



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

# **Bachelorarbeit**

Daniel Jäger

Beacons zur Erkennung von Beratungskontakten

**Daniel Jäger**

Beacons zur Erkennung von Beratungskontakten

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung

im Studiengang Technische Informatik  
am Department Informatik  
der Fakultät Technik und Informatik  
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer : Prof. Dr.-Ing. Andreas Meisel  
Zweitgutachter : Prof. Dr. rer. nat. Thomas Lehmann

Abgegeben am 3. Juni 2016

**Daniel Jäger**

**Thema der Bachelorarbeit**

Beacons zur Erkennung von Beratungskontakten

**Stichworte**

Android, Beacons, Bluetooth Low Energy, Fuzzy Logic, Kundenfeedback

**Kurzzusammenfassung**

Ziel dieser Arbeit ist es einen Proof of Concept für eine intelligente Kundenfeedback Lösung für den stationären Handel zu liefern. Durch den Einsatz von Bluetooth Low Energy Beacons soll eine Android App auf dem Kundensmartphone eigenständig Beratungskontakte zwischen Kunden und Servicemitarbeitern erkennen können, um situationsrelevante Feedback-Aufforderungen zu ermöglichen.

**Daniel Jaeger**

**Title of the paper**

Using Beacons to identify consultations

**Keywords**

Android, Beacons, Bluetooth Low Energy, Customer Feedback, Fuzzy Logic

**Abstract**

The purpose of this paper is to provide a proof of concept for an intelligent customer feedback solution for brick and mortar stores. An Android App on the customer's smartphone is, through the use of Bluetooth low energy beacons, supposed to autonomously identify consultations between customers and service personnel in order to enable feedback requests which are relevant to the situation.

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis.....</b>	<b>7</b>
<b>1 Einführung.....</b>	<b>9</b>
1.1 Motivation und Zielsetzung .....	9
<b>2 Hintergrund und Kontext.....</b>	<b>11</b>
2.1 Problemstellung.....	11
2.2 Marktsituation heute .....	12
2.2.1 Digitale automatisierte Anbieter.....	12
2.2.2 Personalbasierte offline Anbieter.....	13
2.3 Vision.....	14
2.4 Projektplanung.....	15
2.5 Überblick der eingesetzten Technologien.....	16
2.5.1 Bluetooth Low Energy.....	16
2.5.2 Beacon Sendemodule und Firmware Standards .....	17
2.5.3 Android SDK .....	18
2.5.4 Künstliche Intelligenz .....	18
<b>3 Intelligente Beratungskontakterkennung.....</b>	<b>21</b>
3.1 Auswahl geeigneter künstlicher Intelligenz.....	21
3.1.1 Analytische Verfahren.....	23

<b>Einführung</b>	<b>5</b>
3.1.2	Lern Verfahren..... 24
3.1.3	Regelbasierte Verfahren..... 26
3.1.4	Fazit ..... 31
3.2	Einführung in die Anwendung von Fuzzy Logik ..... 32
3.2.1	Veranschaulichung an einem Beispiel ..... 33
3.2.2	Fuzzy Logik Steuerung ..... 35
3.3	Beratungskontakterkennung mit Fuzzy Logik ..... 37
3.3.1	Ausführliche Analyse der verfügbaren Messdaten ..... 37
3.4	Expertenwissen Zusammenstellung ..... 39
3.4.1	Definition der Zugehörigkeitsfunktionen und linguistischen Variablen ..... 43
3.4.2	Definition der scharfen Variablen..... 48
3.4.3	Defuzzification der Ergebnisse ..... 50
3.4.4	Ausarbeitung der Fuzzy Logik Regeln ..... 52
<b>4</b>	<b>Systemanalyse..... 56</b>
4.1	Anforderungsanalyse..... 56
4.1.1	Anforderungserhebung ..... 56
4.1.2	Bewertung der Kundenanforderungen..... 58
4.2	Systemdesign und technische Konzeption..... 61
4.2.1	Festlegung der Systemgrenzen..... 61
4.2.2	High-Level Systemzustände..... 62
4.2.3	High-Level Systemabläufe ..... 64
4.2.4	Detaillierte Systemabläufe der Beratungskontakterkennung..... 67
4.2.5	Systemstruktur und statische Beziehungen ..... 69
<b>5</b>	<b>Implementierung ..... 70</b>
5.1	Android..... 70
5.2	Gimbal ..... 70
5.3	Fuzzy Logic Library..... 71
<b>6</b>	<b>Systemtest der Fuzzy Logik Implementierung ..... 73</b>
6.1	Ausgewählte Testfälle ..... 73

<b>Einführung</b>	<b>6</b>
6.2 Randomisierte Testfälle.....	75
6.3 Ergebnisse und Fazit.....	76
<b>7 Zusammenfassung und Ausblick .....</b>	<b>78</b>
<b>8 Anhang .....</b>	<b>79</b>
8.1 Fuzzy Control Language .....	79
8.2 Kurze Einführung in die jFuzzyLogic Bibliothek .....	80
<b>Quellenverzeichnis .....</b>	<b>82</b>

# Abbildungsverzeichnis

Abbildung 1: Geschäftsprozessdiagramm des Ist-Zustands aus der Kundenperspektive ....	12
Abbildung 2: Wettbewerbsüberblick im Customer Feedback Management Markt.....	13
Abbildung 3: Vision der Feedback-Lösung .....	15
Abbildung 4: Überblick der Projekt Timeline .....	16
Abbildung 5: stationärer Gimbal Beacon (Series 21) .....	17
Abbildung 6: wearable Gimbal Beacon (Series 10) .....	18
Abbildung 7: Smartphone Betriebssystem Marktanteile.....	18
Abbildung 8: Generischer zustandsfreier Funktionsblock.....	21
Abbildung 9: Blackbox Funktionsblock der intelligenten Beratungskontakterkennung .....	22
Abbildung 10: Simple Grundstruktur eines analytischen Funktionsblocks mit gegebenem Funktionsvektor .....	23
Abbildung 11: Überblick der drei Arten von Lernsituationen .....	25
Abbildung 12: Traditionelle Aussagenlogik bildet scharfe Übergänge zwischen den definierten Bereichen .....	27
Abbildung 13: Logische Verknüpfungen bei Fuzzy Mengen .....	28
Abbildung 14: Fuzzy Logik ermöglicht glatte (graduelle) Übergänge zwischen den definierten Bereichen .....	29
Abbildung 15: Fuzzylogik einer Temperaturregelung.....	34
Abbildung 16: Beispiel einer zusammengesetzten Zugehörigkeitsfunktion .....	35
Abbildung 17: Überblick der typischen Struktur einer Fuzzy Logik Steuerung (FLS) .....	36
Abbildung 18: Distanzeinschätzung mit der logarithmischen RSSI Skala des Gimbal SDK ...	38
Abbildung 19: Zugehörigkeitsfunktion der Kunde-Mitarbeiter Distanz mit drei linguistischen Variablen (METRO).....	44
Abbildung 20: Nichtlineare Zugehörigkeitsfunktion der Distanz.....	45
Abbildung 21: Zugehörigkeitsfunktion der Beratungs-Zeitspanne mit drei linguistischen Variablen (METRO).....	46
Abbildung 22: Nichtlineare Zugehörigkeitsfunktion der Beratungs-Zeitspanne .....	47
Abbildung 23: Zugehörigkeitsfunktion für gemeinsames durch den Laden Gehen während der Beratung.....	48
Abbildung 24: Singularität für wiederholtes Betreten der sehr nahen Distanz.....	49
Abbildung 25: Verringerung der Durchschnittsdistanz als Singularität .....	50
Abbildung 26: Simple Ja/Nein Aufteilung der Entscheidung über einen potenziellen Beratungskontakt.....	51
Abbildung 27: Fuzzy Logik Ausgabewert der Beratungskontakterkennung mit fünf linguistischen Variablen.....	52

---

Abbildung 28: Gewünschtes Anwendungsszenario bei Metro .....	60
Abbildung 29: Gewünschtes Anwendungsszenario bei Porsche .....	60
Abbildung 30: Systemgrenzen und Anwendungsfälle für die Feedback Sammlung .....	61
Abbildung 31: Zukünftige zusätzliche Systemfunktionen .....	62
Abbildung 32: Zustände des Systems (App Lifecycle).....	64
Abbildung 33: Aktivitätsdiagramm der High Level Abläufe des Feedback Systems .....	66
Abbildung 34: Detaillierte Abläufe der intelligenten Beratungskontakterkennung mit Fuzzy Logik Implementation.....	68
Abbildung 35: Klassendiagramm der mobilen Feedbacklösung .....	69
Abbildung 36: Datenfluss der Gimbal SDK Kommunikation .....	71
Abbildung 37: Graphische Darstellung des Fuzzy Logik Ergebnis mit nichtlinearen Zugehörigkeitsfunktionen für Testfall 1.....	74
Abbildung 38: Zufällige Eingabewertkombinationen sortiert nach dem Wahrheitsgrad eines Beratungskontaktes.....	75
Abbildung 39: Streudiagramm der Distanz im Verhältnis zum Beratungs-Wahrheitsgrad für zufällige Eingabewertkombinationen .....	76
Abbildung 40: Datenaustausch zwischen Programmen mit Fuzzy Control Language .....	80



# 1 Einführung

## 1.1 Motivation und Zielsetzung

Wissenschaftliche Weiterentwicklungen der Technik sind eine treibende Kraft hinter den Veränderungen in den Lebensgewohnheiten der Menschen. Nach der mobilen Revolution bilden heute Neuerungen aus dem Bereich der verteilten Systeme im Rahmen des Internets der Dinge (IoT) eine wichtige Speerspitze der Innovation. Das IoT ermöglicht es alten Problemstellungen mit neuen Lösungsansätzen zu begegnen. Zuvor rein analoge Lebensbereiche können zunehmend durch digitale Systeme augmentiert werden.

*"Das Ziel des Internets der Dinge ist es, die Informationslücke zwischen der realen und virtuellen Welt zu minimieren."<sup>1</sup>*

Der Mensch soll so unmerklich durch immer kleinere, eingebettete Systeme bei seinen alltäglichen Tätigkeiten unterstützt werden. Besonders im Bereich des "Wearable Computing"<sup>2</sup> wurde in den letzten Jahren erhebliche Marktakzeptanz erzielt und neu entwickelte Hardware findet zunehmend Einzug in die Welt der Konsumgüter. Durch die Kombination von heute weit verbreiteten Mobilgeräten mit neuer IoT Hardware können innovative Systeme geschaffen werden, die völlig neue Anwendungsszenarien in traditionellen Bereichen des Lebens ermöglichen. Die praxistaugliche Zusammenführung moderner Techniken zu wirtschaftlich tragfähigen Anwendungen für den Mainstream bildet die Schnittstelle zwischen Wissenschaft und Gesellschaft.

Ein zentraler Bereich, in dem ein akuter Bedarf für Innovation besteht, ist der traditionelle stationäre Handel. Die Art wie wir Einkaufen hat sich in den letzten Jahren stark gewandelt. Filialisten im stationären Handel leiden unter anhaltendem Kundenschwund durch Online-Abwanderung<sup>3</sup>. Die persönliche Beratung durch Fachverkäufer ist einer der wichtigsten Gründe warum Kunden noch im Filialgeschäft kaufen<sup>4</sup>. Allerdings fehlt es an ausreichend spezifischen und kontinuierlichen Feedbackdaten über das Kundenerlebnis und die Beratungsqualität. Bestehende digitale Touchpad Feedbacklösungen liefern nur unspezifische Daten mit geringer Entscheidungsrelevanz, während traditionelle Lösungen,

---

<sup>1</sup> Vgl. Fleisch, Elgar; Mattern, Friedemann: Das Internet der Dinge, Berlin: Springer, 2005

<sup>2</sup> Computersystem das während der Anwendung am Körper des Benutzers getragen wird, um ihn in seiner Tätigkeit zu unterstützen ohne, dass der Benutzer das das Gerät dabei direkt bedient

<sup>3</sup> Statista Branchenreport Einzelhandel 2015

<sup>4</sup> Statista 2016: Warum kaufen Kunden im Laden

wie persönliche Kundenbefragungen und Mystery Shopping, nur sporadische Daten liefern und zudem erhebliche Kosten verursachen.<sup>5</sup>

Um der Kundendaten-Übermacht der Onlinehändler Paroli bieten zu können, bedarf es neuer Konzepte, um Veränderungen auf der Verkaufsfläche besser messbar zu machen. Die durch Mobilgeräte bereits gegebene Rechnerallgegenwart bildet die Grundlage, es in Zukunft auch dem traditionellen Handel zu ermöglichen, kontinuierliche datengesteuerte Innovation und Optimierung zu nutzen um konkurrenzfähig zu bleiben. Die Zielsetzung dieser Arbeit ist es, eine innovative Kundenfeedback Lösung zu konzipieren, die es dem stationären Handel erlaubt, datengesteuert seine Vorteile im Bereich Kundenerlebnis zu optimieren. Um eine praxistaugliche Lösung zu entwickeln gilt es dafür zwei Kernpunkte zu lösen:

- Implementation einer innovativen, technischen Lösung zur automatisierten Sammlung von spezifischen Kundenfeedbackinformationen auf der Verkaufsfläche
- Integration des Feedback-Gebens als einen situationsrelevanten, interaktiven und bequemen Teil des Einkaufserlebnisses

Der erste Punkt befasst sich damit, dass die Lösung möglichst eigenständig und mit hohem Automatisierungsgrad kontinuierlich Daten sammelt, damit Händler sich auf ihre Kernkompetenzen konzentrieren können, anstatt sich mit der Feedbacksammlung beschäftigen zu müssen. Des Weiteren müssen diese Daten ausreichend spezifisch sein, damit sie für konkrete Maßnahmen auf der Verkaufsfläche entscheidungsrelevant sind. Der zweite Punkt stellt sicher, dass Endkunden die Lösung auch tatsächlich nutzen wollen und somit einen reichhaltigen Datenstrom ermöglichen.

Im Folgenden wird ein akademischer Prototyp für ein solches Kundenfeedbacksystem ausgearbeitet. Fokus ist eine möglichst unmerkliche Einbindung des Feedbackgebens in das Einkaufserlebnis. Durch systematisches Vorgehen im Sinne der Softwaretechnik wird eine zielorientierte und methodische Entwicklung des Systems verfolgt. Diese Arbeit bildet die Grundlage eines „Gründungsprojektes aus der Wissenschaft“ am Startup Dock der Technischen Universität Hamburg-Harburg unter dem Projektnamen „Feedbadge“<sup>6</sup>. Im Rahmen des weitergefassten Gründungsprozesses soll diese akademische Arbeit einen grundsätzlichen Proof of Concept<sup>7</sup> liefern, um als Grundlage für den weiteren Ausbau der Lösung zu einem marktreifen Produkt dienen zu können.

---

<sup>5</sup> Anforderungsanalyse bei Pilotkunden und Validierungsgespräche mit Branchenexperten

<sup>6</sup> <http://www.tuhh.de/startupdock/startup-dock/startups.html>

<sup>7</sup> Wirksamkeitsnachweis

## 2 Hintergrund und Kontext

### 2.1 Problemstellung

Die Arbeitserfahrung des Autors bildet die Grundlage für die Problemfeststellung und Vision einer Lösung. Als Fachverkäufer im Schlafzimmerbereich bei der Höffner Möbelgesellschaft wurde deutlich, dass in der aktuellen Situation spezifisches Kundenfeedback nur äußerst selten das Management erreicht. Kunden, die sich zum Servicemanager durchfragen oder den Namen eines Fachverkäufers aufschreiben, um eine E-Mail an die allgemeine Service Adresse des Unternehmens zu schicken, tun dies fast ausschließlich aufgrund extrem schlechter Kundenerlebnisse - also um sich zu beschweren. Dieses sporadische und sehr einseitige Feedback schlägt bei Mitarbeitergesprächen zur Leistungsbeurteilung als einsamer faktischer Datenpunkt zur Kundenzufriedenheit übermäßig stark zu Buche. Negatives Feedback kann kaum durch faktisches positives Kundenfeedback relativiert werden. Leistungsbewertungen und Personalentscheidungen müssen somit oftmals aufgrund subjektiver Eindrücke des Managers getroffen werden.

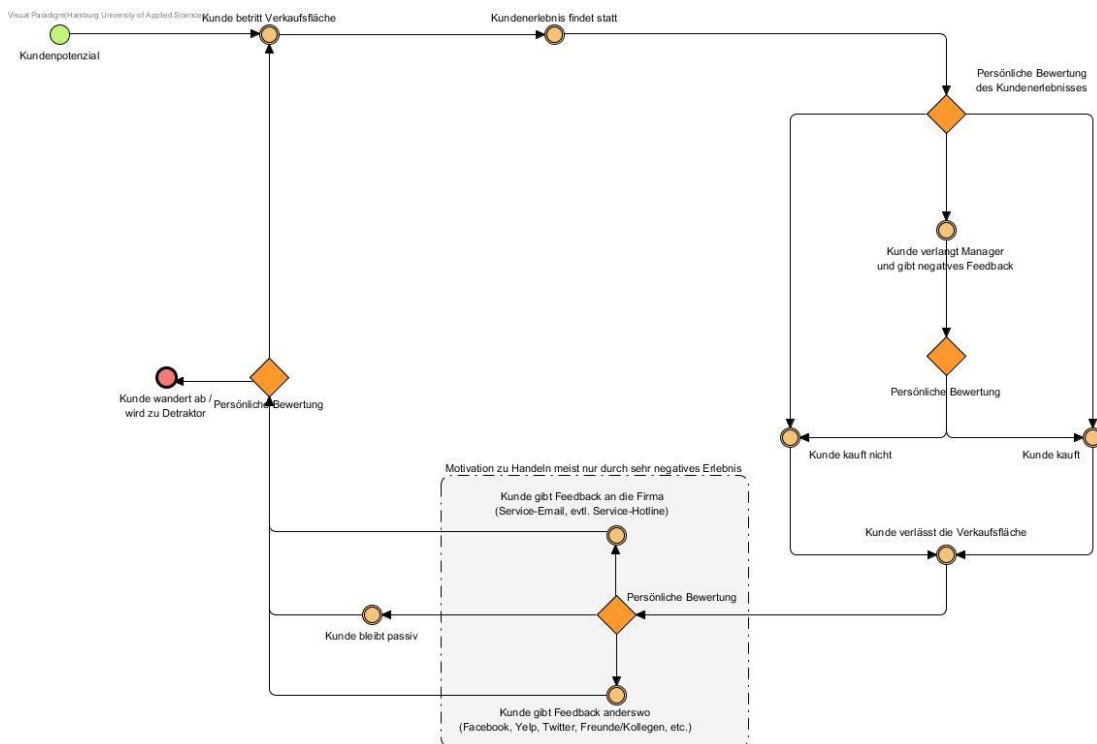


Abbildung 1: Geschäftsprozessdiagramm des Ist-Zustands aus der Kundenperspektive

Das Ziel ist eine neue technische Lösung zur Vereinfachung und Automatisierung des Feedbackprozesses für Kunden im stationären Handel nach dem Vorbild der weitreichenden Feedbackkultur im Online- und Telefonversandhandel. Hierfür ist es wichtig, die zentralen „Painpoints“ des traditionellen Feedbackprozesses zu lösen, um kontinuierlich ausreichend Kunden zu spezifischem Feedback zu motivieren, auch wenn sie keine extrem schlechte Erfahrung hatten. Ausgangspunkt einer Systemanalyse für ein solches verbessertes System ist zunächst eine Bestandsaufnahme der aktuellen Marktsituation existierender Lösungen.

## 2.2 Marktsituation heute

Existierende Feedback Lösungen für den stationären Einzelhandel stellen einen erheblichen Mehraufwand für Kunden dar und schaffen es nicht, verlässliche spezifische Daten kontinuierlich zur Verfügung zu stellen. Wettbewerber spalten sich in zwei generelle Lager:

### 2.2.1 Digitale automatisierte Anbieter

Digitale Touchscreen Feedback Kioske an sogenannten „Choke Points“ (meistens dem Ausgang) fordern vom Kunden nach Beendigung seines Einkaufs am Ausgang anzuhalten,

wohlmöglich seine Einkäufe abzustellen, um unspezifisches Feedback an einem öffentlichen Bildschirm zu geben. Diese Kioske bieten zudem keinerlei Schutz gegen mehrfach Benutzung und missbräuchliche Beeinflussung (z.B. durch Servicemitarbeiter).

### 2.2.2 Personalbasierte offline Anbieter

Persönliche Kundenbefragungen leiden unter Interviewer-Bias und liefern sporadische, stichprobenhafte Daten zu einem erheblichen Preispunkt. Der Einsatz von sogenannten Mystery Shoppern (auch Testkäufern) produziert ebenfalls nur sporadische und stichprobenhafte Daten die von bezahlten Agenten stammen anstelle von echten Kunden. Zudem entstehen durch Mystery Shopper Spionage erhebliche Kosten durch die Bindung von Personal sowie durch die negativen Effekte auf das Arbeitsklima und das Image des Unternehmens.

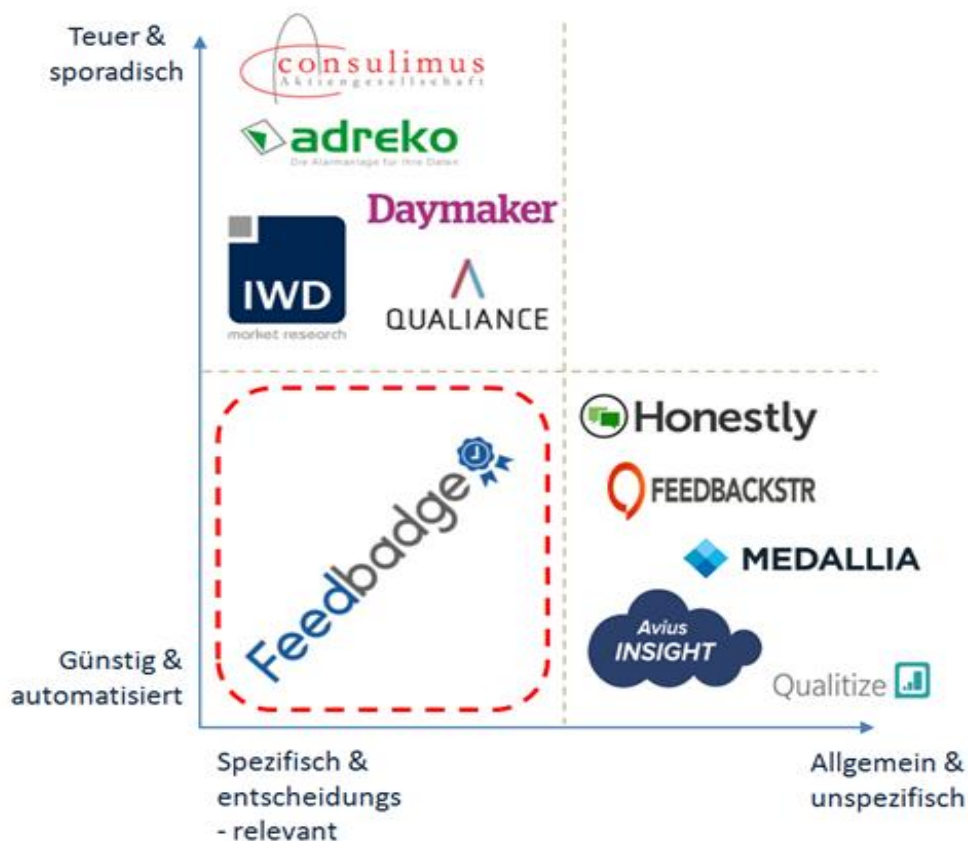


Abbildung 2: Wettbewerbsüberblick im Customer Feedback Management Markt

In dieser Arbeit wird eine neue Feedbacklösung mit dem Namen Feedbadge entwickelt, welche die Probleme der bestehenden Wettbewerber löst und Daten liefern kann, die nicht

nur automatisiert und kosteneffektiv gesammelt werden, sondern zugleich auch spezifisch und entscheidungsrelevant sind.

### 2.3 Vision

Der Feedback-Sammelprozess soll zu einem natürlichen Teil des Einkaufsprozesses für Kunden werden. Dazu muss ein bequemer und sehr situationsrelevanter Ablauf realisiert werden, den die Kunden als interaktive Bereicherung ihres Einkaufserlebnisses wahrnehmen. Dazu soll das technische System möglichst eigenständig arbeiten und Entscheidungen treffen, um Bequemlichkeit und Situationsrelevanz zu bieten. Wenn der Kunde sich der Verkaufsfläche nähert, soll das System eigenständig aktiv werden ohne, dass der Kunde (oder der Händler) etwas dafür tun muss. Das System soll dann eigenständig erkennen können, ob und wann ein Beratungskontakt zwischen dem Kunden und einem Servicemitarbeiter stattgefunden hat. Dazu müssen die Servicemitarbeiter vom System digital erfasst werden können. Wird ein feedbackwürdiger Beratungskontakt erkannt, können dem Kunden prägnante und situationsangemessene Feedbackfragen gestellt werden. Die Abteilung des beratenden Servicemitarbeiters und die Dauer der Beratung sollen ebenfalls automatisch registriert werden. Es ist naheliegend, das System als App auf den Smartphones der Kunden laufen zu lassen und Servicemitarbeiter über sogenannte Beacons<sup>8</sup> für das System identifizierbar zu machen. Der Laden selbst, genauer der Kassenbereich, muss ebenfalls mit einem Beacon ausgestattet werden.

Wenn sich der Kunde anschließend der Kasse bzw. dem Ausgang nähert, soll ihm eine kurze Feedback-Aufforderung direkt auf sein Smartphone gestellt werden. Als zusätzlichen Anreiz können dem Kunden Bonuspunkte oder ein Sofortrabatt im Austausch für gegebenes Feedback angeboten werden. Manager der Händler können die gesammelten Daten zur Auswahl eines angemessenen Anreizes im Sinne der Unternehmens-Rabattstrategie nutzen. Über die Unique Device Identification (UDI) des Smartphones können Daten spezifischen Nutzern zugeordnet werden und mit der Zeit wird die Erstellung reichhaltiger Kundenprofile möglich. Des Weiteren sind die resultierenden Feedbackdaten spezifischen Servicemitarbeitern und ihren Abteilungen zuzuordnen.

---

<sup>8</sup> Engl. für Leuchtfeuer. Handliche Sendemodule die per Funktechnik zyklisch ein kleines Datenpaket senden.



Abbildung 3: Vision der Feedback-Lösung

Abbildung 3 stellt einen Durchgang dieser äußeren Abläufe des Systems schematisch dar. Diese Blackbox Beschreibung der Funktionen des Systems wird im Folgenden technisch aufgearbeitet und zunehmend weiter verfeinert.

## 2.4 Projektplanung

Ziel des gesamten Gründungsprojekts ist es, eine generelle Vision für eine innovative Lösung, basierend auf einem wissenschaftlichen Prototypen und im kontinuierlichen Austausch mit Pilotkunden, schrittweise zur Marktreife zu bringen. Die in dieser akademischen Abschlussarbeit zu leistende Einzelarbeit, soll im Anschluss eine sinnvolle Grundlage für die kundennahe Weiterentwicklung des Produkts in Teamarbeit bilden. Da es sich um ein in der Entstehung befindliches Innovationsprodukt handelt, ist Flexibilität und schnelle Anpassungsfähigkeit entscheidend für einen erfolgreichen Markteintritt. Bei den bereits abgeschlossenen Arbeiten im Vorfeld dieser akademischen Arbeit hat sich bereits eine flexible Philosophie im Sinne agiler Projektmethoden wie Scrum und Kanban als

zielführend erwiesen. Es ist vorgesehen im Anschluss an diese akademische Arbeit im Team zu entscheiden nach welcher konkreten Methodik weiter vorgegangen werden soll.

In Abbildung 4 werden die Projektaktivitäten dieser akademischen Arbeit im Kontext der Timeline des gesamten Projektes eingeordnet. Im Vorlauf dieser Bachelorarbeit wurde bereits die Problemfeststellung auf der Verkaufsfläche im stationären Einzelhandel vollzogen. Ein konkretes Geschäftsmodell wurde im Sommer 2015 ausgearbeitet und mit dem Gewinn des B2B Startup Weekend Warsaw (unter dem Projektnamen Custibio) grundsätzlich validiert.<sup>9</sup>

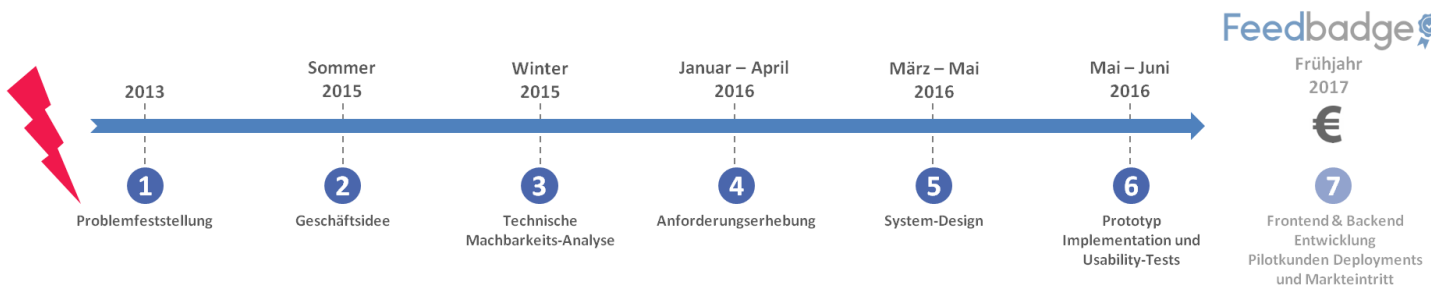


Abbildung 4: Überblick der Projekt Timeline

Ein erste technische Machbarkeits-Analyse wurde im Winter 2015 im Rahmen der Themendefinition dieser Arbeit vollzogen. Die folgenden Abhandlungen erarbeiten die Schritte 4 bis 6 der Timeline. Das in dieser Arbeit entwickelte System bildet anschließend die Grundlage für die geplante Weiterentwicklung der Lösung zur Marktreife in Zusammenarbeit mit Pilotkunden. Der offizielle Markteintritt ist für das Frühjahr 2017 geplant.

