

Entwurf und Implementierung einer Werkzeugunterstützung zur sprachlichen Analyse und automatisierten Transformation von Projektlastenheften im Kontext der Automobilindustrie

An der Fachhochschule Dortmund

im Fachbereich Informatik

Studiengang Informatik

Vertiefung Praktische Informatik

erstellte Thesis

zur Erlangung des akademischen Grades

Bachelor of Science

B. Sc.

von Aaron Schul,

geboren am 24.06.1997

und Felix Ritter

geboren am 31.08.1997

Betreuung durch:

Prof. Dr. Sebastian Bab

Dortmund, 28.02.2019

Zusammenfassung

Ein weitläufiges Problem in der Autoindustrie ist die effiziente Verarbeitung von Pflichtenheften. Einer der Gründe dafür ist, dass viele verschiedene Bereiche bei der Entwicklung der Fahrzeuge und sogar kleinster Einzelteile beteiligt sind. So müssen zu Beginn einer Produktentwicklung etwa Betriebswirtschaftler, Designer, Techniker und Ingenieure zusammen ein Dokument entwerfen. Darin müssen die Anforderungen an das gewünschte Produkt so genau beschrieben sein, dass es anhand dieses Dokuments entwickelt werden kann. Als Experten ihrer jeweiligen Domäne weiß dabei jeder genau was dazu nötig ist. Sobald jedoch Auswirkungen über die eigene Domäne hinausgehen, kann es schnell passieren, dass Widersprüche oder Abhängigkeiten entstehen. Diese werden später schnell übersehen, da solche Zuständigkeiten nicht eindeutig geklärt sind. Das Resultat ist dann ein Fehler in der Entwicklung, welcher von zeitlicher Verzögerung über zusätzliche Kosten bis hin zum Abbruch des Projekts führen kann. Methoden aus dem Natural Language Processing (NLP) können dabei helfen die Überprüfung der Pflichtenhefte auf Konsistenz stark zu vereinfachen oder sogar teilweise zu automatisieren. Dafür kann beispielsweise der Text in den Lastenheften analysiert und dessen Inhalt innerhalb einer Wissensbasis, sog. Ontologie, abgespeichert werden. In dieser Arbeit wird daher die Entwicklung zweier NLP-basierter Werkzeuge zur Vereinfachung bzw. Lösung dieser Probleme vorgestellt werden. Zum einen geht es um ein Programm namens Requirements-to-Boilerplate-Converter (R2BC), welches dem Nutzer helfen soll das Pflichtenheft eines Auftraggebers in die betriebsinternen Richtlinien und Standards zu überführen. Zum anderen wird der sog. Delta-Analyser dargestellt, welcher auf dem R2BC aufbaut, indem er automatisch zwei homogene Lastenheft vergleicht und dadurch im Kontext des gesamten Pflichtenhefts Widersprüche und Abhängigkeiten herausstellt. Ziel ist es dabei jeweils einen Prototypen als Proof of concept, also als Machbarkeitsbeweis, der Programmkonzepte aus ZIC19 zu implementieren. Auf Grund der begrenzten Ressourcen dieser Arbeit wird zusätzlich eine Liste mit weiterem Potential der Programme erläutert, welches sich bei der Entwicklung der Prototypen gezeigt hat. Diese bietet Vorschläge zur weiteren bzw. vollständigen Implementierung der Programme.

Abstract

A widespread problem in the auto industry is the efficient processing of specifications. One of the reasons for this is that many different areas are involved in the development of vehicles and even the smallest of parts. For example, at the beginning of product development, managers, designers, technicians, and engineers all need to design a single document together. The requirements for the desired product must be described in such detail that it can be developed using this document. As experts in their domain, everyone knows exactly what is needed. However, as soon as effects go beyond the own domain, it can quickly happen that contradictions or dependencies arise. These are quickly overlooked later, as such responsibilities are not clearly clarified. The result is then a mistake in the development, which can lead from time delay over additional costs up to the demolition of the project. Methods from Natural Language Processing (NLP) can help to simplify or even partially automate the review of functional specifications for consistency. For example, the text in the specifications can be analyzed and its contents stored within a knowledge base, so-called ontology. In this work, therefore, the development of two NLP-based tools to simplify or solve these problems will be presented. On the one hand, there is a program called Requirements-to-Boilerplate Converter (R2BC), which should help the user to transfer the specifications of a client into the company's internal guidelines and standards. On the other hand, the so-called delta analyzer is shown, which is based on the R2BC, in that it automatically compares two homogeneous specification sheets and thereby highlights contradictions and dependencies in the context of the entire specification. The goal here is to implement a prototype as a proof of concept of the program concepts from ZIC19. Due to the limited resources of this work, a list with further potential of the programs, which has been shown during the development of the prototypes, is additionally explained. This offers suggestions for further or complete implementation of the programs.

