

TECHNISCHE UNIVERSITÄT DRESDEN

PROFESSUR FÜR MENSCH-COMPUTER-INTERAKTION

Masterarbeit

zur Erlangung des akademischen Grades
Master of Science

Aktivitätserkennung in Smarthome-Umgebungen

Thomas Hauptvogel
(Geboren am 8. August 1992 in Herzberg/Elster)

Hochschullehrer: Prof. Dr. rer. nat. habil. Gerhard Weber
Betreuer: David Gollasch, M.Sc.

Dresden, 11. September 2020

Kurzfassung

Wir leben in einer immer älter werdenden Gesellschaft, mit einem stetig steigenden Bedarf an Pflegekräften, welcher nicht mehr gedeckt werden kann. Eine Lösung für dieses Problem sind Assistenzsysteme, vor allem Roboter, welche Pflegende bei ihrer Arbeit unterstützen können. Allerdings sind Roboter in ihrem Wissen und ihrer Handlungsfähigkeit durch ihre Hardware begrenzt, weshalb eine modulare Erweiterung mittels Smarthome-Technologie eine mögliche Lösung für dieses Problem sein könnte. Viele Menschen besitzen heutzutage schon privat Sensoren und diese sind mittlerweile leicht erhältlich und günstig. In dieser Arbeit wurde für diesen Zweck ein System konzipiert, welches mittels im Handel erhältlicher Sensorik in privaten Wohnungen die Aktivitäten des Bewohners erkennen kann. Andere Assistenztechnologien können mit dem hier entwickelten System über REST-Schnittstellen kommunizieren und ihr Wissen über den Haushalt und dessen Bewohner erweitern. Eine Klassifizierung erkannter Sensoraktivitäten erfolgt sowohl über Machine Learning, als auch über vorher definierte Regeln, welche anhand der Standorte der Sensorik die Aktivitäten bestimmen. Über eine grafische Benutzeroberfläche kann jeder im lokalen Netz auf das System zugreifen und seinen Tagesablauf einsehen, sowie selbst Aktivitäten eintragen, aus denen das System für die zukünftige Klassifizierung lernen kann. Für das Konzept wurden die Bedürfnisse der Nutzergruppen analysiert und in Anforderungen umgeschrieben. Aus dem konzeptionierten System wurde ein funktionsfähiger Prototyp erstellt, welcher mit Probanden aus verschiedenen Nutzergruppen getestet wurde. Die Nutzer konnten fast alle problemlos alle Funktionen des Systems nutzen, fanden es sehr transparent und hatten auch Ideen für Erweiterungen. Für eine tatsächliche private Nutzung kam es hauptsächlich bei den älteren Probanden in Frage, was auch die primäre Zielgruppe der Arbeit war. Am Ende dieser Arbeit gab es einen gebrauchstauglichen Prototypen, welcher mit einer guten Genauigkeit Aktivitäten des täglichen Lebens bestimmte und ein großes Potenzial an Erweiterungsmöglichkeiten besaß.

Abstract

We live in a constantly aging society with a rising need for careworkers, which cannot be satisfied anymore. Assistance systems are a solution for this problem, especially robots, which can support careworkers with their work. Though robots are limited in their knowledge and ability to act by their hardware, which makes modular extensions via smart home technology a possible solution for this problem. A lot of people own sensors privately and they are relatively cheap and easy to get. In this work a system has been designed, which can detect activities of residents in their living environments by using sensors available on the market. Other assistance technology can communicate with the system and expand their knowledge about the household and its resident. A classification of recognized sensor activities is done by machine learning as well as by pre-defined rules, which classify the activities by the location of the sensors. With the help of a graphical user interface everyone in the local network can access the system and see their daily routine and insert activities by their own, which are used by the system to train the machine learning model. Before designing the concept the needs of the user groups were analyzed and written into requirements. An useable prototype has been devel-

oped by using this concept. The prototype was tested by users from different groups, which could handle nearly all functionalities of the system, thought it was very transparent and had some ideas for extensions. Mainly the older users, who are also the target audience, could imagine using the system in their homes. At the end of this work there was an usable prototype which could classify activities of daily living with good accuracy and had a lot of potencial for more extensions.

Inhaltsverzeichnis

1 Grundlagen zu Ambient Assisted Living, Aktivitäten des täglichen Lebens und Assistenzrobotern	7
1.1 Theoretische Grundlagen zu Assistenzsystemen und Aktivitäten	7
1.1.1 Ambient Assisted Living	7
1.1.2 Aktivitäten des täglichen Lebens	9
1.2 Smarthome-Sensorik, Verwaltung mit openHAB und Loomo	10
1.2.1 Sensoren und Aktoren	10
1.2.2 openHAB-System	11
1.2.3 Loomo-Roboter	13
1.3 Zusammenfassung	14
2 Beschreibung verwandter Arbeiten aus dem Bereich der Aktivitätserkennung	15
2.1 Erkennung von Aktivitäten in Zusammenspiel mit einem Assistenzroboter	15
2.2 Aktivitätsbestimmung durch Nutzung von tragbarer Sensorik und Standortdaten .	16
2.3 Bestimmung von Aktivitäten mit Hilfe weniger stationärer Sensoren	17
2.4 Zusammenfassung	18
3 Analyse der Anforderungen an ein System zur Erkennung von Aktivitäten	19
3.1 Bedürfnisse der primären Nutzergruppe	19
3.2 Bedürfnisse weiterer Nutzergruppen	20
3.3 Anwendungsszenario	21
3.4 Definition der Anforderungen	22
3.5 Zusammenfassung	24
4 Konzept eines adaptiven Smarthome-Systems zur Erkennung von Tagesabläufen	25
4.1 Entwurf einer grafischen Benutzungsschnittstelle	25
4.2 Technisches Konzept	25
4.2.1 Persistierung der Daten	25
4.2.1.1 Technische Schemata von Aktivitäten und Routinen	27
4.2.1.2 Technisches Schema eines Nutzerprofils	28
4.2.2 Klassifizierung der Aktivitäten	28
4.2.2.1 Machine Learning-Klassifizierung	29
4.2.2.2 Aussagenlogische Klassifizierung	32
4.2.2.3 Evaluation der Ansätze und Wahl des Algorithmus	36
4.2.3 Adaptive und adaptierbare Mechanismen des Systems	36
4.3 Systemarchitektur Design	37
4.4 Zusammenfassung	39
5 Prototypische Umsetzung des Systems	41
5.1 Verwendete Hardware	41
5.2 Installation und Konfiguration des Smarthome-Hubs	42
5.3 Klassifizierungsalgorithmus	45
5.4 Visuelle Ausgabe	46
5.5 Zusammenfassung	47