## 2.5 Überblick der eingesetzten Technologien

Zur Realisierung der Implementierung des Systems werden Technologien aus den Bereichen Mobile Computing, Wearable Computing und der künstlichen Intelligenz kombiniert. Im Rahmen dieser Arbeit wird dabei auf gängige und weit verbreitete Standards und serienmäßig produzierte handelsübliche Produkte zurückgegriffen. Im Folgenden werden die einzelnen Technologien kurz vorgestellt.

### 2.5.1 Bluetooth Low Energy

Bluetooth ist ein IEEE Industriestandard für eine Funktechnik zur kabellosen Datenübertragung zwischen Geräten im Nahbereich (ungefährer Radius von bis zu 100 Metern). Der Bluetooth Low Energy (BLE) Standard vom Dezember 2009 ist eine

<sup>9</sup> Startup Weekend Warsaw 2015, <http://swwaw.com/>, 2.3.2016



Weiterentwicklung des Bluetooth Standards mit deutlich geringerem Stromverbrauch und geringeren Kosten aber einem ähnlichen Kommunikationsbereich.<sup>10</sup>

### 2.5.2 Beacon Sendemodule und Firmware Standards

Der Oberbegriff Beacon (engl. Leuchtfeuer) beschreibt handliche Transmitter Sendemodule (Signalgeber) die, in Übereinstimmung mit dem jeweils eingesetzten Standards, in regelmäßigen Abständen ein kurzes BLE Funksignal senden. Dabei werden gleichbleibende elementare Informationen übermittelt. Mit dem jeweiligen Protokollstandard wird der Universally Unique Identifier (UUID) des Beacons, und wahlweise weitere Informationen, gesendet. Der bekannteste Beacon Standard ist der 2013 von Apple eingeführte, proprietäre iBeacon Standard.

In dieser Arbeit werden, von in 2014 von Qualcomm eingeführte, Gimbal Beacons verwendet (siehe Abbildungen 5 und 6). Diese Hardware ist kostengünstig und bietet Flexibilität in Bezug auf die Anpassungsfähigkeiten des Systems durch die Möglichkeit alternative Firmware zu wählen und anzupassen. Gimbal Beacons unterstützen neben dem hauseigenen Gimbal Firmware Standard auch serienmäßig die beiden populärsten Beacon Standards mit iBeacon und den in 2015 von Google vorgestellten EddyStone Standard. Alle Gimbal Beacons werden durch austauschbare handelsübliche Batterien betrieben. Als wearable Beacons werden die Gimbal Series 10 mit dem Gimbal Standard eingesetzt. Diese haben eine ungefähre Akkulaufzeit von drei Monaten und kosten etwa 5 USD pro Stück. Als stationäre Beacons werden die Gimbal Series 21 mit dem iBeacon Standard eingesetzt. Diese haben eine durchschnittliche Akkulaufzeit von 18 Monaten und kosten etwa 30 USD pro Stück.



Abbildung 5: stationärer Gimbal Beacon (Series 21)

---

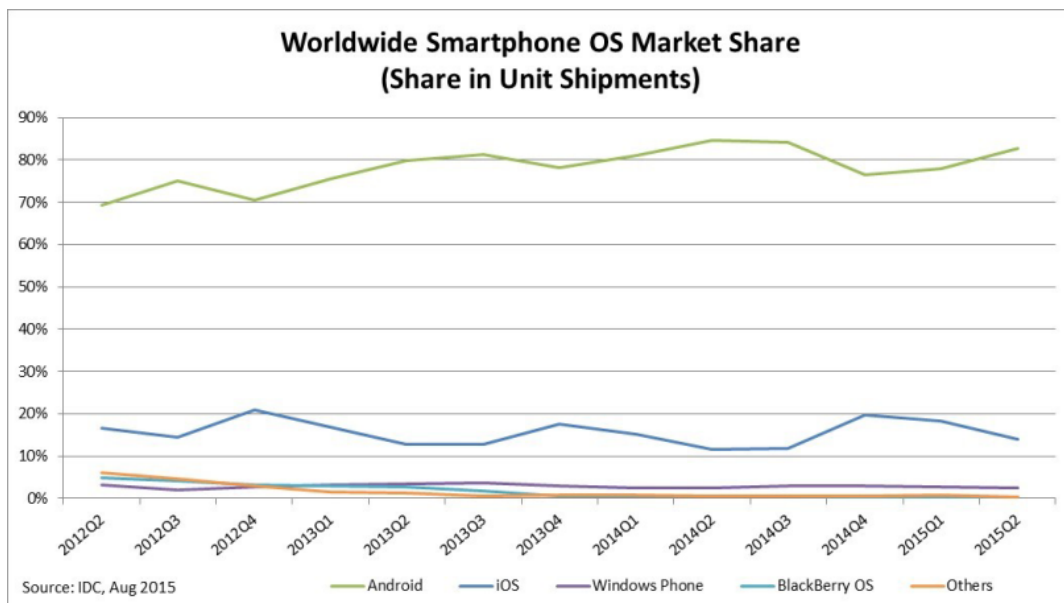
<sup>10</sup> Wikipedia: Bluetooth Low Energy. [https://de.wikipedia.org/wiki/Bluetooth\\_Low\\_Energy](https://de.wikipedia.org/wiki/Bluetooth_Low_Energy) (18. April 2016)



Abbildung 6: wearable Gimbal Beacon (Series 10)

### 2.5.3 Android SDK

Aufgrund der deutlich geringeren Entwicklungskosten und einer erheblich höheren Verbreitung mit über 80% Marktanteil wird der Prototyp des Feedback Systems in dieser Arbeit für das Android Betriebssystem entwickelt. Um die BLE und iBeacon Standards einsetzen zu können, muss die minimale Android SDK Version auf 4.3 gesetzt werden. Niedrigere Android Versionen können das in dieser Arbeit entwickelte Feedback System nicht nutzen.

Abbildung 7: Smartphone Betriebssystem Marktanteile<sup>11</sup>

### 2.5.4 Künstliche Intelligenz

Der Begriff künstliche Intelligenz wird in Fachkreisen als nicht mehr wirklich zeitgemäß angesehen und es mangelt an einer eindeutigen Definition, um die hier gemeinten

<sup>11</sup> International Data Corporation (IDC), August 2015

Techniken klar abzugrenzen. Dennoch soll künstliche Intelligenz in diesem Projekt als ein allgemein verständlicher Sammelbegriff dienen, der grundsätzlich den Versuch der Automatisierung intelligenten Verhaltens beschreibt. Das heißt, den Versuch eine menschenähnliche Intelligenz nachzubilden und ein System so zu entwerfen und zu programmieren, dass dieses eigenständig Probleme bearbeiten und sinnvolle Entscheidungen fällen kann. In diesem Projekt ist es notwendig Ideen und Konzepte unterschiedlichen nicht-technischen Stakeholdern<sup>12</sup> auf einem intuitiven Niveau verständlich zu machen, um effektiv über die Lösung diskutieren zu können. Der Begriff der künstlichen Intelligenz hat sich hierbei als geeignet bewiesen und er stellt zudem ein attraktives Schlagwort dar, das beim Vertrieb der Lösung hilfreich ist.

Ziel des Feedbacksystems ist es die Feedbackaufforderungen möglichst situationsrelevant zu gestalten. Dazu ist eine möglichst korrekte Erkennung von Beratungskontakten zwischen Kunden und Beratern notwendig. Diese Erkennung soll vollautomatisiert und in angemessener Zeit auf dem Smartphone der Einkäufer stattfinden. Um dies zu realisieren, bieten sich Techniken der künstlichen Intelligenz an. Dabei gilt es eine Lösung zu finden, die eine präzise und schnelle Erkennung ermöglicht, ohne unnötig hohen Ressourcenverbrauch auf den Mobilgeräten der Kunden zu verursachen

Die Herausforderung besteht darin, dass die Unterscheidung von feedbackwürdigen Beratungskontakten und unbedeutenden Begegnungen zwischen Einkäufern und Servicemitarbeitern eine inhärent unscharfe und teilweise subjektive Einschätzung ist. Es besteht daher eine gewisse Ungenauigkeit rund um den Begriff „feedbackwürdiger Beratungskontakt“, die als eine Art Wahrheitsverteilung verstanden werden kann. In gewissen Situationen sind die zur Erkennung genutzten Input Parameter eindeutig genug, um mit hoher Sicherheit festzustellen, ob ein relevanter Beratungskontakt stattgefunden hat oder nicht. In solchen Fällen ist der Wahrheitsgrad hoch. So ist es zum Beispiel sehr wahrscheinlich, dass wenn ein Einkäufer und ein Servicemitarbeiter sich für zwanzig Minuten auf Armlänge zusammen durch den Laden bewegt haben, ein feedbackwürdiger Beratungskontakt stattgefunden hat. Problematisch wird die Erkennung nur in den unscharfen Übergangsbereichen und Grenzfällen an den Rändern der eindeutigen Fälle. In diesen unscharfen Bereichen muss eine abwägende Entscheidung getroffen werden, die auf den Werten mehrerer relevanter Input-Variablen basiert.

Im Einkaufsszenario bieten menschliche Erfahrungen erhebliches Vorwissen darüber, was als relevanter Beratungskontakt zu bewerten ist. Besonders die verfügbaren Erfahrungen der Branchenexperten der Pilotkunden von Feedbackge, sind eine wertvolle Ressource. Diese sollte bei der Realisierung des digitalen Systems genutzt werden. Der Einsatz von Techniken der künstlichen Intelligenz bietet sich an um für diese komplexen Problemstellungen in angemessener Zeit automatisiert zu einer sinnvollen Lösung zu kommen. Diese Arbeit beginnt die Suche nach einer effektiven und effizienten Lösung dieser Problemstellung mit Hilfe etablierter Techniken der künstlichen Intelligenz, die sich in industriellen Anwendungen bewährt haben. Es muss also eine geeignet erscheinende Technik ausgewählt werden, um die Ausarbeitung eines akademischen Prototypen

---

<sup>12</sup> Engl. für Interessenvertreter

durchzuführen. Im Falle eines zufriedenstellenden Ergebnisses, kann diese Arbeit als akademischer Proof of Concept dienen und für zukünftige Erweiterungen und Optimierungen der industriellen Ausarbeitung des Produkts eine Grundlage bieten. Ob das Ergebnis zufriedenstellend ist, kann letztlich nur durch Usability Tests mit menschlichen Nutzern des Systems bestätigt werden. Im Falle eines unbefriedigenden Ergebnisses, kann der gewählte Lösungsansatz verworfen und alternative Techniken für die weitere Entwicklung in Betracht gezogen werden. Vor diesem Hintergrund wird im Folgenden darauf Wert gelegt, besonders die intelligente Beratungskontakterkennung als eigenständiges Modul zu kapseln, damit es in Zukunft mit möglichst geringem Mehraufwand modifiziert oder ausgetauscht werden kann.

In wirtschaftlichen Anwendungen sollte der Entwickler stets bemüht sein den Trade-off<sup>13</sup> zwischen Effektivität und Effizienz zu optimieren. Aus diesem Grund sollte auch bei der Entwicklung dieser Feedbacklösung die Eventualität in Betracht gezogen werden, dass ein zufriedenstellendes Ergebnis auch mit vergleichsweise einfachen Mitteln realisiert werden. Somit gilt es zu prüfen ob der Einsatz künstlicher Intelligenz tatsächlich notwendig ist und in wie fern er zu signifikant besseren Ergebnissen bei der Beratungskontakterkennung führt. Besonders durch die, im Rahmen dieser Arbeit, gewonnenen Erkenntnisse könnte es möglich sein, eine ausreichend effektive Lösung mit traditionellen Mitteln und ohne künstliche Intelligenz zu implementieren. Das würde heißen, dass die standardmäßig in Programmiersprachen verfügbaren logischen Operatoren der Aussagenlogik ausreichend sind, um eine adäquate Beratungskontakterkennung zu erschaffen. Ein Vergleich der Effektivität einer solchen traditionellen programmatischen Lösung mit der in dieser Arbeit ausgewählten künstlichen Intelligenz kann somit als Grundlage dienen, um den Trade-off beim Ausbau der Lösung zur Marktreife zu optimieren.

---

<sup>13</sup> Austauschbeziehung, <https://de.wikipedia.org/wiki/Trade-off>

# 3 Intelligente Beratungskontakterkennung

Die intelligente Beratungskontakterkennung ist das Herzstück der Feedbacklösung und entscheidend für die Effektivität und Kundenrelevanz des gesamten Systems. Die Vision der Lösung soll es ermöglichen Kunden prägnantere und entscheidungsrelevantere Feedbackaufforderungen zu stellen, als es bei existierenden Lösungen bisher möglich ist. Der Schlüssel hierfür ist eine automatisierte intelligente Beratungskontakterkennung, die mit den verfügbaren Eingabewerten in angemessener Zeit ein sinnvolles Ergebnis als Ausgabewert produziert. Basierend auf diesem Ausgabewert kann das Feedbacksystem dann die Entscheidungen treffen, wann und welche Feedbackaufforderungen in der gegebenen Situation an den Kunden zu stellen sind.

## 3.1 Auswahl geeigneter künstlicher Intelligenz

Wie zuvor entschieden, soll die intelligente Beratungskontakterkennung als eigenständiges Modul vom Rest des Systems gekapselt werden, um es leicht austauschen oder modifizieren zu können. Dafür kann die Beratungskontakterkennung als Blackbox Funktionsblock modelliert werden, der über festgelegte Schnittstellen mit dem umgebenden System kommuniziert. Abbildung 8 stellt die generische Struktur eines solchen Funktionsblocks schematisch dar.

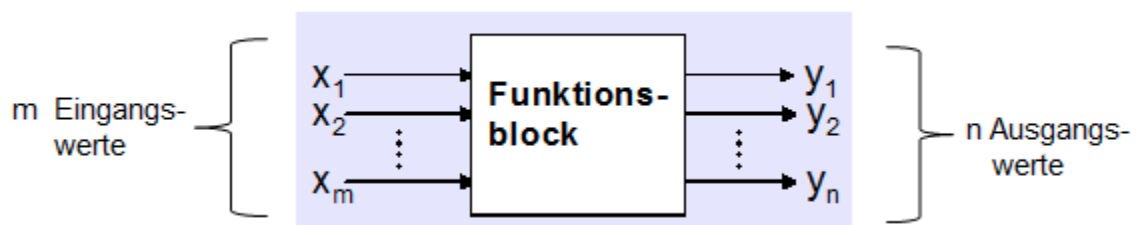


Abbildung 8: Generischer zustandsfreier Funktionsblock<sup>14</sup>

<sup>14</sup> Prof. Dr.-Ing. Andreas Meisel: *Robot Vision Skript*, Kapitel 5, Hochschule für Angewandte Wissenschaften Hamburg, April 2014

Hier ist zu sehen, dass die Ausgangswerte keine direkte Verbindung zu den Eingangswerten des Funktionsblocks haben. Der Funktionsblock ist also zustandsfrei, das heißt, die Ausgangswerte des Funktionsblocks sind nur von den gerade anliegenden Eingangswerten abhängig. Der Funktionsblock selbst wird damit auch als speicherfrei bezeichnet. Diese Struktur ist für das zu lösende Problem angemessen, da die Eingabewerte zur Einschätzung eines möglichen Beratungskontaktes nicht von zuvor anliegenden Eingabewerten abhängig sind. In Absprache mit den Branchenexperten der Pilotkunden wird angenommen, dass ein Beratungskontakt auf der Verkaufsfläche als von vorherigen Beratungskontakten völlig unabhängig anzunehmen ist.

Die Anwendung des generischen Funktionsblocks auf die Gegebenheiten des hier zu realisierenden Systems führt zu der in Abbildung 9 dargestellten Modellierung.



Abbildung 9: Blackbox Funktionsblock der intelligenten Beratungskontakterkennung

Durch den Einsatz der Mobilgeräte der Kunden in Kombination mit dem stationären sowie den wearable Beacons stehen dem System drei relevante Eingabewerte zur Verfügung.

- Die geschätzte Distanz zwischen den Personen – Kunde und Servicemitarbeiter
- Die geschätzte Distanz vom Kunden zum stationären iBeacon im Kassenbereich
- Informationen über relevante Zeitspannen während des potenziellen Beratungskontaktes

Mit diesen begrenzten Eingabewerten muss der Funktionsblock zu einer adäquaten Entscheidung kommen und einen entsprechenden Ausgabewert liefern. Die zu realisierende Ausgabe ist dabei eine Einschätzung darüber ob ein feedbackwürdiger Beratungskontakt stattgefunden hat oder nicht. Der Ausgabewert soll quasi den Wahrheitsgrad eines möglichen Beratungskontaktes widerspiegeln. Wünschenswert, aber nicht zwingend notwendig, wäre dafür natürlich eine Art Konfidenzintervall für die Ausgabe des Funktionsblocks. Das umgebende System kann dann, entsprechend der Nutzervorstellungen, eine Entscheidung fällen darüber wie eine jeweilige Ausgabe das weitere Vorgehen des Systems steuert.

Als nächstes muss nun das innere Verhalten des Funktionsblocks entwickelt werden. Es stehen grundsätzlich drei unterschiedliche Verfahren zur Verfügung um das Verhalten des Funktionsblocks festzulegen:

- analytische Verfahren
- regelbasierte Verfahren
- Lern Verfahren

Im Rahmen dieser Arbeit soll das am sinnvollsten erscheinende Verfahren ausgewählt werden um den Proof of Concept durchzuführen. Hierfür werden die Gegebenheiten der Problemstellung, die verfügbaren Daten und die Stärken und Schwächen der drei Verfahren untersucht, um so eine systematische Entscheidung zu treffen.

### 3.1.1 Analytische Verfahren

Um mit analytischen Verfahren Ausgangswerte zu erhalten, muss das Verhalten des Funktionsblocks durch eine eindeutige Zuordnung durch ein striktes mathematisches Verfahren in Form einer Funktion oder Gleichung modelliert werden. Bei Funktionen handelt es sich um eine eindeutige Beziehung, bei der jedem Element aus der Menge der Eingabewerte genau ein Element der Menge der Ausgabewerte zugeordnet wird. Bei Gleichungen wird eine eindeutige Aussage über die Gleichheit zweier Terme getroffen. Problemstellungen mit eindeutigen und exakten Lösungen lassen sich so besonders gut modellieren. Zusätzlich eignen sich Funktionen um zusätzliche typische Eigenschaften wie z.B. Homogenität oder Stetigkeit zu modellieren.

Die Problemstellung muss klar definierbar sein und dem Entwickler muss eine entsprechende Funktion oder Gleichung bekannt sein, die dann programmatisch implementiert werden kann. In trivialen Fällen können so aus Eingabewerten direkt eindeutige Ausgabewerte mithilfe einer bekannten Funktion berechnet werden.

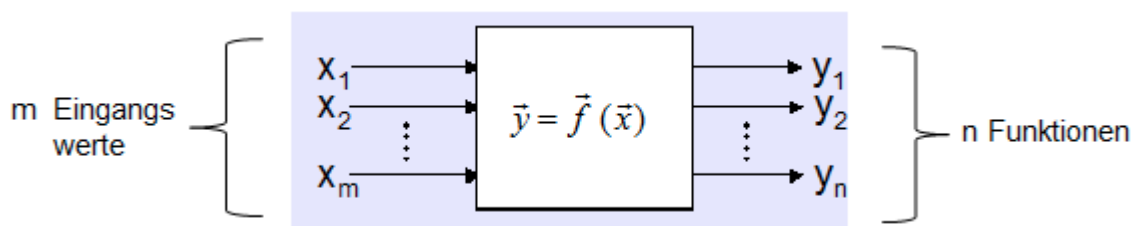


Abbildung 10: Simple Grundstruktur eines analytischen Funktionsblocks mit gegebenem Funktionsvektor

In schwierigeren Fällen kommt es vor, dass notwendige Größen unbekannt sind. Dann müssen mathematische Lösungsverfahren eingesetzt werden, um die Unbekannten zu finden. Bei linearen Gleichungen kann z.B. im Fall von unbekanntem Ausgangsgrößen das Problem als Nullstellensuche formuliert und durch inverse affine Transformation gelöst werden. Im Fall eines nichtlinearen Gleichungssystems kann beispielsweise eine iterative Nullstellensuche für mehrere Unbekannte versucht werden.

In jedem Fall aber muss der Entwickler das Problem strikt mathematisch beschreiben können. Wird mithilfe eines analytischen Verfahrens ein exaktes Ergebnis gefunden, so ist dessen mathematische Korrektheit auch beweisbar. Die Verifizierung der Lösung ist somit sichergestellt. Allerdings bieten analytische Verfahren keine Beweismöglichkeit für die Validität der Ergebnisse in Bezug auf die reale Problemstellung.

Bei der Erkennung von Beratungskontakten ist eine eindeutige und exakte Zuordnung zwischen Eingabewerten und Ausgabewerten sehr schwierig. Die Übergänge zwischen feedbackwürdigen Beratungskontakten und unwichtigen Begegnungen sind fließend, subjektiv und nicht immer einem eindeutigen Ergebnis zuzuordnen. Auch mit der Unterstützung von Branchenexperten ist es nicht möglich eine strikte mathematische Beziehung für diese Problemstellung zu formulieren.

### 3.1.2 Lern Verfahren

Mit maschinellem Lernen kann das Verhalten des Funktionsblocks festgelegt werden, indem das System durch geeignete Beispieldaten einen funktionalen Zusammenhang zwischen den Eingangswerten und den Ausgangswerten erlernt. Dazu ist es notwendig, dass auch tatsächlich ein solcher Zusammenhang besteht, damit dieser aus den zur Verfügung gestellten Beispieldaten erlernbar ist. Der Entwickler benötigt des Weiteren eine relativ große Menge an geeigneten Trainingsdaten um das System zu trainieren. Das lernende System passt sein Verhalten schrittweise an die gegebenen Trainingsdaten an und kann anschließend auch für neue Eingabedaten passende Ergebnisse liefern. Diese Fähigkeit zur Generalisierung ist entscheidend für die Effektivität des Systems. Der Entwickler muss daher darauf achten, dass er dem System eine ausreichende Anzahl von Varianten in Form von repräsentativen Beispielen zum Lernen zur Verfügung stellt. Des Weiteren muss darauf geachtet werden, das System nicht zu exakt auf die Beispiele anzulernen, um Overfitting<sup>15</sup> zu vermeiden und die Generalisierungsfähigkeit zu erhalten.

Der grundsätzliche Ablauf der schrittweisen Adaption des Verhaltens durch lernen durch Beispiele ist immer ähnlich. Dabei können drei Lernsituationen unterschieden werden, die in Abbildung 11 dargestellt sind.

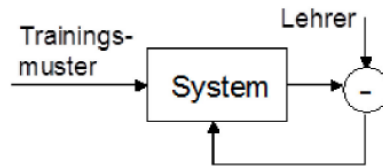
---

<sup>15</sup> Prof. Dr.-Ing. Andreas Meisel: *Robot Vision Skript*, Kapitel 5.4, Hochschule für Angewandte Wissenschaften Hamburg, April 2014



**Überwachtes Lernen**

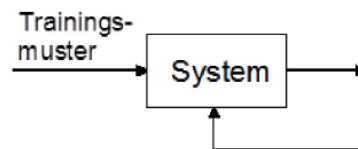
= "supervised learning"  
= Lernen mit Lehrer



Ein Lehrer sagt, was die richtige Ausgabe gewesen wäre.

**Unüberwachtes Lernen**

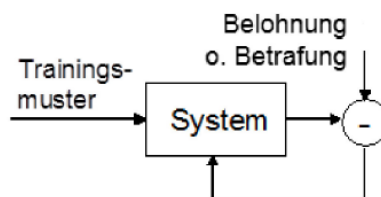
= "unsupervised learning"  
= Lernen ohne Lehrer



Das System bemerkt selbst, dass es unterschiedliche Eingangsklassen gibt.

**Bestärkendes Lernen**

= "reinforcement learning"  
→ Lernen durch Erfahrung



Der Lehrer (Leben / Umgebung) belohnt oder bestraft.

Abbildung 11: Überblick der drei Arten von Lernsituationen <sup>16</sup>

Es wird ersichtlich, dass bei allen Formen des Lernens das System sein Verhalten iterativ an die präsentierten Beispiele anpasst. Hierbei kann ein Lehrer vorgeben was die richtige Ausgabe für ein Beispiel gewesen wäre oder gewünschtes bzw. unerwünschtes Verhalten incentivieren. Alternativ kann auch unüberwacht gelernt werden, wobei das System den funktionalen Zusammenhang eigenständig erlernen muss.

Lernende Systeme eignen sich für Probleme, bei denen es einen funktionalen Zusammenhang zwischen Eingaben und Ausgaben gibt, dieser aber nicht genau bekannt oder zu komplex ist, um direkt vom Entwickler modelliert zu werden. Dafür muss der Entwickler eine erhebliche Menge an Beispieldaten zur Verfügung haben, um das System anlernen zu können und anschließend einen Teil der Beispieldaten einzusetzen, um das erlernte Verhalten zu testen. Dabei gibt es keinen exakten systematischen Prozess, dem der Entwickler folgen kann um ein optimales Ergebnis zu erzielen. Das Anlernen gleicht somit eher einer Kunst als einer Wissenschaft. Es gilt auch zu bedenken, dass die genaue Funktionsweise des resultierenden Systems relativ undurchsichtig ist. Der durch lernen modellierte Zusammenhang zwischen Eingaben und Ausgaben erschließt sich dem Entwickler nicht direkt. Das kann Änderungen bzw. Anpassungen des Systems erschweren. Im Fall von Kundenfeedback ist der Zusammenhang zwischen Eingaben und Ausgaben weitgehend bekannt und die notwendige Komplexität der Modellierung für ein

<sup>16</sup> Prof. Dr.-Ing. Andreas Meisel: *Robot Vision Skript*, Kapitel 5.4, Hochschule für Angewandte Wissenschaften Hamburg, April 2014

zufriedenstellendes Ergebnis scheint begrenzt. Des Weiteren wäre die Beschaffung von einer großen Menge an Beispieldaten relativ aufwendig und impraktikabel. Wirklich repräsentative Beispiele könnten eigentlich nur dadurch erzeugt werden, dass menschliches Verhalten bei verschiedensten möglichen Beratungssituationen auf der tatsächlichen Verkaufsfläche aufgezeichnet würde. Dies würde einen enormen Zeitaufwand bedeuten und der Zugang zur tatsächlichen Verkaufsfläche, um Beispieldaten zu sammeln, ist durch die Pilotkunden sehr begrenzt. Das Verhalten des Funktionsblocks durch Lernen festzulegen ist somit nicht sinnvoll.

### 3.1.3 Regelbasierte Verfahren

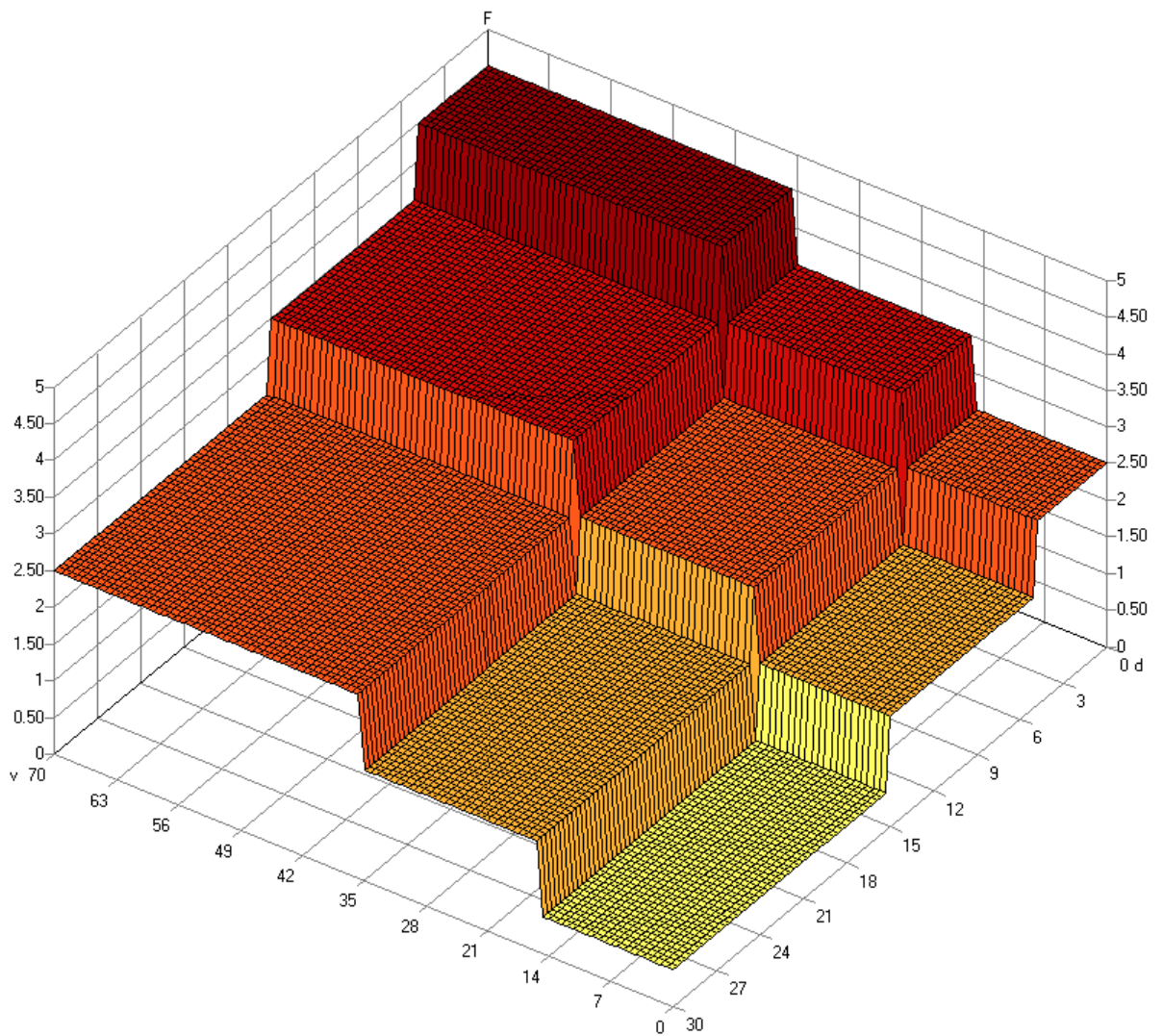
Regelbasierte Verfahren modellieren den funktionalen Zusammenhang zwischen Eingaben und Ausgaben durch ausdrückliche Regeln. Regeln sind dabei logische Ausdrücke, die für bestimmte Eingabewerte die zugehörigen Ausgabewerte festlegen. Das Verhalten des Funktionsblocks bei regelbasierte Verfahren ist somit sehr anschaulich und für den Menschen direkt verstehbar. Zudem ist eine Implementierung des Funktionsblocks mit Regeln grundsätzlich sehr einfach realisierbar.

