ISSN 2278-8808 SJIF 2014 = 4.889

An International Peer Reviewed & Referred

SCHOLARLY RESEARCH JOURNAL
FOR INTERDISCIPLINARY STUDIES

SRJIS

MEASURE PERFORMANCE OFTERTIUS AND SEQUENTIAL PATTERNS FOR SOFTWARE BUG

Dhyan Chandra Yadav

Research scholar: Ph.D program in Computer Science SVU, Gajraula,Amroha (U.P.)

Abstract

Software bug is a major problem in project development process. Association rule presents relationship between attributes and also calculate the intensity of relationship between attributes.In this paper we compare between tertius and sequence patterns performance for bug detection and also describe better performance for rules and number of hypotheses consideration.

Keywords— *Association Algorithms: Tertius , Generalized Sequential Patterns and Weka Tool.*

1. INTRODUCTION

Anthony Williams [1] Duplicity is a technical problem in software project development life cycle. Duplicity is closed as more spaces between codes, change code in another location and error down line. All the technical problems mentation in a report known as problems report .If any bug reported in a problem report but it is already covered by another problem report this happening is known as duplicity of bug. Duplicate bug is created in any phase testing and it is some time automatically created in the coding implementation or phase testing by the help of data mining. It is easily classified and analyzed in the software engineering domain.

2 .RELATED WORK-

Shepperd, Schofield and Kitchenham[2] discussed that need of cost estimation for management and software development organizations and give the idea of prediction also give the methods for estimation.

Alsmadi and Magel[3] discussed that how data mining provide facility in new software project its quality, cost and complexity also build a channel between data mining and software engineering.

Boehm, Clark, Horowitz, Madachy, Shelby and Westland[4] discussed that some software companies suffer from some accuracy problems depend on his data set after prediction software company provide new idea to specify project cost schedule and determine staff time table.

K.Ribu[5] discussed that the need of open source code projects analyzed by prediction and get estimating object oriented software project by case model.

Nagwani and Verma[6] discussed that the prediction of software defect(bug) and duration similar bug and bug average in all software summery, by data mining also discuss about software bug.

Hassan [7] discussed that the complex data source (audio, video, text etc.) need more have buffer for processing it does not support general size and length of buffer.

Li and Reformate[8] discussed that .the software configuration management a system includes documents, software code, status accounting, design model defect tracking and also include revision data.

Elcan[9] discussed that COCOMO model pruned accurate cost estimation and there are many thing about cost estimation because in project development involve more variable so COCOMO measure in term effort and metrics.

Chang and Chu [10] discussed that for discovering pattern of large database and its variables also relation between them by association rule of data mining.

Kotsiantis and Kanellopoulos[11] discussed that high severity defect in software project development and also discussed the pattern provide facility in prediction and associative rule reducing number of pass in database.

Pannurat, N.Kerdprasop and K.Kerdprasop[12] discussed that association rule provide facility the relationship among large dataset as like software project term hug amount , cost record and helpful in process of project development.

Fayyad,PiateskyShapiro,Smuth and Uthurusamy [13] discussed that classification creates a relationship or map between data item and predefined classes.

Shtern and Vassillios[14] discussed that in clustering analysis the similar object placed in the same cluster also sorting attribute into group so that the variation between clusters is maximized relative to variation within clusters.

Runeson and Nyholm[15] discussed that code duplication is a problem which is language independent. It is appear again and again another problem report in software development and duplication arises using neural language with data mining.

Vishal and Gurpreet[16] discussed that data mining analyzing information and research of hidden information from the text in software project development.

Lovedeep and Arti[17]data mining provide a specific platform for software engineering in which many task run easily with best quality and reduce the cost and high profile problems.

Nayak and Qiu[18] discussed that generally time and cost, related problems arises in software project development these problems mentation in problem report ,data mining provide help in to reduce problems also classify and reduce another software related bugs .

In this paper we compare between tertius and sequence patterns performance for bug detection and also describe better performance for rules and number of hypotheses consideration.

3 METHODOLOGY-

Our research approach is present to use some Associationalgorithms. The research methodologyis there divided into 5 steps to achieve the desired results:

Step 1:In this step, prepare the data in addition specify the source of data.

Step 2: In this step select the specific data in addition transform this one into different format by weka.

