Analysis of Sequential Pattern Mining

Nikhil Gundawar[1], Venkatesh Akolekar[2], Piyush Phalak[3], Akshay Gujar[4] and L. A. Bewoor[5]

[1]Student, Department Of Computer Engineering, Vishwakarma Instititute of Information Technology Pune,
Savitribai phule University, Pune, India
[1]nik.gun300@gmail.com
[2]Student, Department Of Computer Engineering, Vishwakarma Instititute of Information Technology Pune,
Savitribai phule University, Pune, India
[2]venkatesh.akolekar@gmail.com
[3]Student, Department Of Computer Engineering, Vishwakarma Instititute of Information Technology Pune,
Savitribai phule University, Pune, India
[3]piyushphalak@yahoo.com
[4]Student, Department Of Computer Engineering, Vishwakarma Instititute of Information Technology Pune,
Savitribai phule University, Pune, India
[4]akshaygujar123@gmail.com
[5]Professor, Department Of Computer Engineering, Vishwakarma Instititute of Information Technology Pune,
Savitribai phule University, Pune, India

ABSTRACT

As the instability in environment is increasing day by day, the task of forecasting has become tedious and gradually difficult. Prediction of natural disasters using technology requires extensive research and funding. In current scenario usually constant surveillance is used for this task. By monitoring ocean currents, weather patterns can be predicted in advance, warning populated areas under risk of hurricanes and tornados. However, these short-term warnings are effective only if relief programs are planned and efficiently carried out. We are focused on making these warnings earlier with the help of sequential pattern data mining in which trends found in similar events from past are used to predict climatic hazards like cloudburst and hailstorms.

Keywords — Sequential Pattern Mining, Apriori, GSP, Freespan.

1. INTRODUCTION

Data mining is the method of obtaining required information knowledge and finding of engaging characteristics and trends that are not explicitly represented in the usual databases. These techniques can play an important role in understanding data and in capturing fundamental relationship among data instances, one such technique is sequential pattern mining.

Sequential pattern mining deals with a sequence database consisting of sequences of well-ordered elements or events.

If any metric space, that helps in attaining either total or partial ordering, is considered it can be seen that a sequence is formed in it. Occurrence of certain events in time, e-commerce website traversal, computer networks in routing and something as simple as the spelling of a text string are examples of where the occurrence of sequences may be significant and where the recognition of frequent (entirely or in partial order) subsequences might prove useful. It is for the discovery of these subsequences that the technology of Sequential pattern mining has arisen.

Data mining is the extraction of hidden predictive information from large databases. It is a powerful new technology with great potential to analyse important information in databases and data warehouses. Data mining scours databases for hidden patterns, finding predictive information that experts may miss, as it goes beyond their expectations. [1]

In this project work, the focus is being implied on structuring and using those techniques of data mining to analyze spatial and spatiotemporal data originated in weather prediction domains. Cases of spatial and spatio-temporal data in weather prediction domains include data describing different weather factors. We propose a framework, a generalized one, to discover different types of spatial and spatio-temporal patterns in weather data sets effectively. These patterns can then be further used to recognize a variety of interactions among factor of weather and prediction of abnormalities in the weather.

2. CONCEPT OF SEQUENTIAL DATA MINING

The problem of sequential pattern mining was initially addressed by Agrawal and Srikant in the year 1995 and was defined by them as follows:

"Given a database of sequences, where each sequence consists of a list of transactions ordered by transaction time and each transaction is a set of items, sequential pattern mining is to discover all sequential patterns with a user-specified minimum support, where the support of a pattern is the number of data-sequences that contain the pattern."[5]

Sequential Pattern mining is a topic of data mining concerned with finding statistically relevant patterns between data cases where the values are provided in a particular sequence. It is usually assumed that the values are discrete, and thus time series mining is very similar, but is considered as a different activity. Sequential pattern mining forms a distinctive case of structured data mining. These include constructing databases that are efficient and indexing of sequence information, recognition and extraction of the frequently occurring patterns, examining the sequences for certain similar factors, and regaining the sequence members which are missing. To summarize, the problems of sequence mining can be categorized as string mining which is typically based on string processing algorithms and item set mining which is usually based on association rule learning.

3. TECHNIQUES FOR SEQUENTIAL PATTERN DATA MINING

Sequence Mining Techniques:-

- Apriori algorithm.
- GSP (Generalized Sequence Pattern) algorithm.

