Automatyzacja w prognozowaniu jakości oprogramowania

Anna Bobkowska
Politechnika Gdańska

Streszczenie

Jest hipoteza mówiąca, że poprawę jakości oprogramowania i skrócenie czasu jego wytworzenia można uzyskać dzięki zastosowaniu takich narzędzi CASE, które wykonają niektóre zadania automatycznie lub przynajmniej będą je inteligentnie wspomagać. Czy jednak jest to możliwe w tak niepowtarzalnym przedsięwzięciu, jakim jest projekt informatyczny?

W referacie zaprezentowano wyniki badań z zakresu automatyzacji prognozowania jakości we wczesnych fazach wytwarzania oprogramowania. Omówiono zagadnienia związane z możliwością automatyzacji, przedstawiono wizję systemu inteligentnego wspomagania prognozowania jakości i prototyp systemu obliczającego metryki prognozowanej jakości oraz zasygnalizowano podobne prace z tego obszaru.

1. Wstęp

Wytwarzane współcześnie oprogramowanie jest coraz bardziej złożone. Jednocześnie obserwowane są dążenia do obniżenia kosztów projektu, skrócenia czasu wykonania i wzrostu jakości rozwiązania. Aby sprostać tym wyzwaniom do wspomagania procesu wytwarzania wykorzystywane są narzędzia CASE (ang. Computer Aided Software Engineering). Niewątpliwą zaletą stosowania narzędzi CASE w firmie jest poprawa jakości dokumentacji i komunikacji w zespole, jednakże zazwyczaj oczekiwania użytkowników są większe niż usługi oferowane przez konkretne produkty, a uzyskiwane w wyniku ich zastosowania zyski czasowe nie są satysfakcjonujące [13]. Powstaje więc pytanie o możliwość automatyzacji zadań na poszczególnych etapach procesu wytwarzania oprogramowania lub przynajmniej inteligentnego ich wspomagania. Przykładem

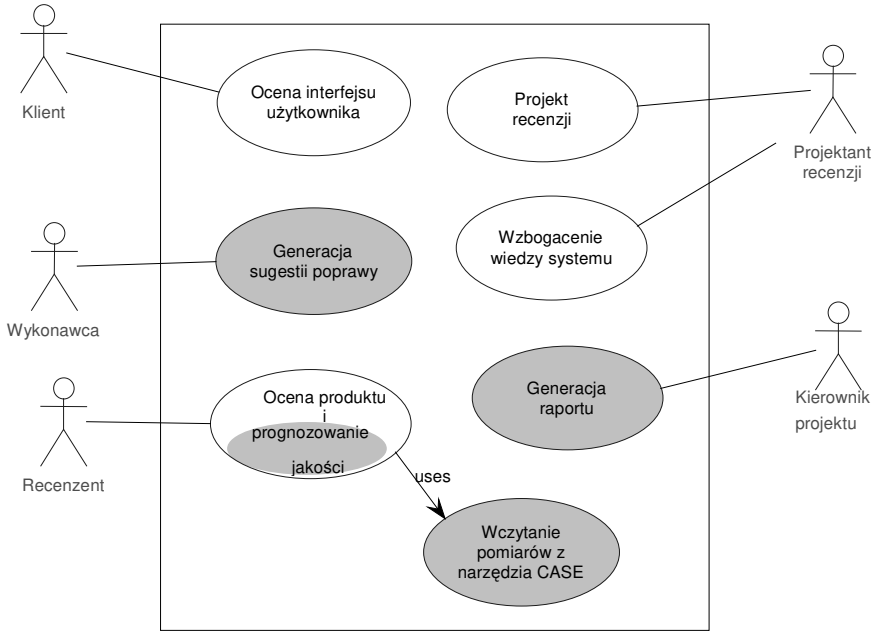
może być dążenie do automatycznej generacji kodu przez projektanta operującego na poziomie dziedziny problemu, sprawdzanie spójności modelu wykonanego w UML, generacja dokumentacji na podstawie modelu, automatyczne wykrywanie określonych typów defektów w kodzie itp. Istnieją nawet propozycje włączenia do narzędzi CASE elementów sztucznej inteligencji [1], takich jak bazy wiedzy, przetwarzanie języka naturalnego i pojęć opartych o pewne struktury, wnioskowanie w warunkach niepewności, modelowanie i budowanie aplikacji z inteligentnych agentów, tzn. obiektów posiadających nie tylko atrybuty i zachowanie, ale także cele i strategie działania oraz pewną znajomość otoczenia lub nawet możliwość uczenia się.

