2015 Fifth International Conference on Instrumentation and Measurement, Computer, Communication and Control

User Behavior Detection for Online Survey via Sequential Pattern Mining

Xiaowei Zhu, Shaochun Wu, Guobing Zou
School of Computer Engineering and Science, Shanghai University
Shanghai 200444, China
wxz_zhuxiaowei@163.com, scwu@shu.edu.cn, gbzou@shu.edu.cn

***Abstract*—With the rapid development of Internet, online survey becomes an emerging industry. It is a very challenging task to get interesting knowledge from the large-scale behavioral data of respondents. This paper firstly makes reduction of user properties and behavior data from an online survey company, and based on which we construct an online survey user model; then, an improved generalized sequential pattern (GSP) algorithm is proposed to mine frequent sequential patterns; finally, we give an in-depth user behavior analysis of online survey, which is from conventional sequential patterns of user behavior, sequential patterns based on specific behavior and time window, and user behavior prediction. The experimental results show that it is effective to analyze the sequence of user behavior thorough improved GSP algorithm. Compared with the classical GSP algorithm, user behavior prediction accuracy rate increases 19% via our proposed sequential pattern analysis approach.**

Keywords—*Online survey; Sequence analysis; User behavior prediction; GSP algorithm*

I. INTRODUCTION

Online survey as the product of the development of the Internet, its development has received many attentions and gradually applied in many applications so far. With the increasing accumulation of respondents' behavior data and useful information, hidden patterns mining from these data will be of vital importance because they can provide decision support for the questionnaire invitation and promote reasonable developing strategies for a company. User behaviors in the online survey are consisted of registration, login, successful response, fast response, termination response, interruption response and exchange prizes. Among those behaviors, fast response refers that respondents are terminated to answer questionnaire, because they answer questionnaire too fast, while termination response refers that respondents are terminated to answer questionnaire due to their incorrect answer logic. Interruption response is that users give up the response on their own initiative. Accordingly, exchange prizes refers to applying response points to exchange corresponding prizes by users. How to use these behavior data effectively to obtain interesting knowledge that can be used for effective online survey will be a huge research challenge to be solved.

Sequential pattern mining, which is an important and powerful tool in data mining, is used to discover ordered events that occur frequently. It is similar to association rule mining in many ways. However, sequential pattern mining is concerned more about the ordered sequence of events. The application fields are highly correlative with intrusion detection, user behavior prediction, user state analysis, natural disaster prediction, and DNA sequences decipher [1]. Though classical GSP algorithm performs a good effect in the field of frequent sequences mining, it still has two obvious deficiencies in the application of user behavior sequence analysis for online survey. Firstly, the effectiveness of using historical frequent sequential pattern library that is produced by classical GSP algorithm cannot be applicable to predict behavior, because precise matching is not used in the support counting phase. Secondly, using classical sequential pattern mining algorithm can get frequent sequential pattern, but it does not take time dimension into account. So, two sequences are the same if they contain same items, however, the time interval between items of each sequence may be different, and it will lead to the meaning loss of sequential patterns in terms of practical applications.

In order to solve these problems, an improved GSP algorithm is proposed in this paper. We give the specific application scene in sequential patterns based on the specific behavior and time window, conventional sequential patterns of user behavior and user behavior prediction respectively. We conduct extensive experiments to validate the effectiveness of our proposed user behavior analysis approach. Compared with the existing one, the experimental results demonstrate that improved approach can be deployed conveniently and effectively for enterprise online survey.

The remainder of this paper is arranged as follows. Section II reviews the related work on sequential pattern mining, while user model and user behavior sequence is presented in Section III. We present user behavior analysis and its algorithm in Section IV. Experimental results are presented in Section V. Finally, we conclude the paper in Section VI.

