

**Gottfried Wilhelm
Leibniz Universität Hannover
Fakultät für Elektrotechnik und Informatik
Institut für Praktische Informatik
Fachgebiet Software Engineering**

Automatische Erstellung von Erklärbarkeitsanforderungen und Erklärungen

Masterarbeit

im Studiengang Informatik

von

Steffen Krätzig

**Prüfer: Prof. Dr. Kurt Schneider
Zweitprüfer: Dr. Jil Ann-Christin Klünder
Betreuer: Martin Obaidi**

Hannover, 26. April 2024

Erklärung der Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Masterarbeit selbständig und ohne fremde Hilfe verfasst und keine anderen als die in der Arbeit angegebenen Quellen und Hilfsmittel verwendet habe. Die Arbeit hat in gleicher oder ähnlicher Form noch keinem anderen Prüfungsamt vorgelegen.

Hannover, den 26.04.2024

Steffen Krätzig

Zusammenfassung

Diese Masterarbeit entwickelt ein Konzept zur automatischen Generierung von Erklärbarkeitsanforderungen und Erklärungen basierend auf Nutzer-Reviews. Ein fundierter Datensatz wird durch Studie mit Anforderungsanalysten aus der Praxis erstellt, die spezifische Erklärbarkeitsanforderungen und Erklärungen zu einzelnen Reviews definieren. Das erarbeitete Konzept nutzt Erkenntnisse aus diese, Datensatz, erarbeitet die erforderlichen Schritte zur Automatisierung und implementiert diese in einer Software. Diese Software befähigt Anforderungsanalysten, effizient und effektiv Zusammenfassungen sowie Erklärbarkeitsanforderungen und Erklärungen zu generieren. Ihre Anwendbarkeit und Wirksamkeit werden durch eine nachfolgende Studie bestätigt, die sowohl die die Nutzung der Software als auch die erzielten Ergebnisse evaluiert. Besonders hervorzuheben ist die positive Resonanz von praktizierenden Anforderungsanalysten, die die Software als wesentliche Unterstützung mit anschließender Verfeinerung der Ergebnisse ansehen. Diese Arbeit trägt signifikant zur Steigerung der Transparenz und Verständlichkeit in der Softwareerklärbarkeit bei.

Abstract

This master's thesis develops a concept for the automatic generation of explainability requirements and explanations based on user reviews. A dataset is created through a study involving requirement analysts from the industry, who define specific explainability requirements and explanations for user reviews. The developed concept leverages insights from this dataset and outlines the necessary steps for automation. These are implemented for automation in a software. This software enables requirements analysts to efficiently and effectively generate summaries, explainability requirements, and explanations based on user reviews. The applicability and effectiveness of the software are validated by a subsequent study that evaluates both the use of the software and the results achieved. Particularly noteworthy is the positive response from practicing requirements analysts, who view the software as an essential tool that facilitates the refinement of the results. This work significantly contributes to enhancing the transparency and understandability in software explainability.

Inhaltsverzeichnis

1	Einleitung.....	1
1.1	Problemstellung	1
1.2	Lösungsansatz	2
1.3	Struktur der Arbeit	4
2	Grundlagen.....	5
2.1	Erklärbarkeitsbegriff.....	5
2.2	Abgrenzung zur KI-Erklärbarkeit.....	6
2.2.1	Software-Erklärbarkeit	6
2.2.2	Künstliche Intelligenz (KI)-Erklärbarkeit	6
2.2.3	Gemeinsamkeiten	6
2.2.4	Fazit.....	7
2.3	Erklärbarkeit im Requirements Engineering	7
2.4	Relevanz von Erklärbarkeitsanforderungen und Erklärungen	8
2.4.1	Verbesserung der Nutzererfahrung und Akzeptanz	8
2.4.2	Reduzierung von Fehlern und Missverständnissen	8
2.4.3	Einhaltung rechtlicher und regulatorischer Anforderungen	9
2.4.4	Kontinuierliche Anpassung an verändernde Anforderungen.....	9
2.5	Datensatz: Erklärungsbedarf in Reviews	9
2.6	Technische Umsetzung	10
2.6.1	NLP.....	10
2.6.2	Generative pre-trained Transformer (GPT).....	11
3	Verwandte Arbeiten.....	12
3.1	Grundlagen Erklärbarkeitsbedarf und Erklärungen.....	12
3.2	Erkennen von Erklärbarkeitsbedarf.....	12
3.3	Weitere Arbeiten zu Erklärbarkeit.....	13
3.4	Automatisierung im Software-Engineering.....	13
3.5	Abgrenzung	13
4	Studie: Formulierung von Erklärbarkeitsanforderungen und Erklärungen	14
4.1	Ziel.....	14
4.2	Methodik	14
4.3	Auswahl der Reviews	15
4.4	Teilnehmer.....	15
4.5	Erwartete Ergebnisse	16
4.6	Durchführung	16
4.7	Formulierung von Anforderungen.....	17
4.8	Analyse der Ergebnisse	17
4.8.1	Nutzen für Anforderungen	17
4.8.2	Nutzen für Erklärungen.....	18
4.9	Reflektion.....	19
4.9.1	Ziele.....	19
4.9.2	Workshop Design und Durchführung.....	19

4.9.3	Teilnehmerfeedback.....	19
5	Konzept.....	20
5.1	Grundsätzliche Ansätze zur Erstellung.....	20
5.1.1	Mapping Table.....	20
5.1.2	Schablone.....	22
5.1.3	Dynamischer Ansatz.....	23
5.1.4	Hybrider Ansatz.....	24
5.2	Zusammenfassung der Ansätze.....	26
5.3	Beurteilung der Aspekte zur Automatisierung.....	27
5.3.1	Zusammenfassung von Reviews.....	27
5.3.2	Erstellung von Erklärbarkeitsanforderungen.....	28
5.3.3	Erstellung von Erklärungen.....	28
5.4	Fazit.....	29
6	Umsetzung der Automatisierung.....	30
6.1	Stakeholder.....	30
6.1.1	Anforderungsanalysten.....	30
6.1.2	Softwareentwickler.....	30
6.1.3	Nutzer der Software.....	30
6.2	Anforderungen erfassen.....	31
6.2.1	Funktionale Anforderungen.....	31
6.2.2	Nicht funktionale Anforderungen.....	33
6.2.3	Optionale Anforderungen.....	33
6.3	Priorisierung der Anforderungen.....	34
6.4	Entwicklung.....	35
7	Studie: Evaluation der Automatisierung.....	38
7.1	Ziel.....	38
7.2	Methodik.....	38
7.3	Auswahl der Reviews.....	40
7.4	Teilnehmer.....	40
7.5	Erwartete Ergebnisse.....	40
7.6	Durchführung.....	41
7.7	Analyse der Ergebnisse.....	42
7.7.1	Kennzahlen des erstellten Datensatzes.....	42
7.7.2	Analyse bei eindeutigen Entscheidungen.....	44
7.7.3	Analyse bei unterschiedlichen Entscheidungen.....	47
7.7.4	Analyse der praktischen Nutzung.....	48
7.7.5	Fazit.....	49
7.8	Reflektion.....	50
7.8.1	Ziele.....	50
7.8.2	Workshop Design und Durchführung.....	51
7.8.3	Teilnehmerfeedback.....	52
8	Diskussion.....	53

8.1	Beantwortung der Forschungsfragen.....	53
8.2	Interpretation	56
9	Threats to validity.....	57
9.1	Construct Validity	57
9.2	Internal Validity	57
9.3	External Validity.....	58
9.4	Reliability	58
10	Zusammenfassung und Ausblick	59
10.1	Zusammenfassung	59
10.2	Ausblick.....	60

1 Einleitung

In der modernen Softwareentwicklung nimmt die Komplexität von Systemen stetig zu, wodurch die Nachfrage nach höherer Transparenz und Verständlichkeit ebenfalls wächst. Erklärbarkeit, besonders in Bereichen wie Künstliche Intelligenz und Datenanalyse, ist von kritischer Bedeutung, um das Vertrauen der Nutzer zu gewinnen [1] und regulatorischen Anforderungen gerecht zu werden [2–4]. Nutzer-Reviews bieten eine reichhaltige Quelle für das Verständnis von Benutzererwartungen und deren Schwierigkeiten mit Softwareprodukten. [5] Trotz des klaren Bedarfs, diese Rückmeldungen systematisch auszuwerten, fehlen oft die Werkzeuge, um aus unstrukturierten Daten wie Reviews automatisch nutzbare Erklärbarkeitsanforderungen abzuleiten.

Diese Masterarbeit stellt ein Konzept vor, das die automatische Generierung von Erklärbarkeitsanforderungen und Erklärungen aus Nutzer-Reviews ermöglicht. Das Ziel ist es, eine Brücke zwischen den unstrukturierten Daten aus Nutzer-Feedback und den strukturierten Anforderungen an Softwareerklärbarkeit zu schlagen, sowie der Versuch einer Angabe der entsprechenden Erklärung. Hierfür wird zunächst ein fundierter Datensatz durch die Zusammenarbeit mit Anforderungsanalysten aus der Praxis erstellt, welche spezifische Anforderungen und Erklärungen direkt aus den Reviews ableiten. Dies bildet die Grundlage für die Entwicklung eines automatisierten Systems, das nicht nur effizient, sondern auch effektiv in der praktischen Anwendung ist.

1.1 Problemstellung

Ein wichtiger Aspekt einer Software ist die Akzeptanz durch die Nutzer. [6] Die Akzeptanz wird durch Reduzierung des bestehenden Erklärbarkeitsbedarfs seitens des Nutzers erhöht. Eine umfassende Analyse auf Erklärbarkeitsbedarf scheint durch die fortlaufende Generierung von Erklärbarkeitsbedarf in Reviews nahezu unmöglich. Dies führt fortlaufend zu weniger Akzeptanz und Nutzung der Software. Ein Prozess zur automatischen Generierung von Erklärbarkeitsanforderungen und Erklärungen auf Grundlage der Reviews würde diesem Problem entgegenwirken und eine höhere Softwarequalität und Nutzerzufriedenheit sicherstellen. Basierend auf dieser Problematik ergeben sich die folgenden Forschungsfragen.

[FF1] Wie kann ein Prozess definiert werden, der effizient Erklärbarkeitsanforderungen und Erklärungen auf Grundlage von Reviews generiert?

[FF2] Wie hilft das Wissen über die Kategorie eines Erklärbarkeitsbedarfs bei der Formulierung von Anforderungen?

[FF3] Wie hoch ist die Akzeptanz der Experten von automatisch generierten Erklärbarkeitsanforderungen?

[FF4] Wie können automatisch generierte Erklärungen in der Praxis genutzt werden?

[FF5] Wie kann die Generierung von Erklärbarkeitsanforderungen automatisiert werden?

[FF6] Wie kann die Generierung von Erklärungen automatisiert werden?

1.2 Lösungsansatz

Um die Forschungsfragen zu beantworten, muss vorerst mittels einer Studie ein nutzbarer Datensatz erstellt werden, der zu Reviews, die Erklärungsbedarf beinhalten, passende Erklärbarkeitsanforderungen und Erklärungen enthält. Diese Erklärbarkeitsanforderungen sollen Grundlage für die Entwicklung einer Automatisierungslösung sein. Anschließend sollen die Ergebnisse der Automatisierung und die zu entwickelnde Software anhand einer weiteren Studie evaluiert werden. So können Rückschlüsse auf das erstellte Konzept und den Nutzen geschlossen werden.

Um einen Datensatz von Erklärbarkeitsanforderungen zu erstellen, soll eine Studie durchgeführt werden. Das Ziel dieser Studie ist die Formulierung von Erklärbarkeitsanforderungen und Erklärungen durch Anforderungsanalysten auf Grundlage von ausgewählten Reviews. Im Übrigen ist zu untersuchen, welche Technik genutzt werden soll, um den Inhalt der Reviews und die darin enthaltenen Erklärbarkeitsbedarf automatisiert zu verstehen und zu formulieren.

In einem weiteren Schritt soll ein Tool entwickelt werden, das das erstellte Konzept umsetzt. Als Input für das Tool werden erklärungsbedürftige Reviews auf Standard des gegebenen Datensatzes von Kupczyk[7]

erwartet. Das Tool soll den Erklärungsbedarf der Reviews analysieren und passende Erklärbarkeitsanforderungen und Erklärungen generieren. Die Ergebnisse können von den Nutzern nachträglich eingesehen und verändert werden.

Diese semi-automatische Auswahl der generierten Erklärbarkeitsanforderungen und Erklärungen soll dabei einen Mehrwert gegenüber der vollautomatisierten Generierung bieten. So wird der Handlungsspielraum des Anwenders erweitert und damit eine qualitative Überprüfung ermöglicht.

Um die Ergebnisse des entwickelten Tools zu validieren, soll eine weitere Studie mit Anforderungsanalysten durchgeführt werden. Hierbei sollen die Anforderungsanalysten die Ergebnisse und die Nutzung bewerten.

Der Kontext der Arbeit ist wie folgt zu beschreiben:

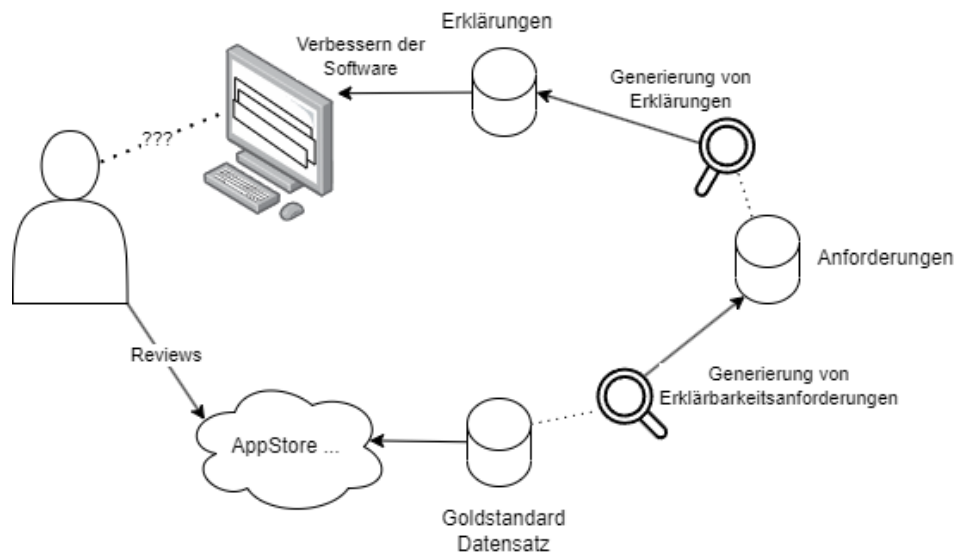


Abbildung 1: Überblick über den Zusammenhang [Eigene Darstellung]

Sobald ein Nutzer einen Erklärungsbedarf in der Software feststellt, wird er seine Erfahrung in einem Review zusammenfassen, das in den gängigen Plattformen zur Bereitstellung dieser Software, wie z.B. dem Appstore, gespeichert wird. Der Datensatz von Kupczyk [7], sowie seine GUI Erweiterung für das Tool von Kurtz [8], filtert die Reviews mit Erklärungsbedarf in einen eigenen Datensatz. Diese Reviews sollen analysiert und Erklärbarkeitsanforderungen generiert werden. Sobald die Anforderungen existieren, können passende Erklärungen darauf basierend generiert werden. Die Erklärungen können wiederum genutzt werden, um die Software zu verbessern.

In dieser Arbeit liegt der Fokus auf der Analyse und Generierung von Erklärbarkeitsanforderungen und Erklärungen.

Die Arbeit erarbeitet also einen Teil eines Prozesses zur Verbesserung von Software bereit.

1.3 Struktur der Arbeit

Diese Arbeit ist wie folgt strukturiert. In Kapitel 2 werden die Grundlagen bezüglich Erklärbarkeit im Kontext des Requirements Engineerings, sowie die Taxonomie von Erklärbarkeitsbedarf nach Droste et al. [9]. Betrachtet wird auch das notwendige technische Wissen über verwendete Methoden.

Anschließend werden in Kapitel 3 Verwandte Arbeiten aufgeführt und eine Abgrenzung zu diesen beschrieben.

In Kapitel 4 wird die Vorbereitung und Durchführung der ersten Studie beschrieben zur Erstellung eines Datensatzes von Erklärbarkeitsanforderungen und Erklärungen basierend auf Reviews. Die Ergebnisse werden analysiert und grafisch aufbereitet dargestellt. Ansätze zu einem Konzept werden zum Abschluss analysiert und die Studie reflektiert.

Nachfolgend wird in Kapitel 5 ein passendes Konzept zur Automatisierung der Formulierung von Erklärbarkeitsanforderungen und Erklärungen betrachtet. Dabei werden unter anderem die Ergebnisse der Studie aus Kapitel 4 genutzt.

Anschließend wird in Kapitel 6 die Umsetzung des Konzeptes durch Entwicklung einer Software beschrieben. Dabei wird die Anforderungsanalyse genauer erläutert.

In Kapitel 7 wird die Software und die Ergebnisse dieser durch eine weitere Studie in Form eines Workshops mit Anforderungsanalysten evaluiert. Die Ergebnisse der Studie werden analysiert und letztlich eine Reflektion der Studie durchgeführt.

Nach der Erfassung der Ergebnisse werden in Kapitel 8 diese diskutiert und interpretiert.

Zum Ende der Arbeit werden in Kapitel 9 die „Threats to Validity“ nach Wohlin et al. [10] analysiert und abschließend in Kapitel 10 eine Zusammenfassung und ein Ausblick der Arbeit gegeben.

2 Grundlagen

Zunächst wird der Begriff Erklärbarkeit näher betrachtet. Anschließend wird die Software-Erklärbarkeit zur KI-Erklärbarkeit abgegrenzt. Dabei werden Gemeinsamkeiten und Unterschiede herausgearbeitet. Anschließend werden Grundlagen zum Requirements Engineering aufgeführt und die Bedeutung der Erklärbarkeit in diesem Kontext. Die verwendete Taxonomie von Erklärbarkeitsbedarf wird erläutert, sowie ein grundlegendes technisches Verständnis für die verwendeten Techniken.

2.1 Erklärbarkeitsbegriff

Es existieren verschiedene Definitionen über Erklärbarkeit. Für ein grundlegendes Verständnis von Erklärbarkeit soll in diesem Abschnitt der Erklärbarkeitsbegriff durch maßgebende Meinungen aus der Literatur zusammengefasst werden.

Eine einheitliche Definition von Erklärungen zu bieten ist weiterhin schwierig, da die Anwendungsbereiche und Kontexte nicht gleich betrachtet werden können. Zudem kann das Verständnis einer Erklärung nur subjektiv bewertet werden kann. Kriterien für eine objektive Erklärung können aufgrund Ihrer Subjektivität daher nur schwer erfasst werden. So halten sich die Definitionen für Erklärbarkeit von Software sehr abstrakt. Folglich werden mehrere Definitionen für erklärable Software aufgeführt.

Allgemein beschreiben Köhl et al. ein System als erklärbar, wenn die Nutzergruppe Zugriff auf Erklärungen im jeweiligen Kontext erhalten kann [11]. Dieser Zugriff kann in diversen Formen bereitgestellt werden. Somit sind Experten, Nutzerhandbücher oder das System selbst nutzbare Entitäten zur Bereitstellung von Erklärungen.

