Bestimmung der semantischen Funktion von Quelltextabschnitten

Dokumentenart:	Exposé für eine Bachelorarbeit
Autor:	Timo Januschke
Matrikel-Nr.:	1943051
Studiengang:	Informatik
Betreuer:	Tobias Hey
Datum:	8. Mai 2020

1 Motivation

Softwaresysteme werden immer größer und komplexer. Um so wichtiger ist es bei der Entwicklung und Dokumentation dieser Systeme eine Rückverfolgbarkeit von Anforderungen zu Quelltext (engl. *requirements-to-code traceability*) zu ermöglichen. Rückverfolgbarkeit bedeutet sowohl von den Anforderungen auf den implementierenden Quelltext, als auch vom Quelltext auf die spezifizierenden Anforderungen schließen zu können [BMP13]. Das Vorhandensein von Rückverfolgbarkeitsinformationen wirkt sich im allgemeinen positiv auf die Qualität von Softwaresystemen aus, da durch sie Wiederverwendung von Softwarekomponenten, Einarbeitung in den Quelltext oder Quelltextnavigation unterstützt werden können. Der Aufwand für die Erstellung und Instandhaltung dieser Informationen ist jedoch sehr hoch.

Das Projekt INDIRECT [Hey19] (Intent-driven Requirements-to-Code Traceability) erzeugt diese Rückverfolgbarkeitsinformationen automatisch aus den Anforderungen und dem Quelltext eines Software-Projekts. Um Verbindungen zwischen Anforderungen und Quelltext herzustellen muss INDIRECT sowohl die Semantik der Anforderungen als auch die Semantik des Quelltextes verstehen. Im Rahmen einer Masterarbeit [Eur20] wurde bereits eine semantische Repräsentation des Quelltextes erarbeitet: Quelltextelemente werden ihrer semantischen Ähnlichkeit nach gruppiert und diese Gruppierungen werden iterativ zu einer Hierarchie zusammengeführt. Diese Gruppierungen beschreiben welche Quelltextelemente eine gemeinsame Absicht implementieren. Ist eine natürlichsprachliche Beschreibung dieser Absicht vorhanden, so ermöglicht dies einen semantischen Vergleich zwischen ihr und den Anforderungen des Systems.

Abbildung 1 zeigt diese Zusammenhänge anhand einer Klasse `Contract`, welche für das Verwalten von Verträgen verantwortlich ist: Zu sehen ist eine

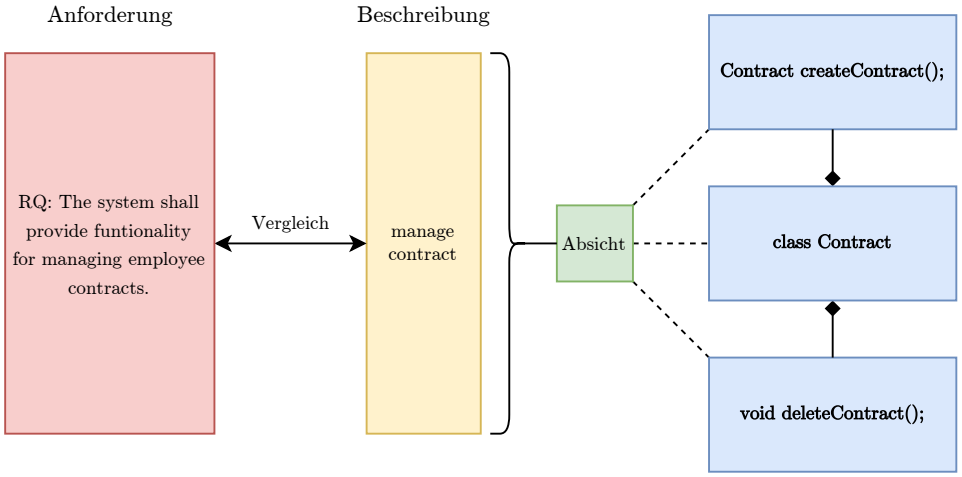

Abbildung 1: Vereinfachtes Vorgehen bei der Herstellung von Verbindungen zwischen einer Anforderung und einer Quelltextgruppierung

Gruppierung von Programmelementen (blau), deren zugehöriger Absichtsknoten (grün), eine mögliche Absichtsbeschreibung (gelb) und die zugehörige Anforderung (rot). Die Programmelemente der Quelltextgruppierung hängen über ihre gemeinsame Absicht, dem Verwalten von Verträgen, zusammen. Für diese Absicht wird eine Beschreibung erzeugt, deren Semantik mit der semantischen Repräsentation der Anforderungen verglichen wird. Obwohl „Löschen“ und „Erstellen“ nicht explizit in der Anforderung vorkommen, kann aufgrund der semantischen Ähnlichkeit zum Konzept „Verwalten“ eine Verbindung hergestellt werden.

