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

Untersuchung von Zusammenhängen zwischen App-Eigenschaften und Arten von Erklärungsbedarf

Analysis of Correlations Between App Features and Types of Explanation Needs

Bachelorarbeit

im Studiengang technische Informatik

von

Johannes Lumpe

Prüfer: Prof. Dr. Kurt Schneider
Zweitprüfer: Dr. Jil Klünder
Betreuer: Martin Obaidi

Hannover, 09.09.2024

Erklärung der Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Bachelorarbeit selbstständig 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 09.09.2024

Johannes Lumpe

Zusammenfassung

In der heutigen digitalisierten Welt, in der Software in immer mehr Bereichen des täglichen Lebens Anwendung findet, wird es immer wichtiger, dass Nutzer mit unterschiedlichem Kenntnisstand nachvollziehen können, was die Software leistet und wieso sie das macht. Die vorliegende Arbeit wurde verfasst, um einen Lösungsansatz für die Vorhersage von zu erwarteten Erklärungsbedarfen zu finden, mit dessen Hilfe Erklärungsbedarfe bereits bei der Softwareentwicklung berücksichtigt werden können.
Dazu wurde in dieser Bachelorarbeit ein Datensatz aus Reviews mit unterschiedlichen Arten von Erklärungsbedarfen analysiert. Für die Analyse wurden die Metadaten ausgewählter Apps gecrawlt. Die Analyse lässt sich in zwei Teile aufteilen. Zum einen wurden unterschiedliche Korrelationsverfahren zwischen Erklärungsbedarfen und App-Eigenschaften angewendet. Als Ergebnis wurde ein moderater Einfluss festgestellt, wenn die App-Eigenschaft aus dem Benutzter-Feedback stammt. Die Ergebnisse weisen jedoch auf einen schwachen Zusammenhang mit einigen moderaten Zusammenhängen zwischen den vom Unternehmen festgelegten App-Eigenschaften und den Erklärungsbedarfen hin.
Der zweite Teil der Analyse wurde mit der logistischen Regression durchgeführt. Für diese wurde ein neuer Datensatz mit Reviews und neuen Metadaten für die neuen Apps erstellt. Mit dem ersten Datensatz wurden die logisitischen Regressionen trainiert und mit dem neuen Datensatz getestet. Bei der Testung der logistischen Regressionen kam heraus, dass mit den App-Eigenschaften, die in dieser Arbeit genutzt wurden, keine Möglichkeit besteht weder eine Vorhersage über die Art des Erklärungsbedarfs zu treffen, noch ob überhaupt Erklärungsbedarf besteht.

Abstract

Analysis of Correlations Between App Features and Types of Explanation Needs

In today's digitalized world, where software is increasingly used in various areas of daily life, it is becoming more important for users with different levels of knowledge to understand what the software does and why it does it. This thesis was written to find an approach for predicting expected explanation needs, which can be considered during software development.
For this purpose, a dataset of reviews with different types of explanation needs was analyzed in this bachelor's thesis. The metadata of selected apps were crawled for the analysis. The analysis can be divided into two parts. On the one hand, different correlation methods between explanation needs and app properties were applied. As a result, a moderate influence was found when the app property comes from user feedback. However, the results indicate a weak correlation with some moderate correlations between the app properties set by the company and the explanation needs.
The second part of the analysis was carried out using logistic regression. For this, a new dataset with reviews and new metadata for the new apps was created. The logistic regressions were trained with the first dataset and tested with the new dataset. The testing of the logistic regressions revealed that with the app properties used in this work, it is not possible to predict the type of explanation need or whether there is any explanation need at all.

Inhaltsverzeichnis

Kapitel 1

Einleitung

Durch den zunehmenden Einfluss von Software in der heutigen Gesellschaft und die daraus komplexer werdenden Softwaresysteme, die Entscheidungen treffen, wird der Wunsch nach Erklärbarkeit der Software größer [8]. Diese Software tritt immer mehr in kritische Bereiche des Lebens ein, wie z.B. in den Gesundheitssektor, in dem eine Nachvollziehbarkeit von Entscheidungen sehr wichtig ist [5]. Zudem ist es für App-Entwickler relevant im Vorhinein Erklärungsbedarfe zu identifizieren und zu beheben [18]. In diesem Kontext entsteht die Frage, ob anhand von App-Eigenschaften eine Vorhersage von Erklärungsbedarf möglich ist [18]. Durch eine Vorhersage, welcher Erklärungsbedarf auftreten wird, besteht die Möglichkeit ein erklärbares System zu entwickeln [8], dass die vom Nutzer gewünschten Erklärungen schon im System integriert hat. Für die Vorhersage ist wichtig, welche Erklärungsbedarfe vorliegen. Aus diesem Grund ist eine Kategorisierung der Erklärungsbedarfe unerlässlich. Dafür gibt es unterschiedliche Formen, in die man Erklärungsbedarfe kategorisieren kann. Unterbusch et al. [23], Droste et al. [10] geben hierzu Beispiele. Eine Möglichkeit ist es in expliziten und impliziten Erklärungsbedarfen zu unterteilen, sowie Droste et al. [10] in ihrer Arbeit. Eine weitere Möglichkeit ist es, den Erklärungsbedarf in Kategorien zu unterteilen, wie aus der Taxonomie von Droste et al. [10] hervorgeht.
App-Eigenschaften sind vielfältig und können unterschiedliche Aspekte einer App klassifizieren. Dadurch könnte sich bei der Softwareentwicklung ein Wettbewerbsvorteil ergeben, wenn anhand von App-Eigenschaften Erklärungsbedarf identifiziert wird und die Software anhand dieser Kriterien entwickelt wird [18].
Dies könnte in den Bereichen Genre einer App, Kompatibilität auf welchen Systemen die App verwendet werden kann oder aus den Bewertungen der App eine Rolle spielen. An dieser Stelle kann beispielsweise die Sterneverteilung hilfreich sein.

1.1 Problemstellung

Durch die Identifizierung des Erklärungsbedarfs kann dieser behoben werden, wodurch die Nuzterzufriedenheit bei der Interaktion angehoben werden kann und die App mehr genutzt wird [18]. Zusätzlich wird mit der Klärung und Berücksichtigung des Erklärungsbedarfes im Vorhinein die Kundenzufriedenheit der Nutzer angehoben. Des Weiteren werden die Aspekte der App für die Nutzer transparenter gestaltet, die für die Nutzer eine Relevanz aufweisen.
Dabei ist es ein Problem für die unterschiedlichen Apps herauszukristallisieren, welche Art von Erklärungsbedarf wahrscheinlich in welcher Art von App auftreten wird, um diesen Erklärungsbedarf im Vorhinein aufzulösen.

1.2 Lösungsansatz

Zuerst werden Reviews gecrawlt und die Metadaten der jeweiligen App in einen Datensatz eingelesen. Diese Daten werden dann aufbereitet. Um eine Vorhersage zu auftretendem Erklärungsbedarf treffen zu können, nutzt man die Korrelationsanalyse.
Durch die Korrelationsanalyse wird untersucht, ob es einen Zusammenhang zwischen den Eigenschaften der App und den Arten von Erklärungsbedarf gibt.
Dies wird durch die logistische Regression noch verfeinert, um mit Hilfe mehrerer App-Eigenschaften eine Vorhersage zum Erklärungsbedarf zu treffen und diese Vorhersage zu automatisieren.

1.3 Struktur der Arbeit

Diese Arbeit ist in mehrere Kapital untergliedert. Im Kapitel 2, wird als Erstes der Begriff der Erklärbarkeit definiert. Danach werden die Analyseverfahren zur Korrelationsanalyse und zur Regression vorgestellt. Im Kapitel 3 werden Studien, Papers und andere Arbeiten vorgestellt, die thematisch zu dieser Arbeit passen. Diese untersuchen jedoch andere Fragestellungen oder es wird ein kleiner Aspekt analysiert. Anschließend erfolgt eine Abgrenzung zu den anderen Arbeiten.
Das Kapitel 4 beschreibt den Aufbau und die Implementierung der Arbeit. Dabei geht es um die Vorverarbeitung des Datensatzes, zum Aufbau der Korrelationsanalyse, sowie den Aufbau der logistischen Regression. Als letztes werden die Forschungsfragen aufgestellt. Es folgt das Kapitel 5, in dem die Ergebnisse aus der Korrelationsanalyse und der logistischen Regression vorgestellt werden. Im Kapitel 6 werden die Forschungsfragen beantwortet und die Ergebnisse aus dem Kapitel 5 interpretiert. Das Kapitel 7 zeigt die Grenzen der Arbeit auf und mögliche Fehlerquellen in der Arbeit. Im letzten Kapitel 8 werden die Ergebnisse nochmals zusammengefasst. Zudem wird

ein Ausblick auf mögliche weitere Fragestellungen gegeben.

Kapitel 2

Grundlagen

In den Grundlagen werden die Begriffe Erklärbarkeit und Arten von Erklärungsbedarf beschrieben, sowie einige Verfahren der Korrelationsanalyse. Bei der Korrelationsanalyse werden die Verfahren vorgestellt, welche in dieser Arbeit angewandt werden. Zum Schluss wird die lineare und logistische Regression beschrieben.

2.1 Erklärbarkeit

Chazette et al. [8] definiert Erklärbarkeit als ein Teil des Gebietes des Requirement Engineerings, welches zu den nicht-funktionalen Anforderungen des Requirement Engineerings zählt. Chazette et al. [8] haben herausgearbeitet, dass für Erklärbarkeit keine eindeutige Definition existiert. Um den Begriff trotzdem zu beschreiben, wurde unter Zuhilfenahme von Definitionen aus der Literatur, Erklärungen von Philosophen und Ergebnissen aus Workshops eine abstrakte Definition von Chazette et al. [8] erarbeite. Die lautet:
A system S is explainable with respect to an aspect X of S relative to an addressee A in context C if and only if there is an entity E (the explainer) who, by giving a corpus of information I (the explanation of X), enables A to understand X of S in C.
Das heißt ein System ist nur erklärbar, wenn es eine Art Erklärer gibt, der die relevante Information zu den Systemaspekt dem Adressaten zur Verfügung stellt, damit der Adressat den Aspekt im System unter diesem Kontext versteht.
Zudem entsteht Erklärungsbedarf nur, wenn jemand um eine Erklärung bittet oder indirekt als notwendig erkannt wird. Dies kann auf zwei unterschiedliche Arten erfolgen: Entweder explizites oder implizites Fragen.

2.1.1 Expliziter Erklärungsbedarf

Expliziter Erklärungsbedarf liegt vor, wenn ein Nutzer nach einer Erklärung oder Klarstellung für den Sachverhalt fragt. Dies ist häufig gut erkennbar durch Fragewörter. Als mögliche Indikatoren für expliziten Erklärungsbedarf treten Begriffe wie zum Beispiel: *warum; weshalb; wieso; könnten Sie mir das erklären; etc* auf.

2.1.2 Impliziter Erklärungsbedarf

Impliziter Erklärungsbedarf liegt vor, wenn ein Nutzer nicht direkt nach einer Klärung fragt, jedoch Unverständnis über einen Sachverhalt äußert oder für ihn ein nicht nachvollziehbarer Sachverhalt vorliegt. Als Begrifflichkeiten, die auf impliziten Erklärungsbedarf hinweisen, dienen zum Beispiel: *Es gibt keine Möglichkeit, die ich finden kann; Ich weiß nicht; Ich kann nicht finden; etc.*

2.1.3 Taxonomie

Über die Taxonomie von Droste et al. [10] lässt sich Erklärungsbedarf in andere Kategorien zuordnen. Da durch die Unterteilung in expliziten und impliziten Erklärungsbedarf keine Einteilung in unterschiedliche Systemaspekte möglich ist, wird für die Arbeit die Taxonomie von Droste et al. [10] herangezogen.

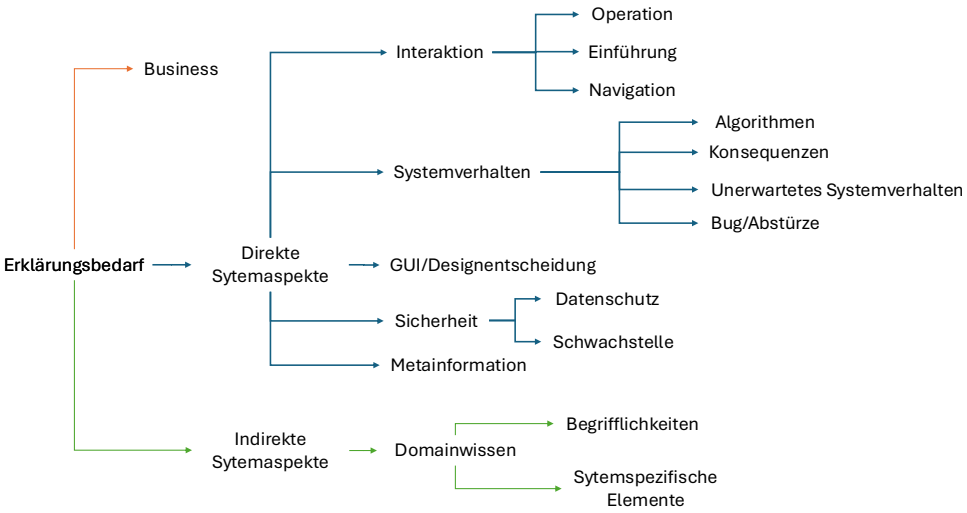

Abbildung 2.1: Taxonomie von Droste et al. [10] übersetzt

Business

Unter Business versteht man Erklärungsbedarf, der sich an die Entwickler oder das Unternehmen richtet. Zum Beispiel: *Warum hat sich der Preis der App geändert?*

Direkte Systemaspekte

Direkte Systemaspekte betreffen aufkommende Unklarheiten im Bezug auf die Software.