Prezentowane w referacie badania miały na celu analizę możliwości automatyzacji prognozowania jakości oprogramowania. Prognozowanie jakości oprogramowania dotyczy określania jakości oprogramowania w trakcie jego wytwarzania, tzn. wtedy, kiedy nie jest dostępny kod, ale dokumentacja systemu z zastosowaniem notacji UML [14]. Takie postępowanie ma na celu zapewnienie jakości już w fazach analizy i projektowania poprzez pokazanie konsekwencji jakościowych konkretnych rozwiązań oraz ich poprawę jak najniższym kosztem. Gromadzenie wskaźników prognozowanej jakości w kolejnych iteracjach poszczególnych etapów wytwarzania oprogramowania oraz innych wskaźników procesu wytwarzania umożliwia sterowanie jakością, a analiza tych danych może służyć zarządzaniu jakością i ciągłemu doskonaleniu procesu wytwarzania oprogramowania. Prognozowanie jakości na podstawie dokumentacji analizy i projektowania zwiększa prawdopodobieństwo wytworzenia produktu o wymaganej jakości. Badania nad automatyzacją procesu prognozowania jakości mają na celu zwiększenie efektywności wykonywania tej techniki.

2. Wizja systemu inteligentnego wspomagania prognozowania jakości

Idealnym rozwiązaniem byłoby wytworzenie aplikacji wykonującej prognozowanie jakości bez udziału recenzenta. Po wykonaniu dokumentacji zawierającej diagramy UML projektant uruchamiałby tę aplikację w celu automatycznego wyszukiwania defektów w modelu i usprawnienia ich usuwania. Jednocześnie dla kierownika projektu generowany by był odpowiedni raport o jakości i postępach w realizacji projektu. Całkowitej automatyzacji w ogólnym przypadku niestety nie można wykonać ze względu na dużą różnorodność projektów i wysokie kompetencje wymagane do oceny poszczególnych rozwiązań. Na rysunku 1 przedstawiono diagram przypadków użycia dla systemu realnego wspomagania prognozowania jakości, a w tabelach 1 i 2 zawarto opis aktorów i przypadków użycia. Wykorzystanie systemu ma następujący przebieg: na początku projektant recenzji wykonuje projekt recenzji, w czasie którego następuje integracja elementów wytwarzanej dokumentacji i wymaganej jakości.

Gdy wykonawca dostarczy produkty poszczególnych faz, klient może ocenić prototyp interfejsu, a recenzent może ocenić dokumentację projektową z zastosowaniem modeli UML. Po wykonaniu prognozowania jakości, które zawiera również zbieranie metryk diagramów i uwzględnienie ich w obliczeniach, wykonawca może wygenerować sugestie dotyczące poprawy produktu, a kierownik projektu – raport przydatny do zarządzania projektem.



Rysunek 1. Diagram przypadków użycia systemu inteligentnego wspomagania prognozowania jakości

Tabela 1. Opis aktorów przedstawionych na rysunku 1.

Aktor	Opis
Klient	Użytkownik dostarczający oceny prototypu interfejsu w zakresie użyteczności, łatwości użycia i satysfakcji.
Wykonawca	Użytkownik wytwarzający system, który oczekuje sugestii dotyczących poprawy jakości dokumentacji.
Projektant recenzji	Użytkownik odpowiedzialny za projekt recenzji w zakresie określenia zakresu zbieranych danych oraz wewnętrznej integracji informacji z różnych źródeł (klient, poszczególni recenzenci, wewnętrzna wiedza systemu).
Recenzent	Użytkownik dokonujący recenzji, którego zadaniem jest znalezienie defektów i dokonanie oceny w określonym zakresie. Jego czynności są wspierane przez ankietę oceny produktu. Recenzent może podać sugestie dotyczące poprawy. Recenzenci posiadający różną specjalizację mogą pracować nad innym zakresem dokumentacji i być ukierunkowani na inne aspekty dokumentacji.
Kierownik projektu	Użytkownik odpowiedzialny za organizację: planowanie projektu, śledzenie i zapewnienie zasobów. Jest on zainteresowany krótkim raportem na temat postępów w realizacji projektu i jakości produktów.