Um regelbasierte Verfahren sinnvoll anwenden zu können, muss zunächst ausreichend relevantes Vorwissen verfügbar sein. Basierend auf diesem Vorwissen muss der Entwickler schließlich explizite Regeln formulieren, mit denen das System logisch von Eingabewerten auf die entsprechenden Ausgabewerte schließen kann. Diese Regeln werden in Form von logischen Aussagen formuliert und programmatisch durch Bedingungsausdrücke (IF / ELSE Statements) implementiert. Voraussetzung für regelbasierte Verfahren ist also, dass das Problem ausreichend vom Entwickler verstanden wird, um sinnvolle, explizite Regeln aufstellen zu können, die ausreichend sind um das Problem adäquat zu modellieren. Bei Problemstellungen mit hochdimensionalen Eingangsvektoren und erheblicher Komplexität kann es zu sogenannter Regelexplosion kommen. In diesen Fällen und bei geringem Verständnis des funktionalen Zusammenhangs und der Abläufe durch den Entwickler sind regelbasierte Verfahren schlecht geeignet.

Allerdings existiert für die hier zu realisierenden Erkennung von Beratungskontakten erhebliches Vorwissen der Branchenexperten. Zudem ist die zu erwartende Komplexität bei nur drei Eingabewerten und einem einzelnen Ausgabewert relativ gering. Daher scheint es grundsätzlich möglich und sinnvoll ausdrückliche Regeln aufzustellen, um eine effektive Beratungskontakterkennung zu realisieren.

Die Schwierigkeit der Problemstellung liegt vielmehr in den unscharfen Übergängen zwischen den durch die Regeln definierten Bereichen. Eine harte Unterscheidung zwischen den potenziellen Beratungskontakten als entweder völlig wahr oder völlig falsch scheint nicht realistisch. Außerdem könnten die sehr harten Übergänge zwischen den durch die Regeln definierten Bereichen zu paradoxen Ergebnissen führen. Die Anwendung von binärer Aussagenlogik ist demnach nicht ausreichend, um die Realität des funktionalen Zusammenhangs sinnvoll zu modellieren. Das sprunghafte Funktionsgebirge bei einer

Modellierung durch scharfe Übergänge bei traditioneller Aussagenlogik wird in Abbildung 12 graphisch am Beispiel einer regelbasierten Bremskraftsteuerung dargestellt.



**Abbildung 12: Traditionelle Aussagenlogik bildet scharfe Übergänge zwischen den definierten Bereichen<sup>17</sup>**

Um neben den sehr harten Übergängen der binären Interpretation auch graduelle Übergänge zu ermöglichen, muss die klassische Aussagenlogik erweitert werden. Grundlage hierfür ist eine Erweiterung des traditionellen Mengenbegriffs, in welchem ein Element entweder in einer bestimmten Menge enthalten ist oder nicht. Um diese kategorische

<sup>17</sup> Prof. Dr.-Ing. Andreas Meisel: *Robot Vision Skript*, Kapitel 5.3.1, S.21, Hochschule für Angewandte Wissenschaften Hamburg, April 2014

Unterscheidung zu erweitern, bedarf es der Einführung des Fuzzy Begriffs (engl. *fuzzy*, „unscharf“, „verschwommen“, „unbestimmt“). Die Implikationen des Fuzzy Begriffs und die dadurch ermöglichten Erweiterungen, lassen sich am besten durch den Vergleich bzw. die Abgrenzung zu den zugehörigen traditionellen Begriffen verständlich machen.

Traditionelle Mengen erfassen nur ob ein Element in einer bestimmten Menge enthalten ist oder nicht. Damit gibt es nur eine absolute Wahrheit über die Zugehörigkeit zu einer Menge. Auf dieser Grundlage erweitern Fuzzy Mengen dieses Konzept um die Möglichkeit die Zugehörigkeit graduell beschreiben zu können. Der Wahrheitsbegriff ist damit nicht mehr exklusiv und kategorisch sondern gleicht einer Wahrheitsverteilung. Über eine Zugehörigkeitsfunktion<sup>18</sup> kann jedem Element  $X$  einer Definitionsmenge ein reeller Wert auf dem Intervall null bis eins zugeordnet werden, der den Grad der Zugehörigkeit dieses Elements zur Zielmenge widerspiegelt. Die Zugehörigkeitsfunktion ist demnach allgemein definiert als  $\mu_A: X \rightarrow [0,1]$ . Eine Implikation dessen ist, dass ein Element gleichzeitig zu mehreren Mengen zugehörig sein kann. Da auch immer ein Grad dieser Zugehörigkeit ( $\mu_A(x)$ ) angegeben ist, können die grundlegenden logischen Verknüpfungen weiterhin angewendet werden. Die Anwendung dieser Operatoren auf Fuzzy Mengen ist in Abbildung 13 dargestellt.

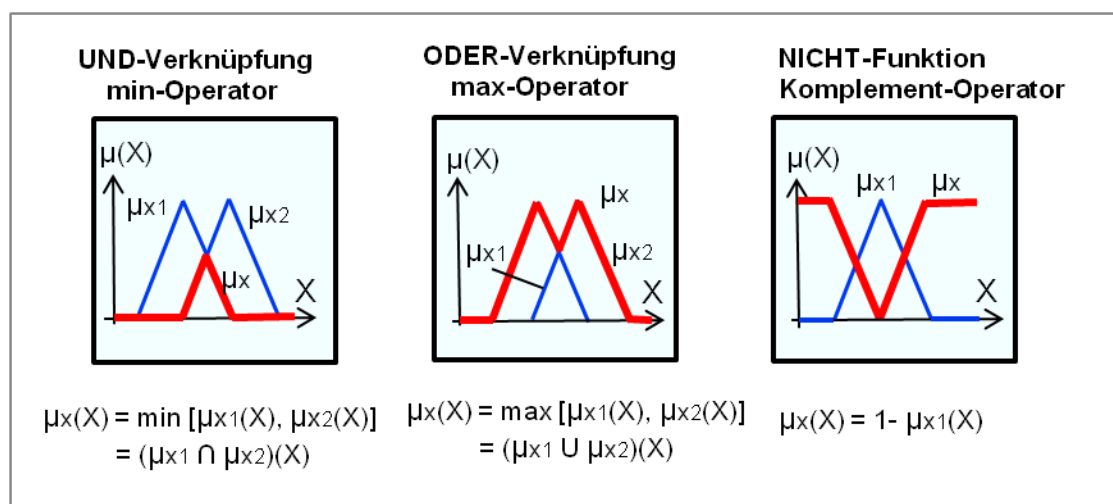


Abbildung 13: Logische Verknüpfungen bei Fuzzy Mengen<sup>19</sup>

Die allgemein bekannte traditionelle boolesche Logik unterscheidet nur genau zwei Wahrheitswerte (Bivalenzprinzip) und ordnet jede Aussage eindeutig zu: entweder eine Aussage ist ganz wahr oder aber die Aussage ist ganz falsch. Diese Art der Logik wird auch als scharfe (engl. „crisp“) Logik bezeichnet. Die Operatoren und Eigenschaften der booleschen Algebra können hierbei verwendet werden. Computer arbeiten entsprechend der booleschen Logik basierend auf den diskreten Stufen 0 oder 1.

<sup>18</sup> Zugehörigkeitsfunktion ist definiert als  $\mu_A: X \rightarrow [0,1]$

<sup>19</sup> <https://commons.wikimedia.org/wiki/File:Fuzzy-Operatoren.png>, 20.5.2016

Fuzzy Logik hingegen ist ein in den 70er Jahren des letzten Jahrhunderts von Dr. Lotfi Zadeh geprägter Begriff und steht für eine Verallgemeinerung der booleschen Logik. Fuzzy Logik erweitert die boolesche Logik um die Möglichkeit mit unscharfen Wahrheitsbegriffen zu arbeiten und sich überlappende, graduelle Übergänge zwischen Zugehörigkeitsbereichen zu definieren. Fuzzy Logik ermöglicht so glatte Übergänge zwischen den durch die Regeln definierten Bereichen. Viele Problemstellungen aus der realen Welt können so wesentlich realistischer modelliert werden. Ein Funktionsgebirge mit glatten Übergängen durch Fuzzy Logik wird in Abbildung 14 beispielhaft für eine Fuzzy-basierte Bremskraftsteuerung dargestellt.

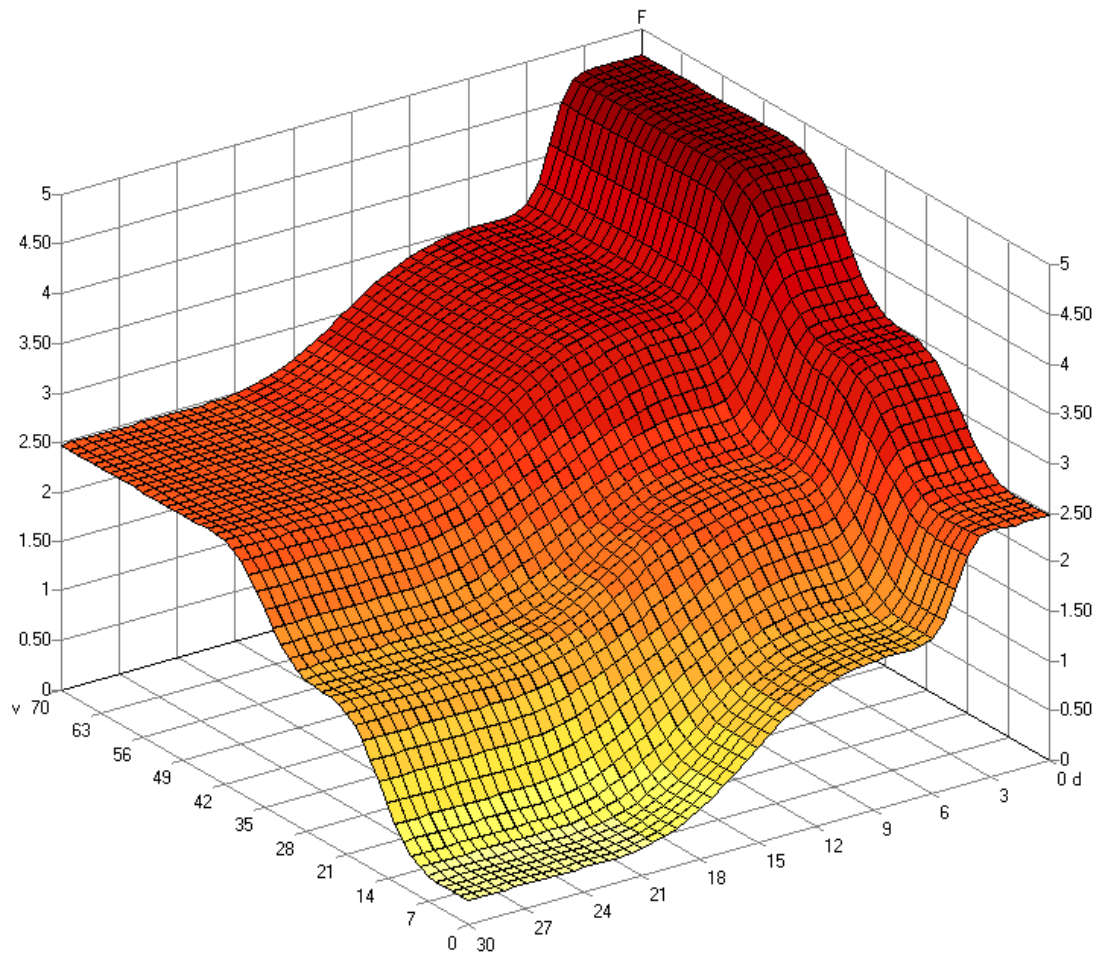


Abbildung 14: Fuzzy Logik ermöglicht glatte (graduelle) Übergänge zwischen den definierten Bereichen<sup>20</sup>

<sup>20</sup> Prof. Dr.-Ing. Andreas Meisel: *Robot Vision Skript*, Kapitel 5.4.1, S.28, Hochschule für Angewandte Wissenschaften Hamburg, April 2014

Die boolesche Logik stellt einen Spezialfall der Fuzzy Logik dar, in dem die Zugehörigkeitsfunktion nur auf die Endpunkte des Intervalls reduziert wird. Die Verteilung der Zugehörigkeiten und der Wahrheit beschränkt sich dann nur noch auf diese zwei, sich gegenseitig ausschließende, Singularitäten (0 oder 1). Somit schließt Fuzzy Logik die Eigenschaften und Möglichkeiten der booleschen Logik mit ein, ist aber darüber hinaus noch erheblich mächtiger in ihren Möglichkeiten Problematiken realistisch zu beschreiben.

Da ein Element in der Fuzzy Logik zu unterschiedlichen Graden zu mehreren Mengen gleichzeitig gehören kann, wird eine exakte Zugehörigkeit sozusagen „verwischt“. Auf diese Weise entsteht eine graduelle Verteilung der Wahrheit. Der Sinn dieser Erweiterung ist es, in der realen Welt weit verbreitete, unscharfe Phänomene mathematisch zu beschreiben und sie verarbeiten zu können. Obwohl Computersysteme fundamental mit binärer Logik funktionieren, ist es leicht möglich Fuzzy Logik auf digitalen Systemen zu implementieren.

Fuzzy Logik eignet sich z.B. oft zur Beschreibung von unscharfen Kategorisierungen aus der natürlichen Sprache. Ein Beispiel hierfür ist die Frage: „Wie viele Sandkörner braucht man für einen Haufen?“ Eine binäre Interpretation dieser Fragestellung mit harten Übergängen würde zu paradoxen Ergebnissen führen. Ein im Kontext dieser Arbeit interessanteres Beispiel ist die Computer gestützte Bestimmung eines angemessenen Trinkgeldes für das Kundenerlebnis in einem Restaurant. Dieses Problem orientiert sich an unscharfen Begriffen der natürlichen Sprache wie ob das Essen „lecker“ war und wie „gut“ der Service war. Ein scharfer Wahrheitsbegriff kann hierfür nicht definiert werden, da diese Einschätzungen auf einem Gefühlsspektrum mit graduellen Abstufungen und sich überlappenden Übergängen einzuordnen sind. Außerdem können diese Einschätzungen je nach aktueller Befindlichkeit variieren und sie sind grundsätzlich subjektiv und somit von Person zu Person unterschiedlich. Ziel ist es also eine ausgewogene und allgemein als richtig empfundene Entscheidung zu treffen. Eine gewisse Unschärfe bei der Lösungsfindung ist also nicht nur akzeptabel, sondern grundsätzlich unvermeidbar und der Realität der Problemstellung angemessen.

Fuzzy Logik eignet sich besonders gut für diese Art der Fragestellung. Mit Fuzzy Logik können Unschärfen der Fragestellung relativ effizient beschrieben werden. Zwar kann ein Fuzzy Logik System eine Lösung nicht selbständig erlernen, sondern hängt von der geschickten Umsetzung durch den Entwickler ab, allerdings kann eine sprachnahe Beschreibung des Problems relativ einfach realisiert werden. In vielerlei Anwendungen hat sich Fuzzy Logik bewährt, um generell zufriedenstellende Ergebnisse für schwierige Problemstellungen zu liefern und dabei die Komplexität niedrig zu halten. Die logischen Operationen erfordern zudem relativ wenig Rechenkapazität und der Entwickler weiß zu jeder Zeit, wie das System seine Entscheidungen trifft. Anpassungen und Optimierungen des Systems sind so mit geringerem Aufwand und weniger Fehlerpotenzial möglich als bei komplexeren Lösungen.

Das Ziel dieser Arbeit ist es ein intelligentes System zu entwickeln, dass die unscharfe und teilweise subjektive Fragestellung beantwortet, ob ein „feedbackwürdiger Beratungskontakt“ stattgefunden hat oder nicht. Diese Fragestellung wohnt von Natur aus eine gewisse Unschärfe inne, besonders in den Grenzbereichen zwischen den eindeutigen

Fällen. Zudem ist im Kontext des traditionellen Handels erhebliches menschliches Vorwissen der Branchenexperten verfügbar, das bei der Lösung dieser Problemstellung genutzt werden sollte. Dieses Vorwissen lässt sich gut in sprachnahe Regeln formulieren und kann somit auch von den nicht-technischen Kunden leicht verstanden werden.

Eine Prototyp-Implementierung mit Fuzzy Logik im Vergleich zu alternativen Techniken der künstlichen Intelligenz ist relativ schnell zu implementieren. Da das System auf Mobilgeräten von Endkunden laufen soll, ist eine effiziente Ressourcennutzung wichtig. Mit Fuzzy Logik können akzeptable Ergebnisse mit relativ geringer Komplexität erreicht werden. Die Eingabedaten die dafür vom System verfügbar sind scheinen ebenfalls grundsätzlich geeignet für die Analyse mittels Fuzzy Logik. Tatsächliche Sensordaten aus der Umwelt beschränken sich auf nur zwei Dimensionen: die Distanzen zu den Beacons und die Zeitspannen während möglicher Kontakte. Diese äußerst überschaubare Zahl der Eingabevariablen für die intelligente Beratungskontakterkennung unterstützt die Hypothese, dass Fuzzy Logik ein sinnvoller Ansatz ist.

### **3.1.4 Fazit**

Durch die Abwägung der Eigenschaften der drei in Betracht gezogenen Verfahren für die intelligente Beratungskontakterkennung können die Vor- und Nachteile im Bezug auf das hier zu realisierende Feedback System zusammengefasst werden. Die resultierende Entscheidungsmatrix (siehe unten) ermöglicht eine systematische Entscheidung für das beste Verfahren. Vor diesem Hintergrund wird im Folgenden das Feedback System entworfen und der Proof of Concept durchgeführt.

Verfahren	Eignung für die Problemstellung	Verfügbare Daten	Komplexität und Ressourcenaufwand
Analytisch	<ul style="list-style-type: none"> <li>- Problemstellung nicht mathematisch exakt</li> <li>- Finden einer sinnvollen Funktion oder Gleichung wäre sehr schwierig</li> </ul>	<ul style="list-style-type: none"> <li>- Keine direkte Quelle für Eingabe / Ausgabe Paare für eine Lösung verfügbar</li> </ul>	<ul style="list-style-type: none"> <li>- Potenziell gut aber nicht relevant wegen mangelnder Eignung</li> </ul>
Lernen	<ul style="list-style-type: none"> <li>- Funktionaler Zusammenhang ist nicht unbekannt</li> <li>- Eingangsvektor nicht lang</li> </ul>	<ul style="list-style-type: none"> <li>- Beschaffung von Beispieldaten wäre aufwendig und unpraktikabel</li> </ul>	<ul style="list-style-type: none"> <li>- Hohe Komplexität und schwer verständliche modellierung</li> <li>- Relativ hoher Ressourcenverbrauch</li> </ul>
Regelbasiert (Fuzzy Logik)	<ul style="list-style-type: none"> <li>✓ Sprachnahe Beschreibung möglich und sinnvoll</li> <li>✓ Kurzer Eingangsvektor</li> <li>✓ Unscharfe Problemstellung</li> </ul>	<ul style="list-style-type: none"> <li>✓ Erhebliches Vorwissen</li> <li>✓ Gutes Verständniss der funktionalen Zusammenhänge</li> </ul>	<ul style="list-style-type: none"> <li>✓ Relativ einfach realisierbar</li> <li>✓ Sehr anschaulich und verständlich</li> <li>✓ Gute Wartbarkeit</li> <li>✓ Geringer Ressourcenverbrauch</li> </ul>

Es wird deutlich, dass eine regelbasierte Lösung mit Fuzzy Logik ein guter Ansatzpunkt ist. Um dieses Verfahren effektiv einsetzen zu können, wird zunächst eine ausführlichere Einführung in die Anwendung von Fuzzy Logik ausgearbeitet.

### 3.2 Einführung in die Anwendung von Fuzzy Logik

Wie zuvor beschrieben, ist Fuzzy Logik eine Erweiterung der boolschen Logik. Elemente können zu mehreren Mengen gleichzeitig graduell zugehörig sein. Diese Zugehörigkeiten können durch entsprechende Zugehörigkeitsfunktionen beschrieben werden. Entscheidungen in der Fuzzy Logik hängen von den unscharfen Zugehörigkeiten eines Elements zu mehreren sogenannten linguistischen Variablen bzw. Fuzzy Mengen ab. Diese Zugehörigkeiten können als eine Art Wahrheitsverteilungen um einen bestimmten Wert verstanden werden. Eine Fuzzy Menge ist demnach eine Abbildung aus der Menge aller möglichen scharfen Eingabewerte des Systems auf ein unscharfes Zugehörigkeitsintervall (typischerweise normalisiert auf  $[0, 1]$ ) das den Grad der Wahrheit der Zugehörigkeit



widerspiegelt. Somit bilden die Zugehörigkeitsfunktionen die scharfen Eingabewerte auf die entsprechenden vom Entwickler definierten linguistischen Variablen der Fuzzy Mengen ab. Dieser Prozess wird „Fuzzification“ genannt.

Nach der Fuzzification können aufgrund der definierten Fuzzy Logik Regeln Ergebnisse berechnet werden. Wie zuvor erwähnt, existiert leider kein systematischer Prozess, um geeignete Fuzzy Logik Regeln für eine Problemstellung zu definieren. Die Regeln müssen basierend auf menschlichem Expertenwissen gefunden werden und ihre Effektivität kann anschließend an der Qualität der Entscheidungen, die das System trifft, überprüft werden.

Die Ergebnisse der Auswertung mit den Fuzzy Regeln sind zunächst ebenfalls graduelle Zugehörigkeiten zu verschiedenen Fuzzy Mengen. Solche unscharfen Ergebnisse können nicht direkt zur Steuerung eines digitalen Systems verwendet werden. Es ist daher notwendig einen Zwischenschritt zu vollziehen, um aus den unscharfen Fuzzy-Ergebnissen wieder einen scharfen Ausgabewert zu erzeugen. Dieser Prozess wird „Defuzzification“ genannt. Hierfür können verschiedene Standardverfahren verwendet werden, um ein scharfes Ergebnis aus den graduellen Zugehörigkeiten zu produzieren. Die häufigsten drei Standardverfahren für Defuzzification sind:

- **Maximierer:** wählt den jeweils höchsten Fuzzy Ergebniswert für die Ausgabe
- **Gewichteter Durchschnitt:** bildet den gewichteten Durchschnitt der Fuzzy Ergebnisse
- **Massenmittelpunkt:** bildet das Masse gewichtete Mittel der Fuzzy Ergebnisse

Mithilfe einer der Defuzzification Techniken kann ein eindeutiges Ergebnis gefunden werden mit dem das digitale System arbeiten kann.

### 3.2.1 Veranschaulichung an einem Beispiel

Im Folgenden soll ein einfaches Beispiel dazu dienen, diesen Ablauf zu veranschaulichen. Für eine Temperaturregelung soll eine Entscheidung basierend auf einer Einschätzung der aktuellen Außentemperatur getroffen werden. Dazu sollen scharfe Temperatur-Eingabewerte auf die Fuzzy Mengen der Begriffe aus der natürlichen Sprache: „kalt“, „warm“ und „heiß“ abgebildet werden. Solche Einschätzungsbegriffe werden in der Fuzzy Logik auch als „linguistische Variablen“ bezeichnet.

Zunächst müssen menschliche Vorgaben basierend auf dem Entwickler zur Verfügung stehendem Vorwissen gemacht werden. Die Bedeutung dieser Begriffe schwankt stark zwischen Eskimos und Tropenbewohnern. Darum wird zunächst die Zielgruppe genauer spezifiziert: Das System soll Entscheidungen für die Region um Hamburg treffen. Dementsprechend wird Vorwissen über die Temperatur-Mittelwerte nach Jahreszeiten bei der Regelfindung hinzugezogen. Da die meisten Bewohner Hamburgs niedrige Temperaturen im Winter als „kalt“ bezeichnen würden und hohe Temperaturen im Sommer als „heiß“, wird das mittlere Temperaturminimum im Winter von minus 3° Celsius als eindeutig „kalt“ definiert. Entsprechen wird das mittlere Temperaturmaximum im

Sommer von plus 23° Celsius als eindeutig „heiß“ definiert. Der Einfachheit halber wird der Mittelwert dieser beiden Extrempunkte mit 13° Celsius als „warm“ definiert. Somit gehören Temperaturen um diese drei definierten Extrempunkte eindeutig zu einer der Fuzzy Mengen und nicht zu den anderen. In den Übergangsbereichen jedoch ist diese Trennung nicht mehr eindeutig. Temperaturwerte gehören dann graduell zu zwei verschiedenen Fuzzy Mengen und es ist nicht eindeutig klar wie die Außentemperatur zu bewerten ist. Die resultierende Verteilung der Zugehörigkeiten kann graphisch dargestellt werden. Abbildung 15 stellt die Zugehörigkeitsfunktionen der drei Fuzzy Mengen auf dem normalisierten Intervall [0, 1] dar.

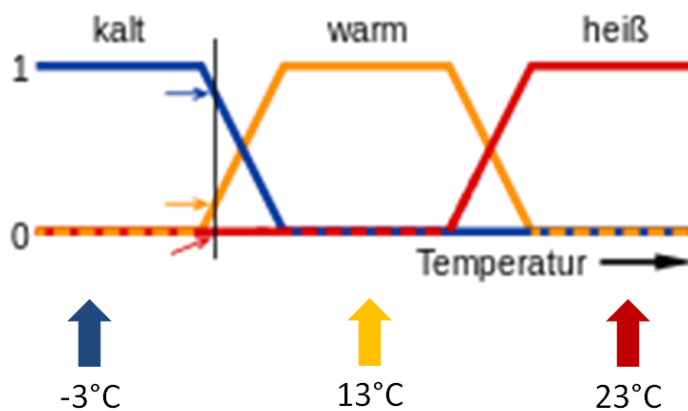


Abbildung 15: Fuzzylogik einer Temperaturregelung<sup>21</sup>

Zusätzlich zu den definierten Extrempunkten muss der Entwickler auch noch Punkte auf der Temperaturachse definieren, welche die Punkte markieren, wo die Zugehörigkeit nicht mehr eindeutig ist ( $0 < \text{Zugehörigkeit} < 1$ ). Außerdem müssen Punkte definiert werden wo die Temperatur eindeutig nicht mehr zu einer Fuzzy Menge gehört (Zugehörigkeit = 0). Der einfachste Ansatz dafür ist es diese Punkte gleichmäßig und mittig zwischen den Extrempunkten zu wählen und die Punkte direkt mit Geraden zu verbinden. So entstehen lineare Verteilungen der Zugehörigkeiten.

Trotz der in Abbildung 15 veranschaulichten Simplizität, liefert diese Lösung erstaunlich sinnvolle Ergebnisse für eine unscharfe Fragestellung, die sich in einem digitalen System nicht unmittelbar erschließt. Um weiterhin bei der einfachsten Lösung zu bleiben, wird zur Defuzzifizierung die Technik des Maximierers gewählt. Somit würde die linguistische Variable mit dem jeweils höchsten Zugehörigkeitsgrad ausgewählt werden und dies könnte direkt als eindeutige (scharfe) Klassifizierung der Außentemperatur dienen.

Natürlich ließe sich diese einfache Lösung beliebig erweitern, um komplexere Verteilungen zu modellieren. So könnten z.B. beliebig viele Zwischenpunkte für den Verlauf der Zugehörigkeitsfunktionen definiert werden oder nicht-lineare Funktionen wie Gauß oder Sigmoid eingesetzt werden. Grundsätzlich können durch die Kombinationen von

<sup>21</sup> Fuzzylogik einer Temperaturregelung: <https://de.wikipedia.org/wiki/Fuzzylogik>, 02.05.2016

mathematischen Ausdrücken beliebige Zugehörigkeitsfunktionen definiert werden. Ebenso können grundsätzlich beliebig viele linguistischer Variablen mit Einfluss auf das Ergebnis definiert werden. Abbildung 16 zeigt ein Beispiel einer solchen nicht-linearen Zugehörigkeitsfunktion zur Kontrastverbesserung für Sportfotos, die aus einer Kombination von Sigmoid (dark und bright) und Gauß-Funktion (mid-gray) zusammengesetzt ist.

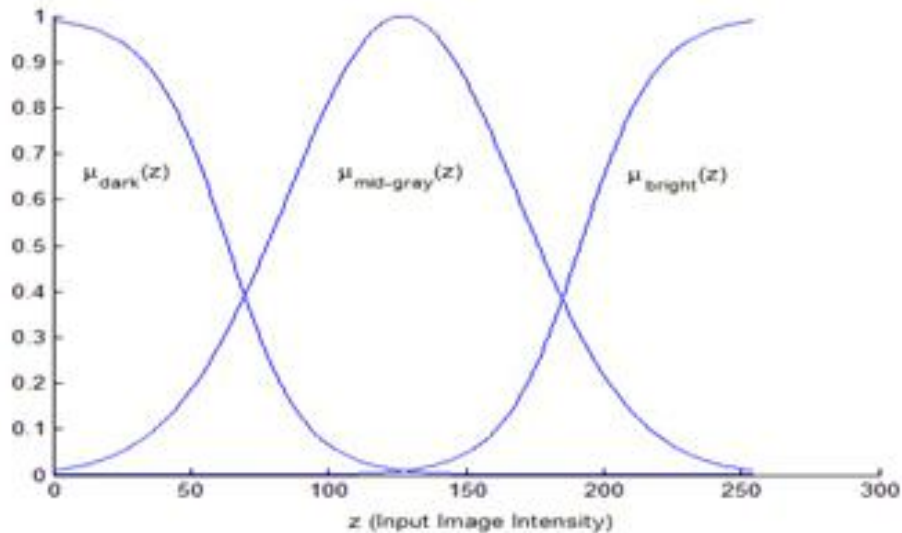


Abbildung 16: Beispiel einer zusammengesetzten Zugehörigkeitsfunktion<sup>22</sup>

Bei der Wahl einer geeigneten Zugehörigkeitsfunktion gilt es natürlich sinnvoll abzuwägen, ob zusätzlicher Entwicklungs- und Berechnungsaufwand durch den Zuwachs an Präzision gerechtfertigt ist.