Inhaltsverzeichnis

1	Einführung	6
1.1	Motivation	6
1.2	Hypothese	8
1.3	Methodik	8
1.4	Inhalt	8
1.5	Autoren	8
2	Stand der Technik	9
2.1	Rückblick auf die Projektarbeit, Grundlagen zu NLP und Ontologien	9
2.2	Verwandte Arbeiten	9
3	Betriebliches Umfeld - Hella Use-Case	10
3.1	Re-Prozesse bei Hella und allgemein in Firmen	10
3.2	Betriebliche Anforderungen	10
3.3	Ansatz und Konzept unserer Werkzeuge	10
4	R2B-Converter	11
4.1	Architektur Klassen und Verteilung der Ressourcen	11
4.2	Implementierung (bisschen Code, GUI, Listenarchitektur, Workflow für User	11
4.3	mögliche Erweiterungen	11
4.4	Test	11
4.4.1	Methodik	11
4.4.2	Durchführung	11
4.4.3	Ergebnisse	11

5	Delta-Analyse	12
5.1	Architektur	12
5.2	Implementierung	12
5.3	mögliche Erweiterungen	12
5.4	Test	12
5.4.1	Methodik	12
5.4.2	Durchführung	12
5.4.3	Ergebnisse	12
6	Evaluation	13
6.1	Auswertung der Testresultate	13
6.2	Ziel erreicht? Hypothese reviewen und schwafeln	13
6.3	Mehrwert?	13
7	Fazit	14
7.1	Zusammenfassung	14
7.2	Ausblick	14

Tabellenverzeichnis

Abbildungsverzeichnis

Danksagung

Ich danke meiner Mama und seiner Mama, das was er gesagt hat.

Kapitel 1

Einführung

1.1 Motivation

Im betrieblichen Umfeld liegen zu Beginn eines jeden Entwicklungsprojektes für neue Produkte die Aufgaben und Ziele für die Entwicklung als Dokumente vor. Forschungsergebnisse finden Anwendung in der Vorentwicklungsphase, in der die Eignung der Erkenntnisse für neue Produkte eines Unternehmens evaluiert wird. Die Produktentwicklung unterliegt dabei bestimmten Kriterien und Faktoren, die den unternehmerischen Erfolg beeinflussen. Neben betriebswirtschaftlichen Einflüssen wie der Einordnung des Produktes in der Wertschöpfungskette sind es dabei besonders technische Anforderungen an das Produkt, die definiert und während der Produktentwicklung eingehalten werden müssen. Verschiedenste Akteure aus einem Unternehmen sind dabei an der Festlegung der Anforderungen an ein Entwicklungsprojekt bzw. Produkt beteiligt.

In der Automobilindustrie betrifft dieser Ablauf zumeist die Entwicklung neuer Fahrzeugkomponenten, heutzutage meist elektronische und mechanische Bausteine. Diese Bausteine werden dabei nicht sämtlich vom Fahrzeughersteller (OEM) selbst, sondern durch eine Vielzahl von Zulieferern produziert und entwickelt. Die Produktspezifikationen liegen meist digital als Texte, Tabellen und Grafiken vor und werden an den Zulieferer übermittelt. Nach dem Entwicklungsprozess steht dann die (Serien-)entwicklung und -fertigung des Produktes für das Ausrollen in großen Stückzahlen an den Hersteller, der das zugelieferte Produkt dann in seinen Produkten verwendet. Um dies zu erreichen, müssen während des gesamten Prozesses die Anforderun-

gen, die das Systemumfeld des Fahrzeugherstellers hat, berücksichtigt und eingehalten werden.

Die Anforderungen an das Produkt, etwa technische Rahmenbedingungen, werden dabei von vielen verschiedenen Domänenexperten beim OEM formuliert und in das sogenannte Pflichtenheft für die Entwicklung eingetragen. Beteiligte sind etwa Produktdesigner, Ingenieure und Systemtechniker, die an verschiedenen Stellen im Lastenheft Anforderungen an eine Komponente festlegen. Diese Beteiligten sind in der Regel auf ihren Bereich spezialisiert und nicht interdisziplinär, zudem gibt es sprachliche Eigenheiten der Autoren und unternehmensinterne Richtlinien für die Formulierung, die das Verständnis erschweren können. Demzufolge sammeln sich im Lastenheft verschiedenste Merkmale einer Komponente, die aber nicht im Bezug zueinander stehen und sich im schlimmsten Fall gegenseitig ausschließen.

Durch diese fachliche Breite und Tiefe der Spezifikationen im Pflichtenheft, aber auch durch den Umfang des Lastenheftes von mehreren tausend Seiten, kommt es häufig insbesondere zu Verständnisproblemen auf Seite des Zulieferers. Die Gewichtung einzelner Anforderungen in einem größeren Systemkontext fällt dort schwer, da nun Projektteammitglieder, die an der Entstehung des Lastenheftes nicht beteiligt waren, dieses verstehen und ein Produkt entwickeln sollen, dass möglichst alle Anforderungen berücksichtigt. In Texten muss also nach Zusammenhängen und Bezügen zwischen mehreren Anforderungen gesucht werden, damit die Korrektheit des späteren Produktes gewährleistet ist.

