Miriam Davydov

Marlyn Momo Fopossi

Maxime Desmarais

Médoune Yade

*Analyse de programmes* INF1018

Partie recherche du projet de session : Aspect mining

Travail présenté à

M. Mahmoud Mohanna



Département de mathématiques et d'informatique

***Tables des matières :***

1. ***Introduction***
2. ***Aspect mining***
3. **Aspect mining supervisée**
4. **Aspect mining non supervisée**
5. **Aspect mining semi-supervisée**
6. **Aspect mining basée sur le lexique**
7. **Aspect mining basée sur la syntaxe**
8. ***Quelques techniques et approches***
9. ***Analyse des modèles récurrents des traces d'exécution***
10. ***Analyse conceptuelle formelle des traces d'exécution***
11. ***Analyse formelle des concepts des noms de classes et de méthodes***
12. ***Analyse des traces d'exécution avec un algorithme de webmining***
13. ***Conclusion***
14. ***Bibliographie***
15. ***Introduction :***

L’«*aspect mining »* tente d’identifier et de résoudre les préoccupations transverses dans le contenu d’un programme. Il peut également fournir des informations permettant de classer les aspects communs qui se produisent dans différents types de contenu, tels que les actualités et les données sociales.

Après presque dix ans depuis sa conception d’origine, cette technologie fait son entrée dans l’industrie ce qui fait émerger de nouveaux problèmes. L’étude et le développement d’approches pouvant faciliter la migration de systèmes ou encore l’application de cette technique sur du code *legacy* est l'objectif des domaines de recherche émergents de l'*« aspect mining »* et *« aspect refactoring »*.

Notre recherche se portera sur l’application de l’*« aspect mining »* sur l’analyse de programme dans un contexte globale ainsi que dans un contexte de *« Data mining* *»*.

1. ***Aspect mining***

Le *« Data mining* *»* est le processus de recherche d'anomalies, de modèles et de corrélations dans de grands ensembles de données pour prédire les résultats. Ce qui était ancien redevient nouveau, car la technologie du *« Data mining* *»* ne cesse d'évoluer pour suivre le rythme du potentiel illimité du Big Data et de la puissance de calcul maintenant abordable. Le processus de migration d'un système existant vers un système utilisant des aspects consiste en deux étapes : l'identification des aspects candidats et le remaniement de ces candidats en aspects. (Sas, 2022)

Cela implique l'utilisation d'algorithmes et de techniques sophistiqués pour extraire des informations précieuses à partir de grands ensembles de données. Ces informations peuvent être utilisées pour prendre des décisions, optimiser les processus et développer de nouveaux produits et services.

Dans un système implémenté à l'aide de techniques classiques, il existe certaines préoccupations qui ne peuvent pas être localisées à l'aide des mécanismes de modularisation disponibles, mais qui traversent le code source du système. En conséquence de l'existence de ces préoccupations transversales, l'implémentation de certaines préoccupations est caractérisée par la duplication du code, la dispersion de la préoccupation dans l'ensemble du système et l'enchevêtrement du code spécifique de la préoccupation avec celui d'autres préoccupations, ce qui rend plus difficile la compréhension, la maintenance et l'évolution du système. (RLa03).

L’«*aspect mining »*  est une sous-catégorie du *« Data mining* *»* qui se concentre spécifiquement sur la compréhension des opinions et des sentiments des données textuelles.

* 1. **Aspect mining supervisée :**

Les techniques d'exploration d'aspect supervisée utilisent des données étiquetées pour identifier les aspects du texte. Cela peut inclure des techniques d'apprentissage automatique supervisées telles que l'arbre de décision, la machine à vecteurs de support et la régression logistique.

* 1. **Aspect mining non supervisée :**

Les techniques d'exploration d'aspects non supervisées utilisent des données non étiquetées pour identifier les aspects du texte. Cela peut inclure des techniques d'apprentissage automatique non supervisées telles que le clustering k-means, le clustering hiérarchique et l'allocation Dirichlet latente.

* 1. **Aspect mining semi-supervisée :**

Les techniques d'exploration d'aspects semi-supervisée combinent des données étiquetées et non étiquetées pour identifier les aspects du texte. Cela peut inclure des techniques d'apprentissage automatique semi-supervisées telles que la machine à vecteurs de support semi-supervisé et la régression logistique semi-supervisée.

