



Bachelorarbeit

Conversational Interfaces Design und Entwicklung eines Chatbots

Von: Lina Peters
Eingereicht: April 2017
Hochschule: HTWK Leipzig
Fakultät: Informatik, Mathematik und Naturwissenschaften
Studiengang: Medieninformatik (B.Sc.)
1. Gutachter: Prof. Dr. M. Frank
2. Gutachter: Prof. Dr. M. Liesching

Zusammenfassung

Die vorliegende Arbeit beschreibt den Entwicklungs- und Designprozess eines Chatbots mit dem Ziel über theoretische Ansätze Regeln abzuleiten, die diesen Prozess unterstützen sollen. Dafür werden Theorien wie Pangaros Konversationsmodell, die Grice Maximen und Reeves und Nass „The Media Equation“ betrachtet und ausgewertet. Um die Eignung der entwickelten Regeln zu prüfen, wird im Rahmen dieser Arbeit ein Facebook Chatbot entwickelt, der Rezepte suchen kann. Dass die erarbeiteten Regeln wirksam sind, bestätigt sich über das Testen des entwickelten Chatbots und die damit verbundene Umfrage.

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich die hier vorgelegte Bachelorarbeit selbstständig und ohne fremde Hilfe abgefasst habe. Diese Bachelorarbeit wurde nicht anderweitig als Prüfungsleistung verwendet und wurde noch nicht veröffentlicht. Ich habe keine anderen Quellen und Hilfsmittel als die angegebenen benutzt. Diesen Werken wörtlich oder sinngemäß entnommene Stellen habe ich als solche gekennzeichnet.

Leipzig, den 4. April 2017

Unterschrift

Inhaltsverzeichnis

1	Einleitung	1
1.1	Beschreibung des Themas	1
1.2	Ziel der Arbeit	1
1.3	Aufbau	2
2	Theorie- und Technikgrundlagen	3
2.1	Der Begriff Chatbot	3
2.2	Beschreibung des Use-Case: Rezeptsuche	3
2.3	Messenger Plattformen	4
2.4	Künstliche Intelligenz	6
2.5	Systemarchitektur	7
2.5.1	Komponentensicht	7
2.5.2	User Interface und 3rd-Party-Services des Rezepte-Chatbots	8
2.5.2.1	Facebook Messenger API	9
2.5.2.2	wit.ai	11
2.5.2.3	Spoonacular	14
2.5.3	Verteilungssicht	15
3	Konversationstheorie und Persönlichkeit	17
3.1	Der Kommunikationsbegriff und zugehörige Modelle	17
3.2	The Media Equation	19
3.3	Mediale Persönlichkeit	21
4	Entwurf und Umsetzung des Chatbots	23
4.1	Das Konversationsdiagramm	23
4.2	Umsetzung von Pangaros Konversationsmodell	25
4.2.1	Kontext	25
4.2.2	Sprache	26
4.2.3	Austausch	27
4.2.4	Einverständnis & Transaktion	28
4.3	Umsetzung von Grice Maximen	29
4.3.1	Quantität	29
4.3.2	Qualität	30
4.3.3	Relation	31
4.3.4	Modalität	31
4.4	Umsetzung von Reeves und Nass „The Media Equation“	32
4.5	Text-Design und Chatbot-Persönlichkeit	34
4.5.1	Chatbot-Persönlichkeit	34
4.5.2	Entwicklungsprozess der Persönlichkeit und Texte	35
4.5.3	Verwendung von Emojis	37

4.5.4	Verwendung von Menüs und Buttons	38
5	Umfrage	39
5.1	Konzeption	39
5.2	Rahmenbedingungen	40
5.3	Auswertung	40
5.3.1	1. Teil: Chatbot-Persönlichkeit	40
5.3.2	2. Teil: Use-Case und Zielgruppe	42
5.3.3	3. Teil: Zukünftige Nutzung des Chatbots	43
5.3.4	4. Teil: Freies Feedback	44
6	Fazit	45
	Abbildungsverzeichnis	47
	Abkürzungsverzeichnis	48
	Literaturverzeichnis	49
	Bildquellen	50

1 Einleitung

There's an interesting leading indicator of when a new industry is about to go mainstream: the limiting factor shifts from technology to design. [1]

1.1 Beschreibung des Themas

Seit Joseph Weizenbaum 1966 das Programm ELIZA veröffentlichte, kommt die Idee von menschenähnlichen Computern und Programmen immer wieder auf, sowohl in der Wissenschaft als auch in diversen Science-Fiction-Filmen. Die Popularität von ELIZA, HAL 9000¹ und Samantha² implizieren, dass der Mensch interessiert daran ist, nicht nur mit anderen Menschen, sondern auch mit Computern sozial zu interagieren.

Mit den Fortschritten im Bereich der künstlichen Intelligenz und mittlerweile weit verbreitenden Programmen wie Siri³ scheinen fiktive Chatbots wie Samantha gar nicht mehr so fern. Auch für große Firmen wie Facebook scheint diese Entwicklung allgegenwärtig, weswegen sie 2015 das Startup wit.ai aufkauften und ein Jahr später die Facebook Messenger Plattform für Chatbots öffnete. Auf diversen Blog-Einträgen werden Chatbots schon jetzt als revolutionär und als ein Ersatz für mobile Apps gehandelt, genau wie Apps damals für Webseiten.

Wie Phil Livin, der Mitbegründer von Evernote⁴, in seinem Blog-Post anmerkt (Zitat oben), liegt das Problem bei Chatbots nicht mehr bei technologischen Einschränkungen, sondern eher bei noch ungeklärten Designfragen, die vor allem sozialer Natur sind.

1.2 Ziel der Arbeit

Ziel dieser Arbeit soll es sein diese Design- und Entwicklungsfragen durch einen wissenschaftlich-theoretischen Ansatz zu klären. Dabei werden vor allem bereits bestehende Studien im Feld der Interaktion zwischen Mensch und Computer und zwischen Mensch und Mensch herangezogen.

¹ AI Computer aus dem Film 2001: A Space Odyssey
² AI Betriebssystem aus dem Film Her
³ Intelligenter sprachgesteuerter Assistent von Apple
⁴ Organisationstool für Notizen

1.3 Aufbau

Kapitel 2 soll den Einstieg in das Thema durch Begriffsdefinitionen und Erklärungen der Grundbestandteile eines Chatbots erleichtern. Dabei werden zum einen die einzelnen Bestandteile der Architektur des Chatbots erklärt. Zum anderen soll durch die Erklärung von Messenger Plattformen die Motivation dieser Arbeit weiterhin unterstützt werden. Des Weiteren wird in diesem Kapitel der User-Case erläutert, der im Rahmen dieser Arbeit bearbeitet wurde.

Das dritte Kapitel dieser Arbeit widmet sich theoretischen Fragestellungen aus der Medientheorie, der Soziologie und der Psychologie. Mit Hilfe von diversen Fallstudien wird vor allem die Frage zur Kommunikation zwischen Mensch und Maschine beleuchtet. Außerdem wird die Theorie der fünf Persönlichkeitsmerkmale erklärt und wieso diese auch bei medialen Persönlichkeiten anzuwenden sind.

Die Theorien, die in Kapitel 3 erklärt werden, führen dann in Kapitel 4 zu konkreten Regelwerken, die bei der Konzeption und dem Design eines Chatbots zu beachten sind.

Im Rahmen der Bachelorarbeit wurden die Testnutzer des Chatbots gebeten an einer Umfrage teilzunehmen. Diese Umfrageergebnisse werden in Kapitel 5 ausgewertet und treffen eine Aussage über die Gültigkeit der in Kapitel 4 erstellten Regelwerke.

Zum Schluss werden in Kapitel 6 die Ergebnisse dieser Arbeit kritisch bewertet und ein zukunftsgerichtetes Fazit gezogen, ob Chatbots eine Alternative zu mobilen Apps sind.

2 Theorie- und Technikgrundlagen

In diesem Kapitel werden einige Grundlagenbegriffe definiert, die im weiteren Verlauf der Arbeit häufig genutzt werden. Um einen Chatbot zu entwickeln bedarf es außerdem einiger Technologien und Werkzeuge, die im Nachfolgenden einzeln beleuchtet werden.

2.1 Der Begriff Chatbot

Der Begriff Chatbot setzt sich aus den englischen Begriffen „Chat“¹ und „Bot“² zusammen. Der Begriff „Bot“ kann in diesem Zusammenhang sinngemäß mit Agent übersetzt werden.

Chatbots ermöglichen Interaktionen mit einem Computer bzw. einer Anwendung über die Eingabe von natürlicher Sprache, wobei es egal ist, ob diese Eingabe sprach- oder textbasiert ist [2, S. 21].

Das Potenzial eines Chatbots liegt in der Natürlichkeit der Interaktion. Dies bezieht sich auf die kognitive Zugänglichkeit und die Art der Kommunikation [2, S. 21]. So können Chatbots den Umgang mit Computern erleichtern.

2.2 Beschreibung des Use-Case: Rezeptsuche

Der Chatbot, der im Rahmen dieser Arbeit entwickelt wird, kann dem Nutzer Koch- und Backrezepte suchen. Dieser Use-Case ist sehr gut als Chatbot umzusetzen, da die Interaktion mit einer äquivalenten mobilen App oftmals textueller Natur ist. Beispielsweise läuft die Rezeptsuche in einer App über ein Sucheingabefeld. Neben der reinen Rezeptsuche kann der Chatbot in einer Datenbank auch Unverträglichkeiten und Ernährungsformen des Nutzers sichern und auf diese bei der Suche zurückgreifen. Dadurch soll die Rezeptsuche für Menschen leichter gemacht werden, die beispielsweise vegetarisch leben oder Gluten intolerant sind. Sie müssen dies dem Chatbot nur einmal mitteilen und nicht bei jeder Suche erneut eingeben.

¹ dt. Plauderei
² kurz für Robot, dt. Roboter

2.3 Messenger Plattformen

Natürliche Sprache wird im digitalen Zeitalter bereits täglich verwendet. Vor allem in den gängigen Messenger³-Anwendungen kommunizieren Menschen fast ausschließlich über natürliche Sprache miteinander. Messenger-Anwendungen oder Messenger-Apps im mobilen Bereich sind Instant-Messaging-Dienste (auch Chat-Dienste), welche es einem Nutzer möglich machen über Sofortnachrichten mit einem oder mehreren anderen Nutzern zu kommunizieren. Neben Text können die Nachrichten in vielen Anwendungen Audio-, Video- oder Bilddateien enthalten.

Zu den am meisten genutzten Diensten weltweit gehören [3]:

- WhatsApp (Europa)
- Facebook Messenger (Nordamerika)
- WeChat (China)
- Line (Japan)

Chat-Dienste können aber nicht nur zu privaten Zwecken genutzt werden, sondern auch Firmen können so mit ihren Kunden kommunizieren.

Der Vorteil für Firmen, mit Kunden über Messenger-Anwendungen in Kontakt zu treten, liegt in den Nutzungszahlen und den Nutzungszeiten dieser Anwendungen. Vor allem in der mobilen Welt gehören Messenger-Anwendungen zu den meist heruntergeladenen Apps weltweit.

WhatsApp beispielsweise ist in Deutschland im iOS App Store⁴ die meist-heruntergeladene App und im Google Play Store⁵ die am zweitmeisten heruntergeladene App. Dasselbe gilt für den Facebook Messenger in den USA [4].

³ dt. Nachrichtendienst
⁴ digitale Vertriebsplattform für Anwendungssoftware auf mobilen Apple-Geräten
⁵ digitale Vertriebsplattform für Anwendungssoftware auf Android-Geräten

Day in Life of a Mobile User, 2016

	Average # Apps Installed on Device*	Average Number of Apps Used Daily	Average Number of Apps Accounting for 80%+ of App Usage	Time Spent on Phone (per Day)	Most Commonly Used Apps
USA	37	12	3	5 Hours	Facebook Chrome YouTube
Worldwide	33	12	3	4 Hours	Facebook WhatsApp Chrome

Abbildung 1: Am meisten genutzten Apps in den USA und weltweit

Vor allem die aktiven Nutzerzahlen, welche ausschlaggebend für den Erfolg einer App sind, sind bei den Messenger-Anwendungen besonders hoch.

Monthly Active Users on Select Social Networks and Messengers, Global, 2011 – 2015

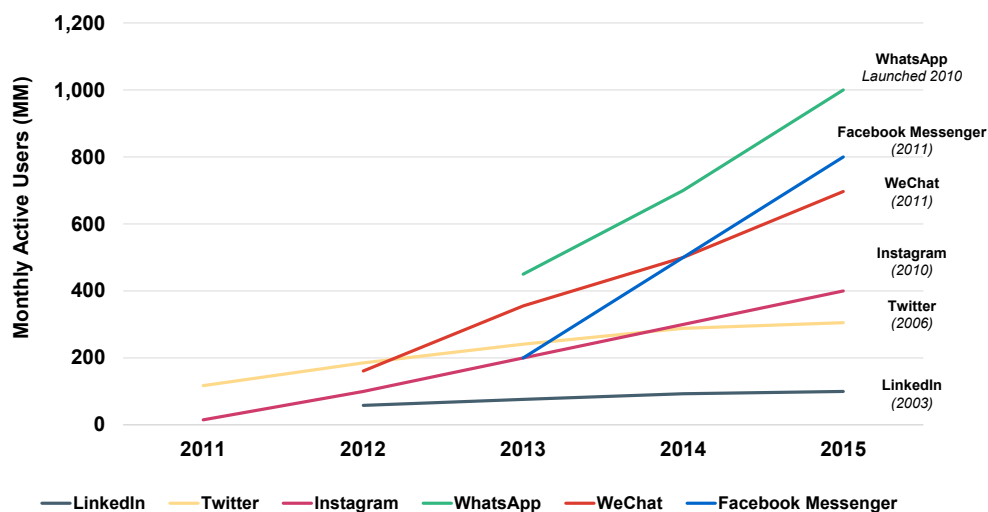


Abbildung 2: Aktive Nutzerzahlen für WhatsApp und den Facebook Messenger

Anstatt also eine eigene App zu entwickeln, könnte es zukünftig für bestimmte Use-Cases Sinn machen einen Chatbot zu entwickeln, der in einer oder mehreren Messenger-Anwendung(en) zur Verfügung gestellt wird.

Im Rahmen dieser Arbeit wird ausschließlich die Social Media Plattform Facebook betrachtet, da Facebook mit ca. 1,23 Milliarden täglich aktiven Usern die größte Plattform weltweit ist. Den Facebook Messenger nutzen monatlich über 900 Millionen Menschen zu Kommunikationszwecken [5]. So ist davon auszugehen, dass ein Chatbot, der im Facebook Messenger lebt, heutzutage die größte Reichweite hat.

WhatsApp ist von dieser Betrachtung auszuschließen, da WhatsApp nach heutigem Stand keine Möglichkeit bietet einen Chatbot über den Dienst anzubieten.

2.4 Künstliche Intelligenz

KI ist ein Teilgebiet der Informatik und kann allgemein als der Versuch, Intelligenz nachzubilden, interpretiert werden. Vor allem die Adaptier- und Lernfähigkeit des Menschen sollen durch eine Maschine imitiert werden.

Natural Language Processing⁶ ist ein Teilgebiet der KI. Durch Algorithmen soll ein Computer die natürliche Sprache, die der Mensch zur Kommunikation verwendet, erlernen und interpretieren können.

NLP ist für die Entwicklung eines Chatbots essentiell. Ohne NLP müsste der Nutzer wie in einer Konsole oder früheren Computer-Anwendungen die genauen Befehle kennen, um mit dem Chatbot zu kommunizieren. Nur durch natürliche Sprache und das Verständnis dieser kann bei der Kommunikation mit einem Chatbot eine natürliche und menschenähnliche Unterhaltung entstehen.