- **Interaktion**
  - **Operation**
    Wenn nach einer bestimmten Interaktionsmöglichkeit gefragt wird, die auch vorhanden ist. Zum Beispiel: *Wie kann ich den Ton aktivieren?*
  - **Einführung**
    Wenn eine Einführung ins System oder in bestimmte Bereiche des Systems gewünscht wird. Zum Beispiel: *Können Sie mir ein Tutorial geben, wie man eine Gruppe erstellt?*
  - **Navigation**
    Hier geht es um Unklarheiten, die bei der Interaktion zum Navigieren durchs System auftreten können. Zum Beispiel: *Wie komme ich ins Menü?*
- **Systemverhalten**
  - **Algorithmen**
    Beschreiben Unklarheiten wie ein Ergebnis zustande kommt. Zum Beispiel: Wieso schlägt mir der Algorithmus dieses Wort vor?
  - **Konsequenzen**
    Wenn eine Frage aufkommt, wie sich das System auf eine Eingabe verhält. Zum Beispiel: *Was passiert, wenn ich die Nachricht lösche?*
  - **Unerwartetes Systemverhalten**
    Unerwartetes Systemverhalten beschreibt, warum ein Ergebnis nach einer Interaktion passiert, die nicht vom Nutzer erwartet wird. Zum Beispiel: *Wieso hat das System nicht automatisch gespeichert?*
  - **Bug/Abstürze**
    Wenn ein Nutzer nach einer Erklärung für Fehlermeldungen sucht. Zum Beispiel: *Warum bekomme ich beim Starten der App eine Fehlermeldung?*

- **GUI/Designentscheidung**

  GUI/Designentscheidung bezieht sich auf die Darstellung und Gestaltung der Oberfläche der Software. Zum Beispiel: *Warum ist der Button rot?*

- **Sicherheit**

  Sicherheit beschreibt Unklarheiten zum Thema *Datenschutz* oder *Schwachstellen* vom System. Zum Beispiel: *Wie werden meine Daten geschützt?*

- **Metainformation**

  Als Metainformation gilt alles, was keiner anderen Kategorie zugeordnet werden kann oder aus mehreren Kategorien kombiniert wird. Ein Beispiel für Metainformation ist: *Wie kann ich ein Update rückgängig machen?*

Indirekte Systemaspekte

Indirekte Systemaspekte betreffen aufkommende Unklarheiten, die nicht direkt für die Nutzung des Systems notwendig sind.
Hierbei wird im *Domainenwissen*, zwischen *Begrifflichkeiten* und *Systemspezifische Elemente* unterschieden. Zum Beispiel: *Was bedeuten Metadaten?*

2.2 Korrelationsanalyse

Die Korrelationsanalyse ist ein statistisches Verfahren. Dieses Verfahren wird genutzt um herauszufinden, ob zwischen verschiedenen Merkmalen ein Zusammenhang besteht [21]. Dies meint nicht, dass die eine Variable der Korrelationsanalyse eine verrauschte Funktion der anderen Variable ist, sondern ob Ausprägungen von den Variablen statistisch häufig gemeinsam auftreten [21]. Für die Korrelationsanalyse gibt es unterschiedliche Verfahren. Diese Verfahren grenzen sich voneinander ab, je nachdem welches Skalenniveau beziehungsweise Messniveau die Variablen haben und wie diese miteinander geprüft werden. In der Tabelle (2.1) sind die Beziehungen der unterschiedlichen Skalenniveaukombinationen aufgelistet.
Die Unterteilung des Skalenniveaus für die Arbeit erfolgt in drei Bereiche: nominal, ordinal und metrisch. Skalenniveaus: Nominal, welches angibt, dass die Ausprägungen unterschieden werden können. Ordinal, welches angibt, dass die Ausprägungen sich sortieren lassen. Metrisch, welches sich nicht nur sortieren lässt, sondern auch die Abstände lassen sich zwischen den Ausprägungen berechnen.

Bei einer negativen Korrelation steigt eine Variable und die andere fällt. Bei der positiven Korrelation würden beide Variablen steigen beziehungsweise fallen. Je näher der Wert der Korrelation an Null ist, desto schwächer ist die Korrelation zwischen den Variablen.
In dieser Arbeit werden die Verfahren Cramer‘s V, Eta-Koeffizient, Rangkorrelation von Spearman und Bravais-Pearson vorgestellt und zur Analyse genutzt.

Skalenniveau	**Nominal**	**Ordinal**	**Metrisch**
Nominal	Kontingenzkoeffizient [13] Phi-Koeffizient [1] Cramer‘s V [1]		
Ordinal	Cramer‘s V [1]	Kendalls Tau [1] Spearman-Rangkorrelation [22] Gamma-Korrelation	
Metrisch	Eta-Koeffizient [12]	Kendalls Tau [1] Spearman-Rangkorrelation [22] Gamma-Korrelation	Bravais-Pearson-Korrelation [22]

Tabelle 2.1: Übersicht welche Korrelationsanalyseverfahren zu welcher Skalenniveaukombination passt.

2.2.1 Cramer‘s V

Nach Akoglu [1] ist Cramer‘s V eine Alternative zum Phi-Koeffizienten welcher nur 2x2 Tabellen miteinander vergleichen kann, wo hingegen mit Cramer‘s V größere Tabellen verglichen werden können. Die Korrelationsanalyse nach Cramer‘s V wird angewendet, wenn die Variablen nominal zu nominal oder nominal zu ordinal skaliert sind. Der Wert, der durch die Analyse berechnet wird, liegt zwischen 0 und 1. Dadurch lässt nach Akoglu [1] dieses Verfahren keine Aussage zu, ob es sich um eine negative oder positive Korrelation handelt.
Zur Berechnung der Formel (2.2) von Cramer‘s V muss zuerst das Chi-Quadrat berechnet werden, mit der Formel (2.1). In der Formel (2.1) wird für die Berechnung des Chi-Quadrates die beobachtete Häufigkeit O und die erwartete Häufigkeit E benötigt. Des Weiteren wird für die Formel (2.2) vom Cramer‘s V die Gesamtmenge der Stichprobe n benötigt, sowie die kleinere Zahl von der Spaltenanzahl oder Zeilenanzahl M.

$$x^2 = \sum_{k=1}^{n} \frac{(O_k - E_k)^2}{E_k} \tag{2.1}$$

$$V = \sqrt{\frac{x^2}{n * (M-1)}} \tag{2.2}$$

2.2.2 Eta-Koeffizient

Frank [12] postuliert, dass der Eta-Koeffizient als geeignetes Korrelationsmaß verwendet wird, wenn die unabhängige Variable nominal und die abhängige Variable metrisch skaliert sind. Das Ergebnis liegt wie bei Cramer‘s V zwischen 0 und 1. Dadurch ist auch keine Aussage über eine negative oder positive Korrelation möglich. Der Eta-Koeffizient berechnet sich wie in der Formel (2.3). Zur Berechnung ist die Gesamtzahl aller Einträge n nötig, sowie der Mittelwert aller metrischen Zahlen $\overline{y}$, die Mittelwerte der unterschiedlichen Positionen $\overline{y_0}$ $\overline{y_1}$ und die Anzahl der Positionen n_0 n_1. Als letzten Punkt wird noch die Stichprobenvarianz s_y^2 benötigt, um den Eta-Koeffizienten zu berechnen. Eine Varianz ist ein Maß, welches die Streuung einer Wahrscheinlichkeitsdichte-Funktion um ihren Schwerpunkt angibt. Ist die Varianz nicht bekannt wird die Stichprobenvarianz genommen, um die unbekannte Varianz zu schätzen.

$$\eta = \sqrt{\frac{\frac{1}{n-1} * \sum_{i=1}^{k} n_i * (\overline{y_i} - \overline{y})^2}{s_y^2}} \tag{2.3}$$

2.2.3 Spearman-Rangkorrelation

Die Spearman-Rangkorrelation gibt im Vergleich zu Cramer‘s V und dem Eta-Koeffizienten nicht nur die Stärke der Korrelation, sondern auch dessen Richtung an [2]. Das Ergebnis des Spearman-Korrelationskoeffizienten liegt zwischen -1 bis +1. Eine negative Korrelation liegt zwischen den Werten -1 bis 0. Bei einer positiven Korrelation liegen die Werte zwischen 0 bis +1. Die Spearman-Rangkorrelation wird für eine ordinal skalierte Variable mit entweder einer ordinal skalierten Variable oder einer metrisch skalierten Variable genutzt. Die Spearman-Rangkorrelation verwendet nicht die Ausgangsdaten zur Berechnung, sondern die Rangplätze der Daten. Die Formel für die Spearman-Korrelationskoeffizienten ist: (2.4)

$$r_s = 1 - \frac{6 * \sum d^2}{n * (n^2 - 1)} \tag{2.4}$$

In der Formel (2.4) gibt es den Spearman-Korrelationskoeffizienten r_s, die Anzahl der Fälle n und die Differenz d zwischen den beiden Rängen der beiden Variablen.

2.2.4 Bravais-Pearson-Korrelation

Das letztes Korrelationsanalyseverfahren, das in dieser Arbeit verwendet wird, ist das Verfahren nach Bravais-Pearson [11]. Welches wie beim Spearman-Verfahren auch ein Ergebnis zwischen -1 bis +1 aufzeigt.
Jedoch bezieht sich der Bravais-Pearson-Korrelationskoeffizient auf die Ausgangsdaten und nicht wie bei Spearman auf die Rangplätze. Das Bravais-Pearson-Verfahren wird genutzt, um zwei metrisch skalierte Variablen miteinander zu vergleichen. Die Formel für den Pearson-Korrelationskoeffizienten lautet(2.5):

$$r = \frac{\sum (x_i - \overline{x}) * (y_i - \overline{y})}{\sqrt{\sum (x_i - \overline{x})^2 * \sum \overline{x}) * (y_i - \overline{y})^2}} \tag{2.5}$$

Die Formel (2.5) besteht aus den Bravais-Pearson-Korrelationskoeffizienten r, die einzelnen Werte x_i und y_i der Daten sowie die Mittelwerte $\overline{x}$ und $\overline{y}$ aller Daten aus den Spalten.

2.3 Regression

Die Regression ist nach Bortz [7] ein statistisches Analyseverfahren zur Untersuchung der Beziehung zwischen einer abhängigen Variable und einer oder mehrerer unabhängiger Variablen.

Mithilfe der multiplen Regression lässt sich das Ergebnis der abhängigen Variable vorhersagen, unter Zuhilfenahme der unabhängigen Variablen.

2.3.1 Lineare Regression

Durch die lineare Regression lässt sich eine Zielvariable, welche stetig sein muss, statistisch untersuchen [3]. Dafür werden mindestens eine oder mehrere Variablen benötigt, welche zur Bestimmung ihrer Beziehung auf die Zielvariable untersucht werden.
Von der linearen Regression gibt es zwei Formen:
1) Die einfache lineare Regression, welche eine Zielvariable und eine Eingangsvariable beinhaltet.
2) Bender et al. [3], die multiple lineare Regression, welche eine Zielvariable und mehrere Variablen zur Zielfindung der Zielvariablen besitzt.
Die Formel für die einfache lineare Regression ist die Geradengleichung (2.7).

$$y = a + \beta x \tag{2.6}$$

Die Formel für die multiple lineare Regression lautet (2.8):

$$y = a + \beta_1 x_1 + ... + \beta_n x_n \tag{2.7}$$

Da die lineare Regression eine stetige Zielvariable zurückgibt, wird sie in dieser Arbeit nicht verwendet. Die Zielvariablen in dieser Arbeit sind nicht stetig.

2.3.2 Logistische Regression

Best und Wolf [4], um eine Funktion zu erhalten, die keine stetige Zielvariable zurück gibt, wurde die logistische Regression entwickelt. Diese ist für nicht stetige Anwendungen gedacht. Die Formel für die einfache logistische Regression ist(2.8):

$$p = \frac{exp(a + \beta x)}{1 + exp(a + \beta x)} \tag{2.8}$$

Bender et al. [3], genauso wie bei der multiplen linearen Regression wird auch die multiple logistische Regression erweitert (2.9).

$$p = \frac{exp(a + \beta_1 x_1 + ... + \beta_n x_n)}{1 + exp(a + \beta_1 x_1 + ... + \beta_n x_n)} \tag{2.9}$$

2.4 Cohens Kappa

Cohens Kappa berechnet die relative Übereinstimmung P_0. Zur Berechnung werden alle Fälle mit Übereinstimmung aufsummiert und durch die Gesamtanzahl der Fälle dividiert. Des Weiteren wird die Wahrscheinlichkeit zufälliger Übereinstimmung P_e berechnet. Diese ergibt sich aus der Wahrscheinlichkeit, mit dem der Rater 1 den ersten Fall genommen hat, multipliziert mit der Wahrscheinlichkeit das Rater 2 den ersten Fall genommen hat. Dies wird durchgeführt, bis alle Fälle addiert sind und im Anschluss aufsummiert werden. Die Formel fürs Cohens Kappa ist: 2.10

$$K = \frac{P_0 - P_e}{1 - P_e} \tag{2.10}$$

Das Ergebnis vom Cohens Kappa haben Landis und Koch [17] interpretiert und in die Tabelle 2.2 eingefügt.

Cohens Kappa	**Stärke der Übereinkunft**
<0,00	Schlecht
0,00-0,20	Leicht
0,21-0,40	Angemessen
0,41-0,60	Mäßig
0,61-0,80	Erheblich
0,81-1,00	Fast perfekt

Tabelle 2.2: Ergebnisse aus der Interpretation von Landis und Koch [17]

2.5 Python-Bibliotheken

Hier sind die Python-Bibliotheken aufgelistet, welche in der Arbeit zur Berechnung und Implementierung der Korrelationsanalysen und logistischen Regressionen genutzt werden.

pandas[1] wird zum Einlesen von Dateien genutzt. In dieser Arbeit findet es Anwendung zum Einlesen der Datensätze.

scipy[2] stellt die Algorithmen für Optimierungen, Integration, algebraische Gleichungen, Statistik und Einiges mehr bereit. Genutzt wird diese Bibliothek, um die Korrelationsverfahren und die logistischen Regressionen zu berechnen. Zudem wird es genutzt, um die logistischen Regressionen zu

[1] https://pandas.pydata.org/
[2] https://scipy.org

optimieren.

numpy[3] fügt in Python multidimensionale Arrays hinzu.

sklearn[4] bietet Methoden für das maschinelle Lernen an. Diese stehen zur Analyse von Klassifikationen, Regressionen, Clustering und weiteren Verfahren bereit. In dieser Arbeit wird das System der logistischen Regression damit implementiert, trainiert und getestet.

[3] https://numpy.org//
[4] https://scikit-learn.org/stable/

Kapitel 3

Verwandte Arbeiten

In diesem Kapitel werden verwandte Arbeiten vorgestellt, welche thematisch zu dieser Arbeit passen und Anwendung gefunden haben.