* 1. **Aspect mining basée sur le lexique** :

Les techniques d'exploration d'aspects basées sur le lexique reposent sur la connaissance de la langue et des lexiques pour identifier les aspects du texte. Cela peut inclure des techniques telles que la reconnaissance d'entités nommées, le marquage des parties du discours et l'analyse des sentiments.

* 1. **Aspect mining basée sur la syntaxe :**

Les techniques d'exploration d'aspects basées sur la syntaxe reposent sur la structure syntaxique du texte pour identifier les aspects du texte. Cela peut inclure des techniques telles que

1. ***Quelques techniques et approches :***

**Navigateurs dédiés :** Se munir de navigateurs de code avancé pour naviguer manuellement dans le code source pour explorer les préoccupations transversales afin de maintenir et de faire évoluer le système. Le navigateur propose d'autres points chauds dans le code qui pourraient être liés à la préoccupation ou fournit à l'utilisateur un langage de requête pour parcourir manuellement la préoccupation. (MPR)

**Identification (semi-)automatique des aspects candidats:** Pour automatiser le processus d’identification des aspects et de d’obtention de proposition d’aspects candidats, il existe des techniques qui agissent sur le code source du système ou sur les données qui en émanent. La recherche des symptômes de préoccupations transversales s’associe à des techniques issues de l’exploration et de l’analyse des données comme l’analyse formelle de concepts et l’analyse de clusters.

Mais aussi à des techniques plus classiques d’analyse de code comme le découpage de programmes, les métriques et les heuristiques logicielles, la da détection de clones et les techniques de comparaison de motifs et l’analyse dynamique. Ils aident de manière semi-automatique un développeur à identifier les problèmes transversaux dans un système existant.

1. ***Approches par analyse des modèles récurrents des traces d'exécution***

Les modèles récurrents dans les traces d'exécution peuvent révéler des informations utiles sur la façon dont les programmes sont utilisés, la fréquence à laquelle certaines fonctions sont appelées et les opérations qui consomment le plus de ressources. Cela peut être utilisé pour identifier les goulots d'étranglement, optimiser les performances et trouver des zones de code qui pourraient bénéficier d'une refactorisation.

L'analyse des modèles récurrents permet également d'identifier des scénarios d'utilisation courants, qui peuvent être utilisés pour créer des interfaces utilisateur plus efficaces et améliorer l'expérience utilisateur globale. De plus, ce type d'analyse peut être utilisé pour détecter les erreurs et les problèmes de sécurité, ainsi que pour identifier les comportements malveillants.

Pour analyser les traces de programme reflétant le comportement d'exécution d'un système à la recherche de modèles d'exécution récurrents, Breu Silvia et Krinke Jens ont introduit la notion de relations d’exécution entre les invocations de méthodes.

À l'aide de ces relations d'exécution, leur algorithme d'exploration identifie les aspects candidats sur la base de modèles récurrents d'invocations de méthodes. Si une relation d'exécution apparaît plus d'une fois et se répète uniformément, elle est considérée comme un aspect candidat. Bien entendu, pour s'assurer que les aspects candidats sont suffisamment transversaux, il faut en outre que les relations récurrentes apparaissent dans des "contextes d'appel" différents. (SBr)

Pour calculer les relations d’exécution, l’expérience a été répétée en utilisant des graphes de flux de contrôle.

Afin de lever les ambigüités et d’améliorer les résultats de la technique, une approche où les informations dynamiques ont été complétées par des informations statiques.

1. ***Approches par analyse conceptuelle formelle des traces d'exécution***

L'analyse de concept formelle (FCA) est une technique utilisée pour analyser des ensembles de données et déterminer la structure organisationnelle en leur sein. Elle peut s'appliquer aux traces d'exécution, qui sont des enregistrements de la séquence d'instructions qu'un programme informatique a exécutées. FCA peut être utilisé pour identifier des modèles dans la séquence d'instructions, tels que des régularités ou des anomalies, qui peuvent être utiles pour le débogage, l'optimisation ou la rétro-ingénierie d'un programme. Cela peut aider à identifier les problèmes potentiels et à améliorer les performances globales du programme. De plus, FCA peut fournir un aperçu de la structure d'un programme, ce qui peut aider à mieux comprendre la logique sous-jacente et à comprendre le comportement du programme.