### 3.2.2 Fuzzy Logik Steuerung

Bei der Anwendung von Fuzzy Logik zur entscheidungsbasierten Steuerung eines digitalen Systems hat sich die in Abbildung 17 dargestellte Struktur einer Fuzzy Logik Steuerung (FLS) bewährt<sup>23</sup>. Mit Steuerung ist in diesem Zusammenhang ein System mit offenem Wirkungsweg gemeint. Das bedeutet, dass die Ausgangsgröße also nicht auf die Eingangsgröße zurückwirkt. Vielmehr handelt es sich hier um ein Subsystem, welches basierend auf scharfen Eingabewerten des umgebenden Systems eine Entscheidung mittels Fuzzy Logik trifft und diese als scharfe Ausgabewerte an das umgebende System zurückgibt.

<sup>22</sup> P. Kannan, S. Deepa, R. Ramakrishnan (2012): *Contrast Enhancement of Sports Images Using Two Comparative Approaches*, American Journal of Intelligent Systems

<sup>23</sup> Pablo Cingolani, Jesus Alcalá-Fdez (2012): *jFuzzyLogic: a Java Library to Design Fuzzy Logic Controllers According to the Standard for Fuzzy Control Programming*. International Journal of Computational Intelligence Systems, Vol. 6.

Die durch die Fuzzy Logik getroffene Entscheidung (ausgedrückt durch scharfe Ausgangsgrößen) steuert das weitere Vorgehen des umgebenden Systems und hat somit einen entscheidenden Einfluss auf dessen Abläufe.

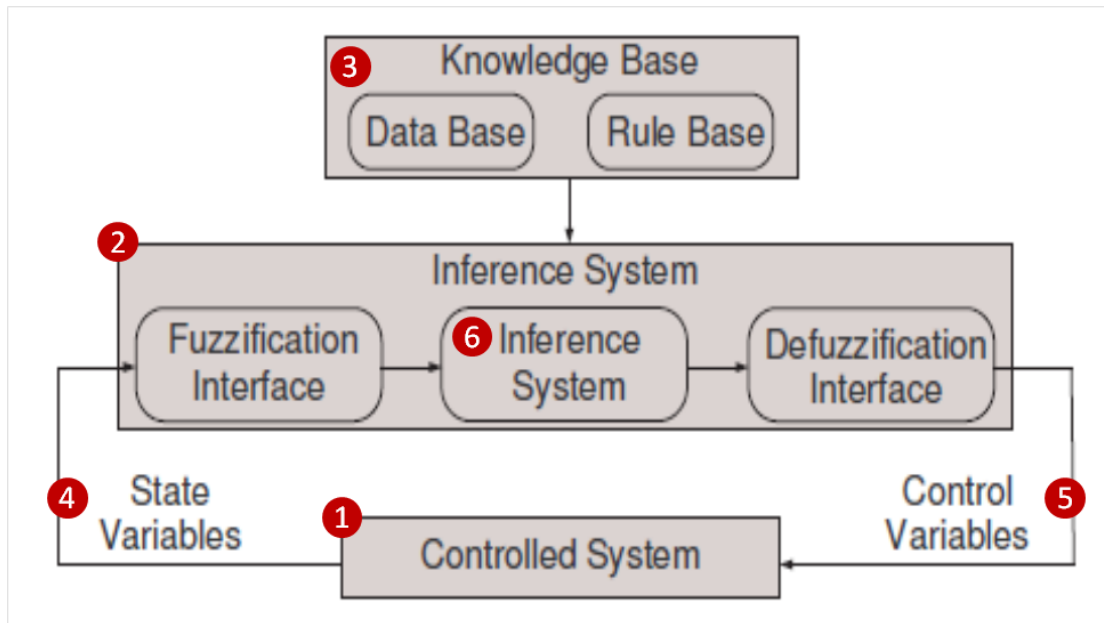


Abbildung 17: Überblick der typischen Struktur einer Fuzzy Logik Steuerung (FLS)

Die FLS gliedert sich generell in drei Teile: das zu steuernde System (roter Punkt P1), das Inferenzsystem (P2) zur Schlussfolgerung mittels Fuzzy Logik und die dafür eingesetzte Wissensbasis (P3). Das FLS definiert dabei typischerweise eine nichtlineare Abbildung vom Zustandsraum (P4) des zu steuernden Systems auf den Kontrollraum (P5). Das zu steuernde System in dieser Arbeit ist das Feedbacksystem auf dem Mobilgerät des Einkäufers. Die vom Mobilgerät gemessenen diskreten Beaconsignalwerte bilden die Eingaben in das Inferenzsystem der FLS. Die scharfen Eingaben müssen im Inferenzsystem zunächst „fuzzifiziert“ werden, um sie in die für die regelbasierte Inferenz notwendigen Fuzzymengen einzuordnen. Die eigentliche Inferenz findet dann im inneren Inferenzsystem (P6) statt. Hier bedient sich das Inferenzsystem (P2) an den, in der Wissensbasis (P3) beinhalteten, Regeln, um zu einer Entscheidung zu kommen. Anschließend müssen die Ergebnisse der Inferenz wieder defuzzifiziert werden, damit sie als scharfe Ausgaben wieder an das zu steuernde System übergeben und verarbeitet werden können. In der Wissensbasis liegt das menschliche Expertenwissen in der Form von sogenannten linguistischen Steuerungsregeln vor. Basierend auf den fuzzifizierten Eingabewerten ergeben diese Fuzzy Regeln die Entscheidungen der Inferenz, die dann defuzzifiziert und zur Steuerung verwendet werden können. In dieser Arbeit ist das zu steuernde System (P1) das Feedbacksystem, welches mit Hilfe der durch die Fuzzy Logik ermittelten Entscheidungen situationsangemessen reagieren kann.

### 3.3 Beratungskontakterkennung mit Fuzzy Logik

In diesem Abschnitt wird die konkrete Anwendung einer Fuzzy Logik Steuerung auf die in dieser Arbeit zentrale Problematik der intelligenten Beratungskontakterkennung mittels Beacons ausgearbeitet. Dabei gilt es zunächst, die dem System zur Verfügung stehenden Messdaten zu analysieren, um deren Relevanz als Zustandsvariablen für die Eingabe in eine Fuzzy Logik Implementation einzuordnen. Diese Zustandsvariablen bilden die fundamentale Entscheidungsgrundlage für die Fuzzy Logik Intelligenz. Anschließend muss möglichst aussagekräftiges Expertenwissen zusammengetragen werden, um aus den verfügbaren Daten möglichst effektive linguistische Variablen für das Inferenzsystem abzuleiten. Darauf aufbauend muss dann die Wissensbasis entwickelt und explizite Fuzzy Logik Regeln formuliert werden, um die verfügbaren Messdaten bestmöglich zur Beratungskontakterkennung zu nutzen. Im Ergebnis müssen dann spezifische Entscheidungen in Form von scharfen Steuerungsvariablen für das Feedbacksystem verfügbar gemacht werden.

#### 3.3.1 Ausführliche Analyse der verfügbaren Messdaten

Eine Innovation des hier zu entwickelnden Systems ist, dass keine winkelbasierte Triangulation von drei oder vier stationären Beaconsignalen benötigt wird, um relevantes Kundenverhalten auf der Verkaufsfläche zu erfassen. Die hier angestrebte intelligente Vereinfachung der Abläufe hat einige klare Vorteile wie z.B. geringerer Installations- und Wartungsaufwand, sowie ein begrenzter Eingriff in die Privatsphären der Anwender. Allerdings bedeutet dies auch eine Herausforderung an das System, da es mit erheblich weniger reichhaltigen Messdaten auskommen muss.

Die einzigen verfügbaren Messdaten für die intelligente Beratungskontakterkennung sind die Signalstärken der mobilen Mitarbeiter-Beacons sowie des stationären iBeacons im Kassenbereich. Die resultierenden Messdaten bilden somit Zeitreihen der Signalstärke der einzelnen Beacons. Das Gimbal Standard Development Kit (SDK) des Herstellers Qualcomm erfasst Messwerte der Signalstärken in Form von Received Signal Strength Indicator (RSSI) Werten. Das Datenblatt des Herstellers definiert die Einheit der RSSI Werte als Dezibel-Milliwatt (dBm)<sup>24</sup>. Dabei wird eine Granularität bei der Signalmessung in diskreten 10 dBm Schritten erreicht. Das Gimbal SDK stellt diese RSSI Werte in ganzzahligen 10er Schritten auf einem Intervall von -20 bis -100 zur programmatischen Weiterverarbeitung zur Verfügung. Für Signalstärken unterhalb von -100 wird der Beacon als außer Reichweite ausgewiesen. Es existiert eine grundsätzliche Beziehung zwischen der gemessenen RSSI Signalstärke und der Distanz zur Signalquelle. Deshalb kann die vom Mobilgerät gemessene Signalstärke zur

---

<sup>24</sup> Gimbal™ Series 10 Proximity Beacon: User Manual, <https://docs.gimbal.com/manuals/s10.html>, 29.05.2016

Einschätzung der Distanz zum Beacon genutzt werden. Das Gimbal SDK stellt bereits einen Schätzwert für die Distanz zur Verfügung. Vereinfacht ausgedrückt, nimmt die Signalstärke immer weiter ab, je weiter sich ein Beacon vom Mobilgerät entfernt. Diese Beziehung ist jedoch nicht linear sondern logarithmisch. Das bedeutet, dass der messbare Unterschied von 10 dBm mit immer kleiner werdenden Distanzunterschieden gleichzusetzen ist. Je weiter Sender und Empfänger voneinander entfernt sind, desto kleiner werden die Distanzunterschiede für eine Signalabschwächung von 10 dBm. Abbildung 18 veranschaulicht die logarithmische Skala der verfügbaren RSSI Signalstärken und ihre generelle Interpretation bei der Distanzeinschätzung.

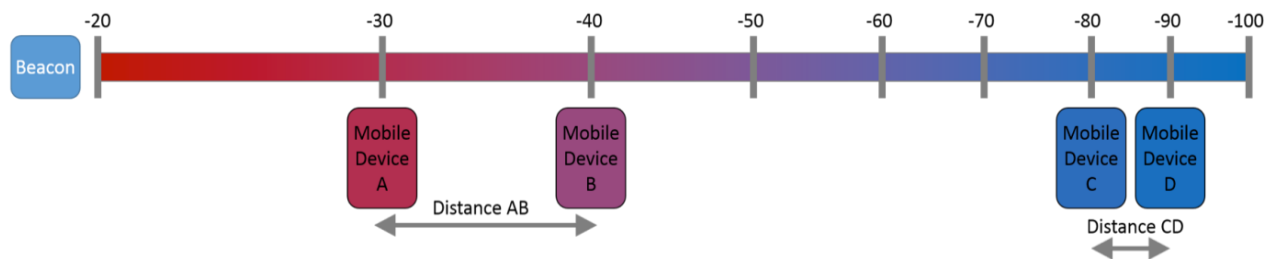


Abbildung 18: Distanzeinschätzung mit der logarithmischen RSSI Skala des Gimbal SDK<sup>25</sup>

Allerdings gilt es zu beachten, dass die gemessenen Signalstärken durch Umweltbedingungen beeinflusst werden können. So kann es z.B. durch verschiedene physikalische Hindernisse zwischen Beacon und Mobilgerät oder auch störende Funksignale anderer Geräte im Umfeld, trotz gleichbleibender Distanz, zu geringfügigen Schwankungen in den gemessenen Signalstärken kommen. Da es sich nicht um ein Echtzeitsystem handelt und weil eine relativ grobkörnige Skala ausreicht um Beratungskontakte zwischen Menschen zu erkennen, liegt es nahe mit Durchschnittswerten für die relevanten Zeitspannen zu arbeiten. Eine solche Zusammenfassung der Messdaten, auf einer für die Erkennung relevanten Ebene, würde durch äußere Störungen bedingte Schwankungen in der Signalstärke weitgehend ausgleichen. Dieser Ansatz macht auch Sinn, weil die Fuzzy Logik nicht kontinuierlich auf neu eintreffende Messdaten angewandt werden sollte, sondern eine abschließende Entscheidung über die gesamte Zeitspanne der potenziellen Beratung ausreichend ist. Es ist nicht anzunehmen, dass durch eine kontinuierliche Verarbeitung der Messdaten durch Fuzzy Logik eine relevante Verbesserung der Ergebnisse zu erreichen wäre.

Letztlich steht also nur eine Einschätzung des Distanz-Verhaltens über die Zeit zwischen Einkäufer und Servicemitarbeiter zur Verfügung, um eine verlässliche Beratungskontakterkennung zu leisten. Diese beschränkten Informationen über die Umwelt bergen die Gefahr, ein ungenaues System zu entwickeln, das regelmäßige Fehleinschätzungen liefert, weil es zufällige Begegnungen von tatsächlichen Beratungskontakten nicht unterscheiden kann. Deshalb ist die Ausarbeitung von

<sup>25</sup> Understanding RSSI values: [https://docs.gimbal.com/proximity\\_overview.html#visits](https://docs.gimbal.com/proximity_overview.html#visits), 14.05.2016

aussagekräftigen Kriterien zur Demarkation von Beratungskontakten mithilfe von Expertenwissen ausschlaggebend für die Effektivität des Systems.

Des Weiteren ist zu prüfen, ob weitere Umweltinformationen hinzugezogen werden können, um präzisere Ergebnisse zu ermöglichen. Die naheliegendste solcher Quellen für zusätzliche Informationen sind die Signale der anderen Beacons, die sowieso vom System empfangen werden. Es ist zu erwarten, dass die Interpretation der Signale weiterer mobiler Servicemitarbeiter-Beacons äußerst schwierig und wenig aussagekräftig wäre. Die Position dieser Beacons kann sich laufend ändern. Zudem sind die Signalstärken dieser Beacons wahrscheinlich überwiegend unabhängig von einem möglichen Beratungskontakt durch einen anderen Servicemitarbeiter. Der stationäre iBeacon im Kassenbereich allerdings könnte als konstanter Referenzpunkt dienen, um die Signale der Servicemitarbeiter-Beacons aussagekräftiger zu machen. Indem das System die Signale eines fraglichen Servicemitarbeiter-Beacons zum Signal des stationären iBeacons in Beziehung setzt, könnte in vielen Situationen beispielsweise gemeinsames nebeneinanderhergehen durch den Laden erkennbar werden. In diesem Fall würde sich die Distanz zum stationären iBeacon ändern, während die Distanz zum Servicemitarbeiter-Beacon relativ konstant bliebe. Es ist zu beachten, dass dieser Referenzpunkt nur eine sehr grobe Einschätzung ermöglicht, ob sich der Kunde durch den Laden bewegt, da Distanzveränderungen nur relativ zur Position des iBeacons messbar sind und daher vom Winkel der Bewegung abhängen. Im schlimmsten Fall bietet das iBeacon Signal keinerlei relevante Informationen, nämlich wenn sich der Kunden in konzentrischen Kreisen um den iBeacon bewegt. Allerdings haben die allermeisten Ladenflächen viereckige Layouts die konzentrische Kreisbewegungen unwahrscheinlicher machen. Da durch die Einbeziehung des iBeacon Signals keine relevanten Informationen verloren gehen oder verfälscht werden können, sollte es trotz seiner variablen Aussagekraft in die Beratungserkennung miteinfließen.

Im Falle eines unbefriedigenden Ergebnisses der Beratungserkennung könnten weitere Sensoren des Mobilgeräts eingesetzt werden. Der Einsatz des Gyroskops etwa könnte Aufschlüsse darauf zulassen, ob sich der Kunde im Gehen befindet oder steht. Allerdings würde der Einsatz weiterer Sensoren zwangsläufig die Komplexität und den Ressourcenverbrauch der App erhöhen. Zudem müssten weitere Permissions für das Mobilgerät der Endkunden dafür eingeholt werden. Darum wird zunächst der Fokus darauf gelegt mit den bestehenden Beacondaten eine zufriedenstellende Lösung zu finden.

### 3.4 Expertenwissen Zusammenstellung

Im Rahmen dieser Arbeit stehen zwei verschiedene Quellen für die Zusammenstellung einer Experten-Wissensbasis zur Verfügung:

1. Gespräche mit Branchenexperten bei den zwei Pilotkunden
2. Meine eigenen persönlichen Erfahrungswerte
  - a. Als Fachverkäufer bei Möbel Höffner
  - b. Als Kunde im Einzel- und Großhandel

Im Vorfeld des Gründungsprojektes war das zeitliche Engagement der Pilotkunden sehr begrenzt. Trotzdem konnten wertvolle Erkenntnisse für den Prototypen gewonnen werden. Eigene Erfahrungswerte wurden als Ausgangspunkt und zur Strukturierung der Mitarbeitergespräche bei den Pilotkunden genutzt. Letztlich werden jedoch in dieser Arbeit die Erfahrungen und Aussagen der Branchenexperten als ausschlaggebende Quelle für die Wissensbasis des Systems betrachtet. Eigene Erfahrungswerte, die nicht durch Branchenexperten bekräftigt wurden, wurden vernachlässigt. Aufgrund der begrenzten Zeit und der zurzeit noch informellen Natur der Beziehung zu den Pilotkunden war die umsetzbare Strenge und Systematik bei der Zusammenstellung des Expertenwissens im Rahmen dieser akademischen Arbeit leider begrenzt. Es wurde versucht mit den verfügbaren Mitarbeitern in der begrenzten Interviewzeit möglichst präzise und brauchbare Aussagen zu erarbeiten. Trotz dieser unoptimalen Umstände sind die so gesammelten Meinungen der Branchenexperten die relevanteste verfügbare Quelle für die Wissensbasis der zu realisierenden Fuzzy Logik Lösung.

Natürlich weist das menschliche Expertenwissen teilweise eine ähnliche Unschärfe auf wie die Fragestellung selbst. Die Identifikation eines feedbackwürdigen Beratungskontaktes ist auch für die Experten selbst ein unscharfes Unterfangen. Zudem basiert das gesammelte Wissen auf stichprobenartigen Befragungen und subjektiven Einschätzungen von einer sehr kleinen Gruppe von Teilnehmern. Trotzdem ist das zusammengetragene Wissen als brauchbare Grundlage für den Proof of Concept mit Fuzzy Logik zu bewerten. Für die Zusammenstellung der Wissensbasis flossen die Erfahrungen und Meinungen von vier Experten ein: zwei Mitarbeiter von jedem Pilotkunden. Die befragten Mitarbeiter können sich dabei auf eine gemeinsame Branchenerfahrung von mehr als 50 Jahren berufen.

Da es sich um nicht-technische Branchenexperten aus den Bereichen Geschäftsführung, Service und Verkauf im Groß- und Einzelhandel handelt, sind die Meinungen zumeist in sehr umgangssprachlicher und informeller Art und Weise kommuniziert worden. Meinungen wurden nach einem Trial and Error Verfahren durch eine Diskussion von beispielhaften Situationen herausgearbeitet. Ausgangspunkt der Diskussionen waren dabei meine persönlichen Annahmen, Ideen und Einschätzungen. So wurde die generelle Relevanz von Kriterien überprüft, Kriterien verworfen und neue wichtige Kriterien entdeckt. Es wurde nochmals sehr deutlich, dass die Ausarbeitung einer allgemeinen mathematischen Zugehörigkeitsfunktion im Sinne eines analytischen Verfahrens unter den gegebenen Umständen absolut nicht praktikabel gewesen wäre. Stattdessen wurden veranschaulichende Beispielsituationen mit allen Experten durchgesprochen und Kriterien, die letztlich von allen drei Experten als relevant eingeschätzt wurden, fließen in die Wissensbasis für die Fuzzy Logik ein.

In den Gesprächen wurde deutlich, dass es extreme Szenarien gibt, bei denen die Identifikation von Beratungskontakten nahezu eindeutig möglich ist. Die Schwierigkeit besteht darin, in den Übergangsbereichen zwischen diesen Extremen, eine korrekte Einschätzung zu finden. Darum wurde versucht möglichst präzise Einschätzungen für die Kriterien in den nicht eindeutigen Szenarien herauszuarbeiten. Mithilfe der Fuzzy Logik soll dann auch in den Übergangsbereichen mit diesen Kriterien zwischen zufälligem Verhalten



und tatsächlichen Beratungskontakten unterschieden werden. Experten wurden angehalten eine ungefähre quantitative Einstufung über die jeweilige Stärke der Zugehörigkeit der relevanten Kriterien in natürlicher Sprache zu treffen (z.B. „schwach“, „mittel“, „stark“). Diese grobe probabilistische Einschätzung soll bei der Definition der Fuzzy Mengen helfen.

Trotz der erheblichen Unterschiede zwischen den Geschäftsmodellen der beiden Pilotkunden, treffen die gefundenen qualitativen Kriterien bei beiden grundsätzlich zu. Allerdings gibt es deutliche Unterschiede bei der Quantifizierung dieser Kriterien. Alle Experten betonten mehrfach, dass es sich dabei letztendlich nur um subjektive Schätzwerte handelt.

#### Allgemeine qualitative Kriterien:

1. Je geringer die durchschnittliche Distanz und je länger die Verweildauer zwischen Kunde und Servicemitarbeiter, desto wahrscheinlicher ist ein Beratungskontakt
2. Es existiert eine Mindestdistanz, außerhalb derer ein Beratungskontakt so gut wie auszuschließen ist
3. Es existiert eine Mindestdauer, unterhalb derer ein Beratungskontakt so gut wie auszuschließen ist
4. Es existiert eine Maximaldauer, nach deren Ablauf stark davon ausgegangen werden kann, dass ein Beratungskontakt stattgefunden hat
5. Es existiert eine typische Distanz für Beratungskontakte
6. Es existiert eine typische Dauer für Beratungskontakte
7. Die meisten Beratungskontakte finden in der typischen Distanz oder näher statt
8. Gemeinsames Gehen durch den Laden ist ein Indikator für einen Beratungskontakt
9. Während eines tatsächlichen Beratungskontakts nimmt die durchschnittliche Distanz typischerweise mit der Zeit ab
10. Mehrfaches sehr nahes herantreten an den Kundenberater ist ein starker Indikator für einen Beratungskontakt
11. Auch während eines tatsächlichen Beratungskontakts kann die Mindestdistanz kurzfristig verlassen werden
12. Verlassen der Mindestdistanz für länger als eine Maximaldauer kann generell mit der Beendigung eines Beratungskontaktes gleichgesetzt werden

Experten beider Pilotkunden waren sich einig, dass diese 12 Kriterien bei der Erkennung von Beratungskontakten entscheidend sind. Um einen Maßstab für gemeinsames Gehen zu bekommen, wurden die Experten gebeten, auch hier eine Mindestdistanz anzugeben, ab der davon ausgegangen werden kann, dass es sich wahrscheinlich nicht um zufälliges Verhalten handelt, sondern um einen Indikator für einen Beratungskontakt. Letztendlich waren die Experten beider Pilotkunden mit einer Distanz von grob 4 Metern zufrieden. Ab

dieser gemeinsam zurückgelegten Distanz ist die Wahrscheinlich, dass ein Beratungskontakt stattgefunden hat, signifikant höher.

METRO spezifische quantitative Kriterien:

- Gemeinsames durch den Laden gehen ist ein starker Indikator
- Mindestdistanz: 4 Meter
- Sehr nahe Distanz: ungefähr 0,5 Meter
- Mindestdauer: 30 Sekunden
- Maximaldauer: 5 Minuten
- Typische Distanz: 1,5 Meter
- Typische Dauer: 3 Minuten
- Maximaldauer für Verlassen der Mindestdistanz: 30 Sekunden

Porsche spezifische quantitative Kriterien:

- Gemeinsames durch den Laden Gehen ist ein mittel-starker Indikator
- Große gemeinsam zurückgelegte Distanzen (>100 Meter) deuten auf Testfahrten hin und sind mit einem Beratungskontakt gleich zu setzen (typischerweise fällt der iBeacon außer Reichweite)
- Mindestdistanz: 5 Meter
- Sehr nahe Distanz: < 1 Meter
- Mindestdauer: 3 Minuten
- Maximaldauer: 15 Minuten
- Typische Distanz: 2 Meter
- Typische Dauer: 30 Minuten
- Maximaldauer für Verlassen der Mindestdistanz: 5 Minuten

Nun müssen aus diesen kundenvvalidierten Kriterien entsprechende Regeln für die Fuzzy Logik Steuerung ausgearbeitet werden. Mit Beginn des Gründungsprojektes können dann weitere Optimierungen zusammen mit den Pilotkunden in einem iterativen Prozess vorgenommen werden. Zunächst fällt auf, dass einige der Expertenaussagen so eindeutig zu sein scheinen, dass eine Nutzung mit konventioneller Aussagenlogik sinnvoll ist. Aussagenlogik zeichnet sich durch geringere Komplexität aus und ist daher bei vergleichbarer Effektivität der Fuzzy Logik vorzuziehen. Deshalb scheint es sinnvoll, Fuzzy Logik hier nur für die wirklich unscharfen Kriterien einzusetzen und für eindeutige Kriterien auf traditionelle Aussagenlogik zurückzugreifen. Die Kriterien 1 bis 8 rechtfertigen den Einsatz von Fuzzy Logik, um die ihnen innenwohnenden Unschärfe für die Übergangsbereiche adäquat modellieren zu können. Die Kriterien 9 bis 11 hingegen können

mit den verfügbaren Messdaten unter Umständen sogar einfacher als binäre Wahrheitswerte modelliert werden. Somit könnten Kriterien 9 bis 11 hier durch einfache Aussagenlogik in die Entscheidungsfindung einfließen. Eine solche Kombination der beiden Logiken zur Entscheidungsfindung ist relativ unkompliziert. Anpassungen, wie eventuelle zukünftige Erweiterungen oder Umwandlung der Kriterien von einer Logik in die andere, sind ebenfalls relativ einfach möglich. Um eventuelle Anpassungen weiter zu vereinfachen, werden die binären Regeln auch über Fuzzy Logik Syntax aber eben als Fuzzy Singletons realisiert.

Die intelligente Beratungskontakterkennung wird gestartet, wenn das System feststellt, dass die jeweilige Mindestdistanz zwischen Kunde und Servicemitarbeiter für die Mindestdauer unterschritten wurde. Während der potenziellen Beratung werden die relevanten Messdaten aufgezeichnet, um sie anschließend für die Fuzzy Logik Auswertung bereitstellen zu können. Das Kriterium 12 wird im Folgenden lediglich dazu dienen um zu wissen, wann ein möglicher Beratungskontakt als beendet anzusehen ist. In diesem Fall können die bis dahin gesammelten Eingabedaten für die Fuzzy Logik bereitgestellt werden, damit diese eine Entscheidung über den potenziellen Beratungskontakt treffen kann.

### **3.4.1 Definition der Zugehörigkeitsfunktionen und linguistischen Variablen**

Da es sich bei den quantitativen Werten der Wissensbasis um Schätzwerte handelt, müssen sie zunächst mithilfe von linguistischen Variablen fuzzifiziert und als Zugehörigkeitsfunktionen beschrieben werden. Es wurde mit den Experten besprochen, dass umgebende Werte, die den Schätzwerten naheliegen, mit relativ hoher Wahrscheinlichkeit zu den gleichen Fuzzy Mengen gehören sollten, wobei die Zugehörigkeit mit zunehmender Distanz vom Schätzwert graduell abnimmt. Hier zeigt sich noch einmal, dass die Struktur von Fuzzy Logik äußerst geeignet erscheint, um diesen Sachverhalt zu modellieren.

Wie bereits bei der Auswahl der künstlichen Intelligenz erwogen, basieren die verfügbaren scharfen Eingabewerte des Systems für die Fuzzy Logik Steuerung dabei grundsätzlich auf drei unterschiedlichen reellen Eingabewerten die der Fuzzification bedürfen:

1. Geschätzte durchschnittliche Distanz zwischen Einkäufer und Servicemitarbeiter
2. Zeitspannen während des potenziellen Beratungskontaktes
3. Geschätzte Distanzveränderung zum stationären iBeacon im Kassen- bzw. Ausgangsbereich während des potenziellen Beratungskontakts

Es gilt nun, gemäß dem Expertenwissen eine sinnvolle Einteilung in linguistische Variablen zu entwerfen und die Zugehörigkeitsfunktionen zu definieren. Dieser Prozess wird im Folgenden beispielhaft für die spezifischen quantitativen Kriterien für METRO erarbeitet. Eine quantitative Adaption auf andere Kunden ist leicht schematisch zu bewerkstelligen.

Zunächst ist eine Entscheidung zu treffen, welcher Wertebereich für den 1. Eingabewert (durchschnittliche Distanz) für das Inferenzsystem relevant ist. Dank der Expertenaussagen liegen hierfür klare Werte für eine Mindestdistanz vor. Aufgrund der Ungenauigkeit und Subjektivität der Schätzwerte der Experten, erscheint es sinnvoll, rund um die exakten Werte eine gewisse Toleranz zu definieren. Darum wurde ein pauschaler Toleranzbereich von 10% um die jeweilige Distanz einzuführen und dem System sozusagen etwas „Leine zu geben“. Das bedeutet, dass innerhalb dieser Toleranzbereiche die Zugehörigkeit zur jeweiligen linguistischen Variablen voll gegeben ist.

Der Wertebereich für die Eingabevariable Distanz ist somit von null Metern bis zur Mindestdistanz (für METRO 4 Meter) + 10% definiert. Eine linguistische Unterteilung in die drei umgangssprachlichen Bereiche „sehr nah“ (engl. „very near“), „nah“ (engl. „near“) und „weit“ (engl. „far“) ist intuitiv verständlich und scheint ein sinnvoller Ausgangspunkt. Abbildung 19 stellt die resultierende Zugehörigkeitsfunktion mit den linguistischen Variablen für den Eingabewert Distanz für METRO dar. Die schwarze Linie („value“) markiert die, von den Experten bei Beratungskontakten als typisch beschriebene, Distanz (ebenfalls mit einem Toleranzbereich von 10% des Distanzwerts).