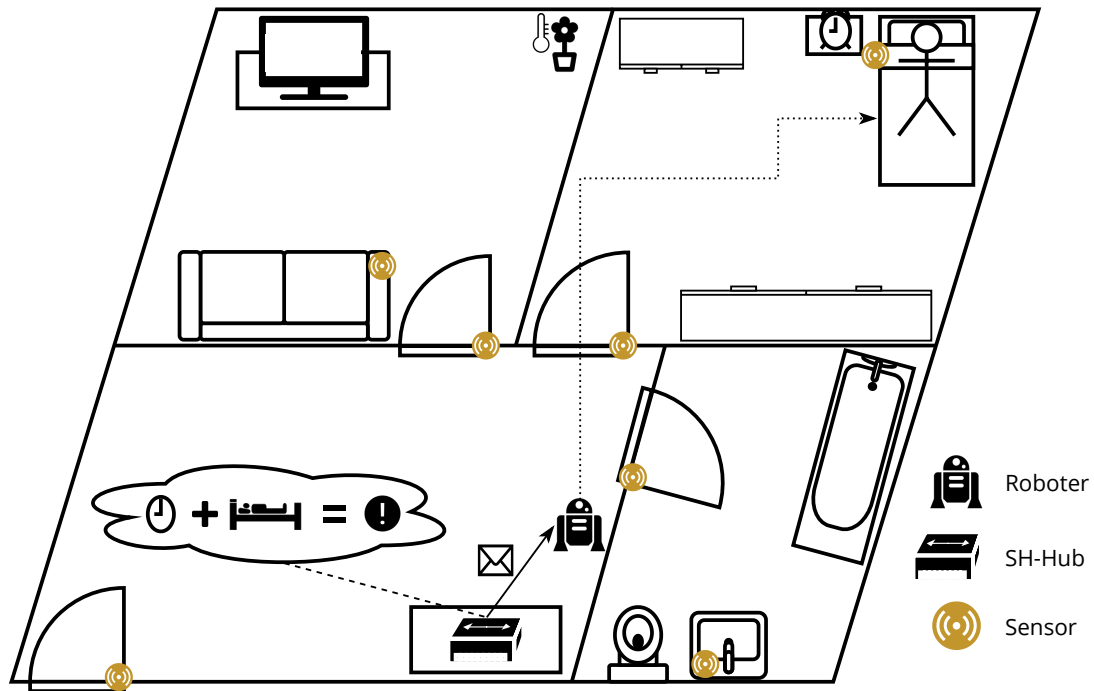
6 Evaluation des prototypischen Systems	49
6.1 Aufbau und Ablaufplan des Versuchs	49
6.2 Versuchsdurchführung	51
6.3 Interpretation der Ergebnisse	54
6.4 Zusammenfassung	59
7 Fazit	61
7.1 Zusammenfassung	61
7.2 Diskussion	63
7.3 Beantwortung der Forschungsfrage	65
7.4 Ausblick	66
Literatur	67
Abkürzungsverzeichnis	75
Programmcode-Verzeichnis	77
Abbildungsverzeichnis	79
A Anhang	81

Bedarf nach einer Aktivitätserkennung zur Erweiterung von Assistenzlösungen im Smarthome-Umfeld

Die Bevölkerung in modernen Gesellschaften wird immer älter und pflegebedürftiger [UDP19], aber zeitgleich gibt es nicht genügend Pflegepersonal, welches dadurch stark überlastet ist. Außerdem wollen viele pflegebedürftige Menschen nicht ihr gewohntes Umfeld verlassen, sodass ein reibungsloser Tagesablauf im heimischen Umfeld ermöglicht werden muss. Eine Erleichterung im Haushalt können Smarthome (SH)-Technologien sein, da diese einerseits tägliche Routinen, wie beispielsweise heizen, lüften, usw. übernehmen und andererseits auch den Haushalt überwachen können. Ein Nachteil der Sensoren und Aktoren (siehe Abschnitt 1.2.1) ist die fehlende bzw. unzugängliche Interaktion mit den betroffenen Personen und das viele unterschiedliche Protokolle zur Ansteuerung existieren, welche oft eine jeweils eigene Schnittstelle bzw. Plattform benötigen [Ngu+17]. Mögliche Erweiterungen können durch die Anbindung eines Sprachassistenten, wie Alexa oder Mycroft, umgesetzt werden. Mit Hilfe natürlicher Sprache lassen sich angebundene Geräte ohne weiteres technisches Hintergrundwissen ansteuern. Während viele SH-Geräte nach der Ersteinrichtung nur über einen Hub¹ gesteuert werden und weitestgehend nebenher ihren Aufgaben nachkommen, kann bei einem Sprachassistenten die zu pflegende Person direkt sagen, dass sie zum Beispiel eine Erhöhung der Raumtemperatur wünscht. Trotzdem bleiben Sprachassistenten starre Objekte denen man Befehle erteilt und welche diese je nach Möglichkeit erfüllen. Menschliches Pflegepersonal bietet eine größere Vielseitigkeit und vor allem auch eine physische Anwesenheit bei den Patienten, welche man durch Roboter erreichen könnte.