Tabela 2. Opis przypadków użycia pokazanych na rysunku 1.

Przypadek użycia *Aktor*	Opis przypadku użycia
Ocena interfejsu użytkownika *Klient*	Klient otrzymuje ankietę umożliwiającą o wyrażenie opinii na temat interfejsu (prototypu lub ostatecznej wersji), które będą pomocne w nanoszeniu korekt w celu zapewniania satysfakcji klienta.
Generacja sugestii poprawy *Wykonawca*	Zwraca wykonawcy listę potencjalnych defektów znalezionych podczas przeglądu jego produktu z sugestiami ich usunięcia. Pozwala także wykonawcy na skomentowanie ich.
Ocena produktu i prognozowanie jakości *Recenzent*	Wspomaga recenzenta w zbieraniu danych dotyczących jakości przez dostarczenie ankiet pomagających w znajdowaniu defektów i dokonywaniu ocen. Umożliwia wywołanie części prognozowania związanej z przekształcaniem tych danych w metryki charakterystyk jakości. *Wykorzystuje:* wczytanie danych z narzędzia CASE
Wczytanie pomiarów z narzędzia CASE	Pobiera metryki z narzędzia CASE wspomagającego modelowanie. Oblicza złożone metryki oraz na podstawie metryk i ich wartości oczekiwanych wskazuje na anomalie w projekcie.
Projekt recenzji *Projektant recenzji*	Dostarcza zbioru szablonów projektantowi recenzji w celu ułatwienia przygotowania recenzji dokumentacji i oceny prototypu przez klienta, integracji oczekiwanych rezultatów z wbudowaną wiedzą oraz zdefiniowania formatu sugestii dla wykonawcy oraz formatu raportu dla kierownictwa.
Wzbogacenie wiedzy systemu *Projektant recenzji*	Umożliwia wzbogacenie wiedzy systemu i pozyskania wiedzy z danych z projektów poprzednich. Wiedza systemu obejmuje wartości oczekiwane metryk diagramów, reguły wnioskowani na temat jakości, heurystyki projektowe, wzorce i anty-wzorce projektowe, reguły generowania sugestii i reguły integracji informacji pochodzących z różnych źródeł. Mogą one pochodzić z literatury, od lokalnych ekspertów, z poprzednich projektów itp.
Generacja raportu *Kierownik projektu*	Dostarcza krótkiego raportu dla kierownika projektu dotyczącego postępów w realizacji projektu i jakości. Raport powinien wspomagać organizację projektu. Atrybutami bezpośrednio powiązanymi z prognozowaniem są jakość, czas i zasoby.

Na diagramie na szaro zaznaczono miejsca potencjalnej automatyzacji. Nie można zautomatyzować pozyskiwania informacji dotyczącej trudnych do zmierzenia aspektów, ale można zautomatyzować znajdowanie anomalii w projekcie na podstawie diagramów i wiedzy zawartej w systemie, można automatycznie wykonać prognozowanie jakości na podstawie zebranych danych oraz przekształcać pozyskane informacje w ramach generacji raportu, czy też generacji sugestii poprawy.

3. Prototyp systemu obliczającego metryki prognozowanej jakości

Zakres prototypu odpowiada funkcjonalności przypadku użycia ‘Ocena produktu i prognozowanie jakości’. Udostępnione są następujące usługi:

- Obliczanie metryk jakości charakterystyk i atrybutów jakości w oparciu o elementarne metryki jakości wprowadzone w fazach recenzji analizy, projektu interfejsu użytkownika i projektu systemu;
- Zmiany parametrów w zakresie uwzględnienia wyników z poprzednich faz oraz uwzględnienia progu prognozowalności;
- Obliczanie samych atrybutów jakości aplikacji na podstawie metryk określających charakterystyki jakości;
- Zapis do pliku raportu z prognozowania metryk jakości dotyczących charakterystyk i atrybutów.