Die Analyse von Zusammenhängen zwischen Anforderungen stellt dabei aus Gründen der Effizienz ein Problem dar, wenn jeder Beteiligte von Hand die für ihn relevanten Anforderungen aus dem Lastenheft extrahieren muss. Auch müssen die Lastenhefte an die Formulierungen und Ausdrucksweisen für Requirements-Management im Unternehmen angepasst werden. Bislang gibt es jedoch kaum Werkzeugunterstützung, die effiziente Möglichkeiten zur automatisierten Überarbeitung und Anpassung einzelner Anforderungen aus dem Dokument bietet. Ansätze aus dem *Natural-Language-Processing* (NLP) stellen gleichzeitig vielversprechende Forschungsfelder in der Informatik dar, die eine solche automatisierte Verarbeitung auf Basis von Sprachanalyse ermöglichen. Syntax und Semantik der einzelnen Sätze und Zusammenhänge in Texten können auf Basis aktueller Trends wie Machine-Learning und dynamischer Programmierung zunehmend besser abgebildet werden.

1.2 Hypothese

Hypothese dieser Arbeit ist, dass sich mithilfe von NLP Lastenhefte effizient automatisiert verarbeiten lassen, womit die Arbeit von Requirements-Engineers, aber auch von Beteiligten an der Entwicklung beim Verständnis der Anforderungen erleichtert wird. Dabei ist die Auswertung der Syntax für ein tieferes Verständnis von Textzusammenhängen, also von verschiedenen Anforderungen, relevant. Insbesondere lassen sich die beschriebenen Probleme bei der Überprüfung der Pflichtenhefte auf Konsistenz lösen oder zumindest vereinfachen. Dies kann geschehen, mit Hilfe der in dieser Arbeit beschriebenen Programme, welche konzeptuell dargestellt und prototypisch implementiert werden. Weiterhin soll geprüft werden, wie stark ein Mehrwert bei der Nutzung der Programme für ein Firma ausfallen kann. Dies beinhaltet ersparte Arbeitszeit, Kosten und vermiedene Fehler.

1.3 Methodik

Zum Verfassen dieser Arbeit waren die Autoren fünf Monate bei der HELLA GmbH Co. KGaA angestellt. Dies beinhaltete zwei Monate Arbeit als Praktikanten und die verbleibenden drei Monate als Bacheloranden. HELLA ist ein international operierender deutscher Automobilzulieferer mit Hauptsitz in Lippstadt, welcher hauptsächlich Scheinwerfer und elektrotechnische Komponenten herstellt. In diesem Rahmen konnte in enger Zusammenarbeit mit der Projects, Methods and Tools-Abteilung (PMT) von HELLA genau ermittelt werden, wie Pflichtenhefte momentan verarbeitet werden, welche Verbesserungspotentiale dabei bestehen und wie diese genutzt werden können. Dadurch entstand eine genaue Vorstellung der zu entwickelnden Software, welche mit Hilfe der Programmkonzepte von Konstantin Zichler ZIC19 entworfen und implementiert werden konnte. Die Prototypen wurden mit originalen Pflichtenheften von verschiedenen OEMs und der HELLA getestet und evaluiert.

1.4 Inhalt

1.5 Autoren

Kapitel 2

Stand der Technik

2.1 Rückblick auf die Projektarbeit, Grundlagen zu NLP und Ontologien

2.2 Verwandte Arbeiten

Kapitel 3

Betriebliches Umfeld - Hella Use-Case

- 3.1 Re-Prozesse bei Hella und allgemein in Firmen
- 3.2 Betriebliche Anforderungen
- 3.3 Ansatz und Konzept unserer Werkzeuge

Kapitel 4

R2B-Converter

- 4.1 Architektur Klassen und Verteilung der Ressourcen
- 4.2 Implementierung (bisschen Code, GUI, Listenarchitektur, Workflow für User)
- 4.3 mögliche Erweiterungen
- 4.4 Test
 - 4.4.1 Methodik
 - 4.4.2 Durchführung
 - 4.4.3 Ergebnisse

Kapitel 5

Delta-Analyse

5.1 Architektur

5.2 Implementierung

5.3 mögliche Erweiterungen

5.4 Test

5.4.1 Methodik

5.4.2 Durchführung

5.4.3 Ergebnisse

Kapitel 6

Evaluation

6.1 Auswertung der Testresultate

6.2 Ziel erreicht? Hypothese reviewen und schwafeln

6.3 Mehrwert?

Kapitel 7

Fazit

7.1 Zusammenfassung

7.2 Ausblick

Literaturverzeichnis