Einige für den Verbraucher erschwingliche Roboter sind bereits in vielen Haushalten im Alltag integriert und helfen beispielsweise beim Staubsaugen oder Rasen mähen. Diese sind allerdings auf eine Aufgabe beschränkt, wie meistens auch die einzelnen SH-Geräte. Damit nicht für jede Aufgabe ein spezieller Roboter benötigt wird, wurde bereits an speziellen Pflegerobotern gearbeitet, welche mehrere Aufgaben erledigen können. Ein aktueller Vertreter davon ist der Care-O-bot 4 [Kit+15], welcher von einem modularen Aufbau profitiert, eine relativ kostengünstige Möglichkeit den Roboter an seine Aufgaben anzupassen. Dies ist das größte Problem der Pflegeroboter: sie sind zu teuer und die Hardware ist nicht flexibel genug. Große Institutionen können sich solch eine Maschine vielleicht leisten, doch dem Großteil der privaten Interessenten bleibt dies verwehrt. SH-Geräte sind wiederum schon auf dem Markt etabliert und können problemlos überall erworben werden. Daher sollte ein Roboter, der mit SH-Geräten Informationen austauschen und interagieren kann, die meisten Probleme ohne teure individuelle Hardware-Anpassungen lösen können. Damit sollten sich die Einsatzmöglichkeiten im häuslichen Umfeld, aber auch in Pflegeheimen, erweitern lassen, ohne für jede Aufgabe einen eigenen Roboter einsetzen zu müssen. Natürlich lassen sich durch Sprachassistenten und SH-Geräte schon viele Anwendungsfälle zwecks einfacher Kommunikation und vielfältiger Aufgabenbewältigung abdecken, aber Roboter bieten im Grunde genommen einen mobilen Sprachassistenten mit greifbarem Körper. Ebenso können Sensoren auch tote Winkel haben, welche durch einen Roboter ausgeglichen werden können, indem er sich beispielsweise um eine gestürzte Person bewegt, um die Situation aus einem anderen Blickwinkel zu betrachten. Roboter bieten also den Vorteil einer physischen Präsenz und könnten pflegebedürftige Personen rund um die Uhr begleiten. Leider sind sie immer noch viel zu teuer, da oft auch das Problem des 'Over-Engineering' auftritt, also dass die Entwickler zu viele komplexe Funktio-

¹Zentrale Verwaltungsstelle für Hausautomatisierung



Voll vernetzte SH-Umgebung, welche einen Roboter bei ungewöhnlichen Aktivitäten benachrichtigt

nen in einer Maschine vereinen wollen, wodurch die Hardware teurer und fehleranfälliger wird [WPW15]. Durch die Auslagerung an SH-Geräte würden viele Aufgaben weniger komplex und der Roboter bräuchte weniger teure individuelle Hardware, da es schon ausreichend Geräte auf dem Markt gibt. Ein Roboter könnte eine Person nachts zur Toilette begleiten und Türsensoren und Beleuchtung auf dem Weg aktivieren, um die Person sicher zum Bad zu führen. Das wäre wesentlich unkomplizierter als eine Realisierung mit Hilfe von Roboterarmen. Die Kombination von Kompanion-Roboter mit SH-Geräten ist ein interessantes Themengebiet, welches ermöglichen könnte, dass pflegebedürftige Personen sicher und problemlos durch den Alltag begleitet werden. Dafür muss der Roboter allerdings die täglichen Routinen der Person kennen, um dort proaktiv zu unterstützen oder bei starken Abweichungen zu kontrollieren, ob alles in Ordnung ist. SmartMeter können bereits Routinen bzw. Tätigkeiten erkennen [Cha+], allerdings wird dabei auch der gesamte Haushalt per Telepräsenz-System überwacht und es müssen immer noch Menschen eingreifen, wenn etwas nicht passt. Deshalb wäre es vorteilhaft, mit beliebigen SH-Geräten zur Laufzeit Aktivitäten des täglichen Lebens erkennen zu können, diese zu klassifizieren und lokal in einem Profil des Nutzers zu speichern.

Zielsetzung der Arbeit

Diese Arbeit soll die Frage klären, ob sich mit SH-Hardware aus dem Konsumentenbereich ein von anderen Assistenzsystemen nutzbarer Tagesablauf eines Nutzers erstellen lässt, welcher sich selbstständig an verändernde Verhaltensweisen des Nutzers anpasst.

Am Ende der Arbeit soll ein Algorithmus entstanden sein, der auf einem SH-Hub mit einem lokalen Netzwerk verbundene Sensoren auswertet und aus deren Daten tägliche Aktivitäten und Routinen des Nutzers erkennt. Zur Klassifizierung der Aktivitäten sollen Machine Learning und aussagenlogische Klassifizierung verglichen werden, um eine gute Erkennung ohne bestehenden Datensatz zu ermöglichen. Erkannte Aktivitäten sollen in einem Nutzerprofil festgehalten werden, wobei sich die Erkennung von Aktivitäten im Laufe der Zeit auch adaptiv anpassen kön-

nen soll, wenn genug individuelle Nutzerdaten zur Verfügung gestellt wurden. Das Profil mit Tagesablauf soll so gestaltet sein, dass es als hilfreiche Ergänzung von weiteren Assistenzsysteme genutzt werden kann. Ebenso sollen die Möglichkeiten aufgezeigt werden, die ein solches Profil in Zusammenarbeit mit Assistenzrobotern und Sprachassistenten bieten würde. Diese Aufgabe umfasst folgende Teilziele:

- Erstellung eines Modells zur Identifikation von Aktivitäten des täglichen Lebens mit Hilfe von SH-Geräten
- Erstellung eines technischen Konzepts zur Erkennung der Aktivität aus den Sensordaten
- Konzept einer Datenpersistierungsstrategie für die Zuordnung von Aktivitäten zu Sensordaten
- Implementierung eines Prototyps mit Hilfe von openHAB und Loomo
- Evaluation: Test des Prototyps in einer echten Umgebung mit anschließender Auswertung der Ergebnisse