Köhl et al. [11] entwickeln eine Erweiterung dieser Definition um zu beschreiben, wann ein System erklärbar ist, die von Chazette et al. [12] als Grundlage für ihre folgende Definition genutzt wurde:

„Ein System S ist in Bezug auf einen Aspekt X von S relativ zu einem Adressaten A im Kontext C nur dann erklärbar, wenn es eine Entität E (den Erklärer) gibt, die es A durch Angabe eines Informationskorpus I (die Erklärung von X) ermöglicht, X von S in C zu verstehen.“ [13]

Diese Definition wird in dieser Arbeit als Grundlage für das Verständnis von Erklärbarkeit genutzt.

2.2 Abgrenzung zur KI-Erklärbarkeit

In der Literatur werden die Begriffe Software-Erklärbarkeit und KI-Erklärbarkeit oft verwendet, jedoch nicht immer klar unterschieden. Dieses Kapitel dient dazu, beide Konzepte voneinander abzugrenzen und deren spezifische Anforderungen und Herausforderungen zu erörtern. Diese Abgrenzung ist für die vorliegende Masterarbeit relevant. Sie legt den Fokus auf die Software-Erklärbarkeit.

2.2.1 Software-Erklärbarkeit

Software-Erklärbarkeit, wie sie in der ISO/IEC 25010:2011 Norm für Systeme und Software-Qualitätsmodelle definiert wird, ist ein Aspekt der Benutzbarkeit und somit ein essenzieller Bestandteil der Softwarequalität. Benutzbarkeit bezieht sich dabei auf die Verständlichkeit, Erlernbarkeit und Bedienbarkeit einer Software. Diese Faktoren ermöglichen es den Benutzern, die Funktionsweise und die Entscheidungsprozesse innerhalb der Software zu verstehen und damit zu interagieren. Was eine Software erklärbar macht, kann durch die Definition von Chazette et al. [14] überprüft werden.

2.2.2 Künstliche Intelligenz (KI)-Erklärbarkeit

Im Gegensatz zur allgemeinen Software-Erklärbarkeit konzentriert sich die KI-Erklärbarkeit spezifisch auf die Transparenz und Nachvollziehbarkeit von Entscheidungen, die von künstlicher Intelligenz getroffen werden. [15] Hierbei geht es nicht nur um die Bedienbarkeit oder Erlernbarkeit der Systeme, sondern vor allem um das Verständnis komplexer Algorithmen und Modelle, die in den Entscheidungsfindungsprozessen der KI-Systeme verwendet werden. Diese Erklärbarkeit ist entscheidend, um Vertrauen in KI-basierte Entscheidungen zu fördern und zu gewährleisten, dass Nutzer die Grundlagen der Entscheidungen der KI verstehen und ethische sowie rechtliche Standards eingehalten werden. [16]

2.2.3 Gemeinsamkeiten

Beide Formen der Erklärbarkeit streben danach, die Nutzererfahrung zu optimieren und das Verständnis komplexer Systeme zu fördern. In beiden Fällen ist die klare Kommunikation darüber, wie Entscheidungen zustande kommen, entscheidend für das Vertrauen der Nutzer. Die

Herausforderungen liegen darin, komplexe Informationen zugänglich und verständlich zu machen, was sowohl technisches Know-how als auch ein tiefes Verständnis der Benutzeranforderungen erfordert.

2.2.4 Fazit

Folglich lässt sich sagen, dass Software-Erklärbarkeit und KI-Erklärbarkeit zwar verwandte Konzepte sind, sich jedoch in ihren Zielen und Herangehensweisen unterscheiden. Die vorliegende Masterarbeit trägt zu einem vertieften Verständnis der Software-Erklärbarkeit bei und untersucht, wie ein Prozess erstellt werden kann, der zu einer Verbesserung der Software-Erklärbarkeit beiträgt.

2.3 Erklärbarkeit im Requirements Engineering

Wesentlicher Bestandteil des Requirements Engineerings (RE) ist das Erlangen von geeigneten Anforderungen für Software. Klare Anforderungen sind grundlegend für die erfolgreiche Umsetzung und die Verbesserung einer Software. Hierbei werden zwei Arten von Anforderungen unterschieden:

- Funktionale Anforderungen
- Qualitative Anforderungen

Die funktionalen Anforderungen stellen die Funktionen und Aktionen dar, die eine Software unterstützen soll. Wobei die qualitativen Anforderungen die qualitativen Aspekte einer Software beschreiben. In der Abbildung 2: Qualitätsaspekte aus ISO/IEC25010 werden die folgenden Kategorien den qualitativen Anforderungen zugeordnet:



Abbildung 2: Qualitätsaspekte aus ISO/IEC25010 [17]

Die Kategorie der Usability beinhaltet Themen, in denen Erklärbarkeit einen wesentlichen Beitrag leistet. Die Erlernbarkeit, Fehlervermeidung, sowie die Zugänglichkeit werden durch Erklärungen innerhalb der Software gestärkt. Somit lassen sich Erklärbarkeitsanforderungen unter den qualitativen Anforderungen einordnen.

Durch die Erklärungen wird die Software erlernbar, weil Unklarheiten der Software aufgedeckt und aufgelöst werden können. Dadurch wird die Anwendbarkeit der Software verbessert welches zur Qualität dieser beiträgt.

2.4 Relevanz von Erklärbarkeitsanforderungen und Erklärungen

Die Betrachtung der Erklärbarkeitsanforderungen bei der Anforderungsanalyse ist ein wichtiger Aspekt zur Sicherung der Qualität einer Software. Erst durch das Erfassen der Anforderungen können Erklärungen formuliert und integriert werden. Nachfolgend werden wichtige Themen genannt, auf die die Erfassung von Erklärbarkeitsanforderungen einen wesentlichen Einfluss hat.

2.4.1 Verbesserung der Nutzererfahrung und Akzeptanz

Die Erfassung von Erklärbarkeitsanforderungen und die Integration entsprechender Erklärungen dienen dazu, die Nutzererfahrung zu verbessern und die Akzeptanz der Softwarelösung zu erhöhen. Einige wissenschaftliche Arbeiten [18,19] haben gezeigt, dass Benutzer eine höhere Zufriedenheit und Vertrauenswürdigkeit in Softwarelösungen aufweisen, wenn sie Erklärungen für die Funktionsweise und Entscheidungen der Software erhalten. Dies führt zu einer erhöhten Benutzerakzeptanz und kann langfristig die Effektivität und Effizienz der Softwareanwendung steigern. [20]

2.4.2 Reduzierung von Fehlern und Missverständnissen

Die Integration von klaren und verständlichen Erklärungen kann dazu beitragen, Fehler und Missverständnisse zu reduzieren. Die Nutzer sind in der Lage, die Funktionsweise und Entscheidungen der Software zu verstehen, wodurch das Risiko von Fehlinterpretationen und falschen Annahmen verringert wird.

2.4.3 Einhaltung rechtlicher und regulatorischer Anforderungen

In vielen Branchen und Anwendungsbereichen gibt es rechtliche und regulatorische Anforderungen, die die Transparenz und Erklärbarkeit von Software vorschreiben. Dazu zählen unter anderen das Gesundheitswesen [2], Finanzwesen [3], sowie das Rechtswesen und öffentliche Verwaltung [4]. Durch die Erfassung von Erklärbarkeitsanforderungen und die Integration entsprechender Erklärungen können Entwickler sicherstellen, dass die Software den geltenden rechtlichen und regulatorischen Anforderungen entspricht.

2.4.4 Kontinuierliche Anpassung an verändernde Anforderungen

Die Erfassung von Erklärbarkeitsanforderungen ist ein kontinuierlicher Prozess, der sich im Laufe der Entwicklung und Nutzung einer Software fortsetzt. Die Anforderungen an die Erklärbarkeit können sich im Zeitverlauf ändern, da sich die Software weiterentwickelt und neue Anforderungen von Stakeholdern auftreten. Daher ist es wichtig, dass Erklärbarkeitsanforderungen kontinuierlich überprüft und angepasst werden, um sicherzustellen, dass die Software weiterhin transparente und nachvollziehbare Erklärungen bietet. Diese kontinuierliche Anpassung trägt dazu bei, die Qualität der Softwarelösung aufrechtzuerhalten und sicherzustellen, dass sie den sich ändernden Anforderungen und Benutzererwartungen gerecht wird.

Diese Themengebiete haben bestärkt, dass die Erfassung von Erklärbarkeitsanforderungen und Erklärungen einen positiven Einfluss auf die Entwicklung einer Software haben und stetig Erklärbarkeitsanforderungen erfasst werden sollten. Der Bedarf einer Automatisierungslösung wird dadurch noch deutlicher.

2.5 Datensatz: Erklärungsbedarf in Reviews

Als Grundlage für die Generierung von Erklärbarkeitsanforderungen wird ein Datensatz verwendet, der in einer vorhergehenden Arbeit erstellt und analysiert wurde [7]. Dieser enthält insgesamt etwa 4500 Reviews aus diversen Apps, die mit Erklärungsbedarf identifiziert wurden. Diese Reviews werden folgend für die Generierung von Erklärbarkeitsanforderungen und Erklärungen genutzt.

Der Datensatz unterscheidet die Reviews grundsätzlich nach implizitem und explizitem Erklärungsbedarf. Weiterhin werden die Reviews genauer

nach einer festgelegten Taxonomie von Erklärungsbedarf, wie in der folgenden Abbildung 3 zu sehen, eingeordnet.

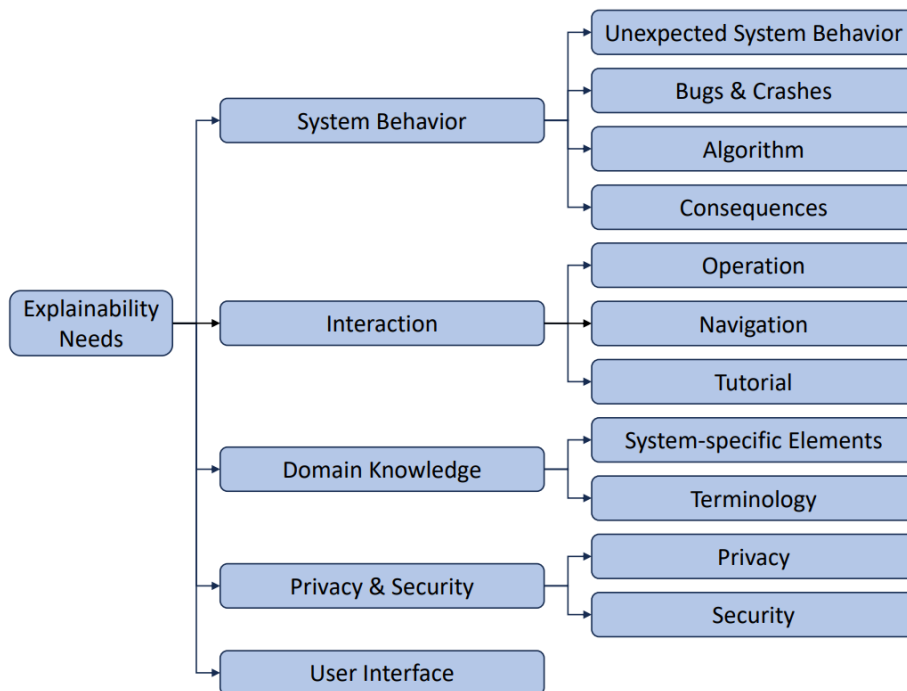


Abbildung 3: Taxonomie von Erklärungsbedarf in alltäglicher Software [9]

Eine Erklärung der einzelnen Kategorien kann aus Kupczyk [7] und Droste et al. [9] entnommen werden.

Die Taxonomie wird für die Erstellung eines neuen Datensatzes genutzt, um alle Kategorien von Erklärbarkeitsbedarf abzudecken.

2.6 Technische Umsetzung

In diesem Kapitel wird das technische Wissen für die verwendeten Technologien ausgeführt.

2.6.1 NLP

Techniken aus dem Bereich des Natural Language Processing (NLP) beschäftigen sich mit der Analyse von natürlicher menschlicher Sprache, um die Grammatik und Zusammenhänge innerhalb eines Satzes zu verstehen. [21] Diese Techniken beschäftigen sich hauptsächlich mit der Analyse, Generierung und Nutzung von Sprachmodellen. Somit ist dies ein wesentliches Gebiet für eine alltagstaugliche Interaktion zwischen Menschen und Software. [22]

2.6.2 Generative pre-trained Transformer (GPT)

Der Bereich des NLP wurde durch die Entwicklung von GPT-Modellen eine leistungsstarke Möglichkeit zur Generierung von Text erweitert. Diese sind eine Weiterentwicklung der Transformer-Architektur, die erstmals von Vaswani et al. [23] vorgestellt wurde. Im Wesentlichen nutzt die Transformer-Architektur so genannte „Attention-Mechanism“, die sinngemäßen Beziehungen zwischen Wörtern herstellen können.

In der folgenden Abbildung 4 von Vaswani et al. werden die erkannten Beziehungen zwischen den Wörtern eines Satzes auf einer Ebene dargestellt. [23]

Attention Visualizations

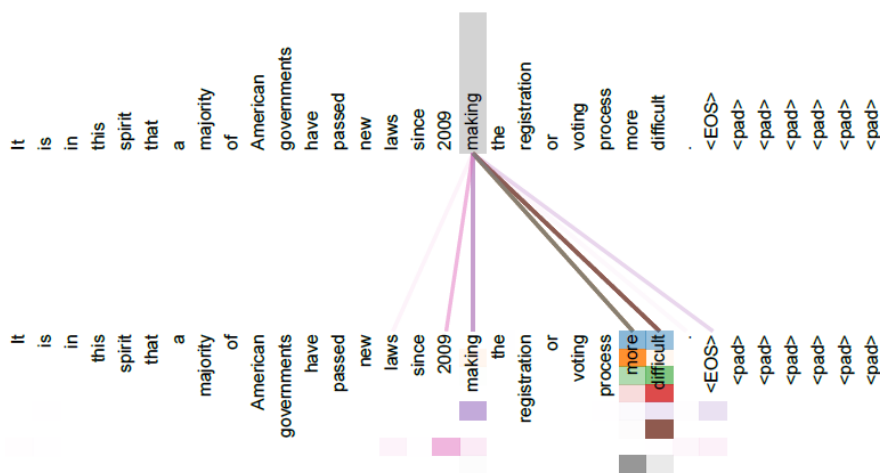


Abbildung 4: Beispiel Visualisierung des Attention-Mechanism [23]

Die farbigen Linien stellen die Beziehung der anderen Wörter zu dem Wort „making“ dar.

GPT-Modelle nutzen diese Mechanismen, um die Eingaben effektiver zu verstehen und gleichzeitig darauf basierend passende Texte zu generieren. Das Verständnis der Sprache wird durch das „Pre-Training“ mit einem Sprachmodell erreicht. [24]

Durch die Fähigkeit den Fokus auf verschiedene Inhalte der Eingabe zu setzen, erreichen GPT-Modelle eine deutlich höhere Effizienz in der Verarbeitung von Eingaben als Menschen. Diese Eigenschaft ist ein wichtiger Aspekt zur Automatisierung von Prozessen, die Fokus auf einen größeren Kontext und Verständnis benötigen.

3 Verwandte Arbeiten

3.1 Grundlagen Erklärbarkeitsbedarf und Erklärungen

Maalej und Nabil [5] beschreiben in ihrer Arbeit das Potenzial Reviews als Quelle von Erklärbarkeitsbedarf zu nutzen und erarbeiten eine Methode zur Klassifizierung des Erklärbarkeitsbedarfs in verschiedene Kategorien.

Droste et al. [25] entwickeln eine Taxonomie für Erklärbarkeitsbedarf. Weiterhin stellen sie Tools für die Erfassung von Erklärbarkeitsbedarf in Reviews bereit, sowie einen Datensatz, der Reviews von Apps enthält, die mit der entsprechenden Taxonomie gekennzeichnet wurden.

Chazette et al. [14] haben durch eine Literaturstudie untersucht wie Erklärbarkeit definiert werden kann und schlagen mit ihrer Arbeit eine Definition, ein Modell und einen Wissenskatalog von Erklärbarkeit vor.

Deters et al. [26] untersuchen in ihrer Arbeit die Bewertung von Erklärbarkeit in Softwaresystemen. Sie kombinieren die Methode der Expertenbewertung mit zielorientierten Heuristiken, die es ermöglichen, Erklärungen effektiv zu evaluieren.

3.2 Erkennen von Erklärbarkeitsbedarf

Kurtz [8] entwickelt in seiner Arbeit ein Tool, um Reviews aus dem App und Playstore zu crawlen. Diese Reviews können nach verschiedenen Kriterien gefiltert und angezeigt werden. Ein Ansatz zur Filterung der Reviews auf Erklärbarkeitsbedarf ist ebenfalls implementiert.

Kupczyk [7] beschreibt in seiner Arbeit Möglichkeiten, um Erklärbarkeitsbedarf in Reviews von Apps zu erkennen. Des Weiteren werden die angewandten Methoden beschrieben, um einen Datensatz zu erstellen, der erklärungsbedürftige Reviews enthält. Dieser Datensatz wurde für diese Arbeit zur Verfügung gestellt.

Droste et al. [25] zeigen in ihrer Arbeit ein Konzept, um Erklärbarkeitsanforderungen aus Sicht der Nutzer zu erhalten. Dafür wurde eine Studie mit 70 Teilnehmern durchgeführt, um aus den Ergebnissen Personas zu erstellen. Personas sind wenige fiktive

Charaktere, denen aggregierte Eigenschaften der Studienteilnehmer zugeordnet werden. Das Erlangen der Erklärbarkeitsanforderungen ist möglich durch ein simuliertes Verhalten einer Persona bei Nutzung der Software. Durch die charakterlichen Besonderheiten hat jede Persona einen anderen Blick auf die Software und erlangt unterschiedliche Erklärbarkeitsanforderungen.

3.3 Weitere Arbeiten zu Erklärbarkeit

Bohnstedt [27] führt in der Arbeit eine Onlinestudie zur Erklärbarkeit von Software durchgeföhrt, die erforscht, welche Software die Nutzer häufig verwenden, wie sie ihr eigenes Domänenwissen einschätzen und welche Erwartungen diese an Erklärungen haben. Durch tiefergehende Analysen wird der Zusammenhang zwischen Domänenwissen und Erklärbarkeitsbedarf untersucht.

Tsakalakis et al. [28] beschreiben ihre Entwicklung der Taxonomie für Erklärungen, um Erklärbarkeit von Software anhand von dessen Design zu fördern.

3.4 Automatisierung im Software-Engineering

Im Bereich Sentiment Analysis beschreiben Obaidi und Klünder [29] welche Methoden zur Automatisierung der Sentiment Analysis genutzt werden.

3.5 Abgrenzung

Diese Arbeit grenzt sich von den genannten Arbeiten wie folgt ab.

Das Erkennen von Erklärbarkeitsbedarf in Reviews, mit denen sich Kurtz [8] und Kupczyk [7] beschäftigten, ist ein Teil eines Prozesses, wie er in Abbildung 1 dargestellt ist, der die Software, auf die sich die Reviews bezieht, verbessern soll. Die Automatisierung zur Erstellung von Erklärbarkeitsbedarf und Erklärungen auf Grundlage von Reviews ist ein wesentlicher Schritt des Prozesses, der in den genannten Arbeiten noch nicht betrachtet wurde und einen Beitrag zur Verbesserung des Requirements Engineerings bietet.

