Application of Data Mining Technology in Software Engineering

Xiaobin Hong1,*

1Guangzhou College of Technology and Business, 510000, Guangzhou, Guangdong, China

*Corresponding author e-mail: xiaobin12041@gzgs.edu.cn

Abstract. With the rapid development of informatization, computer database software systems have entered various fields of society, which has brought about the explosive growth of industry data. Faced with massive amounts of data, computers with limited storage capacity have to abandon some outdated data, and the application of various data mining technologies related to it has gradually matured. The purpose of this article is to discuss the application research of data mining technology in software engineering. This article analyzes the correlation analysis of a large number of bug repair source code update data and bug defect reports in the version control system SVN and the defect tracking system Bugzilla in the software engineering project development process, and tries to classify the bug report by data mining technology: defect changes and potential defects change. Starting from large-scale software engineering projects, apply data mining technology to the huge software engineering knowledge base. Especially the software development and maintenance are explained, as well as the more challenging problems in the future. This paper uses data mining technology to study the dependency of the source code files of each module of the software system, and helps software developers quickly understand the software architecture by understanding the interrelationships between the modules, and provides suggestions for modification paths. Experimental research shows that this paper compares with F-measure and concludes that FL-M-GSpan algorithm is better than TS-M-GSpan algorithm. At the same time, it is found that the FL-M-GSpan algorithm always has a better accuracy rate close to 95%, while the TS-M-GSpan algorithm always has a better recall rate.

Keywords: Data Mining, Software Engineering, Software Database, Software Maintenance

1. Introduction

With the advancement of society and the development of science and technology, computer, communication and Internet technologies have penetrated into various industries of society and are changing the way of life of the whole human society [1-2]. The application of various new technologies in the field of computers has allowed various industries to create, collect and store large
amounts of data and the growing amount of data has become an urgent problem for various industries [3-4]. Data mining is an emerging multiple discipline between statistics, database technology, artificial intelligence, machine learning, pattern recognition, data visualization, and high-performance parallel computing[5-6]. It has extremely wide application perspectives [7-8].

In the application of data mining technology to software engineering research, many scholars have studied it and achieved good results. For example, Etani believes that "the knowledge discovery model is an automated or semi-automated process and must be meaningful and bring benefits." The reason why it is called "data mining" is to mine useful information like mining treasures in the huge information materials like mountains [9]. Gu C introduces the data mining ideas and software reconstruction techniques applied in the system, focusing on the mining method of association rules in data mining, and analyzes the common bad smells in the program code [10].

This article uses the decision tree classification method to classify the current system defects and potential defects by associating the source code historical change XML file extracted from the SVN library and the XML file about the bug report generated by the Bugzilla defect tracking system. Then put more energy on the potential defects introduced, discover and repair them in time to reduce the cost for later maintenance.

2. Application research of data mining technology in software engineering

2.1. Application of knowledge base data mining in software maintenance

The larger the scale of the software system and the more data volume, the more accurate value information can be found by data mining technology. This information greatly reduces the workload of software maintenance personnel and reduces the cost of software maintenance.

(1) SVN Repository and Bugzilla.

During the development and testing of the software system, various defects in the software were discovered and put forward in the form of bug reports. After the software developers were repeatedly modified and updated the source code files of the SVN library, the bug reports were only saved. There is no further analysis and use of these bug reports and program source code to update the data information. If you use data mining technology to analyze and research the resolved bug report, the knowledge mined can help software maintainers understand more quickly Current system defect status and potential defects. Therefore, the mining of bug reports can help software maintainers more effectively implement the predictive analysis of potential defects of the software system, find defects early and repair them in time, and reduce the cost of later maintenance of the software system.

(2) Decision tree algorithm.

The attribute minimizes the amount of information required to classify the samples in the result division. Use this attribute to divide the current sample set, so that the "mixing degree of different categories" in each sample subset to be generated is minimized. Therefore, the use of such an information theory method will help effectively reduce the number of times required for object classification, thereby ensuring that the generated decision tree is simpler.