Struktur der Arbeit

Im folgenden Kapitel 1 werden Begrifflichkeiten rund um das Thema Ambient Assisted Living (AAL) und SH erläutert, welche für den Kontext dieser Arbeit relevant sind. In Kapitel 2 werden weitere Ansätze zur Aktivitätserkennung analysiert und deren Bezug auf diese Arbeit gezeigt. In Kapitel 3 werden die Bedürfnisse der potenziellen Nutzer des Systems untersucht und die daraus resultierenden Anforderungen definiert, damit daraus in Kapitel 4 das technische Konzept des Systems erstellt werden kann. In dem darauf folgenden Kapitel 5 soll die Entwicklung eines Prototyps dokumentiert werden, welcher für Tests mit Versuchspersonen genutzt wird. Diese Tests werden in Kapitel 6 beschrieben und anhand von durch die Versuchspersonen ausgefüllten Fragebögen dort ausgewertet. Die Arbeit wird abgerundet durch eine Diskussion des Ergebnisses und einem Ausblick auf weiterführende Arbeiten.

1 Grundlagen zu Ambient Assisted Living, Aktivitäten des täglichen Lebens und Assistenzrobotern

Für die Erkennung von Aktivitäten muss zunächst geklärt werden, wie diese zu definieren sind. Ebenso braucht es einen Kontext für SH, welcher in dieser Arbeit durch AAL repräsentiert sein soll. Diese theoretischen Grundlagen und die Technologien, auf denen diese Arbeit basiert, werden im folgenden Abschnitt weiter erläutert.

1.1 Theoretische Grundlagen zu Assistenzsystemen und Aktivitäten

Im Folgenden werden zuerst Hintergründe des Ambient Assisted Living erläutert, da sich der SH-Kontext dieser Arbeit in diesem Bereich befinden soll. Anschließend werden noch Aktivitäten des täglichen Lebens (ADL)¹ ausführlicher betrachtet, um eine genaue Definition dieses Begriffs zu erlangen.

1.1.1 Ambient Assisted Living

Das Ambient Assisted Living ist eine spezielle Form von SH, da es sich hauptsächlich auf ältere und auch körperlich beeinträchtigte Nutzer fokussiert. Insgesamt gibt es vier betroffene Nutzergruppen, nämlich Patienten, Ärzte, Betreuende und Verwandte [Cal+17]. Von SH spricht man bei einem gut vernetzten Haushalt, in dem verschiedenste Geräte und Sensoren, wie Leuchtmittel, Schalter, Thermostate, Unterhaltungsgeräte und viele Andere miteinander verknüpft sind. Dieser Haushalt wird über einen Hub gesteuert, welcher die Aktionen der Geräte orchestrieren kann oder auf bestimmte Ereignisse reagiert, wie beispielsweise die Jalousien verdunkeln sobald eine bestimmte Temperatur erreicht wurde. Beim AAL sind es Haushalte von Senioren und hilfsbedürftigen Menschen, die automatisiert werden, um die Lebensqualität dieser Personengruppen und ihrer Pfleger zu verbessern [Pol05]. Es gibt zwar keine konkrete Definition des AAL, aber das AAL-Programm der Europäische Union (EU) gibt eine bestimmte Richtung vor, was damit erreicht werden soll [AAL18]:

- Die Zeit, in der es Menschen möglich ist, zuhause zu leben, soll verlängert werden, indem ihre Autonomie, ihr Selbstbewusstsein und ihre Mobilität verbessert wird.
- Die Erhaltung von Gesundheit und funktionalen Möglichkeiten älterer Menschen soll unterstützt werden.
- Bessere und gesündere Lebensweisen sollen Menschen in Risikogruppen näher gebracht werden.
- Erhöhung der Sicherheit, Verhinderung sozialer Isolation und die Erschaffung eines Unterstützungsnetzwerks für ältere Personen.
- Unterstützung für Pfleger, Familien und Pflegeorganisationen ermöglichen.
- Die immer älter werdende Gesellschaft soll durch mehr Selbstbestimmtheit unterstützt werden.

¹Aus dem Englischen: **A**ctivities of **D**aily **L**iving

Definition: Die Aufgabe des Ambient Assisted Living ist die Automatisierung von Haushalten älterer und pflegebedürftiger Menschen, mit dem Ziel der Verbesserung von Lebensqualität, Selbstständigkeit und Sicherheit dieser Personengruppen.

Um diese ganzen Ziele zu erreichen, werden entsprechende technische Geräte benötigt. Assistenztechnologien können dabei Low-Tech Technologien sein, also Brillen oder Rollatoren, aber auch High-Tech, wie moderne Hardware und Software, welche in dieser Arbeit hauptsächlich betrachtet werden [Bla+16].

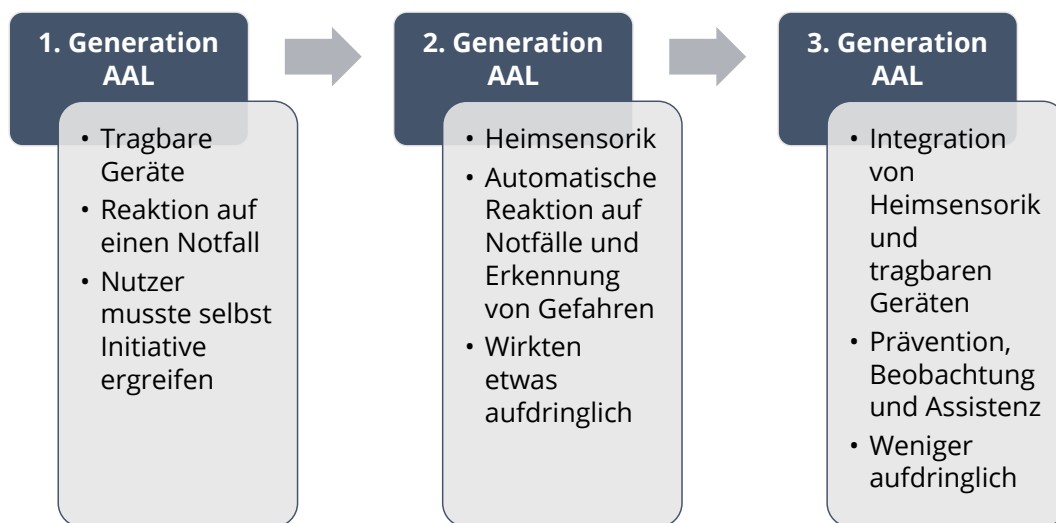


Abbildung 1.1: Entwicklung der AAL Technologien im Laufe der Zeit [Bla+16]