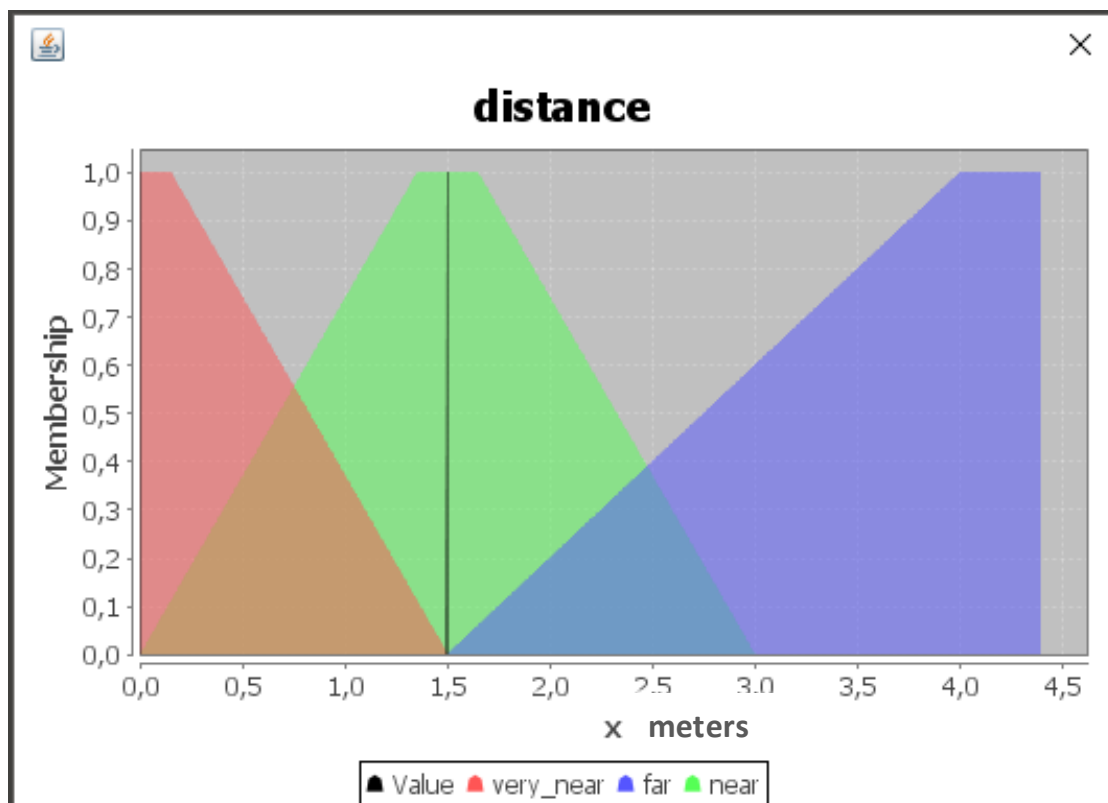


Abbildung 19: Zugehörigkeitsfunktion der Kunde-Mitarbeiter Distanz mit drei linguistischen Variablen (METRO)

Mit dieser Zugehörigkeitsfunktion kann nun der scharfe Eingabewert, der mit den Beacons geschätzten Distanz zwischen Kunde und Servicemitarbeiter, durch die Fuzzification unscharf gemacht werden und anschließend in den Fuzzy Logik Regeln verwendet werden. Hier ist deutlich zu sehen, wie die Bereiche der drei linguistischen Variablen ineinander überlappen und so eine abwägende Einschätzung der Zugehörigkeiten zwischen den eindeutigen Werten ermöglicht. Die Graphik illustriert wie Fuzzy Logik die Unschärfe in natürlicher Sprache modellieren kann. Der graduelle Übergang zwischen „weit“ und „nah“, in dem beide Begriffe als teilweise Wahr empfunden werden, sind durch die Fuzzification deutlich abgebildet.

Es wurde hier zunächst eine Beschreibung der Zugehörigkeitsfunktion in der Form von linearen Trapezen gewählt. Dieser Ansatz hat den Vorteil der Einfachheit und das Expertenwissen bietet keine konkreten Anhaltspunkte, um eine offensichtlich bessere Beschreibung durch komplexere Funktionen, wie z.B. durch nichtlineare Sigmoid oder Gauss Funktionen, zu realisieren. Trotzdem kann ein potenzieller Effekt auf die Güte der der Beratungskontakterkennung mit beiden Methoden untersucht werden, um die beste Beschreibung zu finden. Zu diesem Zweck wird auf Grundlage der gleichen Expertenaussagen eine nichtlineare Zugehörigkeitsfunktion konstruiert. Abbildung 20 zeigt diese nichtlineare Zugehörigkeitsfunktion für die Distanz.

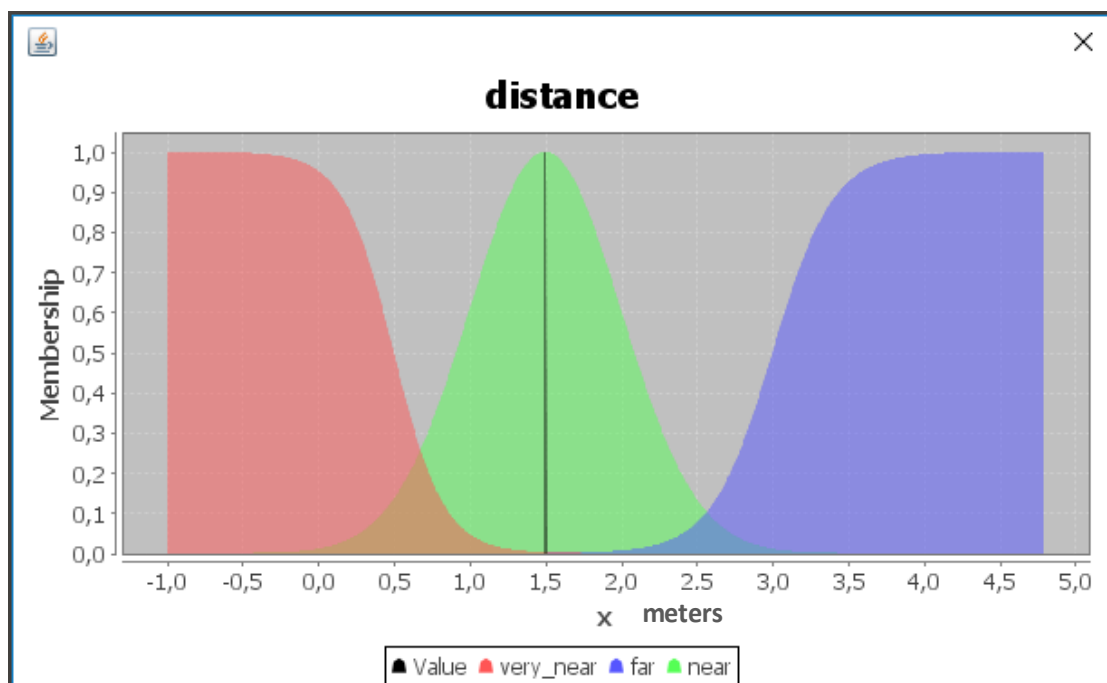


Abbildung 20: Nichtlineare Zugehörigkeitsfunktion der Distanz

Es wird deutlich, dass die Demarkation der linguistischen Variablen äußerst ähnlich verläuft. Einzig der für Sigmoid typische steile Abfall und Anstieg modelliert einen plötzlicheren Wechsel von einem Bereich zum nächsten. Diese Darstellung repräsentiert geringere

Zugehörigkeiten zu beiden betroffenen Fuzzy Mengen in den Übergangsbereichen. Signifikant unterschiedliche Ergebnisse sind somit nicht zu erwarten.

Als nächstes muss die zweite scharfe Eingabevariable der Dauer fuzzifiziert werden. Mithilfe des Expertenwissens kann der Wertebereich definiert werden. Demnach ist die Minstdauer für METRO etwa 30 Sekunden und die Maximaldauer etwa 5 Minuten bzw. 300 Sekunden. Außerdem gibt es eine typische Beratungsdauer von etwa 3 Minuten bzw. 180 Sekunden. Vor diesem Hintergrund scheint eine Einteilung in drei linguistische Variablen „kurz“ (engl. „short“), „mittel lang“ (engl. „medium short“) und „lang“ (engl. „long“) auch hier sinnvoll. Eine gewisse Toleranz von 10% der jeweiligen Dauer soll auch hier gewährt werden um der Subjektivität des Expertenwissens Rechnung zu tragen. Eine lineare Beschreibung mit Trapezen kann diese Sachverhalte gut modellieren. Die Abbildung 21 zeigt die resultierende Zugehörigkeitsfunktion.

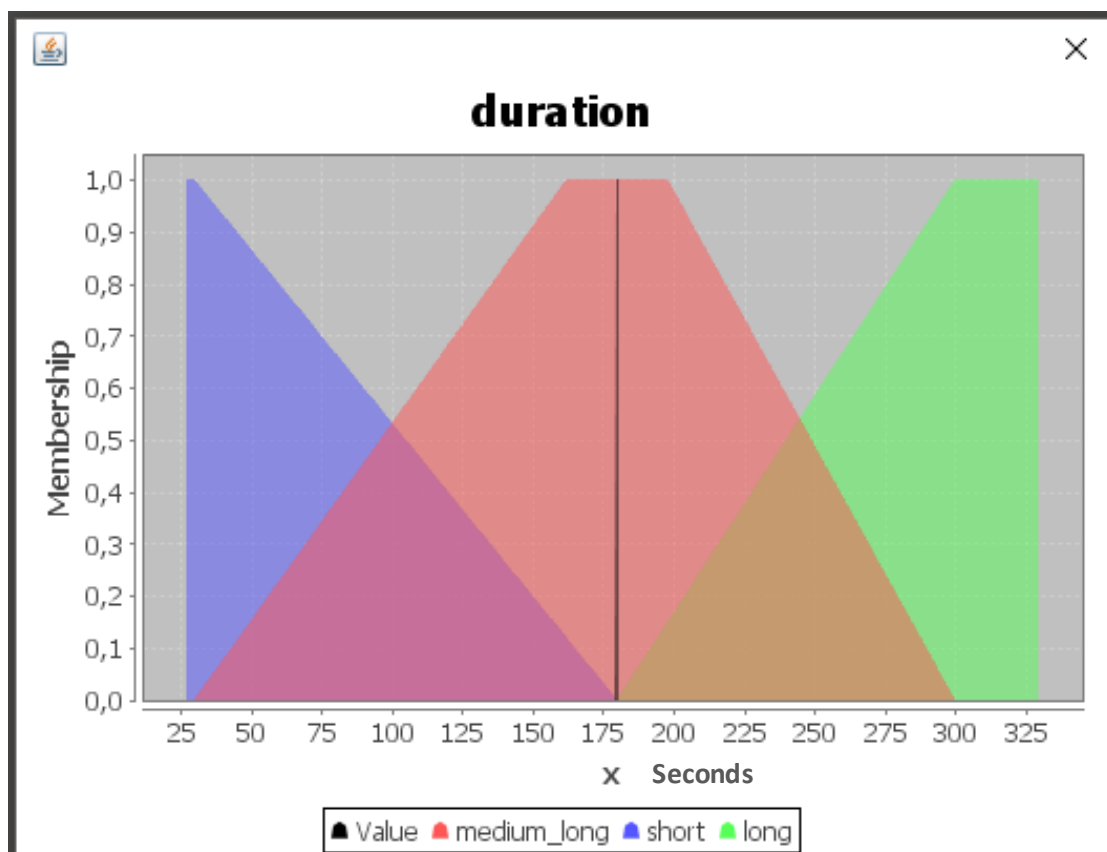


Abbildung 21: Zugehörigkeitsfunktion der Beratungs-Zeitspanne mit drei linguistischen Variablen (METRO)

Die Graphik scheint die aus dem Expertenwissen gewonnenen Erkenntnisse gut zu reflektieren. Trotzdem soll versuchsweise auch eine Modellierung mit Gauss-Sigmoid modelliert werden um den Einfluss auf die Beratungskontakterkennung untersuchen.

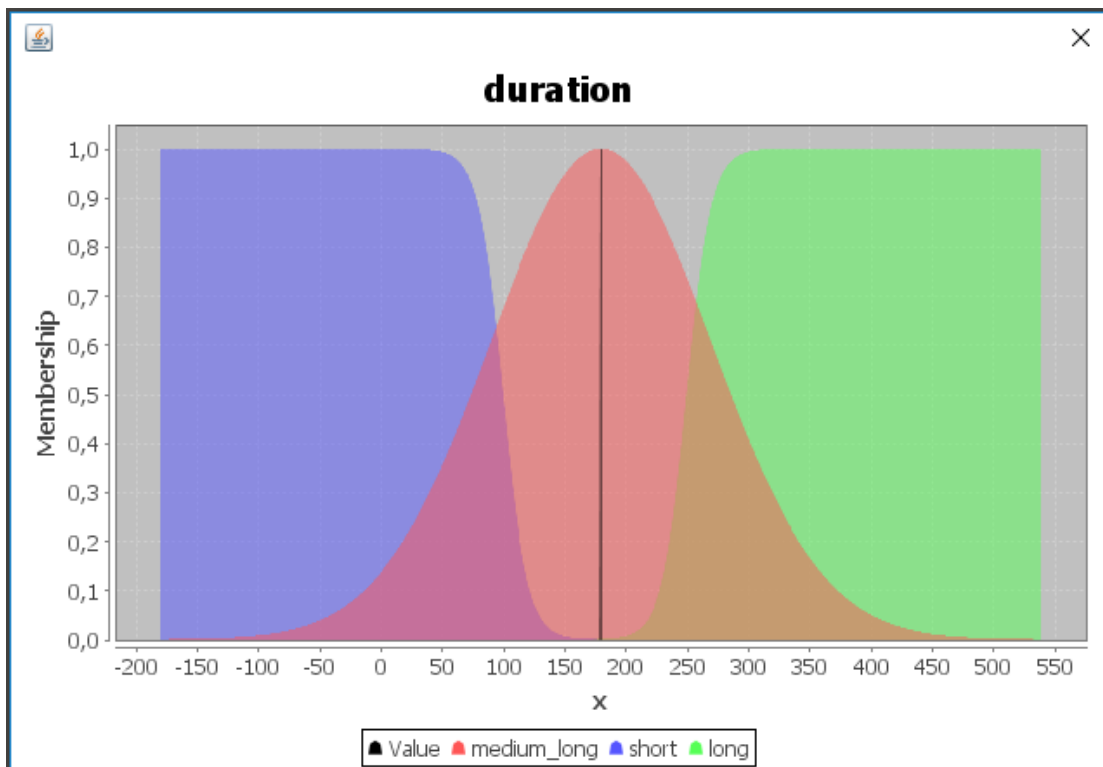


Abbildung 22: Nichtlineare Zugehörigkeitsfunktion der Beratungs-Zeitspanne

Der letzte scharfe Eingabewert der fuzzifiziert werden muss ist die gemeinsam zurückgelegte Distanz. Dieser Eingabewert ist problematisch weil die, mithilfe des stationären iBeacons geschätzte, Distanz stark abhängig ist von dem Winkel der Bewegung relativ zur Position des iBeacons. Außerdem muss diese Distanzschätzung zusätzlich noch zur Distanz zwischen Kunde und Servicemitarbeiter in Beziehung gesetzt werden. Deshalb ist der Eingabewert für diese letzte Fuzzification die mittels iBeacon geschätzte Distanzveränderung während Kunde und Servicemitarbeiter sich innerhalb ihrer Mindestdistanz zueinander befinden.

Die Zugehörigkeitsfunktion für diesen Sachverhalt weist deutlich andere Charakteristika auf als die Anderen. Da die Aussagekraft der Distanzschätzung zum iBeacon variiert, sollte dieses Kriterium nur wirklich ins Gewicht fallen, wenn seine Bedeutung relativ gut sichergestellt ist. Das ist der Fall, wenn deutliche Distanzveränderungen gemessen werden die in der Nähe des Mindestwertes der Experten liegen. Bei diesen Distanzen von um die 4 Meter oder mehr ist sichergestellt, dass mindestens diese Distanz zurückgelegt wurde und potenziell sogar deutlich mehr (abhängig vom Bewegungswinkel zum iBeacon). Bei wesentlich kleineren Werten ist nicht mehr klar wie die gemessene Distanz zu bewerten ist. In diesen Fällen sollte sehr konservativ vorgegangen werden und Werte deutlich kleiner als 4 Meter können bei der Beratungskontaktidentifikation außer Acht gelassen werden. Deshalb muss bei kleinen Distanzen ein rapider Abfall der Zugehörigkeitsfunktion

modelliert werden. Eine Sigmoid Funktion spiegelt diese Eigenschaften gut wieder. Eine Fuzzification mit nur zwei linguistischen Variablen erscheint hierbei sinnvoll. Entweder die geschätzte gemeinsame Distanzveränderung (engl. „joint distance change“) ist signifikant (engl. „significant“) oder „nicht signifikant“ (engl. „not significant“). Abbildung 23 zeigt die resultierende Modellierung der Zugehörigkeitsfunktion für die durch den iBeacon geschätzte Distanzveränderung.

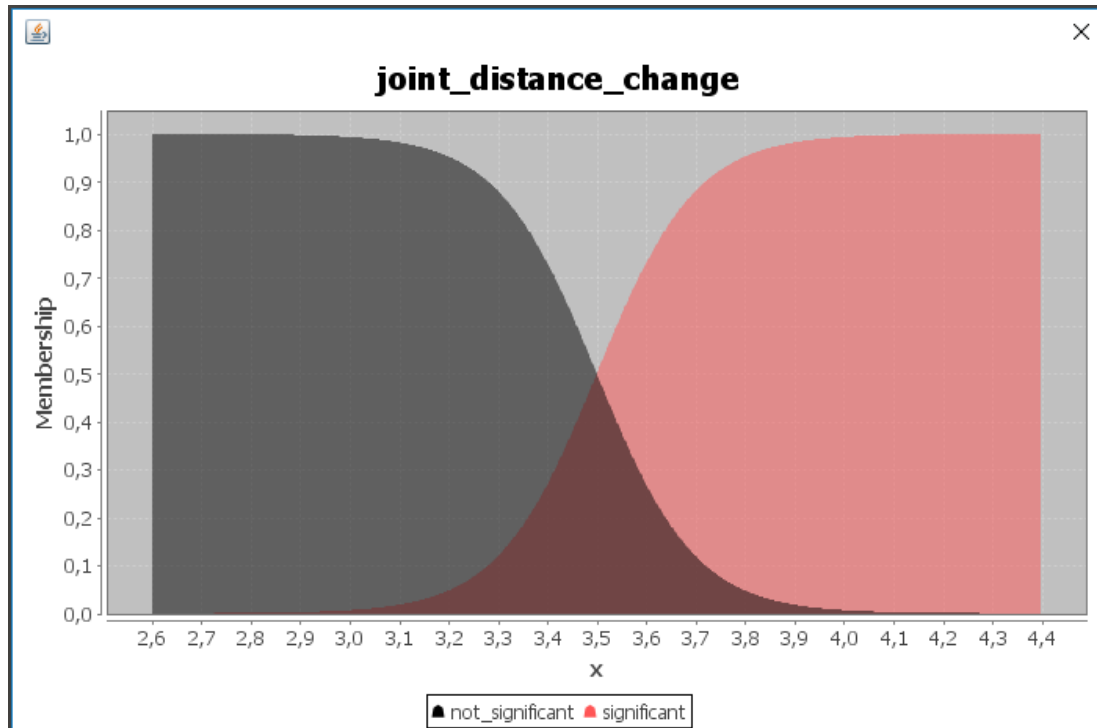


Abbildung 23: Zugehörigkeitsfunktion für gemeinsames durch den Laden Gehen während der Beratung

Es wird deutliche, dass sich bei zusammen zurückgelegte Distanzen von ab 3,5 Metern die Zugehörigkeiten rapide ändern und divergieren. Ab Werten von 3,8 Metern ist es als sehr sicher anzunehmen, dass gemeinsames Gehen stattgefunden hat. Bei Distanzen unter 3,3 Metern ist gemeinsames Gehen schon so gut wie auszuschließen.

Damit sind die unscharfen Übergänge der Eingabewerte für die Auswertung mit Fuzzy Logik vorbereitet. Als nächstes sind noch die scharfen Kriterien zu modellieren.

### 3.4.2 Definition der scharfen Variablen

Die Kriterien 9 und 10 der Experten-Wissensbasis eignen sich gut für eine scharfe Unterscheidung im Sinne der Aussagenlogik. Entsprechend kommen zwei weitere scharfe Eingabewerte für die Fuzzy Logik hinzu. Diese Eingabewerte sind allerdings keine neuen



Messwerte sondern können aus den bestehenden Eingabewerten des Eingabevektors für den Fuzzy Logik Funktionsblock gebildet werden:

1. Verringerung der durchschnittlichen Distanz zwischen Kunde und Servicemitarbeiter über die Zeitspanne des potenziellen Beratungskontakts
2. Mehrfaches Betreten des „sehr nahen“ Bereichs

Diese Variablen könnten gut als einfache IF / ELSE Statements implementiert werden. Im Rahmen dieser Arbeit werden sie als Singularitäten modelliert. Die Modellierung als Singularitäten erlaubt es auch, diese Variablen direkt in der Fuzzy Logik zusammen mit den fuzzifizierten Variablen zu verwenden. Somit sind alle relevanten Kriterien für die Beratungskontakterkennung im gleichen Format und werden an der gleichen Stelle zusammen geführt und ausgewertet. Eine eventuelle zukünftige Fuzzification dieser Werte ist sehr einfach möglich. Abbildung 24 zeigt die graphische Darstellung der Singularität für wiederholtes Betreten der „sehr nahen“ Distanz zwischen Kunde und Servicemitarbeiter.

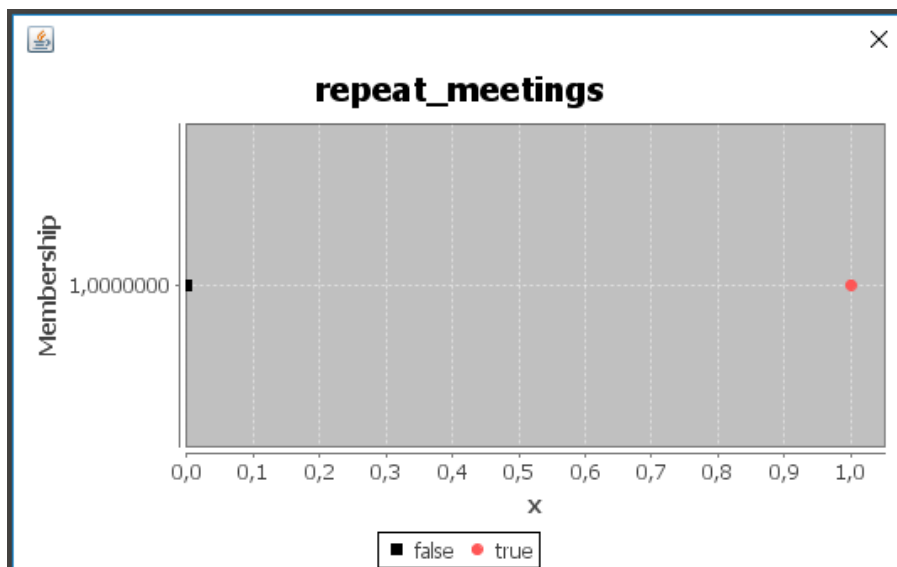


Abbildung 24: Singularität für wiederholtes Betreten der sehr nahen Distanz

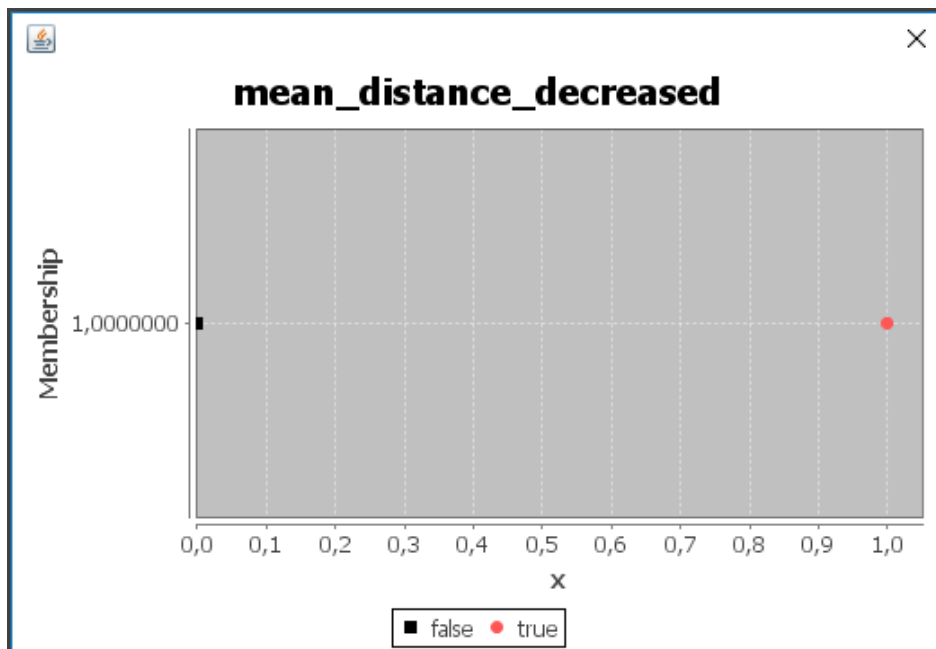


Abbildung 25: Verringerung der Durchschnittsdistanz als Singularität

Abbildung 25 illustriert die Verringerung der Durchschnittsdistanz als Singularität. Um zu illustrieren, dass es sich bei diesem Eingabewerten eigentlich um nur klassische Aussagenlogik handelt, wurden die Variablennamen für die Unterscheidung mit den traditionellen Labels „true“ und „false“ gewählt. Es wird deutlich, dass die Interpretation dieser Variablen mit den fuzzifizierten Variablen grundsätzlich konsistent ist. Allerdings ist die Wahrheitsverteilung der Zugehörigkeitsfunktion bei den Singularitäten auf zwei punktuelle Bereiche beschränkt. Entweder es besteht eine vollständige Zugehörigkeit zu einem Bereich („false“) oder eine vollständige Zugehörigkeit zu dem anderen Bereich („true“). Hier ist noch einmal graphisch deutlich zu erkennen wie Fuzzy Logik die traditionelle Aussagenlogik beinhaltet und deren Möglichkeiten erheblich erweitert.

### 3.4.3 Defuzzification der Ergebnisse

Wie in Abbildung 17 in der Einführung in die Anwendung von Fuzzy Logik besprochen, muss das Ergebnis der Auswertung durch Fuzzy Logik einen Defuzzification Prozess durchlaufen, um wieder in einen scharfen Wert zurückgewandelt zu werden, mit dem das umgebende System weiterarbeiten kann. Der Funktionsblock der Fuzzy Logik soll, wie in Abbildung 9 dargestellt, nur einen Ausgabewert liefern, aufgrund dessen eine Entscheidung getroffen werden kann, ob ein feedbackwürdiger Beratungskontakt stattgefunden hat oder nicht. Hierfür müssen nun zunächst sinnvolle linguistische Variablen für diesen Ausgabewert definiert werden. Die anschließend zu definierenden Fuzzy Logik Regeln bilden dann die Verbindung von den fuzzifizierten Eingabewerten auf diese linguistischen Variablen des

Ausgabewerts. Der Ausgabewert kann dann mit einem der Standardverfahren für Defuzzification zu einem scharfen Ausgabewert defuzzifiziert werden.

Da keine besonderen internen Anforderungen an die Ausgabe bestehen, wird der Wertebereich der Ausgabe, in Anlehnung an eine Prozentuale Aufteilung, in von 1 bis 100 definiert. Die einfachste Interpretation des Ergebnisses wäre eine simple Unterteilung in nur zwei linguistische Variablen. Die graduelle Zugehörigkeit eines Ergebnisses zu einer dieser beiden Variablen würde bereits eine Entscheidung darüber ermöglichen, ob ein Beratungskontakt stattgefunden hat oder nicht. Abbildung 26 zeigt diese simple Aufteilung in nur zwei linguistische Variablen mit linearen Trapezen und einem.