3.1 Verwandte Arbeiten

Biswas et al. [5] untersuchen in ihrem Paper den Einfluss zwischen Datenschutz, Zertifizierungen, klinische Genehmigungen, Funktionalität, Design, Nutzer-Sterne-Bewertungen und Nutzerrezensionen in mHealth-Apps (mobile health App).
Ziel der Analyse ist es herauszufinden, ob die vom Nutzer gegebene Bewertung und die Appeigenschaften über eine mehrdimensionale Perspektive beeinflusst werden. Dafür wurden die Nutzerrezensionen unter Anwendung einer Software zur Textinterpretation ausgewertet. Anschließend wurden alle Eigenschaften mithilfe von Fuzzy-Logik wieder zusammengefügt.
Im Paper von Droste et al. [10] wurde eine Umfrage zum Thema Erklärbarkeit in Apps mit 84 Teilnehmern durchgeführt.
Mit dieser Studie sollte die Frage geklärt werden, wie sich die Arten von Softwaresystemen zwischen den gefundenen Erklärungsbedarfen unterscheiden. Zudem wurde eine Taxonomie erstellt.
In der Studie von Chazette und Schneider [9] wird der Zusammenhang zwischen Erklärungsbedarf und Transparenz der Software analysiert.
Dabei überprüfen sie, ob Erklärung zum unerwarteten Verhalten von Software gewünscht ist und welche Vor- beziehungsweise Nachteile Erklärungen auf die Nutzer haben.
Bohnstedt [6] hat in seiner Bachelorarbeit die Fragestellung „Untersuchung des Einflusses von Domänenwissen auf den Erklärungsbedarf der Nutzenden von Software“ beleuchtet. Er untersuchte welcher Zusammenhang zwischen Erklärungsbedarf und Domänenwissen besteht.
In der Arbeit wurde eine Studie durchgeführt, um zu untersuchen, welche Art von Erklärungsbedarf bei unterschiedlichem Vorwissen zu Domänenwissen

auftritt. Folglich wurde untersucht, ob eine Korrelation zwischen Domänenwissen und Erklärungsbedarf existiert. Dabei wurde festgestellt, dass keine Korrelation besteht.
Jösten [14] hat in seiner Bachelorarbeit mit dem Thema „Clusteranalyse zwischen Stimmung und Erklärbarkeitsanforderungen von Nutzern“ den Zusammenhang von Erklärbarkeit, der Stimmung des Nutzers und den demografischen Daten des Nutzers untersucht.
Hierfür hat Jösten einen Datensatz mithilfe des Korrelationsverfahren Cramer‘s V untersucht, um mit den Ergebnissen des Cramer‘s V-Verfahren eine Clusteranalyse durchzuführen. Dabei stellt sich heraus, dass es nur einen schwachen Zusammenhang zwischen Erklärungsbedarf und Stimmung des Nutzers gibt, sowie einen schwachen Zusammenhang zwischen Erklärungsbedarf und den demografischen Daten des Nutzers.
In der Bachelorarbeit von Kurtz [16] „Entwicklung einer Software zur Extrahierung und Analyse von Reviews aus App Stores“ entwickelt Kurtz [16] ein Tool Namens *feelio*, welches Reviews aus dem Apple App Store und dem Playstore crawlt. Den Reviews werden dann verschiedene Merkmale zugeschrieben, z.B.: eine Sentimentpolarität, eine Emotion gemäß den Ekman’schen Basisemotionen, eine sprachliche Qualität, ein potenzieller Erklärungsbedarf und spezifische Charakteristika. Mit diesem Tool will Kurtz [16] das Problem der manuellen Auswertung von Reviews, und so App-Analysten helfen.
Im Paper von Unterbusch et al. [23] wird ein Datensatz von 1730 App-Reviews manuell codiert und daraus eine Taxonomie über Erklärungsbedürfnisse abgeleitet. Zudem werden verschiedene Ansätze von automatisierter Erkennung von Erklärungsbedarf in Reviews beschrieben und ausprobiert.
Sadeghi et al. [19] haben in ihrer Arbeit eine Taxonomie zum Thema Erklärungsbedarf erstellt. Sie konzentrieren sich dabei auf die Benutzerinteraktion bei der Erstellung der Taxonomie.
Stange et al. [20] haben in ihrer Arbeit den Erklärungsbedarf, der bei Robotern anfällt, untersucht, um auf das Ziel hinzuarbeiten einen autonomen, lebendigen Roboter zu konstruieren. Dafür soll das Verhalten des Roboters besser erklärbar sein.

3.2 Abgrenzung zu verwandten Arbeiten

Zum Thema Erklärbarkeit im Software Engineering gibt es viele Arbeiten, Studien und Papers sowie zahlreiche verwandte Arbeiten, da in diesem Forschungsfeld viel geforscht wird. Das Augenmerk in dieser Arbeit liegt auf anderen Fragestellungen, die noch nicht Forschungsgegenstand sind. Im Paper von Droste et al. [10] wird eine kleine Studie (mit 84 Teilnehmern) zum Thema Erklärungsbedarf durchgeführt, woraus eine Taxonomie entwickelt wird. Auch in den Papers von Sadeghi et al. [19] und Unterbusch et al.

[23] wird eine Taxonomie zum Thema Erklärungsbedarf erstellt. In den Bachelorarbeiten von Bohnstedt [6] und Jösten [14] wird zu Zusammenhängen zwischen verschiedenen Nutzeraspekten und Erklärungsbedarfen geforscht. Chazette und Schneider [9] prüfen in ihrer Studie, ob ein Zusammenhang zwischen Erklärungsbedarf und Transparenz besteht und ob eine Erklärung gewünscht ist. In dem Paper von Biswas et al. [5] werden nur mHealth-Apps auf eine mehrdimensionale Perspektive von Nutzereigenschaften und Appeigenschaften untersucht.
In dieser Arbeit sollen hingegen die App-Eigenschaften auf Korrelationen zur Erklärbarkeit, sowie auf deren unterschiedlichen Kategorien untersucht werden.
Ausgewählt wurden Apps aus den unterschiedlichsten Bereichen mit den verschiedensten App-Eigenschaften. Diese Apps wurden anhand eines Datensatzes von 4.995 Reviews analysiert.
Die untersuchten Apps stammen aus den zwei größten Applikationen für Apps. Dafür wurde der Playstore und der App Store genutzt, da diese die weltweit größten Plattformen für Apps sind. Andere wie zum Beispiel von Amazon (Amazon Appstore) oder Samsung (Samsung Galaxy Store) sind anhand der prozentualen Nutzung von Nutzern nicht relevant.
Anhand der Korrelationsanalyse soll dann versucht werden mit der logistischen Regression eine Vorhersage über den Erklärungsbedarf zu ermitteln. Dies ist ein Unterschied zu den Arbeiten von Bohnstedt [6] und Jösten [14] die Clusteranalysen durchführen um Gruppen zu finden.

Kapitel 4

Aufbau und Implementierung

In diesem Kapitel wird der Aufbau des Goldstandard-Datensatzes sowie die Implementierung der Korrelationsanalyse und der logistischen Regression beschrieben. Am Ende des Kapitels werden die Forschungsfragen und die daraus resultierenden Nullhypothesen vorgestellt. Der Aufbau der Analyse ist in der Abbildung 4.1 dargestellt. Zuerst wird der Goldstandard-Datensatz von Kupczyk [15] auf Fehler überprüft und bereinigt. Zugleich werden in einem zweiten App-Datensatz die Metadaten der Apps gecrawlt. Mit diesen beiden Datensätzen führt man die Korrelationsverfahren durch. Die logistischen Regressionen werden nach den Korrelationsanalysen durchgeführt. Für die Validierung der logistischen Regressionen werden neue Validierung-Datensätze angelegt und die logistischen Regressionen auf die neuen Validierung-Datensätze getestet.

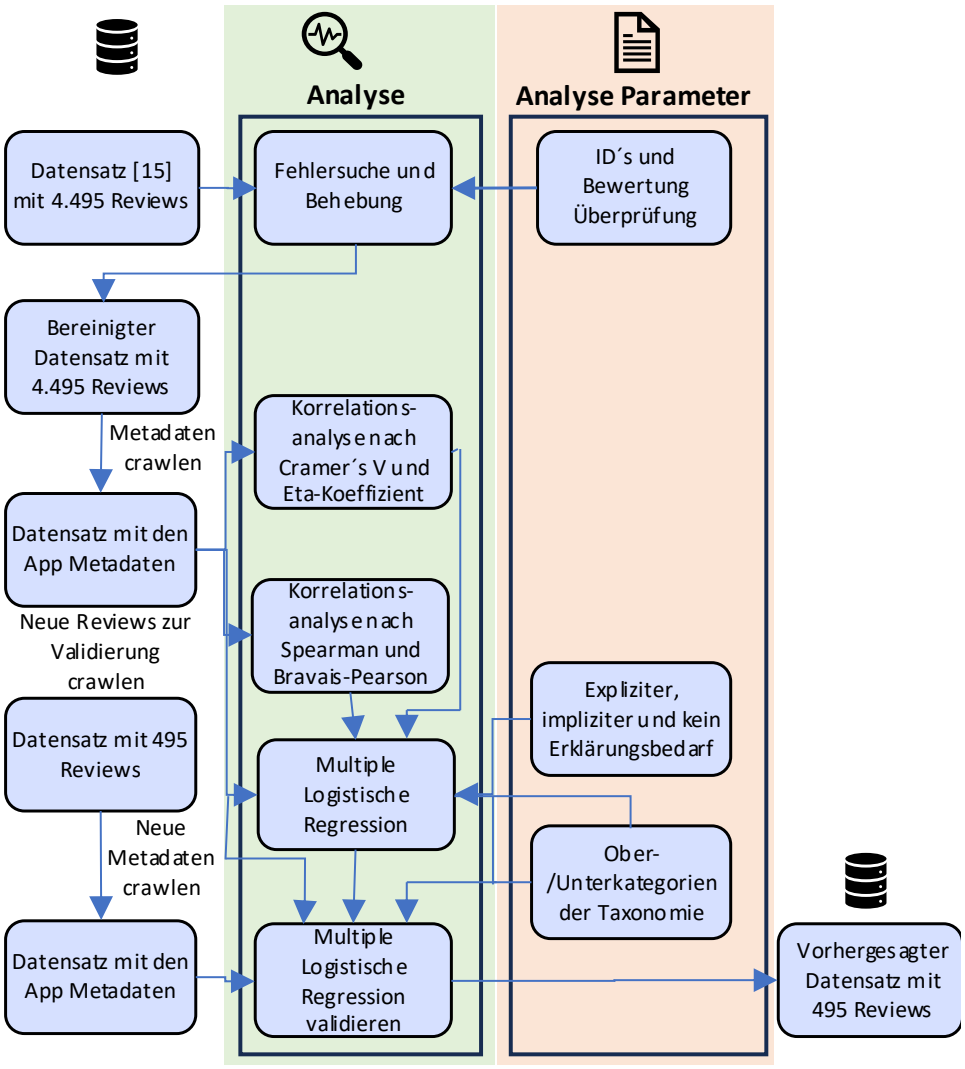

Abbildung 4.1: Aufbau der Analyseschritte

4.1 Forschungsziele und -fragen

In dieser Arbeit wird untersucht, inwiefern es einen Zusammenhang zwischen den Eigenschaften einer App und den Arten von Erklärungsbedarfen gibt. Hierfür wird der Goldstandard-Datensatz, in dem die Reviews in Arten von Erklärungsbedarfen klassifiziert werden, und der App-Datensatz mit den App-Eigenschaften analysiert. Falls keine Zusammenhänge gefunden werden, gelten die Nullhypothesen.

RQ1: Gibt es eine Korrelation zwischen dem vom Unternehmen festgelegten App-Eigenschaften und den Arten von Erklärungsbedarfen?

Diese Fragestellung soll klären, ob die objektiv festgelegten App-Eigenschaften vom Unternehmen einen Einfluss auf die Erklärungsbedarfe ausüben. Falls es einen starken Einfluss gibt, könnte damit der Erklärungsbedarf im Vorfeld identifiziert und behoben werden.

RQ2: Gibt es eine Korrelation zwischen dem vom Nutzerfeedback entnommenen App-Eigenschaften und den Arten von Erklärungsbedarfen?

Diese Untersuchung verfolgt das Ziel, ob sich die Erklärungsbedarfe über den Lebenszyklus der App verändert.

RQ3: Kann man mit einer Kombination von App-Eigenschaften eine Vorhersage über einen Erklärungsbedarf treffen?

Diese Forschungsfrage soll klären, ob über eine logistische Regression mithilfe von App-Eigenschaften der Erklärungsbedarf vorhergesagt werden kann. Wenn diese Möglichkeit besteht, lässt sich mithilfe der App-Eigenschaften schon im vorhinein Erklärungsbedarf identifizieren.

4.2 Hypothesenübersicht

Um die Forschungsfragen besser beantworten zu können, wird für jede App-Eigenschaft, die mithilfe des Crawlers in die Datenbank geladen wurde, eine Nullhypothese aufgestellt. Diese Nullhypothesen werden anschließend den Forschungsfragen zugeordnet. Für die Hypothesen wird die Signifikanz auf das übliche Signifikanzlevel $\alpha = 0,05$ gesetzt und mithilfe der Bonferroni Korrektur geprüft [24]. Die Formel für die Bonferroni Korrektur ist $\alpha_{Bon} = \frac{\alpha}{n}$ [24]. Zum Beispiel wären drei untergeordnete Hypothesen mit $\alpha = 0,05$ ungefähr $0,0167 = \frac{0,05}{3}$. Sollte der Signifikanzwert unterhalb des

Signifikanzlevel liegen, wird das Ergebnis verworfen.