II. RELATED WORK

R. Agrawal et al. proposed the concept of sequential pattern in 1995. At the same time, they put forward three kinds of sequential pattern mining methods, including AprioriAll algorithm, AprioriSome algorithm and DynamicSome algorithm [2]. Later, R. Agrawal et al. proposed a more general algorithm of sequential pattern mining, namely GSP (Generalized Sequential Patterns) algorithm [3]. Mohammed. Zaki et al. considered the lattice of mathematics into sequence pattern mining, and proposed Spade algorithm [4]. J. Han et al proposed TSP-k algorithm that can adjust the degree of support adaptively, so as to meet the needs of the algorithm better in practical application [5]. At present, sequential pattern mining algorithms can be divided into Apriori characteristics based and sequential pattern growth based. As to the former one, it mainly includes the AprioriAll algorithm and the GSP algorithm. This kind of algorithm needs to scan the database repeatedly. For the latter one, it mainly consists of Freespan algorithm and Prefixspan algorithm. This kind of algorithm adopts a "divide and conquer" idea to reduce search space and improve the performance of the algorithm. The

978-1-4673-7723-2/15 $31.00 © 2015 IEEE
DOI 10.1109/IMCCC.2015.110

comparison of four kinds of sequential pattern mining algorithms is shown in the following table 1. Most of researchers improve GSP algorithm through the reduction of the database scanning times and the number of candidate sequences, but few of them have considered time factors and sub-sequence matching in the support counting phase.

Table 1. Comparison of four kinds of sequential mining algorithms

Attribute	Apriori		Pattern growth	
	AprioriAll	***GSP***	***Freespan***	***Prefixspan***
candidate sequence	yes	yes	no	no
data structure	Hash tree	Hash tree	Hash tree	WAP tree
database partition	no	no	yes	yes
number of scans	many times	many times	3	2
execution method	cycle	cycle	recursive	recursive

III. USER MODEL AND USER BEHAVIOR SEQUENCE EXTRACTION

In order to ensure the consistency and validity of user behavior sequence, we need to construct a general user model to represent and describe user characteristics.

A. User model structure

We aim at discovering user behavior sequences, so the user is modeled by three components, i.e., user ID, basic attributes and user behavior sequence. It is formalized as follows.

User= {UserId, BaseAttribute, UserSequence}

Among them, UserId represents user's id, BaseAttribute consists of the basic attributes of the user, including the user's name, age, gender, sex and address, which can be expressed as:

BaseAttribute= {name, age, sex, address ...}

UserSequence stands for the user behavior sequence, which is composed by many user behaviors according to the time order. The formal definition of UserSequence and its Behavior is shown as follows.

UserSequence= (behavior1, behavior2, behavior3...)

Behavior= (BehaviorId, BehaviorType, IntervalTime)

User behavior consists of BehaviorId that represents BehaviorType's id, BehaviorType and IntervalTime that represents the time between two behaviors. The structure of user model is illustrated in the following Figure 1.

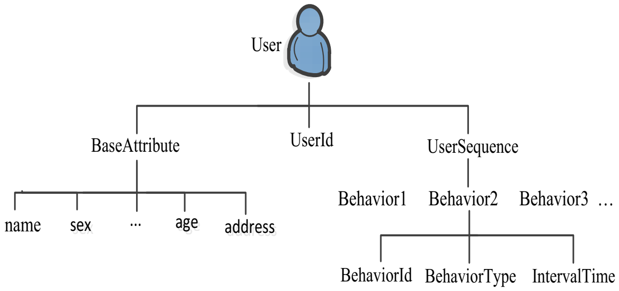

Figure 1. Structure modeling of an active user

B. User Behavior sequence of online survey

Behavior data recorded by online survey is always incomplete and dirty, so it cannot be directly used to make sequence pattern analysis, or it will lead to bad mining results. In order to improve the quality of data mining, we must have a data pretreatment before we extract user behavior sequence. There are many methods for data preprocessing, such as data cleaning, data filling, data transformation, data reduction, etc.

We use the actual business database from an online survey company as the experimental datasets and extract corresponding attributes as the values of our constructed user model. For the data pretreatment, we firstly delete a behavior record if its user ID is NULL, because these data is not only meaningless, but also would bring unnecessary troubles to the algorithm processing. For those data recordings of some user's behavior that often have null values in the field of time, we take compromise step to fill this blank through the calculation of the average time of all behaviors for the user. Furthermore, the field of time of real business database will be accurate to seconds, which will cause the data processing inconvenient. To preprocess this type of data, we make the time field precise to the day. In addition, in order to simplify the data processing, those behaviors that only appear once or rarely appear will not be considered when we build user behavior sequence. For example, registration will only occur once for each user, thus it is excluded in user behavior sequence. Excluding those low frequency user behaviors, we extract six kinds of behaviors totally, including login, successful response, fast response, termination response, interruption response and exchange prize. A simple code on these behaviors will be given to simplify the data processing in our algorithm, and their corresponding codes are shown in Table 2. Figure 2 illustrates an active user behavior sequence after data preprocessing. User's id is 40096 and the time period of occurrence span from 2013-9-21 to 2013-10-31.