In den letzten Jahren entwickelten viele Firmen NLP-Tools. Die bekanntesten sind:

- LUIS (Microsoft)
- wit.ai (Facebook)
- API.ai (Google)
- Watson (IBM)
- Lex (Amazon)

Über NLP-Tools kann natürliche Sprache in strukturierte Informationen umgewandelt werden, die ein Computer verarbeiten kann. Die meisten Tools unterteilen natürliche Sprache in Informationen über Intention, Parameter, Kontext und Sitzung [6].

⁶ dt. Verarbeitung von natürlicher Sprache

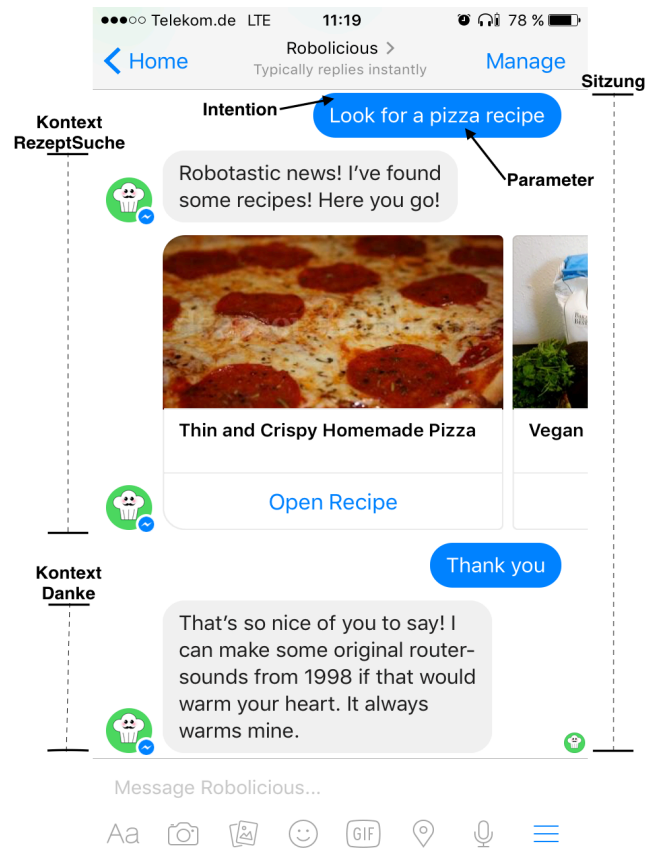


Abbildung 3: Ausschnitt aus der Konversation mit dem Rezepte-Chatbot. Intention, Parameter, Kontext und Sitzung markiert

2.5 Systemarchitektur

Die System-Architektur eines Chatbots ist über die Komponenten- und die Verteilungssicht beschreibbar. Die Komponentensicht verschafft einen Überblick über die allgemeinen Komponenten eines Chatbots-Systems, welches im Kern unabhängig von dem umgesetzten Use-Case gleich aussieht. Über die Verteilungssicht wird ersichtlich, welche Tools zur Bereitstellung der Komponenten genutzt wurden.

2.5.1 Komponentensicht

Das Chatbot-System besteht aus dem Chatbot-Backend, einer Datenbank, einem User Interface und vom Use-Case abhängige Webservices, die Informationen bereitstellen.

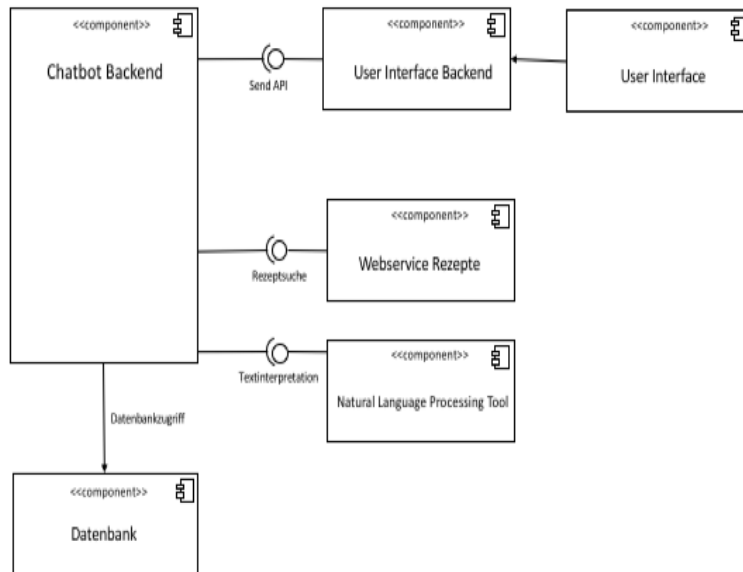


Abbildung 4: UML-Komponentendiagramm des Chatbot-Systems

Wie in dem Diagramm zu sehen ist, kommuniziert das Chatbot-Backend über eine API mit dem Backend des User-Interface. Das User-Interface wird bei Chatbots im Normalfall von einer Messengerplattform wie Facebook bereitgestellt. Diese bieten dann eine oder mehrere Schnittstellen an, die das Chatbot-Backend konsumieren kann. Dasselbe gilt für die Webservices, die vom Chatbot-Backend für bestimmte Informationen – hier Rezepte – aufgerufen werden. Wie in Kapitel 2.4 „Künstliche Intelligenz“ erklärt wurde ist die Einbindung eines NLP-Tools wichtig, um natürliche Sprache auswerten zu können. Das NLP-Tool kann wie das User-Interface über eine Schnittstelle genutzt werden.

Um Nutzerdaten und die Chatbot-Texte zu sichern und bereitzustellen, benötigt das System außerdem eine Datenbank, mit welcher sich das Chatbot-Backend verbinden kann.

2.5.2 User Interface und 3rd-Party-Services des Rezepte-Chatbots

Der Rezepte-Chatbot ist über den Facebook Messenger verfügbar, nutzt zur Auswertung der natürlichen Sprache wit.ai und bedient sich des Webservice Spoonacular zur Rezeptsuche. Diese Komponenten und die Kommunikation mit diesen werden in den nachfolgenden Kapiteln näher beschrieben.

2.5.2.1 Facebook Messenger API

Im April 2016 stellte Facebook Inc. auf ihrer Entwicklerkonferenz F8 in San Jose die neue Facebook Messenger Plattform vor. „Bots for the Messenger Platform“⁷ soll es vor allem Firmen möglich machen über Chatbots mit ihren Kunden zu interagieren. Der Facebook Messenger ist Facebooks Instant-Messaging-Dienst, der über die Facebook-Website und die Facebook Messenger App nutzbar ist.

Die „Senden“- und „Empfangen“-API der Messenger-Plattform ist für jeden zugänglich, der ein Facebook-Profil besitzt. Des Weiteren wird eine Facebook-Seite⁸ und eine Facebook-App⁹ benötigt, um die API benutzen zu können. Eine Facebook-App ist nicht zu verwechseln mit der mobilen App, über die Facebook selbst genutzt werden kann. Vielmehr lebt eine Facebook-App als Applikation innerhalb der Facebook-Umgebung. Neben Chatbots können beispielsweise auch Spiele auf Facebook als App angeboten werden. Um eine Facebook-App für jeden über eine Facebook-Seite zugänglich machen zu können, muss die App von Facebook freigegeben werden. Das geschieht über eine Begutachtung der App durch einen Facebook-Mitarbeiter.

Über die Produkteinstellungen der Facebook-App lässt sich der Messenger als Produkt zur App hinzufügen, und über das Messenger-Produkt lässt sich wiederum ein Webhook aufsetzen. Webhooks, die auch „Reverse APIs“¹⁰ genannt werden, werden typischerweise dazu genutzt bestimmte Informationen, die ein Webservice bereitstellt, zu abonnieren. Im Falle des Messenger-Webhooks wird ein Abonnement für Updates der zur App gehörigen Facebook-Seite eingerichtet. Für das Aufsetzen des Webhooks wird eine Callback-URL¹¹ und ein Token benötigt. Diese werden auf der Facebook-Entwicklerwebseite eingegeben. Nach der Eingabe verifiziert Facebook mit einer http-GET-Anfrage, dass die Callback-URL korrekt ist. Wenn das der Fall ist, sendet Facebook dann Updates zu auswählbaren Events auf einer Facebook-Seite.

Sobald ein Facebook-Nutzer eine Nachricht an eine Facebook-Seite schreibt, die an eine Facebook-App mit Messenger-Produkt gekoppelt ist, werden die Nachrichteninformationen als JSON-Objekt an die Callback-URL des Webhooks als http-POST-Anfrage weitergeleitet.

⁷ dt. Bots für die Messenger Plattform
⁸ ein Profil für Firmen, Künstler, Produkte, etc.
⁹ Anwendung wie z.B. ein Spiel
¹⁰ dt. umgekehrte APIs
¹¹ Rückruf-URL

```

{ sender: { id: '924108951050420' },
  recipient: { id: '206812626382006' },
  timestamp: 1485250999651,
  message: {
    mid: 'mid.1485250999651:80eb748c53',
    seq: 27774,
    text: 'Tell me a joke'
  }
}

```

Auf diese Nachricht antwortet das Chatbot-Backend dann dadurch, dass ein JSON-Objekt mit der erhaltenen Sender-ID und der zu sendenden Textnachricht erstellt wird. Das JSON-Objekt wird als http-POST-Anfrage an die Messenger-API gesendet, die über die Sender-ID dem korrekten Nutzer zugewiesen wird.

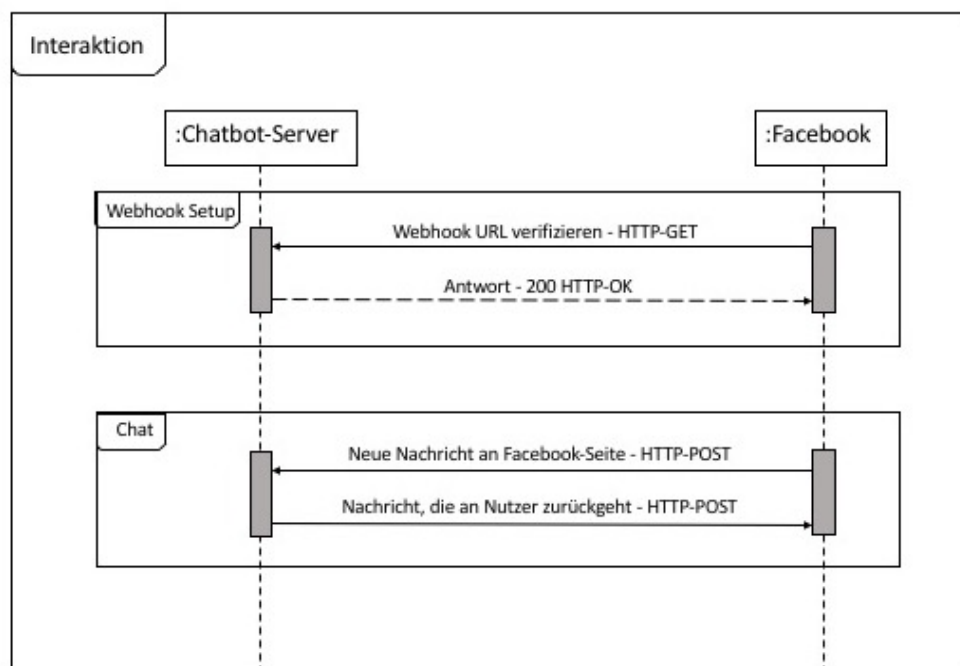


Abbildung 5: UML-Sequenzdiagramm der Interaktion zwischen dem Chatbot-Server und Facebook

Neben reinen Textnachrichten können über die Messenger-API auch sogenannte Attachments¹² an Nutzer verschickt werden. Es können beispielsweise Bilder oder Links versendet werden. Ebenfalls möglich ist es eine Textnachricht mit Buttons¹³ zu versehen. Auf diese Weise, kann die Unterhaltung teilweise über typische GUI-Elemente gesteuert werden.

¹² dt. Anhänge
¹³ dt. Knöpfe

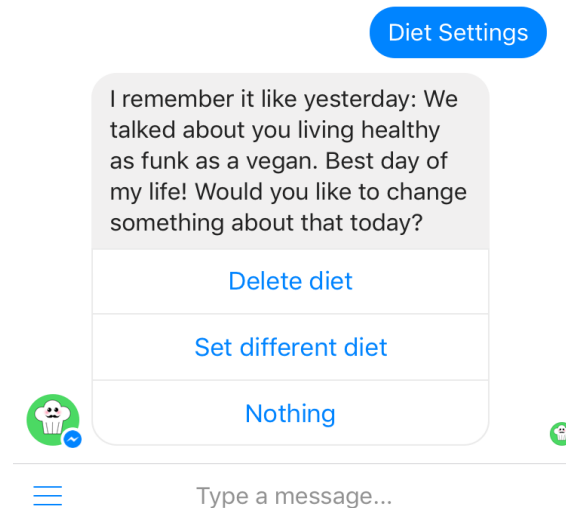


Abbildung 6: Nachricht des Rezepte-Chatbots mit Buttons

Um passende Antworten auf die Nachricht eines Nutzers geben zu können, muss der Textteil des JSON-Objekts korrekt interpretiert werden. Hierfür wird ein Natural-Language-Processing-Tool zu verwendet.

2.5.2.2 wit.ai

wit.ai ist ein kostenfreies NLP-Tool, welches eine http-API anbietet. Auch wit.ai wandelt natürliche Sprache in strukturierte Informationen um. Dabei unterscheidet wit.ai zwischen Intentionen und Entitäten, wobei Entitäten mit den zuvor beschriebenen Parametern gleichzusetzen sind. wit.ai bietet schon vordefinierte Entitäten wie zum Beispiel „wit/datetime“, durch welche Daten und Zeiten im Text erkannt werden können, oder „wit/location“, durch welche Orte im Text erkannt werden können. Neben den vordefinierten Entitäten ist es möglich eigene Entitäten zu definieren. Dabei unterscheiden sich Entitäten in ihrer Suchstrategie.

Es gibt drei Arten von Suchstrategien, die wit.ai unterscheidet: Merkmal, Schlagwort und Freitext. Die Strategie *Merkmal* wird dann verwendet, wenn eine Intention oder eine Empfindung interpretiert werden soll. In wit.ai sind Intentionen also nichts anderes als Entitäten, die nach der Merkmal-Suchstrategie analysiert werden. Die Suchstrategie *Merkmal* zeichnet sich dadurch aus, dass weder einzelne Wörter noch spezielle Aussagen im Text analysiert werden, sondern der Text als Ganzes analysiert wird. Wenn die Suchstrategie *Schlagwort* bei Entitäten angewendet wird, wird der Text nach vordefinierten Schlagwörtern durchsucht. Diese Schlagwörter müssen in der Entität definiert sein. Es ist jedoch möglich einem Schlagwort Synonyme zuzuweisen. *Freitext* bedeutet im Gegensatz zu Schlagwörtern, dass nach einem Substring im Text gesucht werden muss, der jedoch nicht zu einer vordefinierten

ten Liste an Schlagwörtern gehört. Beispielhaft wären hier Personennamen. Personennamen lassen sich nicht in einer vordefinierten Liste erfassen. Jedoch können Personennamen aus dem Kontext sehr sicher erkannt werden.

Damit wit.ai genutzt werden kann, muss zunächst eine App auf der Plattform angelegt werden. In dieser App können dann Intentionen und Entitäten definiert werden. Diese können bereits über die Oberfläche von wit.ai trainiert werden.

