Sequences Modeling and Analysis Based on Complex Network

Li Wan[1], Kai Shu[1], and Yu Guo[2]

[1] Chongqing University, China
[2] Institute of Chemical Defence People Libration Army
{wanli,shukai}@cqu.edu.cn

Abstract. In this paper, we present a method to model frequent patterns and their interaction relationship in sequences based on complex network. First, an algorithm NOSEM is proposed to find non-overlapping pattern instances in sequence. Then, we give a new way to construct state model of sequence formed by non-overlapping patterns, namely pattern state model. The proposed pattern state model of sequence is a graph-like model. We discover that the graph formed by non-overlapping frequent patterns and their interaction relationship is a complex network. Experiments on real-world datasets and synthetic datasets show that the pattern sate models formed by the frequent patterns of sequences in almost all the domain are complex network. However, models in different domains have distinct power-law values, which are used to classify various types of sequence.

Keywords: Complex network, Frequent pattern, Power-law.

1 Introduction

Sequence analysis, discovering knowledge in the structure of sequences, is an important problem in various domains, such as sensor network, biological informatics, natural disasters prediction, and so on.

In the decades, abundant literatures has been dedicated to discovery frequent patterns in sequence [1, 2, 3, 4, 5, 6].Most of them focus on the speed of the algorithm and generative model of sequences based on the discovered frequent patterns [2, 3, 4, 5, 6]. Studying the features of sequences by subsequences they included is a new trend. Lei Zhou etc. [1] try to use symbol state to divided sequences to study weather system, and analyze sequence features based on complex network. However, Lei Zhou etc. formulated the state model by subsequences occur successively in a sequence. In this paper, we consider frequent patterns in sequences and generate a new model based on the patterns. As illustrated inFig.1, if the minimum support is set as 2, we can get frequent patterns (i.e. “RR”,”RD”,”RdD”) from the sequence listed in Fig. 1(a). As shown in Fig.1 (b), the occurrence of a pattern is represented as a time interval. Then, we can model the patterns and their relationships by a graph-like model. Suppose each pattern as a vertex, if two instances of different patterns are *overlapping*(i.e. the corresponding time interval intersect with each other), then an edge exists between the corresponding vertices of the patterns. Finally, we formulate the patterns in the sequence given in Fig.1 (a) into a pattern-state-model which is shown in Fig.1(c).

M. Zhao and J. Sha (Eds.): ICCIP 2012, Part I, CCIS 288, pp. 246–252, 2012.

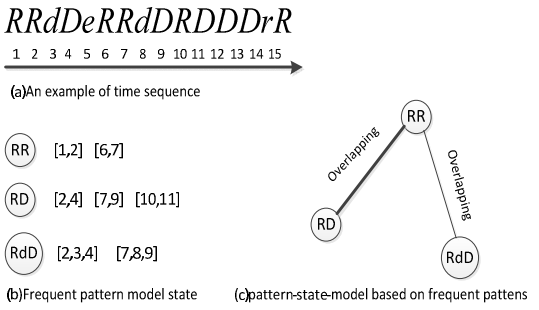


Fig. 1. An example of pattern-state-model

So the aim of this paper is to use frequent patterns to construct *pattern state model* of sequences and study the features of sequences through the features of pattern state model. We find and prove that the pattern state models of sequences generated in many domains are complex network (see details in Sec. 5).

Generally, frequent patterns are subsequences with frequency no less than a user-defined threshold (i.e. minimum support). According to the position occurring in sequences, there are various types of frequent patterns. Therefore, selecting a proper type of pattern plays an important role in formulating our model. One type of frequent pattern that are usually chosen to form generative model of sequence is non-overlapping pattern [3]. If any two instances of a pattern are non-overlapping (i.e. the corresponding time intervals of instances are not intersected with each other) then it is a non-overlapping pattern. For example, the patterns in Fig.1 (b) are all non-overlapping patterns. We also choose *non-overlapping frequent patterns* to construct pattern state model, such that the vertices in pattern sate model do not have self-loop edges.

2 Related Work

Lei Zhou etc. [1] analyze temperature sequence and found its complex feature. They divide the sequence to subsequences with the same length successively and formulate them into a complex network model. However, Lei Zhou etc. does not consider frequent patterns of sequence in their model.

S. Laxman etc. [3] defined non-overlapping episode and proposed the first algorithm to discovery non-overlapping episodes. Meger and C. Rigotti. [2] proposes a complete algorithm (i.e. WinMiner) to find frequent episode pattern in a single long sequence. J. Pei and J. Han [7] presented the first algorithm discovering sequential patterns.