Table 2. Behavior code and its corresponding user action

Behavior ID	Behavior Type
1	login
2	successful response
3	fast response
4	termination response
5	exchange prize
6	interruption response

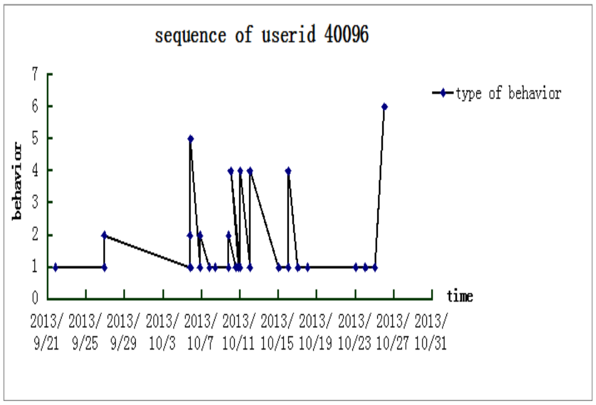

Figure 2. User behavior sequence evolution for an active user

IV. USER BEHAVIOR PATTERN DETECTION

A. *Analysis of user behavior sequence*

Behavior sequence refers to a sequence of user actions that are in accordance with time sequence. The process of using a specific data mining algorithm to discover sequence pattern from the user behavior sequences is called the analysis of user behavior sequence. AprioriAll algorithm, Spade algorithm and GSP algorithm are commonly used to analyze user behavior sequence. Especially, GSP algorithm has been extended from AprioriAll algorithm, through the constrained conditions to reduce the amount of useless candidate set. At the same time, the GSP algorithm that exploits hash tree structure to store the candidate sequences can improve the lookup and counting efficiency of candidate set.

In this paper, we detect the user behavior pattern from the user behavior sequences by three aspects when using an improved GSP algorithm, including sequential patterns based on the specific behavior and time window, conventional sequential patterns of user behaviors and user behavior prediction.

B. *Classical GSP algorithm*

The main idea of the GSP algorithm is that through connecting and pruning operations to get the candidate sequential patterns Ci+1 whose length is i+1 according to the seed set Li whose length is i; then, we scan sequence database to calculate the support of each candidate sequential patterns, and generate the sequential pattern Li+1 that will be chosen as the new seed set, whose length is i+1. This process is iterated until there is no new sequence pattern or new candidate sequences pattern. In the whole process, the counting object is a sequence, rather than a single element [7]. Classical GSP algorithm is descripted as follows.

```
Input: Sequence database S, Minimum support min_sup
Output: Set of frequent sequences
L= {1 sequential pattern}
R=L;
For (k=2; $F \neq \Phi$; k++) do begin
    C = GSP_generate (L);
    For (sequence s in database S) do begin
        If s contains sequence a && $a \in C$, AddCount(a);
    End For
    $F = \{a \in C | Count(a) \geq min\_sup\}$;
    $R = R \cup F$;
End For
Return R
```

In the formal description, GSP_generate (L) contains two phases, connection and pruning. In the connection phase, if the sequential patterns S1 that removes the first item and the sequential pattern S2 that removes the last item are the same sequence, S1 and S2 will be connected. That is to say, the last item of S2 will be added to S1. In the pruning phase, a candidate sequential pattern cannot be a sequential pattern if it has a subsequence that is not a sequential pattern, and then it is removed from the candidate sequential patterns. AddCount (a) is used to add 1 for the support counting of candidate sequence a, while Count (a) is used to get the support counting of candidate sequence a.

C. *Improved GSP algorithm*

In the support counting phase of classical GSP algorithm, the time interval between two events isn't considered, for example, login 10 times within 1 day is equivalent to 10 times within 1 month. However, these two cases are entirely different in the practical application, which significantly affects the decision-making and forecasting for the next act. Moreover, the method used in classical GSP algorithm for support counting phase does not make an exact matching. Therefore, the limit among items between two sequences of user behavior is not very strict. When scanning the database for support counting calculation, as long as a user behavior record S contains the candidate sequence C, then, increase support for sequence pattern C. However, C is not a substring of S, and this will cut down the prediction accuracy to a certain extent. Improved GSP algorithm for user behavior pattern detection is shown as its flow chart in the following Figure 3.