4 Studie: Formulierung von Erklärbarkeitsanforderungen und Erklärungen

4.1 Ziel

Diese Studie verfolgt zwei Hauptziele:

1. **Formulierung von Erklärbarkeitsanforderungen und Erklärungen**
Anforderungsanalysten sollen auf Basis eines gegebenen Datensatzes von Reviews Erklärbarkeitsanforderungen und Erklärungen formulieren. Die gesamte Taxonomie von Erklärbarkeitsbedarf (siehe Kapitel 2.5) soll durch die gegebenen Reviews abgedeckt werden, um einen möglichst vielfältigen und nutzbaren Datensatz zu erstellen.
2. **Ansatz zur Automatisierung der Erklärbarkeitsanforderungen und Erklärungen**
Die Formulierungen der Anforderungsanalysten sollen genutzt werden, um ein Konzept für ein Konzept zur Automatisierung der Erklärbarkeitsanforderungen und Erklärungen zu entwickeln und dieses umzusetzen.

4.2 Methodik

In diesem Abschnitt wird der gewählte Ansatz zur Erreichung des ersten Hauptziels, der Formulierung von Erklärbarkeitsanforderungen und Erklärungen, dargelegt. Die Güte der Ergebnisse hängt entscheidend von der gewählten Methode ab. Hierbei wird zwischen einem quantitativen und einem qualitativen Ansatz unterschieden.

Quantitativer Ansatz: Der quantitative Ansatz ermöglicht durch die Erhebung und Analyse einer hohen Anzahl von Daten die Identifikation von Mustern und Trends. Allerdings zu Lasten der Präzision der einzelnen Anforderungen und Erklärungen, obwohl die große Datenmenge allgemeine Aussagen fördert.

Qualitativer Ansatz: Im Gegensatz dazu fokussiert der qualitative Ansatz auf die detaillierte Erhebung und Analyse einer kleineren Anzahl von

Anforderungen und Erklärungen. Die iterative Verfeinerung und enge Diskussion mit den Anforderungsanalysten gewährleisten eine hohe Präzision der Ergebnisse.

Obwohl das Erkennen von Mustern einen quantitativen Ansatz nahelegt, wurde, mit Hinblick auf die Verfügbarkeit von praktizierenden Anforderungsanalysten, für diese Studie der qualitative Ansatz in Form eines angeleiteten Workshops gewählt. Dieser Ansatz soll es ermöglichen, einen Datensatz zu erstellen, der zwar quantitativ umfangreich ist, jedoch durch qualitative Tiefe der Einzelergebnisse charakterisiert wird. Diese Methode verspricht eine effiziente Erreichung des ersten Hauptziels und schafft eine solide Basis für das zweite Hauptziel.

4.3 Auswahl der Reviews

Dieser Arbeit liegt der Datensatz von Kupczyk [7] zu Grunde. Um aus diesem Datensatz passende Reviews für den Workshop zu identifizieren, wurde der Datensatz auf folgende Eigenschaften näher betrachtet. Die Teilnehmer des Workshops sollten möglichst die gleiche Domäne von Reviews präsentiert bekommen, um sich im Kontext der Reviews sicherer zu fühlen. Zudem soll der Datensatz die Taxonomie des Erklärbarkeitsbedarfs abdecken. Daraus entsteht der Bedarf möglichst eine App auszuwählen, die bereits viele Kategorien von Erklärbarkeitsbedarf abdeckt.

Die Analyse hat ergeben, dass der Datensatz mit den Reviews der App „Spotify“ die meisten erkannten Erklärbarkeitsbedarfe abdeckt. Der Datensatz konnte jedoch nicht alle Kategorien des Erklärbarkeitsbedarf für die App abdecken. Für die fehlenden Kategorien wurden neue Reviews durch Inspiration der Bestehenden konstruiert. Es wird angenommen, dass die Grundlagen der Domäne der App Spotify bei den Teilnehmern der Studie bekannt ist.

4.4 Teilnehmer

Die Teilnehmer des Workshops sollten praktischen Bezug zu Softwareanforderungen haben. Daher wird der Workshop mit Mitarbeitern aus einer Softwareabteilung durchgeführt. Teilgenommen haben insgesamt 4 Mitarbeiter Zur Sicherstellung der Erfahrung der Teilnehmer im Zusammenhang mit Anforderungen wurde am Ende des Workshops ein Fragebogen verteilt, der die individuelle Erfahrung im

Zusammenhang mit Erklärbarkeitsanforderungen abfragt. Die Auswertung der Umfrage hat ergeben, dass alle Teilnehmer mindestens regelmäßigen (wöchentlichen) Kontakt mit Anforderungen haben. 50% der Teilnehmer haben ständigen Kontakt (täglich) und 25% haben durchgehend (mehrmals täglich) mit Anforderungen zu tun. Daher kann eine grundlegende Kompetenz der Teilnehmer des Workshops und praxistaugliche Ergebnisse angenommen werden.

4.5 Erwartete Ergebnisse

Am Ende der Studie 1 soll ein Datensatz entstehen, der passende Erklärbarkeitsanforderungen und Erklärungen enthält, die die gesamte Taxonomie des in Kapitel 2.5 vorgestellten Erklärbarkeitsbedarfs beinhaltet.

Es wird erwartet, dass Muster bezüglich der Formulierungen in dem erstellten Datensatz erkannt werden können. Dadurch soll ein Konzept erstellt werden, um die Formulierung der Erklärbarkeitsanforderungen und Erklärungen zu automatisieren.

4.6 Durchführung

Der durchgeführte Workshop hatte folgende Gliederung:

- Einführung in das Thema Erklärbarkeit
- Aktivierung der Teilnehmer durch Beispiele
- Praxisphase:
 - Formulierung von Anforderungen
 - Formulierung von Erklärungen
(Zweite Hälfte der Formulierungen mit Angabe der Kategorie)
 - Formulierung von Anforderungen
 - Formulierung von Erklärungen
- Umfrage

Zu Beginn des Workshops wurde den Teilnehmern eine Einordnung des Workshops zur Masterarbeit ermöglicht. Den Teilnehmern wurden die Grundlagen von Erklärbarkeit im Kontext der Software und Definitionen von erklärbarer Software präsentiert. Anschließend wurden gemeinsam Beispiele aus dem Berufsalltag identifiziert, die als Erklärbarkeitsanforderungen aufgenommen werden könnten. Dieses Hinführen bietet einen hohen Faktor an Aktivierung der Teilnehmer. Sie

dient dazu, das Vorwissen zu aktivieren und an dieses mit neuem Wissen anzuknüpfen.

4.7 Formulierung von Anforderungen

Nach der Einführung begann der erste Teil der Arbeitsphase mit der Formulierung von passenden Erklärbarkeitsanforderungen zu ausgewählten Reviews. Dazu wurde ein Excel Dokument geteilt, welches nur die ausgewählten Reviews der App „Spotify“ beinhaltet. Auf Grundlage dieser Reviews wurde die Aufgabe gestellt eine passende Erklärbarkeitsanforderung zu formulieren. Das Ergebnis wurde in der Excelliste notiert.

Ab Hälfte der erfolgreichen Formulierung von Erklärbarkeitsanforderungen wurde zusätzlich zum Review die entsprechende Kategorie des Erklärbarkeitsbedarfs eingeblendet. So stand den Teilnehmern mehr Informationen zum Review bereit, um den Einfluss dieses Wissens auf die Teilnehmer zu analysieren und damit die Forschungsfrage 2 zu beantworten.

4.8 Analyse der Ergebnisse

Zur Beantwortung der Forschungsfrage 2 wird die Umfrage bezüglich der Kenntnis über die Kategorie des Erklärbarkeitsbedarfs zur Formulierung von Erklärbarkeitsanforderungen folglich interpretiert.

Nach der Formulierung der Erklärbarkeitsanforderungen konnte durch das zusätzliche Wissen der Kategorie keine zeitliche Abweichung, zwischen der Formulierung beider Hälften, gemessen werden. Die Auswertung der Umfrage diesbezüglich, zeigt eine tendenzielle subjektive Sicht bezüglich der Kenntnis der Kategorie. So empfanden einige Nutzer die Kenntnis über die Kategorie sinnvoll und hilfreich, andere stellten keinen Unterschied fest. Der Nutzen des Wissens scheint objektiv keine Auswirkungen zu haben.

4.8.1 Nutzen für Anforderungen

Anhand der Formulierungen der Erklärbarkeitsanforderungen lässt sich ein Muster identifizieren. Vermehrt tritt dabei der folgende Satzbau auf:

„Das System muss erklären ...“

Es scheint für die Teilnehmer des Workshops ein wichtiges Merkmal einer Erklärbarkeitsanforderung zu sein. Vereinzelt variiert der Satzbau wie folgt:

„Das System muss ... erklären“.

Diese Version ist jedoch seltener zu beobachten und die Wahl der Formulierung kann keiner bestimmten Kategorie von Erklärbarkeitsbedarf zugeordnet werden. Eine Abweichung von diesen Formulierungen ist nur in einem Fall zu sehen.

„Der Kundensupport muss die Nutzer besser abholen (Zeitpunkt der Fehlerbehebung, Grund, Workaround)“

Diese Anforderung bezieht sich auf ein Review, in dem kein Erklärbarkeitsbedarf für die App besteht, sondern Erklärungsbedarf bezüglich der Interaktion mit dem Kundenservice. Daher weichte die Wahl der gängigen Formulierung ab.

Die bewussten monotonen Formulierungen der Anforderungen deckt sich ebenfalls mit den Antworten aus der Umfrage.

„Erklärbarkeitsanforderungen sollten immer den gleichen Satzbau bzw. das gleiche Vokabular verwenden.“

Dieser Aussage stimmten 3 von 4 Teilnehmern des Workshops voll zu und einer stimmte der Aussage eher zu.

Um eine hohe Akzeptanz von automatisch erstellten Erklärbarkeitsanforderungen zu erhalten, sollten die Anforderungen daher den folgenden Satzbau nutzen.

„Das System muss erklären ...“

Die weiteren Antworten der Umfrage bezüglich des Konzeptes werden im Kontext des Kapitel 4 näher analysiert.

4.8.2 Nutzen für Erklärungen

Die Formulierungen der Erklärungen der Anforderungsanalysten ließ keine Rückschlüsse auf eine Schablone zu. Sie variieren im Detailgrad, sowie dem Kontext des jeweiligen Reviews. Teilweise wurde Domänenwissen genutzt, um Erklärungen in den Kontext einzuordnen. Andererseits wurden bei

Erklärungsbedarfen angenommen, dass eine Erklärung nicht in der Software hinterlegt werden sollte, sondern dem Nutzer eine direkte Antwort beschrieben wird. Dadurch war diese Erklärung stark an das Review angelehnt, sodass dem Nutzer individuell eine bestmögliche Lösung formuliert wurde.

Eine Entscheidung, ob eine Erklärung individuell betrachtet, oder für eine Software geeignet sein sollte, muss durch Hinzuziehen von weiterem Domänenwissen und Entscheidungsexpertise erfolgen.

4.9 Reflektion

4.9.1 Ziele

Der Workshop hat dazu beigetragen einen vollständigen Datensatz von Erklärbarkeitsanforderungen und Erklärungen zu erstellen. Dadurch wurde das erste Hauptziel erreicht. Durch die Analyse des Datensatzes ließ sich ein Muster erkennen, wodurch das zweite Hauptziel ebenfalls erreicht wurde.

4.9.2 Workshop Design und Durchführung

Nach der Einführung in das Thema Erklärbarkeit konnte Folgendes bei der anschließenden Formulierung der Anforderungen beobachtet werden. Die Teilnehmer benötigten für die ersten 5 Anforderungen deutlich mehr Zeit und empfanden es schwerer als für die folgenden Anforderungen. Es scheint, trotz der praktischen Einführung und Aktivierung der Teilnehmer durch Formulierung von Erklärbarkeitsanforderungen im eigenen Umfeld, Schwierigkeiten bei der Eingewöhnung der gegebenen Domäne „Spotify“ zu geben. Die Einführung hätte bereits Beispiele bezüglich möglicher Erklärungsbedarfe aus der Domäne enthalten sollen, um den Teilnehmern den Einstieg in die gegebenen Reviews zu erleichtern.

Für den restlichen Teil des Workshops zur Formulierung der Erklärbarkeitsanforderungen konnte die Motivation der Teilnehmer trotz der hohen Anzahl an Reviews aufrecht gehalten werden.

4.9.3 Teilnehmerfeedback

Die Teilnehmer des Workshops empfanden den Workshop durch die Einführung in das Thema Erklärbarkeit als wertvollen Aspekt der Anforderungsanalyse, den Sie in Ihrem Arbeitsalltag intensiver betrachten möchten. Das persönliche Feedback bezüglich der Durchführung der Ergebnisse

5 Konzept

Dieser Abschnitt bezieht sich auf den Entwurf eines Konzeptes zur automatischen Erstellung von Erklärbarkeitsanforderungen und Erklärungen. Es werden verschiedene Ansätze vorgestellt und diskutiert. Anschließend wird ein Ansatz ausgewählt, sowie die notwendigen Aspekte der Automatisierung betrachtet.

5.1 Grundsätzliche Ansätze zur Erstellung

Im Folgenden werden verschiedene Ansätze zur Erstellung von Erklärbarkeitsanforderungen und Erklärungen aufgeführt und im Anschluss bezüglich ihrer Eignung diskutiert.

5.1.1 Mapping Table

Ein technisch einfacher und zuverlässiger Ansatz für das Erlangen von Erklärbarkeitsanforderungen ist das Hinterlegen vordefinierter Erklärbarkeitsanforderungen in einer Tabelle. Diese können anhand von einer Kategorisierung des Reviews bzw. des zu erklärenden Sachverhaltes ausgewählt werden. In der folgenden Abbildung 5: Beispiel Prinzip einer Mapping Table [Eigene Darstellung] ist das Prinzip der Mapping Table dargestellt.

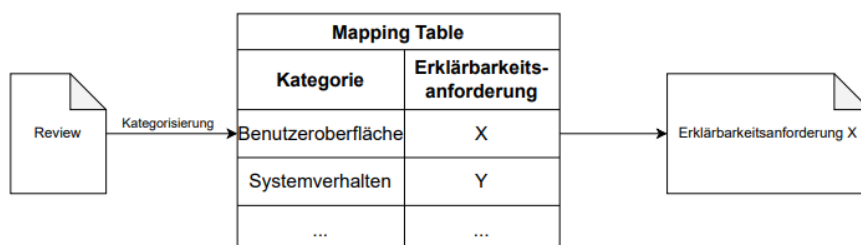


Abbildung 5: Beispiel Prinzip einer Mapping Table [Eigene Darstellung]

Das Review wird anhand des Inhaltes kategorisiert. Diese Kategorie wird mit einer vorher definierten Mapping Table abgeglichen und die jeweilige gemappte Erklärbarkeitsanforderung bereitgestellt. Das Erlangen von Erklärungen kann anschließend über das gleiche Prinzip erfolgen. Dabei werden Erklärbarkeitsanforderungen anstatt des Reviews genutzt und anstatt der Erklärbarkeitsanforderungen werden die Erklärungen ausgewählt.

Die Spalte Kategorie steht stellvertretend für eine relevante Metrik, die genutzt wird, um aus dem Review eine effektive Kategorisierung von Erklärbarkeitsanforderungen bzw. Erklärungen zu gewährleisten. Bezüglich der Kategorisierung einer Erklärbarkeitsanforderung ist eine intensivere Analyse für eine geeignete Metrik und eines Kategorisierungsalgorithmus erforderlich.

Diskussion

Ein Vorteil dieser Methode ist die Nachvollziehbarkeit zur Auswahl der Erklärbarkeitsanforderung bzw. der Erklärung. Die Mapping Table kann von den jeweiligen Nutzern individuell angepasst werden, um passende Anforderungen bzw. Erklärungen zu hinterlegen. So ist eine nachträgliche Anpassung ebenfalls möglich. Die Kategorien bzw. die ausgewählte Metrik müssen jedoch im Vorhinein festgelegt werden.

Die Abdeckung des Erklärbarkeitsbedarfs ist damit abhängig von der Güte der gewählten Metrik, des Klassifizierungsalgorithmus und der formulierten Erklärbarkeitsanforderungen. Neu zu erfassender Erklärungsbedarf ist über die reine Mapping Table nicht automatisch zu generieren, sondern nur bereits bestehenden Erklärbarkeitsanforderungen zuzuordnen. Das hat ebenfalls zur Folge, dass die Erklärbarkeitsanforderungen generalisiert anzusehen sind, also einen geringeren Detailgrad bezüglich des Reviews haben. Diese Eigenschaften für Erklärbarkeitsanforderungen wird von den Anforderungsanalysten aus der Praxis nicht gewünscht. Auf die Aussage

„Erklärbarkeitsanforderungen sollten möglichst grob gehalten werden, um viel Erklärungsbedarf abzudecken.“

antworteten alle Teilnehmer der Umfrage abgeneigt. So stimmten 50% der Aussage eher nicht zu und die anderen 50% gar nicht. Des Weiteren hatten alle Teilnehmer Bedenken bezüglich des Konzeptes der Mapping Table. Tendenziell lehnten sie die Methodik unter anderem wegen des Aufwandes zur Formulierung der Erklärbarkeitsanforderungen ab.

Sollte die Mapping Table alle nötigen Erklärbarkeitsanforderungen für den entsprechenden Detailgrad enthalten, ist der Aufwand gleichzusetzen, wie eine reine Formulierung der Anforderungen. Zudem müsste das Wissen der Erklärbarkeitsanforderungen bereits vorher bestehen. So könnte die Anzahl an Erklärbarkeitsanforderungen festgestellt werden, jedoch keine neuen Anforderungen generiert werden, die den Eigenschaften der Anforderungsanalysten genügen. Für die Generierung von Erklärungen

sieht es ähnlich aus, da die Details der Anforderung je nach Kontext notwendig ist und somit eine treffende Generierung nicht gewährleistet werden kann.

5.1.2 Schablone

Um die Vorteile der Mapping Table zu nutzen, jedoch auch individuellen Erklärungsbedarf zu gewährleisten, ist die Schablone eine mögliche Optimierung zur Mapping Table. Das Prinzip ist analog zur Mapping Table in der folgenden Abbildung 6 dargestellt.

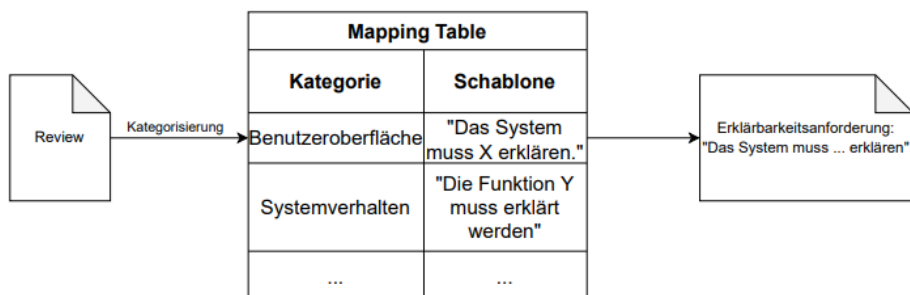


Abbildung 6: Beispiel Prinzip Mapping Table mit Schablone [Eigene Darstellung]

Statt konkreten Erklärbarkeitsanforderungen bzw. Erklärungen werden Schablonen bereitgestellt, die einen vordefinierten Satzbau zu den jeweiligen Kategorien beinhalten. Diese Vorlagen müssen vom Benutzer mit den jeweils benötigten Informationen, im Beispiel „X“ bzw. „Y“ gefüllt werden. Die Schablonen müssen dafür bereits vorher erstellt werden und können durch die Nutzer angepasst werden.