2 Zielsetzung

Ziel dieser Arbeit ist es ein Verfahren zu entwickeln, welches auf der semantischen Quelltextrepräsentation von INDIRECT aufbaut und für eine gegebene Menge an Quelltextelementen eine Beschreibung ihrer gemeinsame Absicht erzeugt. Als Teil dieses Ziels muss außerdem eine geeignete Repräsentation für diese Absichtsbeschreibungen erarbeitet werden.

3 Ansatz

Das Finden von Beschreibungen für Dokumente deren Absicht unbekannt ist, wird im Allgemeinen als Themenanalyse (engl. *topic modelling*) bezeichnet. Vorhandene Verfahren zur Themenanalyse sind oft auf die Verarbeitung von Fließtext ausgelegt. Anders als Fließtext weist Quelltext jedoch einige

Besonderheiten auf: Das genutzte Vokabular ist meist stark eingeschränkt, domänenabhängig und viele Worte wiederholen sich oft. Klassische Themenanalyseverfahren, deren Training große Datenmengen benötigt, eignen sich deshalb nur bedingt für die Themenanalyse von Quelltext [MN15].

Allerdings gibt es auch Arbeiten, welche auf diesen klassischen Verfahren aufbauen, sie erweitern und speziell für die Verarbeitung von Quelltext anpassen. Eine solche Arbeit ist STM (engl. *semantic topic models*) von Mahmoud und Bradshaw [MB17]. STM baut insofern auf den klassischen Verfahren auf, dass jede Klasse im Quelltext als ein Dokument behandelt wird und die ausgegebenen Beschreibungen für die gesamte Klasse gelten. Es gilt hierbei zu beachten, dass diese Klassengranularität nicht direkt auf INDIRECT angewendet werden kann, da die Quelltextgruppierungen der semantischen Repräsentation von INDIRECT Quelltextabschnitte aus mehreren Klassen enthalten können. Aus diesem Grund wird die Eignung dieses Ansatzes in Bezug auf mehrere disjunkte Quelltextabschnitte überprüft werden. Verfahren wie STM, welche auf klassischen Themenanalyseverfahren aufbauen, erzeugen meist eine Ausgabe in Form einer Menge von einzelnen Worten, welche die Absicht eines Dokuments beschreiben.

Handelt es sich bei den Dokumenten, deren Absicht man bestimmen möchte, explizit um Quelltextdokumente spricht man auch von Quelltextzusammenfassung (engl. *source code summarization*). Neuere Arbeiten aus diesem Umfeld basieren auf Verfahren und Techniken aus dem Bereich der maschinellen Übersetzung (engl. *machine translation*). Genauer auf neuronalen Codierer-Decodierer-Architekturen (engl. *neural encoder-decoder architectures*). Diese Architekturen erlauben, anders als klassische Verfahren, eine Ausgabe von ganzen natürlichsprachlichen Sätzen. Haque *et al.* [HLWM20] haben ein Verfahren entwickelt, welches unter Berücksichtigung des Programmkontext, natürlichsprachliche Beschreibungen von Methoden erzeugt. Es basiert auf der Annahme, dass die Informationen, welche notwendig sind um eine Methode zu beschreiben, oft nicht in der Methode selbst vorkommen, sondern in ihrem unmittelbaren Kontext. In ihrer Arbeit wird dieser Kontext auf die Datei beschränkt, in der die Methode definiert wurde. Konkret besteht der Kontext aus dem Quelltext aller anderen Methoden die in derselben Datei definiert sind, wie die betrachtete Methode. Ein kontextsensitives Verfahren wie es von Haque *et al.* vorgeschlagen wurde ist auch als Grundlage für die Implementierung für INDIRECT denkbar. Zu beachten ist hierbei jedoch, dass im Rahmen von INDIRECT Beschreibungen von Quelltextgruppierungen gesucht werden, nicht für einzelne Methoden. Analog zu dem Verfahren von Haque *et al.* könnte für jedes Quelltextelement einer Gruppierung eine Zusammenfassung erzeugt werden, wobei der Kontext aus dem Quelltext aller anderen Quelltextelemente der Gruppierung besteht. In einem weiteren Schritt können diese Zusammenfassungen genutzt werden, um eine Beschreibung der Gruppierung als Ganzes zu erzeugen.

Falls der unmittelbar in einer Gruppierung vorliegende Kontext nicht ge-

nügend Informationen liefert, um diese zu beschreiben können externe Wissensquellen wie Wikipedia oder WordNet zur Kontextgewinnung eingesetzt werden. Mithilfe dieser externen Quellen können eventuell Verbindungen zu abstrakten Konzepten hergestellt werden, welche in der Gruppierung nur implizit vorliegen.

Das Vorgehen der Arbeit wird es sein zu prüfen, welcher dieser Ansätze besser auf INDIRECT anwendbar ist und welche Repräsentation der Absichtsbeschreibungen am besten für weitere Verarbeitung geeignet ist.