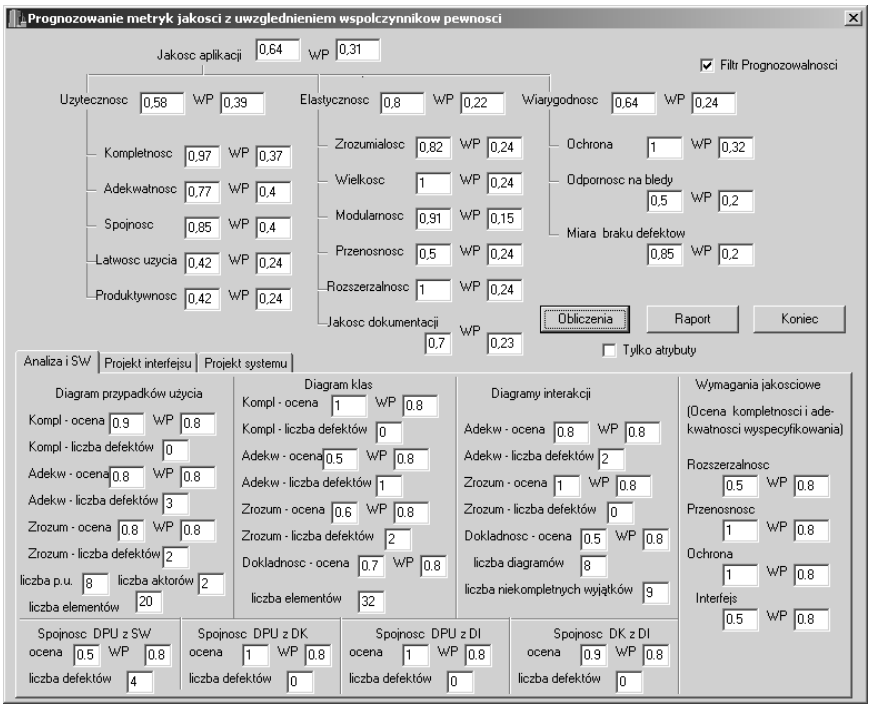
Podstawą budowy prototypu był model prognozowania i poprawy jakości uwzględniający metryki diagramów, oceny i defekty. Zostały one poddane kwantyfikacji i uwzględniono niepewność wnioskowania w postaci współczynników pewności (WP). Do przekształcania informacji jakościowej wykorzystano modele przekształceń zaproponowane w [4]. Założono kwantyfikację w przedziale [0..1] i współczynniki pewności również w przedziale [0..1]. Metryki diagramów są odniesione do wartości oczekiwanych i na tej podstawie obliczana jest elementarna metryka jakości związana z adekwatnością poszczególnych aspektów diagramu. Wartości oczekiwane zostały obliczone na podstawie podobnych projektów [3]. Oceny, które są związane z subiektywnymi albo trudnymi do precyzyjnego zmierzenia aspektami, występują jako liczby z przedziału [0..1]. Elementarnymi metrykami jakości są także lokalne metryki defektów, tzn. metryki wskazujące liczbę określonego typu defektów podzielone przez miarę lokalnej wielkości.

Interfejs użytkownika prototypu systemu obliczającego prognozowane metryki jakości dla fazy analizy i specyfikacji wymagań przedstawiono na rysunku 2. Zawarte tam dane pochodzą z projektu studenckiego pt. „Rezerwacja miejsc w hotelu przez Internet”. Dla każdej z faz pokazane jest drzewo jakości (będące rozszerzeniem struktury FCM [7]) oraz elementarne metryki jakości uwzględniane w danej fazie. Możliwość uwzględnienia prognozowalności występuje jako ‘Filtr prognozowalności’, a w fazach projektu interfejsu i projektu systemu jest możliwość uwzględnienia wyników z poprzednich faz poprzez zaznaczenie ‘Łącznie z poprzednimi fazami’.

W celu weryfikacji przydatności prototypu wykonano studium przypadku dla faz analizy, projektu interfejsu i projektu systemu wspomnianego powyżej projektu. W ramach interpretacji wyników przeanalizowano:

- różnice wyników bez uwzględnienia współczynników pewności i z ich uwzględnieniem;
- interpretację wyników z zastosowaniem progów akceptowalności;
- informacje wynikające z zastosowania filtra prognozowalności;
- zmiany wartości w poszczególnych fazach wytwarzania oprogramowania.