Nr.	Hypothese	RQ
$H1_0$	**Es besteht kein Zusammenhang zwischen dem (Ja/Nein) Erklärungsbedarf und der Kategorie der untersuchten App.**	**RQ1**
$H2_0$	**Es besteht kein Zusammenhang zwischen den Arten von Erklärungsbedarf und der Kategorie der untersuchten App.**	**RQ1**
$H2.1_0$	Es besteht kein Zusammenhang zwischen der expliziten/impliziten Erklärungsbedarf und der Kategorie der untersuchten App.	RQ1
$H2.2_0$	Es besteht kein Zusammenhang zwischen der Oberkategorie des Erklärungsbedarfs und der Kategorie der untersuchten App.	RQ1
$H2.3_0$	Es besteht kein Zusammenhang zwischen der Unterkategorie des Erklärungsbedarfs und der Kategorie der untersuchten App.	RQ1
$H3_0$	**Es besteht kein Zusammenhang zwischen dem (Ja/Nein) Erklärungsbedarf dem Kaufpreis der untersuchten App.**	**RQ1**
$H4_0$	**Es besteht kein Zusammenhang zwischen den Arten von Erklärungsbedarf und (dem Kaufpreis kostenpflichtig oder kostenlos) der untersuchten App.**	**RQ1**
$H4.1_0$	Es besteht kein Zusammenhang zwischen dem expliziten/impliziten Erklärungsbedarf und dem Kaufpreis der untersuchten App.	RQ1
$H4.2_0$	Es besteht kein Zusammenhang zwischen der Oberkategorie des Erklärungsbedarfs und dem Kaufpreis der untersuchten App.	RQ1
$H4.3_0$	Es besteht kein Zusammenhang zwischen der Unterkategorie des Erklärungsbedarfs und dem Kaufpreis der untersuchten App.	RQ1
$H5_0$	**Es besteht kein Zusammenhang zwischen dem (Ja/Nein) Erklärungsbedarf den zu zahlenden Kaufpreis der untersuchten App.**	**RQ1**
$H6_0$	**Es besteht kein Zusammenhang zwischen den Arten von Erklärungsbedarf und den zu zahlenden Kaufpreis (Höhe des Kaufpreises) der untersuchten App.**	**RQ1**
$H6.1_0$	Es besteht kein Zusammenhang zwischen dem expliziten/impliziten Erklärungsbedarf und dem zu zahlenden Kaufpreis der untersuchten App.	RQ1
$H6.2_0$	Es besteht kein Zusammenhang zwischen der Oberkategorie des Erklärungsbedarfs und dem zu zahlenden Kaufpreis der untersuchten App.	RQ1
$H6.3_0$	Es besteht kein Zusammenhang zwischen der Unterkategorie des Erklärungsbedarfs und den zu zahlenden Kaufpreis der untersuchten App.	RQ1
$H7_0$	**Es besteht kein Zusammenhang zwischen dem (Ja/Nein) Erklärungsbedarf dem Ingame Preis der untersuchten App.**	**RQ1**

Tabelle 4.1: 1. Hypothesenübersicht für die Übersicht der durchs Unternehmen eingesetzten App-Eigenschaften

Nr.	Hypothese	RQ
$H8_0$	**Es besteht kein Zusammenhang zwischen den Arten von Erklärungsbedarf und dem Ingame Preis der untersuchten App.**	**RQ1**
$H8.1_0$	Es besteht kein Zusammenhang zwischen dem expliziten/impliziten Erklärungsbedarf und dem Ingame Preis der untersuchten App.	RQ1
$H8.2_0$	Es besteht kein Zusammenhang zwischen der Oberkategorie des Erklärungsbedarfs und dem Ingame Preis der untersuchten App.	RQ1
$H8.3_0$	Es besteht kein Zusammenhang zwischen der Unterkategorie des Erklärungsbedarfs und dem Ingame Preis der untersuchten App.	RQ1
$H9_0$	**Es besteht kein Zusammenhang zwischen dem (Ja/Nein) Erklärungsbedarf und der aktuellen Version der untersuchten App.**	**RQ1**
$H10_0$	**Es besteht kein Zusammenhang zwischen den Arten von Erklärungsbedarf und der aktuellen Version der untersuchten App.**	**RQ1**
$H10.1_0$	Es besteht kein Zusammenhang zwischen dem expliziten/impliziten Erklärungsbedarf und der aktuellen Version der untersuchten App.	RQ1
$H10.2_0$	Es besteht kein Zusammenhang zwischen der Oberkategorie des Erklärungsbedarfs und der aktuellen Version der untersuchten App.	RQ1
$H10.3_0$	Es besteht kein Zusammenhang zwischen der Unterkategorie des Erklärungsbedarfs und der aktuellen Version der untersuchten App.	RQ1
$H11_0$	**Es besteht kein Zusammenhang zwischen dem (Ja/Nein) Erklärungsbedarf und dem Mindestalter der untersuchten App.**	**RQ1**
$H12_0$	**Es besteht kein Zusammenhang zwischen den Arten von Erklärungsbedarf und dem Mindestalter der untersuchten App.**	**RQ1**
$H12.1_0$	Es besteht kein Zusammenhang zwischen dem expliziten/impliziten Erklärungsbedarf und dem Mindestalter der untersuchten App.	RQ1
$H12.2_0$	Es besteht kein Zusammenhang zwischen der Oberkategorie des Erklärungsbedarfs und dem Mindestalter der untersuchten App.	RQ1
$H12.3_0$	Es besteht kein Zusammenhang zwischen der Unterkategorie des Erklärungsbedarfs und dem Mindestalter der untersuchten App.	RQ1
$H19_0$	**Es besteht kein Zusammenhang zwischen dem durchschnittlichen Erklärungsbedarf und der Kategorie der untersuchten App.**	**RQ1**
$H20_0$	**Es besteht kein Zusammenhang zwischen dem durchschnittlichen Erklärungsbedarf und dem Kaufpreis der untersuchten App.**	**RQ1**
$H21_0$	**Es besteht kein Zusammenhang zwischen dem durchschnittlichen Erklärungsbedarf und den zu zahlenden Kaufpreis der untersuchten App.**	**RQ1**
$H22_0$	**Es besteht kein Zusammenhang zwischen dem durchschnittlichen Erklärungsbedarf und dem Ingame Preis der untersuchten App.**	**RQ1**
$H23_0$	**Es besteht kein Zusammenhang zwischen dem durchschnittlichen Erklärungsbedarf und der aktuellen Version der untersuchten App.**	**RQ1**
$H24_0$	**Es besteht kein Zusammenhang zwischen dem durchschnittlichen Erklärungsbedarf und dem Mindestalter der untersuchten App.**	**RQ1**

Tabelle 4.2: 2. Hypothesenübersicht für die Übersicht der durchs Unternehmen eingesetzten App-Eigenschaften

Nr.	Hypothese	RQ
H13$_0$	**Es besteht kein Zusammenhang zwischen dem (Ja/Nein) Erklärungsbedarf und der Sterne Bewertung der untersuchten App.**	**RQ2**
H14$_0$	**Es besteht kein Zusammenhang zwischen den Arten von Erklärungsbedarf und der Sterne Bewertung der untersuchten App.**	**RQ2**
H14.1$_0$	Es besteht kein Zusammenhang zwischen dem expliziten/impliziten Erklärungsbedarf und der Sterne Bewertung der untersuchten App.	RQ2
H14.2$_0$	Es besteht kein Zusammenhang zwischen der Oberkategorie des Erklärungsbedarfs und der Sterne Bewertung der untersuchten App.	RQ2
H14.3$_0$	Es besteht kein Zusammenhang zwischen der Unterkategorie des Erklärungsbedarfs und der Sterne Bewertung der untersuchten App.	RQ2
H15$_0$	**Es besteht kein Zusammenhang zwischen dem (Ja/Nein) Erklärungsbedarf und der Anzahl an Reviews der untersuchten App.**	**RQ2**
H16$_0$	**Es besteht kein Zusammenhang zwischen den Arten von Erklärungsbedarf und der Anzahl an Reviews der untersuchten App.**	**RQ2**
H16.1$_0$	Es besteht kein Zusammenhang zwischen dem expliziten/impliziten Erklärungsbedarf und der Anzahl an Reviews der untersuchten App.	RQ2
H16.2$_0$	Es besteht kein Zusammenhang zwischen der Oberkategorie des Erklärungsbedarfs und der Anzahl an Reviews der untersuchten App.	RQ2
H16.3$_0$	Es besteht kein Zusammenhang zwischen der Unterkategorie des Erklärungsbedarfs und der Anzahl an Reviews der untersuchten App.	RQ2
H17$_0$	**Es besteht kein Zusammenhang zwischen dem (Ja/Nein) Erklärungsbedarf und der Anzahl an Downloads der untersuchten App.**	**RQ2**
H18$_0$	**Es besteht kein Zusammenhang zwischen den Arten von Erklärungsbedarf und der Anzahl an Downloads der untersuchten App.**	**RQ2**
H18.1$_0$	Es besteht kein Zusammenhang zwischen dem expliziten/impliziten Erklärungsbedarf und der Anzahl an Downloads der untersuchten App.	RQ2
H18.2$_0$	Es besteht kein Zusammenhang zwischen der Oberkategorie des Erklärungsbedarfs und der Anzahl an Downloads der untersuchten App.	RQ2
H18.3$_0$	Es besteht kein Zusammenhang zwischen der Unterkategorie des Erklärungsbedarfs und der Anzahl an Downloads der untersuchten App.	RQ2
H25$_0$	**Es besteht kein Zusammenhang zwischen dem durchschnittlichen Erklärungsbedarf und der Sterne Bewertung der untersuchten App.**	**RQ2**
H26$_0$	**Es besteht kein Zusammenhang zwischen dem durchschnittlichen Erklärungsbedarf und der Anzahl an Reviews der untersuchten App.**	**RQ2**
H27$_0$	**Es besteht kein Zusammenhang zwischen dem durchschnittlichen Erklärungsbedarf und der Anzahl an Downloads der untersuchten App.**	**RQ2**

Tabelle 4.3: 1. Hypothesenübersicht für die Übersicht der durchs Nutzerfeedback erzeugten App-Eigenschaften

4.3 Datensatz

Der in dieser Arbeit genutzte Goldstandard-Datensatz stammt aus der Masterarbeit von Kupczyk [15], welcher mithilfe des Tools *feelio* von Kurtz [16] erstellt wird. Dieses Tool wird von Kurtz [16] in seiner Bachelorarbeit entwickelt. Der vorliegende Goldstandard-Datensatz besteht aus 4495 Reviews, welche bereits von drei Ratern in der Arbeit von Kupczyk [15] in die Kategorien impliziter, expliziter und kein Erklärungsbedarf unterteilt sind, sowie eine Unterteilung in die Taxonomie aus der Abbildung 2.1 aus dem Paper von Droste [10] zugeordnet werden. Die Aufteilung der Reviews erfolgt zu jeweils einem Drittel impliziter Erklärungsbedarf, expliziter Erklärungsbedarf und kein Erklärungsbedarf.

4.3.1 Datenvorbereitung

Um den Goldstandard-Datensatz für diese Arbeit nutzen zu können, müssen die Daten aufbereitet werden.
Dafür werden zuerst die Metadaten der Apps gecrawlt und in den App-Datensatz geschrieben. Die ausgewählten Apps stammen aus dem App Store und dem Playstore. Die Metadaten der Apps werden mithilfe einer selbst geschriebenen Software gecrawlt und in dem App-Datensatz abgespeichert. Die Metadaten, die vom Crawler in den App-Datensatz geladen werden, sind: ID, Name, Description Categories, Primary Category, Age Rating, Rating, Reviews, App Price, Ingame Price, Version, Installs und Meta. Als Inspiration für die selbst entwickelte Software dient eine Software von Kurtz [16].
Bei der Anpassung kommt es zu unterschiedlichen Problemen. Ein Problem ist, dass im App Store die Metadaten zum Ingame Preis und zu den App Downloads vom Crawler nicht mit übergeben werden. Ein weiteres Problem ist, dass im Playstore sich teilweise die Namen der Apps ändern oder Apps von kostenpflichtig zu kostenlos wechseln. Als Lösungsansatz für diese Probleme können unter Zuhilfenahme der Wayback[1] Maschine die Apps gefunden und die fehlenden Metadaten händisch eingetragen werden.
Damit der vorhandene Goldstandard-Datensatz der Reviews genutzt werden kann, muss dieser Goldstandard-Datensatz ebenfalls aufbereitet werden. Grund dafür sind enthaltene Fehler im vorhandenen Goldstandard-Datensatz in den Punkten: ID‘s, Sterne-Bewertungen und den Zuordnungen zur Taxonomie. Diese Fehler sind vermutlich bei der Konvertierung aus der csv-Datei in eine Excel-Datei entstanden, beziehungsweise einer fehlerhaften Zuweisung der ID‘s bei der Erstellung des Goldstandard-Datensatzes. Bei der Konvertierung des Goldstandard-Datensatzes sind Emoji‘s und Umlaute (Ä, Ü, Ö) in ihren Programmcode, in den Reviewtexten konvertiert. Das muss korrigiert werden. Dafür wird der Goldstandard-Datensatz anhand der

[1] Internet Archive: https://web.archive.org

Reviews, des Ratings und den Titel mit den Reviews aus dem Rohdatensatz verglichen und bei dem Mapping der Daten, die ID in den Goldstandard-Datensatz überschrieben. Um weitere Fehler zu finden, wird als nächster Schritt das Rating aus den Vergleichen ausgelesen. Es werden nur die Excel-Daten genutzt, die vorher keine Übereinstimmung aufweisen. Diese werden neu verglichen und die ID sowie das Rating der Review beim Mapping in den Goldstandard-Datensatz übertragen. Für die wenigen Daten, für die keine Übereinstimmung zu finden ist, gilt es, diese händisch zu überprüfen und hinzuzufügen.
Nachdem der Goldstandard-Datensatz von Fehlern befreit ist, kann die Auflistung der Taxonomie in ihre Einzelteile getrennt erfolgen. Diese findet Anwendung, da mehrere Kategorien der Taxonomie auf einer Review zutreffen können. Des Weiteren werden zu den Reviews im Goldstandard-Datensatz die Punkte *App Store* (Play Store oder Appstore), sowie die *Primär-Kategorie* hinzugefügt.
Anhand der ermittelten Metadaten werden die Nullhypothesen aufgestellt.

4.4 Analyseaufbau

In diesem Unterkapitel wird der Aufbau der in dieser Arbeit genutzten Korrelationsanalysen und der logistischen Regressionen beschrieben.

4.4.1 Korrelationsanalyse des Datensatzes

Nach der Aufarbeitung des Goldstandard-Datensatz werden anhand der Nullhypothesen die Korrelationsanalysen implementiert. Für die Korrelationsanalysen werden die Verfahren Cramer‘s V [1] und Eta-Koeffizient [12], sowie die Verfahren der Spearman-Rangkorrelation [22] und die Bravais-Pearson-Korrelation [22] implementiert.
Um herauszufinden welches Korrelationsanalyseverfahren genutzt werden soll, werden die Variablen, aus Abbildung 4.4 nach ihrem Skalenniveau sortiert. Aufgrund dieser Skalierung wird das Korrelationsanalyseverfahren ausgewählt (wie in Abbildung (2.1)).