3.1. Apriori Algorithm

Apriori is an algorithm for frequent item set mining and association rule learning over transactional databases. It ensues by finding the individual items, which occur frequently, in the database and encompassing them to item sets, which are although larger but the item sets still appear sufficiently often in the database. These item sets, which occur frequently and discovered by Apriori, can be utilized to determine association rules which highlight general trends in the database: the applications of which can be seen in domains such as market basket analysis.

Apriori is intended to operate on databases encompassing transactions like compilation of stuff purchased by customers. Whereas some algorithms are structured to discover association rules in data which consist of no transactions, no timestamps (a common example being the sequencing of DNA). In the Apriori, each transaction is considered to be a set of items. Given a threshold *C*, the item sets are identified by the Apriori algorithm which are subsets of at least *C* transactions in the database. The "bottom up" approach is used by Apriori, where frequent subsets are extended one item at a time (a step known as *candidate generation*), and clusters of candidates are verified against the data. The algorithm ends when no further positive extensions are acknowledged.[5]

Apriori uses breadth-first search and a Hash tree structure to count candidate item sets efficiently. Then a candidate item sets of length *k* is generated from item sets of length *K-1*. Then it prunes the candidates whose sub pattern are infrequent. According to the downward closure lemma, the candidate set contains all frequent *k*-length item sets.[2] After that is done, the transaction database is scanned to determine frequent item sets among the candidates.

Following is the pseudo code for the algorithm for a transaction database T, and a support threshold of ϵ. The usual set theoretic notation is employed, though it is to be noted that *T* is a multiset. C_k is the candidate set for level k. Generate() algorithm is assumed to generate the candidate sets from the large item sets of the preceding level, by notifying the downward closure lemma. The data structure that represents candidate set C, which is initially assumed to be zero, is accessed by the count[c]. Many details are omitted below, the data structure used for storing the candidate sets, and counting their frequencies is usually the most important part of implementation.

```
Apriori($T, \epsilon$)
    $L_1 \leftarrow \{\text{large } 1 - \text{itemsets}\}$
    $k \leftarrow 2$
    while $L_{k-1} \neq \emptyset$
        $C_k \leftarrow \{a \cup \{b\} \mid a \in L_{k-1} \wedge b \in \bigcup L_{k-1} \wedge b \notin a\}$
        for transactions $t \in T$
            $C_t \leftarrow \{c \mid c \in C_k \wedge c \subseteq t\}$
            for candidates $c \in C_t$
                $count[c] \leftarrow count[c] + 1$
        $L_k \leftarrow \{c \mid c \in C_k \wedge \ count[c] \geq \epsilon\}$
        $k \leftarrow k + 1$
    return $\bigcup_k L_k$
```

3.2. GSP Algorithm

GSP Algorithm [3] (*Generalized Sequential Pattern* algorithm) is an algorithm used for sequence mining. The algorithms for solving the problem based on sequence mining are mostly based on the a priori (level-wise) algorithm. One way to utilize the level-wise paradigm is to initially find all the frequent items in a fashion which is level-wise. This simply accounts to the counting of occurrences of all singleton elements in the database. Then, non-frequent items are removed to filter the transactions. At the end of this stage, each transaction contains of only the recurrent elements it initially contained. This modified database is then provided as an input to the GSP algorithm. This process requires one pass over the whole database.

Multiple passes over the database are made by the GSP Algorithm. In the initial pass, all singular items (1 sequences) are counted. Then from the found frequent items, a set of candidate 2-sequences are formed, and next pass is made to identify their frequency. These frequent 2-sequences are used to generate the candidate 3-sequences, and this method is repeated until no more recurrent sequences are discovered. The algorithm consists of two main steps.

- Candidate Generation. Provided the set of frequent (i-1)-frequent sequences Fn(i-1), the candidates for the next pass are generated by joining Fn(i-1) with itself. At least one of whose subsequences are not frequent is eliminated by the pruning phase.
- Support Counting. Usually, a hash tree–based search is used for efficient support counting. Then finally non-maximal frequent sequences are removed.

