

Prof. dr hab. Inż. Alicja Wakulicz – Deja
Instytut Informatyki
Wydział Informatyki i Nauki o Materiałach
Uniwersytet Śląski



Tytuł rozprawy: Automatyczna identyfikacja nieprawidłowych zmian w systemach integracji ciągłej z wykorzystaniem wnioskowania w warunkach niepewności.

Automated Identification of Breaking Changes in Continuous Integration Systems Using Reasoning Under Uncertainty

Autor rozprawy: mgr inż. Stanisław Świerc

Promotor rozprawy: prof. dr hab. inż. Krzysztof Cyran

RAU	Biuro Dziekana
	Wpłynęło dnia <u>13.10.2015</u>
	Nr <u>20</u> / zał.

Dziedzina: nauki techniczne

Dyscyplina: informatyka

Praca składa się z siedmiu rozdziałów. Rozpoczyna ją spis treści, spis ilustracji i tabel. Pracę zamyka bibliografia licząca 80 pozycji.

W rozdziale 1 autor sugeruje, że wprowadzenie metod inteligentnych w inżynierii oprogramowania stwarza szanse na podejmowanie bardziej kompetentnych decyzji w procesie projektowania oprogramowania. Następnie omawia praktykę Integracji Ciągłej stosowanej w inżynierii oprogramowania oraz praktykę korekty wstecznej. Autor stawia tezę możliwości stworzenia systemu ekspertowego, który pozwoli na:

- zdolność diagnozowania defektów w trakcie integracji kodu wynikowego systemu,
- zdolność do automatycznego naprawiania defektów poprzez wycofywanie zestawów zmian, które spowodowały defekty w kodzie źródłowym projektu.

Autor zamierza udowodnić postawioną hipotezę poprzez zaprojektowanie oprogramowania w postaci autonomicznego agenta, wyposażonego w zdolności (a) i (b), i wykazanie jego użyteczności w środowisku systemu CI.

W rozdziale 2 autor w sposób systematyczny i dokładny prezentuje koncepcję CI. Rozdział ma charakter przeglądowy, autor systematycznie wskazuje wykorzystywane źródła, nie stroniąc jednak od własnych opinii oraz przytaczając różne punkty widzenia.

Rozdział 2 dowodzi, że autor bardzo dobrze rozumie opisywaną problematykę, jest świadomy różnych aspektów zastosowania podejścia CI, nie tylko w wymiarze technicznym ale również w wymiarze socjologii programistycznych grup roboczych. Autor wykazał się umiejętnością krytycznej analizy źródeł, materiały źródłowe cytowane są poprawny sposób, co istotne, autor nie stroni od własnych opinii. Szerokie nakreślenie tematyki CI, jakże wartościowe w sensie ogólnego poznania, stanowić może jednak swoistą wadę, jeżeli rozważamy je w kontekście doktorskiej dysertacji. Przedstawione rozważania nie formułują jasno rozważanego problemu badawczego, ani nie dostarczają informacji na temat proponowanych przez autora naukowych metod rozwiązywania problemów.

W tym kontekście rozdział 2 stanowi bardziej dobrze opracowany fragment książki z inżynierii oprogramowania, niż element rozprawy doktorskiej.

Rozdział 3 zawiera krótkie wprowadzenie dotyczące wiedzy niepewnej oraz wnioskowania w warunkach niepewności. W podrozdziale 3.1 dokonany jest krótki przegląd prac związanych z przetwarzaniem wiedzy niepewnej, autor krótko wspomina o współczynnikach pewności CF, zbiorach przybliżonych, zbiorach rozmytych, teorii Dempstera-Shafera, następnie wskazuje wybrane prace związane z wykorzystaniem sieci Bayesa.

W kolejnych podrozdziałach szczegółowo omówiono sieć Bayesa jako reprezentację złożonych systemów ekspertowych. Nawiązuje też do możliwości wykorzystania sieci Bayesa w systemach diagnozowania uszkodzeń.