Wartości współczynników pewności związanych z wynikami dostarczają informacji na temat dokładności wyników. Porównanie metryk jakości bez uwzględnienia współczynników pewności i z uwzględnieniem ich wskazuje, że zazwyczaj wartości otrzymane z uwzględnieniem współczynnika pewności są trochę niższe, co jest zgodne z oczekiwaniami (niepewność miała zmniejszać wartość wyniku). Dla przyjętej wartości współczynnika pewności równej 0.8, różnica wyników nie przekraczała 0.1.



Rysunek 2. Interfejs użytkownika prototypu systemu obliczającego metryki prognozowanej jakości dla fazy analizy i specyfikacji wymagań.

W celu wykorzystania uzyskanych danych, istnieje konieczność interpretacji wyników z zastosowaniem progów akceptowalności. Np. jeżeli dla elastyczności i wiarygodności ustawimy progi wynoszące dla jakości bardzo dobrej (5) – [0.8..1]; jakości dobrej (4) – [0.6..0.8); jakości średniej (3) – [0.4..0.6); jakości słabej (2) – [0.2 ..0.4) i jakości bardzo słabej (1) – [0 ..0.2) otrzymamy dla elastyczności wyniki bardzo dobre, a dla wiarygodności wyniki poszczególnych faz: dobry, średni i dobry.

Filtr prognozowalności pokazuje, w jakim stopniu możemy być przekonani o jakości produktu w kontekście całego projektu. Jeżeli jakość jest zadawalająca i współczynnik pewności z uwzględnieniem prognozowalności przyjmuje wysoką wartość, świadczy to o małym ryzyku słabej jakości rozpatrywanego produktu. Cechą charakterystyczną poprawnie wytwarzanego projektu jest wzrost współczynnika pewności dla danych z uwzględnieniem poprzednich faz, np. w analizowanym projekcie dla jakości aplikacji dla fazy analizy wynosi on 0.25, dla fazy projektu interfejsu 0.46, a dla fazy projektu systemu 0.53. W tym przypadku wartości prognozowanej jakości są do siebie zbliżone, co oznacza, że jakość produktów z kolejnych faz utrzymuje się na zbliżonym poziomie. Słabe wartości metryk jakości wskazują na fazy, w których należy poszukiwać przyczyn braku jakości wytwarzanego systemu. Współczynniki pewności o małych wartościach wskazują na słabą jakość dokumentacji w tych fazach.

W celu weryfikacji parametrów prototypu dokonano porównania uzyskanych z jego zastosowaniem wyników z wynikami otrzymanymi podczas wnioskowania intuicyjnego wspomaganego odpowiednimi ankietami w ramach eksperymentów weryfikujących technikę prognozowania jakości. Przy badaniu 10 najlepszych projektów (pominięto dane, które mają inne przyczyny nieprawidłowości) w 90% uzyskano różnicę wyników nie przekraczającą jednego stopnia w skali [1..5]. Można z tego wnioskować, że otrzymane wyniki są zbieżne, a model został prawidłowo sparametryzowany.

4. Podobne prace

W literaturze, można znaleźć opis szeregu badań mających na celu zautomatyzowanie znajdowania defektów w modelach obiektowych. Jedną z pierwszych pozycji była praca [10] badająca progi wartości metryk w celu wykrycia anomalii i zidentyfikowania problemu. Z każdym zidentyfikowanym problemem skojarzono akcje sugerowane w celu poprawy modelu. Autorzy dla każdej metryki zdefiniowali następujący szablon:

- statystyczne pomiary ze zbadanych projektów,
- znaczenie metryki - opis, co logicznie wynika ze zmierzenia metryki,
- podobne metryki - mierzą podobne czynniki albo uzupełniające,
- progi - zakresy pożądanych wartości metryk, które służą do wykrycia anomalii i tym samym wskazują na istnienie problemów,
- sugerowane akcje w przypadku identyfikacji problemu, przy czym decyzja o zastosowaniu sugerowanych akcji należy do projektanta.

