

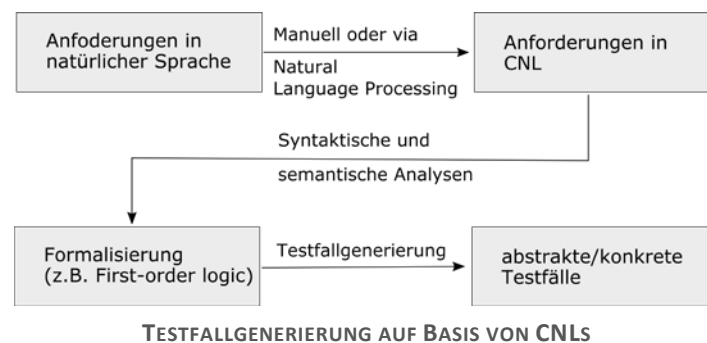
Masterarbeit

Evaluation of the SysReq Controlled Natural Language for the Generation of Functional Test Cases

Problemstellung

Im Kontext der Softwareentwicklung gewinnt die automatische Testfallgenerierung mit Hilfe von Modellen (*modellbasiertes Testen*) immer mehr an Bedeutung. Obwohl es bereits erfolgreiche Ansätze gibt, strukturelle Testfälle auf Basis von Codeanalysen zu generieren, werden funktionale Testfälle auf Basis der Anforderungen meist noch manuell abgeleitet. Dieser Prozess ist sehr zeitaufwendig und fehleranfällig.

Anforderungen werden häufig in natürlicher Sprache spezifiziert und beinhalten u.a. Tabellen oder Grafiken. Idealerweise möchte man die natürlich sprachlichen Anforderungen analysieren, formalisieren und daraus Testfälle ableiten. Aufgrund der Ambiguität und Komplexität natürlicher Sprache ist die Formalisierung der Anforderungen jedoch eine große Herausforderung. Im Rahmen dieser Arbeit soll untersucht werden, inwiefern sich Controlled Natural Languages (CNLs) zur Generierung von funktionalen Testfällen auf Basis von Anforderungen eignen. CNLs sind Sprachen, die der natürlichen Sprache ähnlich und durch bestimmte Regeln eingeschränkt sind. CNLs lassen sich formal beschreiben und somit ist es zumindest theoretisch möglich, Testfälle automatisch abzuleiten. Folgende Grafik beschreibt den Testfallgenerierungsprozess.



Aufgabenstellung

In einem Framework namens *NAT2TEST* von G. Carvalho et. Al. [1] wird eine Methodik präsentiert um aus Anforderungen funktionale Testfälle zu generieren. Die Anforderungen werden in einer CNL namens *SysReq-CNL* spezifiziert und eine Formalisierung in verschiedene intermediäre Modelle ist möglich. Im Rahmen dieser Arbeit sollen folgende Punkte evaluiert und prototypisch umgesetzt werden:

- Ist es möglich, Anforderungen aus einem Industrieprojekt automatisiert in die SysReq-CNL zu übersetzen?
- Ist die Ausdrucksstärke der CNL ausreichend um sämtliche Anforderungen darzustellen? Falls nein, welche Erweiterungen sind notwendig?
- Am Lehrstuhl wurde eine domänenspezifische Testfallbeschreibungssprache entwickelt. Welche intermediäre Formalisierung eignet sich am besten für die Testfallgenerierung in diesem Format?
- Welches Abstraktionsniveau erreichen die generierten Testfälle? Ist es möglich, ohne viel zusätzlichen Aufwand ausführbare Testfälle zu erzeugen?

Vorkenntnisse

Vorkenntnisse in folgenden Bereichen sind hilfreich, aber nicht erforderlich: Natural Language Processing, Mathematische Logik, Java Programmierung.

Ansprechpartner

Alexander Kugler, M. Sc. RWTH
akugler@embedded.rwth-aachen.de

[1] Carvalho G et. Al. (2015) NAT2TEST Tool: From Natural Language Requirements to Test Cases Based on CSP. In: Calinescu R., Rumpe B. (eds) Software Engineering and Formal Methods. Lecture Notes in Computer Science, vol 9276. Springer, Cham