Diskussion

Eine individuelle Bearbeitung des Erklärungsbedarfs ist durch die Schablone gewährleistet und eine automatische Unterstützung zur Generierung scheint ebenfalls möglich, da der Kontext und die Details der Anforderungen abhängig von den Nutzern der Schablonen sind. Sie müssen das Review weiterhin auf notwendigen Erklärbarkeitsbedarf analysieren und die Informationen in die Schablone einpflegen. Die Generierung von Erklärungen ist über den gleichen Weg möglich.

Der Vorteil der Methode ist die Sicherstellung eines einheitlichen Satzbaus und Formulierungen. Aus der Umfrage aus Kapitel 4 geht hervor, dass diese Eigenschaft ebenfalls durch die Antworten der Anforderungsanalysten gewünscht wird.

„Erklärbarkeitsanforderungen sollten immer den gleichen Satzbau bzw. das gleiche Vokabular verwenden.“

Die Tendenz der Antworten geht mit 75% zur vollen Zustimmung dieser Aussage. 25% stimmen der Aussage eher zu. Das Prinzip einer Schablone scheint dadurch erwünscht zu sein.

Die Formulierung von Erklärungen anhand einer Schablone garantiert zwar, die Erfüllung des Erklärbarkeitsbedarfs, jedoch ist für den Nutzer ggf. eine abweichende individuelle Formulierung zielführender, wie in Kapitel 4.8.2 festgestellt wurde.

Die Herausforderung bei dieser Methode besteht in der Formulierung und Erkennung von passenden Schablonen und der Identifizierung und Evaluation einer geeigneten Metrik.

5.1.3 Dynamischer Ansatz

Einen anderen Ansatz besteht in der Nutzung von GPT-Modellen. Die Grundlagen dieser Modelle wurden in Kapitel 2.6.2 näher erläutert. GPT-Modelle werden genutzt, um komplexe Eingaben zu analysieren und basierend darauf ein gefordertes Ergebnis auszugeben. Die Ausgaben werden also dynamisch, je nach Eingabe erzeugt. Die Anweisungen zur Verarbeitung werden durch den sogenannten Prompt vorgegeben.

Das Prinzip kann mit der folgenden Abbildung 7 beschrieben werden.

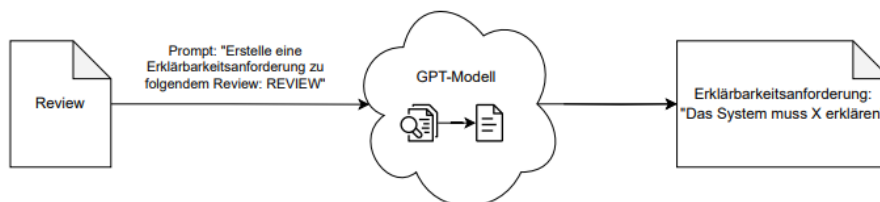


Abbildung 7: Prinzip Nutzung von GPT [Eigene Darstellung]

In diesem Beispiel wird einem GPT-Modell ein Prompt übergeben mit der Aufforderung eine Erklärbarkeitsanforderung zu erstellen. Der Inhalt des Reviews wird ebenfalls übergeben. Standardmäßig werden komplexere GPT-Modelle als Cloud Lösungen mit einer Programmierschnittstelle (API) des jeweiligen Anbieters genutzt, da diese einen hohen Speicherbedarf und Rechenleistung erfordern. Das GPT-Modell verarbeitet den Prompt,

indem es das Review analysiert und eine Erklärbarkeitsanforderung generiert.

Ebenso ist es möglich in einem nachfolgenden Schritt einen neuen Prompt zu übergeben, der eine generierte Erklärbarkeitsanforderung und die Aufforderung eine Erklärung zu dieser zu formulieren enthält.

Diskussion

Insbesondere bei der Formulierung von Erklärungen können bei komplexen Modellen, durch den Zugriff auf diverse Wissensquellen, praxistaugliche Ergebnisse generiert werden.

Dieses Prinzip hat den Vorteil, dass eine vollständige Automatisierung zur Generierung von Erklärbarkeitsanforderungen bzw. Erklärungen möglich ist und nur wenig Vorbereitung durch den Nutzer erforderlich ist.

Die Ergebnisse der Umfrage im Rahmen des Workshops aus Kapitel 3 ergeben eine hohe Akzeptanz für die Nutzung eines GPT-Modells. Wenige Bedenken beziehen sich auf die Nutzung des öffentlichen GPT-Modells ChatGPT[30], welches von OpenAI entwickelt wurde. Es wird empfohlen auf „sichere Varianten“ zuzugreifen. Ebenfalls wird angemerkt, dass ChatGPT fehlerhafte und schlechte Anforderungen generieren kann. Dieser Einwand könne jedoch durch das gezielte Trainieren auf Anforderungen eines eigenen GPT-Modells beseitigt werden.

Ein wesentlicher Nachteil bei diesen Modellen ist die Nachvollziehbarkeit der Verarbeitung, da die Modelle meist unüberwacht lernen und eine ausreichende Analyse und Formulierung zwar von diesen gefordert, aber nicht immer garantiert werden können. [31,32] Es ist möglich, dass der gegebene Erklärbarkeitsbedarf nicht erkannt, oder missinterpretiert wird und das Modell Halluzinationen als Antworten ausgibt. [33] Dieses Verhalten deckt sich mit den Aussagen über Bedenken eines GPT-Modells der Umfrageteilnehmern.

Sejnowski [34] illustriert genau dieses Verhalten von GPT-Modellen und zeigt durch mehrere Gespräche mit den gleichen Fragen, aber unterschiedlichen Prompts, wie die Ausgabe eines GPT-Modells positiv beeinflusst werden kann.

Unter Berücksichtigung der Besonderheiten und Einstellungen des jeweiligen GPT-Modells können daher automatisiert gute Erklärbarkeitsanforderungen und Erklärungen formuliert werden.

5.1.4 Hybrider Ansatz

Eine weitere Möglichkeit Erklärbarkeitsanforderungen zu generieren ist die Vermischung mehrerer Prinzipien. Der größte negative Aspekt des

dynamischen Ansatzes mittels GPT-Modellen ist die Ausgabe falscher Ergebnisse. Dem kann durch ausgewählte Prompts entgegengewirkt werden. Es ist denkbar, dass eine definierte Schablone für Erklärbarkeitsanforderungen, wie in Kapitel 5.1.2 beschrieben, einem GPT-Modell als Vorgabe für die Ausgabe übergeben wird. Die folgende Abbildung 8 illustriert das Prinzip.

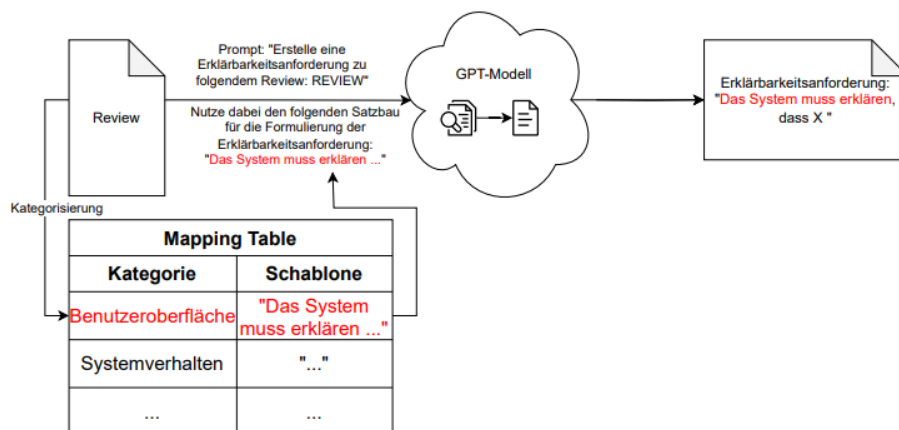


Abbildung 8: Prinzip GPT mit Mapping Table [Eigene Darstellung]

Das Prinzip nutzt die Mapping Table, um durch die Kategorisierung des Reviews eine passende Schablone auszuwählen und dem GPT-Modell die Anweisung übergibt, diese Schablone als Ausgabe zu nutzen. Im Beispiel aus Abbildung 8 wird der Erklärungsbedarf des Reviews zur Kategorie Benutzeroberfläche zugeordnet. Die Mapping Table stellt die Schablone „Das System muss erklären ...“ bereit und das GPT-Modell nutzt das Review und die Schablone, um eine passende Anforderung auszugeben, die der Schablone entspricht.

Dieses Prinzip scheint die positiven Aspekte aller vorgestellten Konzepte zu vereinen und die negativen zu minimieren. So ist eine vollständige Automatisierung nach einmaliger Einrichtung möglich. Ebenfalls kann die Erstellung der Anforderungen und Erklärungen durch Ändern des Promptes beeinflusst werden. Eine wichtige Voraussetzung hierfür ist jedoch weiterhin das Vorbereiten einer Mapping Table und Identifizieren von möglichen Schablonen und zugehörigen Metriken.

5.2 Zusammenfassung der Ansätze

In diesem Kapitel wurden vier unterschiedliche Ansätze vorgestellt und diskutiert, die folgend zusammengefasst und letztlich ein Ansatz ausgewählt wird.

Mapping Table

Diese Methode bietet eine strukturierte Möglichkeit, Erklärbarkeitsanforderungen auf Basis vordefinierter Kategorien zu generieren. Sie ermöglicht eine Nachvollziehbarkeit und individuelle Anpassung der Anforderungen. Allerdings erfordert sie eine intensive Analyse zur Festlegung geeigneter Kategorien und eine vorherige Formulierung der Anforderungen, was einen hohen zeitlichen und personellen Aufwand bedeutet.

Schablone

Die Verwendung von Schablonen gewährleistet eine einheitliche Formulierung von Erklärbarkeitsanforderungen, wodurch die Nachvollziehbarkeit und die Konsistenz verbessert werden können. Allerdings ist die Auswahl und Anpassung der Schablonen entscheidend und individuelle Formulierungen könnten bezüglich Erklärungen zielführender sein.

Dynamischer Ansatz

Die Nutzung von GPT-Modellen ermöglicht eine automatische Generierung von Erklärbarkeitsanforderungen und Erklärungen. Dieser Ansatz bietet eine hohe Flexibilität und kann komplexe Zusammenhänge erfassen. Allerdings ist die Nachvollziehbarkeit der Ergebnisse und die Möglichkeit von Fehlern oder Missinterpretationen durch das Modell eine Herausforderung.

Hybrider Ansatz

Durch die Kombination von Mapping Table, Schablonen und GPT-Modellen können die Vorteile der verschiedenen Ansätze genutzt und ihre Nachteile minimiert werden. Dieser Ansatz ermöglicht eine weitgehende Automatisierung und Flexibilität bei der Generierung von Erklärbarkeitsanforderungen.

Aus diesen Überlegungen geht hervor, dass die Nutzung eines hybriden Ansatzes mit Beachtung der Besonderheiten des auszuwählenden GPT-

Modells die effektivste Methode ist, um die Erstellung von Erklärbarkeitsanforderungen und Erklärungen zu automatisieren.

5.3 Beurteilung der Aspekte zur Automatisierung

Um die Forschungsfragen [FF4] und [FF5] näher zu beantworten werden nachfolgend die relevanten Aspekte bezüglich der Automatisierung beurteilt, um einen Prozess zur Generierung von Erklärbarkeitsanforderungen und Erklärungen erfolgreich umzusetzen.

Aus dem Datensatz wurden probeweise Reviews mit diversen Prompts einer Chat-GPT-3.5 Instanz übergeben zur Generierung von Erklärbarkeitsanforderungen. Dabei ist aufgefallen, dass GPT als Ausgabe zwar einen vorgegebenen Satzbau verwendet, jedoch zusätzlich noch überflüssige Formulierungen ergänzt, da die Reviews häufig laienhafte Formulierungen und unnötige Informationen enthalten. Dieser Effekt ist durch die in Kapitel 5.1.4 genannte Halluzination zu beschreiben. Um diesem entgegenzuwirken, wurde im ersten Schritt eine Zusammenfassung des Reviews generiert. Bei der Aufforderung zur Erstellung einer Erklärbarkeitsanforderung aus der Zusammenfassung konnte beobachtet werden, dass nur relevante Aspekte in die Erklärbarkeitsanforderung aufgenommen wurden.

Daraus ergeben sich die folgenden Schritte zur erfolgreichen Automatisierung:

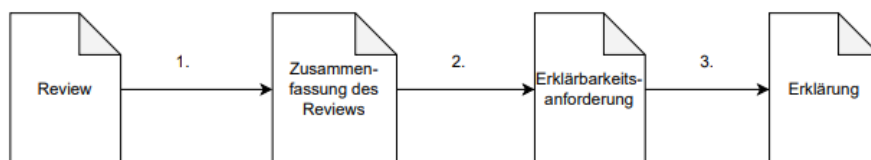


Abbildung 9: Workflow zur Automatischen Erstellung [Eigene Darstellung]

1. Zusammenfassung von Reviews
2. Erstellung von Erklärbarkeitsanforderungen
3. Erstellung von Erklärungen

Diese Schritte werden nachfolgend genauer betrachtet.

5.3.1 Zusammenfassung von Reviews

Das Zusammenfassen von Reviews ist ein notwendiger Schritt, um das Halluzinieren eines GPT-Modells zu vermeiden und den Inhalt der

Erklärbarkeitsanforderungen auf das Essenzielle zu beschränken. GPT-Modelle sind darauf ausgelegt Text zu verstehen und den Inhalt wiederzugeben. Die Erstellung von Zusammenfassung mittels eines GPT-Modells stellt daher keine Herausforderung dar. Wichtig dabei ist die Erprobung von geeigneten Prompts, sodass die relevanten Aspekte beibehalten werden.

5.3.2 Erstellung von Erklärbarkeitsanforderungen

Die Erstellung von Erklärbarkeitsanforderungen basiert auf der zuvor erstellten Zusammenfassung. Mittels der Vorgabe einer Schablone ist der Inhalt der Erklärbarkeitsanforderung hauptsächlich abhängig von der Zusammenfassung, die bereits alle relevanten Aspekte beinhalten sollte. Die Auswahl von passenden Schablonen wurde durch die Analyse des erstellten Datensatzes in Kapitel 4.8 auf eine unabhängige Schablone beschränkt. Dadurch ist die Automatisierung der Erstellung einer Erklärbarkeitsanforderung durch ein GPT-Modell ein effektives Prinzip zur Automatisierung.

5.3.3 Erstellung von Erklärungen

Während der ersten Studie der Arbeit ist bei der Formulierung von Erklärungen auf Grundlage der Erklärbarkeitsanforderungen aufgefallen, dass für die Formulierungen weiteres Domänenwissen erforderlich ist. Es wurden teilweise Annahmen aufgestellt, um mögliche Erklärungen zu formulieren und das fehlende Domänenwissen auszugleichen. Die Auswertung der Umfrage zur Formulierung der Erklärungen ergab, dass die Anforderungsanalysten unsicherer als bei Formulierung der Anforderungen waren. Daraus lässt sich schließen, dass ein Bezug zu Domänenwissen der App für die Formulierung von Erklärungen notwendig ist.

Ein GPT-Modell kann bezüglich dieses Domänenwissens trainiert werden, sodass Erklärungen sicher generiert werden können. Öffentliche GPT-Modelle bzw. solche, die Zugriff auf aktuelle Daten haben, können teilweise dieses Domänenwissen abdecken.

Die Erstellung von Erklärungen erfordert das Verständnis einer gegebenen Problematik. Eine Erklärbarkeitsanforderung beinhaltet alle geforderten Informationen zum Verständnis der Problematik. Ein GPT-Modell ist in der Lage den Inhalt einer Erklärbarkeitsanforderung, also die Problematik zu erfassen und durch Zugriff auf das vorhandene Domänenwissen eine Lösung formulieren. Eine Gewährleistung, dass diese Erklärung auf Niveau

der Entwickler ist, kann nicht gegeben werden. Die Generierung von Erklärungen mittels GPT-Modellen ist daher als Unterstützung, jedoch nicht als finale Erklärung zu sehen. Eine Überprüfung und Anpassung durch Experten der Domäne ist sinnvoll.

5.4 Fazit

In diesem Kapitel wurden mögliche Konzepte zur Automatisierung vorgestellt und näher betrachtet. Es ergibt sich, dass die Folgenden Schritte zur Automatisierung notwendig und möglich sind.

- Zusammenfassung des Reviews
- Erstellung einer Erklärbarkeitsanforderung-
- Erstellung einer Erklärung

Die Methodik zur Umsetzung der einzelnen Schritte legt eine Verwendung eines GPT-Modells nahe. Grundsätzlich sind die Ergebnisse der einzelnen Schritte voneinander abhängig, jedoch die Methodik eines Schrittes unabhängig. Die Methodik der einzelnen Schritte kann somit beliebig angepasst oder ausgetauscht werden, ohne den Prozess im Ganzen anpassen zu müssen.

Ein manuelles Überprüfen oder Eingreifen der generierten Ergebnisse ist insbesondere nach Erstellung der Erklärbarkeitsanforderung und der Erklärung sinnvoll.

6 Umsetzung der Automatisierung

Dieses Kapitel beschäftigt sich mit der Implementierung einer Software, um die Erstellung von Erklärbarkeitsanforderungen und Erklärungen zu automatisieren.

Die Anforderungsanalyse bildet einen wesentlichen Schritt in der Entwicklung jeder Softwarelösung, indem sie die Grundlage für das Design, die Entwicklung und die Implementierung legt. Diese beginnt mit der Betrachtung der Stakeholder. Anschließend werden die Anforderungen für die geplante Software erfasst und analysiert. Darauf folgt die Priorisierung der Anforderungen, sowie deren Interpretation und Umsetzung.

6.1 Stakeholder

6.1.1 Anforderungsanalysten

Als primäre Benutzer des Tools sind Anforderungsanalysten der betroffenen Software die Stakeholder, die direkt von der Entwicklung und Implementierung des Tools profitieren. Ihre Anforderungen und Arbeitsabläufe bilden die Grundlage für das Design und die Funktionalität der Software. Hauptziele für Anforderungsanalysten sind die Effizienzsteigerung bei der Erstellung von Erklärbarkeitsanforderungen und die Verbesserung der Qualität und Vollständigkeit dieser Anforderungen.

6.1.2 Softwareentwickler

Die Softwareentwickler, welche die Software, auf die die Reviews bezogen sind, weiterentwickeln müssen sind ebenfalls Stakeholder des zu entwickelnden Tools. Diese müssen den erkannten Erklärbarkeitsbedarf verstehen. Somit nutzen sie die generierten Erklärbarkeitsanforderungen und Erklärungen, um die betroffene Software zu verbessern und den Erklärungsbedarf für die Nutzer der Software zu vermeiden.

6.1.3 Nutzer der Software

Als Nutzer der Software sind die Kunden zu verstehen, die einen Erklärbarkeitsbedarf identifiziert haben, oder die Software nutzen. Die Kunden erstellen ein Review, welches Erklärungsbedarf beinhaltet. Dieses nutzt das Tool als Input, um Erklärbarkeitsanforderungen und Erklärungen

zu generieren. Somit haben die Formulierungen der Reviews direkten Einfluss auf die Ergebnisse des Tools. Ebenfalls werden die Ergebnisse des Tools genutzt, um die Software zu verbessern. Daher müssen die Erklärungen ebenfalls verständlich für die Nutzer der bestehenden Software sein.