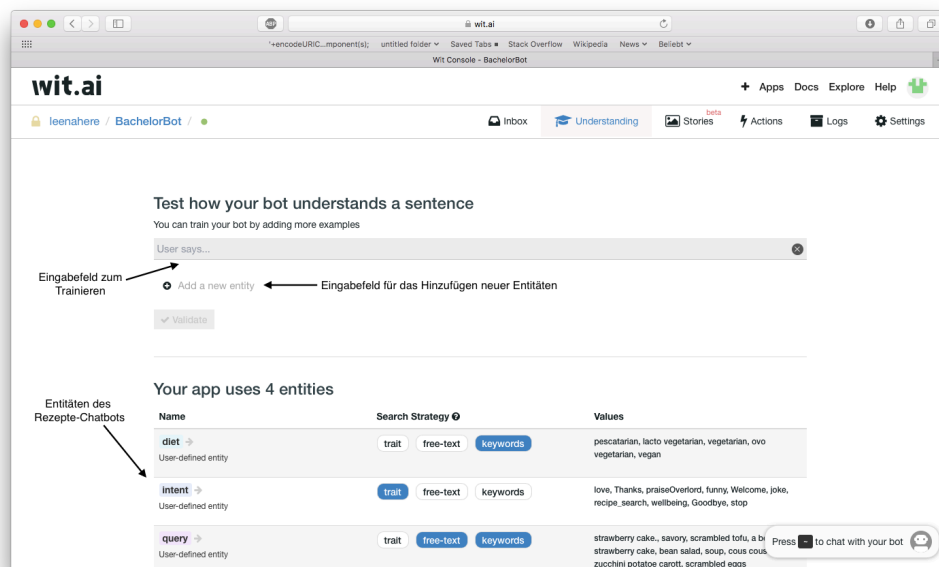


Abbildung 7: Web-Oberfläche von wit.ai – Understanding

Über die Inbox-Funktion werden alle Texte gesammelt, die über die Schnittstelle geschickt werden. In der Oberfläche wird übersichtlich dargestellt, wie wit.ai die jeweiligen Texte interpretiert hat.

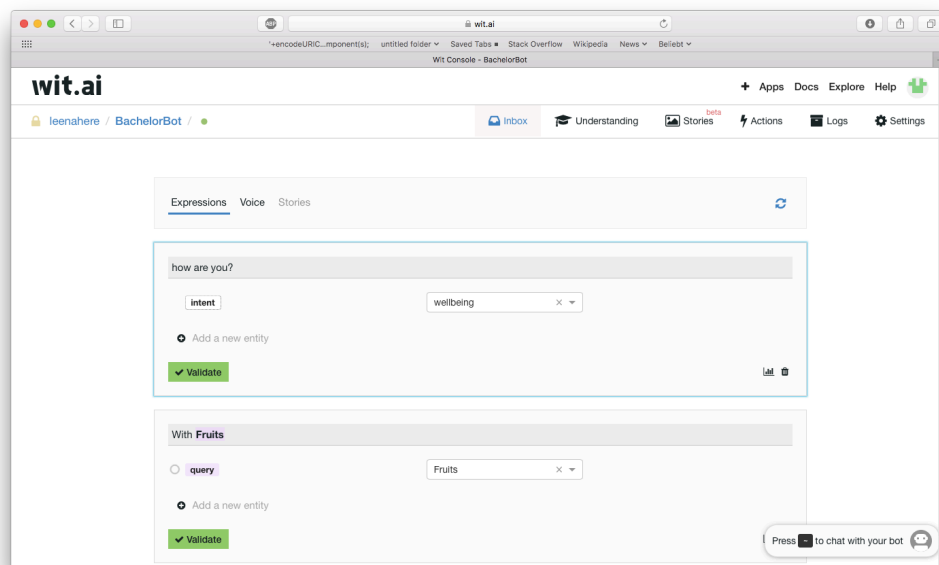


Abbildung 8: Web-Oberfläche von wit.ai – Inbox

Über die Inbox kann man die Interpretation der Texte auch ändern, um Fehlinterpretationen zu beheben.

Angebunden wird die erstellte wit.ai-App ebenfalls mit Hilfe eines zur App zugehörigen Token über http-Anfragen, wobei ein Text als http-GET-Anfrage an wit.ai gesendet wird.

```
// wit node module function call to send user text to wit
wit.message(text, {})
.then((data) => {
  // Stringify data to display it in console
  var datawit = JSON.stringify(data);
  console.log(datawit);
});
```

Als Antwort erhält man ein JSON-Objekt, welches die Nachricht selbst, die erkannten Entitäten und einen Sicherheitswert zurückgibt.

```
{
  "msg_id": "61a69373-ebce-4704-86a4-2cd35c76bc14",
  "_text": "look for potato salad",
  "entities": {
    "query": [
      {
        "confidence": 0.9974262695665073,
        "type": "value",
        "value": "potato salad"
      }
    ],
    "intent": [
      {
        "confidence": 0.9995659092532362,
        "value": "recipe_search"
      }
    ]
  }
}
```

Der Sicherheitswert drückt dabei aus, wie sicher sich wit.ai mit den erkannten Entitäten und Werten ist.

2.5.2.3 Spoonacular

Da im Rahmen dieser Arbeit ein Chatbot entwickelt wird, welcher Rezepte auf Anfrage ausspielen kann, integriert die App zusätzlich Anfragen an die Spoonacular Food¹⁴ API. Die API bietet mehr als 360.000 Rezepte und liefert außerdem noch zusätzliche Informationen wie Nahrungsanalysen, Kosten für Zutaten, Gesundheitshinweise, Witze über Essen und Trivia.

Die API ist wie wit.ai über http-Anfragen erreichbar. Das Chatbot-Backend nutzt die http-GET-Anfrage „Search Recipes“ und „Get Recipe Information“. Die „Search Recipes“-Anfrage sieht wie folgt aus.

```
// HTTP GET requests to food API via unirest node module
urlrequest = "https://spoonacular-recipe-food-nutrition-
v1.p.mashape.com/recipes/search?" + diet + "instructionsRequired=false&"
+ intolerance + "limitLicense=false&number=6&offset=0&query=" + qf;
unirest.get(urlrequest)
.header("X-Mashape-Key", SPOONAPI_TOKEN)
.header("Accept", "application/json")
.end(function (result) {
...
})
```

Als Antwort erhält man ein JSON-Objekt, welches zu jedem Rezept eine ID, einen Titel, eine Zubereitungszeit und eine Bild-URL zurückgibt.

```
"results": [
  {
    "id": 666798,
    "title": "Slow-Cooker Vegetarian Lasagna",
    "readyInMinutes": 120,
    "image": "Slow-Cooker-Vegetarian-Lasagna-666798.jpg",
    "imageUrls": [
      "Slow-Cooker-Vegetarian-Lasagna-666798.jpg"
    ]
  }
]
```

Der Link, der zum Rezept führt, muss separat über „Get Recipe Information“ abgefragt werden. Dafür benötigt man die ID des Rezepts, welche die erste GET-Anfrage geliefert hat.

¹⁴ dt. Essen

```
// HTTP GET requests to food API via unirest node module
unirest.get("https://spoonacular-recipe-food-nutrition-
v1.p.mashape.com/recipes/" + id + "/information?includeNutrition=false")
.header("X-Mashape-Key", SPOONAPI_TOKEN)
.header("Accept", "application/json")
.end(function (result) {
...

```

Auch hier wird wieder ein JSON-Objekt zurückgegeben, welches den Rezeptlink enthält.

2.5.3 Verteilungssicht

Das Chatbot-Backend läuft auf einem Server des Cloud-PaaS Heroku. Heroku ist ein Online-Web-Host, der die gesamte Infrastruktur liefert. Das Backend wird in einem virtuellen Container, die Heroku *Dynos* nennt, bereitgestellt.

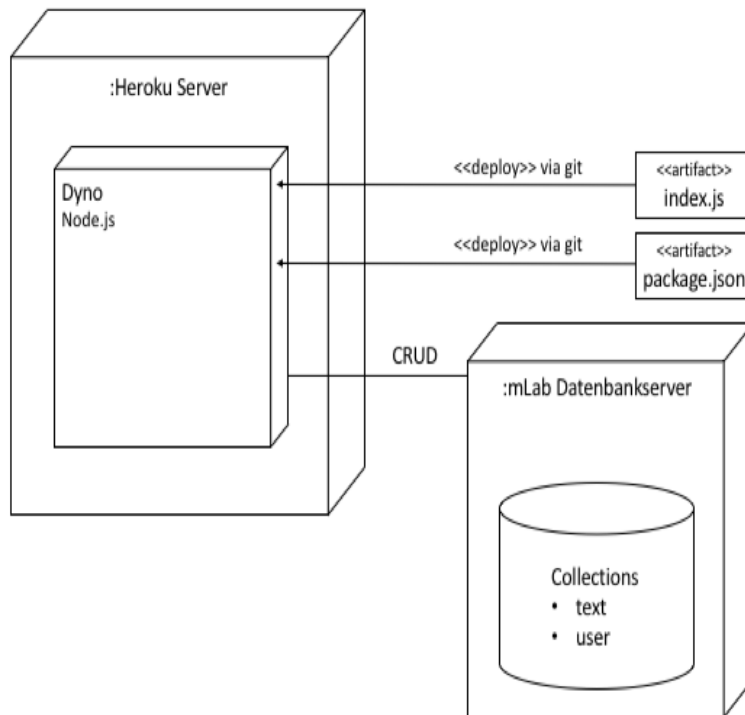


Abbildung 9: Deploymentdiagramm des Chatbot-Backend

Das Backend wurde mit Node.js umgesetzt, was eine serverseitige JavaScript-Umgebung ist, die auf Googles V8 JavaScript Engine aufgebaut ist [7]. Node.js eignet sich vor allem gut für einen Chatbot, der viele Anfragen gleichzeitig beantworten können muss, da es auf einem nicht-blockenden Input/Output-Modell basiert und ereignisgesteuert arbeitet. Der Programmcode wird über das Versionsverwaltungs-

Tool Git bereitgestellt. Wenn eine App auf Heroku erstellt wird, wird automatisch ein Git-Remote für das lokale Git-Repository der App angelegt.

Ergänzend macht es Sinn eine Datenbank hinzuzufügen. Zum einen sollten die Texte des Chatbots getrennt vom Programmcode gesichert werden, da sie so einfacher zu pflegen sind. Zum anderen ist es in manchen Fällen sinnvoll Nutzerdaten zu sichern, mit denen der Bot individuell mit dem Nutzer interagieren kann. In dieser Arbeit wird eine MongoDB-Datenbank genutzt, die auf dem Cloud-Datenbank-Service mLab bereitgestellt wird. Die Datenbank enthält zwei Collections, eine für die Nutzerdaten und eine für die Bot-Texte, in welchen folgende Daten gesichert werden:

```
// Bot-Texte
{
  "_id": "id",
  "Name": "name",
  "A/B-Test-Info": "a-b-info",
  "Text": "text"
}
// Nutzer-Daten
{
  "_id": "id",
  "First Name": "first_name",
  "Last Name": "last_name",
  "Gender": "gender",
  "Facebook ID": "facebookID",
  "Diet": "diet",
  "Intolerance": "intolerance"
}
```

Das Chatbot-Backend nutzt die gängigen Datenbankzugriffe CRUD – Create, Read, Update, Delete. Über das Node.js-Modul „mongodb“ können diese Datenbankzugriffe im Programmcode realisiert werden. Hierfür wird als erstes ein Verbindungs-Pool erstellt.

```
//Open MongoDB Connection Pool
MongoClient.connect(dbURL, function(err, database) {
  if (err) throw err;
  db = database;
  app.listen((process.env.PORT || 3000));
});
```

Der Chatbot kann dann beispielsweise nach einem Nutzer in der Datenbank suchen, um zu schauen, ob dieser schon einmal mit dem Chatbot kommuniziert hat.

```
// callback function to find existing user in database
function findUser (senderID, callback) {
  db.collection('user').findOne({facebookID: senderID}, function(err,
  result) {
    callback(result);
    console.log('Result of database find' + result);
  });
};
```

3 Konversationstheorie und Persönlichkeit

Bevor der Rezepte-Chatbot realisiert werden kann, werden im folgenden Kapitel einige theoretische Ansätze beschrieben, die zu betrachten sind, um erfolgsversprechende Regeln zu entwerfen, die den Entwicklungsprozess eines Chatbots nachhaltig verbessern können.

3.1 Der Kommunikationsbegriff und zugehörige Modelle

Kommunikation ist im allgemeinen Sinn der Austausch von Information oder die Übertragung solcher. Kommunikation ist eine alltägliche soziale Handlung, der erst eine übergeordnete Bedeutung zugewiesen wird, wenn sich Missverständnisse zeigen und durch Kommunikation Probleme auftreten. Um Kommunikation und ihre Probleme wissenschaftlich aufzuarbeiten, gibt es verschiedene Ansätze.

Teilweise wird neben dem Begriff Kommunikation auch der Begriff Konversation genutzt. Konversation meint ein Gespräch, das unter Beachtung von Umgangsformen geführt wird. Konversation kann somit als Unterbegriff der Kommunikation verstanden werden. In dieser Arbeit werden die Begriffe nicht getrennt voneinander behandelt.

Der Begriff Kommunikation ist über die Informationstheorie zugänglich. Das bekannteste Werk, was im Zusammenhang mit Informationstheorie und Kommunikation entstanden ist, ist wahrscheinlich das Shannon-Weaver-Modell, welches auch unter dem Namen Sender-Empfänger-Modell bekannt ist.

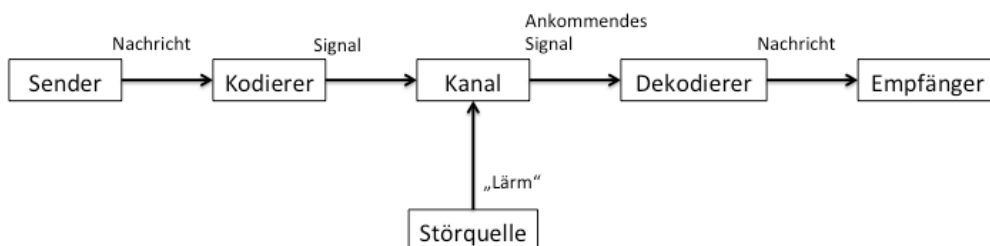


Abbildung 10: Shannon-Weaver-Modell

Das Modell besagt, dass Information stets von einem Sender an einen Empfänger geleitet wird. Der Sender oder auch die Informationsquelle leitet eine Botschaft an ein Sendergerät, also einen Kodierer weiter. Dieser verwandelt die Botschaft in Signale und leitet sie über einen Kanal an den Empfänger. Innerhalb des Kanals können

Störquellen auftreten, die die Information verfälschen. Das Empfängergerät dekodiert die Signale wiederum in eine Botschaft und leitet diese an die Adressaten [8]. Dieses Modell fokussiert sich auf einen Teil der Kommunikation, und zwar auf die Information beziehungsweise die Nachricht. Paul Pangaro entwickelte auf Basis der Konversationstheorie von Gordon Pask ein Kommunikation-Modell, welches sich nicht auf die Nachricht konzentriert, sondern alle Bestandteile von Kommunikation berücksichtigt. Nach Pangaro ist Shannons und Weavers Modell schon allein nicht auf die menschliche Kommunikation anwendbar, da der Kontext einer Information nicht berücksichtigt wird [9, S. 2].

Pangaro entwickelte deswegen ein Modell, welches er kurz als C-L-E-A-T bezeichnet. Dabei steht CLEAT für Context, Language, Exchange, Agreement und Transaction, welche die essentiellen Bestandteile von Konversation bilden.