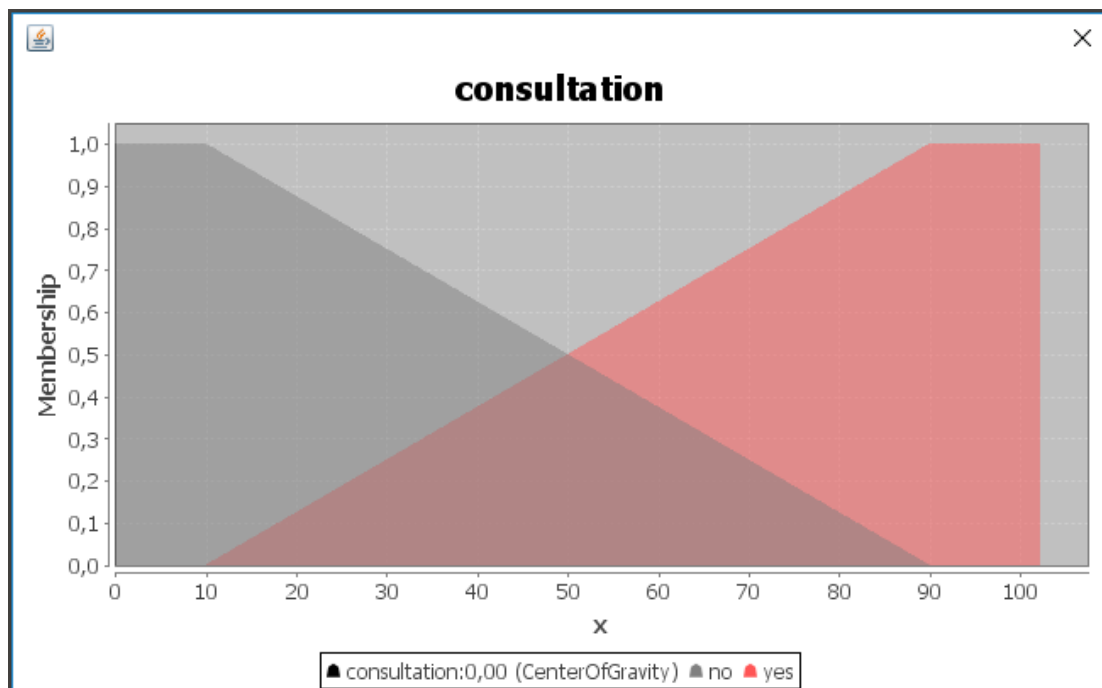


Abbildung 26: Simple Ja/Nein Aufteilung der Entscheidung über einen potenziellen Beratungskontakt

Allerdings bietet Fuzzy Logik gute Möglichkeiten eine höhere Granularität zu modellieren. Deshalb sollte eine möglichst intuitiv interpretierbare linguistische Aufteilung angestrebt werden. Es ist wünschenswert, dass der durch die Fuzzy Logik ermittelte Grad der Wahrheit des Ergebnisses sofort deutlich wird. Bei der Ausarbeitung der Fuzzy Logik Regeln bietet eine feinere Unterteilung zudem mehr Möglichkeiten, um nuancierte Regeln aufzustellen und spezifische Eingabewert-Situationen genauer zu unterscheiden. So ist es besser möglich im Ergebnis alle verfügbaren Informationen einzubeziehen, die bei einer simplen Ja/Nein Aufteilung verloren gehen könnten.

Eine linguistische Unterteilung in fünf Bereiche bietet ausreichende Abstufungen des Ergebnisses, um nuancierte Regeln aufzustellen und ist trotzdem für menschliche Leser noch gut zu überschauen. Die Namen der Variablen sollen den Grad der Wahrheit eines

Beratungskontaktes widerspiegeln. Eine allgemein verständliche Aufteilung von „sehr niedrige Wahrheit“ (engl. „very low truth“) bis „sehr hohe Wahrheit“ (engl. „very high truth“) scheint hierfür geeignet und bietet eine gute Grundlage um mit den Pilotkunden zu besprechen ab welchem Wahrheitsgrad eine Feedbackaufforderung ausgelöst werden soll. Bei der Ausarbeitung der Fuzzy Logik Regeln ist die Annahme sinnvoll, das die Zugehörigkeiten in den extremen Fällen der Randbereiche des Wertebereichs als relativ sicher anzunehmen sind. Diese Sicherheit nimmt jedoch mit zunehmender Distanz von den Randbereichen rapide ab. Eine Modellierung mit Sigmoid Funktionen scheint deshalb für die Randbereiche sinnvoll. Für die mittleren Variablen („niedrige Wahrheit“, „mittlere Wahrheit“ und „hohe Wahrheit“) kann die Annahme getroffen werden, dass der Grad der Zugehörigkeit normalverteilt ist. Abbildung 27 stellt diese nichtlineare Modellierung des Ausgabewertes graphisch dar.

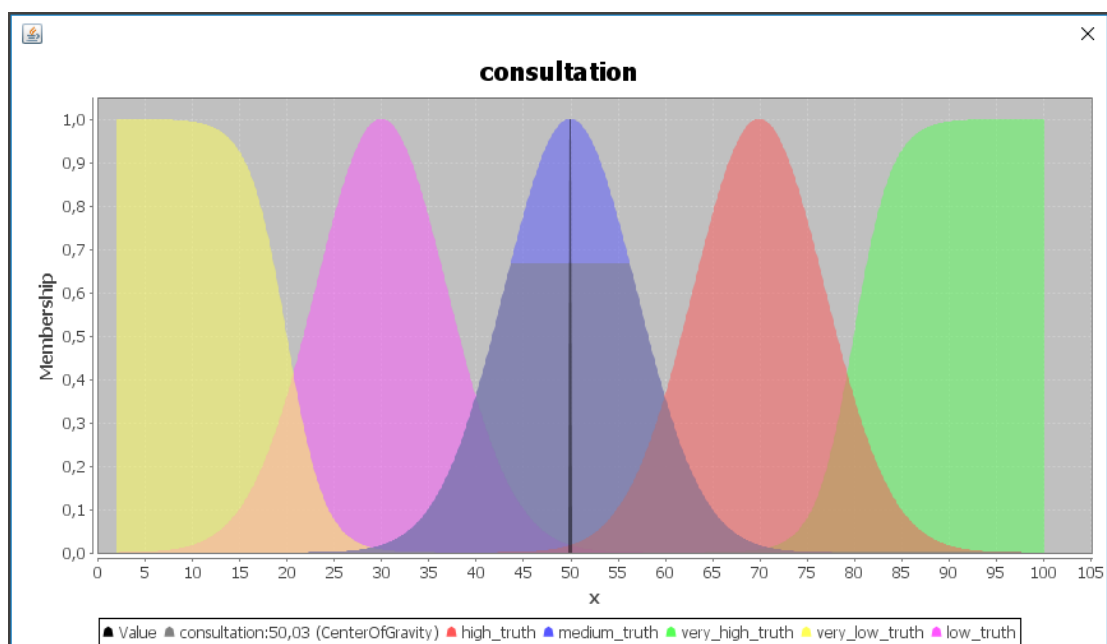


Abbildung 27: Fuzzy Logik Ausgabewert der Beratungskontakterkennung mit fünf linguistischen Variablen

### 3.4.4 Ausarbeitung der Fuzzy Logik Regeln

Die definierten linguistischen Variablen werden im Folgenden für die Ausarbeitung der Fuzzy Logik Regeln zur Beratungskontakterkennung angewendet. Hier ist zu bedenken, dass bei den gewählten fünf Eingabevariablen mit ihren Unterteilungen in linguistische Variablen bereits eine beachtliche Anzahl von Kombinationen möglich sind, die potenziell explizit als Regel beschrieben werden müssen. Eine mögliche Regelexplosion ist ein Vorbehalt bei der Verwendung von Fuzzy Logik. Dennoch ist im vorliegenden Fall ausreichendes Verständnis

der zu bewertenden Situation auf der Ladenfläche vorhanden um die Regelbasis deutliche einzugrenzen und mit effektiven Regeln die entscheidenden Fälle effizient abzudecken. Wenn, beispielsweise, durch die wichtigste Beziehung aus der Expertenwissensbasis zwischen Dauer und Distanz eine eindeutige Einschätzung der Situation angedeutet wird, sind die zusätzlichen Variablen weitgehend zu vernachlässigen. Die problematische Schätzung der zusammen zurückgelegten Distanz sollte zudem nur in sonst schwer zu entscheidenden Fällen ins Gewicht fallen. Die scharfen Variablen sind generell als unterstützend einzusetzen in Fällen in denen die Distanz-Dauer Beziehung keine klare Entscheidung zulässt. Dennoch ist besonders in diesem Aspekt der Regeldefinition noch Raum für weitere zukünftige Ergänzungen und Optimierungen<sup>26</sup>. Für den Proof of Concept werden hier die entscheidenden Fälle herausgearbeitet um eine erste sinnvolle Beratungskontakterkennung zu ermöglichen. In Zusammenarbeit mit den Pilotkunden kann darauf aufbauend eine Erweiterung und Präzisierung der Regeln mithilfe von Usability-Tests auf der Verkaufsfläche stattfinden.

Auf Basis der gewonnenen Einsichten aus den Pilotkundengesprächen sowie der Wissensbasis, sind die im Folgenden aufgelisteten 16 Fuzzy Logik Regeln herausgearbeitet worden. Um einen Proof of Concept zu erreichen, muss das System mit diesen 16 Regeln in der Lage sein, für eine Vielzahl erdenklicher Situationen, sinnvolle Entscheidungen über potenzielle Beratungskontakte zu treffen. Sinnvoll bedeutet hierbei eine Entscheidung über den Grad der Wahrheit eines Beratungskontaktes der mit der Wissensbasis konsistent ist. Die Regeln sind in der Syntax der Fuzzy Control Language (FCL)<sup>27</sup> verfasst. FCL ist allgemein gut lesbar und die hier aufgelisteten logischen Ausdrücke sind auch ohne tieferes Verständnis des FCL Standards verständlich. Im Anhang dieser Arbeit steht eine Einführung in den FCL Standard zur Verfügung.

Zunächst wird ein grundlegender Regelblock entworfen, der alle möglichen Kombinationen zwischen den zentralen Eingabevariablen Distanz und Dauer definiert. Auf dieser Basis können dann weitere Kriterien hinzugezogen werden um auch in den schwierigen Bereichen eine konkrete Regelbasis für die Entscheidungsfindung zu definieren. Der Basis-Regelblock umfasst die folgenden 9 Regeln:

1. RULE 1 : IF distance IS far AND duration IS short THEN consultation IS very\_low\_truth;
2. RULE 2 : IF distance IS far AND duration IS medium\_long THEN consultation IS low\_truth;
3. RULE 3 : IF distance IS far AND duration IS long THEN consultation IS medium\_truth;
4. RULE 4 : IF distance IS near AND duration IS short THEN consultation IS low\_truth;

---

<sup>26</sup> Feng Wan *et al.*: *How to determine the minimum number of fuzzy rules to achieve given accuracy: A computational geometric approach to SISO case*, Fuzzy Sets and Systems, März 2005

<sup>27</sup> IEC 61131-7, INTERNATIONAL ELECTROTECHNICAL COMMISSION (IEC) TECHNICAL COMMITTEE No. 65: INDUSTRIAL PROCESS MEASUREMENT AND CONTROL , IEC 1131 - PROGRAMMABLE CONTROLLERS Part 7 - Fuzzy Control Programming, 1997

5. RULE 5 : IF distance IS near AND duration IS medium\_long THEN consultation IS medium\_truth;
6. RULE 6 : IF distance IS near AND duration IS long THEN consultation IS high\_truth;
7. RULE 7 : IF distance IS very\_near AND duration IS short THEN consultation IS low\_truth;
8. RULE 8 : IF distance IS very\_near AND duration IS medium\_long THEN consultation IS high\_truth;
9. RULE 9 : IF distance IS very\_near AND duration IS long THEN consultation IS very\_high\_truth;

Durch diesen Basis-Regelblock wird zunächst die eindeutigste Beziehung der Eingabewerte wiederspiegelt: geringere Distanzen für längere Zeitspannen deuten auf eine höheren Wahrheitsgrad für einen Beratungskontakt hin. Die Regeln 1 steht für sehr eindeutige Fälle in denen ein entsprechend deutlich niedriger Wahrheitsgrad für eine Beratung realisiert wird. Die Gültigkeit dieser Regel ist weitgehend von den anderen drei Eingabewerten unabhängig und deckt somit Variationen in diesen anderen drei Variablen implizit mit ab. Ähnlich sicher sind die Regeln 6 bis 9. Die Regeln 2 bis 5 jedoch bilden für sich allein eine schwache Grundlage um in den schwierigeren Fällen mit denen sie sich befassen klar entscheiden zu können. Aus diesem Grund werden im Folgenden weitere 7 explizite Regeln formuliert die sich der anderen drei Eingabewerte bedienen um bessere Entscheidungen zu ermöglichen wenn Distanz und Dauer allein keine klare Aussage ermöglichen. Dieser speziellere Regelblock umfasst die Regeln 10 bis 16:

10. RULE 10 : IF distance IS far AND duration IS medium\_long AND repeat\_meetings IS true THEN consultation IS medium\_truth;
11. RULE 11 : IF distance IS far AND duration IS medium\_long AND repeat\_meetings IS true AND mean\_distance\_decreased IS true THEN consultation IS high\_truth;
12. RULE 12 : IF distance IS far AND duration IS long AND mean\_distance\_decreased IS true THEN consultation IS high\_truth;
13. RULE 13 : IF distance IS far AND duration IS long AND repeat\_meetings IS true THEN consultation IS high\_truth;
14. RULE 14 : IF distance IS near AND duration IS short AND repeat\_meetings IS true AND mean\_distance\_decreased IS true THEN consultation IS medium\_truth;
15. RULE 15 : IF distance IS near AND duration IS short AND repeat\_meetings IS true AND mean\_distance\_decreased IS true THEN consultation IS medium\_truth;
16. RULE 16 : IF distance IS near AND duration IS medium\_long AND repeat\_meetings IS true THEN consultation IS high\_truth;

Eine grundlegende Fuzzy Logik Regelbasis, basierend auf dem verfügbaren Expertenwissen, steht damit zur Verfügung um einen Proof of Concept zu unternehmen. Das Herzstück der geplanten Feedbacklösung, die intelligente Beratungskontakterkennung mit Beacons, ist somit konzipiert. Die durch diese Regeln produzierten Ergebnisse werden im Abschnitt „6. Systemtest der Fuzzy Logik Implementierung“ untersucht. Als nächstes kann mit der Systemanalyse des umgebenden Feedbacksystems begonnen werden.

# 4 Systemanalyse

In diesem Abschnitt wird eine systematische Systemanalyse für die zu entwickelnde Feedback Lösung erarbeitet. In Gesprächen mit strategischen Partnern und zukünftigen Pilotkunden wird zuerst eine marktorientierte Anforderungsanalyse durchgeführt um spezifische Anwendungsszenarien der generischen Lösung für zukünftige Kunden zu identifizieren. Die so gewonnenen Kundenanforderungen werden analysiert und bilden die Basis für das Systemdesign und die technische Konzeption. Es werden Anwendungsfälle spezifiziert und die Systemgrenzen festgelegt. Anschließend werden die Systemabläufe und die Komponenten des Systems modelliert. Hierbei wird nach einem Top-Down-Verfahren vorgegangen um ausgehend vom Entwurf des Gesamtsystems eine immer detailliertere Sicht des Systems herauszuarbeiten. Um eine allgemein verständliche Vereinheitlichung der Terminologien und Standardisierung der Notation zu erreichen, orientiert sich diese Arbeit bei der Modellierung an den Entwurfsverfahren und Konzepten der Unified Modeling Language (UML 2.1.1) der Object Management Group<sup>28</sup>.

## 4.1 Anforderungsanalyse

### 4.1.1 Anforderungserhebung

Basierend auf der Vision und Geschäftsidee wurde eine intensive Suche nach branchenübergreifenden Pilotkunden durchgeführt. Dabei wurde Wert darauf gelegt, Pilotkunden aus verschiedenen Bereichen des stationären Handels zu gewinnen, um eine möglichst weitreichende Marktrelevanz der zu entwickelnden Lösung sicher zu stellen. Eine erste Erkenntnis während der Suche war, dass eine elaborate neue Feedbacklösung zunächst nur für Händler mit beratungsintensiven Produktgruppen ausreichend relevant ist. Außerdem war ein steigendes Interesse bei Filialisten festzustellen, die anteilig viel Stammkundschaft haben und Kunden die einen relativ hohen Einkaufsbetrag je Ladenbesuch ausgeben. Somit konnte die Gruppe der relevantesten Zielkunden auf Einzelhändler mit komplexen Produktgruppen, Händler im Premiumsegment und Großhändler eingegrenzt werden.

Als Ergebnis dieser Suche konnten zwei strategische Partner bzw. Pilotkunden für die Anforderungserhebung gewonnen werden: Porsche Niederlassung Hamburg GmbH (im Folgenden Porsche genannt) und Metro Cash & Carry, Deutschland – Region Nord (im

---

<sup>28</sup><http://www.omg.org/spec/UML/2.1.1/>, 10.5.2016



Folgenden Metro genannt). Beide Partner haben sich zudem in unverbindlichen Absichtserklärungen bereit erklärt, in Zukunft den Prototypen des Systems zu Testzwecken auf der Verkaufsfläche einzusetzen und Feedback während der Entwicklung der Lösung zu geben. Es war im Vorfeld möglich, beide Pilotkunden von der Vision einer intelligenten Beratungserkennung zur automatisierten Feedbacksammlung zu begeistern. Das Konzept, die Rabatt- und Bonuspunktstrategien mit der Sammlung von Kundenfeedback zu verbinden, wurde als äußerst interessant aufgefasst. Abweichungen in den Vorstellungen gab es nur bei der Frage, auf welchem Level die Feedbackdaten zusammenzufassen und zu anonymisieren sind. Grundsätzlich wurde damit die Validität der Vision am Markt bestätigt und der Kern der vorgeschlagenen Lösung, eine intelligente Beratungskontakterkennung mittels Beacons, ist als generisches System auf potenzielle Kunden in verschiedensten Branchen mit nur geringen Anpassungen anwendbar.

Durch Befragungen der fachseitigen Experten vor Ort bei den Pilotkunden konnten die wichtigsten Anforderungen, an das System herausgearbeitet werden. Zu unterscheiden sind dabei die allgemeinen Anforderungen, die beiden Pilotkunden gemein sind und zusätzliche kundenspezifische Anforderungen die auf dem generellen System aufsetzen. Diese Ergebnisse sind bereits als ein Indiz zu bewerten, dass das System weit möglichst modular konzipiert werden sollte, um auf spezifische Kundenzusatzwünsche flexible und schnell eingehen zu können.

Allgemeine Anforderungen:

1. Feedback soll automatisch und kontinuierlich gesammelt werden
2. Die Feedbackfragen müssen spezifisch genug sein, um konkrete Entscheidungsgrundlagen zu bieten
3. Es soll möglich sein für unterschiedliche Situationen unterschiedliche Feedbackfragen an die Kunden zu stellen
4. Das Beratungsgespräch darf nicht durch die Feedbackaufforderung unterbrochen werden
5. Kundenbefragung soll noch auf der Verkaufsfläche möglich sein
6. Der Zeitpunkt/Auslöser der Feedback Aufforderung soll festlegbar sein
7. Stammkunden sollen wiedererkannt werden, um Trends in der individuellen Kundenzufriedenheit identifizieren zu können
8. Bei bestimmten Trendentwicklungen oder sehr negativem Feedback soll automatisch ein Alert ausgelöst werden
9. Zusätzliche Beacons sollen von Managern flexibel an interessanten Punkten auf der Verkaufsfläche positioniert werden können, um den ungefähren Weg der Kunden nachzuvollziehen zu können (ohne indoor Positionsortung) und Metadaten über Verweildauer, Reihenfolge und anschließendes Kaufverhalten sammeln zu können

Metro spezifische Anforderungen:

1. Feedback darf nicht zu individuellen Fachverkäufern zuzuordnen sein (Anonymisierung und Zusammenfassung auf Abteilungsebene)
2. Kunden sollen zusätzlich kurz vor dem Betreten der Verkaufsfläche über ihre Erwartungen an den bevorstehenden Einkauf befragt werden können

Porsche spezifische Anforderungen:

1. Individuelle Fachverkäufer sollen ihren Kunden zugeordnet werden können
2. Feedback soll für individuelle Fachverkäufer gegeben werden können
3. Fachverkäufer sollen durch ein zusätzliche App informiert werden, wenn ein Kunde die Verkaufsfläche betritt und zusätzliche Kundeninformationen auf ihrem Mobilgerät zur Verfügung gestellt bekommen

#### 4.1.2 Bewertung der Kundenanforderungen

In diesem Abschnitt werden die gesammelten Anforderungen überprüft und bewertet. Im Zuge der ausführlichen Gespräche mit den Pilotkunden ist die Qualität der allgemeinen Anforderungen bereits auf Notwendigkeit, Korrektheit (Widerspruchsfreiheit), Machbarkeit und Nützlichkeit überprüft worden. Zudem ist die Reihenfolge aller Anforderungen entsprechend ihrer Priorität für die Kunden sortiert worden. Auf eine vollständige Qualitätsprüfung entsprechend IEEE Kapitel 4.3 wird im Rahmen dieser akademischen Arbeit aus Effizienzgründen verzichtet.

Die Entwicklung des Prototypen im Rahmen dieser Arbeit orientiert sich am Konzept des Minimum Viable Products (MVP).<sup>29</sup> Dementsprechend soll der Prototyp nur gerade ausreichend Leistungsmerkmale beinhalten, um die zentrale Markthypothese des Produktes zu testen und Erkenntnisse für die marktrelevante Weiterentwicklung zu gewinnen. Daher konzentriert sich diese Arbeit auf die Entwicklung des innovativen Teils des Systems zur Feedback Sammlung. Ein Backend bzw. Frontend für die Analyse und Darstellung des gesammelten Feedbacks ist zwar für die Lieferung der Analytics an die Kunden notwendig, wird jedoch erst im Anschluss an diese akademische Arbeit hinzugefügt werden.

Vor diesem Hintergrund liegen die Anforderungen 1 bis 7 der allgemeinen Anforderungen innerhalb des Scope dieser Arbeit und können als Grundlage für das Systemdesign dienen. Die in Anforderung 8 geforderten Alerts werden nur soweit miteinbezogen, als dass diese innerhalb des hier zu entwickelnden Softwaresystems ausgelöst werden. Eine Schnittstelle, um die Alerts zusätzlich auf externen Systemen anzeigen zu können, wird als außerhalb des

---

<sup>29</sup> Eric Ries: *Lean Startup. How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*, S. 93, Portfolio Penguin, 2011

Scope dieser Arbeit gewertet. Außerdem wird die allgemeine Anforderung 9 zunächst als eventuelle spätere Zusatzfunktionalität gewertet und liegt somit auch außerhalb des Scope. Aus Gesprächen mit der Führungsebene bei der Tchibo GmbH und Einzelhandelsexperten bei The Boston Consulting Group ist hervorgegangen, dass die Metro spezifische Anforderung 1 für die Mehrheit der potenziellen Kunden essenziell ist. Anforderung 2 ist eine sehr einfach umzusetzende Erweiterung des Systems. Deshalb gehören beide Metro-spezifischen Anforderungen in das Systemdesign.

Die Porsche spezifischen Anforderungen 1 und 2 entsprechen der Grundfunktionalität der eigentlichen Vision der Lösung und könnten in Zukunft eine attraktive Funktion für internationale Kunden der Lösung sein (z.B. in den USA). Zwar stehen diese Anforderungen im Widerspruch zu den Metro-spezifischen Anforderungen, jedoch stellt eine entsprechende Anpassung auf den jeweiligen Kunden keinen erheblichen Mehraufwand dar. Somit fließen beide Anforderungen in das Systemdesign mit ein. Porsches Anforderung 3 hingegen wird als außerhalb der Scope dieser Arbeit gewertet und wird nur soweit in beachtet, als dass das hier entwickelte System möglichst einfach zu erweitern ist, um diese Anforderung in Zukunft hinzuzufügen.

Ergebnis der Anforderungsanalyse sind die Grundfunktionalitäten für das zentrale Anwendungsszenario der Lösung. Eine Iteration durch den Feedbackprozess der Lösung stellt sich entsprechend der Anforderungsanalyse wie folgt dar. Ein Einkäufer hat ein Bedürfnis nach Produkten und Beratung und sucht deshalb die Filiale auf. Wenn sich der Einkäufer der Verkaufsfläche nähert wird die App automatisch aktiv und es können nun bei Bedarf Feedbackaufforderungen ausgelöst werden. In der Filiale erkennt die künstliche Intelligenz der App, wenn ein Beratungsgespräch stattgefunden hat. Im Anschluss, wenn sich der Einkäufer der Kasse nähert, wird eine situationsrelevante Feedbackaufforderung ausgelöst. Hierbei besteht die Möglichkeit dem Einkäufer als Anreiz einen situations- bzw. Kundenrelevanten Rabattcode oder Bonuspunkte als Dankeschön für das Feedback-Geben anzubieten. Rabattcodes erhöhen den Anreiz sofort etwas zu kaufen oder später wieder auf die Verkaufsfläche zurückzukehren. Die gesammelten Daten können optional anonymisiert und auf Abteilungsebene zusammengefasst werden. Abschließend sendet die App die Daten an das Backend, wo sie analysiert werden können, um sie als Analytics darzustellen. Die Abbildungen 28 und 29 zeigen die graphischen Darstellungen der zentralen Use-Cases (mit jeweils einer Iteration durch den Feedbackprozess), wie sie mit den nicht-technischen Managern bei Metro und Porsche besprochen wurden.



Abbildung 28: Gewünschtes Anwendungsszenario bei Metro



Abbildung 29: Gewünschtes Anwendungsszenario bei Porsche

## 4.2 Systemdesign und technische Konzeption

### 4.2.1 Festlegung der Systemgrenzen

Zunächst gilt es die verfügbaren Anwendungsfälle festzulegen, um die Systemgrenzen im Rahmen dieser Arbeit zu definieren. Hierzu werden im Anwendungsfalldiagramm in Abbildung 30 für das Subjekt 1, der Feedback Sammlung auf der Verkaufsfläche, die relevanten Akteure sowie die mit ihnen assoziierten Anwendungsfälle graphisch dargestellt.

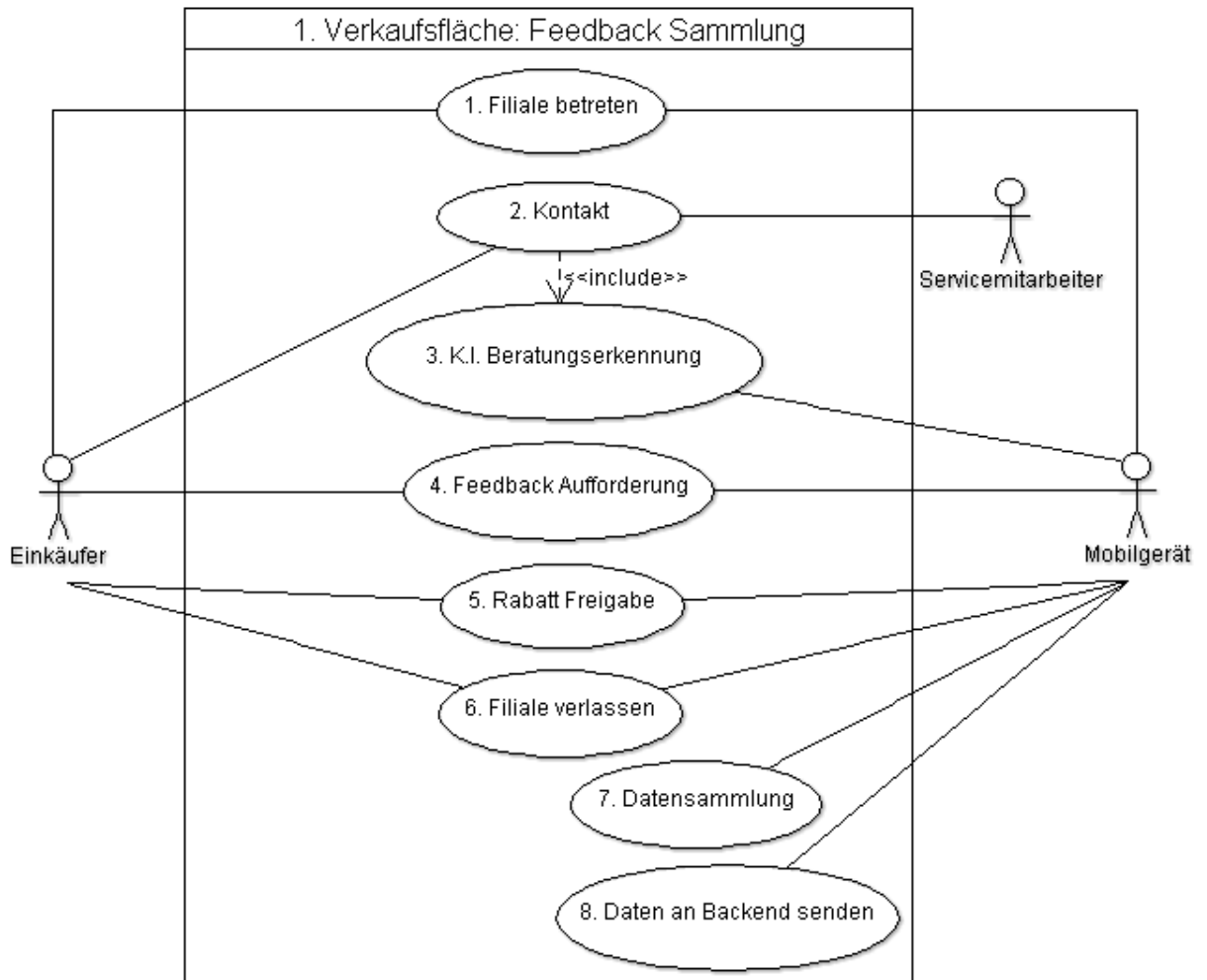
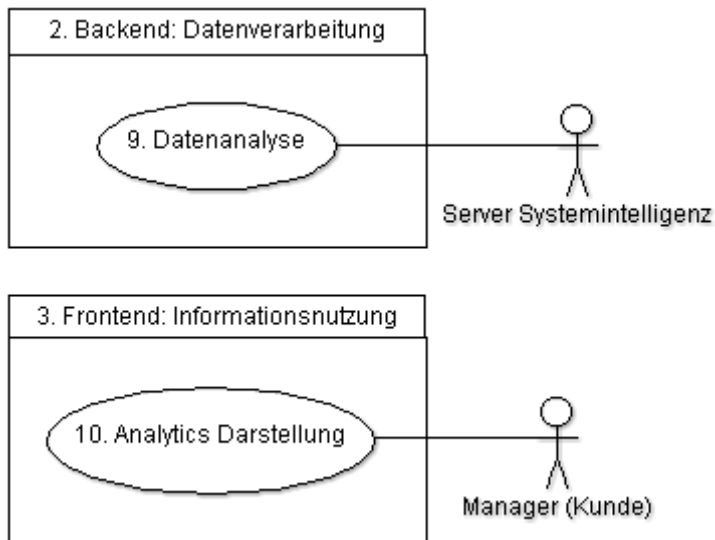


Abbildung 30: Systemgrenzen und Anwendungsfälle für die Feedback Sammlung

Der Einkäufer, als Akteur des Systems, kann 5 Anwendungsfälle in Anspruch nehmen. Im Anwendungsfall 1 wird das System automatisch aktiv, wenn der Einkäufer mit seinem Smartphone die Filiale betritt. Ein mit einem Beacon ausgestatteter Servicemitarbeiter ist ein weiterer Akteur des Systems. Wenn der Einkäufer mit einem Servicemitarbeiter für eine bestimmte Zeit eine bestimmte räumliche Nähe hat, besteht ein Kontakt (Anwendungsfall

2). Dieser Anwendungsfall enthält immer auch die K.I. Beratungserkennung des mobilen Systems, es wird also indirekt auch der Anwendungsfall 3 in Anspruch genommen. Das Mobilgerät kann als Akteur Feedback Aufforderungen auslösen und der Einkäufer kann auf diese reagieren. Wenn der Einkäufer der Feedback Aufforderung Folge geleistet hat, löst das Mobilgerät eine Rabatt Freigabe aus, die der Einkäufer somit in Anspruch nehmen kann. Wenn der Einkäufer die Filiale verlässt (Anwendungsfall 6), wird dies vom mobilen System registriert. Das Mobilgerät beansprucht außerdem die Anwendungsfälle 7 und 8 zur Datensammlung und zum Senden der Daten an ein externes Backend.