Abbildung 1.1 zeigt, wie sich die verwendete Technologie im Assistenzbereich weiterentwickelt hat. Musste früher die betroffene Person noch selbst Initiative ergreifen und einen Notfall melden, gibt es mittlerweile Systeme, welche Notfälle schon selbst erkennen und diese auch verhindern können. Die Geräte wurden mit der Zeit auch immer unauffälliger, da sich die meisten Menschen mit Kameras und Mikrofonen in ihrem Haushalt nicht wohl fühlen. Außerdem wollen die betroffenen Personen häufig nicht, dass man Assistenzsysteme in ihrem Umfeld direkt als solche erkennt.

Der Großteil der Arbeiten zum Thema AAL decken einen oder mehr von acht Themenbereichen ab [Cal+17]:

- **Aktivitätserkennung:** Die Identifizierung von ADLs
- **Vitalitätskontrolle:** Die Remote Überwachung von Vitalwerten des Nutzers
- **Positionsverfolgung:** Indoor und Outdoor Lokalisierung des Nutzers
- **Interaktion:** Die Interaktion des Nutzers mit Assistenztechnologien
- **Multimedia Analyse:** Die Auswertung von gesammelten Multimedia Daten
- **Datenanalyse:** Datenauswertung bezüglich der Entdeckung von Beziehungen, Eigenschaften und Wissen innerhalb verschiedener Datensätze

- **Data Sharing:** Fokus auf das Teilen von Informationen und Wissen zwischen verschiedenen AAL-Interessengruppen und Nutzern
- **Kommunikation:** Die Vereinfachung der Zusammenarbeit zwischen verschiedenen Nutzern

Für eine bessere Übersichtlichkeit lassen sich die Kategorien in Indoor und Outdoor Assistenz unterteilen [Neh+06]. Der Fokus dieser Arbeit liegt auf Indoor Assistenz und umfasst dabei mehrere der acht genannten Kategorien, wie die Aktivitätserkennung und die Positionsverfolgung.

1.1.2 Aktivitäten des täglichen Lebens

Ein weiteres wichtiges Themengebiet in dieser Arbeit sind ADLs. Damit werden Aktivitäten bezeichnet, welche täglich von Menschen zur Selbsterhaltung ausgeführt werden. Personen mit Pflegebedarf können bestimmte ADLs nicht mehr ohne Hilfe ausführen und benötigen deshalb Assistenztechnologie oder menschliches Pflegepersonal. ADLs können bei jedem Menschen unterschiedlich ausfallen, trotzdem gibt es einige elementare Dinge, die von den meisten Menschen täglich durchgeführt werden [Wil+14]: eigenständig essen; sich duschen, baden oder anderweitig komplett säubern; die Haare pflegen, Rasieren, Zähne putzen, ...; selbst auf Toilette gehen, sich alleine säubern und wieder gehen; sich eigenständig einkleiden und sich unabhängig von Anderen fortbewegen können. Eine Unterstützung bei diesen Tätigkeiten durch Assistenzsysteme ist sehr speziell und benötigt dafür zugeschnittene Hardware. Außerdem ist es äußerst mühselig, ohne ständige Betreuung zu leben, wenn diese Aktivitäten nicht mehr selbstständig durchgeführt werden können. Die besonders für diese Arbeit interessanten Aktivitäten sind die *instrumentellen ADLs*. Diese sind nicht entscheidend für das menschliche Überleben, aber sie sind bedeutend für das eigenverantwortliche Leben im eigenen Zuhause [Boo+08; SML10]. Diese umfassen unter anderem Aktivitäten, wie sich um den Haushalt kümmern, Mahlzeiten zubereiten, einkaufen, soziale Kontakte pflegen und viele Weitere. Eine Zusammenfassung dieser ganzen Aktivitätsklassen ist in Abbildung 1.2 zu sehen.