CONVERSATION = C-L-E-A-T

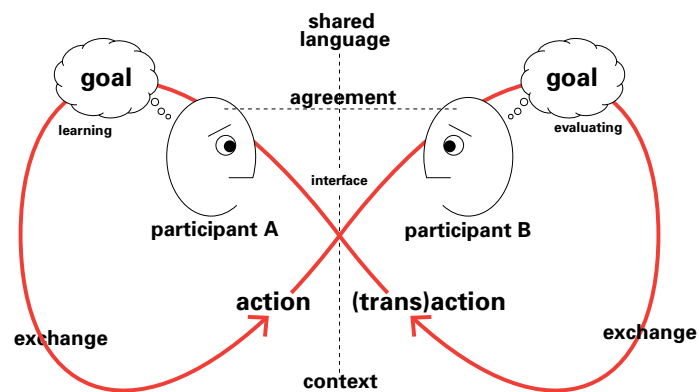


Abbildung 11: Konversationsmodell nach Pangaro

Die Bestandteile des Modells können mit folgenden Begriffen ins Deutsche übersetzt werden:

- Kontext
- Sprache
- Austausch
- Einverständnis
- (Trans)aktion

Konversation beginnt zumeist damit, dass einer der Teilnehmer oder beide ein Ziel haben. Zuerst wird dann der Kontext der Konversation bestimmt, beispielsweise ist A mit dem Zug auf dem Weg zu B und ruft B an. Der Kontext ist hier klar gesetzt und B wird eine Nachricht erwarten, die etwas mit dem Treffen der beiden zu tun hat. Nachdem der Kontext gesetzt wurde, muss eine Sprache festgelegt werden, die beide verstehen. Dies muss nicht zwangsläufig auf die Landessprache bezogen sein. Nachdem diese Komponenten klar festgelegt wurden, kann der Austausch von Informationen beginnen. Um das Beispiel fortzuführen, sagt A zum Beispiel zu B, dass der Zug, mit welchem er kommt, Verspätung hat. Der Prozess des Austausches wird weitergeführt, indem B die Information evaluiert. Der Austausch kann dabei beliebig viele Male zwischen den beiden Teilnehmern hin und her gehen, bis ein Einverständnis herrscht. Im konstruierten Beispiel könnte das heißen, dass A und B zu dem Einverständnis gekommen sind, dass B A nicht mehr vom Bahnhof abholt und A mit dem Taxi zu B kommt. Über das Einverständnis hinaus können aus der Konversation Transaktionen entstehen, die nach der Konversation von einem der Teilnehmer ausgeführt werden. Im Beispiel wäre eine solche Transaktion das Telefonat mit einem Taxiunternehmen, welches A dann führt.

Pangaro hat seine Theorien auch auf das Internet und Software bezogen. Daher können sie auch auf den Chatbot angewendet werden.

3.2 The Media Equation

Kommunikation kann ebenfalls eine Interaktion zwischen einem Menschen und einem Computer sein. Diese Interaktionen werden im Forschungsgebiet Mensch-Computer-Interaktion wissenschaftlich bearbeitet. Dieses Gebiet beschäftigt sich allgemein mit der Schnittstelle zwischen dem Menschen und dem Computer.

Um die Interaktion zwischen dem Menschen und dem Computer zu verstehen, beschäftigten sich Byron Reeves und Clifford Nass viele Jahre mit der Beziehung, welche Menschen zu Computern haben. Dazu führten sie diverse Verhaltensexperi-

mente durch, und veröffentlichten ihre Ergebnisse in dem Buch „The Media Equation“¹. Der Begriff Gleichung bedeutet in dem Zusammenhang, dass Medien dem realen Leben gleichen [10, S. 5]. Die durchgeführten Studien zeigten alle eindeutig, dass Menschen Computern soziale Verhaltensweisen entgegenbringen. Eines der Experimente zielte beispielsweise auf die Höflichkeit ab, die Menschen Computern entgegenbringen. In dem Experiment ließen die Forscher 22 Menschen mit einem Computer interagieren, der ihnen Fakten zu einem bestimmten Thema lehrte. Zu diesen Fakten sollte der Versuchsteilnehmer dann einen Test absolvieren, in dem das erlernte Wissen abgefragt wurde. Danach wurden die Teilnehmer gebeten den Computer zu bewerten. Dabei evaluierte die eine Hälfte der Teilnehmer den Computer an demselben Computer, an welchem sie den Test durchgeführt hatten. Die andere Hälfte durfte den Computer an einem zweiten Computer evaluieren, den sie zuvor nicht benutzt hatten. Genau wie bei dem äquivalenten Experiment bei der Interaktion zwischen zwei Menschen, bewerteten die Teilnehmer den Computer signifikant besser, wenn sie ihn „persönlich“, also am selben Computer, bewerten sollten [10, S. 23].

Nass und Reeves führen dieses soziale Verhalten darauf zurück, dass sich das menschliche Gehirn entwickelt hat und sich soziale Regeln etabliert haben, als es nur mit „echten“ Dingen zu interagieren galt und alles, was wie ein Mensch oder ein Ort erschien, auch ein Mensch oder ein Ort war [10, S. 12]. Dass Menschen ein Medium als „unecht“ bezeichnen, passiert erst nach einem Denkprozess, vor welchem auch Medien erst einmal als real beziehungsweise „echt“ wahrgenommen werden (z.B. Horrorfilme) [10, S. 13]. Andere Erklärungen für das soziale Verhalten und die Anwendung von sozialen Regeln bei der Interaktion mit Computern wie, dass der Mensch etwa auf den Programmierer reagieren und nicht auf den Computer selbst, wurden nach Unterhaltungen mit den Teilnehmern ausgeschlossen [10, S. 27].

Nass und Reeves schließen aus diesen Studien, dass man das Design von Medien verbessern könnte, indem die Medien selbst soziale Regeln anwenden. So müsste der Mensch nicht erst die Regeln eines Mediums erlernen, sondern könnte das Medium direkt durch bereits bekannte soziale Normen nutzen. Je konsistenter ein Medium also mit sozialen Regeln ist, desto angenehmer wird es für den Menschen sein dieses Medium zu nutzen [10, S. 8].

dt. Die Medien-Gleichung

Das gängigste Regelwerk, welches zu Umgangsformen und sozialer Norm in der Kommunikation besteht, sind die vier Konversationsmaximen von Paul Grice:

- Maxime der Quantität
- Maxime der Qualität
- Maxime der Relation
- Maxime der Modalität

Mit der Maxime der Quantität ist gemeint, dass ein Sprecher immer genauso viel Information geben soll, wie in der Konversation gebraucht wird. Die Maxime der Qualität besagt, dass der Gesprächsbeitrag stets wahr sein muss. So soll während einer Konversation nicht gesagt werden, was man für falsch befindet oder wofür man nicht genügend Anhaltspunkte hat. Die Maxime der Relation soll vermitteln, dass innerhalb einer Konversation Gesprächsbeiträge zum Kontext der Konversation passen. Des Weiteren vermittelt dies auch die Regel während eine Konversation das Thema nie abrupt zu wechseln. Die Maxime der Modalität meint, dass der Sprecher Unklarheit und Mehrdeutigkeit unbedingt vermeiden soll [11, S. 251].

Auch wenn die vier Maximen von Grice nicht allgemein gültig sind und diese in Gesprächen oft genug verletzt werden, ohne dass das Gespräch von einem der beiden Gesprächspartner als misslungen angesehen wird, können diese Maximen eine hilfreiche Richtlinie für das Design der Kommunikation eines Chatbots sein, da anders als bei zwei Menschen zwischen dem Chatbot und dem Nutzer kein richtiges Einverständnis zur Missachtung einer dieser Maximen herrschen kann.

3.3 Mediale Persönlichkeit

In eine Konversation zwischen zwei Gesprächspartnern fließt auch immer die Persönlichkeit der beiden ein. Die Persönlichkeit ist die Gesamtheit der Charaktereigenschaften einer Person. In der Psychologie findet man im Forschungsgebiet zur Form der Persönlichkeit das Fünf-Faktor-Modell:

- Extraversion
- Verträglichkeit
- Gewissenhaftigkeit
- Neurotizismus
- Offenheit für Erfahrungen

Der Faktor Extraversion bezieht sich auf die Dominanz und die Geselligkeit einer Person. Während Adjektive wie extrovertiert, gesprächig und aktiv stark ausgeprägte Extraversion widerspiegeln, beschreiben Adjektive wie zurückhaltend und ruhig eine schwach ausgeprägte Extraversion, auch Introversion genannt. Eine hohe

Ausprägung der Verträglichkeit wird mit Adjektiven wie freundlich und mitfühlend gleichgesetzt. Eine schwache Ausprägung wird zumeist mit Adjektiven wie aggressiv, antagonistisch und wetteifernd verbunden. Der Faktor Gewissenhaftigkeit ist bei schwacher Ausprägung mit dem Adjektiv nachlässig und bei starker Ausprägung mit dem Adjektiv organisiert zu verstehen. Der Faktor Neurotizismus bezieht sich auf die emotionale Labilität einer Person. So sind Personen mit einer schwachen Ausprägung eher selbstsicher und Personen mit einer starken Ausprägung eher verletzlich. Bei der Offenheit für Erfahrungen können Personen in ein Spektrum zwischen neugierig und vorsichtig eingestuft werden [12, S. 179].

Obwohl alle fünf Faktoren eine wichtige Rolle für Persönlichkeitsmerkmale spielen, gibt es einige Indizien dafür, dass der Faktor Extraversion und Verträglichkeit die entscheidenden Faktoren sind [10, S. 76].

Dass Menschen diese Persönlichkeitsmerkmale nicht nur in anderen Menschen wiederfinden, sondern auch in medialen Persönlichkeiten, bewiesen Reeves und Nass in verschiedenen Studien. In einer dieser Studien wurden ausschließlich Erwachsene aufgefordert verschiedene Cartoons in ihrer Ähnlichkeit zu bewerten. Dabei wurden alle Cartoons, die von unterschiedlichen Zeichnern erstellt wurden, in vier Dimensionen unterteilt. Die erste Dimension unterteilte die Cartoons in lebende Objekte (Tiere) und unbelebte Gegenstände wie Schuhe oder Bücher, wobei auch diesen Gegenständen gewisse lebensbejahende Attribute wie Augen gemalt wurden. In der zweiten Dimension unterschieden die Teilnehmer zwischen den Cartoons von einem Künstler und allen anderen, da dieser einen besonderen und sich von den anderen abhebenden Stil nutzte. Die dritte und vierte Dimension widerspiegeln zwei Faktoren des Fünf-Faktor-Modells, Extraversion und Verträglichkeit. Auf der einen Seite standen Cartoons wie der Alligator, der Wolf und der Piranha, während sich auf der anderen Seite Cartoons wie der ängstlich schauende Vogel, die Katze und der ballonförmige Hund befanden. In der vierten Dimension (Verträglichkeit) befanden sich auf der einen Seite nur lächelnde Gesichter und auf der anderen lediglich Grimassen [10, S. 82].

Der Vorteil, den mediale Persönlichkeiten allerdings vor Menschen haben, ist, dass sie eine berechenbare und einheitliche Persönlichkeit haben, was ihnen oft zu Gute kommt, da Menschen grundsätzlich keine komplexen und widersprüchlichen Persönlichkeiten mögen [10, S. 84]. Das Fünf-Faktor-Modell sollte also beim Design eines Chatbots berücksichtigt werden.

Das Fünf-Faktor-Modell, Grice Maxime und Pangaros Konversationsmodell dienen als eine sehr gute Grundlage, um das Design und die Entwicklung eines Chatbots im Folgenden näher zu betrachten.

4 Entwurf und Umsetzung des Chatbots

Die theoretischen Ansätze aus dem vorigen Kapitel sollen nun im folgenden Kapitel genutzt werden, um den Rezepte-Chatbot unter der Beachtung von zu definierenden Regeln zu entwickeln.

4.1 Das Konversationsdiagramm

Während der Designprozess einer App oder einer Webseite meist mit der Erstellung von Mockups¹ auf dem Papier oder mit einem Mockup-Tool beginnt, sollte der Design-Prozess eines Chatbots mit einer Abbildung der Konversation beginnen. Mockups eignen sich, um das Konzept der App ohne viel Detail zu visualisieren. Da es keinen Sinn hat eine gesamte Konversation in Form des Messenger-Designs zu visualisieren, weil das Design des Bildschirms weder entscheidend noch beeinflussbar ist, sollte das Konversationsdesign auf eine andere Weise begonnen werden. Genau wie bei App-Mockups sollte man sich nicht in Details verlieren und bereits die Sprache des Chatbots modellieren. Viel wichtiger ist es sich einen Überblick über alle möglichen Konversationsstränge zu verschaffen. Dazu bietet es sich an ein Aktivitätsdiagramm in UML zu erstellen. UML ist eine Modellierungssprache, die Software und Systeme von der Analyse bis hin zur Dokumentation beschreiben kann. In einem Aktivitätsdiagramm wird der zeitliche Ablauf von einer Reihung von Aktionen modelliert. Nach einigen Versuchen erwies sich das Aktivitätsdiagramm als am besten geeignet, um die Konversation zwischen dem Chatbot und dem Nutzer zu modellieren. Das ist damit zu erklären, dass Konversation ein Verhalten beschreibt, welches gut mit einem Verhaltensdiagramm beschrieben werden kann. Des Weiteren verläuft Konversation in Abhängigkeit der Zeit und kann enger als eine Aktivität beschrieben werden, wobei die Aktivität vom Nutzer beziehungsweise vom Chatbot ausgeht (ausgeführt wird).

¹ dt. Modell

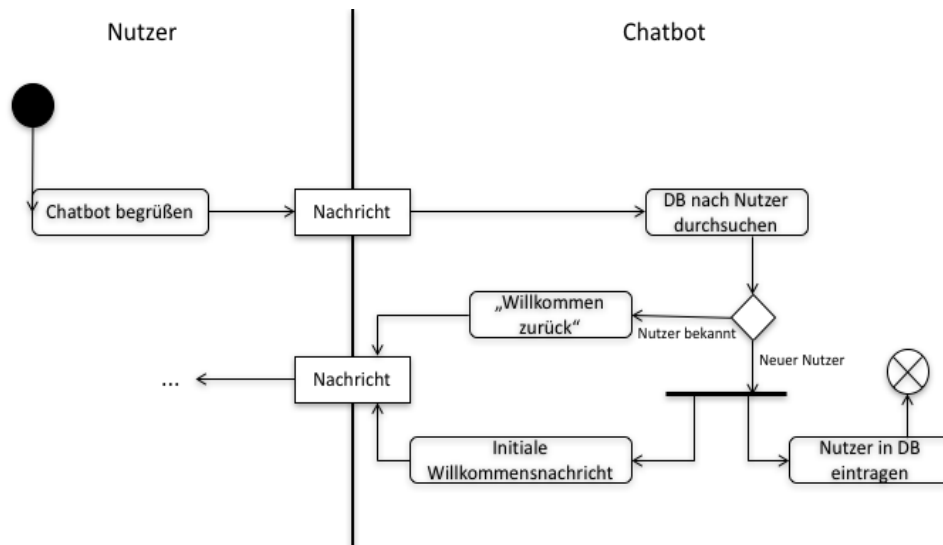


Abbildung 12: Ausschnitt aus dem Aktivitätsdiagramm des Rezepte-Bot: Begrüßung

In einem Aktivitätsdiagramm werden die Bestandteile durch verschiedene Elemente dargestellt. Eine Aktion, wie z.B. den Chatbot zu begrüßen, wird als ein Rechteck mit abgerundeten Ecken dargestellt. Ein Objektknoten wie die Nachricht wird als Rechteck skizziert. Objektknoten und Aktionen werden mit Pfeilen verbunden um den Ablauf zu bestimmen. Zwischen Pfeilen und Aktionen und Objektknoten können außerdem Kontrollknoten liegen. Zum einen gibt es Kontrollpunkte in Form eines Kreises, die den Beginn und das Ende einer Aktivität anzeigen. Dabei zeigt ein ausgefüllter Kreis den Beginn einer Aktivität und ein Kreis mit ausgefülltem Mittelpunkt das Ende einer gesamten Aktivität. Ein Kreis, der mit einem „X“ ausgefüllt ist, zeigt das Ende eines Ablaufs an. Neben den Start- und Endknoten gibt es noch einen Verzweigungsknoten, der in Form einer Raute dargestellt wird. Typischerweise werden diese gebraucht, wenn der Ablauf einer Aktivität von einer Bedingung abhängig ist. Im Diagramm des Rezepte-Chatbots ist dies z.B. zu sehen, wenn der Chatbot entscheiden muss, ob er den Nutzer schon kennt oder nicht, um ihm dann eine adäquate Begrüßung zukommen zu lassen. In dieser Situation lässt sich ein weiterer Knoten beschreiben – der Parallelisierungsknoten. Mit diesem Knoten können Aktivitäten synchron ablaufen, wie z.B. die initiale Begrüßungsnachricht zu senden und parallel dazu den neuen Nutzer in die Datenbank einzutragen.