Step 3: In this step, implement data mining algorithms also checking of all the relevant bugs and errors be present perform.

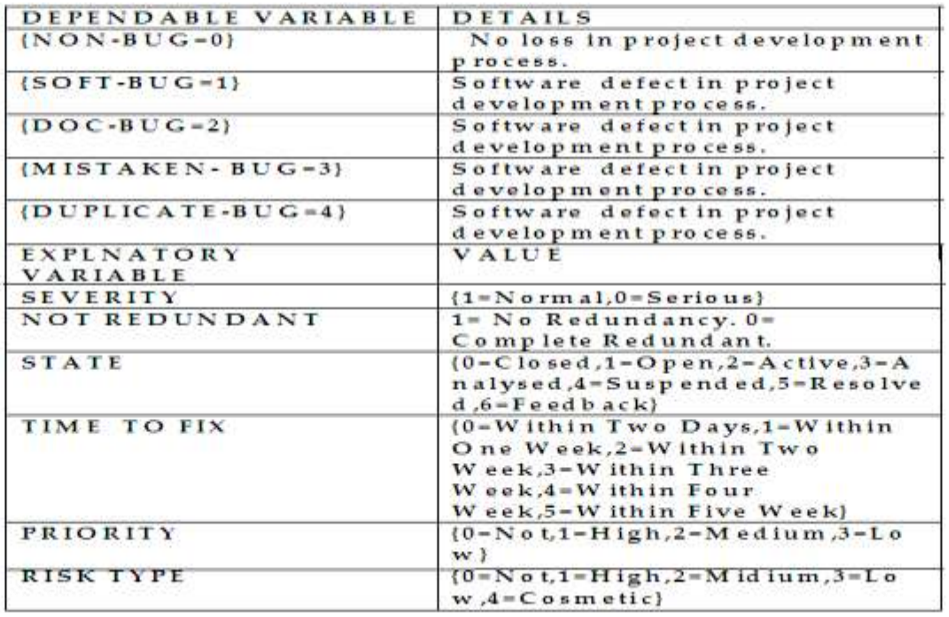
Step 4: We classify the relevant bugs by means ofassociation algorithms by the side of particular time.

Step 5: At the end, the results be located display and evaluated completed.

3.1 DATA PREPARATION-

A software defect tracking system, “GANTS” which be present a bug tracking system in software bug .It be located set on “MASC” intranet in the direction of collect and maintain all problem reports from every department of “MASC”.