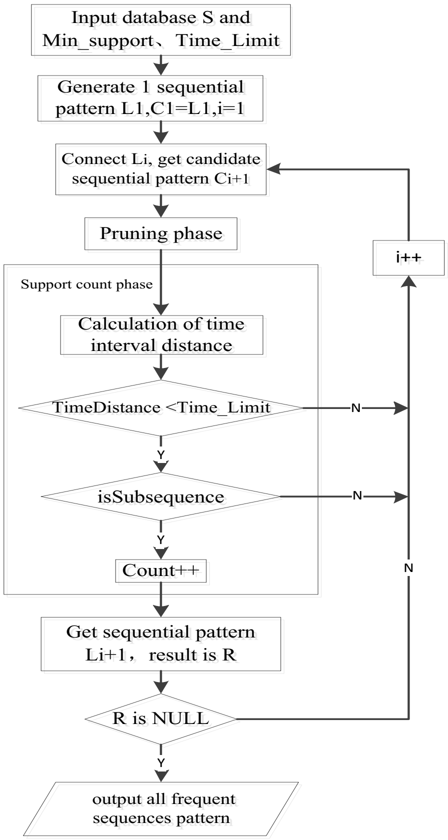
Input database S and
Min_support、Time_Limit
Generate 1 sequential pattern L1,C1=L1,i=1
Connect Li, get candidate sequential pattern Ci+1
Pruning phase
Support count phase
Calculation of time interval distance
TimeDistance <Time_Limit
N
Y
isSubsequence
N
Y
Count++
N
i++
Get sequential pattern Li+1，result is R
R is NULL
Y
output all frequent sequences pattern

Figure 3. Flow chart of our improved GSP algorithm for user behavior pattern detection

To solve the first application deficiency of classical GSP algorithm, we add a behavior time interval distance calculation when we judge whether candidate sequence S1 is the subsequence of a user behavior sequence S2. Cosine distance is used to calculate the time interval distance. Sequence S1 is the subsequence of S2 if the time interval distance is less than the given threshold. In this case, we increase the support of sequence S1. To solve the second deficiency, we apply the string “S1” which corresponds to the behavior sequence S1 to match the string “S2” that maps to the behavior sequence S2. By doing so, sequence S1 is considered as a subsequence of S2 if “S1” is the substring of “S2”, rather than all elements of S1 appear in the S2 by order. In addition, we use List<sequence<element>> as the data structure and set List as all user sequences’ container. Sequence class represents a user's behavior sequence, and encapsulates the operation of the sequence, while the element class represents item sets and encapsulates the operation of item set.

In the counting stage of improved GSP algorithm, for one candidate sequence Ci, this algorithm will scan the user behavior database to get behavior sequence Si for each user, and take the function timeDistance (Ci, Si) to obtain the time interval distance between sequence Si and its candidate sequence Ci. Let’s compare the distance d that function timeDistance (Ci, Si) returned with the threshold Time_Limt that we predefined, if d is less than Time_Limt, we use the improved method to judge whether candidate sequence Ci is the subsequence of a user behavior sequence Si in database. Only simultaneously meet the above two conditions, the candidate sequence Ci will have an increment by adding 1 operation for its support counting. When no new candidate sequences or sequential pattern can be generated, the algorithm stops counting operation. Thus, all the frequent sequences mined will be outputted. Through the above improved GSP algorithm, the deficiencies that include unsatisfactory matching and lack of time factor in practical application in classical GSP algorithm can be solved for our online survey user behavior prediction.

V. EXPERIMENTAL EVALUATION

A. Experimental data set

The experimental data comes from actual business database of a survey company. After pretreatment including data extraction, cleaning and filling, a total of 38,658 relatively complete behavior sequences are selected. These sequences’ length is more than 25. There are some losses of behavioral data in the business database, in order to ensure the integrity of the data and the effectiveness of the experimental results. The time spans from 2013-9-16 to 2014-6-30, since any behavior data get involved in this period. Finally, 30,000 user data is chosen as the training set for the experiment, and the remaining 8658 user data is chosen as the test set for user behavior prediction.