6.2 Anforderungen erfassen

Die Anforderungen der Software wurden auf Grundlage des folgenden Mock Up im Gespräch mit den primären Nutzern der Software, genauer den Anforderungsanalysten aus dem absolvierten Workshop, durchgeführt.

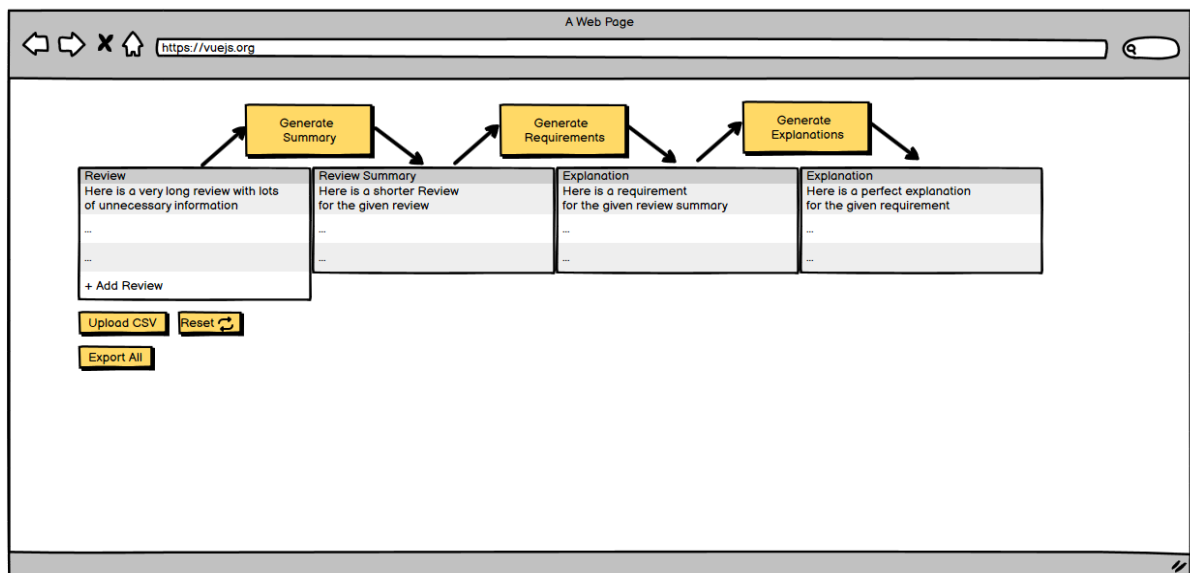


Abbildung 10: Mock Up Tool [Eigene Darstellung]

Nach Besprechung der erwarteten Funktionen und Eigenschaften, ergaben sich die folgenden Anforderungen.

6.2.1 Funktionale Anforderungen

[FR1] Einfügen von Reviews

Die Software muss die Möglichkeit bieten Reviews in verschiedene Formate einzufügen, einschließlich direkter Eingabe, Kopieren und Einfügen sowie Datei-Upload.

[FR2] Erstellen von Zusammenfassungen für Reviews

Die Software muss Zusammenfassungen für einzelne Reviews und für eine Gruppe von Reviews erstellen können.

[FR3] Erstellen von Erklärbarkeitsanforderungen für Zusammenfassungen

Die Software muss Erklärbarkeitsanforderungen für einzelne Zusammenfassungen und für eine Gruppe von Zusammenfassungen erstellen können.

[FR4] Erstellen von Erklärungen für Erklärbarkeitsanforderungen

Die Software muss Erklärungen für einzelne Erklärbarkeitsanforderungen und für eine Gruppe von Erklärbarkeitsanforderungen erstellen können.

[FR5] Hochladen einer CSV mit Reviews

Die Software muss die Möglichkeit bieten, eine CSV-Datei mit der Kodierung UTF-8 hochzuladen, die mehrere Reviews enthält, um eine schnelle Eingabe großer Mengen von Reviews zu ermöglichen.

Die hochgeladenen Reviews müssen verarbeitet werden können, um Zusammenfassungen, Erklärbarkeitsanforderungen und Erklärungen auf den Daten basierend erstellen zu lassen.

[FR6] Export von Ergebnissen

Die Software muss die Möglichkeit bieten, erstellte Zusammenfassungen, Erklärbarkeitsanforderungen und Erklärungen als CSV-Datei mit der Kodierung UTF-8 zu exportieren.

[FR7] Verändern der Einstellungen von GPT-Model

Die Software muss Einstellungen bieten, um die Parameter für die GPT-Generation anzupassen, einschließlich des Promptes, der Länge des generierten Textes und der Temperatur des Modells.

Benutzer sollten in der Lage sein, die Einstellungen entsprechend ihren spezifischen Anforderungen und Vorlieben anzupassen, um die Qualität und Relevanz der generierten Texte zu beeinflussen.

[FR8] Verändern der Ergebnisse

Die Software muss den Nutzern die Möglichkeit bieten, die generierten Ergebnisse manuell zu bearbeiten. Die Software soll dann auf Grundlage der Änderungen neue Ergebnisse generieren können.

6.2.2 Nicht funktionale Anforderungen

[NFR1] Modularer Aufbau

Es sollte möglich sein, neue Technologien oder Algorithmen in vorhandene Komponenten der Software zu integrieren, um die Leistung der Textgenerierung der Software zu verbessern, ohne die Gesamtarchitektur zu beeinträchtigen.

[NFR2] Einfluss auf Generierung

Die Software muss erklären, wie der Nutzer Einfluss auf die Generierung von Texten nehmen kann.

[NFR3] Überprüfen der Ergebnisse

Die Software muss erklären, dass die generierten Texte manuell von den Nutzern überprüft werden sollte.

[NFR4] Funktionen erklären

Die Software muss erklären, welche Auswirkungen das Anklicken eines Buttons hat.

6.2.3 Optionale Anforderungen

[OR1] Automatische Erkennung einer Änderung

Die Software erkennt automatisch eine manuelle Anpassung und gib einen Hinweis die Ergebnisse, die auf der Änderung basieren neu zu generieren.

[OR2] Markieren von Inhalten

Die Software muss dem Nutzer eine Möglichkeit bieten, die generierten Inhalte eindeutig zu markieren, um dem Nutzer eine visuelle Übersicht über ausgewählte Inhalte zu geben. Der Nutzer soll dabei verschiedene Möglichkeiten haben einen Inhalt eindeutig zu markieren, um Unterschiede zwischen

- Notwendigkeit eines Erklärbarkeitsbedarfs
- Akzeptanz eines Inhalts
- Nachbesserung notwendig

visuell darzustellen.

[OR3] Priorisieren von Inhalten

Die Software muss dem Nutzer eine Möglichkeit bieten, die Erklärungsbedarfe zu priorisieren, um höher relevante Inhalte schneller zu bearbeiten. Die Priorisierung muss bei Export der Ergebnisse angegeben sein.

6.3 Priorisierung der Anforderungen

Zur Priorisierung der Anforderungen wird in dieser Arbeit das Kano Modell der Kundenzufriedenheit [35] genutzt, um die Anforderungen in klare Kategorien einzuordnen.

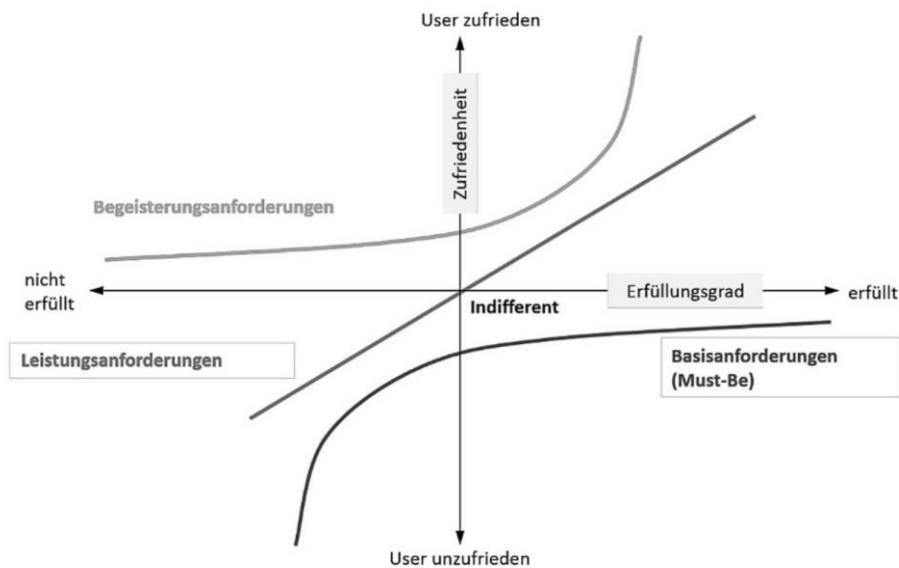


Abbildung 11: Kano Modell der Kundenzufriedenheit [36]

Das Kano Modell aus Abbildung 10 nutzt die folgenden Kategorien

- Basisanforderungen
- Leistungsanforderungen
- Begeisterungsanforderungen

Diese unterscheiden sich bezüglich der Notwendigkeit der Umsetzung. Basisanforderungen sind die Grundlagen einer Software, die der Nutzer erwartet. Sie werden als Minimum für die Kundenzufriedenheit angesehen. Leistungsanforderungen beeinflussen die Kundenzufriedenheit abhängig von ihrer Umsetzung. So können mangelhafte Umsetzungen sogar zu einer Kundenunzufriedenheit führen, wohingegen gute Umsetzungen die Kundenzufriedenheit erhöhen. Die Begeisterungsfaktoren erhöhen die Kundenzufriedenheit stark bei vollständiger Umsetzung. Bei Vernachlässigung dieser sinkt die Kundenzufriedenheit jedoch nicht. Anzustreben ist daher erst die Erfüllung

der Basisanforderungen, dann der Leistungsfaktoren und letztlich der Begeisterungsfaktoren.

Die Einordnung in die Kategorien mit Priorisierung der Anforderungen ist der folgenden Tabelle zu entnehmen.

Kategorie	Anforderung	Priorisierung
Basisanforderungen	NFR1	Hoch
	FR1	Hoch
	FR2	Hoch
	FR3	Hoch
	FR5	Hoch
	FR8	Hoch
	FR4	Mittel
Leistungsanforderungen	FR6	Hoch
	FR7	Hoch
	NFR4	Hoch
	NFR2	Mittel
	NFR3	Mittel
Begeisterungsanforderungen	OR1	Niedrig
	OR2	Niedrig
	OR3	Niedrig

Tabelle 1: Einordnung und Priorisierung der Anforderungen

Die Anforderung [FR4] wurde nicht hoch priorisiert, da die Erstellung von Erklärungen, wie in Kapitel 4.8.2 festgestellt, häufig Hintergrundinformationen benötigt und meist manuell nachgepflegt werden muss.

6.4 Entwicklung

Für die Umsetzung des Tools wird sowohl eine grafische Oberfläche für die Interaktionen mit dem User, das Frontend, sowie ein Backend, um die eingegebenen Inhalte als Anfrage an die entsprechenden GPT-API zu schicken. Durch die Trennung in die Komponenten Frontend und Backend soll ebenfalls die Anforderung [NFR1] erfüllt werden. Für das Frontend wird das Web Framework Vue.js genutzt, um eine plattformunabhängige Software zu entwickeln. Das Backend nutzt die Programmiersprache Python, sodass viele GPT-API als Bibliotheken einfach integriert werden können. Die Kommunikation zwischen Frontend und Backend ist über eine RESTful API möglich. Dabei nutzt das Backend einen Flask Server. Flask ist

ein Mikro-Framework für die Bereitstellung einer RESTful API. [37] So ergibt sich die folgende Architektur für die Kommunikation zwischen den API.

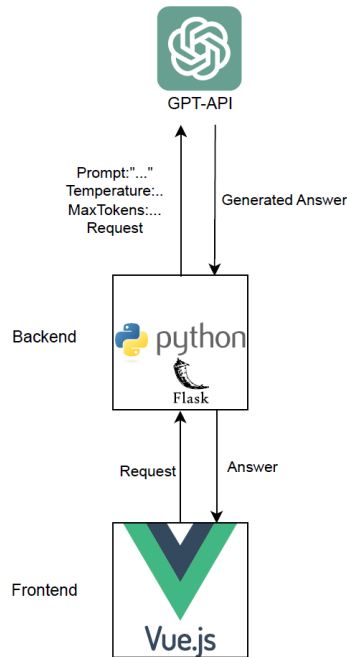


Abbildung 12: API Architektur Tool [Eigene Darstellung]

Das Design der Software soll sich am Mock Up orientieren. So sind im Hauptbereich der Anwendung 4 Spalten einer Tabelle zu sehen, um Reviews, Zusammenfassungen, Erklärbarkeitsanforderungen und Erklärungen anzuzeigen. Die Anforderung [FR8] wird abgedeckt, indem jede Zelle der Tabelle bearbeitet werden kann.

Die Hauptfunktionen zur Generierung sind sowohl in jeder Zelle einzeln durch einen Icon-Button aufrufbar und ebenfalls für jeweils die gesamte Spalte unterhalb der Tabelle. Zum Verständnis des Datenflusses wurden neben die Buttons Pfeile eingefügt, die auf die Tabellenspalten verweisen. Die Generation erfolgt mit unterschiedlichen Prompts und Einstellungen für den jeweiligen Inhalt. Damit sind die Hauptfunktionen mit den Anforderungen [FR2], [FR3] und [FR4] ebenfalls abgedeckt.

Zur Interaktion der Tabelle sind unterhalb über die Länge der Tabelle 4 Buttons angeordnet. Durch die Buttons „Add Row“ und „Remove Row“ ist es möglich Zeilen manuell hinzuzufügen oder zu entfernen, um manuell neue Reviews einzugeben. Dadurch wird die Anforderung [FR1] abgedeckt. Die beiden anderen Buttons decken die Funktionen für das Einfügen

mehrerer Reviews durch das Hochladen einer CSV-Datei für Anforderung [FR5], sowie den Export der Tabelleninhalte als CSV-Datei [FR6].

Neben den Buttons für die spaltenweise Erstellung von Zusammenfassungen, Erklärbarkeitsanforderungen und Erklärungen ist jeweils ein Icon-Button, um die Einstellungen für die Generierungen individuell anzupassen. Dazu öffnet sich jeweils ein Dialog, in dem die Einstellungen angepasst werden können. Hierdurch wird auch die Anforderung [FR7] abgedeckt.

Die Leistungsanforderung [NFR4] wird erfüllt, indem jeder Button einen Tooltip erhält. Wenn der Mauszeiger über dem Button schwebt, wird eine kurze Beschreibung über die Funktion des Buttons eingeblendet.

Die Leistungsanforderung [NFR3] wird erfüllt, indem vor Export der Ergebnisse ein Dialog erscheint, der den Nutzer auf eine Überprüfung der Inhalte hinweist. Um den Nutzern eine grundlegende Erklärung für die Software zu bieten, erscheint durch die Auswahl des „Help“ Buttons ein Overlay, der alle Interaktionsmöglichkeiten beschreibt. Insbesondere auch die Möglichkeit Einfluss auf die generierten Ergebnisse zu nehmen, sodass die Leistungsanforderung [NFR2] ebenfalls erfüllt ist.

7 Studie: Evaluation der Automatisierung

Im Rahmen der Evaluierung der Automatisierung werden in diesem Kapitel das entwickelte Tool und dessen Ergebnisse untersucht. Dafür soll ein Vergleich zwischen manuell erstellten Ergebnissen und automatisiert erstellten Ergebnissen, durchgeführt werden.

7.1 Ziel

Das Hauptziel dieser Studie liegt in der Bewertung der Qualität der automatisiert generierten Ergebnisse sowie in der Analyse der Akzeptanz der Software und ihrer Ergebnisse durch Anforderungsanalysten. Durch die systematische Auswertung dieser Aspekte soll ein vertieftes Verständnis für die Effektivität und Effizienz der entwickelten Automatisierungslösung gewonnen werden. Darüber hinaus zielt die Studie darauf ab, potenzielle Verbesserungsbereiche aufzudecken und Empfehlungen für die Weiterentwicklung der Software abzuleiten.

7.2 Methodik

Zur Evaluation der Qualität und Akzeptanz der generierten Ergebnisse wird das entwickelte Tool eingesetzt, um automatisch Anforderungen basierend auf den Reviews der App "Spotify" zu erstellen. Diese Reviews wurden bereits in Kapitel 3 für die Erstellung von Erklärbarkeitsanforderungen durch Anforderungsanalysten genutzt. Der Ansatz ermöglicht einen direkten Vergleich zwischen den von menschlichen Anforderungsanalysten formulierten Anforderungen und den automatisch generierten durch das Tool. Somit entsteht ein Datensatz, der jeweils zwei Erklärbarkeitsanforderungen pro Review enthält, wodurch eine umfassende Bewertung der Leistungsfähigkeit und Effektivität des Tools im Vergleich zu menschlicher Expertise ermöglicht wird.

Innerhalb dieses Datensatzes wird den Anforderungsanalysten die Aufgabe zugewiesen, für jedes Review eine Anforderung auszuwählen, die sie den Entwicklern der Software als Erklärbarkeitsanforderung übergeben würden. Dabei sollen die Anforderungsanalysten ihre Expertise und Erfahrung im Bereich der Anforderungsanalyse nutzen, um diejenigen Anforderungen zu identifizieren, die eine klarere und umfassendere

Erfassung des Erklärbarkeitsbedarfs ermöglichen. Die manuelle Auswahl dient ebenfalls dazu, die Validität und Relevanz der generierten Erklärbarkeitsanforderungen zu gewährleisten.

Diese Vorgehensweise trägt dazu bei, potenzielle Unterschiede und Gemeinsamkeiten zwischen den automatisch generierten und den manuell ausgewählten Anforderungen zu identifizieren und zu analysieren, wodurch Einblicke in die Leistungsfähigkeit und Präzision des Tools gewonnen werden können.

Die Auswahl der Anforderungen erfolgt innerhalb der vorbereiteten Excel Liste. Ein Auszug aus dieser ist der folgenden Abbildung 13 zu entnehmen.

Review	Review Summary	Requirement Option 1.	Requirement Option 2.	Welche Erklärbarkeitsanforderungen würden Sie den Entwicklern der Software weitergeben?
Not the best:Well first of all, when I'm in a playlist or album it won't let me see what songs are in there. In my iPad on the other hand; it lets me play any song and lets me skip as many times as I want (and I don't have premium). On my phone it's the complete opposite. That's really annoying. I don't get why being able to play any song in a playlist or album is a premium feature. If I'm trying to choose a certain song from a playlist or album; I can't because it won't let me see the songs. If it does (which is rare), it only lets me see a preview of the song or piece. I feel like some of the non-premium features are	If I'm trying to choose a certain song from a playlist or album; I can't because it won't let me see the songs.	Das System muss erklären, warum bestimmte Songs in einer Playlist oder einem Album nicht ausgewählt werden können.	Das System muss über das Abo Modell aufklären und welche Funktionen dadurch eingeschränkt werden.	<input checked="" type="radio"/> 1 <input type="radio"/> 2

Abbildung 13: Studie - Beispiel der Entscheidungs-Tabelle [Eigene Darstellung]

Die Tabelle enthält sowohl die Reviews, die Zusammenfassungen und die entsprechenden Erklärbarkeitsanforderungen. Dabei ist nicht gekennzeichnet, welche Anforderung durch das Tool und welche manuell erstellt wurden. Durch die Auswahl einer Checkbox können die Teilnehmer ihre Auswahl treffen.