Algorithm

```
Fn1 = the set of frequent 1-sequence
  i=2,
  do while Fn(ii-1)!= Null;
     Generate candidate sets Cni (set of candidate i-sequences);
        For all i/p sequences Q in the database D
           do
        Raise count of all a in Cni if Q supports a
        Fni = {a ∈ Cni such that its frequency exceeds the threshold}
           i= i+1;
   Result = Set of all recurrent sequences is the union of all Fnis
           End do
  End do
```

This algorithm is similar to the Apriori algorithm. But there is one main difference, the generation of candidate sets. Let us assume that:

A → B and A → C

are two frequent 2-sequences. The items being used in these sequences are (A, B) and (A,C) respectively. The candidate generation of a normal Apriori algorithm would display output (A, B, C) as a 3-itemset, but in the present context, following is the 3-sequences due to joining of the above 2- sequences

A → B → C, A → C → B and A → BC

This is taken into account by the candidate–generation phase. Frequent sequences are detected by the the GSP algorithm, which allows for time constraints like the maximum gap and minimum gap among the sequence elements. Not only that,the notion of sliding window is supported by it, i.e., even if the time events originate from different events of a time interval, items are observed as belonging to the same event.

4. DATA PROCESSING

4.1 Data Collection

The dataset used for this work is to be collected from Indian Meteorological Department, Pune. The dataset would contain the data from last ten years.

4.2 Data Preprocessing

In this Project, the weather data is used which consists of various parameters as temperature, humidity, rain, wind speed etc., pre-processing means removing the other unwanted parameters from the dataset. Data pre-processing includes following processes:

4.3.1 Data Cleaning

In this stage, a consistent format for the data model is to be developed which will take care of missing data, finding duplicated data, and weeding out of bad data. Finally, the cleaned data will be transformed into a format suitable for data mining.

4.3.2 Data selection

At this stage, data relevant to the analysis will be decided on and retrieved from the dataset.

4.4 Data Transformation

This is also known as data consolidation. It is the stage in which the selected data is transformed into forms appropriate for data mining.

4.5 Analyzing Data Using Predictive Data Mining Concepts

In this phase the extracted patterns are processed using predictive data mining algorithms which would lead to a improved approximate prediction.

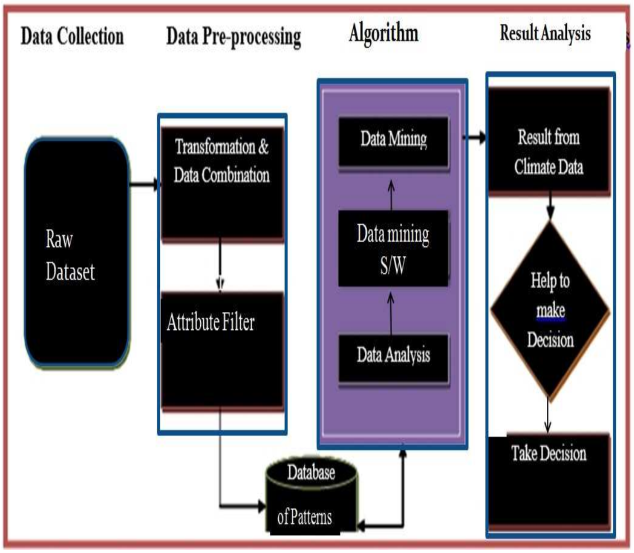

Fig 1: Basic workflow of the system.

5. CONCLUSION

There have been previous attempts of using sequential pattern mining in the fields of E-commerce, website intrusions, Book recommendation system with profitable consequences. We have tried to use this effective concept for the social cause of preventing the loss of life and resources by predicting those calamities beforehand and much quicker than the existing norms. As the uncertainties rises so does the patterns and easier the predictions becomes.

REFERENCES

[1] Han J., Kamber M., "Data Mining: Concepts & Techniques", Morgan & Kaufmann, 2000.

[2] Anita Agustina and Nor Azura Husin, "Sequential pattern mining on library transaction data", IEEE conference, 2010. doi:978-1-4244-6716-7/10/©2010

[3] Arthur Pitman and Markus Zanker, "Insights from Applying Sequential Pattern Mining to Ecommerce Click Stream Data", IEEE conference, 2010. doi:10.1109/ICDMW.2010.31

[4] Soumadip Ghosh and Amitava Nag, "Weather Data Mining using Artificial Neural Network", IEEE conference, 2011. doi:10.1109/RAICS.2011.6069300

[5] Carl H. Mooney and John F. Roddick, "Sequential Pattern Mining – Approaches and Algorithms", ACM Journal Name, Vol. V, No. N, M 20YY.