3 Problem Formulation

3.1 Notations and Basic Concepts

Definition1 (Sequence). A sequence of objects, $T = \{<o_1, t_1>, <o_2, t_2>, ..., <o_i, t_i>\}$, Where $O = \{o_1, o_2, ..., o_i\}$ represents the symbols of different types of objects and t_i the time of occurrence of the i th object.

Definition2 (Serial Episode). Given a sequence and a minimum support γ. A sequence occurs once in this sequence contributes 1 to its support. If a sequence's support is larger than γ, it's a serial episode. We also use episode as a short form of serial episode. If no specific instructions, serial episode and episode are equal.

Definition 3 (Non-overlapping episode). Suppose an episode occurs in the sequence twice, if any object associated with either occurrence doesn't occur between the objects associated with theother occurrence, then these two instances of the episode are called non-overlapping episode.The frequency of an episode is defined as the maximum number of non-overlapping occurrences of the episode in sequence.

Definition 4 (Minimum Occurrence). Let $[t_s, t_e]$ be an occurrence of an episode ∂ in the sequence S If there is no other occurrence $[t_s', t_e']$ such that $(t_s < t_s' \wedge t_e' \leq t_e) \vee (t_s \leq t_s' \wedge t_e' < t_e)$ (i.e. $[t_s', t_e'] \subset [t_s, t_e]$), then the interval $[t_s, t_e]$ is called a minimum occurrence of ∂. As shown in Fig.1, [2,3,4] is a minimum occurrence of "RdD",while [1,3,4] is not.

3.2 Problem Formulation

Definition 5 (Overlapping relation). Serial episode α overlaps β, if and only if $\alpha.begin < \beta.begin \leq \alpha.end < \beta.end$, where $e.begin$ and $e.end$ denotes the begin time and end time of episode e respectively. The overlapping relationship of α and β is donated as $\alpha\_overlap\_\beta$. As shown in Fig.1, [1,2] and [2,4] are overlapping.

Definition 6 (pattern-state-model). An undirected graph with vertices state frequent pattern instances and edges present the overlapping relationship of vertices. If two pattern vertices are overlapping, an edge between them is added. As shown in Fig.1(c), it's a pattern-state-model. Pattern "RR" and "RD" are overlapping, so an edge is connected between them.

Problem: To represent a sequence by a pattern-state-model and study the features of sequences by the features of their corresponding pattern-state-model.

Sub-problem 1: discovering all the non-overlapping episode in sequences
Sub-problem2: constructing pattern-state-model and analyze the feature of the sequences by the degree distributions in pattern-state-models.

4 From Frequent Pattern to Complex Network

In this section, we first present our method to generate model state: non-overlapping pattern. Then the algorithm NOSEM is given to discovery non-overlapping patterns.

Based on the model states, we build network graph and analyze its topological characteristics. Finally, we discover the complex network feature in sequences.

4.1 Frequent Pattern Discovery

The algorithm NOSEM is used to discovery non-overlapping patterns with episode rules of window sizes. We first introduce the algorithm NOSEM using the example in Fig.2,the detail description of NOSEM will be present later.

For the sequence in Fig.2, we definite the minimum min-support γ=2and gap-max ω=3,the main algorithm steps are as follows:

Step1: Scan the whole sequence, finding size-1 frequent patterns. There are 5 types of size-1 patterns, <R>, <r>, <D>, <d>, <e>.

Step2: Join existing frequent patterns with every size-1 patterns, getting frequent patterns with size greater than 1 with non-overlapping instances.

Step3: Iterate the join process, finding all non-overlapping frequent patterns.

4.2 Pattern-State-Model

In this section, we construct pattern-state-model based on all non-overlapping patterns discovered from sequence. Then we analyze the model and found it's a complex network through power-law distribution. Based on complex network features, for example degree distribution, shortest path of graph, clustering coefficient and power-law, we analyze the characteristics of sequence. In this paper, we focus on power-law value because it's the core feature of complex network.

We consider the sequence in Fig.1 with minimum non-overlapping min-support γ=2,and gap-max ω=3. Through algorithm NOSEM we can get all the non-overlapping instances of frequent patterns. If two instances are overlapping, a direct edge is added between them (e.g. illustrated in Fig.2).