1. ***Analyse formelle des concepts des noms de classes et de méthodes***

L'analyse de concept formelle est un type de technique d'analyse de données utilisée pour identifier et catégoriser des modèles dans de grands ensembles de données. Il peut être utilisé pour analyser les noms de classes et de méthodes afin de mieux comprendre la structure d'un programme. Cela peut être utile pour identifier les zones de code qui sont difficiles à comprendre ou qui présentent un potentiel d'amélioration. L'analyse peut également aider à identifier les conflits potentiels entre les classes ou les méthodes, ainsi que les éventuelles redondances. L'analyse peut également aider à identifier les zones du code qui peuvent nécessiter une documentation ou des explications plus détaillées.

1. ***Approches par analyse des traces d'exécution avec un algorithme de webmining***

Les algorithmes d'exploration Web peuvent être utilisés pour analyser les traces d'exécution afin de découvrir des modèles et des tendances dans le comportement des utilisateurs. Ce faisant, l'algorithme peut être utilisé pour identifier les zones d'amélioration potentielles ou les zones à risque, telles que les risques de sécurité. L'algorithme peut également être utilisé pour identifier les domaines d'optimisation potentielle et pour fournir des informations sur la manière dont les utilisateurs interagissent avec le système.

L'algorithme de web mining HITS (HITS) analyse les liens hypertextes dans un certain nombre de pages Web et identifie les hubs (pages qui font référence à de nombreuses autres pages) et les autorités (pages qui sont référencées à partir d'un grand nombre d'endroits). Appliqué à des informations dynamiques obtenues en exécutant (une partie) du système logiciel dans lequel on veut identifier les classes les plus importantes. On obtient une collection de classes du système qui sont étroitement couplées.

1. ***Conclusion***

L’«*aspect mining »* est le processus d'extraction automatique d'aspects à partir de données textuelles. Il est utilisé pour identifier les sujets d'intérêt à partir d'un ensemble de documents. Il est couramment utilisé dans des domaines tels que le marketing, la détection des fraudes et la gestion de la relation client. Le processus d'analyse de la dynamique d'un système en suivant son comportement dans le temps, aide à identifier les tendances, les corrélations et les modèles de comportement afin de prendre des décisions plus éclairées. En combinant ces deux techniques, il est possible de découvrir les relations entre les différents aspects d'un système, d'identifier les sujets qui sont liés les uns aux autres et de mieux comprendre comment le système évolue au fil du temps.

Les progrès récents de la technologie informatique ont facilité *« Data mining* *»* sur de grands ensembles de données. À l'aide d'algorithmes puissants, les ordinateurs peuvent désormais parcourir rapidement de grands ensembles de données et identifier des modèles et des relations entre différents aspects. Cela a permis aux chercheurs de faire des prédictions plus précises et de mieux comprendre la dynamique des différents ensembles de données. De plus, les progrès de l'intelligence artificielle ont permis aux ordinateurs d'apprendre à partir de grands ensembles de données et de prendre de meilleures décisions dans diverses applications.

# Bibliographie

HITS. (s.d.). *J. Kleinberg. Authoritative sources in a hyperlinked environment ". Jour- nal of the ACM, 46(5):604-632, 1999.*

MPR. (s.d.). *M. P. Robillard et G. C. Murphy. Concern graphs : Fnding and describing concerns using structural program dependencies. In Pro- ceedings of the 24th International Conference on Software engineering, pages 406-416. ACM Press, 2002.*

RLa03. (s.d.). *R. Laddad. AspectJ en action : Practical Aspect-Oriented Programming. Manning Publications, 2003.*

Sas. (2022). *Data mining*. Récupéré sur Sas: https://www.sas.com/en\_us/insights/analytics/data-mining.html#:~:text=Data%20mining%20is%20the%20process,relationships%2C%20reduce%20risks%20and%20more.

SBr. (s.d.). *S. Breu et J. Krinke. Aspect mining using event traces. Dans la conférence sur le génie logiciel automatisé (ASE), septembre 2004.*