TABLE-1 The variables used in the computational technique



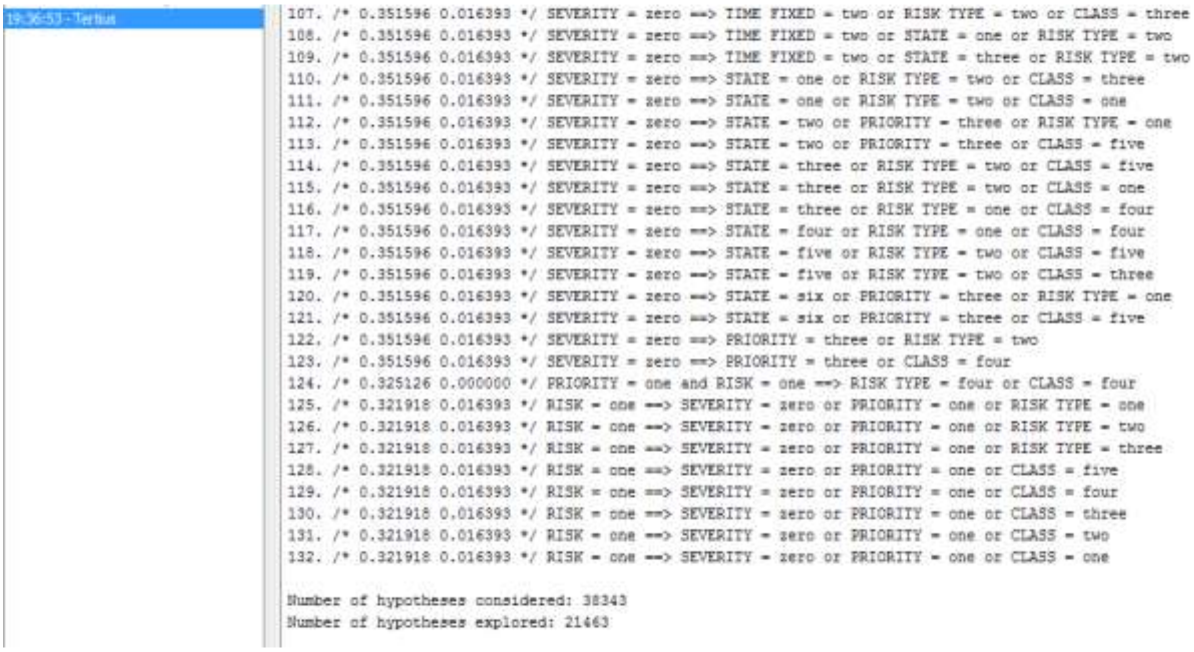
DEPENDABLE VARIABLE	DETAILS
{NON-BUG=0}	No loss in project development process.
{SOFT-BUG=1}	Software defect in project development process.
{DOC-BUG=2}	Software defect in project development process.
{MISTAKEN- BUG=3}	Software defect in project development process.
{DUPLICATE-BUG=4}	Software defect in project development process.
EXPLNATORY VARIABLE	VALUE
SEVERITY	{1=Normal,0=Serious}
NOT REDUNDANT	1= No Redundancy. 0= Complete Redundant.
STATE	{0=Closed,1=Open,2=Active,3=Analysed,4=Suspended,5=Resolved,6=Feedback}
TIME TO FIX	{0=Within Two Days,1=Within One Week,2=Within Two Week,3=Within Three Week,4=Within Four Week,5=Within Five Week}
PRIORITY	{0=Not,1=High,2=Medium,3=Low}
RISK TYPE	{0=Not,1=High,2=Midium,3=Low,4=Cosmetic}

The soft-bug, doc-bug, mistaken-bug and duplicate-bug are present parts of class field in software development. Now performing designed for classification of software defect using several standard algorithms of data mining classifier algorithms. The database is situated designed in “MS-Excel, MS word 2010 database” and database management system on the way to store the collect data.

3.2 DATA SELECTION AND TRANSFORMATION-

In this step only those fields are located selected which is there required for data mining. A few derived variables are present selected. We select some Association algorithms in specific way as:

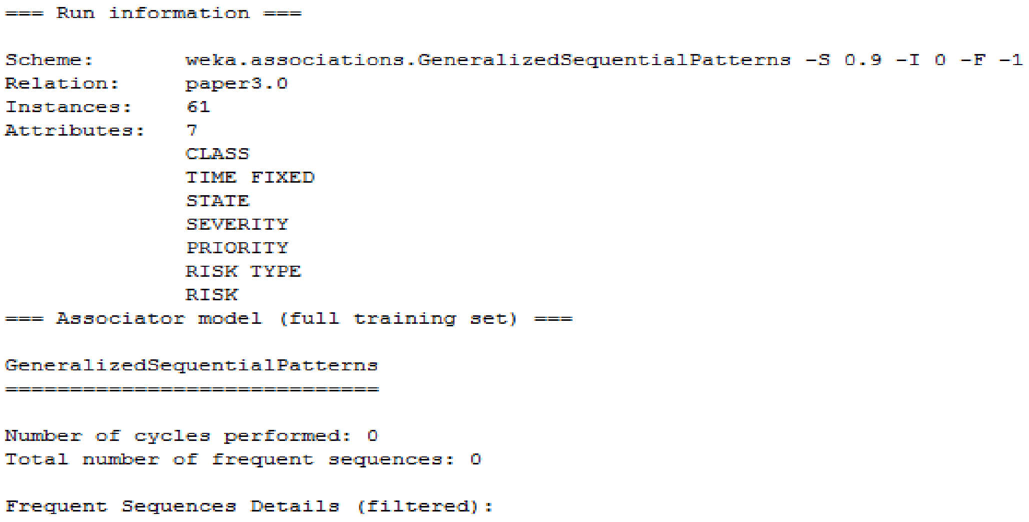
Fig-1. Experimental Setup ForTertius



```
107. /* 0.351596 0.016393 */ SEVERITY = zero ==> TIME FIXED = two or RISK TYPE = two or CLASS = three
108. /* 0.351596 0.016393 */ SEVERITY = zero ==> TIME FIXED = two or STATE = one or RISK TYPE = two
109. /* 0.351596 0.016393 */ SEVERITY = zero ==> TIME FIXED = two or STATE = three or RISK TYPE = two
110. /* 0.351596 0.016393 */ SEVERITY = zero ==> STATE = one or RISK TYPE = two or CLASS = three
111. /* 0.351596 0.016393 */ SEVERITY = zero ==> STATE = one or RISK TYPE = two or CLASS = one
112. /* 0.351596 0.016393 */ SEVERITY = zero ==> STATE = two or PRIORITY = three or RISK TYPE = one
113. /* 0.351596 0.016393 */ SEVERITY = zero ==> STATE = two or PRIORITY = three or CLASS = five
114. /* 0.351596 0.016393 */ SEVERITY = zero ==> STATE = three or RISK TYPE = two or CLASS = five
115. /* 0.351596 0.016393 */ SEVERITY = zero ==> STATE = three or RISK TYPE = two or CLASS = one
116. /* 0.351596 0.016393 */ SEVERITY = zero ==> STATE = three or RISK TYPE = one or CLASS = four
117. /* 0.351596 0.016393 */ SEVERITY = zero ==> STATE = four or RISK TYPE = one or CLASS = four
118. /* 0.351596 0.016393 */ SEVERITY = zero ==> STATE = five or RISK TYPE = two or CLASS = five
119. /* 0.351596 0.016393 */ SEVERITY = zero ==> STATE = five or RISK TYPE = two or CLASS = three
120. /* 0.351596 0.016393 */ SEVERITY = zero ==> STATE = six or PRIORITY = three or RISK TYPE = one
121. /* 0.351596 0.016393 */ SEVERITY = zero ==> STATE = six or PRIORITY = three or CLASS = five
122. /* 0.351596 0.016393 */ SEVERITY = zero ==> PRIORITY = three or RISK TYPE = two
123. /* 0.351596 0.016393 */ SEVERITY = zero ==> PRIORITY = three or CLASS = four
124. /* 0.325126 0.000000 */ PRIORITY = one and RISK = one ==> RISK TYPE = four or CLASS = four
125. /* 0.321918 0.016393 */ RISK = one ==> SEVERITY = zero or PRIORITY = one or RISK TYPE = one
126. /* 0.321918 0.016393 */ RISK = one ==> SEVERITY = zero or PRIORITY = one or RISK TYPE = two
127. /* 0.321918 0.016393 */ RISK = one ==> SEVERITY = zero or PRIORITY = one or RISK TYPE = three
128. /* 0.321918 0.016393 */ RISK = one ==> SEVERITY = zero or PRIORITY = one or CLASS = five
129. /* 0.321918 0.016393 */ RISK = one ==> SEVERITY = zero or PRIORITY = one or CLASS = four
130. /* 0.321918 0.016393 */ RISK = one ==> SEVERITY = zero or PRIORITY = one or CLASS = three
131. /* 0.321918 0.016393 */ RISK = one ==> SEVERITY = zero or PRIORITY = one or CLASS = two
132. /* 0.321918 0.016393 */ RISK = one ==> SEVERITY = zero or PRIORITY = one or CLASS = one

Number of hypotheses considered: 38343
Number of hypotheses explored: 21463
```

Fig-2. Experimental Setup forGeneralized Sequential Patterns



```
=== Run information ===

Scheme:       weka.associations.GeneralizedSequentialPatterns -S 0.9 -I 0 -F -1
Relation:     paper3.0
Instances:    61
Attributes:   7
              CLASS
              TIME FIXED
              STATE
              SEVERITY
              PRIORITY
              RISK TYPE
              RISK
=== Associator model (full training set) ===

GeneralizedSequentialPatterns
=============================

Number of cycles performed: 0
Total number of frequent sequences: 0

Frequent Sequences Details (filtered):
```

3.3 DATA MINING IMPLEMENTATION-

Wekabe presentexposedbasis software that implements a large gathering of machine learning algorithms in addition is extensively used in data mining applications. As of the above data bug.csv file remained created. This file be located loaded into weka explorer also analyzes risk of software defects predicts. Predicts the relationship between attributes.Finally accomplish the best Association algorithms designed for software defect data set.