Für die Bewertung durch die Teilnehmer wird der quantitative Ansatz verwendet, um gezielter Ergebnisse zu analysieren. Dadurch können eindeutig ausgewählte Ergebnisse, die von allen Teilnehmern gleich entschieden, von den unstimmgigen Ergebnissen abgegrenzt werden. Dieser Ansatz ermöglicht es, die Ergebnisse genauer zu analysieren und Rückschlüsse auf die Eigenschaften der generierten Ergebnisse im Vergleich zu den manuell formulierten zu ziehen.

Im nächsten Schritt wird die praktische Anwendung des Tools einer Evaluation unterzogen. Hierbei sind die Anforderungsanalysten dazu aufgefordert, für ausgewählte Reviews Anforderungen zu erstellen. Dabei wird für die erste Hälfte der Anforderungen das Tool eingesetzt, während die zweite Hälfte manuell erstellt wird. Zur Bewertung der Effizienz des Tools im Vergleich zur manuellen Methode wird die Zeit gemessen, die jeweils für die Erstellung mit dem Tool bzw. manuell benötigt wird. Um vergleichbare Ergebnisse zu erzielen, werden die Anforderungsanalysten in zwei Gruppen aufgeteilt. Beide Gruppen haben die Aufgabe,

Erklärbarkeitsanforderungen für dieselben Reviews zu erstellen. Allerdings verwendet die erste Gruppe für die ersten Reviews das Tool, während die zweite Gruppe die Anforderungen manuell formuliert. Nach der Hälfte der Reviews tauschen die Gruppen ihre Methoden, um potenzielle Unterschiede in der Effizienz und Qualität zwischen der automatisierten und manuellen Erstellung zu untersuchen.

7.3 Auswahl der Reviews

Im ersten Teil dieser Studie werden die Rezensionen des vorhergehenden Workshops als Ausgangspunkt für die Untersuchung genutzt. Der zweite Teil der Studie konzentriert sich auf die praktische Anwendung des Tools und die Generierung neuer Anforderungen auf Basis neuer Reviews. Für diesen Teil werden sowohl neue Reviews aus der App "Spotify" als auch aus anderen Anwendungsdomänen herangezogen. Durch die Einbeziehung von Reviews aus verschiedenen Domänen wird eine breitere Validierung der Tool-Leistung angestrebt. Diese Vorgehensweise ermöglicht es, den Einfluss der Domäne auf die Wahl der Formulierungen zu überprüfen und potenzielle Unterschiede der automatisierten Anforderungsgenerierung zwischen verschiedenen Anwendungsbereichen zu identifizieren.

7.4 Teilnehmer

Ähnlich zur ersten Studie sollen die Teilnehmer Anforderungsanalysten mit praktischem Bezug sein. Zur Sicherstellung der Kompetenz wurde erneut der Fragebogen zur fachlichen Kompetenz verteilt. Insgesamt haben 8 Anforderungsanalysten an dieser Studie teilgenommen. Zur Sicherstellung der neutralen Bewertung der Ergebnisse wurden neue Teilnehmer ausgewählt.

7.5 Erwartete Ergebnisse

Im Rahmen der vorliegenden Studie wird als primäres Ergebnis ein Datensatz erstellt, der die Entscheidungen der Teilnehmer über die als geeigneter erachteten Anforderungen umfasst. Dieser Datensatz bildet eine Grundlage für die Analyse zur Bewertung der Leistungsfähigkeit des Tools. Es wird erwartet, dass die Ergebnisse des Tools in wenigen Fällen als geeigneter erachtet werden, obwohl das Tool primär für die Interaktion mit Anforderungsanalysten konzipiert ist und die Anforderungen im Idealfall durch menschliche Expertise überprüft und verfeinert werden

sollten. Dennoch wird angenommen, dass die Einsicht in die Entscheidungen der Teilnehmer wichtige Erkenntnisse über die relevanten Merkmale der Formulierung im Kontext der Reviews bieten, insbesondere bei eindeutigen Bewertungen.

Des Weiteren wird im zweiten Teil der Studie die Effizienz des Tools im Vergleich zur manuellen Formulierung überprüft. Dabei wird erwartet, dass die Nutzung des Tools einen signifikanten Zeitvorteil sowie eine gesteigerte Zufriedenheit mit den erstellten Anforderungen mit sich bringt. Diese Erwartung bezieht sich auf die Annahme, dass die automatisierte Generierung von Anforderungen durch das Tool einen effizienteren Prozess ermöglicht und dabei gleichzeitig bei manueller Überprüfung die Qualität der Ergebnisse aufrechterhalten oder sogar verbessern kann. Durch die systematische Untersuchung dieser Aspekte wird angestrebt, ein umfassendes Verständnis für die Leistungsfähigkeit und Effektivität des Tools in der praktischen Anwendung zu erlangen.

7.6 Durchführung

Der Workshop gliederte sich nach folgenden Punkten:

- Einführung in das Thema Erklärbarkeit
- Aktivierung der Teilnehmer durch Beispiele
- Beispiel für den Erklärbarkeitsbedarf der Domäne Spotify vorstellen
- Praxisphase: Auswahl der geeigneteren Anforderungen
- Live-Vorstellung des Tools und der Aufgabenstellung
- Gruppenweise Formulierung der Erklärbarkeitsanforderungen mit Zeitmessung
- Umfrage

Der Workshop beginnt mit einer einführenden Präsentation, die die Teilnehmer mit dem Konzept und der Bedeutung von Erklärbarkeit vertraut macht. Hierbei werden grundlegende Definitionen erläutert, um eine gemeinsame Basis zu schaffen.

Die Teilnehmer sollen ein besseres Gefühl für Erklärbarkeitsanforderungen erlangen, indem gemeinsam Erklärbarkeitsanforderung aus ihrem Alltag erfasst und formuliert werden. Abschließend zur Einführung werden Beispiele für Erklärbarkeitsbedarf im Kontext von Reviews besprochen.

Im Anschluss an die Einführung werden die Teilnehmer aufgefordert die zuvor verteilten Excel Listen zu öffnen und die Aufgabe gestellt,

auszuwählen, welche Erklärbarkeitsanforderungen besser geeignet sind, um sie an die Entwickler der Software weiterzugeben.

Nachdem alle Excel Listen erfolgreich abgeschlossen sind, wird das entwickelte Tool [Kapitel 6.4] vorgeführt. Sobald die Fragen zur Nutzung geklärt sind, wurden die Teilnehmer in zwei Gruppen aufgeteilt. Die Aufgabe beider Gruppen ist die Formulierung von Erklärbarkeitsanforderungen der 8 gegebenen Reviews unterschiedlicher Domänen, die in einer CSV-Datei, für die Nutzung des Tools, bzw. einer Excel Liste, für die manuelle Formulierung, bereitgestellt wurden. Die erste Gruppe nutzte für die ersten 4 Reviews das Tool. Die zweite Gruppe formulierte die Anforderungen für die gleichen Reviews ohne Hilfe. Die Zeit, die für die Formulierung der Anforderungen wurde dabei individuell je Teilnehmer gemessen.

Nach erfolgreicher Formulierung nutze die zweite Gruppe das Tool, um zu 4 neuen Reviews Anforderungen zu erstellen, wobei die erste Gruppe dies ohne Hilfe absolvieren musste. Ebenfalls wurde hier die Zeit gemessen. Abschließend wurde ein Fragebogen verteilt, um die Erfahrungen und Einschätzungen der Teilnehmer zu erfassen.

7.7 Analyse der Ergebnisse

Der Datensatz beinhaltet 58 Reviews jeweils mit einer Erklärbarkeitsanforderung automatisch generiert vom Tool, sowie eine manuell formulierte Erklärbarkeitsanforderung durch Anforderungsanalysten. Es gibt jeweils 8 Entscheidungen von Anforderungsanalysten, welche Erklärbarkeitsanforderung für die Praxis geeigneter ist. In diesem Kapitel soll die Entscheidung der 8 Anforderungsanalysten näher analysiert werden und Rückschlüsse bezüglich der Ergebnisse des Tools erfasst werden.

7.7.1 Kennzahlen des erstellten Datensatzes

In der folgenden Abbildung sind die Entscheidungen der Teilnehmer je Review dargestellt.

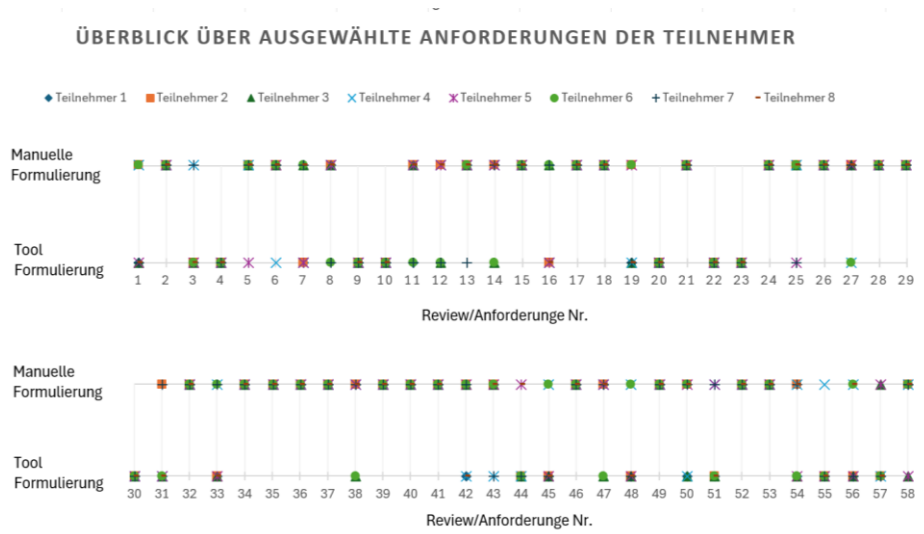


Abbildung 5: Entscheidungen der Anforderungsanalysten [Eigene Darstellung]

Auffallend ist, dass die Teilnehmer bei vielen manuell erstellten Anforderungen häufiger eine eindeutige Meinung haben. Weniger häufig ist die einstimmige Auswahl der Ergebnisse des Tools. Die folgende Abbildung 14 zeigt die Verteilung der eindeutigen Entscheidungen.

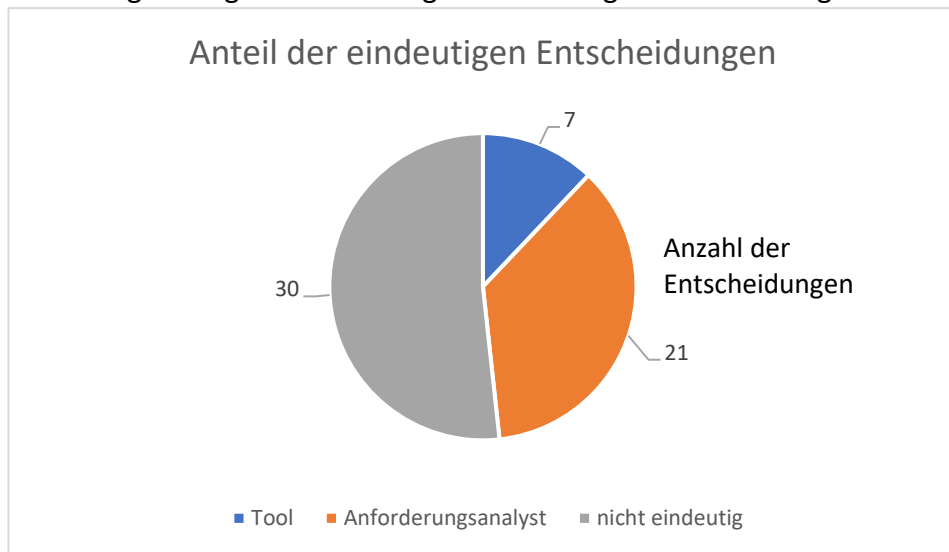


Abbildung 14: Anteil der eindeutigen Entscheidungen [Eigene Darstellung]

Insgesamt gab es 21 einstimmige Entscheidungen für die Formulierungen durch die Anforderungsanalysten, sowie 7 einstimmige Entscheidungen für die Formulierungen durch das Tool. Dies ergibt eine einstimmige Güte des Tools von etwa 12%. Der Anteil der einstimmigen Wahlen für die

Formulierung der Anforderungsanalysten beträgt etwa 36%. Damit erreicht das Tool 1/3 Anteil der einstimmigen Antworten. Dadurch liegt bei den übrigen 30 Datensätzen, also etwa 52%, keine eindeutige Entscheidung der Teilnehmer vor.

Wird der Mittelwert der Entscheidungen angenommen, so ergibt sich über den gesamten Datensatz die in Abbildung 15 dargestellte Statistik.

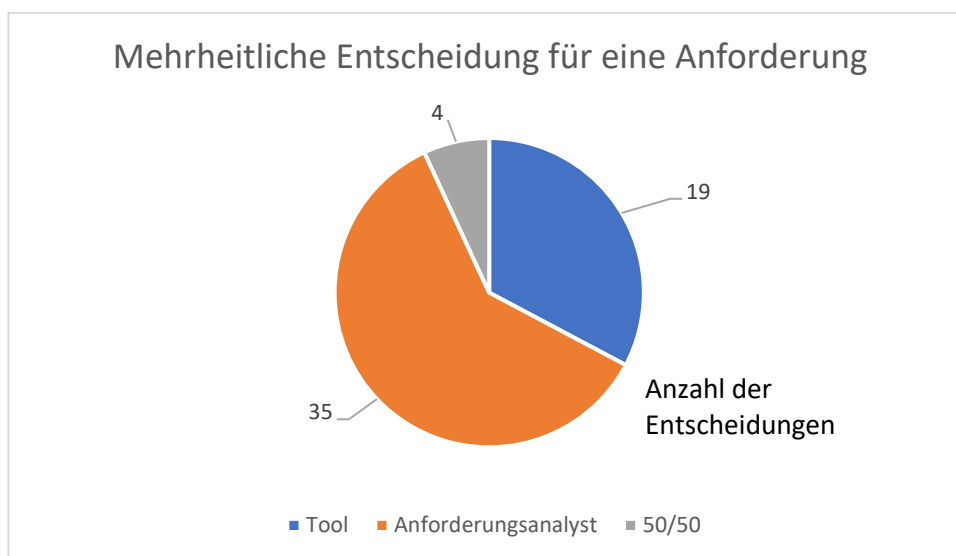


Abbildung 15: Mehrheitliche Entscheidung für eine Anforderung [Eigene Darstellung]

Somit erhält die Entscheidung für das Tool mit 19 Stimmen, die einen Anteil von etwa 33% ausmachen. Die Formulierungen der Anforderungsanalysten erhalten 35 Stimmen, was einen Anteil von etwa 60% ausmacht. Die Übrigen 4 Stimmen teilen sich weiterhin auf beide Anforderungen auf.

Insgesamt kann dazu gesagt werden, dass die Ergebnisse ähnlich zu den erwarteten Ergebnissen sind. Anzunehmen war die deutliche Mehrheit für die Auswahl der Formulierungen durch Anforderungsanalysten, die mehrheitlich bei etwa 60% liegt. Abweichend von der Erwartung ist jedoch der hohe Anteil in Höhe von 33% der Fälle, in denen das Tool durch mehrheitliche Entscheidung die bessere Anforderung formuliert hat.

7.7.2 Analyse bei eindeutigen Entscheidungen

In diesem Abschnitt werden die eindeutigen Entscheidungen und deren Ursachen für die jeweiligen Anforderungen näher betrachtet. Aus dem Datensatz ergibt sich eine Anzahl von 21 eindeutigen Entscheidungen für

die Formulierungen der Anforderungsanalysten und 7 für die Formulierungen des Tools. Betrachtete werden nur einzelne dieser Anforderungen. Für eine bessere Darstellungen werden nur die Zusammenfassungen des Reviews angegeben.

7.7.2.1 Entscheidung für die Formulierung der Anforderungsanalysten
 Im Folgenden wird ein Beispiel angegeben für die Entscheidung der Teilnehmer für die Formulierung der Anforderungsanalysten.

Zusammenfassung	Tool	Anforderungsanalysten
If I'm trying to choose a certain song from a playlist or album; I can't because it won't let me see the songs.	Das System muss erklären, warum bestimmte Songs in einer Playlist oder einem Album nicht ausgewählt werden können.	Das System muss über das Abo Modell aufklären und welche Funktionen dadurch eingeschränkt werden.

Tabelle 2: Gegenüberstellung Einstimmige Entscheidung für Formulierung der Anforderungsanalysten

Die Tabelle 2 enthält ein Beispiel aus dem Datensatz, der eindeutig für die Formulierung der Anforderungsanalysten entschieden wurde.

Der Erklärungsbedarf des Nutzers bezieht sich auf die App Spotify, genauer auf den Wunsch des Abspielens eines Songs aus einer Playlist. Die App lässt keine Auswahl des Songs zu.

Die Erklärbarkeitsanforderung des Tools orientiert sich stark an der Zusammenfassung des Reviews, nämlich die Anforderung, dass das System erklären muss, warum bestimmte Songs in einer Playlist [...] nicht ausgewählt werden können. Die Erklärbarkeitsanforderungen der Anforderungsanalysten hingegen bezieht sich bereits auf die Problematik, die dazu geführt hat, dass der Nutzer keine Songs auswählen kann. Es wird angenommen, dass der Nutzer nicht das notwendige Abonnement bezahlt, die die Funktionen ermöglichen würden. Es wird gefordert, dass grundsätzlich über die Funktionen und Einschränkungen des jeweiligen Abo Modells aufgeklärt werden muss.

Die Erkenntnisse, die für die Formulierung dieser Anforderung notwendig waren, gehen über die Informationen der Zusammenfassung hinaus. Es ist ein tieferes Wissen über die Domäne erforderlich, um die Ursache der Problematik zu identifizieren und diese in die Erklärbarkeitsanforderung

aufzunehmen. Grundsätzlich lässt sich daraus schließen, dass durch den Einfluss des Detailwissen der Domäne, es möglich sein kann, eine geeignetere Anforderung zu erstellen.

Diese Eigenschaft lässt sich auch bei weiteren Beispielen identifizieren, die eindeutig für die Formulierungen der Anforderungsanalysten entschieden wurden.

7.7.2.2 Entscheidung für die Formulierung des Tools

Im Gegensatz zum vorherigen Abschnitt werden hier die Beispiele betrachtet, in denen das Tool die bessere Anforderung generiert hat.

Zusammenfassung	Tool	Anforderungsanalysten
Die App stoppt zufällig und versucht, sich mit dem Internet zu verbinden, wenn sie offline ist, um Lieder abzuspielen, die ich bereits heruntergeladen habe. Warum muss ich online sein, um heruntergeladene Lieder abzuspielen?	Das System muss erklären, warum es eine Internetverbindung benötigt, um bereits heruntergeladene Lieder abzuspielen.	Das System muss dem Nutzer erklären, wofür der Online-Zugriff notwendig ist.

Tabelle 3: Gegenüberstellung Einstimmige Entscheidung für Formulierung des Tools

In der Tabelle 3 ist ein Beispiel abgebildet, dass die eine einstimmige Entscheidung für die Formulierung des Tools ergeben hat.