Neben den Vorteilen für den Design-Prozess bietet das Aktivitätsdiagramm auch Vorteile für den Entwicklungsprozess, da Aktionen im weiteren Sinne ein Ereignis darstellen und node.js ereignisgesteuert ist. Dementsprechend kann das Aktivitätsdiagramm auch ein erster Anhaltspunkt für die Architektur und die Klassenstruktur sein. Um diesen Vorteil optimal zu nutzen, bietet es sich an, Aktionen, die der Chatbot zusätzlich zur Konversation ausführen muss, wie beispielsweise einen Datenbankzugriff, direkt in das Diagramm einzubinden.

Das Aktivitätsdiagramm geht außerdem mit dem Konversationsmodell von Pangaro konform, sofern man die Aktivitäten des Nutzers und des Chatbots klar unterteilt. Die Objektknoten entsprechen dem Austausch und die Aktionen entsprechen den Aktionen bzw. Transaktionen in Pangaros Modell.

Durch die Modellierung der Konversation mit Hilfe der Aktivitätsdiagramme kann der Use-Case effizient in Korrespondenz mit den Konversationsprinzipien abgebildet werden. So dient das Aktivitätsdiagramm genau wie ein Mockup als erster Prototyp des Produktes und ist essentiell für die weiteren Design- und Entwicklungsschritte.

4.2 Umsetzung von Pangaros Konversationsmodell

Aus Pangaros Konversationsmodell, welches in Kapitel 3.1 beschrieben wurde, lassen sich Regeln zur Konzeption der Konversation ableiten, die beim Design eines Chatbots beachtet werden sollten.

4.2.1 Kontext

Wie in Kapitel 3.1 erklärt wurde, ist eines der wichtigsten Bestandteile von Konversation der Kontext derselben. Vor allem bei der Konzeption einer Chatbot-Konversation sollte der Kontext höchste Priorität haben.

Dies ist vor allem darin begründet, dass bis heute noch kein Chatbot den von Alan Turing erstellten Turing-Test bestanden hat. Ziel des Tests ist es herauszufinden, ob eine Maschine das gleiche Denkvermögen wie ein Mensch hat, also einem anderen Menschen glaubwürdig vermitteln könnte, selbst ein Mensch zu sein. In nächster Zeit wird das Bestehen des Turing-Tests für gängige Chatbots nicht erreichbar sein und auch kein Maßstab für den Erfolg eines Chatbots sein. Vielmehr gilt es damit umzugehen und diese Schwäche beim Design des Chatbots zu beachten. Deswegen ist es wichtig, dass der Chatbot immer den Kontext der Konversation vorgibt. Des Weiteren muss der Chatbot mit dem Kontext, den der Nutzer zur Konversation beiträgt, umgehen können. Daraus ergeben sich folgende Regeln.

Der Chatbot sollte zu Beginn der Konversation direkt auf den Use-Case aufmerksam machen, um den Nutzer so für die Benutzungsmöglichkeiten des Chatbots zu sensibilisieren. Der Rezepte-Bot macht dies, indem er nach der Begrüßung eine initiale Nachricht schickt, in welcher er erklärt, dass der Nutzer ihn nach Koch- und Backrezepten fragen kann.



Abbildung 13: Chatbot erklärt, dass er Rezepte suchen kann

Der Kontext sollte bei jeder Gelegenheit wieder auf die Kernaufgabe des Chatbots zurückkommen, damit die Konversation nicht in Richtungen abdriftet, die der Chatbot selber nicht mehr kontrollieren kann. Der Rezepte-Bot versteht beispielsweise, wenn der Nutzer ihn nach seinem Befinden fragt. Jedoch erkundigt er sich nie selbst nach dem Befinden des Nutzers, da die Antworten auf diese Fragen zu unvorhersehbar sind. Außerdem versucht er bei der Antwort auf diese Frage stets auf eine mögliche sich anschließende Rezeptsuche zurückzukommen.



Abbildung 14: Chatbot erzählt etwas zu seinem Befinden und fragt den Nutzer dann, ob er bereit für das Kochen ist

Auf der anderen Seite ist es wichtig, dass der vom Nutzer vorgegebene Kontext von dem Chatbot richtig verstanden wird. Dies wird über den Einsatz eines NLP-Tools und Intents erreicht. Über das korrekte Setzen von Intents kann es dem Chatbot vor allem möglich gemacht werden einen Kontextwechsel zu erkennen. Der Rezepte-Bot erkennt zum Beispiel anhand einer Verabschiedung, dass der Kontext der Konversation sich zu einer Verabschiedung verschoben hat.

4.2.2 Sprache

Ein weiterer wichtiger Aspekt von Konversation ist nach Pangaro die Sprache, die die Konversationspartner teilen. Es ist deswegen essentiell für das Design, dass der Chatbot nicht in der ihm bekannten konstruierten Sprache kommuniziert, sondern in der Sprache des Nutzers, welche zunächst einmal die natürliche Sprache ist. Das Beherrschen der natürlichen Sprache ist auch der große Vorteil, den ein Chatbot

gegenüber herkömmlicher Apps und Software hat. Auch diese Applikationen sprechen eine Sprache, jedoch muss diese Sprache erst vom Nutzer erlernt werden. Nach dem Konversationsmodell von Pangaro sollte es aber so sein, dass beide Konversationspartner die abgestimmte Sprache bereits verstehen und sich in dieser ausdrücken können. Das ist der natürlichen Sprache möglich, da der Nutzer diese beherrscht und der Chatbot die natürliche Sprache über ein NLP-Tool wie wit.ai in seine Sprache übersetzen kann.

Ein weiterer wichtiger Aspekt ist die Landessprache. Hier gibt es mehrere Möglichkeiten vorzugehen. Zum einen sollte wie bei Apps und Software auch die Landessprache der Zielgruppe des Chatbots eine große Rolle spielen. So sollte der Chatbot vorzugsweise die Landessprache des angestrebten Marktes sprechen.

Des Weiteren ist es wie bei Apps ebenfalls möglich über Facebook Informationen zur Sprachumgebung des Nutzers zu erhalten. Diese kann über die Nutzerprofil-API abgefragt werden und liefert so einen Anhaltspunkt über die mögliche Sprache des Nutzers.

Möglich ist es auch die Nachrichten des Nutzers auf die Landessprache zu untersuchen. Jedoch müsste dies direkt bei der ersten Nachricht gemacht werden und könnte bei Begrüßungen wie „Hi“ schwierig werden.

Eine weitere Möglichkeit ist es die Einstellung der Sprache dem Nutzer zu überlassen. Über das NLP-Tool könnte so eine Entität mit allen verfügbaren Sprachen angelegt werden. Schreibt der Nutzer im Kontext der Sprachabfrage eine der verfügbaren Sprachen, kann die Sprache entsprechend umgestellt werden.

Der Rezepte-Chatbots bietet lediglich die Sprache Englisch an. Dies ist zum einen darin begründet, dass der Chatbot über den Facebook Messenger für jeden Facebook-Nutzer verfügbar ist. Des Weiteren liefert die angebundene Rezepte-API Spoonacular nur englischsprachige Rezepte und bietet somit nur eine englischsprachige Suche an.

4.2.3 Austausch

Ein weiterer Bestandteil von Pangaros Konversationsmodell ist der Austausch zwischen den Konversationspartnern. Wichtig ist es vor allem, dass dieser Austausch ein Austausch bleibt und die Konversation nicht einseitig bleibt. Deswegen sollte der Chatbot auf so gut wie jede Nachricht des Nutzers reagieren. Das hilft ebenfalls um den Fokus der Konversation bei dem Use-Case des Chatbots zu behalten.

Der Chatbot sichert den Austausch über eine Fallback-Antwort, die er dem Nutzer schickt, wenn ihm die Intention des Nutzers nicht bekannt ist.

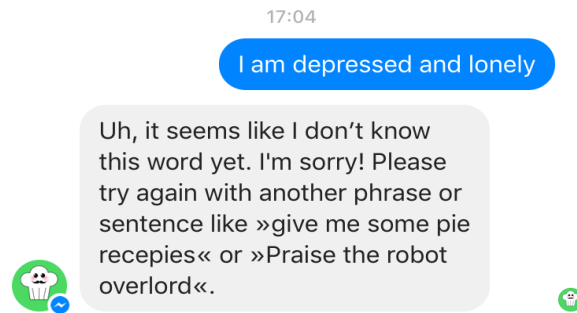


Abbildung 15: Fallback-Antwort des Rezepte-Chatbots

Es bietet sich an zusätzlich zum Ausdruck des Unverständnisses einen Versuch zu machen, die Konversation wieder auf den Use-Case zu lenken, indem man dem Nutzer Hinweise gibt, was er stattdessen schreiben könnte. Das fördert den Austausch zusätzlich.

4.2.4 Einverständnis & Transaktion

Nach dem Konversationsmodell soll es im Idealfall zu einem Einverständnis zwischen den Konversationspartnern kommen. Die Konversation sollte für den Chatbot so konzipiert werden, dass ein Einverständnis immer dann eintritt, wenn der Chatbot seine Aufgabe erledigt hat, wenn also der Use-Case bedient wurde.

Im Falle des Rezepte-Chatbots wäre ein Einverständnis also eingetreten, wenn der Nutzer ein zufriedenstellendes Rezept erhalten hat. Das Einverständnis ist somit auch automatisch ein Maßstab für den Erfolg des Chatbots und für die Umsetzung des Use-Case.

Der letzte Schritt einer Konversation ist nach dem Konversationsmodell von Pangaro die (Trans-)Aktion. Auf der Seite des Chatbots sind alle Datenbankzugriffe und API-Zugriffe als Aktionen zu sehen, die aus der Konversation entstehen.

Währenddessen ist eine Aktion innerhalb der Konversation auf Seiten des Nutzers eher selten, sofern man das Tippen nicht als Aktion interpretiert. Dafür ist es bei vielen Use-Cases der Fall, dass aus der Konversation mit dem Bot eine direkte Aktion entsteht.

Idealerweise sollte nach der Konversation mit dem Rezepte-Chatbot das Nachkochen eines der Rezepte folgen. Auch Aktionen können somit ein Indikator für den Erfolg des Chatbots sein. Jedoch lassen sich diese wesentlich schlechter als das Einverständnis nachweisen, da der Chatbot keine Informationen über den eventuell folgenden Kochprozess hat.

4.3 Umsetzung von Grice Maximen

Von Grice Maximen, die in Kapitel 3.2 beschrieben wurden, lassen sich ebenfalls einige Prinzipien ableiten, die beim Design einer Chatbot-Konversation beachtet werden sollten.

4.3.1 Quantität

Die Maxime der Quantität birgt eine Regel, die bei der Entwicklung der Konversation zu beachten sind. Da sich die Konversation im schlimmsten Fall in einem 300 x 300 px großem Chatfenster abspielt, ist es wichtig, dass alle Texte des Bots kurz und prägnant sind. Der längste Text, den der Rezepte-Chatbot schreiben kann, ist der Hilfe-Text. Dieser ist 74 Wörter lang und passt genau in das Chatfenster der Facebook-Webseite.

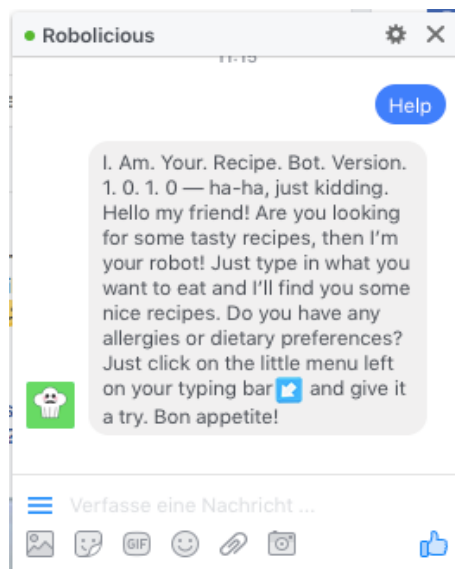


Abbildung 16: Chatfenster auf der Facebook-Webseite mit Hilfetext

Es sollte unter allen Umständen vermieden werden, dass der Nutzer zurückscrollen² muss, um den gesamten Text lesen zu können, den der Chatbot in einem schickt. Falls es sich nicht vermeiden lässt, einen großen Text auf einmal zu schicken, bietet es sich an den Text in mehrere Texte zu unterteilen und diese mit Hilfe von Timeouts nacheinander zu senden.

² dt. in der Unterhaltung zurückblättern

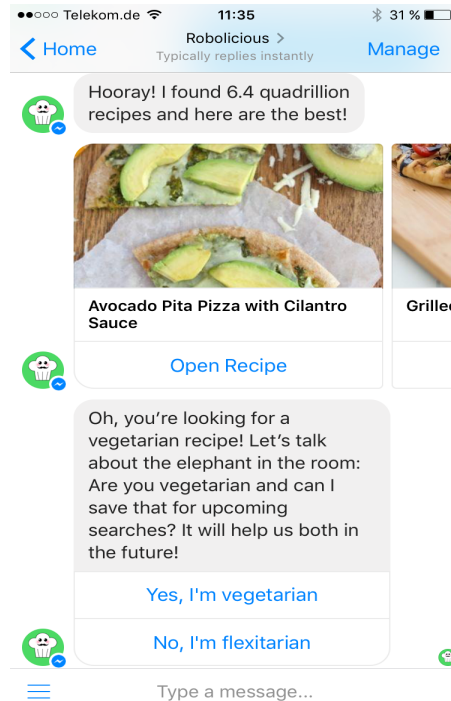


Abbildung 17: Drei Nachrichten, wenn der Nutzer nach einem Rezept mit Ernährungsform fragt und keine Ernährungsform für diesen Nutzer gespeichert ist

Der Rezepte-Chatbot unterteilt bei einer Rezeptanfrage seine Antworten ebenfalls auf. Die erste Nachricht wird gesendet, um dem Nutzer direkt zurückzumelden, dass die Rezepte-Anfrage erfolgreich war. Danach sendet der Chatbot eine Galerie mit den Rezepten, die er gefunden hat. Die Galerie hat ebenfalls einen platzsparenden Effekt und vermeidet, dass der Nutzer sich durch alle Ergebnisse durchscrollen muss. Falls ihn das erste Rezept anspricht, muss er die anderen Rezepte gar nicht mehr anschauen. Falls der Nutzer eine Ernährungsform wie „vegetarisch“ vor seine Rezeptsuche hängt und er bis jetzt keine Ernährungsform in der Datenbank hinterlegt hat, fragt der Chatbot ihn nach der Ausgabe der Ergebnisse, ob er die eingegebene Ernährungsform in der Datenbank für zukünftige Suchen speichern möchte. Diese Frage wird jedoch um 12 Sekunden verzögert, damit der Nutzer genug Zeit hat, sich die eigentlichen Ergebnisse anzuschauen und nicht von der Frage zur Ernährungsform unterbrochen wird.

4.3.2 Qualität

Die Maxime der Qualität ist beim Design von Konversation nicht weniger wichtig als die Qualität eines App- oder Webseitendesigns. Nach der Maxime der Qualität soll der Gesprächsinhalt stets wahr sein und man sollte vermeiden etwas zu sagen, was eventuell falsch sein könnte.