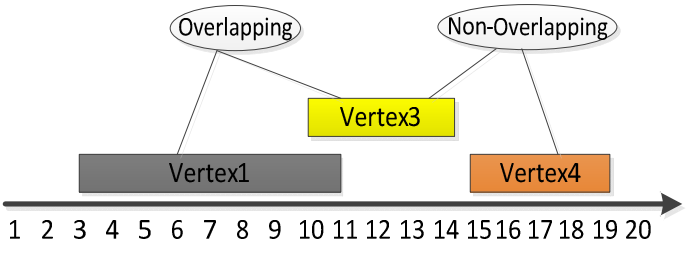


Fig. 2. Examples of pattern relations: overlapping and non-overlapping

The degree distribution of complex network follows power-law distribution. Different sequence may have different network model, therefore analyzing the degree distribution is an important step to discovery complex feature.

We analyze accumulated distribution graph of degrees and discovery sequence system's complex feature through power-law values. Various complex systems can be distinguished with different power-law values.

5 Experiments

We do experiments on sensor network dataset of temperature, gene sequence (splice), Lorenz system simulation dataset and synthetic datasets. We use exponential function to fit the accumulated distribution of degree. If the *fitting correlation*, which means similarity between real data and target function, is above 90%,we accept the complex feature. The result shows that all the datasets accord with complex feature to a very high probability.

All the datasets are generated as follows: Intel Lab Sensor Dataset [10] (using IL Sensor for short) has been collected from 54 sensors from February 28th to April 5th, 2004. We only select continuous 50000 data and discretize them using equal probability thoughts with symbols in {R, r, d, D}; we also evaluate Lorenz System and discretize it with the algorithm SAX [11].

All the experiments are performed on a 2.10GHZ Intel Core 2 PC machine with 2.00GB main memory, running Microsoft Windows 7.All algorithms are implemented in Java. Data fitting is completed in Matlab2010.

5.1 Experiments on Real-World Sequences

We use the algorithm NOSEM to discover non-overlapping pattern instances from the sequence.Every instance has been treated as vertex. If any two vertices are overlapping, both degree of them increase by 1.We use exponential function to fit degree's accumulative degree distribution.Result is acceptable when fitting correlation is above 90%. Finally we get Power Law Value Distribution (PLVD for short) graph.

5.1.1 IL Sensor Network Dataset

Fig.3 illustrate 46 of 48 sequences (with probability of 95.83%) accord with power-law distribution. The average value is approximately equal to -0.242.

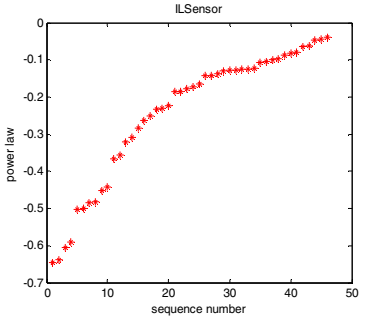


Fig. 3. The power law value distribution of 46 IL Sensor dataset sequences

5.1.2 Real Gene Sequence (Splice)

Fig.4 shows that the accumulation distribution of degree accords with power-law distribution to a high probability. The fit correlation is greater than 90%, which means this sequence is a complex network.

Fig.5 illustrate nearly all 100 sequences (with the probability of 99%) accord with complex feature (i.e. power-law distribution).While the power-law value is concentrated between -0.15 and -0.25.The average value isapproximately equal to-0.389.

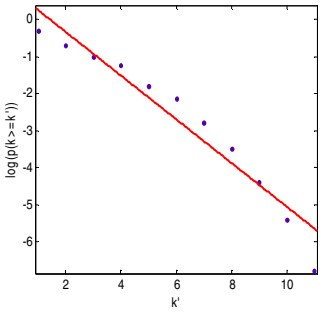


Fig. 4. Accumulation distribution of degree in a sequence: splice

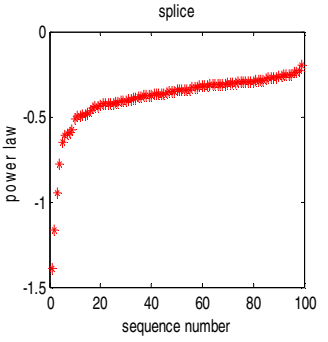
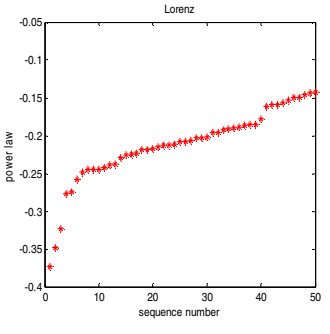