Metryki produktów dotyczą głównie diagramu klas. Podobne metryki dla diagramu przypadków użycia zostały zaproponowane w pracy [5].

Ocena jakości modeli UML bazująca na metodzie GQM [2] zaprezentowana została w pracy [12]. Celem opisanych badań jest pomiar jakości i poprawa produktów. Metryki mają wskazywać miejsca zwiększonego ryzyka. Model

zawiera metryki, progi i heurystyki. Autor twierdzi, że trudno jest znaleźć metryki interpretujące heurystyki. Należy rozważać metryki w kontekście, np. języka programowania, platformy, architektury, używanych mechanizmów, więc nie ma uniwersalnych progów (chociażby biorąc pod uwagę różnice pomiędzy komponentami 'serwera' i 'klienta'). Heurystyki znajdują nie defekty, ale najbardziej zagrożone fragmenty, które wymagają poświecenia im dodatkowego czasu. Ostatnio opublikowana została także praca [16], w której autorzy starają się określać jakość na podstawie diagramów UML z fazy projektu. Poszukują oni bezpośrednich wskaźników jakości oraz starają się wnioskować o jakości na podstawie wzorców projektowych oraz anty-wzorców wskazujących na defekty.

Propozycję automatyzacji pomiarów i ocen w oparciu o bazę wiedzy zaprezentowano w pracy [8]. Celem systemu jest wspomaganie projektantów w obiektywnym wnioskowaniu na temat jakości produktów projektowania i w wyborze poszczególnych konstrukcji projektowych, które odbywa się poprzez zautomatyzowane przeglądy w zakresie wskazywania potencjalnych defektów i podawania propozycji, jak rozpoznane defekty mogą być usunięte. W tym modelu występują następujące elementy:

- heurystyki - zdroworozsądkowe reguły poprawności projektu, wśród których wyróżnione są heurystyki sprawdzalne, sugerowane heurystyki (czasami nie wiadomo, jaka wartość jest optymalna) oraz niesprawdzalne automatycznie.
- reguły przekształcające fragment projektu - zalecenia jak wyeliminować potencjalne błędy, dla każdej heurystyki są dobrane inne reguły przekształcające;
- pomiary - które pomagają rozwiązywać konflikty tzn. dają możliwość porównania schematów przed i po wprowadzeniu zmian.

Inne badania dotyczą zapewnienia spójności pomiędzy różnymi diagramami UML. W pracy [15] wykorzystano przekształcenia z teorii grafów (ang. attributed typed graphs) do sprawdzenia spójności pomiędzy diagramami klas i sekwencji w zakresie istnienia poszczególnych elementów oraz ich widzialności i krotności. Prowadzone są także prace oparte na modelu SPIN. Praca [9] zawiera opis badań związanych z weryfikacją zachowania na podstawie rozszerzonych diagramów stanów (ang. statechart), a zaproponowany system w przypadku znalezienia defektów generuje diagramy sekwencji pokazujące jak znaleźć wykryty błąd. Natomiast w pracy [6] opisano problem weryfikacji poprawności protokołów uwierzytelniania oraz określonych własności poprawności (np. zakleszczenia) dla różnych architektur systemów rozproszonych.

Trwają także prace [11] nad zbudowaniem systemu uczącego się oceniać automatycznie poszczególne decyzje projektowe pod względem adaptowalności i wielokrotnego użycia oraz nad systemem wykorzystującym technologię automatycznych agentów do potwierdzania jakości w zakresie prognozowalności, łatwości utrzymania, testowalności, sterowalności, obserwowalności i automatycznej naprawialności (ang. survivability).

5. Podsumowanie

W referacie przeanalizowano możliwość automatyzacji prognozowania jakości aplikacji we wczesnych fazach wytwarzania oprogramowania. Badania pokazują, że całkowita automatyzacja nie jest możliwa ze względu na kompetencje wymagane do pozyskiwania informacji o jakości. Jednakże poszczególne techniki bazujące na wzorcach oraz na danych statystycznych z poprzednich projektów mogą wskazywać miejsca anomalii w dokumentacji, które pokazują miejsca zwiększonego ryzyka występowania defektów. Automatyzacja natomiast może być zastosowana do przetwarzania zgromadzonej informacji o jakości. Generacja sugestii poprawy dla wykonawcy powinna pozwalać na udoskonalenie jakości produktu, a generacja raportu dla kierownika projektu powinna podawać wskaźniki jakości przydatne do zarządzania projektem.