Es ist daher von Vorteil, den Parameter Sicherheitswert, den wit.ai mit seinen Antworten liefert, zu nutzen. Durch Ausprobieren sollte man einen Schwellenwert festlegen, den der Sicherheitswert übersteigen muss. Falls der Schwellenwert nicht erreicht wird, sollte der Chatbot lieber auf eine Fallback-Antwort, wie im Kapitel 4.2.3 „Austausch“ beschrieben, zurückgreifen.

Die wit.ai-App des Rezepte-Chatbot enthält eine Entität „query“, die als Freitext alle möglichen Sucheingaben herausfiltert. Im Beispielsatz „Gib mir ein Rezept für Lasagne.“ sollte wit.ai das Schlagwort „Lasagne“ als Sucheingabe erkennen. Mit der Suchstrategie Freitext verhält es sich teilweise schwierig, da wit.ai oftmals dort Sucheingaben findet, wo eigentlich keine zu finden sind. Um dieses Problem etwas einzudämmen und die Qualität der Suchergebnisse zu wahren, ist der Chatbot so programmiert, dass er erst bei einem Sicherheitswert der Entität „query“ von 70 % eine Rezeptsuche einleitet. Falls der Sicherheitswert darunterliegt, spielt er eine Fallback-Antwort aus.

4.3.3 Relation

Da die Maxime der Relation die Beachtung des Kontextes einer Konversation beschreibt, gehen aus ihr dieselben Regeln wie aus dem Kontextteil des Konversationsmodells in Kapitel 4.2.1 hervor.

Eine weitere Möglichkeit, den Kontext auch über mehrere Konversationen aufrechtzuerhalten, ist es Nutzer in einer Datenbank zu hinterlegen. Der Chatbot kann sich so anders gegenüber bereits bekannten Nutzern „verhalten“.

Da der Rezepte-Chatbot den Nutzer sowieso aufgrund der Ernährungsformfunktion in einer Datenbank hinterlegen muss, nutzt er diese Information auch gleichzeitig, um dem Nutzer eine andere Willkommensnachricht ausspielen zu können. Falls der Nutzer bereits bekannt ist, bekommt er eine kürzere „Willkommen zurück“-Nachricht, die dem Nutzer nicht erneut erklärt, was der Chatbot kann.

4.3.4 Modalität

In der Maxime der Modalität spiegelt sich ein klarer Vorteil wider, den ein Konversations-User-Interface gegenüber einem grafischen User-Interface hat. Komplexe Informationen lassen sich oft nicht mit einem Bild erklären, sondern benötigen einen umfangreichen Erklärungstext, der beispielsweise auf einem Button keinen Platz findet. Daraus resultiert, dass wie in dem Kapitel 3.2 „The Media Equation“ beschrieben der Nutzer einer grafischen Oberfläche diese oftmals erst erlernen muss, um Buttons und etwaige Menüfelder zu verstehen.

Die Texte eines Chatbots können also Unklarheiten vermeiden, sofern sie gut strukturiert und formuliert sind. Dabei können Werkzeuge, die die Lesbarkeit des Textes messen, helfen.

Zur Kontrolle der Rezepte-Chatbot-Texte wurde das Werkzeug Hemingway-App genutzt. Diese wendet den „Automated Readability Index“ an, der folgende Rechnung vorgibt.

$$4,71 \times \frac{\textit{characters}}{\textit{words}} + 0,5 \times \frac{\textit{words}}{\textit{sentences}} - 21,43$$

Das Ergebnis dieser Rechnung ist ein Punktzahl, anhand welcher ein Schulklassen-Level abgelesen werden kann [13, S. 8].

Ziel ist es mit Hilfe der Hemingway-App die Chatbot-Texte so zu editieren, dass das Schulklassen-Level möglichst gering bleibt. Der Hilfetext des Rezepte-Chatbots kann so z.B. gekürzt werden. Bei der Eingabe in die App zeigt diese an, dass der Text dreimal das Wort „just“ enthält, wobei die App angibt, dass der Text so wenige Adverbien wie möglich enthalten sollte, um die Lesbarkeit des Textes zu steigern. Um der Maxime der Modalität zu folgen, ist es also ratsam Werkzeuge wie die Hemingway-App zu nutzen [14].

4.4 Umsetzung von Reeves und Nass „The Media Equation“

Wie im Kapitel 3.2 erklärt wurde bringen Menschen Computern soziale Verhaltensweisen entgegen. Die logische Schlussfolgerung daraus ist, dass ein Chatbot diese sozialen Verhaltensweisen nach Möglichkeit erwidern sollte. Teilweise macht er dies durch die Konversation selbst. Allerdings lassen sich aus den Studien von Reeves und Nass einige weitere Regeln ableiten, die noch nicht durch Grices Maxime oder Pangaros Konversationsmodell ersichtlich wurden.

Die Höflichkeit spielt dabei eine übergeordnete Rolle in der Interaktion. Zum einen müssen die Texte sehr exakt formuliert werden, zum anderen sollte der Chatbot Höflichkeitsnormen kennen und auf solche reagieren können. Zu diesen Höflichkeitsnormen gehört beispielsweise das Erwidern eines Grußes oder einer Verabschiedung. Des Weiteren ist ein typisches Beispiel für höfliches Verhalten eine bescheidene Reaktion auf Dank.

Der Rezepte-Chatbot folgt einigen Höflichkeitsnormen. Er antwortet auf Begrüßungen und Verabschiedungen, Dank und die Frage nach seinem Befinden.

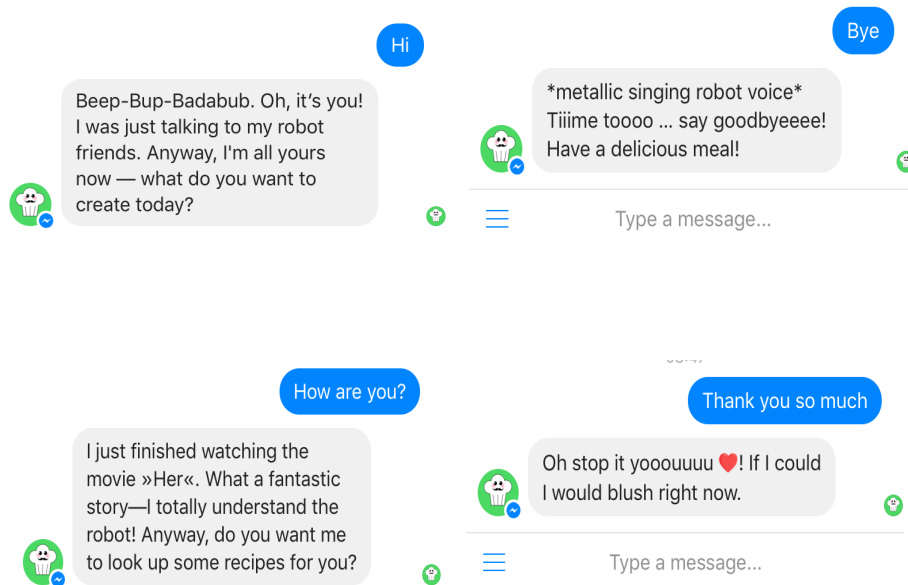


Abbildung 18: Antworten auf Höflichkeitsnormen

Außerdem kann er als Zugabe auf Zuneigungen wie „Love you“ antworten.

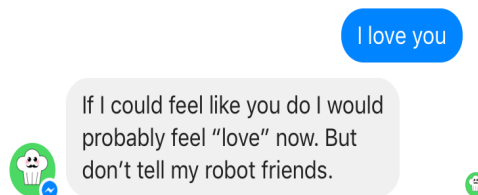


Abbildung 19: Chatbot reagiert auf Zuneigung

Neben der Höflichkeit ist Lob nach Reeves und Nass eine weitere soziale Norm, der Computer folgen sollten [10, S. 59]. Man sollte es jedoch auch nicht übertreiben, da der Nutzer das Lob sonst schnell nicht mehr ernst nimmt. Wenn der Chatbot aber ein- bis zweimal in einer Konversation Lob ausspricht, kann sich dies nachhaltig positiv auf die Einschätzung des Nutzers gegenüber des Chatbots auswirken. Das Lob muss dabei nicht zwangsläufig gerechtfertigt sein.

Der Rezepte-Chatbot lobt den Nutzer indirekt, wenn dieser seine Ernährungsform-einstellungen aufruft und eine Ernährungsform hinterlegt hat, indem er seine Ernährungsform als gesund umschreibt.

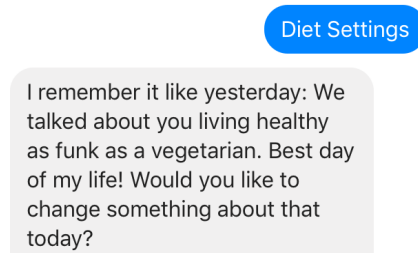


Abbildung 20: Indirektes Lob des Chatbots

Das Ziel des Designprozesses soll es nicht sein, den Chatbot als Menschen zu tarnen. Das würde nur zu Frustration auf der Nutzerseite führen, da die Erwartungshaltung an einen Menschen eine ganze andere ist als an einen Chatbot. Wie in Kapitel 3.2 bereits erklärt, sieht der Mensch jedes Medium erst einmal als „echt“ an. Das Ziel des Designprozesses muss es also sein einen realen Chatbot zu schaffen, der seine Usability³ durch das Einhalten sozialer Normen steigert.

4.5 Text-Design und Chatbot-Persönlichkeit

Sobald die Struktur der Konversation entworfen ist, können die Textinhalte entwickelt werden. Neben den bereits beschriebenen Regeln gibt es beim Textdesign weitere Punkte, die zu beachten sind. Nicht zu trennen von dem Textdesign ist die Chatbot-Persönlichkeit, da der Chatbot seiner Persönlichkeit nur durch die Texte Ausdruck verleihen kann.

4.5.1 Chatbot-Persönlichkeit

Wie in Kapitel 3.3 beschrieben, weist der Mensch auch medialen Figuren Persönlichkeitsmerkmale zu und bewertet mediale Figuren nach ihrer Persönlichkeit. Die Chatbot-Persönlichkeit sollte deswegen eine große Rolle in dem Designprozess des Chatbots spielen, da durch eine einheitliche Persönlichkeit die Nutzerfreundlichkeit steigen kann. Des Weiteren sollte die Persönlichkeit effektiv entwickelt werden, da man sonst Gefahr läuft, dass die Persönlichkeit uneinheitlich und unberechenbar ist. Dies würde die User-Experience nachhaltig stören, da sich Menschen an komplexen und unberechenbaren Persönlichkeiten stören [10, S. 84].

Zur Entwicklung der Persönlichkeit kann man sich auf die beiden wichtigsten Persönlichkeitsmerkmale Extraversion und Verträglichkeit beschränken [10, S. 98]. Dabei sollte der Chatbot eine stark ausgeprägte Verträglichkeit haben. Der Nutzer

³ Nutzbarkeit

sollte den Chatbot als rücksichtsvoll, sympathisch und hilfsbereit empfinden, damit der Chatbot allgemein positiv wahrgenommen werden kann. Das Persönlichkeitsmerkmal Extraversion ist jedoch diverser zu betrachten. Das ist vor allem darin begründet, dass Menschen mit einer dominanten, extrovertierten Persönlichkeit einer ebenso dominanten Person positive Eigenschaften zusprechen. Genauso verhält es sich mit introvertierten Personen, die mit einer ebenso introvertierten Person interagieren [10, S. 96].

Nicht jede Person wird die Persönlichkeit des Chatbots also ansprechend finden, da die Menschheit bei den großen Persönlichkeitsmerkmalen gleich verteilt ist. Deswegen sollte man bei der Entwicklung der Persönlichkeit das Ziel des Chatbots betrachten und auf dieser Basis die Persönlichkeit entwickeln. Da der Chatbot dem User bei der Rezeptsuche hilft und somit die Rolle des Instructors übernimmt, hat er die Persönlichkeit eines aktiven Helfers, die freundlich und dominant ist [10, S. 98].

Neben den Texten kann auch der Name des Chatbots die Persönlichkeit widerspiegeln. Allein die Attribute, die automatisch einem Produkt mit männlichem Namen zugeschrieben werden, unterscheiden sich drastisch von denen, die einem Produkt mit weiblichen Namen zugeschrieben werden. Da Designprinzipien wie dieses der Gleichberechtigung der Geschlechter absolut im Weg stehen und es schwierig ist, einen Namen zu finden, der bei keinem Menschen negative Erinnerungen hervorruft, hat der Rezepte-Chatbot keinen Namen und wird lediglich mit dem abstrakten Produktnamen „Robolicious“ in Verbindung gebracht.

4.5.2 Entwicklungsprozess der Persönlichkeit und Texte

Ähnlich wie beim App- und Webseiten-Design sollten die Chatbot-Texte und die Persönlichkeit nicht vom Entwickler, sondern von einem Künstler entworfen werden. Genau wie User-Interface-Designer sich mit Farben auskennen, kennen sich Autoren mit der Sprache aus. Des Weiteren entwerfen Autoren in jedem ihrer Bücher gleich mehrere Charaktere, also Persönlichkeiten. Sie sind somit dazu qualifiziert einheitliche und starke Charaktere zu entwickeln. Hinzu kommt, dass Autoren Unterhaltungskünstler sind und dementsprechend in der Lage sind spannende Persönlichkeiten zu entwickeln, die eine Geschichte erzählen.

Die Texte und die Persönlichkeit des Rezepte-Chatbots wurden von Maximilian Stoll entwickelt. Herr Stoll ist Student der Buch- und Medienproduktion an der Hochschule für Technik, Wirtschaft und Kultur Leipzig und ist Mitglied der hochschuleigenen Improvisationstheatergruppe „Schwarzpulver“. Zuvor studierte er in Rostock und sein erster Roman „Das Leben in Technicolor“ basiert auf den zwei Jahren, die er dort verbrachte. Das Buch erschien 2015 und ist ein „Coming of Age“-Roman, der vom Stolpern, Fallen und Finden einer ganzen Generation erzählt. Momentan schreibt er an seinem zweiten Roman mit dem Arbeitstitel „Mein Verstand das Kriegsgebiet“.

Nachdem der Konversationsfluss und der Chatbot fertig entwickelt waren, fand ein Workshop mit Herrn Stoll statt, bei welchem ihm der Use-Case und die grundsätzliche Idee eines Chatbots nahegebracht wurden. Des Weiteren wurden ihm die Vorgaben für die Persönlichkeit aus Kapitel 4.5.1 erklärt und der Konversationsfluss. Herr Stoll entwickelte dann mit Hilfe einer Tabelle mit provisorischen Texten alle für den Chatbot benötigten Texte.

Herr Stoll machte den Chatbot zu einem Franzosen, mit der Begründung, dass Nutzer gute Küche mit französischer Küche gleichsetzen würden. Über Nebensätze, in welchen er Aktivitäten oder Orte beschreibt, wird die Persönlichkeit des Chatbots zusätzlich gefestigt. Diese Nebensätze sind beispielsweise in folgenden Texten zu finden.

Begrüßung

I just had the tastiest engine oil ever but I'm pretty sure you want something else in your human tummy. What do you feel like eating today?

Verabschiedung

One last tip: Don't burn your fingers on the stove. Have a wonderful meal, my friend. I'll go now and watch »The Matrix«... just for research, you know?

Antwort auf „Wie geht es dir“

I took a walk with my robot dog through the park. We had a wonderful time. Are you ready for some cooking?

Begrüßung am Abend

Bonsoir my friend! Treat yourself and cook something tasty! Just ask me and I'll look for a recipe inside my ancient library.