Im Rahmen dieser Arbeit begrenzen sich die Systemfunktionen auf die in Subjekt 1 dargestellten Anwendungsfälle. Allerdings wird es in Zukunft notwendig sein zusätzliche Subjekte und Systemfunktionen zu implementieren um eine vollständige Feedbacklösung zu bieten. Diese notwendigen Zusätze sind in Abbildung 31 dargestellt, liegen allerdings außerhalb der hier zu behandelnden Systemgrenzen.

Abbildung 31: Zukünftige zusätzliche Systemfunktionen

#### 4.2.2 High-Level Systemzustände

Zunächst ist es hilfreich zu erfassen, welche möglichen Zustände das System durchlaufen kann und in welcher Beziehung diese zueinander stehen. Das Zustandsdiagramm in Abbildung 32 dient als Grundlage der notwendigen Funktionen und als Referenz des Systemverhaltens für die im Folgenden zu realisierenden Aktivitäten des Systems. Auf der Makroebene des Gesamtsystems sind nur 11 relevante Zustände erreichbar. Das System wird aktiv, wenn die App hochgefahren wird. Dies kann durch das Betriebssystem (über den iBeacon Standard) automatisch oder durch den App-Nutzer manuell geschehen. Sofern die App über die Beaconsignale registriert, dass sich das Mobilgerät auf der Verkaufsfläche befindet, wechselt es in den Zustand „Verkaufsfläche betreten“. Anschließend geht das System dann direkt zur Beacon Signalmessung im gleichnamigen Zustand über. Hier können nun zwei unterschiedliche Ereignisse unterschiedliche Übergänge auslösen:

1. Zum einen kann sich der Einkäufer einem Servicemitarbeiter nähern und das System geht in den Zustand „Möglicher Beratungskontakt“ über. In diesem Zustand werden die Signalwerte von in Frage kommenden Servicemitarbeitern aufgezeichnet, und erst wenn sich der Einkäufer wieder aus der Beratungsdistanz entfernt, geht das System in den Zustand „Beratungserkennung“ über. In diesem Zustand ist die zu realisierende künstliche Intelligenz anzusiedeln. Mithilfe der KI muss, basierend auf nicht eindeutigen und unsicheren Signalmesswerten, zeitnah eine eindeutige Entscheidung getroffen werden, die scharfe Zustandsübergänge ermöglicht. Im Anschluss an diese Bewertung des möglichen Beratungskontaktes geht das System wieder in den Zustand „Beacon Signalmessung“ zurück.
2. Zum anderen kann über das stationäre iBeacon Signal der Einkauf als abgeschlossen bewertet werden, wodurch das System in den Zustand „Einkauf abgeschlossen“ übergeht. Diese Bewertung ist basierend auf Signalschwellwerten auch ohne KI eindeutig entscheidbar.

Mittels festzulegender Zeitspannen kann aus diesem Zustand heraus entschieden werden, ob ein Feedback würdiger Einkauf überhaupt stattgefunden haben kann, oder ob der Einkäufer keine relevante Zeit auf der Verkaufsfläche verbracht hat. Sollte die Zeit zu kurz gewesen sein, werden die verfügbaren gesammelten Daten trotzdem persistiert und können z.B. als Grundlage für eventuelle Systemverbesserungen dienen. Sollte ausreichend Zeit auf der Verkaufsfläche vergangen sein, geht das System vom Zustand „Einkauf abgeschlossen“ in den Zustand „Feedback Auswahl“ über. Hier sind, basierend auf eventuell erkannten Beratungskontakten, situationsrelevante Feedbackfragen auszuwählen. Es wird eine Notification an den Einkäufer ausgelöst und das System geht in den Zustand „Feedback geben“ über. Der Einkäufer kann hier Feedback geben oder auch nicht. In beiden Fällen geht das System durch das Input des Einkäufers in den Zustand „Daten Persistierung“ über, speichert die Daten und beendet die App automatisch wieder.

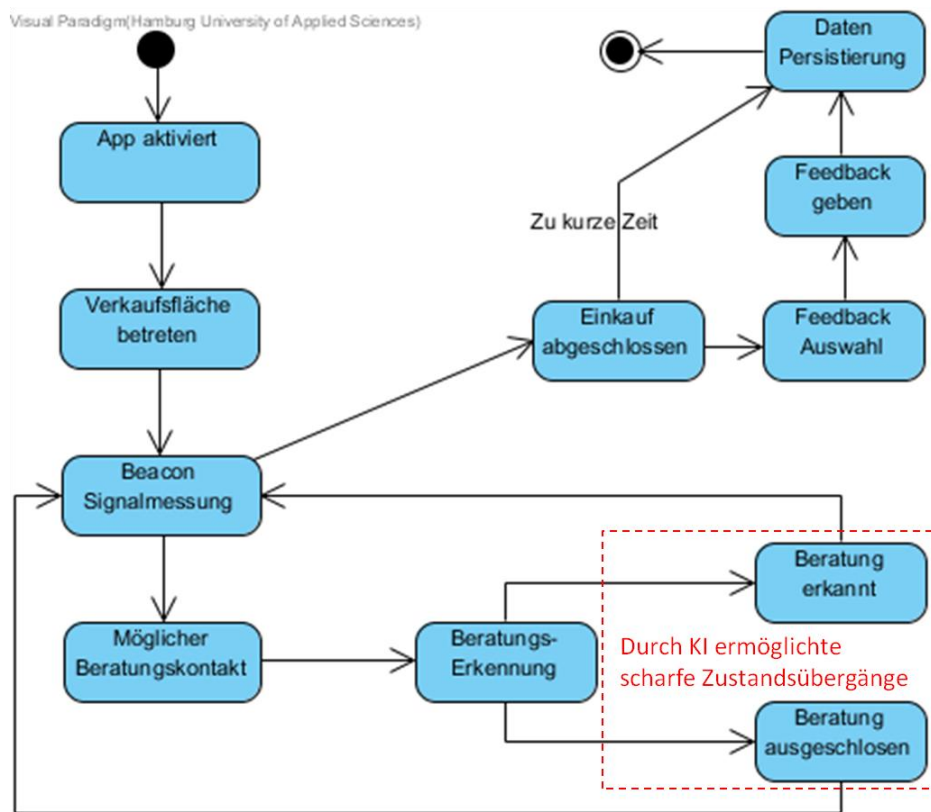


Abbildung 32: Zustände des Systems (App Lifecycle)

### 4.2.3 High-Level Systemabläufe

Um die verfügbaren Anwendungsfälle als Funktionalitäten im IT-System zu realisieren, werden im Folgenden die zentralen Prozesse und deren Abläufe herausgearbeitet. Hierbei wird nach einem Top-Down-Verfahren vorgegangen, bei dem auf einem niedrigen Detaillierungsgrad (High Level) mit der Modellierung der groben Aktivitäten des Systems begonnen wird, um zunehmend detailliertere Abläufe der darunter liegenden Prozesse herauszuarbeiten.

Die zentralen Aktivitäten des Feedbacksammelns finden auf den Mobilgeräten der Einkäufer statt. Dabei bildet die intelligente Beratungskontakterkennung das Herzstück des Feedbacksystems. Die präzise Erkennung von Beratungskontakten in angemessener Zeit auf der Verkaufsfläche ist von zentraler Bedeutung für die Kundenrelevanz des Systems. Ziel ist es, möglichst viele aussagekräftige Kundeninformationen zu sammeln und zur Analyse an ein Backend senden zu können. Gleichzeitig dürfen diese Abläufe nicht übermäßig viel Speicherplatz oder Leistung auf den Mobilgeräten der Einkäufer erfordern. Somit gilt es,



eine sowohl möglichst effektive als auch effiziente Implementierung der Anforderungen zu realisieren.

Des Weiteren ist zu beachten, dass es sich bei diesem System um einen ersten Entwurf eines völlig neuen Produkts für ein Startup Projekt handelt. Somit ist es grundsätzlich zu erwarten, dass sich die Anforderungen in naher Zukunft noch ändern können. Aufgrund dieser Rahmenbedingungen ist es sinnvoll, mittels eines Lean Management Ansatzes die kontinuierliche Verbesserung des Systems aufgrund neugewonnener Informationen im Austausch mit den Pilotkunden in den Fokus der Entwicklung zu rücken. Flexible Anpassungsfähigkeit bei der Suche nach maximaler Kundenrelevanz und Marktreife ist ein entscheidender Erfolgsfaktor für junge Unternehmen.<sup>30</sup>

Aus diesen Gründen wird das hier zu entwickelnde Feedbacksystem grundsätzlich modular konzeptioniert und Schnittstellen-basiert aufgebaut. Durch Kapselung von Funktionen und die Definition von generell wirksamen Schnittstellen zwischen in sich abgeschlossenen Systemteilen können Veränderungen, Tests und Optimierungen lokalisiert und schnell vorgenommen werden ohne das andere Systemteile in Mitleidenschaft gezogen werden. Hinzukommt, dass bei der zukünftigen Weiterentwicklung im Gründerteam die Teammitglieder an verschiedenen Komponenten parallel arbeiten können, ohne sich gegenseitig aufzuhalten oder einander zu behindern. Im Sinne dieses Ansatzes erscheint der Einsatz von objektorientierten Programmierkonzepten besonders passend und wird fortan in dieser Arbeit für die Implementierung eingesetzt.

Die grundsätzlich zu realisierenden Abläufe des Feedback Systems sind im Aktivitätsdiagramm in Abbildung 33 graphisch dargestellt. Der Prozess beginnt mit dem Erscheinen eines Einkäufers (mit der App auf dem Smartphone) an der Filiale. Diese Aktion ist gekennzeichnet durch den roten Punkt 0 (P0) in der Abbildung. Ein am Eingangsbereich angebrachter stationärer Beacon mit dem iBeacon Standard ermöglicht es die App über das Betriebssystem (ab Android 4.3 oder iOS) des Smartphones automatisch zu starten, wenn sich der Einkäufer nähert. Voraussetzung hierfür ist es allerdings, dass der Einkäufer Bluetooth an seinem Smartphone aktiviert hat. Soweit notwendig wird der Einkäufer bei Installation und beim manuellen Starten der App über einen Pop-up Dialog aufgefordert Bluetooth zu aktivieren. Im Fall, dass die App nicht aktiv ist und der Einkäufer Bluetooth nicht aktiviert hat, wenn er die Verkaufsfläche betritt, kann das Feedback System seine Funktion nicht erfüllen. Wenn Bluetooth aktiviert ist, startet die App automatisch und registriert den zeitlichen Beginn des Einkaufs (P1). Während des Einkaufs misst die App die Signalstärken der BLE Beacons auf der Verkaufsfläche (P2). Dabei können drei relevante Szenarien eintreten:

1. Wenn der Einkäufer nah genug an einen Servicemitarbeiter herantritt und damit der Schwellwert der Beacon Signalstärke überschritten wird, wird die intelligente Beratungskontakterkennung des Systems aktiv (P3). Wird ein Beratungskontakt erkannt, wird dieser zwischen gespeichert und das System misst weiter Beacon Signale. Eventuelle weitere Beratungskontakte werden ebenfalls

---

<sup>30</sup> Pawel Gorecki / Peter R. Pautsch: *Lean Management*, S.102ff, Hanser, 2015

zwischengespeichert. Die Informationen über diese Beratungskontakte werden später als Faktoren bei der Auswahl situationsrelevanter Feedbackfragen in Betracht gezogen.

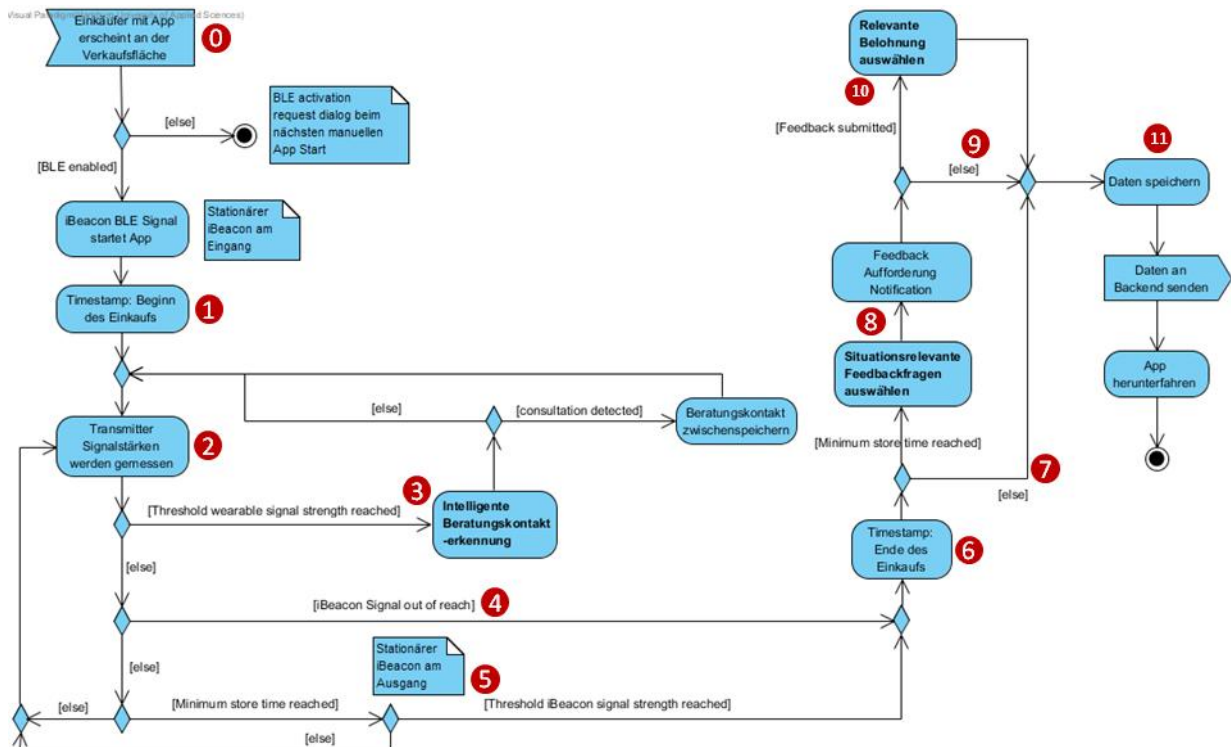


Abbildung 33: Aktivitätsdiagramm der High Level Abläufe des Feedback Systems

2. Ein zweites relevantes Szenario beim Messen der Beacon Signalstärken ist es, wenn das Signal des stationären iBeacon am Ausgang dauerhaft außer Reichweite kommt (P4). In diesem Fall hat der Einkäufer die Verkaufsfläche verlassen und der Einkauf ist beendet.
3. Letztlich kann sich der Einkäufer dem iBeacon am Ausgang soweit nähern, dass der Signalschwellwert überschritten wird (P5). In manchen Filialen liegen Eingang und Ausgang im selben Bereich und manche Einkäufer verweilen im Eingangsbereich oder gehen kurzzeitig auf und ab. In solchen Situationen gilt es zu verhindern, dass diese Einkäufer schon vor ihrem eigentlichen Einkauf eine Feedbackaufforderung erhalten. Deshalb ist eine Mindesteinkaufszeit festzulegen (basierend auf Erfahrungswerten der Filialisten) bevor davon ausgegangen werden kann, dass ein Einkäufer sich dem Ausgangs-iBeacon nähert, weil er einen Einkauf abgeschlossen hat.

Sind die Bedingungen der letzteren beiden Szenarien erfüllt, wird der Einkauf als beendet registriert (P6). Für den Fall, dass der iBeacon außer Reichweite gekommen ist, wird überprüft, ob die Mindesteinkaufszeit überschritten wurde. Andernfalls kann davon ausgegangen werden, dass der Einkäufer, aus nicht relevanten Gründen dem iBeacon nahe genug gekommen ist, als dass die App gestartet wurde aber kein Einkauf stattgefunden hat (P7).

Allerdings soll eine Feedbackaufforderung für den Fall, dass ein Einkäufer die Mindesteinkaufszeit auf der Verkaufsfläche verbracht hat gesendet werden. Dieses Szenario soll Fälle abdecken, in denen ein Einkäufer zum Eingang auf die Verkaufsfläche tritt und aus unbekanntem Gründen nach kurzer Verweildauer durch den Eingang wieder geht. Feedback über die Gründe für diese Art von Verlust potenzieller Einkäufer die sich schon auf der Verkaufsfläche befunden hatten, ist für die Kunden des Systems äußerst interessant.

Immer wenn die Mindesteinkaufszeit überschritten wurde, wählt das System situationsrelevante Feedbackfragen basierend auf den zuvor gesammelten Daten über das Einkäuferverhalten und stattgefundenen Beratungskontakte (P8) aus. Eine entsprechende Feedbackaufforderung wird als Notification auf dem Mobilgerät des Einkäufers ausgelöst. Diese Notification weist das gleiche Verhalten auf wie externe Nachrichten an das Gerät (z.B. SMS, Email, Kalender-Erinnerungen) und wird durch einen Signalton/Vibration begleitet und auch auf dem Sperrbildschirm angezeigt. Die App bleibt solange aktiv, bis der Einkäufer auf diese Aufforderung reagiert und entweder die Aufforderung ablehnt (P9) oder Feedback abgibt (P10).

In allen drei Fällen (P7, P9, und P10) werden die gesammelten Daten gespeichert (P11) und können an dieser Stelle an ein (in Zukunft zu entwickelndes) Backend gesendet werden – sofern eine Datenverbindung besteht. Abschließend wird die App automatisch wieder heruntergefahren, da nun alle relevanten Aktivitäten abgeschlossen wurden.

Die detaillierten Abläufe hinter den Aktivitäten P3, P8 und P10 werden im Folgenden schrittweise verfeinert und konkret ausgearbeitet. Eine Parallelität von mehreren Beratungserkennung-Prozessen im Prototyp wird aufgrund der langen Zeitspannen im Bereich von menschlichen Interaktionen nicht für notwendig angesehen.

#### **4.2.4 Detaillierte Systemabläufe der Beratungskontakterkennung**

In Abbildung 34 werden die detaillierten Abläufe der intelligenten Beratungskontakterkennung unter Verwendung einer Fuzzy Logik Implementation dargestellt. Angestoßen wird diese Aktivität durch das Eintreten des Ereignisses, dass der Einkäufer sich einem Servicemitarbeiter nah genug genähert hat, als dass der Signalschwellwert des Mitarbeiterbeacons überschritten wird (roter Punkt P0). Die Personen sind sich nun nahe genug, dass eine Beratung stattfinden könnte und die intelligente Beratungskontakterkennung wird notwendig. Als erstes wird ein Timer gestartet (P1) um messen zu können wie lange sich die zwei Personen innerhalb des Schwellwertes aufhalten. Das System beginnt die Beacon Messwerte der Signale der

Mitarbeiterbeacons und des stationären iBeacons aufzuzeichnen (P2). Tritt nun eine von zwei Situationen ein, kann der Prozess fortschreiten:

Zum einen kann der Timer für die festgelegte maximale Beratungszeit ablaufen (P3). Diese Zeitspannen muss für den jeweiligen Händler so gewählt werden, dass nach deren Ablauf mit sehr hoher Sicherheit davon ausgegangen werden kann, dass ein Beratungskontakt stattgefunden hat. In diesem Fall ist keine weitere Prüfung notwendig und die Aktivität weist einen Beratungskontakt als Ergebnis aus und wird beendet.

Zum anderen kann der Signalschwellwert des Mitarbeiterbeacons wieder unterschritten werden (P4). Dies bedeutet das potenzielle Ende des Beratungskontaktes. In diesem Fall muss als nächstes überprüft werden ob eine bestimmte Mindestberatungszeit abgelaufen ist (P5). Diese Zeitspanne muss so gewählt werden, dass bei einer kürzeren Kontaktzeit keine feedbackwürdige Beratung möglich ist. Sollte diese Zeitspanne noch nicht abgelaufen sein (P6), war der potenzielle Beratungskontakt kein tatsächlicher Beratungskontakt.

Wenn aber eine ausreichende Zeitspanne abgelaufen ist, werden aus dem aufgezeichnetem Signalverhalten die für die gewählten Inputvariablen notwendigen Charakteristika ausgewertet und als crisp Inputwerte abstrahiert (P7). Diese Inputvariablen werden dann der Fuzzy Logik Implementation zur Verfügung gestellt um zu einer Entscheidung zu kommen. Wie im Abschnitt 3.2 bereits erläutert, muss nun zunächst die Fuzzification (P8) der scharfen Inputwerte vorgenommen werden. Anschließend kann der Prozess aufgespalten werden und es können die fuzzy Eingabewerte parallel von den eingesetzten Fuzzy Logik Regeln ausgewertet werden (P9). Sobald die Ergebnisse dieser Auswertung vorliegen findet eine Synchronisation des Prozesses statt und die Ergebnisse werden aggregiert und durchlaufen die Defuzzification um scharfe Ergebniswerte an das Feedbacksystem liefern zu können (P10).

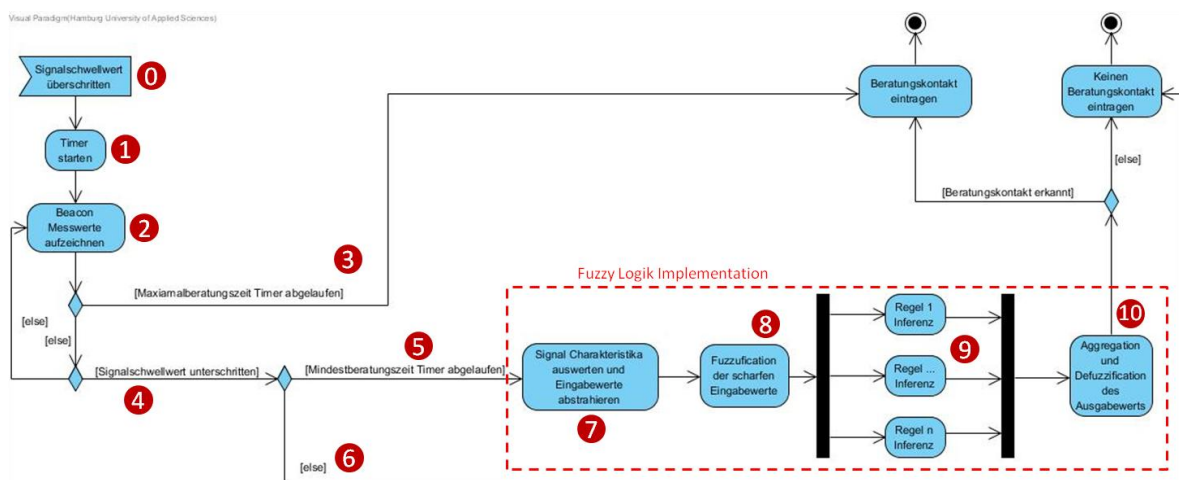


Abbildung 34: Detaillierte Abläufe der intelligenten Beratungskontakterkennung mit Fuzzy Logik Implementation

Basierend auf diesen Ergebnissen kann nun die eigentliche Entscheidung getroffen werden, ob ein Beratungskontakt stattgefunden hat oder nicht. Die letzten Aktivitäten tragen das entsprechende Ergebnis für den potenziellen Beratungskontakt ein und der Ablauf wird beendet.

#### 4.2.5 Systemstruktur und statische Beziehungen

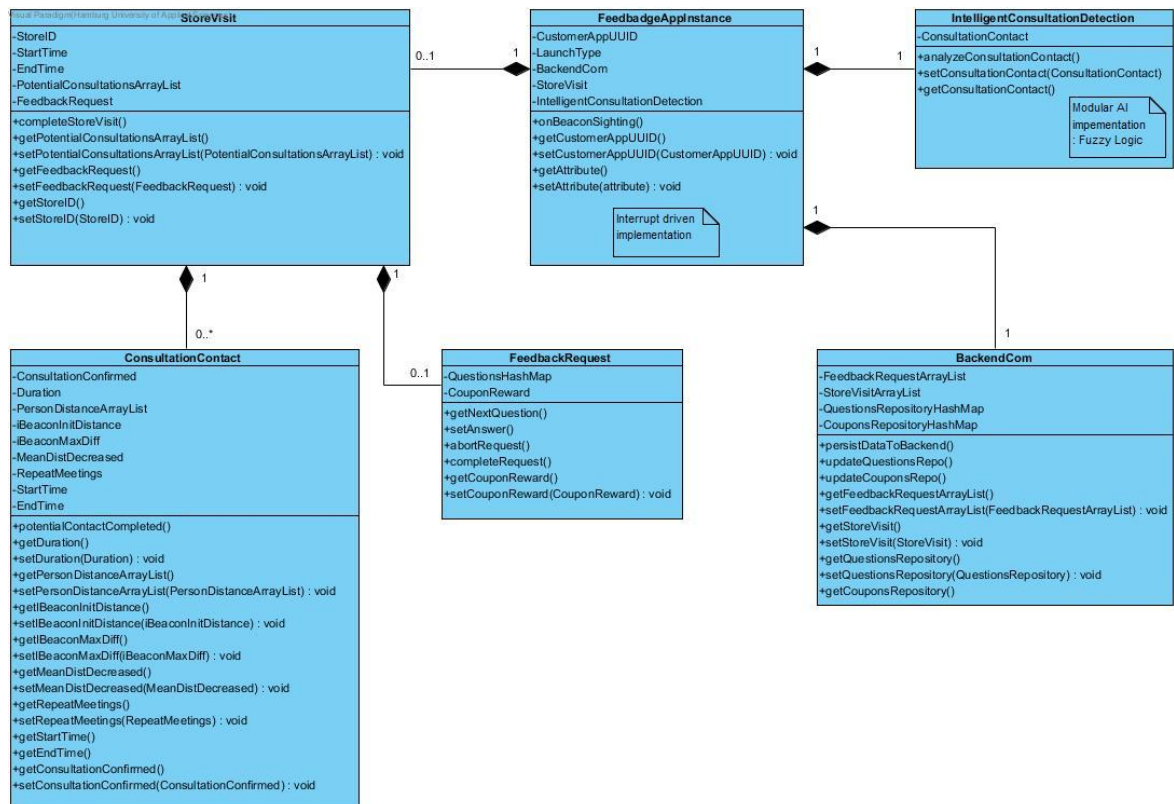


Abbildung 35: Klassendiagramm der mobilen Feedbacklösung

# 5 Implementierung

## 5.1 Android

Die Implementierung des Prototyps wird mit Android Studio 2.1 für Android 5.1 (API Version 23)<sup>31</sup> als Zielversion entwickelt. Die minimal Anforderung ist dabei Android 4.3 (API Version 18), weil diese die erste Android Version ist, die den benötigten BLE Standard für die Beacon Signale unterstützt. Kunden mit niedrigeren Android Versionen sind somit als potenzielle Nutzer der Lösung leider ausgeschlossen.

Eine in Zukunft zu beachtende Problemstellung ist, dass mit API Version 23 (Android 6.0) die für die Feedbacklösung unbedingt notwendigen Android Permissions BLUETOOTH und BLUETOOTH\_ADMIN neuerdings zur Laufzeit vom Nutzer gefordert werden müssen<sup>32</sup>. Ein zentrales Konzept für die Usability (Gebrauchstauglichkeit) und Benutzerfreundlichkeit der Feedbacklösung ist ein hohes Maß an Automatisierung und selbständige Entscheidungsfähigkeit der App. Es sollen keine Handlungen vom Nutzer notwendig sein, wenn die Verkaufsfläche betreten wird. Die App soll unmerklich ihre Aufgaben erfüllen bis eine situationsrelevante Feedbackaufforderung das Input des Nutzers erfordert. Um möglichst viele Nutzer für regelmäßiges Feedback zu gewinnen muss eine Lösung für diese Problematik gefunden werden bevor mehr Nutzer zu Android 6.0 wechseln (zurzeit sind es nur 7.5%)<sup>33</sup>. Die Suche nach einer sinnvollen Lösung kann am besten im direkten Austausch mit den Pilotkunden und deren Endkunden stattfinden. Der hier entwickelte Prototyp bildet eine hervorragende Grundlage für die dafür notwendigen Gespräche.