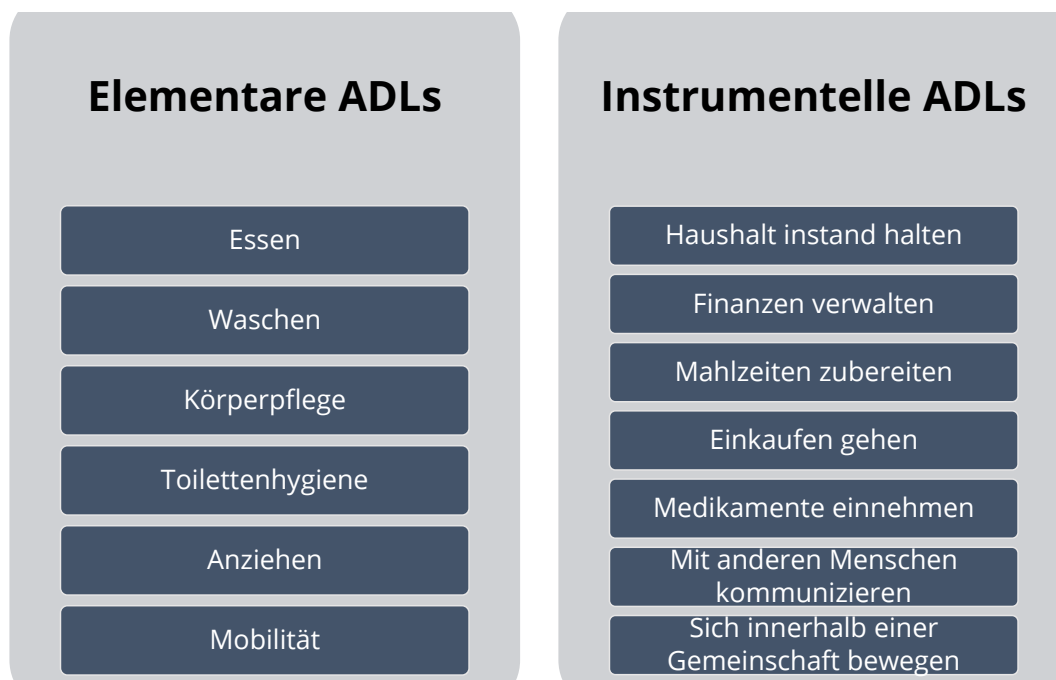


Abbildung 1.2: Kategorisierung von Aktivitäten des täglichen Lebens

Definition: Aktivitäten des täglichen Lebens umfassen einerseits elementare Tätigkeiten, welche zur Selbsterhaltung dienen und andererseits instrumentelle Aktivitäten, welche zum selbstbestimmten Leben innerhalb einer Gemeinschaft dienen.

Eine Granularisierung dieser Aktivitäten ist möglich, beispielsweise kann "Mahlzeiten zubereiten" in Frühstück, Mittagessen, Abendessen oder Snack zubereiten unterteilt werden. Diese Unterteilung könnte man wiederum noch weiter separieren und einzelne Schritte, wie beispielsweise den Ofen vorheizen, Teller aus dem Schrank holen, den Tisch decken, usw. als Aktivitäten auswählen. Würde man dies mit jeder dieser Kategorien machen, dann gäbe es schnell eine riesige Anzahl an Aktivitäten, welche sich teilweise stark ähneln und damit schwer unterscheidbar wären. Dieser Auszug an Kategorien von Aktivitäten soll daher alle granularen Aktivitäten zusammenfassen, um eine bessere Klassifizierung zu ermöglichen. Aktivitäten aus beiden Gruppen können erkannt werden, wobei sich hier auf Indoor Aktivitäten beschränkt wird, sodass vorerst nur untersucht wird, ob der Nutzer das Haus verlässt, aber nicht was außerhalb der eigenen Räumlichkeiten passiert.

1.2 Smarthome-Sensorik, Verwaltung mit openHAB und Loomo

Ein wichtiger Bestandteil von AAL und der Erkennung von Aktivitäten, sind die Sensoren inklusive deren Steuerung und Verwaltung. Deshalb wird der folgende Abschnitt einige Grundlagen zu Sensoren und die in dieser Arbeit verwendeten Typen von Sensoren näher erläutern. Außerdem wird das openHAB (OH)-System genauer betrachtet, welches als zentrales Verwaltungssystem für die Sensor- und Nutzerdaten agiert. Der Loomo Roboter von Segway wird weiterhin detailliert aufgeführt, da das Nutzerprofil, welches im Laufe der Arbeit entstehen soll, potenziell von einem Assistenzroboter zur Informationsbeschaffung genutzt wird.

1.2.1 Sensoren und Aktoren

Sensoren sind technische Geräte, welche bestimmte Eigenschaften ihrer Umgebung erfassen können. Aktoren wiederum senden Signale um eine Aktion anzustoßen, wie beispielsweise das Erhöhen der Raumtemperatur durch Veränderung des Thermostats. Die meisten Geräte in einer SH-Umgebung gehören einer dieser beiden Gruppen an und deren Zusammenspiel erzeugt meistens beträchtliche Synergien. Als Beispiel könnte man einen CO₂-Sensor und einen smarten Fenster-Antrieb nehmen, welcher immer das Fenster öffnet, sobald der CO₂-Gehalt in der Luft zu hoch ist und somit immer für frische Luft im Haushalt sorgt.

Zur Kommunikation nutzen die Geräte verschiedene Protokolle, wobei fast alle modernen Protokolle kabellos funktionieren. Wenn man sich die meisten aktuellen Geräte ansieht, dann gibt es neben Bluetooth und W-LAN noch zwei weitere Protokolle, die häufig Erwähnung finden, Z-Wave und ZigBee.

Z-Wave

Z-Wave ist ein internationaler Standard für Hausautomatisierungslösungen und verschiedenste Unternehmen, die diese Technologie nutzen, sind Mitglieder der Z-Wave Alliance [ZWa20]. Laut dieses Verbandes wird Z-Wave in bereits mehr als 2400 Produkten verwendet, welche alle von der Allianz zertifiziert wurden, um die Interoperabilität der Geräte sicherzustellen. Das Protokoll verwendet je nach Region unterschiedliche Frequenzen, aber diese liegen meist im Bereich zwischen 800 und 900 Megahertz. Dieses Frequenzband ist für Haussteuerungs-Anwendungen reserviert und unterliegt somit nicht einer starken Belastung wie das 2,4 GHz und 5 GHz-Band.

Darüber hinaus ist die Reichweite dieser Frequenz höher und könnte theoretisch bis zu einer Entfernung von 100 Metern genutzt werden, wobei dies aus Stabilitätsgründen nicht empfehlenswert ist. Daten werden bei Z-Wave mit einer Geschwindigkeit von bis zu 100 kB/s übertragen und sind mit AES-128 verschlüsselt [Zwa18]. Oft werden die verschiedenen Z-Wave Geräte über einen zentralen Hub verwaltet, der mit dem Internet verbunden ist, obwohl das Z-Wave Netz eine Mesh Topologie aufweist, d.h. das Signal springt von Gerät zu Gerät weiter.