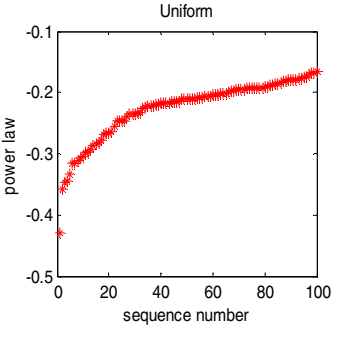
Fig. 5. The powerlaw value distribution of 100 splice sequences

5.2 Experiments on Synthetic Dataset

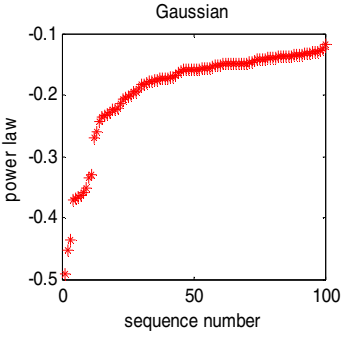
Fig.6 shows the power-law value distribution of sequences generated from synthetic datasets with length 500.They illustrate that all the synthetic sequences accord with complex feature with the probability 100%.



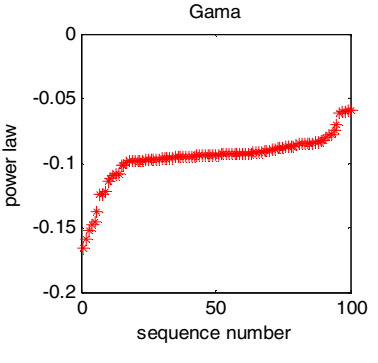
(a) The PLVD of Lorenz system sequences



(b) The PLVD of Uniform sequence



(c) The PLVD of Gaussian gene sequences



(d) The PLVD of Gama sequences

Fig. 6. The power law value distribution of 4 kinds of synthetic sequences (Lorenz, Uniform, Gaussian, Gama)

In summary, our performance study proved that all the datasets in this experiment are complex network system. We use frequent pattern instances to state sequence model and build complex network model. Power-law value can be significant approach to predict different system.

6 Conclusions

In this paper, we present complex network features in sequences using frequent pattern mining method. We first present that using non-overlapping frequent pattern to construct pattern state model, and we propose the algorithm NOSEM to mining non-overlapping pattern instances from sequence. We state the pattern network distribution and find various sequence including gene sequence, sensor network and synthetic dataset are complex network systems. While these systems has different inner feature (i.e. power-law value) so we can use this to predict and separate them.

References

1. Zhou, L., Gong, Z.-Q., Zhi, R., Feng, G.-L.: An Approach to Research the Topology of Chinese Temperature Based on Complex Network. Acta Physica, Sinica (2008)
2. Méger, N., Rigotti, C.: Constraint-Based Mining of Episode Rules and Optimal Window Sizes. In: Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D. (eds.) PKDD 2004. LNCS (LNAI), vol. 3202, pp. 313–324. Springer, Heidelberg (2004)
3. Laxman, S., Sastry, P.S., Unnikrishnan, K.P.: Discovering Frequent Episodes and Learning Hidden Markov Models: A Formal Connection. IEEE Computer Society 17(11), 1505–1517 (2005)
4. Mannila, H., Toivonen, H.: Discovering Generalized Episodes Using Minimal Occurrences. In: Proceedings of SIGKDD (1996)
5. Laxman, S.: Stream Prediction Using A Generative Model Based On Frequent Episodes In Event Sequences. In: Proceeding of KDD 2008 (2008)
6. Carl, H.M., John, F.R.: Mining: Relationships Between Interacting Episodes. SIAM (2004)
7. Pei, J., Han, J., Pinto, H., Chen, Q., Dayal, U., Hsu, M.-C.: PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth. In: 17th Int' l Conference Data Eng., pp. 215–224 (2001)
8. Newman, M.E.J.: The Structure and Function of Complex Network. SIAM 45(2), 167–256 (2003)
9. Newman, M., Barabasi, A.L., Watts, D.J.: The Structure and Dynamics of Networks. Proceeding of Journal of Statistical Phicsics 126(2), 419–421 (2007)
10. Intel Lab Data, `http://db.csail.mit.edu/labdata/labdata.html`
11. SAX (Symbolic Aggregate approXimation), `http://www.cs.ucr.edu/~eamonn/SAX.htm`