## 5.2 Gimbal

Gimbal Beacons wurden wegen des hohen Abstraktionsgrades des zugehörigen Gimbal SDK gewählt um die zügige Implementierung eines Prototypen zu erleichtern. Allerdings hat sich diese Wahl als problematisch herausgestellt. Zum einen traten unvorhersehbare Fehler auf bei der kontinuierlichen Ortung von Beaconsignalen mit dem Gimbal SDK für Android. Aus ungeklärten Gründen hört das System nach etwa 53 Sekunden auf die Beacon Signale zu empfangen. Der technische Support für Gimbal Kunden war nicht in der Lage diese Fehlfunktion zu erklären oder eine Lösung anzubieten. Abschließend wurde die Smartphone

---

<sup>31</sup> Android 5.1 APIs: <https://developer.android.com/about/versions/android-5.1.html>, 1.6.2016

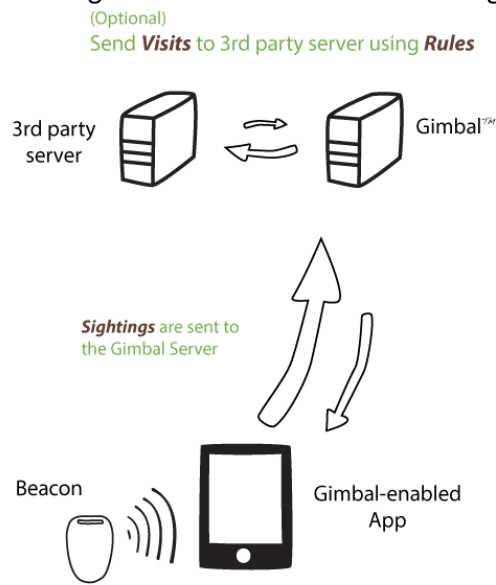
<sup>32</sup> Android 6.0 Changes: <https://developer.android.com/about/versions/marshmallow/android-6.0-changes.html>, 1.6.2016

<sup>33</sup> Dashboards: <https://developer.android.com/about/dashboards/index.html?hl=es>, 1.6.2016

Hardware für die Fehlfunktion verantwortlich gemacht. Allerdings trat das gleiche Problem auf zwei sehr unterschiedlichen Geräten auf:

- Wiko Rainbow Jam – Version 8 mit Android 5.1
- Asus Nexus 7 (2013) mit Android 6.0.1

Eine weitere Problematik bei der Nutzung der Gimbal Beacons ist, dass die Kommunikation zwangsweise, in einem Zwischenschritt, über Gimbal Server laufen muss. Abbildung 36 stellt den genauen Ablauf des Datenflusses graphisch dar.



Apps receive *Sightings* and/or *Visits* from Beacons

Abbildung 36: Datenfluss der Gimbal SDK Kommunikation <sup>34</sup>

Damit laufen potenziell sensible Kundendaten unnötiger Weise durch eine dritte Partei (Gimbal). Die Pilotkunden haben diesbezüglich bereits Bedenken geäußert und würden es vorziehen, wenn ihre Daten nur über die deutschen Server ihres direkten Vertragspartners (Feedbadge) laufen. Mit Beginn der Gründungsphase soll deshalb ein neuer Hardwarepartner gefunden werden.

### 5.3 Fuzzy Logic Library

Um die Fuzzy Logik für die intelligente Beratungskontakterkennung zu implementieren wurde die jFuzzyLogic Bibliothek in Java verwendet. Diese Fuzzy Logik Bibliothek zeichnet sich durch seine Flexibilität und Mächtigkeit aus. Eine große Auswahl Möglichkeiten für typische Zugehörigkeitsfunktionen, Regel Aggregation, Defuzzification Methoden und

<sup>34</sup> Gimbal Documentation, <https://docs.gimbal.com/>, 1.6.2016

Visualisierung der Ergebnisse stehen zur Verfügung.<sup>35</sup> Darüber hinaus bietet die Bibliothek viele weiterführende Funktionen wie z.B. völlig eigene Zugehörigkeitsfunktionen zu definieren oder die Möglichkeit FCL in C++ Code zu übersetzen. Zusätzlich hat sich jFuzzyLogic, während der Implementierung, als äußerst anwenderfreundlich und intuitiv zu bedienen erwiesen. Für die Implementierung des Prototypen wurde die Vollversion der Bibliothek verwendet. Allerdings steht auch eine besonders effiziente „Core Library“ Version speziell für die Entwicklung mit Android für Mobilgeräte zur Verfügung.

jFuzzyLogic implementiert den Fuzzy Control Language (FCL) Standard der Internationalen Elektrotechnischen Kommission<sup>36</sup>. Dies bringt wichtige Vorteile bei der Programmierung mit sich:

- Industrietauglicher, geprüfter Standard einer renommierten internationalen Normierungsorganisation die ihre Normen gemeinsam mit der International Organization for Standardization (ISO) entwickelt
- Die Fuzzy Logik Implementierung wird Programmiersprachen unabhängig und kann in Zukunft leicht portiert werden (z.B. für eine iOS Implementierung)
- Grundsätzliche Trennung der Fuzzy Logik Regeln vom Quellcode fördert ein modulares Design und erleichtert die Anpassung und den Austausch von Regeln

---

<sup>35</sup> Pablo Cingolani / Jesús Alcalá-Fdez: *jFuzzyLogic: a Java Library to Design Fuzzy Logic Controllers According to the Standard for Fuzzy Control Programming*, International Journal of Computational Intelligence Systems, 2013

<sup>36</sup> IEC 61131-7



# 6 Systemtest der Fuzzy Logik Implementierung

Das hier entwickelte Feedbacksystem weist eine sehr geringe Kritikalität auf<sup>37</sup> da es nur geringen materiellen Schaden verursachen kann und Menschen generell nicht gefährdet. Entsprechend wird das System im Rahmen des Proof of Concepts in dieser Arbeit nur verkürzt getestet. Die korrekte Funktionsweise des Quellcodes kann dafür mit Standard jUnit Tests in Android Studio überprüft werden. Die generelle Effektivität des gesamten Systems kann Szenario basiert durch stichprobenartige Testfälle der wichtigsten Beratungssituationen überprüft werden indem das erwartete Sollergebnis mit dem tatsächlichen Systemergebnis verglichen wird. Abschließend kann durch eine große Menge zufälliger Eingabewertekombinationen die Belastbarkeit des Systems zu überprüft werden und eventuelle Blind-Spots durch Outlier Ergebnisse für bestimmte Eingabewertkombinationen aufgedeckt werden. Leider war es im Rahmen dieser Arbeit zeitlich nicht mehr möglich Usability Tests mit dem Prototypen auf der Verkaufsfläche bei den Pilotkunden durchzuführen. Dies soll im weiteren Gründungsprojekt bald möglichst nachgeholt werden.

## 6.1 Ausgewählte Testfälle

Im Folgenden werden die Ergebnisse für fünf Testfälle aufgeführt, welche die zentralen Szenarien für potenzielle Beratungskontakte widerspiegeln. Hierfür wurde zunächst die korrekte Funktionsweise des Systems bei den eindeutigen Fällen überprüft. Die von den Experten als typische Durchschnittswerte bezeichneten Eingabewerte wurden ebenfalls untersucht. Zusätzlich wurden zwei schwierige Fälle mit uneindeutigen Eingabewerten überprüft.

Da keinerlei Performanzschwierigkeiten bei der Rechenleistung aufgetreten sind, wurde als Defuzzification Methode ausschließlich die in der jFuzzyLogic vorimplementierte Methode des Massenmittelpunkts (Center of Gravity) verwendet. Center of Gravity ist eine weitverbreitete Standardmethode und gilt als die attraktivste Defuzzification Methode.<sup>38</sup>

---

<sup>37</sup> Software Sicherheit und Kritikalität: [http://www.informatik.uni-bremen.de/gdpa/part3\\_d/p3si.htm](http://www.informatik.uni-bremen.de/gdpa/part3_d/p3si.htm), 29.05.2016

<sup>38</sup> Timothy Ross: *Fuzzy Logic With Engineering Applications (Third Edition)*, Wiley, 2010

Es wurde überprüft, ob die im Abschnitt „3.3 Beratungskontakterkennung mit Fuzzy Logik“ ausgearbeiteten linearen bzw. nichtlineare Zugehörigkeitsfunktionen signifikant unterschiedliche Ergebnisse liefern.

Virtuelle Szenarien als stichprobenartige Testfälle im Systemtest:

Testfall	1	2	3	4	5
Szenario	Eindeutige Beratung	Eindeutig keine Beratung	Durchschnittsberatung	Schwieriger Fall	Schwieriger Fall
Distanz	Klein	Groß	Typische	Groß	Mittel
Dauer	Lang	Kurz	Typische	Mittel	Mittel
Gemein. Gehen	Signifikant	Nicht Sign.	Nicht Sign.	Nicht Sign.	Nicht Sign.
Wiederh. Näherung	Wahr	Falsch	Falsch	Wahr	Wahr
Durchschn. Distanz Verr.	Wahr	Falsch	Wahr	Wahr	Wahr
Erwarteter Wahrheitsgrad	Sehr hoch	Sehr niedrig	Mittel	Mittel	Hoch
Ergebnis linear	88,60	11,34	49,99	49,99	61,78
Ergebnis nichtlinear	88,61	14,89	51,43	50,00	61,57
Ergebnis-kongruenz	Gegeben	Gegeben	Gegeben	Gegeben	Gegeben

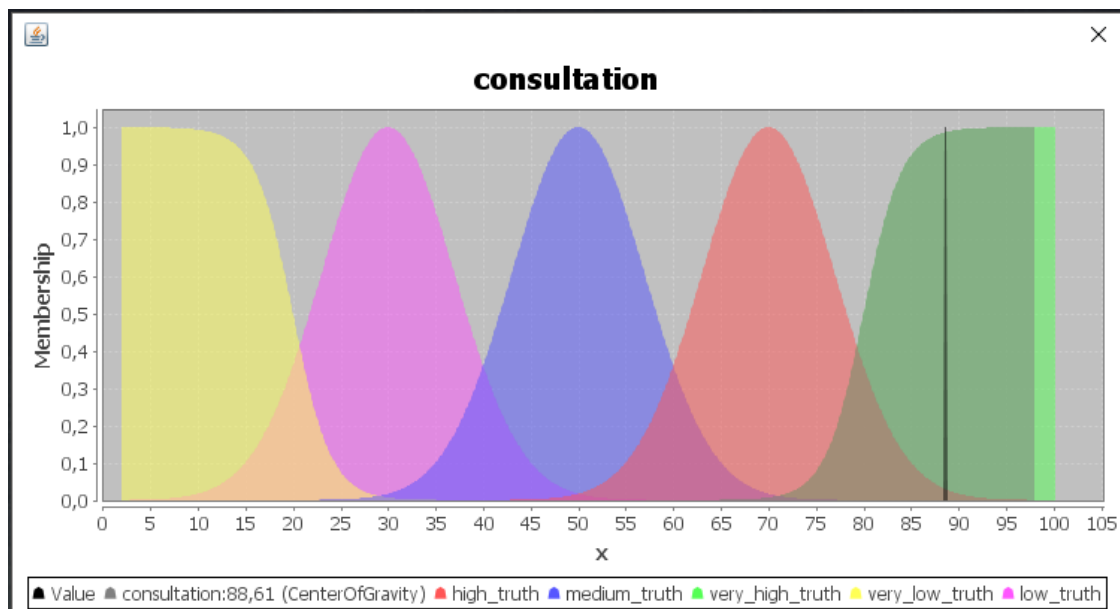
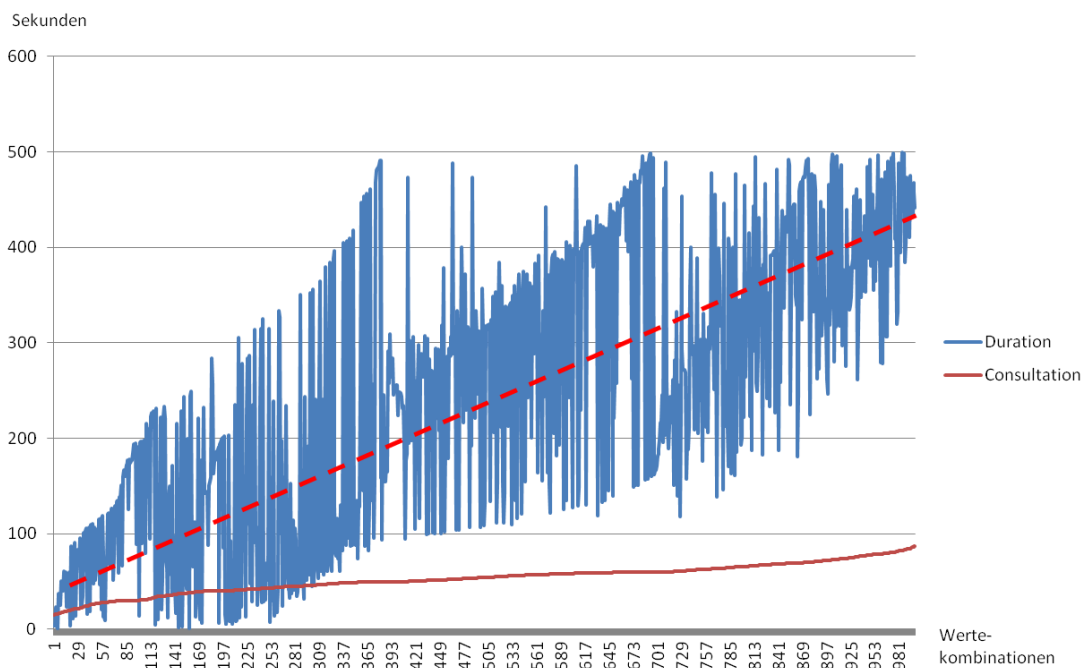


Abbildung 37: Graphische Darstellung des Fuzzy Logik Ergebnisses mit nichtlinearen Zugehörigkeitsfunktionen für Testfall 1

Die Effektivität des Feedbacksystems konnte zumindest für diese typischen Szenarien im Systemtest sichergestellt werden. Dabei scheint der Einsatz von nichtlinearen Zugehörigkeitsfunktionen bei der Beratungskontakterkennung keine signifikanten Vorteile zu liefern. Aus diesem Grund kann im Rahmen des Gründungsprojektes mit den einfacheren linearen Zugehörigkeitsfunktionen weiter gearbeitet werden.

## 6.2 Randomisierte Testfälle

Es wurden jeweils eintausend zufällige Kombinationen von allen fünf Eingabewerten programmatisch erzeugt und zusammen mit den zugehörigen Ergebnissen der Beratungskontakterkennung gespeichert. Die resultierenden Daten wurden nach aufsteigendem Beratungs-Wahrheitsgrad der Ergebnisse sortiert. In Abbildung 38 wurde die zentrale Beziehung zwischen dem Ergebnis und dem Eingabewert Dauer (duration) graphisch dargestellt.



**Abbildung 38:** Zufällige Eingabewertkombinationen sortiert nach dem Wahrheitsgrad eines Beratungskontaktes

Erwartungsgemäß ist eine deutliche Trendlinie zu erkennen nach der eine längere Dauer mit höheren Wahrheitsgraden für Beratungskontakte korreliert. Es wird deutlich, dass die Streuung der Dauerwerte für sehr geringe und sehr hohe Beratungs-Wahrheitsgrade geringer ist als für mittlere Werte. Dieser Trend scheint konsistent mit der Realität der zu

lösenden Fragestellung wie sie in Abschnitt „3. Intelligente Beratungskontakterkennung“ erläutert wurde. Demnach liegt die Ungenauigkeit bei der Zuordnung des potenziellen Beratungskontakts hauptsächlich in den Übergangsbereichen zwischen den Extremen Fällen. Es lässt sich erahnen, dass eine Implementierung des Beratungskontakt-Funktionsblocks durch eine gegebene Funktion oder Gleichung äußerst schwierig gewesen wäre.

Ein ähnlicher grundsätzlicher Trend ist zwischen der Kunden/Service-Mitarbeiter Distanz und dem Beratungs-Wahrheitsgrad zu erkennen. Abbildung 39 stellt diese Beziehung graphisch dar. Es wird deutlich, dass die Streuung hier noch größer ist. Dieses Ergebnis ist sinnvoll, da bei sehr kurzer Dauer auch sehr geringe Distanzen nicht ausreichen um einen hohen Wahrheitsgrad für einen Beratungskontakt zu rechtfertigen.

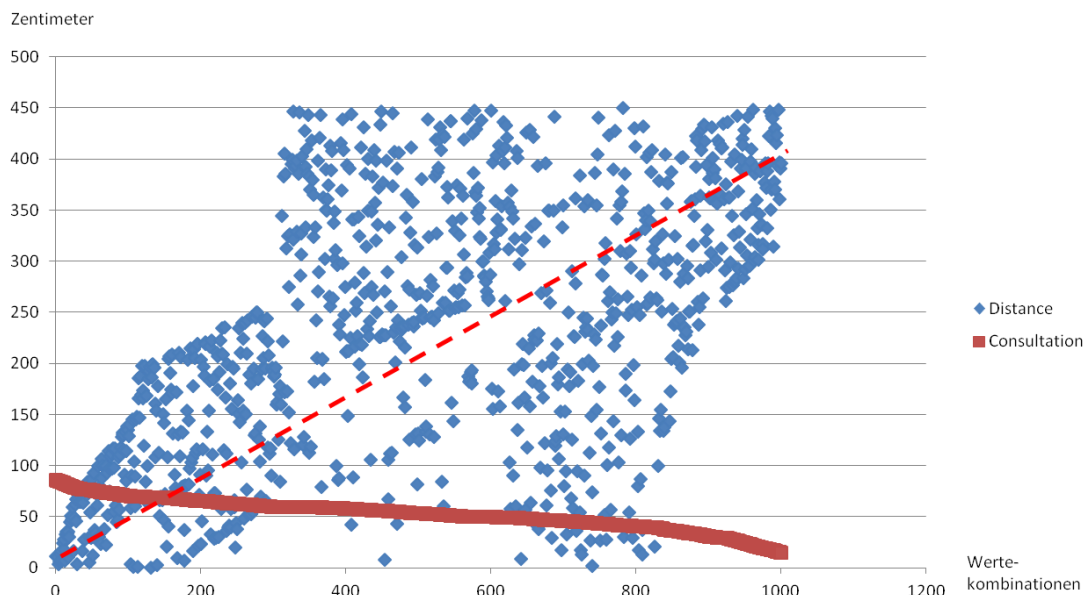


Abbildung 39: Streudiagramm der Distanz im Verhältnis zum Beratungs-Wahrheitsgrad für zufällige Eingabewertkombinationen

In den Tests waren die Ressourcenbelastung und die Performanz des Systems durchweg zufriedenstellend und auch bei einem hohen Anforderungsvolumen sind Ergebnisse in mehr als zufriedener Zeit verfügbar gewesen. Ein vergleichbar hohes Anforderungsvolumen ist im realen Einsatz der App in keiner Weise zu erwarten. Zudem kann im Rahmen des Gründungsprojektes von der hier verwendeten Vollversion von jFuzzyLogic auf die effizientere Mobilversion gewechselt werden um weitere Effizienzvorteile zu realisieren.

### 6.3 Ergebnisse und Fazit

Das Systemverhalten scheint auf dem gesamten Spektrum generell konsistent mit der Experten-Wissensbasis. Widersprüchliche Ausreißer-Ergebnisse sind im Rahmen der Tests nicht aufgetreten. Der allgemeine Proof of Concept ist damit bestanden. Das Feedbacksystem mit Fuzzy Logik liefert sinnvolle Ergebnisse die mit der bisherigen Wissensbasis kongruent sind. Im Anschluss an diese Arbeit gilt es nun den Prototypen unter realistischen Bedingungen auf der Verkaufsfläche zu testen um die Fuzzy Regeln zu optimieren und weiter auszubauen.

## 7 Zusammenfassung und Ausblick

In dieser Arbeit wurde, ausgehend von der Vision einer situationsrelevanten Feedbacklösung, eine intelligente Beratungskontakterkennung mit Fuzzy Logik realisiert. Der Proof of Concept für die grundsätzliche Eignung und Effektivität dieses Ansatzes wurde erreicht. Die hier erarbeiteten Ergebnisse bilden eine gute Grundlage für den zukünftigen Ausbau der Feedbacklösung im Rahmen des Gründungsprojektes Feedbadge am Startup Dock TUHH. Ein Umstieg auf andere Beacon Hardware ist im Zuge dieser Weiterentwicklung vorgesehen. Der modulare Aufbau des hier konzipierten Systems ermöglicht es eine solche Umstellung mit akzeptablem Aufwand zu bewältigen. Raum für Verbesserung besteht besonders beim Ausbau und der Optimierung der Wissensbasis und der Fuzzy Logik Regeln.<sup>39</sup> Durch Usability Tests in Zusammenarbeit mit den Pilotkunden kann der Prototyp in einem iterativen Prozess zu einem marktreifen Produkt weiter entwickelt werden.

---

<sup>39</sup> Feng Wan *et al.*: *How to determine the minimum number of fuzzy rules to achieve given accuracy: A computational geometric approach to SISO case*, Fuzzy Sets and Systems, März 2005

# 8 Anhang

## 8.1 Fuzzy Control Language

Fuzzy Control Language ist eine durch die IEC standardisierte Sprache zur Implementierung von Fuzzy Logik. Die leicht verständliche Syntax schreibt eine Gliederung in Funktionsblöcke vor in denen die notwendigen Abläufe für Fuzzy Logik (Variablen Definitionen, Fuzzification, Aggregation, Defuzzification und Fuzzy Regeln) sequenziell definiert werden können. Typische Methoden für die Definition von Zugehörigkeitsfunktionen und für die Defuzzifizierung können durch dafür vorgesehene Keywords der FCL Sprache einfach ausgewählt werden. Kernstück einer FCL Implementierung ist der Regelblock (RULEBLOCK) in dem der Entwickler die Fuzzy Logik Regeln nach dem folgenden Schema definieren kann:

```
RULE 0: IF (Temperature IS Cold) THEN (Output IS High)
```

Syntaktische Regeln für Klammern Setzung und Auswertungsreihenfolge von Logischen Ausdrücken entsprechen dem Standard vieler bekannter Programmiersprachen (z.B. Java). Eine Ausführliche Beschreibung der FCL Sprache steht im IEC 1131-7 CD1 Standard zur Verfügung.<sup>40</sup>

---

<sup>40</sup> IEC 61131-7, INTERNATIONAL ELECTROTECHNICAL COMMISSION (IEC) TECHNICAL COMMITTEE No. 65: INDUSTRIAL PROCESS MEASUREMENT AND CONTROL , IEC 1131 - PROGRAMMABLE CONTROLLERS Part 7 - Fuzzy Control Programming, 1997

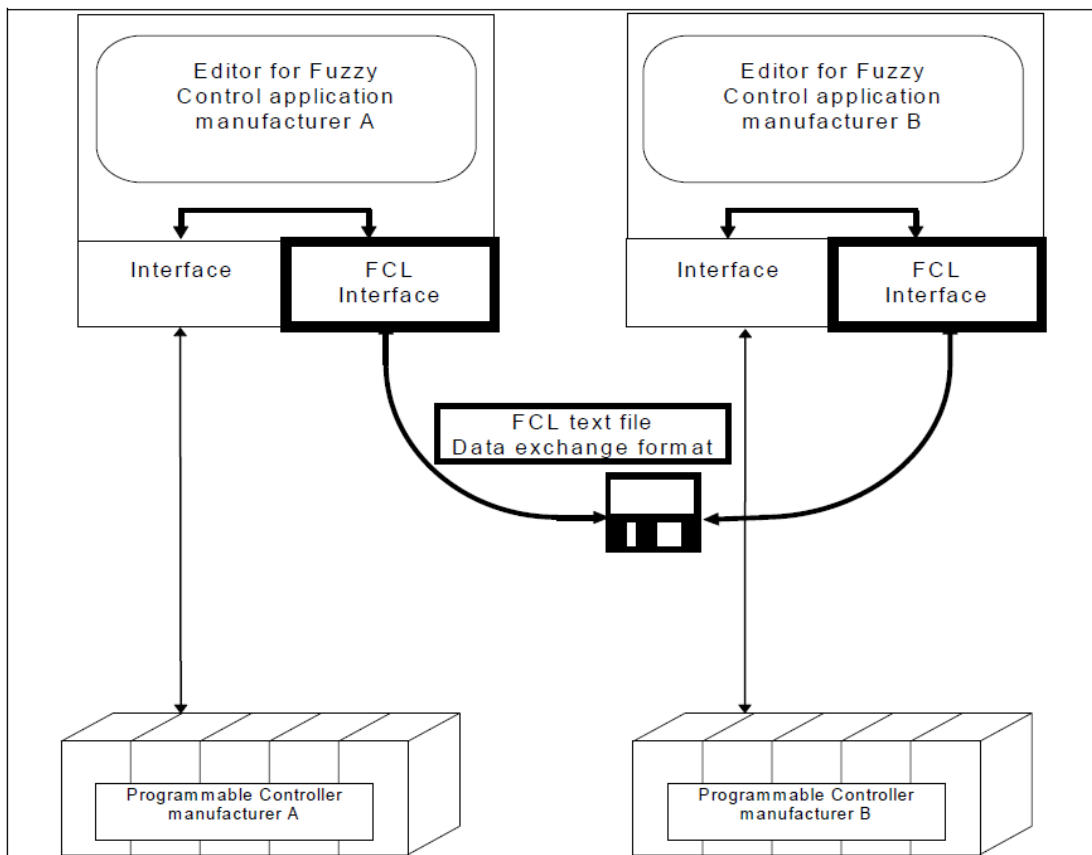


Abbildung 40: Datenaustausch zwischen Programmen mit Fuzzy Control Language <sup>41</sup>

## 8.2 Kurze Einführung in die jFuzzyLogic Bibliothek

Die aktuellste Version der jFuzzyLogic Bibliothek kann auf Sourceforge heruntergeladen werden.<sup>42</sup> Ein Plugin für das Eclipse IDE steht dort ebenfalls zur Verfügung.

Zunächst muss eine Instanz eines Fuzzy Inferenz Systems (FIS) im Sinne einer Fuzzy Logik Steuerung (siehe Abbildung 17 auf S.27) instanziiert werden. Der Konstruktor der FIS Klasse benötigt dafür den Namen einer FCL Datei im Root Ordner des Projekts. In der FCL Datei muss die konkrete Fuzzy Logik definiert werden.

Anschließend können programmatisch spezifische Funktionsblöcke aus der FCL Datei referenziert werden durch die vorimplementierte Methode *getFunctionBlock(String name)*;

<sup>41</sup> IEC 61131-7, INTERNATIONAL ELECTROTECHNICAL COMMISSION (IEC) TECHNICAL COMMITTEE No. 65: INDUSTRIAL PROCESS MEASUREMENT AND CONTROL, IEC 1131 - PROGRAMMABLE CONTROLLERS Part 7 - Fuzzy Control Programming, 1997

<sup>42</sup> jFuzzyLogic: <http://jfuzzylogic.sourceforge.net/html/index.html>, 2.6.2016



Mit der Referenz auf den Funktionsblock können Variablen programmatisch gesetzt werden und der Auswertungsprozess mit der vorimplementierten Methode *evaluate()*; ausgelöst werden. Scharfe Ergebnisse von Ausgabevariablen können durch die Methode *ausgabevariable.defuzzify()*; referenziert werden.

Mithilfe der Bibliotheksklasse *JFuzzyChart* können alle in FCL definierten Zugehörigkeitsfunktionen des Funktionsblocks graphisch dargestellt werden. Das illustrative Beispielprogramm *Short Introduction to jFuzzyLogic.7z* ist auf der dieser Arbeit beigelegten CD enthalten.

# Quellenverzeichnis

- Alan Dennis / Barbara Haley Wixom / David Tegarden: *System Analysis & Design. An Object Oriented Approach with UML (5<sup>th</sup> edition)*, Wiley, 2015
- Carlos E. Otero: *Software Engineering Design. Theory and Practice*, CRC Press, 2012
- Patrick Grässle/ Henriette Baumann / Philippe Baumann: *UML 2 projektorientiert*, Galileo Computing, 2007
- Stamatios V. Kartalopoulos: *Understanding Neural Networks and Fuzzy Logic*, IEEE Press, 1996
- John Yen / Reza Langari / Lotfi A. Zadeh: *Industrial Applications of Fuzzy Logic and Intelligent Systems*, IEEE Press, 1995
- Paul P. Wang / Da Ruan / Etienne E. Kerre (Eds.): *Fuzzy Logik. A Spectrum of Theoretical and Practical Issues*, Springer, 2007
- Hans-Jürgen Zimmermann: *Neuro + Fuzzy. Technologie – Anwendungen*, VDI Verlag, 1995
- George J. Klir / Bo Yuan: *Fuzzy Sets and Fuzzy Logik. Theory and Applications*, Prentice Hall PTR, 1995
- Thomas Künneth: *Android 5. Apps entwickeln mit Android Studio*, Rheinwerk Computing, 2015
- Eric Ries: *Lean Startup. How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*, Portfolio Penguin, 2011
- Pawel Gorecki / Peter R. Pautsch: *Lean Management*, Hanser, 2015
- INTERNATIONAL ELECTROTECHNICAL COMMISSION (IEC): *IEC 1131 - PROGRAMMABLE CONTROLLERS. Part 7 - Fuzzy Control Programming*, Committee Draft CD 1.0, 1997
- Pablo Cingolani / Jesús Alcalá-Fdez: *jFuzzyLogic: a Java Library to Design Fuzzy Logic Controllers According to the Standard for Fuzzy Control Programming*, International Journal of Computational Intelligence Systems, 2013
- Cingolani, Pablo, and Jesus Alcala-Fdez: *jFuzzyLogic: a robust and flexible Fuzzy-Logic inference system language implementation*, Fuzzy Systems (FUZZ-IEEE), IEEE International Conference, 2012
- Lotfi A. Zadeh: *Toward a perception-based theory of probabilistic reasoning with imprecise probabilities*, special issue on imprecise probabilities, Journal of Statistical Planning and Inference, Vol. 105, S. 233-264, 2002
- Petr Hajek, Lluís Godó, Francesc Esteva: *Fuzzy Logic and Probability*, Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, 2013
- HyperLogic Technical Notes: *Fuzzy Logic Inference Techniques in CubiCalc Version 2. Number 1 Revision 5*

P. Kannan, S. Deepa, R. Ramakrishnan (2012): *Contrast Enhancement of Sports Images Using Two Comparative Approaches*, American Journal of Intelligent Systems

Feng Wan et al.: *How to determine the minimum number of fuzzy rules to achieve given accuracy: A computational geometric approach to SISO case*, Fuzzy Sets and Systems, März 2005

Timothy Ross: *Fuzzy Logic With Engineering Applications (Third Edition)*, Wiley, 2010

Pablo Cingolani: *Documentation & a brief introduction to jFuzzylogic*, <http://jfuzzylogic.sourceforge.net/html/manual.html>, 22.5.2016

Gimbal Documentation, <https://docs.gimbal.com/>, 25.5.2016

Android SDK Documentation, <http://developer.android.com/guide/index.html>, 29.5.2016

## Versicherung über Selbstständigkeit

*Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.*

Hamburg, den \_\_\_\_\_