Variable	Name	Skalierung	Wertebereich	Beschreibung
E_{Noetig}	Erklärungsbedarf nötig	Nominal	$\{E_{noetigJa};E_{noetigNein}\}$	Ja, Nein
E_{Art}	Art des Erklärungsbedarfs	Nominal	$\{E_{Art1};E_{Art2}\}$	Implizit, explizit
$E_{Kategorie}$	Kategorie des Erklärungsbedarf	Nominal	$\{E_{Kategorie}\}$	Oberkategorie, Unterkategorie
$E_{Durchschnitt}$	Durchschittliche Anzahl an Kategorie	Metrisch	$n \in \mathbb{R}$	Durchschittliche Anzahl an Oberkategorie, Unterkategorie
$D_{Kategorie}$	Kategorie	Nominal	$\{D_{Kategorie}\}$	Primär-Kategorie der App
$D_{Kaufpreis(Ja/Nein)}$	Kaufpreis (kostenlos/kostenpflichtig)	Nominal	$\{D_{Ja};D_{Nein}\}$	Ja, Nein
$D_{Kaufpreis}$	Kaufpreis	Metrisch	$n \in \mathbb{R}$	Kosten der App
P_{Ingame}	Ingame Preis (Ja/Nein)	Nominal	$\{P_{Ja};P_{Nein}\}$	Preisspanne der möglichen Ingame Käufe
$D_{Version}$	Versionsstufe	Ordinal	$n \in \mathbb{R}$	Aktuelle Versionsstufe
D_{Alter}	Mindestalter	Ordinal	4,9,12,17	Nach IARC
$D_{Bewertung}$	Sterne Bewertung	Ordinal	1,00...5,00	Durchschnittliche Sterne Bewertung
D_{Review}	Anzahl Reviews	Metrisch	$n \in \mathbb{N}_0$	
$P_{Download}$	Anzahl Downloads	Metrisch	$n \in \mathbb{N}_0$	

Tabelle 4.4: Variablenübersicht

Die Auswahl für die ersten Nullhypothesen fällt auf Cramer‘s V und Eta-Koeffizient, da die zu testenden Variablen E_{Noetig}, $E_{Kategorie}$ und E_{Art} ein nominales Skalenniveau aufweisen. Die Variablen, die mit Cramer‘s V analysiert werden, sind in Abbildung 4.6 dargestellt. Die Variablen, die beim Eta-Koeffizienten analysiert werden, sind in der Abbildung 4.5 aufgeführt. Dabei wird Cramer‘s V genutzt, um eine nominale Variable mit einer anderen nominalen Variable, beziehungsweise nominale Variable mit ordinalen Variablen zu vergleichen. Mit dem Eta Koeffizient werden nominale mit metrischen Variablen verglichen.

abhängige Variable	unabhängige Variable	Korrelationsverfahren
E_{Noetig}	$D_{Kaufpreis}$	Eta-Koeffizient
$E_{Kategorie}$	$D_{Kaufpreis}$	Eta-Koeffizient
E_{Art}	$D_{Kaufpreis}$	Eta-Koeffizient
E_{Noetig}	D_{Review}	Eta-Koeffizient
$E_{Kategorie}$	D_{Review}	Eta-Koeffizient
E_{Art}	D_{Review}	Eta-Koeffizient
E_{Noetig}	$P_{Download}$	Eta-Koeffizient
$E_{Kategorie}$	$P_{Download}$	Eta-Koeffizient
E_{Art}	$P_{Download}$	Eta-Koeffizient

Tabelle 4.5: Übersicht der Variablen zu den Eta-Koeffizient

abhängige Variable	unabhängige Variable	Korrelationsverfahren
E_{Noetig}	$D_{Kategorie}$	Cramer‘s V
$E_{Kategorie}$	$D_{Kategorie}$	Cramer‘s V
E_{Art}	$D_{Kategorie}$	Cramer‘s V
E_{Noetig}	$D_{Kaufpreis(Ja/Nein)}$	Cramer‘s V
$E_{Kategorie}$	$D_{Kaufpreis(Ja/Nein)}$	Cramer‘s V
E_{Art}	$D_{Kaufpreis(Ja/Nein)}$	Cramer‘s V
E_{Noetig}	P_{Ingame}	Cramer‘s V
$E_{Kategorie}$	P_{Ingame}	Cramer‘s V
E_{Art}	P_{Ingame}	Cramer‘s V
E_{Noetig}	$D_{Version}$	Cramer‘s V
$E_{Kategorie}$	$D_{Version}$	Cramer‘s V
E_{Art}	$D_{Version}$	Cramer‘s V
E_{Noetig}	D_{Alter}	Cramer‘s V
$E_{Kategorie}$	D_{Alter}	Cramer‘s V
E_{Art}	D_{Alter}	Cramer‘s V
E_{Noetig}	$D_{Bewertung}$	Cramer‘s V
$E_{Kategorie}$	$D_{Bewertung}$	Cramer‘s V
E_{Art}	$D_{Bewertung}$	Cramer‘s V

Tabelle 4.6: Übersicht der Variablen zu den Cramer‘s V

Diese beiden Verfahren werden als Erstes in Python programmiert, um die Hypothesen, welche von $H1.1_0$ bis $H18.4_0$ (siehe die Tabellen (4.1) und (4.2)) reichen auf Korrelationen zu testen. Dadurch, dass die abhängigen Variablen der Hypothesen ein nominales Skalenniveau aufweisen, werden die Spearman-Rangkorrelation und die Bravais-Pearson-Korrelation zu einem späteren Zeitpunkt implementiert.
Bei der Implementierung der Verfahren werden über Schleifen jedem Review einer App die zugehörigen Metadaten zugeschrieben und in einer Liste zwischengespeichert.
Danach erfolgt die Berechnung der Korrelation nach Cramer‘s V. Dafür wird das Chi-Quadrat berechnet, bevor die Formel für Cramer‘s V angewandt wird, welche als Rückgabewert den Korrelationskoeffizienten aufzeigt. Der p-Wert wird durch das Chi-Quadrat berechnet und ausgegeben.
Für den Eta-Koeffizienten wird eine eigene Funktion geschrieben, die den Eta-Koeffizienten als Rückgabewert enthält. Auch beim Eta-Koeffizienten wird zusätzlich das Chi-Quadrat berechnet, um den p-Wert zurückzugeben. Mit diesen beiden Verfahren werden alle unabhängigen Variablen (4.4) auf Korrelationen geprüft.

Nachdem die Korrelationsanalysen erstellt wurden, kam die Frage auf, ob es eine Korrelation zwischen dem Durchschnitt der Taxonomie per

App oder eine Korrelation zwischen dem Durchschnitt der Taxonomie per Primär-Kategorie und den unabhängigen Variablen aus der Tabelle (4.4) gibt. Die Primär-Kategorie gibt die Kategorie an, unter welcher die App in den App Store‘s kategorisiert wird. Diese Frage wird in den Hypothesentabellen (4.2) und (4.3) in den Nullhypothesen $H19.1_0$ bis $H19.9_0$ geprüft. Da das Skalenniveau der abhängigen Variable $E_{Durchschnitt}$ metrisch ist, werden andere Korrelationsverfahren angewendet. Die dafür geeigneten Verfahren für die Korrelationsanalysen in dieser Arbeit bei metrisch skalierten Variablen sind die Spearman-Rangkorrelation und die Bravais-Pearson-Korrelation.
Hierfür werden die Spearman-Rangkorrelation und die Bravais-Person-Korrelation implementiert. Die Spearman-Rangkorrelation wird genutzt, um eine Korrelation zwischen einer abhängigen metrischen Variable und einer unabhängigen ordinalen oder nominalen Variable zu überprüfen. Die hier geprüften Variablen sind in der Tabelle 4.7 aufgelistet.

abhängige Variable	**unabhängige Variable**	**Korrelationsverfahren**
$E_{Durchschnitt}$	$D_{Kategorie}$	Spearman-Rangkorrelation
$E_{Durchschnitt}$	$D_{Kaufpreis(Ja/Nein)}$	Spearman-Rangkorrelation
$E_{Durchschnitt}$	P_{Ingame}	Spearman-Rangkorrelation
$E_{Durchschnitt}$	$D_{Version}$	Spearman-Rangkorrelation
$E_{Durchschnitt}$	D_{Alter}	Spearman-Rangkorrelation
$E_{Durchschnitt}$	$D_{Bewertung}$	Spearman-Rangkorrelation

Tabelle 4.7: Übersicht der Variablen zu den Spearman-Rangkorrelation

Um auf Korrelationen bei einer metrisch abhängigen Variable und einer anderen metrisch unabhängigen Variable zu prüfen (in Abbildung 4.8), wird das Bravais-Pearson Verfahren angewendet und implementiert.

abhängige Variable	**unabhängige Variable**	**Korrelationsverfahren**
$E_{Durchschnitt}$	$D_{Kaufpreis}$	Bravais-Pearson-Korrelation
$E_{Durchschnitt}$	D_{Review}	Bravais-Pearson-Korrelation
$E_{Durchschnitt}$	$P_{Download}$	Bravais-Pearson-Korrelation

Tabelle 4.8: Übersicht der Variablen zu den Bravais-Pearson-Korrelation

Zur Implementierung der beiden Verfahren wird die gleiche Schleife, wie bei Cramer‘s V und dem Eta-Koeffizienten angewendet. Danach wird mithilfe des Paketes „from scipy.stats import spearmanr“ die Berechnung für die Spearman-Rangkorrelation und den p-Wert ausgeführt.
Wie bei der Implementierung der Spearman-Rangkorrelation wird auch bei der Bravais-Pearson-Korrelation das Paket „from scipy.stats import pearsonr“ für die Berechnung angewandt.

Die Ergebnisse der Berechnungen werden in eine Tabelle eingetragen und im Kapitel 5 beschrieben.

4.4.2 Logistische Regression des Datensatzes

Nach der abgeschlossenen Korrelationsanalyse werden mehrere multiple logistische Regressionen durchgeführt. Damit soll überprüft werden, ob mit den App-Eigenschaften eine Vorhersage über den Erklärungsbedarf möglich ist.
Für alle Verfahren der logistischen Regressionen in dieser Arbeit wird der Goldstandard-Datensatz in zwei Teile aufgeteilt. Der erste Teil des Goldstandard-Datensatzes dient zum Trainieren der logistischen Regression mit 70% der Daten (3147 Reviews) als Trainingsdaten. Der zweite Teil des Goldstandard-Datensatzes mit 30% (1348 Reviews) als Testdaten wird zum Testen der logistischen Regression genutzt. Diese Aufteilung soll dazu dienen, dass bei der Vorhersage alle Fälle der Ober- und Unterkategorien getestet werden. Dies ist notwendig, da einige der Unterkategorien nur in geringer Häufigkeit vorkommen, wie z.B. die Unterkategorie „Datenschutz“.
Das Ziel der logistischen Regressionen ist die Vorhersage darüber zutreffen, ob eine Review impliziten, expliziten oder keinen Erklärungsbedarf aufweist, beziehungsweise, ob überhaupt Erklärungsbedarf vorliegt. Nach der Vorhersage, ob ein Erklärungsbedarf vorliegt, ist ein weiteres Ziel die Taxonomie in Oberkategorien und in Unterkategorien vorherzusagen. Die beiden Datensätze zu den Metadaten aus den App Store‘s weisen für die logistischen Regressionen ein Problem auf, da diese nicht die gleichen Metadaten enthalten. Das Problem entsteht, da der Crawler des Apple App Stores nicht alle Metadaten überträgt. Aufgrund dessen sind für die logistischen Regressionen nicht alle Metadaten nutzbar, die vom Play Store durch den Crawler übertragen werden. Die genutzten Metainformationen der App sind die Anzahl an Reviews, das Mindestalter, die Versionsstufe, der Kaufpreis als metrisch skalierter Wert, die Bewertung und die Primär-Kategorie in den App Store’s. Bei den impliziten, expliziten und keinen Erklärungsbedarfen kommen noch als weitere genutzte Variablen die Ober- und Unterkategorien der Taxonomie hinzu. Die logistische Regression zu impliziten, expliziten und keinen Erklärungsbedarfen wird einmal ohne die weiteren Variablen der Ober- und Unterkategorien der Taxonomie durchgeführt. Bei den weiteren logistischen Regressionen wird den Ober- und Unterkategorien der Taxonomie, der implizite, explizite und kein Erklärungsbedarf als Variable hinzugefügt.
Da nicht alle Variablen als Zahlenwert dargestellt sind, wie die Variable $D_{Kategorie}$ aus der Tabelle 4.4, müssen die Variablen für die logistische Regression angepasst werden. Deshalb wird der Variable $D_{Kategorie}$ der Primär-Kategorie eine natürliche Zahl zugewiesen. Für die Zuweisung müssen als Erstes alle möglichen Kategorien aus den Reviews herausgefiltert

werden. Dafür wird von allen Reviews, die Primär-Kategorie in eine Liste gespeichert, wenn aber die Primär-Kategorie schon vorhanden ist, wird sie nicht mehr gespeichert. Anschließend wird den Kategorien eine natürliche Zahl zugewiesen und den Reviews zugeordnet.
Alle weiteren Werte, die in den logistischen Regressionen vorkommen, werden direkt zugeordnet, da es sich um Zahlen handelt. Bei den logistischen Regressionen über die Ober- und Unterkategorie der Taxonomie werden die Erklärungsbedarfs-Typen als 0, 1 und 2 Werte (kein, expliziter, impliziter Erklärungsbedarf) eingetragen. Bei der logistischen Regression über die Typen des Erklärungsbedarfs wird für die Oberkategorien der Taxonomie immer eine eigene Liste angelegt, da es bei Reviews auch mehrere Erklärungsbedarfs-Anfragen geben kann. Diese können sich auch in der Art der Ober- und Unterkategorien wiederholen. Für eine gute Abbildung wird die Spalte, die zu einem bestimmten Punkt in der Taxonomie gehört hochgezählt, wenn in der Analyse der Review der Punkt der Taxonomie erwähnt wird.
Nachdem alle App-Eigenschaften den Reviews zugeordnet sind, werden die Daten aufgeteilt, in die oben beschreibenden Test- und Trainingsdaten unterteilt und in Listen gespeichert. Danach beginnt die eigentliche Berechnung der logistischen Regression. *Die Anzahl an Reviews* haben einen zu großen Einfluss auf den Ausgang der logistischen Regression. Dies zeigt sich daran, dass als Ergebnis immer kein Erklärungsbedarf vorhergesagt wird. Daher wird Pipeline erzeugt, die die Werte nochmals anpasst, damit die Extremwerte nicht einen zu großen Einfluss auf das Endergebnis haben. Folglich wird das Modell mit den Trainingsdaten trainiert, um mithilfe der Testdaten festzustellen wie gut das Modell funktioniert. Der *Score* wird in % zurückgegeben. Der *RMSE* „regression mean squared error “ ist eine Risikofunktion, die dem Erwartungswert des Quadratischen Fehlerverlustes entspricht und angibt wie nah eine Regressionskurve an den Datenpunkten liegt. Der Bestimmungskoeffizient r^2 gibt die Qualität der logistischen Regression an und liegt zwischen 1 (perfekt) und 0 (unbrauchbar), r^2 kann auch negativ sein.

Kapitel 5

Ergebnisse

In diesem Kapitel werden die Ergebnisse aus den unterschiedlichen Korrelationsanalyseverfahren sowie der logistischen Regressionen und deren Validierungen beurteilt.