Der Erklärungsbedarf des Nutzers bezieht sich auf die App Spotify, genauer das Verhalten der App, das bei Verfügbarkeit die Internetverbindung zum Abspielen heruntergeladener Inhalte genutzt wird, wofür keine Internetverbindung nötig sei. Die Anforderungsanalysten erkannten den Erklärbarkeitsbedarf bezüglich der generellen Notwendigkeit einer Internetverbindung der App und forderten eine grundsätzliche Erklärung für den Online-Zugriff. Diese Anforderung scheint zu versuchen einen größeren Problembereich abzudecken, wobei jedoch der Erklärungsbedarf nicht klar formuliert wird und zu grob gehalten wird.

Das Tool hingegen erzielte in diesem Beispiel eine bessere Anforderung, da es alle Informationen zum Erklärbarkeitsbedarf aus der Zusammenfassung erfasst und klarer und präziser formuliert hat.

Die Abweichung der Anforderungsanalysten vom konkreten Erklärbarkeitsbedarf ist durch eine zu hohe Abstraktion des Erklärbarkeitsbedarfes zu erklären.

Ähnliche Sachverhalte sind an anderen Beispielen zu beobachten.

7.7.3 Analyse bei unterschiedlichen Entscheidungen

Ein wichtiger Einblick in die Ergebnisse des Datensatzes bieten die Beispiele, in denen die Stimmen der Anforderungsanalysten insgesamt nicht eindeutig einer Anforderung zuzuordnen war. In diesem Abschnitt werden die Beispiele betrachtet, in denen die Stimmverteilung genau die Hälfte beträgt.

Zusammenfassung	Tool	Anforderungsanalysten
Wie funktioniert das Smart-Shuffle nach dem Update?	Das System muss erklären, wie das Smart-Shuffle nach dem Update funktioniert.	Das System muss die "Smart-Shuffle" Funktion erklären.

Tabelle 4: Anforderungsauswahl basierend auf gleichem Stimmenanteil der Teilnehmer

Im Beispiel aus der Tabelle 4 hat der Nutzer einen Erklärungsbedarf bezüglich der Smart-Shuffle Funktion, da sich die Funktionalität dieser nach einem Update verändert hat. Die Anforderungen des Tools und der Anforderungsanalysten haben beide den grundsätzlichen Erklärbarkeitsbedarf, nämlich die Erklärung der Smart-Shuffle Funktion, erkannt. Die unterschiedlichen Meinungen über die geeignetere Formulierung entsteht vermutlich durch den Zusatz der Anforderung des Tools, dass das Update den Erklärbarkeitsbedarf ausgelöst hat. Die Meinungen über den Bedarf des zeitlichen Aspektes der Anforderung scheint hier strittig zu sein. Dies zeigt auch die Auswertung der Umfrage diesbezüglich. So geben die Teilnehmer ebenfalls sehr unterschiedliche Antworten und Begründungen bezüglich der Relevanz des zeitlichen Aspektes (z.B. durch ein Update) für die Formulierung des Erklärbarkeitsbedarfes, der Erklärbarkeitsanforderung und der Erklärung. Die Unentschlossenheit der Teilnehmer über die Relevanz der Information scheint daher die Ursache der ausgewogenen Stimmanteile zu sein.

Bei weiteren Beispielen mit gleichem Stimmenanteil ist ähnlicher Sachverhalt zu bemerken. Grundsätzlich wird der gleiche Erklärbarkeitsbedarf erkannt, jedoch gibt es geringe Unterschiede in den Formulierungen. So sind die Formulierungen des Tools meist etwas präziser an die Zusammenfassung angelehnt und die Formulierungen der Anforderungsanalysten abstrakter formuliert, ohne die grundlegenden Erklärungsbedarf auszulassen.

7.7.4 Analyse der praktischen Nutzung

Dieser Abschnitt bezieht sich auf die Analyse während der Nutzung des Tools durch Anforderungsanalysten, wobei insbesondere ein Vergleich der manuellen Methode zur Formulierung und der Automatisierung des Tools durchgeführt wird.

7.7.4.1 Zeitbedarf

Im zweiten Teil des Workshops wurde das Tool im direkten Vergleich zur manuellen Methode getestet. Die Zeitmessung ergab einen deutlichen Zeitvorteil. Die ersten 4 Anforderungen wurden durch die Gruppe, die das Tool nutzte, durchschnittlich in 2 Minuten und 40 Sekunden erstellt. Die Gruppe ohne Hilfe benötigte für die gleichen Anforderungen durchschnittlich 5 Minuten und 44 Sekunden. Daraus ergibt sich für den ersten Teil der Anforderung eine durchschnittliche Zeitersparnis von 3 Minuten und 4 Sekunden. Somit ist die Nutzung des Tools um etwa 53,48% schneller als die manuellen Formulierungen der Anforderungsanalysten.

Für die zweite Hälfte der Anforderungen ergeben sich folgende Werte:

Die Nutzung des Tools beanspruchte einen durchschnittlichen Zeitbedarf von 3 Minuten und 12 Sekunden. Dagegen benötigte die andere Gruppe durchschnittlich 5 Minuten und 37 Sekunden. Daraus ergibt sich ein Zeitvorteil von 2 Minuten und 25 Sekunden des Tools gegenüber der manuellen Formulierung. Das entspricht einer etwa 43% schnelleren Formulierung von Erklärbarkeitsanforderungen.

Insgesamt hat das Tool den Zeitbedarf für die Formulierung der Anforderungen im Durchschnitt um etwa 48% verbessert.

7.7.4.2 Beobachtungen während Formulierung von Anforderungen

Die Teilnehmer begannen mit der Generierung aller Zusammenfassungen und Erklärbarkeitsanforderungen unmittelbar nach dem Hochladen der

Reviews. Anschließend analysierten sie die generierten Zusammenfassungen und Erklärbarkeitsanforderungen und führten gelegentlich Anpassungen an den Zusammenfassungen durch, gefolgt von einer erneuten Generierung der entsprechenden Erklärbarkeitsanforderungen. In einigen Fällen erfolgte eine Kürzung der generierten Erklärbarkeitsanforderungen, wobei nur vereinzelt Formulierungen angepasst wurden. Der benötigte Zeitaufwand der Nutzer resultierte hauptsächlich aus diesen Interaktionen.

Bei der Formulierung der Erklärbarkeitsanforderungen ohne Unterstützung konnte folgendes beobachtet werden.

Die Teilnehmer benötigten die meiste Zeit für die Analyse des Reviews auf Erklärbarkeitsbedarf. Nach der ersten Anforderung wurden vor Analyse des Erklärbarkeitsbedarfs eine entsprechende Schablone aufgeschrieben, die erst nach Analyse des Reviews gefüllt wurde. Die Formulierungen des Erklärbarkeitsbedarfs benötigte bei den meisten Teilnehmern mehrere Versuche. Es schien, dass die Teilnehmer Herausforderungen hatten die passenden Formulierungen für den Erklärbarkeitsbedarf zu finden.

Bemerkt wurde außerdem, dass die Ergebnisse des Tools der letzten beiden gegebenen Reviews direkt von allen Teilnehmern übernommen wurden. Die Gruppe, die diese Reviews manuell erstellt hat, kommt zu unterschiedlichen Formulierungen, die jedoch den gleichen Erklärbarkeitsbedarf beinhalten.

7.7.4.3 Akzeptanz der Anforderungen

Ein entscheidender Faktor für die Bewertung der Automatisierung liegt in der Akzeptanz der generierten Ergebnisse. In einer Umfrage wurden die Teilnehmer sowohl nach ihrer Einschätzung gefragt, ob die erstellten Anforderungen mittels des Tools an die Softwareentwickler übermittelt werden sollen, als auch nach ihrer Einschätzung ohne Hilfe. Die Umfrageergebnisse deuten darauf hin, dass die Nutzer tendenziell ein höheres Maß an Vertrauen in die Ergebnisse des Tools haben. Von den 8 Teilnehmern gaben 3 an, dass sie die generierten Anforderungen mit Hilfe des Tools geeigneter einschätzten als ohne. Die übrigen Teilnehmer sahen keinen Unterschied in der Qualität zwischen den Ergebnissen mit und ohne Nutzung des Tools.

7.7.5 Fazit

Die Analyse des Datensatzes offenbart wertvolle Einblicke in die Präferenzen und Entscheidungsprozesse von Anforderungsanalysten

bezüglich automatisch generierter und manuell formulierter Erklärbarkeitsanforderungen. Während die manuell formulierten Anforderungen mit einem einstimmigen Anteil von 36% deutlich bevorzugt wurden, zeigt die vergleichsweise niedrige Rate von 12% bei den automatisch generierten Anforderungen, dass es Raum für Verbesserungen in der Qualität der Tool-gestützten Formulierungen gibt. Zu beachten ist bei diesem Ergebnis, dass es sich um die reine Ausgabe des Tools handelt ohne Überprüfung eines Anforderungsanalysten. Insbesondere aus diesem Umstand ist die von dem Tool erreichte Quote von 33% bei den mehrheitlichen Entscheidungen bemerkenswert und deutet auf das Potenzial der Automatisierung hin, relevante und präzise Anforderungen zu formulieren.

Die tiefergehende Betrachtung der eindeutigen Entscheidungen zeigt, dass das umfassende Domänenwissen der Anforderungsanalysten oft zu präziseren und situationsgerechteren Anforderungen führt. Diese Erkenntnis kann in der Weiterentwicklung des Tools genutzt werden, um die generierten Anforderungen besser auf den spezifischen Kontext anzupassen.

Abschließend zeigt die Analyse der praktischen Nutzung des Tools im direkten Vergleich zur manuellen Methode einen deutlichen Zeitvorteil in der Erstellung von Erklärbarkeitsanforderungen. Dies unterstreicht den Nutzen der Automatisierung für die Effizienz in der Anforderungserhebung. Die Beobachtungen während der Formulierung von Anforderungen und die Akzeptanz der generierten Ergebnisse weisen darauf hin, dass das Tool, trotz der notwendigen manuellen Verbesserungen, eine wertvolle Unterstützung in der praktischen Anwendung darstellt.

Insgesamt lässt sich feststellen, dass die Automatisierung von Erklärbarkeitsanforderungen bereits zu einer effizienteren Formulierung führen kann und das Potenzial hat, durch weiterführende Anpassungen und Integration von Domänenwissen eine gleichwertige und teilweise bessere Alternative zur manuellen Methode zu werden.

7.8 Reflektion

7.8.1 Ziele

Das primäre Ziel des Workshops war die Bewertung der automatisierten Ergebnisse des Tools im direkten Vergleich zu den von Anforderungsanalysten manuell erstellten Erklärbarkeitsanforderungen. Durch die quantitativen Entscheidungen der Anforderungsanalysten, die

ohne Kenntnis über den Ursprung der Erklärbarkeitsanforderungen getroffen wurden, konnte eine unvoreingenommene Einschätzung der Qualität der automatisierten gegenüber den manuellen Ergebnissen gewährleistet werden. Ein weiteres Ziel war die Analyse der Nutzung und Akzeptanz des Tools durch die Anforderungsanalysten. Insbesondere wurde die Effizienz des Tools durch Zeitmessungen während seiner aktiven Nutzung im Vergleich zur traditionellen manuellen Methode evaluiert. Die Ziele konnten alle erreicht werden.

7.8.2 Workshop Design und Durchführung

Für die systematische Bewertung und den Vergleich der Ergebnisse wurde ein quantitatives Studiendesign gewählt. Die Teilnehmer wurden in zwei Gruppen aufgeteilt, um sowohl die automatisierte als auch die manuelle Methode unter gleichen Bedingungen zu testen, was einen direkten und objektiven Vergleich ermöglichte. Diese Gruppenaufteilung zielte darauf ab, eine gleichmäßige Verteilung der subjektiven Eigenschaften der Teilnehmer zu erreichen, um den Einfluss individueller Faktoren auf die Ergebnisse zu minimieren und einen direkten Vergleich der Methoden zu ermöglichen. Obwohl die Aufteilung der Gruppen theoretisch subjektive Eigenschaften in den Vergleich einbringen könnte, zeigten die Auswertungen der Ergebnisse keinen signifikanten Einfluss dieser Variablen. Die Akzeptanz des Tools wurde durch die Reflexion der Teilnehmer über ihre eigenen erstellten Ergebnisse und durch Analyse ihrer getroffenen Entscheidungen für die automatisierten Ergebnisse ermittelt, wodurch teilweise die Präferenz für die automatisierten Erklärbarkeitsanforderungen bestätigt werden konnte.

Durch eine kleinschrittigere Zeitmessung hätten im Nachhinein weitere Erkenntnisse gezogen werden können. Dadurch hätte der Zeitbedarf für die Erstellung einer Zusammenfassung oder Erklärbarkeitsanforderung genauere Schlüsse ermöglicht. Somit konnte nur der Prozess über jeweils vier unterschiedliche Anforderungen ausgewertet werden. Diese Zeitmessung hätte jedoch einen deutlich höheren Zeitbedarf für jeden Teilnehmer benötigt, welche in der vorgegebenen Zeit und in Form des Workshops unangebracht gewesen wäre.

Die Form des Workshops erbrachte den Vorteil der gemeinsamen Abstimmung über das Verständnis von Erklärbarkeit und der Einführung in das Thema, sowie das Aufrechterhalten der Motivation während des Entscheidungsprozesses.

7.8.3 Teilnehmerfeedback

Die Teilnehmer des Workshops empfanden die Durchführung des Workshops und insbesondere die Thematik zur Automatisierung als interessantes Themenfeld. Sie gaben in direktem Gespräch positives Feedback zur Aufbereitung der Ergebnisse, sowie zur Durchführung des Workshops und wirkten während der Praxisphase durchgehend motiviert. Während der Präsentation, insbesondere während der Definition von Erklärbarkeit, konnte eine Unruhe bei den Teilnehmern festgestellt werden. Nach Anwendung der Definition anhand von alltäglichen Beispielen konnte unter den Teilnehmern eine höhere Zufriedenheit beobachtet werden.

Ein interaktiver praktischer Bezug zur Theorie von Erklärbarkeit und Erklärbarkeitsanforderungen scheint eine höhere Bereitschaft zur Mitwirkung bei den Teilnehmern auszulösen. Letztlich bedankten sich die Teilnehmer für die Teilnahme am Workshop. In der durchgeführten Umfrage konnten keine weiteren Anmerkungen zum Workshop gefunden werden.

8 Diskussion

8.1 Beantwortung der Forschungsfragen

[FF1] Wie kann ein Prozess definiert werden, der effizient Erklärbarkeitsanforderungen und Erklärungen auf Grundlage von Reviews generiert?

Die durchgeführten Studien und die Entwicklung des Automatisierungstools haben gezeigt, dass ein effizienter Prozess zur Generierung von Erklärbarkeitsanforderungen durch die Kombination von automatisierten und manuellen Methoden definiert werden kann. Das Tool, das zur Generierung der Inhalte auf einem GPT-Model basiert, war in der Lage, 58 Erklärbarkeitsanforderungen ohne manuellen Eingriff zu erstellen, wobei mehrheitlich mit einem Anteil von 33% eine geeignetere Anforderung erstellt wurde. Daraus ergibt sich die Leistungsfähigkeit der Automatisierung, die durch anschließende Überprüfung und Anpassung durch Anforderungsanalysten den traditionellen Prozess zur Erstellung von Erklärbarkeitsanforderungen und Erklärungen qualitativ mindestens beibehält und in einigen Fällen verbessert, sowie eine deutliche Zeitersparnis gewährleistet.

[FF2] Wie hilft das Wissen über die Kategorie eines Erklärbarkeitsbedarfs bei der Formulierung von Anforderungen?

In der ersten Studie wurden bei der Formulierung von Erklärbarkeitsanforderungen die Effektivität des Wissens über die Kategorie überprüft. Nach Formulierungen beider Hälften von Erklärbarkeitsanforderungen, jeweils mit und ohne Angabe einer Kategorie, konnte kein unterschiedlicher Zeitbedarf festgestellt werden. Die Auswertung der anschließenden Umfrage ergab Unstimmigkeiten zwischen den Teilnehmer über den Nutzen der Kategorie.

Die Nutzung der Kategorie als Metrik für die Auswahl der Schablone wurde im Rahmen dieser Arbeit nicht weiter betrachtet, da eine universelle Schablone unabhängig der Kategorie identifiziert wurde.

Eine Analyse der zweiten Studie, durch Betrachtung der Häufigkeit der gewählten Antworten mit und ohne Kategorie, ergibt ebenfalls keine messbaren Vorteile.

Daraus lässt sich schließen, dass das Wissen über die Kategorie effektiv keine Vorteile bringt. Die Qualität der Anforderungen scheint nicht von der Kategorie des Erklärbarkeitsbedarfs beeinflusst zu werden. Es kann nicht ausgeschlossen werden, dass in Einzelfällen die Kenntnis der Kategorie eine Unterstützung zur Formulierung des Erklärbarkeitsbedarfs darstellt. Eine statistisch messbare Tendenz hierzu konnte hingegen nicht ermittelt werden.

[FF3] Wie hoch ist die Akzeptanz der Experten von automatisch generierten Erklärbarkeitsanforderungen?

Die Akzeptanz der Experten von automatisch generierten Erklärbarkeitsanforderungen, basierend auf den Ergebnissen der zweiten Studie, zeigt ein differenziertes Bild. Während das automatisierte Tool in der Lage war, Erklärbarkeitsanforderungen effizient zu generieren, wiesen die Ergebnisse eine gemischte Akzeptanz unter den Experten auf. Die Mehrheit bevorzugte im direkten Vergleich häufiger die traditionellen Formulierungen. Während der Nutzung des Tools konnte durch Interaktion mit den automatisierten Ergebnissen eine höhere Sicherheit bezüglich der Eignung der formulierten Anforderung festgestellt werden.

Daraus lässt sich schließen, dass die automatisch generierten Ergebnisse des Tools durch die Experten tendenziell für ungeeigneter empfunden, jedoch den Mehrwert bei Anpassung der Ergebnisse schätzen und damit verbunden eine höhere Qualität ihrer Anforderung. Die Akzeptanz der Experten zeigt eine klare Tendenz zur Akzeptanz der Ergebnisse, als Unterstützung.

[FF4] Wie können automatisch generierte Erklärungen in der Praxis genutzt werden?

Automatisch generierte Erklärungen können in der Praxis als erste Entwürfe genutzt werden, die von Experten überprüft und verfeinert werden. Sie bieten eine Grundlage für schnelle und kosteneffiziente Updates von Software-Dokumentationen und Hilfesystemen, insbesondere in dynamischen Umgebungen, wo Softwarefunktionen häufig aktualisiert werden. Die Auswertung der Umfrage der ersten Studie zeigt außerdem, dass Anforderungsanalysten nicht die Notwendigkeit sehen, für jeden Erklärbarkeitsbedarf in Reviews eine Erklärbarkeitsanforderung für die Software zu erfassen. Trotzdem sollten diese Erklärbarkeitsbedarfe genutzt werden, um den Nutzern ein besseres

Nutzererlebnis zu verschaffen. Die Erklärungen können genutzt werden, um eine direkte Antwort an die Nutzer bereitzustellen. Sie haben durch den definierten Prozess zur Erstellung der Erklärung einen starken Bezug zum Review und können bereits die benötigten Informationen für den Nutzer enthalten, um Ihren Erklärbarkeitsbedarf zu befriedigen.

[FF5] Wie kann die Generierung von Erklärbarkeitsanforderungen automatisiert werden?