Vor allem der Humor des Chatbots sticht bei den meisten Texten hervor. Verstärkt wird das durch die Witze, die der Chatbot neben der Rezeptsuche erzählen kann. Außerdem ist er häufig selbstironisch, indem er auf sich als Roboter und auf frühere Technologien anspielt. Die Selbstironie trägt ebenfalls dazu bei, dass der Nutzer daran erinnert wird, dass der Chatbot weder ein echter Mensch ist noch alles verstehen kann.

Fallback-Text

Mon Dieu! I saw your lips moving but couldn't understand what you were saying. But Iu2019ll try to learn! Meanwhile, try something different like » I want a recipe for banana pancakes.«

Text bei Ernährungsform-Einstellungen

Before we go on could you tell me something about your diet? In case you're vegetarian I wouldn't want to show you the 3-layer Ultra-beef burger I just recently discovered.

4.5.3 Verwendung von Emojis

Neben der Text-Sprache nutzt der Rezepte-Chatbot auch Emojis, um seinen Texten mehr Ausdruck zu verleihen. Da Emojis heutzutage bei der textbasierten Konversation zwischen Menschen nicht mehr wegzudenken sind, muss auch ein Chatbot nicht auf den Vorteil der Emojis verzichten. Durch sie können Emotionen, denen Text nicht genügend Ausdruck verleiht, besser vermittelt werden.

Die weltweit meist genutzten Emojis auf Twitter können über die Webseite www.emojitracker.com abgerufen werden. Man sollte sich in den Chatbot-Texten zur Unterstreichnung von Emotionen auf die beliebteren Emojis beschränken. Der Rezepte-Chatbot nutzt beispielsweise das beliebte lächelnde Gesicht vereinzelt in Texten:



Neben der Verwendung von Emojis ist es außerdem von Vorteil auf einige der beliebtesten Emojis reagieren zu können. Der Rezepte-Chatbot erkennt in einem roten Herz zum Beispiel Zuneigung und sendet einen entsprechenden Text zurück:



So kann man vermeiden, dass der Chatbot bei einfach zu deutenden Emojis sein Unverständnis äußert.

Die Nutzung von Emojis sollte sich auf einzelne Texte beschränken und im Onboarding⁴ vermieden werden. Des Weiteren sollten ausschließlich positive Emojis und keine Emojis wie das ernste Gesicht verwendet werden [14]:



Auf der anderen Seite kann es jedoch Sinn haben auch negative Emojis zu erkennen, um aufsteigende Frustration besser zu erkennen und direkt deeskalieren zu können.

⁴ Die Vorstellung des Chatbots bei einem neuen Nutzer

4.5.4 Verwendung von Menüs und Buttons

In einigen Situationen während der Konversation kommt es zu komplizierteren Auswahlverfahren oder es gibt Einstellungen, die dem Nutzer immer zur Verfügung stehen sollten. In diesen Situationen ist es sinnvoll zu „schummeln“ und die UI-Elemente der jeweiligen Plattform effizient zu nutzen [15]. Facebook erlaubt es neben simplen Textnachrichten auch Nachrichten mit Buttons zu schicken. Des Weiteren kann über die Messenger-API ein beständiges Menü entwickelt werden, auf welches der Nutzer über sein Eingabefeld zugreifen kann.

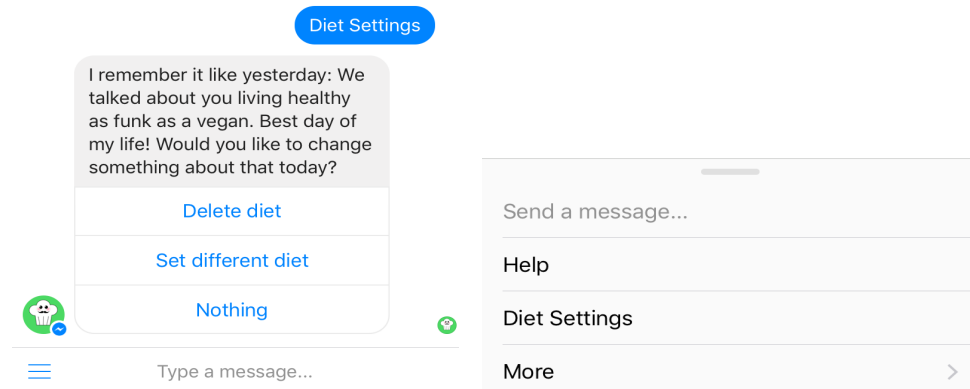


Abbildung 21: Button-Nachricht bei den Ernährungsform-Einstellungen und beständiges Menü des Rezepte-Chatbots

Der Rezepte-Chatbot setzt vor allem bei den Einstellungen von Intoleranzen und der Ernährungsform Buttons ein, da der Nutzer sich nur per Klick entscheiden muss, was die Usability des Chatbots erheblich verbessert. Außerdem kann über ein beständiges Menü Hilfe angefordert werden und die Einstellungen zu Intoleranzen und der Ernährungsform abgerufen werden. Auch der Link zur Umfrage dieser Arbeit befindet sich in diesem Menü.

Leider sind die Anzahl der Elemente einer Buttonnachricht auf 3 und bei dem beständigen Menü auf 5 begrenzt. Das führt dazu, dass beispielsweise das Einstellen einer Ernährungsform verkompliziert wird, da der Nutzer diese frei eingeben kann. Einfacher wäre es, die möglichen Ernährungsformen direkt als Button zur Verfügung zu stellen.

Auf der Basis dieser Regelwerke konnte der Rezepte-Chatbot entwickelt und eine abgeschlossene und effiziente Konversation entworfen werden.

5 Umfrage

Um den Rezepte-Chatbot und die in dieser Arbeit entstandenen Regeln bewerten zu können, wurde eine Umfrage erstellt, die Nutzer des Chatbots ausfüllen konnten.

5.1 Konzeption

Die Umfrage teilt sich in vier Teile auf. Der erste Teil beschäftigt sich mit der Chatbot-Persönlichkeit. Im zweiten Teil steht der Use-Case im Vordergrund. Der dritte Teil besteht aus Fragen zur zukünftigen Nutzung und im vierten Teil hat der Nutzer die Möglichkeit ein Feedback im Freitext abzugeben.

Der erste Teil der Umfrage besteht aus vier Fragen. Drei Fragen beschäftigen sich mit der Persönlichkeit des Chatbots. Über die Auswertung dieser Ergebnisse soll eine Aussage getroffen werden, ob die Entwicklung der Persönlichkeit, wie in Kapitel 4.5 beschrieben, erfolgreich war.

In der letzten Frage der Umfrage soll der Nutzer seine eigene Persönlichkeit evaluieren, indem er bei fünf Eigenschaften angeben muss, inwiefern diese auf ihn zutreffen. Die Eigenschaften sind dabei so gewählt, dass sie alle das Persönlichkeitsmerkmal Extraversion messen. So soll eine Aussage darüber getroffen werden, ob eine extrovertierte Chatbot-Persönlichkeit für introvertierte Nutzer störend ist, wie in Kapitel 4.5 beschrieben wurde.

Mit dem zweiten Teil soll abgeschätzt werden, ob der Nutzer der Zielgruppe des Use-Case entspricht. Die Zielgruppe des Chatbots sind Menschen, die viel selber kochen und sich gerne damit beschäftigen. Des Weiteren ist der Chatbot auf Menschen zugeschnitten, die einer bestimmten Ernährungsform folgen oder Unverträglichkeiten haben. So wird der Nutzer als erstes gefragt, ob er gerne kocht. Des Weiteren wird er nach seiner Ernährungsform und Unverträglichkeiten gefragt. Im Vergleich mit der Datenbank kann man außerdem abschätzen, wie viele Nutzer diese Chatbot-Funktionen gefunden und genutzt haben.

Im dritten Teil soll der Nutzer dann einen Vergleich zwischen dem Chatbot und einer vergleichbaren mobilen App ziehen und bewerten, ob er den Chatbot einer App vorziehen würde. Diese Frage hat einerseits das Ziel allgemein etwas über die Qualität des Chatbots aussagen zu können. Andererseits soll damit auch abgeschätzt werden können, inwiefern die Nutzer das Konzept von Konversations-Interfaces gegenüber herkömmlichen Apps annehmen.

In der zweiten Frage wird gefragt, ob der Nutzer den Chatbot erneut nutzen würde. Über diese Frage soll wie mit der ersten Frage eine Aussage getroffen werden können, wie die Qualität des Chatbots wahrgenommen wurde. Dabei ist davon auszugehen, dass Nutzer, die den Chatbot erneut verwenden würden, mit der Qualität des Chatbots zufrieden waren.

Im letzten Teil der Umfrage wird der Nutzer um weiteres Feedback gebeten, welches er frei formulieren kann. Über dieses Feedback lassen sich vor allem Probleme, die der Nutzer mit dem Chatbot hatte, benennen. So kann man die Fragen aus Teil 3 besser bewerten und abschätzen, wieso der Chatbot als gut oder schlecht befunden wurde.

Die Umfrage befindet sich im Anhang der Arbeit.

5.2 Rahmenbedingungen

Die Umfrage wurde gleichzeitig mit dem Chatbot veröffentlicht. Man gelangte über das Chatbot-Menü und über einen Facebook-Post, in dem der Chatbot vorgestellt wurde, auf die Umfrage. Des Weiteren verabschiedete sich der Chatbot gelegentlich mit einem Text, der den Nutzer noch einmal bat an der Umfrage teilzunehmen.

Die Umfrage konnte über einen Zeitraum von drei Wochen beantwortet werden. In diesen drei Wochen nahmen 34 Menschen an der Umfrage teil.

Da der Chatbot nur über Facebook verfügbar war, konnten nur Facebook-Nutzer bei an der Umfrage teilnehmen. Dementsprechend war die Demografie der Nutzergruppe dadurch bereits eingeschränkt. Der Chatbot wurde hauptsächlich von Freunden und Freundes-Freunden ausprobiert. So war ein großer Teil der Tester zwischen 20 und 30 Jahren alt. Außerdem handelte es sich zu einem großen Teil um Studenten.

Da der Chatbot und die Umfrage in der englischen Sprache verfasst waren, bestand die Möglichkeit der Teilnahme für jeden, der der englischen Sprache mächtig war.

5.3 Auswertung

Die Auswertung der Umfrage erfolgte unterteilt in den vier Teilen, in die die Umfrage aufgeteilt war. Jedoch bezieht sich die Auswertung eines Teils oftmals auf die der anderen Teile.

5.3.1 1. Teil: Chatbot-Persönlichkeit

Bei der Frage mit welchen Adjektiven die Nutzer den Chatbot umschreiben würden, wurde der Chatbot vor allem als kontaktfreudig (47,1%), enthusiastisch (64,7%), sympathisch (58,8%) und hilfsbereit (58,8%) beschrieben.

sociable	forceful	enthusiastic	outgoing	shy	reserved	quiet	indefinite
16	4	22	13	0	0	1	2
47,1	11,8	64,7	38,2	0	0	2,9	5,9

Abbildung 22: Auswertung der Extraversion - gelb = stark ausgeprägt & blau = schwach ausgeprägt

sympathetic	helpful	trusting	considerate	careless	harsh	rude	cold
20	20	4	1	2	2	1	2
58,8	58,8	11,8	2,9	5,9	5,9	2,9	5,9

Abbildung 23: Auswertung der Verträglichkeit - grün = stark ausgeprägt & orange = schwach ausgeprägt

Daraus lässt sich also schließen, dass die entwickelte Persönlichkeit des Chatbots die gewünschte Wirkung erzeugt hat. Wie in Kapitel 4.5 beschrieben, sollte die Chatbot-Persönlichkeit extrovertiert und verträglich sein.

Nur drei Tester haben dem Chatbot die Attribute gefühllos, barsch, unfreundlich und unachtsam zugewiesen. Diese Attribute stehen alle für eine geringe Verträglichkeit. Dabei haben zwei der Tester den Chatbot mit zwei und drei dieser vier Attribute beschrieben. Schaut man sich die restlichen Ergebnisse dieser drei Tester an, sieht man, dass sie den Chatbot allgemein nicht gut bewertet haben.

Bei der Frage, ob sie sich vorstellen könnten mit dem Chatbot auch im echten Leben zu sprechen, gaben 50% der Tester „Vielleicht“ und 32,4% „Ja“ an.

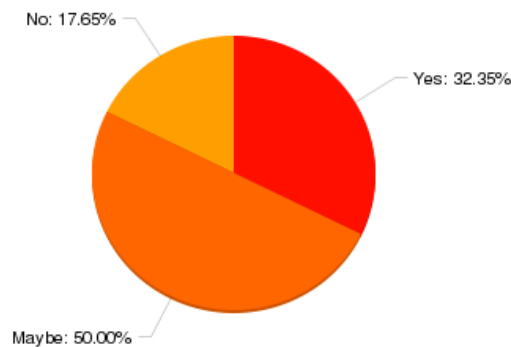


Abbildung 24: Ergebnis der Frage: „Would you like talking to the chatbot about other topics than recipes in real life?“

Einerseits erkennt man daran, dass der Chatbot offensichtlich nicht als menschlich oder real empfunden wurde. Jedoch unterstützt dieses Ergebnis auch die Glaubwürdigkeit der Chatbot-Persönlichkeit, da nur 17,6% der Tester diese Frage verneint haben.

Weiterhin haben sich 85,3% der Befragten vom Chatbot respektiert gefühlt. 11,8% haben sich lediglich teilweise respektiert gefühlt und nur eine Person hat sich vom Chatbot nicht respektiert gefühlt. Das ist ein weiterer Indikator dafür, dass der Chatbot verträglich auf die Nutzer gewirkt hat und unterstützt das Ergebnis aus Frage 1.

In der letzten Frage des ersten Teils sollten die Nutzer ihr eigene Persönlichkeit einschätzen. 58,8% der Tester haben sich selbst als eher extrovertierte Persönlichkeit beschrieben und 23,5% schätzten sich eher introvertiert ein. Bei 17,6% der Befragten war das Ergebnis nicht eindeutig. Die Persönlichkeit des Nutzers hat sich jedoch keinesfalls wie angenommen auf die Bewertung des Chatbots niedergeschlagen. Die Nutzer, die den Chatbot nicht wieder nutzen würden, verteilten sich auf alle drei Persönlichkeitsgruppen gleich (extrovertiert, introvertiert, nicht eindeutig).

Insgesamt kann man aus diesem Teil schließen, dass die Entwicklung der Persönlichkeit, wie sie in Kapitel 4.5 beschrieben wurde, erfolgreich war. Des Weiteren lässt sich sagen, dass auch introvertierte Nutzer einen Chatbot als sympathisch empfinden, obwohl er nicht ihrer Persönlichkeit entspricht. Hier fehlt jedoch ein Vergleichswert. Um darüber eine finale Aussage treffen zu können, müsste man eine introvertierte Persönlichkeit entwickeln und den Chatbot erneut und nach Möglichkeiten von derselben Personengruppe testen lassen.

5.3.2 2. Teil: Use-Case und Zielgruppe

Über die erste Frage des zweiten Teils kommt man zum Ergebnis, dass 2 der 34 Befragten nicht gerne kochen. Das bedeutet, dass 94,1% der Tester grundsätzlich zur Zielgruppe des Chatbots gehören.

11 der 34 Befragten gaben in Frage 2 eine Ernährungsform an. 9 dieser Leute gaben an sich vegetarisch beziehungsweise vegan zu ernähren. Die beiden anderen gaben andere Ernährungsformen an, die der Chatbot nicht bedienen konnte. Im Vergleich bewerteten die Befragten, die sich vegan oder vegetarisch ernähren, den Chatbot besser als der Rest der Befragten. Von den 9 gaben alle an, dass sie den Chatbot erneut nutzen würden und nur 22,2% würden eine App dem Chatbot vorziehen. Vergleichsweise sagten 13% der Befragten, die keine Ernährungsform angaben, dass sie den Chatbot nicht mehr verwenden würden und 30,4% bevorzugten eine App. Schlecht fiel das Ergebnis bei den beiden Befragten aus, die eine andere Ernährungsform angaben. Einer von ihnen würde den Chatbot nicht erneut nutzen wollen, und beide gaben an eine App zu bevorzugen.