ZigBee

Im Gegensatz zu Z-Wave basiert ZigBee auf dem offenen Standard IEEE 802.15.4² und nutzt somit bereits vorhandene, stabile und getestete Technologie [Zig20]. ZigBee wird somit einerseits durch die ZigBee Alliance und andererseits indirekt durch die Standards Development Organization gepflegt und gefördert. Die Zigbee Alliance besteht aus Mitarbeitern verschiedener Firmen und möchte offene, globale Standards für kabellose Gerätekommunikation entwickeln und verbreiten. Deshalb wird ZigBee auch von einer ähnlich hohen Anzahl an Produkten wie Z-Wave genutzt. Anders als Z-Wave befindet sich bei diesem Protokoll der Frequenzbereich im hoch ausgelasteten 2.4 GHz-Band, weshalb es öfters zu Interferenzen kommen kann. Im Indoor-Bereich liegt die durchschnittliche Reichweite bei 75–100 Meter mit einer Datenübertragungsrate von 250 kB/s und einer AES-128 Verschlüsselung. Die angegebene Reichweite wird effektiv, aber auch nur durch die Mesh Topologie erreicht, da höhere Frequenzen meistens nur geringere Distanzen überbrücken. Somit ist ZigBee in vielen Punkten Z-Wave sehr ähnlich, wobei es eine höhere Datenübertragungsrate hat, aber dafür mehr mit Interferenzen kämpfen muss.

Für den Endnutzer sind letztendlich die Interoperabilität und der Kostenfaktor entscheidend. Die Interoperabilität ist bei beiden Protokollen gegeben, aber aufgrund des offenen Standards und um die Kosten möglichst gering zu halten, wird diese Arbeit nur Produkte mit dem ZigBee-Protokoll verwenden. Das Spektrum an Aktoren ist breit und erstreckt sich von Leuchtmitteln über Alarmanlagen bis hin zu Motoren zum Öffnen von Fenstern. Der Fokus dieser Arbeit liegt bei der Erkennung von Nutzeraktivitäten, auch die Erweiterung der Funktionalität von anderen Assistenzsystemen angesprochen wird. Aktoren werden oft durch Sensorereignisse oder bestehende Automatisierungslösungen gestartet, sodass dort nicht immer eine direkte Interaktion mit dem Nutzer ableitbar ist. Deshalb werden Aktoren hier nicht weiter betrachtet, aber bleiben trotzdem relevant, wenn es um die Nutzung durch andere AAL-Lösungen geht.

1.2.2 openHAB-System

Der **open Home Automation Bus**, kurz openHAB, ist eine Open-Source-Lösung zur Verwaltung und Steuerung einer SH-Umgebung [ope20]. Das Ziel dieser Plattform ist es bereits vorhandene Systeme nicht zu ersetzen, sondern in einer zentralen Stelle zu vereinen. Dies ermöglicht eine gemeinsame Nutzung von Produkten unterschiedlicher Hersteller, sogar wenn deren Geräte nicht dieselben Funkprotokolle verwenden. Somit ist der Endnutzer nicht von einer Firma bzw. einem Firmenkollektiv abhängig und könnte beispielsweise auch vereinzelte Z-Wave Geräte im Zusammenspiel mit bereits vorhandenen ZigBee Produkten nutzen. Ein weiterer Vorteil, vor allem im Hinblick auf die Erkennung von Tagesrhythmen und medizinischen Daten, ist dass der Nutzer entscheiden kann, welche Daten überhaupt genutzt werden und ob das System mit dem Internet verbunden sein soll. Die SH-Verwaltung mit OH funktioniert sowohl online als auch offline und bezeichnet sich selbst gern als "Intranet of Things".

Da OH komplett in Java geschrieben ist, hängt es nur von der Java Virtual Machine ab und ist deshalb auf vielen Plattformen verfügbar. So kann es auf einem eigenen lokalen Rechner un-

²Technischer Standard für WPAN mit niedrigen Übertragungsraten

ter macOS, Windows oder Ubuntu betrieben, aber auch beispielsweise auf einem Raspberry Pi installiert und genutzt werden. Nach der Installation erfolgt eine Ersteinrichtung durch den Nutzer, bei der ein Paket zur Installation bestimmter Komponenten und Module ausgewählt wird, welche aber auch nachträglich hinzugefügt oder entfernt werden können. Sobald eines dieser Pakete installiert wurde, hat der Nutzer die Möglichkeit, über ein grafisches User Interface neue *Bindings*, *Things*, *Items*, diverse Datenbanken und OH-Komponenten zu installieren und diese zu verwalten. Die zuerst genannten Objekte sind spezielle Begriffe des OH-Kosmos und werden daher im Folgenden genauer betrachtet.