Rozdział 3 jest relatywnie krótki, momentami zanadto zwięzły. Ponieważ do tego momentu rozprawa nie precyzuje jeszcze dokładnie problemu badawczego, trudno stwierdzić, czy zawartość rozdziału jest wystarczająca i adekwatna. Mimo iż autor uzasadnia konieczność wykorzystania wiedzy niepewnej, wybór podejścia bayesowskiego nie poprzedza analiza innych podejść, autor uzasadnia wykorzystanie modelu Bayesa jego popularnością oraz zastosowaniem w komercyjnych systemach diagnozowania uszkodzeń.

Rozdział 4 poświęcony jest opisowi zbioru z danymi, które reprezentują informacje o nieudanych przebiegach procesu integracji kodu wynikowego w ramach CI. Zbiór ten ma zawierać zarówno dane dla generowania modelu bayesowskiego, jak i dane pozwalające na jego ocenę. Autor szczegółowo analizuje jakie dane są konieczne, czego dotyczą, w jaki sposób i skąd je pozyskać (podrozdział 4.1). Zaskakuje nieco fakt, że nie został podany szczegółowy, fizyczny format reprezentacji danych.

Podrozdział 4.2 prezentuje scenariusze opisujące zakończone porażką przebiegi integracji systemu. Autor prezentuje stosowne diagramy sekwencji dla progresywnej jak i wstecznej korekty stanu projektu, wskazując również na niejednoznaczności jakie mogą wystąpić w przypadku korekty wstecznej.

Podrozdział 4.3 prezentuje koncepcję poprawy jakości danych, tak aby stanowiły one właściwy zestaw danych treningowych. Proponowane metody poprawy jakości danych zawierają m.in. propozycję wykrywania danych odstających.

Wydaje się, że to właśnie podrozdziały 4.2. i 4.3 zawierają kluczowe informacje charakteryzujące analizowany problem dziedzinowy. Tutaj właśnie, w sposób konkretny przedstawione zostały problemy wstecznej korekty stanu systemu, która w myśl tezy pracy, ma podlegać procesowi automatyzacji z wykorzystaniem modelu bayesowskiego. Dziwi nieco umieszczenie tych kluczowych informacji tak późno, w rozdziale, wg tytułu, poświęconym opisowi zbioru danych.

W rozdziale 5 autor określa wymagania dla proponowanego modelu diagnostycznego. Wskazuje, że powinien być to system uczący się uwzględniający wiedzę dziedzinową ekspertów, związaną z błędnie zakończonymi integracjami systemów.

Następnie autor stwierdza, że jedynym satysfakcjonującym rozwiązaniem jest opisywane wcześniej podejście bayesowskie. Niestety, odrzucenie wspomnianych systemów regułowych i drzew decyzyjnych nie znajduje żadnego głębszego uzasadnienia, podobnie jak wskazanie podejścia bayesowskiego jako bezsprzecznie zadawalającego.

Autor przedstawia proces tworzenia grafu, który pokazuje złożony proces integracji kodu wynikowego systemu. Wprowadza pojęcie celu jako kolekcji plików tworzonych w procesie integracji kodu wynikowego. Wierzchołkami grafów są cele a krawędzie modelują warunki pozwalające na osiągnięcie kolejnych celów. W grafie pamiętane są nieosiągnięte cele, które emulują wiedzę eksperta w zaproponowanym modelu sieci Bayesa.

Kolejne rozdziały opisują proces wnioskowania w zaproponowanej sieci, proces uczenia sieci oraz pozyskiwanie wiedzy ekspertów na temat defektów występujących w procesie integracji.

Efektom działania proponowanego modelu diagnostycznego jest lista przyczyn nieudanego przebiegu integracyjnego, uporządkowana wg wyznaczonego prawdopodobieństwa.

Autor wskazuje, że opracowany model może być wykorzystywany w systemie wspomagania decyzji podejmowanych przez człowieka, w systemie automatycznej korekty zmian, oraz w trybie mieszanym, w którym automatycznie korygowane są zmiany o najwyższym prawdopodobieństwie. Całość tych rozważań została osadzona w kontekście praktyki działania systemów CI.