5.1 Ergebnisse der Korrelationsanalyse

In der Tabelle 5.1 werden die Ergebnisse der Korrelationsanalysen dargestellt und die Nullhypothesen der Tabellen 4.1, 4.2 und 4.3 referiert. Danach werden die Ergebnisse des jeweiligen Korrelationsanalyseverfahrens und deren p-Werte präsentiert. Zum Schluss wird die Interpretation der Ergebnisse mit der Bonferroni Korrektur dargestellt. Die Nullhypothesen werden verworfen, wenn das Signifikanzniveau α unterschritten wird, dafür wird die Bonferroni Korrektur berechnet. Die Bonferroni Korrektur berechnet das Signifikanzniveau für die Oberhypothesen *H2, H4, H6, H8, H10, H12, H14, H16 und H18* $\alpha = 0,01667$. Für die Oberhypothesen *H1, H3, H5, H7, H9, H11, H13, H14, H17, H19, H20, H21; H22, H23, H24, H25, H26 and H27* bleibt das Signifikanzniveau auf den Standartwert $\alpha = 0,05$. Die Bonferroni Korrektur verändert zum Standart Signifikanzniveau nur die Hypothese *H18.1*. Durch die Bonferroni Korrektur werden die Oberhypothesen *H1, H2, H3, H4, H6, H7, H8, H9, H10, H11, H12, H13, H14, H19, H20, H23, H24 und H26* verworfen. Dadurch wird die Gegenhypothese zur Nullhypothese angenommen und die Ergebnisse dieser Oberhypothesen gelten als signifikant.
Für die Forschungsfrage RQ1, welche sich auf vom Unternehmen festgelegten App-Eigenschaften bezieht, bleiben die Oberhypothesen *H1, H2, H3, H4, H6, H7, H8, H9, H10, H11, H12, H19, H20, H23 und H24* übrig. In der Tabelle 5.1 ist abzulesen, dass ein schwacher Zusammenhang zwischen Erklärungsbedarf und der Primär-Kategorie, sowie der Kostenpflichtigkeit der App besteht. Zudem gibt es auch nur einen schwachen Zusammenhang zwischen dem Kaufpreis der App und Arten von Erklärungsbedarf. Bei den

vier Hypothesen zum durchschnittlichen Erklärungsbedarf, die signifikant sind, gibt es einen schwachen Einfluss auf den durchschnittlichen Erklärungsbedarf. Beim Ingame Preis gibt es keinen Zusammenhang, ob Erklärungsbedarf vorhanden ist, jedoch gibt es einen schwachen Zusammenhang darauf, welche Art von Erklärungsbedarf vorliegt. Bei den Hypothesen zu den Oberhypothesen *H9 und H10* liegt teilweise ein mittlerer Zusammenhang auf den Erklärungsbedarf vor.
Die Forschungsfrage RQ2 bezieht sich auf die durch den Nutzer erzeugten App-Eigenschaften. Die signifikanten Oberhypothesen sind *H13, H14 und H26*. Aus der Tabelle 5.1 geht hervor, dass die meisten Hypothesen zur RQ2 nicht verworfen werden. Die einzigen Hypothesen die signifikant sind, überprüfen den Zusammenhang zwischen den durchschnittlichen Erklärungsbedarf pro App mit der Review Anzahl der App und den Erklärungsbedarf zu der durchschnittlichen Sterne Bewertung der App. Die Ausprägung, die diese Hypothesen aufweisen, ist ein mittlerer Zusammenhang.

Hypothese	**Ergebnis**	**P-Wert**	**Verfahren**	**Interpretation**	**Bonferroni**
$H1_0$	0.2215	$1.3794*10^{-38}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H2.1_0$	0.1645	$2.0978*10^{-29}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H2.2_0$	0.1822	$6.5051*10^{-28}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H2.3_0$	0.1520	$1.4871*10^{-34}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H3_0$	0.1218	$6.6660*10^{-16}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H4.1_0$	0.1218	$1.0891*10^{-29}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H4.2_0$	0.1207	$1.9756*10^{-7}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H4.3_0$	0.1909	$1.2142*10^{-9}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H5_0$	0.0327	0.1807	Eta	Kein Zusammenhang	Nicht verwerfen
$H6.1_0$	0.1242	$7.8189*10^{-31}$	Eta	Schwacher Zusammenhang	Verwerfen
$H6.2_0$	0.0030	0.0543	Eta	Kein Zusammenhang	Nicht verwerfen
$H6.3_0$	0.0127	0.1129	Eta	Kein Zusammenhang	Nicht verwerfen
$H7_0$	0.0785	0.0036	Cramer‘s V	Kein Zusammenhang	Verwerfen
$H8.1_0$	0.0555	0.0396	Cramer‘s V	Kein Zusammenhang	Nicht verwerfen
$H8.2_0$	0.1021	0.1157	Cramer‘s V	Schwacher Zusammenhang	Nicht verwerfen
$H8.3_0$	0.2626	0.0025	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H9_0$	0.3885	$4.9284*10^{-88}$	Cramer‘s V	Mittlerer Zusammenhang	Verwerfen
$H10.1_0$	0.3033	$5.4634*10^{-79}$	Cramer‘s V	Mittlerer Zusammenhang	Verwerfen
$H10.2_0$	0.2789	$3.4282*10^{-239}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H10.3_0$	0.2291	0	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H11_0$	0.1274	$9.8731*10^{-16}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H12.1_0$	0.0911	$4.5475*10^{-14}$	Cramer‘s V	Kein Zusammenhang	Verwerfen
$H12.2_0$	0.0900	$1.8186*10^{-29}$	Cramer‘s V	Kein Zusammenhang	Verwerfen
$H12.3_0$	0.1171	$7.8209*10^{-35}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H13_0$	0.3983	$1.83074*10^{-85}$	Cramer‘s V	Mittlerer Zusammenhang	Verwerfen
$H14.1_0$	0.3104	$2.6761*10^{-72}$	Cramer‘s V	Mittlerer Zusammenhang	Verwerfen
$H14.2_0$	0.2982	$1.8092*10^{-261}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H14.3_0$	0.2417	0	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H15_0$	0.1598	0.3997	Eta	Schwacher Zusammenhang	Nicht verwerfen
$H16.1_0$	0.0219	0.1481	Eta	Kein Zusammenhang	Nicht verwerfen
$H16.2_0$	0.0253	0.1589	Eta	Kein Zusammenhang	Nicht verwerfen
$H16.3_0$	0.0270	0.1643	Eta	Kein Zusammenhang	Nicht verwerfen
$H17_0$	0.0216	0.1471	Eta	Kein Zusammenhang	Nicht verwerfen
$H18.1_0$	0.0067	0.0820	Eta	Kein Zusammenhang	Nicht verwerfen
$H18.2_0$	0.0026	0.0508	Eta	Kein Zusammenhang	Nicht verwerfen
$H18.3_0$	0.0109	0.1043	Eta	Kein Zusammenhang	Nicht verwerfen
$H19_0$	-0,2022	0,0249	Spearman	Schwacher Zusammenhang	Verwerfen
$H20_0$	-0,2236	0,0129	Spearman	Schwacher Zusammenhang	Verwerfen
$H21_0$	-0.088	0,3353	Pearson	Kein Zusammenhang	Nicht verwerfen
$H22_0$	-0.2289	0.0735	Spearman	Schwacher Zusammenhang	Nicht verwerfen
$H23_0$	0,1857	0,0398	Spearman	Schwacher Zusammenhang	Verwerfen
$H24_0$	0,2575	0,0040	Spearman	Schwacher Zusammenhang	Verwerfen
$H25_0$	0,0181	0,8427	Spearman	Kein Zusammenhang	Nicht verwerfen
$H26_0$	0,383	0,00001	Pearson	Mittlerer Zusammenhang	Verwerfen
$H27_0$	-0,172	0,182	Pearson	Schwacher Zusammenhang	Nicht verwerfen

Tabelle 5.1: Ergebnis der Korrelationsanalysen

5.2 Ergebnisse der Logistischen Regressionen

Die logistischen Regressionen werden durchgeführt, um die Forschungsfrage RQ3 zu beantworten. In der Tabelle 5.2 werden die Wahrscheinlichkeiten *Score's* zur Richtigkeit der Vorhersagen, der RMSE und r^2 der logistischen Regression dargestellt.
Die logistischen Regressionen werden sowohl für den gesamten Goldstandard-Datensatz, wie für die unterschiedlichen App Stores berechnet. Hierbei ist zu beachten, dass die logistischen Regressionen zu den Unterkategorien nur auf den gesamten Goldstandard-Datensatz durchgeführt werden. Dies liegt daran, dass die Anzahl der Zuordnungen einiger Unterkategorien so gering ist, dass sie nicht in den einzelnen Datensätzen des jeweiligen App Stores vorkommen. Zudem kommt durch die Aufteilung in 13 Unterkategorien das Auftauchen im Goldstandard-Datensatz generell seltener vor.
In der Tabelle 5.2 ist deutlich zu erkennen, dass der Bestimmungskoeffizient r^2 der beiden logistischen Regressionen zu expliziten, impliziten und keinem Erklärungsbedarf recht hoch ist. Dieser liegt um einen Wert von 0,7. Aufgrund des Wertes kann davon ausgegangen werden, dass die logistische Regression eine recht genaue Aussage zur Einteilung in expliziten, impliziten und keinen Erklärungsbedarf treffen. Zudem hat die logistische Regression beim gesamten Goldstandard-Datensatz eine Trefferwahrscheinlichkeit zum richtig prognostizierten Wert in Höhe von 84%. Bei den getesteten Oberkategorien fällt auf, dass nur die Oberkategorie *Direkte Systemaspekte* einen hohen Wert für den Bestimmungskoeffizienten r^2 hat. Dies lässt sich darauf zurückführen, dass die *Direkten Systemaspekte* am häufigsten im Goldstandard-Datensatz vorkommen, wodurch mehr und besser auf den Fall trainiert werden kann.
Alle anderen Fälle haben einen niedrigen r^2 Wert. Sie sind daher nicht zur Vorhersage von Ober- und Unterkategorien geeignet.

Vorhergesagter Erklärungsbedarf	**Genutzter Erklärung.**	**Genutzte Daten**	**Score**	**RMSE**	r^2
Kein, explizit, implizit	Oberkategorie	Beide Stores	0.84	0.4	0.703
Kein, explizit, implizit	Oberkategorie	Playstore	0.87	0.37	0.726
Kein, explizit, implizit	Oberkategorie	Appstore	0.83	0.41	0.695
Oberkategorie (Direkte Systemaspekte)	Kein, explizit, implizit	Beide Stores	0.9	0.32	0.574
Oberkategorie (Business)	Kein, explizit, implizit	Beide Store	0.9	0.32	-0.115
Oberkategorie (Timing/Context)	Kein, explizit, implizit	Beide Store	0.98	0.14	-0.021
Oberkategorie (Indirekte Systemaspekte)	Kein, explizit, implizit	Beide Store	0.96	0.2	-0.042
Oberkategorie (Direkte Systemaspekte)	Kein, explizit, implizit	Playstore	0.9	0.32	0.581
Oberkategorie (Business)	Kein, explizit, implizit	Playstore	0.92	0.29	-0.092
Oberkategorie (Timing/Context)	Kein, explizit, implizit	Playstore	1.0	0.06	-0.004
Oberkategorie (Indirekte Systemaspekte)	Kein, explizit, implizit	Playstore	0.98	0.14	-0.02
Oberkategorie (Direkte Systemaspekte)	Kein, explizit, implizit	Appstore	0.89	0.33	0.564
Oberkategorie (Business)	Kein, explizit, implizit	Appstore	0.89	0.33	-0.12
Oberkategorie (Timing/Context)	Kein, explizit, implizit	Appstore	0.98	0.16	-0.025
Oberkategorie (Indirekte Systemaspekte)	Kein, explizit, implizit	Appstore	0.95	0.21	-0.047
Unterkategorie (Unerwartetes Verhalten)	Kein, explizit, implizit	Beide Stores	0.83	0.41	-0.403
Unterkategorie (Bugs/Abstürze)	Kein, explizit, implizit	Beide Stores	0.99	0.11	-0.013
Unterkategorie (Algorithmen)	Kein, explizit, implizit	Beide Stores	1.0	0.05	-0.003
Unterkategorie (Operation)	Kein, explizit, implizit	Beide Stores	0.86	0.37	-0.158
Unterkategorie (Navigation)	Kein, explizit, implizit	Beide Stores	0.99	0.09	-0.009
Unterkategorie (Einführung)	Kein, explizit, implizit	Beide Stores	0.99	0.12	-0.015
Unterkategorie (Metainformationen)	Kein, explizit, implizit	Beide Stores	0.96	0.2	-0.043
Unterkategorie (Änderung)	Kein, explizit, implizit	Beide Stores	0.98	0.14	-0.02
Unterkategorie (Zukunftsplan)	Kein, explizit, implizit	Beide Stores	0.98	0.13	-0.018
Unterkategorie (Systemspezifische Aspekte)	Kein, explizit, implizit	Beide Stores	0.96	0.2	-0.04
Unterkategorie (Begrifflichkeiten)	Kein, explizit, implizit	Beide Stores	1.0	0.03	-0.001
Unterkategorie (GUI/Designentscheidungen)	Kein, explizit, implizit	Beide Stores	0.99	0.1	-0.01
Unterkategorie (Privacy)	Kein, explizit, implizit	Beide Stores	1.0	0.07	-0.004

Tabelle 5.2: Ergebnis der logistischen Regression

5.2.1 Validierung der Logistischen Regressionen

Um die ermittelten Ergebnisse der logistischen Regressionen zu validieren wird ein neuer Validierungs-Datensatz erstellt und in expliziten, impliziten und keinen Erklärungsbedarf, sowie in den Ober- und Unterkategorien gelabelt. Dieser Validierungs-Datensatz basiert auf 10 unterschiedlichen Apps, mit jeweils fünf Apps aus dem Appstore und fünf Apps aus dem Playstore. Pro App wurden die letzten 50 Reviews geladen (495 Reviews). Daraus stammen 245 aus dem Appstore und 250 aus dem Playstore. Diese Reviews werden gesichtet und händisch separat von zwei Informatikern gelabelt mit Wissen zu Erklärungsbedarf, um die Qualität des Validierungs-Datensatzes zu verbessern. Anschließend wird die Labelung der Reviews verglichen. Bei den Labelungen, bei denen es zu Abweichungen zwischen den beiden Ratern kommt, werden diese besprochen und im Konsens korrigiert.
Um zu überprüfen, wie objektiv die Labelungen der Reviews des Validierungs-Datensatzes sind, wird das Cohens Kappa berechnet 2.10, welches 1977 von Landis und Koch [17] interpretiert wurde. Dafür werden in den Tabellen 5.3 und 5.4 alle Einträge der gelabelten Reviews mit gleichen Aussagen gezählt. Dies gilt für die Fälle *no-explanation-need* zu *no-explanation-need*, *explizit* zu *explizit* und *implizit* zu *implizit*. Danach werden die Fälle zusammengezählt, die von den Ratern unterschiedlich gelabelt werden. Die Aufteilung erfolgt je nachdem, um welchen Fall es sich handelt. Das Cohens Kappa K beträgt für diesen Validierungs-Datensatz bei den expliziten, impliziten und keinen Erklärungsbedarf 0,89, und bei der Oberkategorie 0,88 was für beide Werte auf einen nahezu perfekten Wert nach Landis und Koch [17] schließen lässt.
Durch das Ergebnis der Berechnung kann davon ausgegangen werden, dass der Validierungs-Datensatz objektiv gelabelt ist und für die Validierung der logistischen Regressionen genutzt werden kann.