4 Evaluation

Das Ziel der Evaluation ist es die Präzision und Aussagekraft der erzeugten Beschreibungen zu überprüfen. Verfahren zur Themenanalyse sind schwer automatisch zu evaluieren. Das liegt daran, dass die „optimale“ Beschreibung eines Themas sehr subjektiv ist und diese Optimalität somit nicht einfach quantifizierbar ist. Aus diesem Grund findet die Evaluation von Arbeiten in den Bereichen der Themenanalyse und Quelltextzusammenfassung häufig als Nutzerstudie statt.

Im Bereich der klassischen Themenanalyse ist der Wort-Störungs-Test (engl. *word intrusion test*) eine beliebte Form der Nutzerstudie [MB17]. Der Nutzer erhält eine Liste von n Worten, welche aus der Beschreibung eines Dokuments stammen. Zusätzlich wird ein weiteres, für die beschriebene Absicht irrelevantes, Wort hinzugefügt. Der Nutzer muss nun entscheiden welches der $n + 1$ Worte semantisch nicht zu den anderen Worten der Beschreibung passt. Der Test basiert auf der Annahme, dass ein gutes Modell Beschreibungen mit hohem inneren Zusammenhang liefert und es dem Nutzer somit leicht fallen sollte das irrelevante Wort zu identifizieren.

Eine andere Möglichkeit eine Nutzerstudie durchzuführen ist es dem Nutzer eine Quelltextgruppierung vorzulegen, für welche er eine geeignete Beschreibung erstellt. Diese nutzergenerierte Beschreibung wird dann mit der von INDIRECT erzeugten Beschreibung verglichen. Zu beachten ist hierbei, dass der Nutzer, welcher die Beschreibungen erstellt, eventuell über ein gewisses Domänenwissen verfügen muss, um die Quelltextgruppierung adäquat zu beschreiben.

Sollte ein Datensatz mit Quelltextgruppierungen sowie deren Beschreibungen vorliegen können auch die in der maschinellen Übersetzung beliebten BLEU (bilingual evaluation understudy) und ROUGE (recall-oriented understudy for gisting evaluation) Metriken eingesetzt werden. Diese Metriken geben ursprünglich die semantische Ähnlichkeit von maschinell übersetzten Sätzen zu deren tatsächlichen Übersetzung an. Interpretiert man allerdings Quelltext als Ausgangssprache und die natürlichsprachliche Beschreibung von diesem als Zielsprache können diese Metriken auch für Themenanalyseverfahren verwendet werden [HLWM20]. Ein solcher Datensatz müsste in

der benötigten Form manuell erstellt werden, was einen nicht zu vernachlässigenden Arbeitsaufwand mit sich bringt.

Literatur

[BMP13] Bouillon, Elke ; Mäder, Patrick ; Philippow, Ilka: A Survey on Usage Scenarios for Requirements Traceability in Practice. In: Doerr, Joerg (Hrsg.) ; Opdahl, Andreas L. (Hrsg.): *Requirements Engineering: Foundation for Software Quality*. Berlin, Heidelberg : Springer, 2013 (Lecture Notes in Computer Science). – ISBN 978–3–642–37422–7, S. 158–173

[Eur20] Eurich, Felix: *Entwurf und Aufbau einer semantischen Repräsentation von Quelltext*, Karlsruher Institut für Technologie (KIT) – IPD Tichy, Master's Thesis, 2020. `https://code.ipd.kit.edu/hey/indirect/wikis/Theses/eurich_ma`

[Hey19] Hey, Tobias: INDIRECT: Intent-Driven Requirements-to-Code Traceability. In: *2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*. Montreal, QC, Canada : IEEE, Mai 2019. – ISBN 978–1–72811–764–5, S. 190–191

[HLWM20] Haque, Sakib ; LeClair, Alexander ; Wu, Lingfei ; McMillan, Collin: Improved Automatic Summarization of Subroutines via Attention to File Context. In: *arXiv:2004.04881 [cs]* (2020), April. `http://dx.doi.org/10.1145/3379597.3387449`. – DOI 10.1145/3379597.3387449

[MB17] Mahmoud, Anas ; Bradshaw, Gary: Semantic Topic Models for Source Code Analysis. In: *Empirical Software Engineering* 22 (2017), August, Nr. 4, S. 1965–2000. `http://dx.doi.org/10.1007/s10664-016-9473-1`. – DOI 10.1007/s10664–016–9473–1. – ISSN 1382–3256, 1573–7616

[MN15] Mahmoud, Anas ; Niu, Nan: On the Role of Semantics in Automated Requirements Tracing. In: *Requirements Engineering* 20 (2015), September, Nr. 3, S. 281–300. `http://dx.doi.org/10.1007/s00766-013-0199-y`. – DOI 10.1007/s00766–013–0199–y. – ISSN 1432–010X