Rozdział 6 prezentuje część eksperymentalną pracy. Eksperymenty zostały zrealizowane z wykorzystaniem danych zgromadzonych w trakcie działania systemu CI w firmie Microsoft. Dane dotyczą realizowanych tam projektów i zbierane były przez 16 miesięcy. Badania obejmowały szereg analiz, zarówno z wykorzystaniem wiedzy eksperckiej jak i bez niej. Rozdział w sposób systematyczny opisuje przeprowadzone eksperymenty. Autor wskazuje specyficzne przypadki występujące w trakcie integracji kodu wynikowego, opisuje w jaki sposób są one interpretowane w kontekście zmiennych (i ich wartości) w ramach stosowanego modelu bayesowskiego. Dla oceny jakości proponowanego modelu autor proponuje nieznacznie zmodyfikowaną metodę k -krotnej walidacji krzyżowej. Ze względu na specyfikę danych, modyfikacja bierze pod uwagę parametry czasowe owych danych, utworzone podzbiory nie są generowane całkowicie losowo.

W ramach badań analizowane są eksperymentalnie różne własności proponowanego modelu: zmiana jakości modelu związana z rozmiarem danych, reakcja na różne typy defektów itp. Autor przeprowadza wiele szczegółowych rozważań związanych z proponowanymi eksperymentami. Jednak analiza przeprowadzona jest jedynie w kontekście konkretnego zastosowania.

Analiza samego podejścia i proponowanego modelu jest wyraźnie drugoplanowa. W przeprowadzonych badaniach nie udało się znaleźć prostego porównania skuteczności działania proponowanego modelu w porównaniu z rutynowo działającym inżynierem oprogramowania. Trudno również znaleźć informacje o parametrach czasowych działania systemu,

Wydaje się, że ze względu na postulowany w pracy, wdrożeniowy charakter proponowanego rozwiązania, badania takie są możliwe do zrealizowania jak i pożądane ze względów praktycznych.

W podsumowaniu autor wskazuje na podatność wstecznej korekty defektów na automatyzację. Wskazuje, że proponowany model oparty na sieci Bayesa posiada właściwości samouczenia się, zawiera wiedzę ekspercką i był skuteczny w 50 % nieudanych przebiegów integracyjnych. Proponowane rozwiązanie nie jest uniwersalne, jednak można wskazać cechy projektów dla których proponowane rozwiązanie powinno być skuteczne.

Mocne strony pracy.

- Bardzo dobra znajomość problematyki.
- Zaproponowanie własnego spojrzenia autora na proces integracji ciągłej i proces projektowania oprogramowania.

- Zaproponowanie automatycznego modelu systemu opartego na sieci Bayesa.

Zaproponowany model uwzględniający podgrafy związane z węzłami grafu (podcelami) pozwala na automatyczne wskazanie wystąpienia defektów a także pozwala na automatyczny proces naprawy poprzez korektę wsteczną. Autor przeprowadził szereg eksperymentów na danych rzeczywistych pozyskanych w firmie Microsoft i dokonał wielu ciekawych analiz.

- Bardzo dobra redakcja pracy.

Praca została napisana w języku angielskim, zastosowana angielszczyzna nie budzi zastrzeżeń. Autor zachował właściwe proporcje pomiędzy zrozumiałością tekstu a dbałością o poprawność i elegancję sformułowań. Przedstawione treści są czytelne i zrozumiałe. Czasem wątpliwości budzi zastosowanie specyficznej terminologii z zakresu inżynierii oprogramowania, momentami bliskiej specyficznemu żargonowi używanemu w tej dziedzinie.

Wady pracy.

- Zbyt późne przedstawienie aspektu naukowego rozprawy (rozdział 4).
- Brak analizy poprzedzającej wybór modelu opartego na sieciach Bayesa.
- Brak analizy zaproponowanego modelu i jego porównania z tradycyjnym rozwiązaniem.

Reasumując: Biorąc pod uwagę wprowadzone własne podejście autora do problemu integracji ciągłej w inżynierii oprogramowania i szeroką analizę aspektów praktycznych tego podejścia uważam, że praca w pełni spełnia wymagania ustawowe i wnioskuję o dopuszczenie przedstawionej rozprawy doktorskiej do publicznej obrony.