B. Conventional analysis of user behavior sequences

Conventional analysis of user behavior sequence refers to the process that obtains the frequent sequence pattern of user behavior through sequential pattern mining algorithm. Here, we use the maximum frequent sequences [8] to analyze the user behavior, and then provide decision support for the company's business staff. Therefore, in the case of a given support, we can get the set of common behavior sequential patterns of users. Definitely, sequential patterns of user behavior are various when given different supports. In the experiment of conventional analysis of user behavior sequences, the support degree is set as 0.7, time is chosen from 2013-9-16 to 2014-6-30, 20,000 user data is selected randomly from the training set for the algorithm training. Table 3 lists partial results of the experiment, where L (i) refers to i sequential pattern.

Table 3. Conventional analysis of user behavior sequences

Sequential pattern	Support	Analysis
L(7)= <2 2 2 2 2 2 6>	0.7	After 6 successful responses, there is 70% chance for the user to interrupt the answer
L(8)=<1 1 1 1 1 1 1 2>	0.7	After 7 logins, there is 70% chance for the user to have a successful response
L(9)=<4 4 4 4 4 4 4 4 6 >	0.7	After 8 termination response, there is 70% chance for the user to interrupt the answer

According to the above experimental results, we can give some practical applications. For example, by learning from the third sequential pattern in the above table, we should give certain incentives to the user who experiences 8 termination responses, in order to prevent users from discouraging in response, and take the initiative to give up the job.

C. Analysis of user behavior sequence based on the specific behavior and time window

Analysis of user behavior sequence based on the specific behavior and time window refers to get interesting knowledge by observing the changes of the sequence before and after specific behavior within a certain period of time window. Because the influence of specific behavior to other behaviors changes in different periods, two time window periods are chosen, respectively from 2013-9-16 to 2014-2-16 and 2014-2-16 to 2014-6-30. Support is also set as 0.7, while the size of time window is set as 30 days. Additionally, specific behavior is set by exchange prize. The behavior of exchange prize is combined with other user behaviors to observe the experimental results. For example, in order to mine the influence of exchange prize to the activity of the user, we can combine login with exchange prize. 20,000 user data is selected randomly from the training set for the algorithm training. Table 4 and table 5 list partial experimental results, indicating the influence of exchanging prize to other behaviors in deferent periods of time windows.

Table 4. Experimental results of user behavior analysis based on specific behavior and time window (2013-9-16 to 2014-2-16)

Experiment name	Before exchanging prize	After exchanging prize	Analysis
exchange prize + login	<1 1 1 1 1 1 1 1 1>	<1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 >	A strong stimulating effect
exchange prize + successful response	<2 2 2>	<2 2 2 2>	The influence is almost flat
exchange prize + all behaviors	<2 2 2 4 2>, <4 4 4 4 4 4 4>	<2 2 2 4 2> , <4 4 4 4 >	Termination response appears after 3 successful responses, and the total termination response is lessened

Table 5. Experimental results of user behavior analysis based on specific behavior and time window (2014-2-16 to 2014-6-30)

Experiment name	Before exchanging prize	After exchanging prize	Analysis
exchange prize + login	<1 1 1 1 1 1 1 1 1 1>	<1 1 1 1 1 1 1 1 1 1>	The influence is flat
exchange prize + successful response	<2 2 2>	<2 2 2>	The influence is flat
exchange prize + all behaviors	<4 4 4 4 4 2 4 4 4 4 4>, <2 2 2 2 2 2 2>	<4 4 4 4 2 4>,<2 2 2>	Termination response is lessened, but successful response is also lessened

From the experimental results of "exchange prize + login" as shown in Table 4, we can find that exchange prize has a dramatic stimulus to users' active degree from 2013-9-16 to 2014-2-16, while the stimulation influence of exchanging prize to users' active degree from 2014-2-16 to 2014-6-30 is very little. In practical application, the company can change the exchange prize mechanism dynamically according to the experimental results. For example, we can take the awarding motivation between 2013-9-16 and 2014-2-16 to stimulate the activity of users. On the contrary, we should decrease the awarding incentive slightly in the period from 2014-2-16 to 2014-6-30 to reduce the company's operating costs.