Zaprezentowano także prototyp systemu obliczającego metryki prognozowanej jakości na podstawie elementarnych metryk jakości. Metryki defektów oraz oceny są wprowadzane do systemu, natomiast metryki diagramów są przekształcane automatycznie z wykorzystaniem wartości oczekiwanych wyznaczonych na podstawie podobnych projektów. Do przekształcania na metryki charakterystyk jakości wykorzystano badania nad kwantyfikacją jakości z uwzględnieniem niepewności prognozowania. W celu weryfikacji systemu i jego parametrów dokonano porównania wyników obliczonych przez prototyp z rezultatami uzyskanymi w wyniku wnioskowania intuicyjnego oraz wykonano studium przypadku na podstawie jednego projektu eksperymentalnego.

Korzyścią wynikającą z zastosowania systemu w porównaniu z wnioskowaniem intuicyjnym jest skrócenie czasu związanego z obliczaniem metryk jakości oraz eliminacja subiektywizmu związanego z indywidualnym wnioskowaniem. Podawanie współczynnika pewności wnosi dodatkową informację o dokładności, która pomaga interpretować wyniki w postaci metryk jakości. Wyniki prognozowania pomagają w śledzeniu postępów w projekcie, jakości powstających produktów oraz ryzyka związanego z projektem. W ramach dalszych prac planowane jest rozszerzenie prototypu o inne metody automatycznego wykrywania anomalii i defektów w dokumentacji projektowej.

Bibliografia

[1]. *Automated Software Engineering*, internet: www.cs.vassar.edu/faculty/welty/ase/

[2]. V.R. Basili, G. Caldiera, D.H. Rombach. The Goal Question Metric Approach, *ISERN* 1996.

[3]. A. Bobkowska. Quantitative and qualitative methods in process improvement and product quality assessment., *Proceedings of ESCOM 2001*.

[4]. A. Bobkowska. Kwantyfikacja prognozowaniej jakości oprogramowania, *Materiały konferencyjne KKIO'2000*.

[5]. A. Bobkowska, Quality Indicators Related to Use Case Model, *Proceedings of ACS'99*.

[6]. P. Bose. Automated Translation of UML Models of Architectures for Verification and Simulation using SPIN, *Proceedings of the 14th IEEE Conference on Automated Software Engineering,* 1998.

[7]. N.E. Fenton. *Software Metrics. A Rigorous Approach*, Chapman&Hall, 1993.

[8]. T. Grotehen, K.R. Dittrich. *The MeTHOOD Approach: Measures, Transformation Rules, and Heuristics for Object-Oriented Design*, Technical Report., internet: http://www.ifi.unizh.ch/groups/dbtg/MeTHOOD/index.html

[9]. J. Lilius, I.P. Paltor. vUML: A Tool for Verifying UML Models, *Proceedings of the 14th IEEE Conference on Automated Software Engineering,* 1998.

[10]. M. Lorenz, J. Kidd. *Object-oriented Software Metrics. A Practical Guide.*, Prentice Hall, 1994.

[11]. T. Menzies. HQkb - *The High Quality Knowledge Base Initiative*, internet: http://research.ivv.nasa.gov/docs/techreports.

[12]. B. Lyons. Partner Perspective: Quantifying Quality in Your OO Models, *Rose Architect,* volume 1, 1998.

[13]. M. Piechówka, S. Szejko, CASE jaki jest, każdy widzi., *Informatyka,* numer 4, 1998.

[14]. Rational Software Corporation, *UML v.1.3.*, internet: www.rational.com

[15]. A. Tsiolakis, Consistency Analysis of UML Class and Sequence Diagrams based on Attributed Typed Graphs and their Transformation, internet: http://www.informatik.uni-bremen.de/~tsio/

[16]. A.I. Verkamo, J. Gustafsson, L. Nenonen, J. Paakki., Measuring design diagrams for product quality evaluation, *Proceedings of ESCOM 2001*.