Assuming that $s_i$ is the number of samples in category, then the amount of information required to classify a given data object sample is:

$$I(t_1, t_2, \ldots, t_m) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

(1)

Among them, $P_i$ is the probability of any sample belonging to, and it is calculated using. The entropy, or expected information of dividing the subset according to $A$, is given by:

$$E(A) = \sum_{j=1}^{t} \frac{t_{i_1} + \cdots + t_{i_j}}{t} I(t_{i_j} + \cdots + t_{i_m})$$

(2)

The term $\frac{t_{i_1} + \cdots + t_{i_j}}{t}$ serves as the weight of the jth subset.

(3) Feature attributes and defect classification.

1) Feature attribute selection.
After determining the data set, this article can learn the classification model, and can quickly identify defects and potential defects. Software maintainers can focus more attention on bug reports that potentially introduce defects or discover them in the future maintenance process. Similar knowledge.

2) Research on defect classification.

This article automatically categorizes bug fixes that have been fixed throughout the life of the current software system stored in Bugzilla and lists bug fixes: fixed-Bug is a bug that has been fixed by the software system. Never resubmitted during operation (ReOpen). These errors are the second most important defects in the software maintenance process, but maintenance personnel are required to understand these defects. When similar problems occur in the operation of the system, the source code of the error can be repaired quickly and quickly.

2.2. Software engineering knowledge base data extraction

All manuscripts must be in English, also the table and figure texts, otherwise we cannot publish your paper. Please keep a second copy of your manuscript in your office. When receiving the paper, we assume that the corresponding authors grant us the copyright to use the paper for the book or journal in question. Should authors use tables or figures from other Publications, they must ask the corresponding publishers to grant them the right to publish this material in their paper.

(1) Software engineering version control library.

With the widening and deepening of the scope of computer applications, the scale and complexity of application software have become larger and more complex. This has led to the gradual transformation of software development methods from the early single-soldier combat or manual workshop type to the group and factory Streamlined team collaboration development method. There are some very difficult problems in this development mode:

1) The entire software version needs to be restored to the state at a certain time before;
2) Restrict the program to be modified at will;

(2) Software engineering introduces version control software.

When the first programmer After modification test is complete, put the patch on Chink-in so other developers can choose the same program.

2.3. Application of knowledge base mining in software development

(1) Software architecture understanding process.

This article introduces the software reflection framework method, and integrates the method of this article on the basis of this method-the heuristic method of adding source code annotations to quickly guide developers to understand the system architecture.

(2) Annotate the source code to the program static dependency graph.

The source control system SVN stores the description of the file code line level, but this is not an accurate description level for generating source code annotations. Therefore, this article needs to map source code changes to appropriate source code entities (such as classes, functions, variables, or data types), so as to be able to determine whether the source code changes are caused by adding or deleting dependencies. Then, associate the modified characteristic attributes (developer, modification reason, and date) to the modification dependency relationship between the mapped source code entities.

3. Data mining technology applied experimental research in software engineering

3.1. Evaluation criteria

In order to prove the effectiveness of this method, this article uses traditional information retrieval concepts: recall and accuracy. The change propagation method proposed in this paper is A→B and A→X, which includes the change set of A, which must include B and X. Similarly, this paper obtains B→Y, B→W and C→D. This paper calls the proposed entity set as the prediction set P={B, X, Y, W, D}, and the set that needs to be predicted becomes O={B, C, D}. 

3
F-measure is the weighted average of recall and accuracy, defined as:

$$F_p = \frac{(\beta^2+1) \cdot \text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precision} + \text{Recall}}$$

(3)

Among them $\beta \epsilon (0, \infty)$, $\beta$ is the weight of recall and accuracy.

3.2. Experimental process and data collection

Classify the different types of modification records of the studied software system. This article deletes the general maintenance modification records of the software system and the modification records of adding entities. The remaining source code modification records are the most representative source code changes in the software system life cycle. Historical sample data set. Provide an entity method of association modification based on the frequency of modification records in the XML file. For example, for a changed entity A, find out from the system dependency graph at least two modifications and appear together A% of the time. There are other entities.