3.4 RESULT AND DISCUSSION-

There arepresentseveral algorithms designed for association of which the most well-known in addition widely applicable ones be present run taking place the given dataset. The results of each of these runs using weka are provided below.

TABLE .2 To Measure the performance of Tertius , Sequential Patterns

Algorithms	**NO. Rules**	**Number of hypotheses considered**	**Number of hypotheses explored**
Tertius	132	38343	21463
G.Sequetial Patterns	0	Nill	Nill

From table .2 we easily calculate total instances of data set. Given table-.2 shows the comparison between theassociative attribute by weka tool.

table.2 we easily analyzed as:

- In the above table give highest number of rules(132)formeasure relationship between attribute.
- Total highest Number of hypotheses considered (38343).
- The Number of hypotheses explored(21463).
- In the above table it is clear that number of rules,highest Number of hypotheses considered andNumber of hypotheses explored for measure relationship between attribute are totally Nill.

4 CONCLUSION-

In this analysis we choose the **Tertius** algorithmis the best data mining Association algorithms to be applied over selected datasets. Because Generalized Sequential Patterns has Nill values for rules, highest Number of hypotheses considered and Number of hypotheses explored. In this analysis different Association algorithms results are evaluated. So the future work will be based on other Association algorithms that can be applied on the data set and also to apply other data mining tools on the data set such that the best techniques can be identified.

REFERENCES

Anthony Williams "Database Tip: Eliminate Duplicate Data" Friday 25 January 2008.

M. Shepperd, C. Schofield, and B. Kitchenham, "Effort estimation using analogy," in of the 18th International Conference On Software Engineering, pp.170-178. Berlin, Germany, 1996.

Alsmadi and Magel, "Open source evolution Analysis," in proceeding of the 22nd IEEE International Conference on Software Maintenance (ICMS'06), phladelphia, pa.USA, 2006.

Boehm,Clark,Horowitz,Madachy,Shelby and Westland, "Cost models for future software life cycle Process: COCOMO2.0." in Annals of software Engineering special volume on software process and prodocuct measurement, J.D.Arther and S.M.Henry, Eds, vol.1, pp.45-60, j.c.Baltzer AG, science publishers, Amsterdam, The Netherlnds, 1995.

Ribu,Estimating object oriented software projects With use cases,M.S.thesis, University of Oslo Department of informatics, 2001.

N.Nagwani and S. Verma,"prediction data mining Model for software bug estimation using average Weighted similiarity,"In proceeding of advance Computing conference (IACC), 2010.

A.E.Hassan ,"The road ahead for mining software Repositories", in processing of the future of software Maintenance at the 24th IEEE international Conference on software maintenance, 2008.

Z.Li and Reformat,"A practical method for the Software fault prediction",in proceeding of IEEE Nation conference information reuse and Integration (IRI), 2007.

C.Elcan,"The foundations of cost sensitive learning In processing of the 17 International conference on Machine learning, 2001.

C.Chang and C.Chu,"software defect prediction Using international association rule mining", 2009.

S.Kotsiantis and D.Kanellopoulos,"Associanrule mining:Arecentoverview",GESTS international transactiona on computer science and Engineering, 2006.

N.Pannurat,N.Kerdprasop and K.Kerdprasop"Database reverses engineering based On Association rule mining",IJCSI international Journal Of computer science issues 2010.

U.M.Fayyad,GPiateskyShapiro,P.Smuth and R. Uthurusamy ,"Advances in knowledge discovery And data mining",AAAI Press,1996.

M.Shtern and Vassilios,"Review article advances in Software engineering clustering methodologies for Software engineering",Tzerpos volume,2012.

P.Runeson and O.Nyholm,"Detection of duplicate Defect report using neural network processing",inProceeding of the 29th international conference on Software engineering 2007.

G.Vishal and S.L. Gurpreet,"A servey of text mining Techniques and applications",journal of engineering Technologies in web intelligence, 2009.

Lovedeep and VarinderKaurArti" Application of Data mining techniques in software engineering"International journal of electrical,electronics and computer system(IJEECS) Volume-2 issue-5, 6. 2014.

RichiNayakandTianQiu"Adata mining application "international journal of software engineering and Knowledge engineering volume.15,issue-04,2005.