Aussage	**Labelung der Daten**			**N**
	Kein	Explizit	Implizit	
#	430	43	22	495
%	86,87%	8,69%	4,44%	100,00%

Tabelle 5.3: Labelung des Validierungs-Datensatzes in expliziten, impliziten und keinen Erklärungsbedarf

Aussage	**Labelung der Oberkategorien**				**N**
	Business	Direkte. Sys.	Indirekte Sys.	Kein	
#	9	54	3	429	495
%	1,82%	10,91%	0,61%	86,67%	100,00%

Tabelle 5.4: Labelung des Validierungs-Datensatzes der Oberkategorien

Nachdem der Validierungs-Datensatz gelabelt ist, werden die Metainformationen der Reviews und Apps in die logistische Regression als Variablen eingebettet. Mit dem Goldstandard-Datensatz wird die logistische Regression trainiert und anhand des Validierungs-Datensatzes getestet. Die Werte sind in der Tabelle 5.5 eingefügt. In der Tabelle 5.5 werden wieder die logistischen Regressionen mit ihren Werten Score, RMSE und r^2 aufgelistet. Dabei fallen direkt die Werte auf, die eine Score von 1, eine RMSE von 0 und r^2 von 1 haben. Dies liegt daran, dass diese Werte im zu validierenden Validierungs-Datensatz nicht vorkommen und auch nicht vorhergesagt werden. Der höhere Score des Wertes lässt sich darauf zurückführen, dass weniger Reviews zur Validierung genutzt werden. Des Weiteren ist zu sehen, dass die expliziten, impliziten und keine Erklärungsbedarfe mit den Variablen der Oberkategorien einen hohen r^2-Wert haben. Damit besteht eine hohe Wahrscheinlichkeit, dass das Ergebnis richtig vorhergesagt wird. Dies ist nur möglich durch die Oberkategorien der Review. Hierbei ist zu beachten, dass kein impliziter Erklärungsbedarf vorhergesagt wird. Dies kann daran liegen, dass zu wenige der Reviews einen impliziten Erklärungsbedarf aufweisen. Insgesamt sind alle Fälle von keinen Erklärungsbedarf richtig identifiziert. In dem Fall, dass ein Erklärungsbedarf vorliegt, wurden vier Fälle nicht erkannt. Davon sind zwei Fälle expliziter Erklärungsbedarf und zwei Fälle impliziter Erklärungsbedarf. Hierbei ist jedoch zu beachten, dass von den vier falsch kategorisierten Fällen drei von der logistischen Regression zur Oberkategorie die Kategorie *Direkte Systemaspekte* zugewiesen wurden. Des Weiteren ist interessant, dass alle impliziten Erklärungsbedarfe als *Direkte Systemaspekte* deklariert sind. Zudem konnten von der logistischen Regression zu den Oberkategorien nur *Direkte Systemaspekte* vorhergesagt werden. Bei den Unterkategorien konnte nur drei Mal *Unerwartetes Systemverhalten* festgestellt werden. Vom festgestellten *unerwarteten Systemverhalten* sind zwei falsch zugeordnet.
Bei der Durchführung zu den expliziten, impliziten und kein Erklärungsbedarfen ohne die Variablen zu den Oberkategorien, konnte kein Erklärungsbedarf festgestellt werden. Aufgrund der Ergebnisse der Analyse ist die Vorhersage möglich.
Die Vorhersage des spezifischen Erklärungsbedarfs, beziehungsweise um welche Art von Erklärungsbedarf es sich handelt, ist aufgrund der Ergebnisse der angewandten Prüfverfahren nicht möglich. Zudem ist die Vorhersage, ob Erklärungsbedarf überhaupt vorhanden ist, nicht möglich, außer, wenn die Oberkategorien mit eingebunden sind. In diesem Fall ist eine sehr genaue Vorhersage, ob Erklärungsbedarf vorhanden ist, möglich. Jedoch ist klar, wenn eine Oberkategorie vorhanden ist, muss auch Erklärungsbedarf vorhanden sein.

Vorhergesagter Erklärungsbedarf	**Genutzter Erklärung.**	**Genutzte Daten**	**Score**	**RMSE**	r^2
Kein, explizit, implizit	Oberkategorie	Beide Stores	0.95	0.25	0.735
Kein, explizit, implizit	Oberkategorie	Playstore	0.93	0.29	0.632
Kein, explizit, implizit	Oberkategorie	Appstore	0.95	0.25	0.735
Kein, explizit, implizit		Beide Stores	0.86	0.52	-0.137
Kein, explizit, implizit		Playstore	0.87	0.54	-0.133
Kein, explizit, implizit		Appstore	0.86	0.5	-0.141
Oberkategorie (Direkte Systemaspekte)	Kein, explizit, implizit	Beide Stores	0.97	0.18	0.644
Oberkategorie (Business)	Kein, explizit, implizit	Beide Store	0.98	0.14	-0.021
Oberkategorie (Timing/Context)	Kein, explizit, implizit	Beide Store	1	0	1
Oberkategorie (Indirekte Systemaspekte)	Kein, explizit, implizit	Beide Store	0.99	0.08	-0.006
Oberkategorie (Direkte Systemaspekte)	Kein, explizit, implizit	Playstore	0.95	0.22	0.488
Oberkategorie (Business)	Kein, explizit, implizit	Playstore	0.9	0.32	-4.205
Oberkategorie (Timing/Context)	Kein, explizit, implizit	Playstore	1	0	1
Oberkategorie (Indirekte Systemaspekte)	Kein, explizit, implizit	Playstore	0.89	0.32	-16.439
Oberkategorie (Direkte Systemaspekte)	Kein, explizit, implizit	Appstore	0.97	0.18	0.644
Oberkategorie (Business)	Kein, explizit, implizit	Appstore	0.98	0.14	-0.021
Oberkategorie (Timing/Context)	Kein, explizit, implizit	Appstore	1	0	1
Oberkategorie (Indirekte Systemaspekte)	Kein, explizit, implizit	Appstore	0.99	0.08	-0.006
Unterkategorie (Unerwartetes Verhalten)	Kein, explizit, implizit	Beide Stores	0.99	0.08	0
Unterkategorie (Bugs/Abstürze)	Kein, explizit, implizit	Beide Stores	1	0.06	-0.004
Unterkategorie (Algorithmen)	Kein, explizit, implizit	Beide Stores	1	0.04	-0.002
Unterkategorie (Operation)	Kein, explizit, implizit	Beide Stores	0.98	0.15	-0.023
Unterkategorie (Navigation)	Kein, explizit, implizit	Beide Stores	1	0.04	-0.002
Unterkategorie (Einführung)	Kein, explizit, implizit	Beide Stores	1	0.04	-0.002
Unterkategorie (Metainformationen)	Kein, explizit, implizit	Beide Stores	1	0	1
Unterkategorie (Änderung)	Kein, explizit, implizit	Beide Stores	1	0	1
Unterkategorie (Zukunftsplan)	Kein, explizit, implizit	Beide Stores	1	0	1
Unterkategorie (Systemspezifische Aspekte)	Kein, explizit, implizit	Beide Stores	1	0	1
Unterkategorie (Begrifflichkeiten)	Kein, explizit, implizit	Beide Stores	1	0	1
Unterkategorie (GUI/Designentscheidungen)	Kein, explizit, implizit	Beide Stores	1	0	1
Unterkategorie (Privacy)	Kein, explizit, implizit	Beide Stores	1	0.04	-0.002

Tabelle 5.5: Ergebnis der Validierung der logistischen Regression

Kapitel 6

Diskussion

In diesem Kapitel werden die Forschungfragen beantwortet und interpretiert.

6.1 Beantwortung der Forschungsfragen

RQ1: *Gibt es eine Korrelation zwischen dem vom Unternehmen festgelegten App-Eigenschaften und den Arten von Erklärungsbedarfen?*
Ja, zwischen den Arten von Erklärungsbedarfen und den vom Unternehmen festgelegten App-Eigenschaften gibt es teilweise mittlere Korrelationen, die auf einen moderaten Grad an Zusammenhängen hinweisen. Im Allgemeinen ist der Einfluss jedoch schwach bis gar nicht vorhanden. Die Ausnahme ist die Korrelation zwischen Erklärungsbedarf und die Versionsstufe der App, welche teilweise einen mittleren Zusammenhang aufweisen.
RQ2: *Gibt es eine Korrelation zwischen dem vom Nutzerfeedback entnommenen App-Eigenschaften und den Arten von Erklärungsbedarfen?*
Ja, es gibt eine mittlere Korrelation zwischen den durchschnittlichen Erklärungsbedarf pro App und der Anzahl an Reviews von der App, sowie teilweise mittlere Einflüsse zwischen Erklärungsbedarf und der Sterne Bewertung der App. Jedoch sind alle anderen Hypothesen, die sich auf **RQ2** beziehen, nicht signifikant und haben dadurch keinen Einfluss auf den Erklärungsbedarf. Aufgrund der mittleren Korrelation ist davon auszugehen, dass ein moderater Einfluss vorliegt.
RQ3: *Kann man mit einer Kombination von App-Eigenschaften eine Vorhersage über einen Erklärungsbedarf treffen?*
Nein, es kann keine Vorhersage mit der logistischen Regression anhand der App-Eigenschaften erbracht werden.

6.2 Interpretation

Wie bereits in 6.1 erwähnt, geht aus den Korrelationsanalysen hervor, dass die vom Unternehmen festgelegten App-Eigenschaften teilweise eine mittlere

Korrelation haben. Dies bedeutet nicht, dass es eine Kausalität zwischen dem Erklärungsbedarf und den App-Eigenschaften gibt. Dazu wurden die logisitischen Regressionen durchgeführt, um eine mögliche Kausalität nachzuweisen. Aus den Ergebnissen der logistischen Regressionen ist zu lesen, dass keine Kausalität vorliegt. Hierbei könnten weitere Forschungen zu den App-Eigenschaften erfolgen, wie zum Beispiel: *neue Funktionen* und ob dadurch mehr Erklärungsbedarfe von bestimmten Ober- und Unterkategorien auftreten können oder ob *die Chart-Platzierung* einen Einfluss auf den Erklärungsbedarf hat.
Die Korrelationsanalysen zu dem Nutzerfeedback zeigen, dass nur wenige Aspekte des Nutzerfeedbacks einen Einfluss auf den Erklärungsbedarf aufweisen. Diese haben jedoch einen moderaten Einfluss auf den Erklärungsbedarf. Dies lässt nicht darauf schließen, dass das Nutzerfeedback immer einen mittleren Einfluss auf den Erklärungsbedarf hat. Die Ergebnisse sollten Anreiz dafür sein, weiter im Forschungsfeld Erklärungsbedarf/Nutzerfeedback zu forschen.
Bei der logistischen Regression kam heraus, dass mit den App-Eigenschaften aus dieser Arbeit keine Vorhersage darüber getroffen werden kann, ob Erklärungsbedarf vorhanden ist. Daraus sollte nicht generell die Schlussfolgerung gezogen werden, dass mit App-Eigenschaften keine Vorhersage zu Erklärungsbedarf möglich ist. Sondern, dass die hier genutzten App-Eigenschaften nicht ausreichen, um eine Vorhersage zu treffen. Des Weiteren ist interessant, dass bei den Ergebnissen zu den Oberkategorien eine meist zutreffende Vorhersage zu den *Direkte Systemaspekte* getroffen wurde. Dieses kommt vermutlich dadurch zustande, dass bei der logistischen Regression der Faktor, ob Erklärungsbedarf vorliegt, mitgegeben wird. Ein weiterer Grund könnte darauf zurückzuführen sein, dass *Direkte Systemaspekte* die am häufigsten gelabelte Kategorie ist, wodurch die Wahrscheinlichkeit die Kategorie richtig zu treffen steigt. Im Umkehrschluss könnte daraus auch geschlossen werden, weshalb die anderen Oberkategorien nicht vorhergesagt werden können, da für diese zu wenig Daten vorliegen. Diese Erkenntnis lässt sich ebenfalls auf die Unterkategorien übertragen, da zu wenig Daten vorliegen, um sie vorherzusagen. Hier wäre interessant mit einem größeren Datensatz zu forschen, um festzustellen, ob dadurch eine Vorhersage zu den Ober- und Unterkategorienen möglich ist.

Kapitel 7

Validität

In diesem Kapitel werden die *Threats to Validity* nach Wohlin et al. [25] betrachtet und auf diese Arbeit angewendet. Die *Threats to Validity* lassen sich in vier Unterpunkte aufteilen. *Threats to Construct Validity* beschreibt, welche Fehler bei der Planung passiert sind oder sich nicht vermeiden ließen. *Threats to Internal Validity* erläutert, ob es Messfehler gibt oder Fehler bei der Datenauswertung entstehen. *Threats to Conclusion Validity* beschreibt, wie belastbar die Ergebnisse sind. *Threats to External Validity* beschreibt, ob die Ergebnisse auch auf andere Kontexte zu übertragen sind.
Der Goldstandard-Datensatz wurde auf Fehler überprüft, dabei könnten aber Fehler übersehen worden sein. Zudem könnten auch bei den gelabelten Reviews Fehler in der Labelung vorliegen. In diesen Fällen liegt ein *Threats to Construct Validity* vor.
Bei der Durchführung der Korrelationsanalyse und der logistischen Regression könnte, trotz mehrfacher Durchführungen der Korrelationsanalyse und der logistischen Regression, ein Fehler in der Zuweisung der Metainformationen zu den Reviews erfolgt sein. Dadurch kann eine Verfälschung der Ergebnisse entstehen. Dies ist ein *Threats to Internal Validity*. Zudem wurden die Ergebnisse nicht durch eine zweite Instanz validiert, was die Fehleranzahl verringern würde. Dies weist auf ein *Threats to Construct Validity* hin.
Beim Erstellen des Validierungs-Datensatzes wurde von zwei Ratern der Validierungs-Datensatz gelabelt. Hierbei können trotzdem Fehler bei der Labelung erfolgt sein. Die Fehleranfälligkeit wird zwar durch zwei Rater minimiert, aber nicht ausgeschlossen. Dadurch ist in dem Fall von einem *Threats to Construct Validity* auszugehen.
Da die Ergebnisse nur durch eine Person interpretiert wurden, kann es zu Fehlern in der Interpretation kommen. Fehlinterpretationen würden durch eine zweite Person, die die Ergebnisse interpretiert, minimiert werden. Da eine Person die Ergebnisse interpretiert, liegt ein *Threats to Internal Validity* vor. Im Validierungs-Datensatz sind einige Unterkategorien nicht vertreten. Zudem sind über den gesamten Goldstandard-Datensatz die Ober- und

Unterkategorien nur in geringer Anzahl vorhanden. Generell gibt es im Validierungs-Datensatz wenig impliziten und expliziten Erklärungsbedarf und die Stichprobe des Validierungs-Datensatzes ist recht gering. Dies sind mögliche Probleme der *Threats to Conclusion Validity*. Aufgrund der besprochenen möglichen Problematiken bei der Validität sind die erworbenen Ergebnisse aus der Validierung nicht optimal und es könnten mit einem größeren Datensatz genauere Daten erstellt werden. Was ein *Threats to Construct Validity* ist.
Durch die Anwendung der Bonferroni-Korrektur wird der Typ-I-Fehler (false Positiv) gesenkt, dafür kann der Typ-II-Fehler (false Negativ) angehoben werden. Dadurch liegt ein *Threats to Construct Validity* vor.
Die Apps aus den Goldstandard-Datensatz und dem Validierungs-Datensatz spiegeln nicht alle App-Kategorien wieder, und geben daher auch nicht alle App Kombinationen wieder. Somit besteht ein *Threats to External Validity*, da die Ergebnisse nicht auf die Allgemeinheit angewendet werden können.