4. Data mining technology applied experimental research analysis in software engineering

4.1. FL-M-Gspan algorithm performance analysis

According to the time sequence in the XML file, the entity method of association modification is provided, for example, from the system dependency graph, all entities that have changed with A in the past N days are found. In order to compare the pros and cons of the two heuristic algorithms, after understanding the actual situation of the software system architecture, this article sets $A=\{1,3,5\}$ in the experiment, and the execution results are shown in Table 1.

| Frequency parameter | Accuracy | Recall rate |
|---------------------|----------|-------------|
| 1                   | 0.268    | 0.604       |
| 3                   | 0.327    | 0.548       |
| 5                   | 0.408    | 0.519       |

Table 1. FL-M-GSpan algorithm performance histogram

![Figure 1. FL-M-GSpan algorithm performance histogram](image)
As shown in Figure 1, after experimental comparison and analysis, the FL-M-GSpan algorithm has a better accuracy and recall rate when the frequency value is 5. Study the source code history data in the SVN software version control system, use the graph mining algorithm that finds the software system dependency graph, and then add source code annotations to the dependency graph. Based on these source code annotation dependency charts.

### 4.2. Comparison and analysis of F-measure performance of two algorithms

In order to extract more accurate source code entities from the source code change history XML file, this paper uses two sets of heuristic methods to conduct experiments, β=1.0 two algorithms F-measurement performance comparison results are shown in Table 2.

**Table 2. F-measure performance comparison of two algorithms with β = 1.0**

| Software system | TS-M-GSpan | FL-M-GSpan |
|-----------------|------------|------------|
| GBOMC           | 0.926      | 0.489      |
| GNMC            | 0.831      | 0.452      |
| GBAM            | 0.797      | 0.417      |

![Figure 2](image)

**Figure 2.** Performance comparison of F-measure between two algorithms with β = 1.0

Compared with F-measure as shown in Figure 2, FL-M-GSpan algorithm is better than TS-M-GSpan algorithm. At the same time, this paper also found that the FL-M-GSpan algorithm always has a better accuracy rate close to 95%, while the TS-M-GSpan algorithm always has a better recall rate.

### 5. Conclusion

This article uses the Java-RESC-generated XML file as a data source, uses the correlation dependency chart extraction algorithm to extract the software system class operation diagram, and then adds source code annotations (developer, date, reason, etc.) in the relevant calling relationship. This way, software developers can quickly understand the system architecture and related modification of system source code files based on the function call dependency graph added with source code annotations. In terms of the application of data mining technology to the life cycle of software engineering, the source code extraction level needs to be more detailed. With the continuous evolution of the system, dynamic operation dependency graphs are created and the problems of functions and variables of the same name and different types with the same name are solved.
References

[1] Miandoab E E, Gharehchopogh F S. A Novel Hybrid Algorithm for Software Cost Estimation Based on Cuckoo Optimization and K-Nearest Neighbors Algorithms[J]. Engineering, 2016, 6(3):1018-1022.

[2] Li Z. A research based on NCURD clustering algorithm for data mining in software engineering[J]. Boletin Tecnico/Technical Bulletin, 2017, 55(14):102-109.

[3] Abdi Y, Parsa S, Seyfari Y. A hybrid one-class rule learning approach based on swarm intelligence for software fault prediction[J]. Innovations in Systems & Software Engineering, 2015, 14(4):1-13.

[4] Li Z. A research based on NCURD clustering algorithm for data mining in software engineering[J]. Boletin Tecnico/Technical Bulletin, 2017, 55(14):102-109.

[5] Huang S, Hou B, Chang J, et al. Spatial-temporal change in precipitation patterns based on the cloud model across the Wei River Basin, China[J]. Theoretical & Applied Climatology, 2015 120(1-2):391-401.

[6] Abdi Y, Parsa S, Seyfari Y. A hybrid one-class rule learning approach based on swarm intelligence for software fault prediction[J]. Innovations in Systems & Software Engineering, 2015, 14(4):1-13.

[7] Seo J H, Park N H, Joo K H. A Study on the Decision-Making of Effective S/W Education based on Opinion Mining Analysis[J]. International Journal of Engineering & Technology, 2018, 7(4.16):5-9.

[8] Gao Q, Zhang F L, Wang R J. Mining Frequent Sets Using Fuzzy Multiple-Level Association Rules[J]. Journal of Electronic Science and Technology, 2018, 16(02):51-58.

[9] Etani, Noriko. Database application model and its service for drug discovery in Model-driven architecture[J]. Journal Of Big Data, 2015, 2(1):1-17.

[10] Gu C, Liu Z, Fang L. Big Data Mining Analysis of Key Indicators of Online New Energy Vehicles[J]. Journal of Physics: Conference Series, 2020, 1631(1):012102 (7pp).