In Kapitel 5 wurden verschiedene Ansätze und Aspekte der Automatisierung besprochen und diskutiert. Es ergibt sich, dass der hybride Ansatz die meisten Vorteile der anderen Ansätze vereint und eine vollständige Automatisierung ermöglichen könnte. Durch Erprobung der Erstellung von Erklärbarkeitsanforderung mittels GPT-Modellen, konnte eine Verbesserung der Ergebnisse durch eine Zusammenfassung des Reviews erzielt werden. Für die Erstellung der Zusammenfassung ist die Methodik eines GPT-Modells als effektivste Methode zu bewerten. Bei der Auswertung der ersten Studie konnte eine Schablone identifiziert werden. Daraus ergab sich nach dem hybriden Ansatz für die Formulierung von Erklärbarkeitsanforderungen die Nutzung eines GPT-Modells mit Vorgabe einer Schablone als optimalen Ansatz.

Die Ergebnisse sind zwar automatisiert erstellt, jedoch benötigt die Nutzung des GPT-Modells eine genauere Betrachtung bezüglich der Auswahl eines Modells, des spezifischen Trainings dessen und der Einstellung der entsprechenden Parameter, um eine zuverlässige Automatisierung ohne Eingriff durch Anforderungsanalysten zu gewährleisten.

[FF6] Wie kann die Generierung von Erklärungen automatisiert werden?

In Kapitel 5.3.3 wurde diskutiert, welche Besonderheiten bei der Generierung von Erklärungen zu beachten sind. Erarbeitet wurde der Ansatz zur Nutzung eines GPT-Modells. Die Ergebnisse der ersten Studie haben gezeigt, dass für die Formulierung von guten Erklärungen Domänenwissen der App erforderlich ist. Ein GPT-Modell kann bereits Zugriff auf Domänenwissen der App haben und den Umstand des fehlenden Domänenwissens damit ausgleichen. Eine vollständige Automatisierung ist wie bei der Automatisierung der Erklärbarkeitsanforderungen zu berücksichtigen. Die automatisiert erstellten Erklärungen sollten durch Experten überprüft werden.

8.2 Interpretation

Die Motivation der Arbeit war die Entwicklung einer Automatisierungslösung für die Erstellung von Erklärbarkeitsanforderungen und Erklärungen, um den ständig wachsenden Erklärbarkeitsbedarf durch Reviews entgegenzuwirken und einen Schritt für die Akzeptanz der betroffenen Software durch die Nutzer zu gehen.

Die Forschungsergebnisse haben gezeigt, dass automatisierte Methoden zur Generierung von Erklärbarkeitsanforderungen und Erklärungen effektiv implementiert werden können, um den Prozess zu unterstützen und die Kosten zu senken. Allerdings hat die Studie auch die Grenzen der Automatisierung aufgezeigt, insbesondere in Bezug auf die Kontextsensitivität und die Tiefe der generierten Inhalte.

Aus den Ergebnissen lässt sich schlussfolgern, dass der Mensch ein entscheidender Faktor bleibt, um die Qualität und Relevanz der generierten Anforderungen und Erklärungen zu sichern. Automatisierte Tools diesbezüglich sollten daher nicht als Ersatz, sondern als Ergänzung zu den Fähigkeiten der Anforderungsanalysten gesehen werden. Die Tools steigern die Effizienz, erfordern jedoch eine Überprüfung und Anpassung durch Fachkundige.

Schließlich hat die Arbeit gezeigt, dass es möglich ist die Erstellung von Erklärbarkeitsanforderungen und Erklärungen zu automatisieren und ermöglicht eine schnellere und sichere Verarbeitung des Erklärbarkeitsbedarfs. Eine vollständige Automatisierung ohne Überprüfung von Experten ist nicht zu empfehlen. Dadurch konnte ein wichtiger Beitrag für die Problemstellung der Arbeit erreicht werden, jedoch sind weitere Schritte, wie z.B. die Integration der Erkennung von Erklärbarkeitsbedarf, notwendig, um das übergeordnete Ziel zu erreichen.

9 Threats to validity

Wohlin et al. [10] beschreibt in seiner Arbeit verschiedene Kategorien die, die Glaubwürdigkeit und Wissenschaftlichkeit der genutzten Methoden und Daten beeinträchtigen können. Im Folgenden werden Aspekte dieser Arbeit zu den jeweiligen Kategorien beschrieben.

9.1 Construct Validity

Die in der Studie generierten Erklärungen könnten von dem Verständnis abweichen, was in der Theorie und Praxis unter "Erklärbarkeit" verstanden wird.

Um diesem Threat entgegenzuwirken, wurden in der Einleitung der Studien die Definition von Erklärbarkeitsbedarf vorgestellt und mit den Teilnehmern ein Beispiel zu einer Erklärbarkeitsanforderung formuliert.

9.2 Internal Validity

In Bezug auf die Verwendung von GPT-Modellen als Black-Box-Systeme besteht die Gefahr, dass nicht klar ist, welche internen Prozesse zu den generierten Antworten führen. Diese Unklarheit kann kausale Schlüsse schwächen, da nicht eindeutig ist, ob die generierten Erklärbarkeitsanforderungen direkt auf die manipulierten Eingaben (z. B. geänderte Prompts oder unterschiedliche Reviews) oder auf unbekannte, nicht kontrollierte Faktoren innerhalb des Modells zurückzuführen sind.

Um diesem Threat entgegenzuwirken wurden die geänderten Prompts mehrfach überprüft und auf gleiche Auswirkungen getestet. Daher kann trotz des Black-Box-Systems angenommen werden, dass die Ergebnisse aktiv durch manipulierte Eingaben beeinflusst werden können.

In der Masterarbeit könnte die Validität der Schlussfolgerungen aus dem zeitlichen Vergleich in der zweiten Studie zwischen der automatisierten Methode und der traditionellen Formulierung von Erklärbarkeitsanforderungen durch externe Variablen und die Unterschiede in den Reviews kompromittiert werden.

Um diesem Threat entgegenzuwirken wurden die Teilnehmer in Gruppen aufgeteilt, denen jeweils die gleichen Reviews vorgelegt wurden.

9.3 External Validity

Die geringe Anzahl von Studienteilnehmern kann die Generalisierbarkeit der Ergebnisse einschränken, da die Stichprobe möglicherweise nicht repräsentativ für die größere Zielgruppe ist. In dieser Arbeit bedeutet dies, dass die Ergebnisse möglicherweise nicht auf andere Kontexte, in denen Erklärbarkeitsanforderungen eine Rolle spielen, übertragbar sind.

9.4 Reliability

Die Verwendung von GPT-Modellen, deren interne Verarbeitungsmechanismen nicht vollständig transparent sind, könnte die Reliability beeinträchtigen. Unterschiedliche Instanzen des Modells oder sogar unterschiedliche Versionen desselben Modells könnten bei denselben Eingaben unterschiedliche Ergebnisse liefern. Dies stellt eine Herausforderung für die Konsistenz der Daten dar und könnte die Reliability der Studie beeinflussen, da hypothetisch nicht sichergestellt werden kann, dass ein anderer Forscher genau die gleichen Ergebnisse reproduzieren könnte.

10 Zusammenfassung und Ausblick

In diesem Kapitel werden das Forschungsziel und die erreichten Ergebnisse zusammengefasst.

10.1 Zusammenfassung

Der Nutzen Erklärungsbedarf in Reviews festzustellen, wurde durch mehrere Arbeiten bereits erkannt und bearbeitet. Eine Automatisierung, um aus diesem Erklärbarkeitsbedarf passende Anforderungen und Erklärungen zu erstellen, die in der Praxis genutzt werden können, gab es noch nicht. Diese Arbeit sollte das Problem für eine effiziente Bearbeitung des Erklärbarkeitsbedarfs angehen. Dafür wurde mittels einer Studie ein Datensatz erstellt, der Reviews zu der App „Spotify“ enthält. Diese Reviews decken die gesamte Taxonomie von Erklärbarkeitsbedarf ab. Anforderungsanalysten aus der Praxis haben Erklärbarkeitsanforderungen und Erklärungen zu diesem formuliert. Auf der Grundlage konnte eine Schablone für die Formulierungen von Erklärbarkeitsanforderungen identifiziert werden, die in den erarbeiteten Ansatz zur Automatisierung integriert wurde. Daraus ergab sich die Nutzung eines GPT-Modells zur Formulierung der Erklärbarkeitsanforderungen und Erklärungen. Zur Erhebung von Anforderungen für eine Automatisierungslösung wurde im Gespräch mit zwei Teilnehmern des Workshops ein Mock-Up erstellt, sowie darauf basierend Anforderungen für das Tool erfasst. Auf Grundlage der Anforderungen wurde das Tool entwickelt. Für die Evaluierung der Ergebnisse wurden in einer weiteren Studie die reinen Ergebnisse des Tools mit denen der Anforderungsanalysten verglichen, sowie die Interaktion und Akzeptanz während der Formulierung von Erklärbarkeitsanforderungen mit und ohne des Tools. Erstaunlicherweise haben bereits die reinen Ergebnisse des Tools ohne Verbesserung eines Anforderungsanalysten bei einem Anteil von 33% der Anforderungen die besseren Ergebnisse erzielt. Nach aktiver Nutzung durch die Anforderungsanalysten konnte ebenfalls eine Tendenz zu einer höheren Zufriedenheit bezüglich der Eignung der Anforderungen festgestellt werden. Die Generierung von Erklärungen ergab durch die erste Studie, dass ein Bedarf von Domänenwissen für die Bewertung einer passenden Erklärung notwendig ist. Eine Bewertung der Automatisierung ist daher hauptsächlich durch die Experten der jeweiligen Domäne möglich. Daher

wurde der Fokus der Arbeit auf die Automatisierung der Erklärbarkeitsanforderungen gelegt. Zu Beachten ist sowohl bei der automatischen Generierung der Erklärbarkeitsanforderungen und Erklärungen das Überprüfen durch Anforderungsanalysten.

Abschließend hat die Arbeit auf Grundlage einer Studie ein Konzept zur Automatisierung betrachtet, sowie dessen Automatisierung und Evaluierung in der Praxis.

10.2 Ausblick

Auf dieser Arbeit aufbauend können folgende Probleme angegangen werden. Die Automatisierung zur Erstellung von Erklärbarkeitsanforderungen und Erklärungen ist nur ein Teil eines Prozesses.

Folglich ist die Entwicklung einer Schnittstelle erforderlich, die den Erklärbarkeitsbedarf in den Reviews identifiziert und darauf aufbauend die Automatisierung von Erklärbarkeitsanforderungen und Erklärungen, wie in dieser Arbeit dargestellt, ermöglicht. Weiterhin sollten Möglichkeiten betrachtet werden, wie die generierten Erklärbarkeitsanforderungen zusammengefasst werden. Ebenso sollten Methoden analysiert werden, wie eine Entscheidung getroffen werden kann, ob für die Reduzierung des Erklärbarkeitsbedarfs eines Reviews eine Anforderung an die Software erstellt werden muss, oder nur eine individuelle Antwort auf den Nutzer erfordert. Weiterhin kann die Qualität der Ergebnisse der Automatisierung verbessert werden, indem neue GPT-Modelle verwendet werden.

Literaturverzeichnis

- [1] "Hambacher Erklärung zur Künstlichen Intelligenz: Entschließung der 97. Konferenz der unabhängigen Datenschutzaufsichtsbehörden des Bundes und der Länder vom 3. April 2019," *Datenschutz und Datensicherheit*, 2019.
- [2] L. Pechmann, M. Mildner, T. Suthau et al., "Regulatorische Anforderungen an Lösungen der künstlichen Intelligenz im Gesundheitswesen," in *Künstliche Intelligenz im Gesundheitswesen*, M. A. Pfannstiel, Ed., pp. 175–198, Springer Fachmedien Wiesbaden, Wiesbaden, 2022.
- [3] K. Amann, J. Petzold, and M. Westerkamp, "Künstliche Intelligenz im Umfeld des Managements und Controllings," in *Management und Controlling*, K. Amann, J. Petzold, and M. Westerkamp, Eds., pp. 243–250, Springer Fachmedien Wiesbaden, Wiesbaden, 2020.
- [4] J. Wagner, "Auswirkungen von Legal Tech," in *Legal Tech und Legal Robots*, J. Wagner, Ed., pp. 77–99, Springer Fachmedien Wiesbaden, Wiesbaden, 2020.
- [5] W. Maalej and H. Nabil, "Bug report, feature request, or simply praise? On automatically classifying app reviews," in *2015 IEEE 23rd International Requirements Engineering Conference (RE)*, pp. 116–125, IEEE, 2015.
- [6] U. John, P. Hofstedt, and A. Wolf, eds., *Deklarative Ansätze zur Künstlichen Intelligenz - punktuelle Beiträge: W(C)LP 2018 : Post-Proceedings des 32. Workshops für (Constraint) Logische Programmierung*, Hochschule für Wirtschaft Technik und Kultur hwtk - University of Applied Sciences, Berlin, September 2019.
- [7] D. Kupczyk, *Automatisierte Detektion von Erklärungsbedarf in Nutzerfeedback zu Software*, Masterarbeit, Leibniz Universität Hannover, 2023.
- [8] T. Kurtz, *Entwicklung einer Software zur Extrahierung und Analyse von Reviews aus App Stores*, Bachelorarbeit, Leibniz Universität Hannover, 2023.
- [9] J. Droste, H. Deters, M. Obaidi et al., "Explanations in Everyday Software Systems: Towards a Taxonomy for Explainability Needs," in *IEEE 32nd international requirements engineering conference (RE)*.
- [10] C. Wohlin, P. Runeson, M. Höst et al., *Experimentation in Software Engineering*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [11] M. A. Kohl, K. Baum, M. Langer et al., "Explainability as a Non-Functional Requirement," in *2019 IEEE 27th International Requirements Engineering Conference (RE)*, pp. 363–368, IEEE, 2019.
- [12] L. Chazette, W. Brunotte, and T. Speith, "Explainable software systems: from requirements analysis to system evaluation," *Requirements Engineering*, vol. 27, no. 4, pp. 457–487, 2022.
- [13] L. Chazette, *Requirements engineering for explainable systems*, Dissertation, Gottfried Wilhelm Leibniz Universität Hannover.
- [14] L. Chazette, W. Brunotte, and T. Speith, "Exploring Explainability: A Definition, a Model, and a Knowledge Catalogue," 2021.
- [15] D. Gunning, M. Stefik, J. Choi et al., "XAI-Explainable artificial intelligence," *Science robotics*, vol. 4, no. 37, 2019.

- [16] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser et al., "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82–115, 2020.
- [17] "Systems and software engineering - Systems and software quality requirements and evaluation (SQuaRE) - System and software quality models: International Standard ISO/IEC 25010 = Ingénierie des systèmes et du logiciel - exigences de qualité des systèmes et du logiciel (SQuaRE) - Modèles de qualité du système et du logiciel," Internationale Organisation für Normung; Internationale Elektrotechnische Kommission, 2011.
- [18] J. Ooge and K. Verbert, "Explaining Artificial Intelligence with Tailored Interactive Visualisations," in *27th International Conference on Intelligent User Interfaces*, pp. 120–123, ACM, New York, NY, USA, 2022.
- [19] A. Dillon and M. Morris, "User Acceptance of Information Technology: Theories and Models," *Annual Review of Information Science and Technology*, vol. 31, 1996.
- [20] N. Scharowski, S. A. C. Perrig, M. Svab et al., "Exploring the effects of human-centered AI explanations on trust and reliance," *Frontiers in Computer Science*, vol. 5, 2023.
- [21] K. R. Chowdhary, "Natural Language Processing," in *Fundamentals of Artificial Intelligence*, K. R. Chowdhary, Ed., pp. 603–649, Springer India, New Delhi, 2020.
- [22] S. C. Fanni, M. Febi, G. Aghakhanyan et al., "Natural Language Processing," in *Introduction to Artificial Intelligence*, M. E. Klontzas, S. C. Fanni, and E. Neri, Eds., pp. 87–99, Springer International Publishing, Cham, 2023.
- [23] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention Is All You Need," 6/12/2017, <http://arxiv.org/pdf/1706.03762v7>.
- [24] Radford et al., "Improving language understanding by generative pre-training," 2018, <https://www.mikecaptain.com/resources/pdf/gpt-1.pdf>.
- [25] J. Droste, H. Deters, J. Puglisi et al., "Designing End-User Personas for Explainability Requirements Using Mixed Methods Research," in *2023 IEEE 31st International Requirements Engineering Conference Workshops (REW)*, pp. 129–135, IEEE, 2023.
- [26] H. Deters, J. Droste, and K. Schneider, "A Means to what End? Evaluating the Explainability of Software Systems using Goal-Oriented Heuristics," in *Proceedings of the 27th International Conference on Evaluation and Assessment in Software Engineering*, pp. 329–338, ACM, New York, NY, USA, 2023.
- [27] J. Bohnstedt, *Untersuchung des Einflusses von Domänenwissen auf den Erklärungsbedarf der Nutzenden von Software*, Bachelorarbeit, Gottfried Wilhelm Leibniz Universität Hannover, 2024.
- [28] N. Tsakalakis, S. Stalla-Bourdillon, T. D. Huynh et al., "A taxonomy of explanations to support Explainability-by-Design," 6/9/2022, <http://arxiv.org/pdf/2206.04438v1>.
- [29] M. Obaidi and J. Klünder, "Development and Application of Sentiment Analysis Tools in Software Engineering: A Systematic Literature Review," in *Evaluation and Assessment in Software Engineering*, R. Chitchyan, J. Li, B. Weber et al., Eds., pp. 80–89, ACM, New York, NY, USA, 2021.
- [30] T. N. Fitria, "Artificial intelligence (AI) technology in OpenAI ChatGPT application: A review of ChatGPT in writing English essay," *ELT Forum: Journal of English Language Teaching*, vol. 12, no. 1, pp. 44–58, 2023.

- [31] E. M. Bender, T. Gebru, A. McMillan-Major et al., "On the Dangers of Stochastic Parrots," in *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pp. 610–623, ACM, New York, NY, USA, 2021.
- [32] I. S. Schwartz, K. E. Link, R. Daneshjou et al., "Black Box Warning: Large Language Models and the Future of Infectious Diseases Consultation," *Clinical Infectious Diseases: An Official Publication of the Infectious Diseases Society of America*, vol. 78, no. 4, pp. 860–866, 2023.
- [33] N. Hafner and S. Hundertmark, "Large Language Models im Kundendialog – Chancen, Risiken, Ausblicke," in *Kundendialog-Management*, N. Hafner and S. Hundertmark, Eds., pp. 239–248, Springer Fachmedien Wiesbaden, Wiesbaden, 2024.
- [34] T. J. Sejnowski, "Large Language Models and the Reverse Turing Test," *Neural computation*, vol. 35, no. 3, pp. 309–342, 2023.
- [35] E. Sauerwein, "Das Kano-Modell der Kundenzufriedenheit," in *Das Kano-Modell der Kundenzufriedenheit*, E. Sauerwein, Ed., pp. 27–55, Deutscher Universitätsverlag, Wiesbaden, 2000.
- [36] C. Schmidt and C. Piccin, "Anforderungen an den kollaborativen Arbeitsplatz," *HMD Praxis der Wirtschaftsinformatik*, vol. 54, no. 6, pp. 874–886, 2017.
- [37] M. Grinberg and W. Moch, *Flask: Tworzenie aplikacji internetowych w Pythonie*, Helion, Gliwice, opyright 2020.