 2001 until 31 December 2001, with minimum support = 2, time period = 6 days, using all province and without multidimensional. The result in ICD code can be seen in Figure-8. Rule 255 showed that 'venom of spider' and 'acute transverse myelitis in demyelinating disease of central nervous system' is followed by 'malignant neoplasm of thyroid gland' and 'venom of spider'.

No	Prefix	Rule in ICD	Number
252	F88	<F88, F88, F88>	2
253	S85.8	<S85.8, S85.8, S85.8>	2
254	T63.3	<T63.3, (T63.3, F37.3)>	2
255		<(T63.3, G37.3), (C73.X, T63,3)>	2
256		<(T63.3, G37.3), (C73.X, G37.3)>	2
257		<(C73.X, T63.3), (T63.3, G37.3)>	2
258		<(C73.X, G37.3), (T63.3, G37.3)>	2
259		<(T63.3, G37.3), (T63.3, G37.3)>	2
260		<(T63.3, C73.X), (G37.3, T63.3)>	2
261		<(G37.3, C73.X,), (T63.3, G37.3)>	2

Figure-8. Sequential pattern non multidimensional.

The rule can be read in graph as seen in Figure-9. Rule 254 showed that 'venom of spider' is followed by 'venom of spider' and 'acute transverse myelitis in demyelinating disease of central nervous system'.

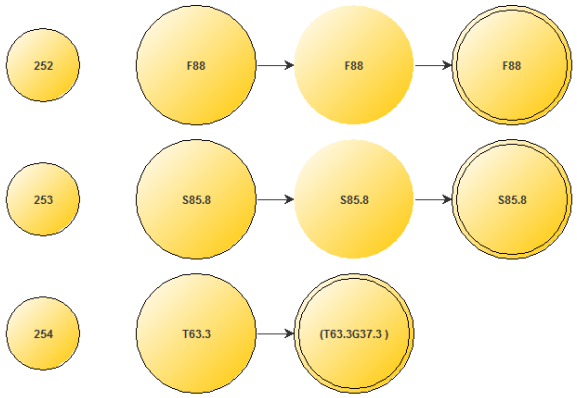

Figure-9. Graph non multidimensional result.

CONCLUSIONS

Mining result is to display the correlation between data (association rules) along with information on support and confidence that can be analysed. The less specified minimum support, the smaller the rules are generated. FreeSpan and PrefixSpan produce the same output. Data from dr. Soetomo General Hospital is more appropiate with FreeSpan rather than PrefixSpan because the processing time of FreeSpan is twice faster than PrefixSpan.

ACKNOWLEDGMENTS

This research is supported by Direktorat Jendral Pendidikan Tinggi, Departemen Pendidikan Nasional (110/SP2H/PP/DP2M/IV/2009, 326/SP2H/PP/DP2M/IV/ 2010, 0082/SP2H/PPK7/KL/IV/2011) with title "Design and Development of Medical Record Data Warehouse Application System for Supporting RSU Dr. Soetomo Strategic Decisions.

REFERENCES

[1] Han J. and Micheline Kamber. 2011. Data mining: concepts and techniques. 3rd ed. Morgan Kaufmann.

[2] Han J. *et al*. 2001. PrefixSpan: mining sequential patterns efficiently by prefix-projected pattern growth. 17th International Conference on Data Engineering.

[3] Parikh M., Bharat Chaudhari and Chetna Chand. 2013. A Comparative study of sequential pattern mining algorithms. International Journal of Application or Innovation in Engineering and Management. 2(2).

[4] Verma M. and Devarshi Mehta. 2014. A comparative study of techniques in data mining. International Journal of Emerging Technology and Advanced Engineering. 4(4).

[5] Chand C. 2012. Sequential pattern mining: survey and current research challenges. International Journal of Soft Computing and Engineering. 2(1).

[6] Grover N. 2014. Comparative study of various sequential pattern mining algorithms. International Journal of Computer Applications. 90(17).

[7] Motegaonkar V.S. and Madhav V. Vaidya. 2014. A survey on sequential pattern mining algorithm. International Journal of Computer Science and Information Technologies. 5(2).

[8] Slimani T. 2013. Sequential mining: patterns and algorithms analysis. International Journal of Computer and Electronics Research. 2(5).

[9] Han J. *et al*. 2004. Mining sequential patterns by pattern-growth: the PrefixSpan approach. IEEE Transactions on Knowledge and Data Engineering.

[10] Han *et al*. 2000. FreeSpan: frequent pattern-projected sequential pattern mining. Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Boston MA, USA.