Kapitel 8

Zusammenfassung und Ausblick

In diesem Kapitel wird eine Zusammenfassung der wichtigsten Analyseverfahren und Ergebnisse gegeben, sowie ein Ausblick auf weitere Möglichkeiten zur Forschung auf dem Gebiet.

8.1 Zusammenfassung

Das Ziel der Arbeit war es einen möglichen Zusammenhang zwischen dem Erklärungsbedarf und den Metadaten einer App zu identifizieren und diesen Zusammenhang vorherzusagen. Durch die Vorhersage könnten voll Erklärbare Systeme entwickelt werden, wie z.B. *AI* oder *neu entwickelte Apps*. Dies würde die Nachvollziehbarkeit der Systeme verbessern und auch die Nutzerzufriedenheit anheben. Um das Ziel zu erreichen, wurden bereitgestellte und für diese Arbeit ermittelte Daten genutzt, mit denen mehrere Korrelationsanalysen mit den Verfahren Cramer‘s V, Eta-Koeffizient, Spearman-Rangkorrelation und Bravais-Pearson-Korrelation durchgeführt wurden. Nach den Korrelationsanalysen wurden mehrere logistische Regressionen durchgeführt.
Die Ergebnisse der Korrelationsanalysen zeigen, dass die App-Eigenschaften, die aus den Nutzerfeedback entstehen, nur einen mittleren Einfluss auf den Erklärungsbedarf aufweisen, aber nur wenige überhaupt einen Einfluss haben. Die App-Eigenschaften, die vom Unternehmen eingestellt werden, weisen teilweise eine mittlere Korrelation mit Erklärungsbedarf auf. Diese mittlere Korrelation ist seltener vertreten und eher an der unteren Grenze zur schwachen Korrelation angesiedelt. Mit Hilfe der Ergebnisse der Korrelationsanalysen kann durch eine logistische Regression überprüft werden, ob eine Vorhersage möglich ist, um einen Kausalzusammenhang zu überprüfen. Durch die logistischen Regressionen konnten mehrere Eigenschaften auf die Art des Erklärungsbedarfs überprüft werden. Bei den logistischen

Regressionen kam jedoch heraus, dass keine sinnvolle Möglichkeit der Vorhersage für die Ober- und Unterkategorien der Taxonomie, sowie für den expliziten, impliziten und keinen Erklärungsbedarf vorliegt.

8.2 Ausblick

In weiteren Arbeiten ist es sinnvoll einen größeren Datensatz anzulegen, in dem die seltener vorkommenden Ober- und Unterkategorien zahlreicher vertreten sind, damit diese in einer logistischen Regression besser vorhergesagt werden können. Zudem wäre eine logistische Regression mit den App-Eigenschaften und dem Domänenwissen interessant, um zu überprüfen, ob eine bessere Vorhersage mit den zusätzlichen Informationen möglich ist. Es könnten auch alternative Verfahren zur Erkennung genutzt werden, wie beispielhaft eine Support-Vektor-Maschine oder die Probit-Regression. Diese Verfahren haben andere Nachteile als die logistische Regression. Die Vorteile sind jedoch das Erfassen nicht linearer Beziehungen oder eine bessere Verarbeitung von Extremwerten.
Des Weiteren wäre es sinnvoll mehr App-Eigenschaften zu analysieren, da diese eventuell einen größeren Einfluss auf den Erklärungsbedarf aufweisen, als die hier genutzten Eigenschaften. Diese wären zum Beispiel: *neue Funktionen* oder *die Chart-Platzierung*. Zudem könnte eine Clusteranalyse durchgeführt werden, um zusammenhängende Strukturen in den Daten zu finden.

Anhang A

Anhang

A.1 Korrelationsanalyse

Nr.	Hypothese	RQ
$H1_0$	Es besteht kein Zusammenhang zwischen der expliziten Erklärungsbedarf und der Kategorie der untersuchten App.	RQ1
$H2_0$	Es besteht kein Zusammenhang zwischen dem impliziten Erklärungsbedarf und der Kategorie der untersuchten App.	RQ1
$H3_0$	Es besteht kein Zusammenhang zwischen dem expliziten Erklärungsbedarf und dem Kaufpreis der untersuchten App.	RQ1
$H4_0$	Es besteht kein Zusammenhang zwischen dem impliziten Erklärungsbedarf und dem Kaufpreis der untersuchten App.	RQ1
$H5_0$	Es besteht kein Zusammenhang zwischen dem expliziten Erklärungsbedarf und den zu zahlenden Kaufpreis der untersuchten App.	RQ1
$H6_0$	Es besteht kein Zusammenhang zwischen dem impliziten Erklärungsbedarf und den zu zahlenden Kaufpreis der untersuchten App.	RQ1
$H7_0$	Es besteht kein Zusammenhang zwischen dem expliziten Erklärungsbedarf und dem Ingame Preis der untersuchten App.	RQ1
$H8_0$	Es besteht kein Zusammenhang zwischen dem impliziten Erklärungsbedarf und dem Ingame Preis der untersuchten App.	RQ1
$H9_0$	Es besteht kein Zusammenhang zwischen dem expliziten Erklärungsbedarf und der aktuellen Version der untersuchten App.	RQ1
$H10_0$	Es besteht kein Zusammenhang zwischen dem impliziten Erklärungsbedarf und der aktuellen Version der untersuchten App.	RQ1
$H11_0$	Es besteht kein Zusammenhang zwischen dem expliziten Erklärungsbedarf und dem Mindestalter der untersuchten App.	RQ1
$H12_0$	Es besteht kein Zusammenhang zwischen dem impliziten Erklärungsbedarf und. dem Mindestalter der untersuchten App	RQ1
$H13_0$	Es besteht kein Zusammenhang zwischen dem expliziten Erklärungsbedarf und der Sterne Bewertung der untersuchten App.	RQ2
$H14_0$	Es besteht kein Zusammenhang zwischen dem impliziten Erklärungsbedarf und der Sterne Bewertung der untersuchten App.	RQ2
$H15_0$	Es besteht kein Zusammenhang zwischen dem expliziten Erklärungsbedarf und der Anzahl an Reviews der untersuchten App.	RQ2
$H16_0$	Es besteht kein Zusammenhang zwischen dem impliziten Erklärungsbedarf und der Anzahl an Reviews der untersuchten App.	RQ2
$H17_0$	Es besteht kein Zusammenhang zwischen dem expliziten Erklärungsbedarf und der Anzahl an Downloads der untersuchten App.	RQ2
$H18_0$	Es besteht kein Zusammenhang zwischen dem impliziten Erklärungsbedarf und der Anzahl an Downloads der untersuchten App.	RQ2

Tabelle A.1: Hypothesenübersicht für die Aufteilung in expliziten oder impliziten Erklärungsbedarf

Hypothese	**Ergebnis**	**P-Wert**	**Verfahren**	**Interpretation**	**Bonferroni**
$H1_0$	0.1399	$2.3046*10^{-12}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H2_0$	0.1413	$1.0956*10^{-12}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H3_0$	0.0805	$9.6233*10^{-8}$	Cramer‘s V	Kein Zusammenhang	Verwerfen
$H4_0$	0.0624	$3.5720*10^{-5}$	Cramer‘s V	Kein Zusammenhang	Verwerfen
$H5_0$	0.0197	0.1402	Eta	Kein Zusammenhang	Nicht verwerfen
$H6_0$	0.0126	0.1122	Eta	Kein Zusammenhang	Nicht verwerfen
$H7_0$	0.0555	0.0396	Cramer‘s V	Kein Zusammenhang	Verwerfen
$H8_0$	0.0396	0.1420	Cramer‘s V	Kein Zusammenhang	Nicht verwerfen
$H9_0$	0.2921	$1.0066*10^{-35}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H10_0$	0.2512	$3.2880*10^{-20}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H11_0$	0.0786	0.000004	Cramer‘s V	Kein Zusammenhang	Verwerfen
$H12_0$	0.0735	0.00002	Cramer‘s V	Kein Zusammenhang	Verwerfen
$H13_0$	0.2996	$1.2961*10^{-32}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H14_0$	0.2562	$5.8841*10^{-17}$	Cramer‘s V	Schwacher Zusammenhang	Verwerfen
$H15_0$	0.0910	0.3016	Eta	Kein Zusammenhang	Nicht verwerfen
$H16_0$	0.0658	0.2566	Eta	Kein Zusammenhang	Nicht verwerfen
$H17_0$	0.0105	0.1024	Eta	Kein Zusammenhang	Nicht verwerfen
$H18_0$	0.0160	0.1263	Eta	Kein Zusammenhang	Nicht verwerfen

Tabelle A.2: Ergebnis der Korrelationsanalysen

Literaturverzeichnis

[1] H. Akoglu. User's guide to correlation coefficients. *Turkish Journal of Emergency Medicine*, 18(3):91–93, 2018.

[2] K. Ali Abd Al-Hameed. Spearman's correlation coefficient in statistical analysis. *International Journal of Nonlinear Analysis and Applications*, 13(1):3249–3255, 2022.

[3] R. Bender, A. Ziegler, and S. Lange. Logistische regression. *DMW-Deutsche Medizinische Wochenschrift*, 132(S 01):e33–e35, 2007.

[4] H. Best and C. Wolf. *Logistische Regression*, pages 827–854. VS Verlag für Sozialwissenschaften, Wiesbaden, 2010.

[5] M. Biswas, M. H. Tania, M. S. Kaiser, R. Kabir, M. Mahmud, and A. A. Kemal. Accu3rate: A mobile health application rating scale based on user reviews. *PloS one*, 16(12):e0258050, 2021.

[6] J. Bohnstedt. Untersuchung des einflusses von domänenwissen auf den erklärungsbedarf der nutzenden von software. *Bachelorarbeit, Gottfried Wilhelm Leibniz Universität Hannover, FG Software Engineering*, 2024.

[7] J. Bortz. *Multiple Korrelation und Regression*, pages 550–577. Springer Berlin Heidelberg, Berlin, Heidelberg, 1989.

[8] L. Chazette, W. Brunotte, and T. Speith. Exploring explainability: A definition, a model, and a knowledge catalogue. In *2021 IEEE 29th International Requirements Engineering Conference (RE)*, pages 197–208, 2021.

[9] L. Chazette and K. Schneider. Explainability as a non-functional requirement: challenges and recommendations. *Requirements Engineering*, 25(4):493–514, 2020.

[10] J. Droste, H. Deters, M. Obaidi, and K. Schneider. Explanations in everyday software systems: Towards a taxonomy for explainability needs. In *2024 IEEE 32nd International Requirements Engineering Conference (RE)*, pages 55–66, 2024.

[11] E. C. Fieller and E. S. Pearson. Tests for rank correlation coefficients: Ii. *Biometrika*, 48(1/2):29–40, 1961.

[12] K. Frank. *Critical online reasoning: Validierung neu entwickelter Aufgaben zur Erfassung und Förderung des kritischen Umgangs mit Online-Medien*, volume 1. Springer, 2022.

[13] J. A. Harris and A. E. Treloar. On a limitation in the applicability of the contingency coefficient. *Journal of the American Statistical Association*, 22(160):460–472, 1927.

[14] N. Jösten. Clusteranalyse zwischen stimmung und erklärbarkeitsanforderungen von nutzern. *Bachelorarbeit, Gottfried Wilhelm Leibniz Universität Hannover, FG Software Engineering*, 2024.

[15] D. Kupczyk. Automatisierte detektion von erklärungsbedarf in nutzerfeedback zu software. *Masterarbeit, Gottfried Wilhelm Leibniz Universität Hannover, FG Software Engineering*, 2023.

[16] T. Kurtz. Entwicklung einer software zur extrahierung und analyse von reviews aus app stores. *Bachelorarbeit, Gottfried Wilhelm Leibniz Universität Hannover, FG Software Engineering*, 2023.

[17] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, 1977.

[18] X. Lu, Z. Chen, X. Liu, H. Li, T. Xie, and Q. Mei. Prado: Predicting app adoption by learning the correlation between developer-controllable properties and user behaviors. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3), sep 2017.

[19] M. Sadeghi, V. Klös, and A. Vogelsang. Cases for explainable software systems: Characteristics and examples. In *2021 IEEE 29th International Requirements Engineering Conference Workshops (REW)*, pages 181–187, 2021.

[20] S. Stange, H. Buschmeier, T. Hassan, C. Ritter, and S. Kopp. Towards self-explaining social robots. verbal explanation strategies for a needs-based architecture. 2019.

[21] A. Steland. *Deskriptive und explorative Statistik*, pages 1–74. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016.

[22] A. Steland. Schließende statistik. In *Basiswissen Statistik: Kompaktkurs für Anwender aus Wirtschaft, Informatik und Technik*, pages 177–251, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[23] M. Unterbusch, M. Sadeghi, J. Fischbach, M. Obaidi, and A. Vogelsang. Explanation needs in app reviews: Taxonomy and automated detection. In *2023 IEEE 31st International Requirements Engineering Conference Workshops (REW)*, pages 102–111, 2023.

[24] E. W. Weisstein. *"Bonferroni Correction.From https://mathworld.wolfram.com- -A Wolfram Web Resource.* https://mathworld.wolfram.com/BonferroniCorrection.html.

[25] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, A. Wesslén, et al. *Experimentation in software engineering*, volume 236. Springer, 2012.