D. *User behavior prediction*

User behavior prediction refers to obtaining users' frequent sequence patterns through the improved GSP algorithm. Thus, these sequence patterns will be stored with a certain format in a specific file. In this case, when a behavior sequence is given, the prediction to the next behavior for this active user is made by matching all the sequences patterns in the file. In this experiment, support degree and the interval distance threshold are set as 0.6, and the time period is chosen from 2013-10-1 to 2014-6-30. For the training set, 25,000 user data records whose behavior sequence length is more than 30 have been selected randomly for the classical and improved GSP algorithm. Moreover, 4000 user data records are selected from test set randomly for prediction validation. The experimental results of classical GSP algorithm and improved GSP algorithm are shown in Table 6. Among them, unable to handle number refers to the users' number that can't have a matching prediction operation, because there is not a corresponding behavior sequence in the file that saves the sequence patterns to match the given sequence.

Table 6. Comparison between traditional and improved GSP algorithm for user behavior prediction

Attribute	Classical GSP	Improved GSP
Size of test set	4000	4000
Max sequence pattern length	21	14
Correct prediction number	1661	2423
Unable to handle number	1084	1133
Prediction accuracy rate	41.5%	60.6%

From the experimental results, we can conclude that the prediction accuracy rate for improved GSP algorithm is significantly higher than that of the classical GSP algorithm. However, the maximum sequence pattern length of the improved GSP algorithm is less than that of the classical GSP algorithm. Also, the number of unable sequences to handle is more than that of traditional GSP. That's because condition of improved GSP algorithm is stricter in support counting phase, which causes less sequential patterns than the classical GSP algorithm. In practical application scenario, we often scan user's behavior sequence in recent days to predict users' behavior. Then, we only choose those users whose next behavior is predicted as successful response to send questionnaire. Thus, we can improve user's response rate and decrease company's push cost.

VI. CONCLUSIONS

In this paper, we propose an improved GSP algorithm considering the application deficiencies of the classical GSP algorithm in the analysis of online survey user behaviors. According to the characteristics of online survey user, we construct a user model with part of the important attributes. To ensure the authenticity of behavior sequential patterns and the accuracy of user behavior prediction, in the counting phase of improved GSP algorithm, matching condition is stricter and the processing for the time interval factor is taken into account. Finally, the improved GSP algorithm is applied to the analysis of the online survey user behavior. We conduct extensive experiments, including sequential patterns mining based on the specific behavior and time window, conventional sequential patterns mining and user behavior prediction. From the experimental results, we demonstrate that it is effective to analyze the sequence of user behavior through improved GSP algorithm, by comparing the accuracy rate of user behavior prediction with the existing classical GSP algorithm.

VII. ACKNOWLEDGEMENT

This work was partially supported by a National Natural Science Foundation of China (61303096), a Shanghai Natural Science Foundation (13ZR1454600), an Innovation Program of Shanghai Municipal Education Commission (14YZ017), and a Specialized Research Fund for the Doctoral Program of Higher Education (20133108120029). We also would like to thank all of the anonymous reviewers for your insightful comments and useful suggestions.

REFERENCES

[1] Wu Kongling, Miao Yuqing, Su Jie, Zhang Xiaohua. Sequential pattern mining research. Computer Systems & Applications, 21(6):263-271, 2012.

[2] Agrawal R, Imieliński T, Swami A. Mining association rules between sets of items in large databases. ACM SIGMOD Record. ACM, 1993, 22(2): 207-216.

[3] Srikant R, Agrawal R. Mining sequential patterns: Generalizations and performance improvements. Springer Berlin Heidelberg, 1996.

[4] Zaki J. SPADE: An efficient algorithm for mining frequent sequences. Machine learning, 42(1-2): 31-60, 2001.

[5] Tzvetkov P, Yan X, Han J. TSP: Mining top-k closed sequential patterns. Knowledge and Information Systems, 7(4): 438-457, 2005.

[6] Wang Hu, Ding Shifei. Research and development of sequential pattern mining (SPM). Computer Science, 36(12):14-17, 2009.

[7] Zhang M, Kao B, Yip C, et al. A GSP-based efficient algorithm for mining frequent sequences. In Proc of IC-AI, 2002.

[8] Liu Lijun, Cui Jie, Mei Hongyan Comparison and analysis between algorithm of GSP and PrefixSpan. Journal of Liaoning Institute of Technology, 2006.