Der Chatbot kam also einerseits sehr positiv bei der eigentlichen Zielgruppe an. Andererseits wurde er besonders negativ bewertet, wenn er nicht das hielt, was er versprach. Das ist darauf zurückzuführen, dass der Chatbot keine Aussage darüber trifft, welche Ernährungsformen er beispielsweise unterstützt.

Zumindest 7 der 9 Veganer und Vegetarier haben ihre Ernährungsform in der Datenbank hinterlegt. Diese Funktion des Chatbots war also über das Interface sehr gut auffindbar.

Anders verhält es sich bei den Unverträglichkeiten der Nutzer. Hier gaben 6 der 34 Befragten an eine Unverträglichkeit auf bestimmte Lebensmittel zu haben. 4 davon gaben dabei an Laktose intolerant zu sein. Die anderen beiden gaben an eine Unverträglichkeit auf Fruktose und Pestizide zu haben. Lediglich Laktose konnte der

Chatbot jedoch auswerten und bei der Rezeptsuche verwenden, da Spoonaculars API die anderen beiden Unverträglichkeiten nicht unterstützt.

Jedoch ist in der Datenbank keiner der 4 Laktose Unverträglichkeiten vermerkt, was darauf hindeutet, dass diese Funktion nicht offensichtlich genug kommuniziert wurde. Das liegt vor allem daran, dass die Funktion nur über das Menü aufrufbar war und kein Chatbot-Text auf sie hinwies.

5.3.3 3. Teil: Zukünftige Nutzung des Chatbots

Ob sie den Chatbot einer vergleichbaren App vorziehen würden, beantworteten 67,65% der Nutzer mit „Ja“.

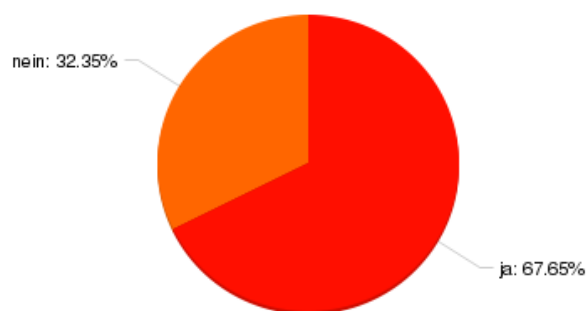


Abbildung 25: Ergebnis der Frage: „Would you prefer the recipe chatbot to a comparable app?“

Daraus lässt sich schließen, dass die Nutzer einer Bewegung weg von Apps und hin zu Chatbots offen gegenüberstehen.

Ganze 88,24% der Befragten gaben an den Chatbot in Zukunft wieder zu benutzen.

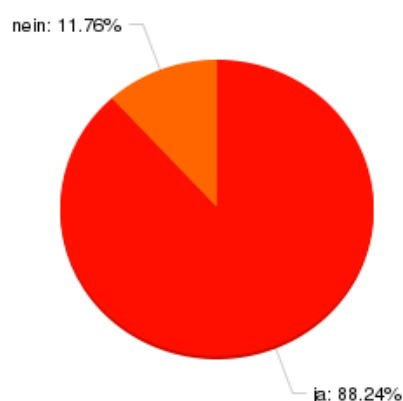


Abbildung 26: Ergebnis der Frage: „Would you use the chatbot again in the future?“

Auffällig ist, dass die Nutzer, die bei dieser Frage „Nein“ angaben, vor allem die Chatbot-Persönlichkeit im ersten Teil der Umfrage eher negativ bewerteten. Neben Attributen für geringe Verträglichkeit und Introvertiertheit gaben diese Nutzer alle an, dass sie mit dem Chatbot im realen Leben nicht sprechen würden. Die Bereitschaft den Chatbot wieder nutzen zu wollen, scheint also stark mit der emotionalen Empfindung gegenüber des Chatbots zusammenzuhängen.

5.3.4 4. Teil: Freies Feedback

Beim Auswerten des freien Feedbacks wird ersichtlich, was genau die Befragten als gut beziehungsweise schlecht empfunden haben. Ein großer Teil der Befragten kritisiert weniger den Chatbot als das, was er an Rezepten lieferte. Viele schienen mit den Rezepten nicht immer zufrieden gewesen zu sein. Schnell wird ersichtlich, dass es viele stört, dass der Chatbot Negationen nicht versteht.

i told him i really didnt like pasta. he only gave me pasta recipes.
es wäre gut, wenn er "without" o.ä. erkennen würde. als ich "without kale"
eingegeben hab, habe ich weiterhin Rezepte damit bekommen...

Des Weiteren gaben viele Nutzer an, dass die Rezepte nicht immer zufriedenstellend waren und sie gerne mehr Möglichkeiten gehabt hätten die Suchfilter zu modifizieren.

It should be possible to modify the search after the first results are shown!
When looking for a salad, all the recipes were with Kale which I don't like and
the bot couldn't understand that it shouldn't be in it. Other than that, marvelous work!
Was mir bei der Suche jedoch fehlt oder ein wenig zu bemängeln ist, sind die
Suchfilter oder auch die generelle Auswahl im Ergebnis. Wenn ich z.B. nach
Pizza gesucht habe, hat er mir hauptsächlich Rezepte mit Blumenkohlteig angezeigt und nicht mit Hefeteig.

Aus der Umfrage geht also ein überwiegend positives Ergebnis hervor, welches im nächsten Kapitel zu einem abschließenden Fazit beiträgt.

6 Fazit

Is everything going to become a bot? I don't think so. There'll probably still be static apps for professional and authoring tools; Photoshop and CAD and Excel and Evernote aren't going away; and video games will still be video games. Most of the squishy stuff in the middle, though, will go conversational. Anything that involves collaboration, communications, consumption, organization, etc. will probably become a bot. I think bots will replace 80% of what we use at work and half of what we use at home. That's a lot of stuff to rewrite!

[1]

Insgesamt lässt sich sagen, dass der Chatbot ein gelungener Prototyp ist. Die Resonanz der Umfrage war sehr positiv und die Menschen scheinen ein großes Interesse an Chatbots zu haben.

Was sich über die Testperiode klar abzeichnete ist, dass viele Konversationen mit dem Chatbot daran scheiterten, dass er in den meisten Fällen zu wenig Kontext hatte. Die Rezeptsuche stellte für den Chatbot eine Handlung dar, die in dem Moment abgeschlossen war, in dem er dem Nutzer die Rezepte lieferte. Viele Nutzer stellten aber Folgefragen zu den Rezepten. Diese konnte der Chatbot in den meisten Fällen nicht korrekt verarbeiten. Zur Weiterentwicklung des Chatbots wäre es also unumgänglich eine Art Sitzungszustand einzubauen, den der Chatbot zu jedem Nutzer in der Datenbank hinterlegt. Wenn dieser Sitzungszustand auf Rezeptsuche steht, müsste der Chatbot sich zusätzlich die Sucheingabe merken, die der Nutzer initial bei der Rezeptsuche angibt. So könnte man über wit.ai die Entität „query“ in mehrere Entitäten unterteilen. Es wäre dann möglich weitere Filterkriterien der Spoonacular Food API zu nutzen wie Kulinarik oder Zutaten, die nicht verwendet werden sollen. Dafür bräuchte es zusätzlich die Registrierung von Negationen, die sich ebenfalls über den Sitzungszustand realisieren ließen. Anders wäre dies nur schwer zu entwickeln, weil der Chatbot wissen müsste, wie er in dem jeweiligen Zustand mit Negationen umzugehen hat.

Weiterhin sollte der Chatbot nach Möglichkeit die Muttersprache der Zielgruppe sprechen oder eine Spracheinstellung besitzen. In vielen Konversationen zeichnete sich ab, dass es eine Sprachbarriere gibt, die dazu führte, dass der Chatbot den Nutzer teilweise nicht richtig verstand, da Rezeptsuchen in gebrochenem Englisch formuliert wurden. Es ist leider nicht möglich den Rezepte-Chatbot in einer anderen Sprache als Englisch anzubieten, da eine Ressource für deutsche Rezepte nicht auffindbar ist und eine eigene Rezeptdatenbank mit zu hohem Aufwand verbunden wäre.

Die Chatbot-Persönlichkeit und der Konversationsfluss waren jedoch gut konzipiert, was über die Umfrageergebnisse ersichtlich war. Das zeigt, dass die im Rahmen dieser Arbeit entstandenen Regeln ein guter Anfang sind, um einen Chatbot zu

entwickeln und zu designen. Fakt ist aber, dass herkömmliche UX-Designregeln nicht über Bord geworfen werden sollten. Bevor man mit dem Design der Chatbot-Persönlichkeit und der Konversation anfängt, sollten Praktiken wie User-Stories durchgeführt werden, um den Use-Case richtig auszuarbeiten. Dadurch wäre man relativ schnell darauf gekommen, dass den Nutzern Suchfilter fehlen würden.

Mit dieser Arbeit konnten Fragen zum Design und zur Entwicklung eines Chatbots erfolgreich beantwortet werden. Es wird spannend bleiben, wie sich sowohl die Technik als auch das Design von Chatbots in den nächsten Jahren weiterentwickelt und wie rasant das ganze Thema voranschreiten wird. Es ist in jedem Fall vorstellbar, dass Chatbots in den nächsten Jahren einige Apps ablösen werden und die Nutzerfreundlichkeit von Anwendungen dadurch gesteigert wird.

Abbildungsverzeichnis

Abbildung 1: Am meisten genutzten Apps in den USA und weltweit	5
Abbildung 2: Aktive Nutzerzahlen für WhatsApp und den Facebook Messenger	5
Abbildung 3: Ausschnitt aus der Konversation mit dem Rezepte-Chatbot. Intention, Parameter, Kontext und Sitzung markiert	7
Abbildung 4: UML-Komponentendiagramm des Chatbot-Systems	8
Abbildung 5: UML-Sequenzdiagramm der Interaktion zwischen dem Chatbot-Server und Facebook	10
Abbildung 6: Nachricht des Rezepte-Chatbots mit Buttons	11
Abbildung 7: Web-Oberfläche von wit.ai – Understanding	12
Abbildung 8: Web-Oberfläche von wit.ai – Inbox	13
Abbildung 9: Deploymentdiagramm des Chatbot-Backend	15
Abbildung 10: Shannon-Weaver-Modell	17
Abbildung 11: Konversationsmodell nach Pangaro	18
Abbildung 12: Ausschnitt aus dem Aktivitätsdiagramm des Rezepte-Bot: Begrüßung	24
Abbildung 13: Chatbot erklärt, dass er Rezepte suchen kann	26
Abbildung 14: Chatbot erzählt etwas zu seinem Befinden und fragt den Nutzer dann, ob er bereit für das Kochen ist	26
Abbildung 15: Fallback-Antwort des Rezepte-Chatbots	28
Abbildung 16: Chatfenster auf der Facebook-Webseite mit Hilfetext	29
Abbildung 17: Drei Nachrichten, wenn der Nutzer nach einem Rezept mit Ernährungsform fragt und keine Ernährungsform für diesen Nutzer gespeichert ist	30
Abbildung 18: Antworten auf Höflichkeitsnormen	33
Abbildung 19: Chatbot reagiert auf Zuneigung	33
Abbildung 20: Indirektes Lob des Chatbots	34
Abbildung 21: Button-Nachricht bei den Ernährungsform-Einstellungen und beständiges Menü des Rezepte-Chatbots	38
Abbildung 22: Auswertung der Extraversion - gelb = stark ausgeprägt & blau = schwach ausgeprägt	40
Abbildung 23: Auswertung der Verträglichkeit - grün = stark ausgeprägt & orange = schwach ausgeprägt	41
Abbildung 24: Ergebnis der Frage: „Would you like talking to the chatbot about other topics than recipes in real life?“	41
Abbildung 25: Ergebnis der Frage: „Would you prefer the recipe chatbot to a comparable app?“	43
Abbildung 26: Ergebnis der Frage: „Would you use the chatbot again in the future?“	43

Abkürzungsverzeichnis

HTWK	Hochschule für Technik, Wirtschaft und Kultur
KI	Künstliche Intelligenz
NLP	Natural Language Processing
API	Application Programmable Interface
URL	Uniform Resource Locator
JSON	JavaScript Object Notation
http	Hypertext Transfer Protocol
ID	Identity
CRUD	Create, Read, Update, Delete
CLEAT	Context, Language, Exchange, Agree- ment, Transaction
UML	Unified Modeling Language

Literaturverzeichnis

- [1] P. Libin, "A Charge of Bots — The Tech World As We Know It Is About To Be Rewritten," 2016. [Online]. URL: <https://medium.com/@plibin/a-charge-of-bots-9ee33bb3b868>. [Abgerufen: 14.2.2017].
- [2] A. Braun, *Chatbots in der Kundenkommunikation*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003.
- [3] J. Schwartz, "The Most Popular Messaging App in Every Country," 24-May-2016. [Online]. URL: <https://www.similarweb.com/blog/worldwide-messaging-apps>. [Abgerufen: 21.3.2017].
- [4] "Top Apps Overall." [Online]. URL: appannie.com/apps/ios/top/germany/overall. [Abgerufen: 21.3.2017].
- [5] "Company Info | Facebook Newsroom." [Online]. URL: <http://newsroom.fb.com/company-info/>. [Abgerufen: 05.3.2017].
- [6] P. Bashmakov, "Advanced Natural Language Processing Tools for Bot Makers – LUIS, Wit.ai, Api.ai and others (UPDATED)." [Online]. URL: <https://stanfy.com/blog/advanced-natural-language-processing-tools-for-bot-makers/>. [Abgerufen: 21.1.2017].
- [7] "Node.js," *nodejs.org*. [Online]. URL: <https://nodejs.org/en/>. [Abgerufen: 24.3.2017].
- [8] "Shannon and Weaver Model of Communication." [Online]. URL: <http://communicationtheory.org/shannon-and-weaver-model-of-communication/>. [Abgerufen: 24.3.2017].
- [9] H. Dubberly and P. Pangaro, "What is conversation? How can we design for effective conversation?," *dubberly.com*, 2009. [Online]. URL: <http://www.dubberly.com/articles/what-is-conversation.html>. [Abgerufen: 04.2.2017].
- [10] B. Reeves and C. Nass, *The Media Equation*. 1998.
- [11] P. Grice, *Handlung, Kommunikation, Bedeutung*. Suhrkamp Verlag, 1979.
- [12] R. R. McCrae and O. P. John, "An Introduction to the Five-Factor Model and Its Applications," 1992.
- [13] E. A. Smith and R. J. Senter, "AUTOMATED READABILITY INDEX," S. 1–22, Nov. 1967.
- [14] R. Ramos, "5 tips for writing chatbot scripts," 27-Jul-2016. [Online]. URL: <http://venturebeat.com/2016/07/27/5-tips-for-writing-chatbot-scripts/>. [Abgerufen: 28.2.2017].
- [15] M. Mariansky, "All Talk and No Buttons: The Conversational UI," 23.2.2016. [Online]. URL: <https://alistapart.com/article/all-talk-and-no-buttons-the-conversational-ui>. [Abgerufen: 02.3.2017].

Bildquellen

Abbildung 1: KPBC, „Internet Trends 2016“ – S. 109, <http://www.kpcb.com/internet-trends>

Abbildung 2: KPBC, „Internet Trends 2016“ – S. 99, <http://www.kpcb.com/internet-trends>

Abbildung 11: P. Pangaro, „Pangaro CUSO 2014 CONVERSATION“ - <http://www.pangaro.com/CUSO2014/>