

Criteria for comparing Aspect mining Techniques

Anil Kumar Pandey,

Computer Centre

Banaras Hindu University

akpandey@bhu.ac.in

Manjari Gupta

DST Centre of Interdisciplinary Sciences,

Banaras Hindu University

Manjari_gupta@rediffmail.com

Introduction:

It is very difficult and very complex task for designing and implementing software system. The development of software and evaluation with comparison of the techniques used in determination of crosses cutting concern. The concerns are divided into two part i.e. core concern and cross cutting concern. The tendency of core concern in developed software tools is to capture from central functionality of module and cross-cutting concern from system level. In any enterprise application, there are a number of concerns which need to be taken care of in addition to the main business logic. These concerns are spread across the application and into multiple application layers. Such concerns are logging, transaction handling, performance monitoring, security etc. These concerns are knows as cross cutting concerns of the application. T Tourwe et al [14] has given aspect aspectual views using formal concept analysis. The idea of P. Tonella et al [15] regarding formal concept analysis of execution Traces. The type of cross cutting concern have discovered through JHotDraw in a case study and comparison with other techniques. M. Ceccato et al[8] has applied and combined the three aspect mining techniques for software quality control using recalled method and seed quality as evaluation measure. A Kellens et al [11] has proposed his idea for survey of automated code- level aspect mining techniques.

2. Criteria for Comparison

A set of new comparison criteria will be discussed with their importance in comparison of aspect mining techniques.

2.1 Input Data Requirement

In the process of comparison of aspect mining techniques the role of input data is quite important. S. Beru et al [1] has projected his idea regarding aspect mining applying the norms of event traces. While M. Martin et al[17] floated an idea and used Fan-in Analysis to identify the Aspects.

Particulars	Kinds of Input Data	
	Static	Dynamic
Execution pattern	X	X
Dynamic Analysis	-	X
Identification Analysis	X	-
Language Clues	X	-
Unique Methods	X	X
Method Clustering	X	-
Call Clustering	X	X
Fan-in-Analysis	X	X
Clone Detection(PGD)	X	X
Clone Detection(token)	X	-
Clone Detection(AST)	X	X

2.1.a. Source Code: Many techniques require source code as input of the system which provide the system within legacy. The mining of source code is very common in all techniques. Tuourvwe and Mens [22] developed tools for the analysis of Source code of result was that interesting concerns in the source code are reflected by the using naming conventions in the class. Shepherd et al (2005) proposed the techniques that are based on assumption that cross cutting concern as natural language processing on source code.

2.1.b. CVS Transaction:- The identification of aspects is based on co addition: the same set of calls is added in many locations. We compute all co- additions of a transaction efficiently using concept analysis and identify the most likely aspects. We leverage this observation to mine aspect fro CVS repositories, because transaction are typically small and we mine one transaction at a time, our mining scales well defined. Software may contain functionality that does not align with its architecture. Such cross-cutting concerns do not exist from the beginning but emerge over time. By analyzing where developers add code to a program, our history-based mining identifies cross-cutting concerns in a two-step process. First, we mine CVS archives for sets of methods where a call to a specific single method was added. In a second step, such simple cross-cutting concerns are combined to complex cross-cutting concerns. To compute these efficiently, we apply formal concept analysis.

Language Dependency:

As for as theory is concern aspect mining techniques is independent but when tool is associate in techniques are dependent on language.(Fint et al 2006) described the techniques the language is only based on Java. The work of A. Sampaio etal(2005) and in the process of Wmatic corpus analysis it was found that the aspect mining technique is only dependent on English neither French and Italian whose requirement written in these.

Availability:

In comparison of aspect mining techniques the availability of tools, results and cse study play a vital role for new researchers. It was noted form various papers that aspect mining techniques have an associated tool. In non- availability of associated tools which are not publically available the new user first tries to build a tool. The tool availability is also

important when the aspect mining techniques compared with new case study. Few tools are already available i.e. FINT for Fan-In, Dynamo for Execution and WMATRIX for EA-Miner respectively. The developers of other aspect mining techniques can use them in comparison and evolution. The availability of case study is also very important in comparison of Aspect Mining techniques. Without unavailability of case study publically the comparison with other techniques are not possible.

Evaluation Measure:

Measures play an important role when we approach for comparing the aspect mining techniques. Only two statistical measures especially applicable (the number of false positives and the number false negatives). Evaluation measures are needed in process of comparison of aspect mining techniques. Kellen et al (2007) have proposed the six aspect mining techniques evaluation measures

Conclusion:

In this paper criteria for comparison aspect mining techniques are presented and it should be as under:

1. In the process of comparison aspect mining techniques the role of input data is quite important.
2. Aspect mining techniques is independent but when tool is associated in technical part the aspect mining techniques are dependable on language.
3. Evaluation measures are needed in process of comparison of aspect mining

REFERENCES:

1. S. Breu. *Aspect Mining Using Event Races*. Master's thesis, University of Passau, Germany, March 2004.
2. S. Breu and J Krinke. *Aspect Mining using Event Traces*. In *Proceeding of International Conference on Automated Software Engineering*, pages 310-315, 2004
3. S. Breu and T. Zimmermann. *Mining aspects from History*. In *Proceedings of 21st IEE/ACM International Conference on Automated Software Engineering*, pages 221-230, September 2006.
4. S. Breu, T. Zimmermann, and C. Lindig. *Mining Eclipse for Crosscutting Concerns*. In *Proceeding of the 2006 International Workshop on Mining Software Repositories (MSR'06)*, pages 94-97, New York, NY, USA, 2006. ACM Press.
5. M. Bruntik, A. Van Deursen, T. Tourw e, R. Van Engelen. *An Evaluation of Clone Detection Techniques for Identifying Crosscutting Concerns*. In *ICSM'04: Proceedings of the 20th IEEE International Conference on Software Maintenance*, pages 200-209, Washington, DC, USA, 2004 IEEE Computer Society.

6. M. Bruntik, A. Van Deursen, T. Tourw e, *On the use of clone detectionfor identifying crosscutting concern code. IEEE Transactions on Software Engineering, 31(10):804-818,2005.*
7. M. Ceccato, M Marin, K. Mens, L. Moonen, P Tonelaa, T. Tourwé. *A Qualitative Comparison of Three Aspect Mining Techniques. In IWPC '05"Proceedings of the 13th International Workshop on Program Comprehension, pages13-22. IEEE Computer Society, 2005.*
8. M. Ceccato, M Marin, K. Mens, L. Moonen, P Tonelaa, T. Tourwé. *Applying and Combining Three Different Aspect Mining Techniques. Software Quality Control, 14(3):209-231, 2006.*
9. *Dynamo-Dynamic Aspect Mining Tool. <http://star.itc.it/dynamo/>.*
10. E. Gamma. *JHotDraw Project. <http://sourceforge.net/projects/jhotdraw>.*
11. A Kellens, Mens, and P. Tonella. *A Survey of Automated Code-level Aspect Mining Techniques. Transactions on Aspect-Oriented Software Development, Special Issue on Software Evolution, 2007, to appear.*
12. J Krike. *Mining Control Flow Graphs for Crosscutting Concerns. In Proceedings of the 13th Working Conference on Reverse Engineering (WCRE 2006), pages334-342, Washington, DC, USA, 2006, IEEE Computer Society.*
13. J. Krinke and S. Breu. *Control-Flow-Graph-Based Aspect Mining. In Workshop on Aspect Reverse Engineering(WARE),2004.*
14. C.D. Manning and H.Sch'utze. *Foundations of Statistical Naturla Language Processing. The MIT Press, Cambridge, Massachusetts, 1999.*
15. M. Marin, L. Moonen and A. Van Deursen. *A common framework for aspect mining based on crosscutting concern sorts. In WCRE'06: Proceedings of the 13th Working Conference on Reverse Engineering (WCRE 2006), pages 29-38, Washington, DC, USA, 2006, IEEE Computer Society.*
16. M. Marin, L. Moonen and A. Van Deursen. *FINT: Tool Support for Aspect Mining In WCRE'06: Proceeding of the 13th Working Conference on Reverse Engineering (WCRE 2006), pages 299-300, Washington, DC, USA, 2006, IEEE Computer Society.*
17. M. Marin, L. Moonen and A. Van Deursen. *Identifying Aspects Using Fan-in Analysis. In Proceeding of the 11th Working Conference on Reverse Engineering (WCRE 2004), pages 132-141. IEEE Computer Society, 2004.*
18. B. Nora, G Said and A Fadila. *A comparative classification of aspect mining approaches. Journal of Computer Science,2(4):322-325,2006.*
19. A. Sampaio, R. Chitchyan, A Rashid, and P. Rayson. *Ea-miner: a Tool for Automating Aspect-Oriented Requirements Identification. IN ASE '05: Proceedings of the 20th IEEE/ACM international Conference on Automated software engineering, pages 352-355, New York, NY, USA, 2005. ACM Press.*
20. A. Sampaio, R. Chitchyan, A Rashid, and P. Rayson. *Mining Aspect in Requirements. In Early Aspects 2005: Aspect-Oriented Requirements Engineering and Architecture Design Workshop(help with AOSE 2005), Chicago, Illinois, USA,2005.*

21. *P. Tonella and M. Ceccato. Aspect Mining through the Formal Concept Analysis of Execution Traces. In Proceedings of the 11th Working Conference on Reverse Engineering (WCRE'04), pages112-121, Washington, DC, USA, 2004. IEEE Computer Society.*
22. *T. Tourw's and K. Mens. Mining Aspectual Views using Formal Concept Analysis. In Proc. IEEE International Workshop on Source Code Analysis and Manipulation, 2004.*
23. *Wmatrix corpus analysis and comparison tool. <http://www.comp.lancs.ac.uk/ucrel/wmatrix/>.*