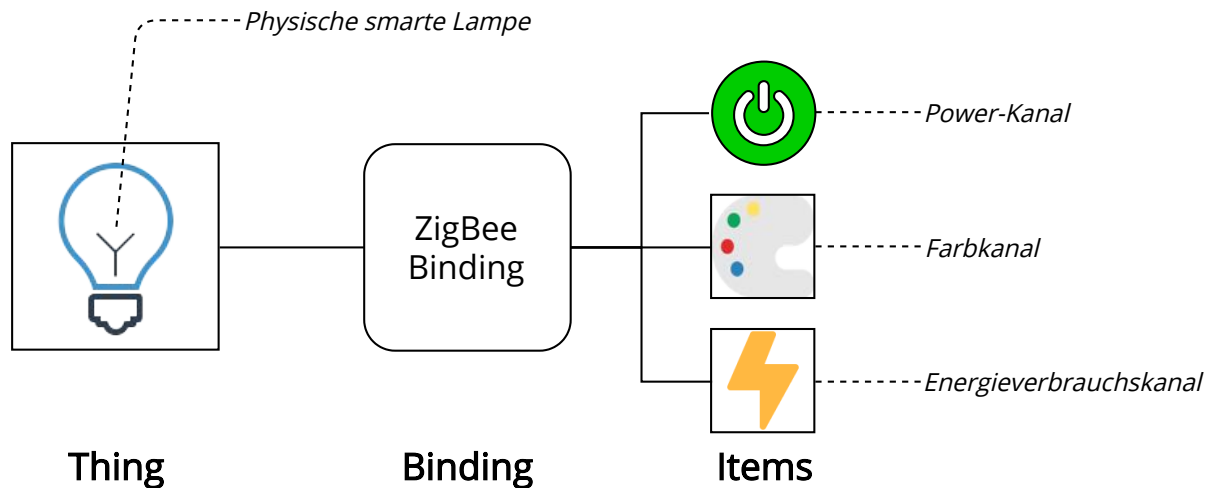


Abbildung 1.3: Zusammenspiel von Bindings, Things und Items in openHAB

Bindings

Die Bindings sind die Schnittstelle zu anderen Systemen und den physischen Geräten. Um bestimmte Geräte überhaupt erst in der OH-Verwaltungsschnittstelle zu sehen, muss das passende Binding installiert sein, welches eine Verknüpfung des Gerätes mit den virtuellen Komponenten ermöglicht. Sollte es das passende Binding nicht geben, so kann man es sich nach Bedarf auch selbst entwickeln, da die Plattform eine Open-Source-Lösung ist.

Things

Die sogenannten Things sind Abbildungen der physischen Geräte, aber sie können andere Datenquellen und Dienste abbilden, wie beispielsweise einen Web Service, der die lokalen Wetterdaten abrufen. Sie dienen zur Einrichtung und Konfiguration, aber nicht direkt zur Steuerung der Abläufe im SH. Die Einrichtung kann manuell erfolgen, sodass notwendige Eigenschaften, wie die IP Adresse des Geräts, vom Nutzer eingetragen werden. Vom System erkannte Things können ebenfalls automatisiert eingerichtet werden. Eine spezielle Art von Things sind die "Bridges", welche als IP-Gateway für nicht IP-basierte Automatisierungslösungen dienen und dem System Zugang zu diesen Geräten verschaffen. Nachdem ein Thing hinzugefügt wurde, bietet es verschiedene Kanäle an, aus denen Informationen über das verknüpfte Gerät bezogen werden können. Eine smarte Glühlampe könnte beispielsweise nicht nur ihren aktuellen Betriebszustand, sondern auch ihre Farbe oder Intensität anzeigen.

Items

Die virtuelle Schicht besteht aus den Items, welche eine Funktionalität abbilden. Diese Funktionalität wird vom Nutzer bestimmt, sodass dieser beispielsweise ein Item anlegt, welches nur

mit dem Farbkanal des eingerichteten Things seiner Glühbirne verknüpft ist, weil nur dieses Attribut relevant ist. Ein Item ist mit einem Kanal verknüpft, aber die Kanäle eines Things können mit mehreren Items verknüpft sein. Items können auch gruppiert werden, um beispielsweise Räume oder die Gesamtheit aller Geräte mit der gleichen Funktionalität abzubilden.

Das alles bedeutet für den Nutzer, dass das passende Binding installiert werden muss, die lokalen Geräte als Things konfiguriert und deren Kanäle als Items mit der jeweils gewünschten Funktionalität eingestellt werden müssen. In Abbildung 1.3 ist dieses Zusammenspiel exemplarisch dargestellt, indem eine smarte Lampe, welche mit dem ZigBee Protokoll arbeitet, über das ZigBee Binding als Thing angemeldet wird und die einzelnen Kanäle des Things werden als Items deklariert, um diese anzu steuern zu können. Trotz eines grafischen User Interfaces und der automatischen Erkennung von Things erfordert der komplette Prozess technisches Knowhow. Auch das Aufstellen von Regeln zur Steuerung der Prozesse erfolgt momentan noch rein programmatisch. Somit hat der Nutzer einen initialen Einarbeitungsaufwand, aber das betrifft das Thema Hausautomatisierung im Allgemeinen.

1.2.3 Loomo-Roboter

Loomo ist ein Roboter von Segway Robotics und soll hier stellvertretend für alle bisher am Markt erhältlichen Assistenzroboter in SH-Umgebungen stehen, welche Nutzen aus einem Profil des Pflegebedürftigen ziehen könnten. Der Loomo ist dafür gut geeignet, da dieser nicht auf einen speziellen Anwendungsfall zugeschnitten ist und auch verglichen zu anderen Assistenzrobotern relativ kostengünstig, aber trotzdem sehr flexibel ist [Seg20a]. Der Roboter ist als eine Art mobiler Begleiter gedacht, hat eine Vielzahl an Sensoren verbaut und besitzt an der Rückseite eine Fläche (Extension Bay), um eigene Hardware Erweiterungen anzubauen, wie in Abbildung 1.4 zu sehen ist. Folgende Sensoren sind verbaut [Seg20b]:

- Intel RealSense Modul (Tiefenkamera zur Erkennung von Menschen, Wegfindung, Lokalisierung und Kartografierung)
- HD Kamera
- Fünf Mikrofone (Zur Lokalisierung von Geräuschquellen und Reduktion von Hintergrundgeräuschen während der Stimmerkennung)
- Touch Display
- Ultraschall-Sensor (Erkennung von Hindernissen)
- Hall-Sensoren in den Rad-Motoren zur Stabilisierung
- Inertialsensoren (Bestimmung der Pose des Roboters und der Orientierung des Kopfes)

Zusätzlich zu den Sensoren sitzt auf dem Kopf noch eine Recheneinheit mit einem speziellen auf Android 5.1 basierenden System für Loomo. Um eigene Funktionalitäten zu entwickeln, kann daher eine eigene Android App entwickelt und darüber das Loomo-SDK³ eingebunden werden. Mit Hilfe des SDK lassen sich so gut wie alle Sensoren und Motoren des Roboters auslesen und steuern, um eigene Ideen zu realisieren. Außerdem kann sich Loomo auch mit W-LAN Netzwerken verbinden und Nachrichten mit einer Größe bis zu 1 MB versenden, was einen Austausch mit anderen kommunikationsfähigen Geräten im Haushalt ermöglicht.

³Software Development Kit

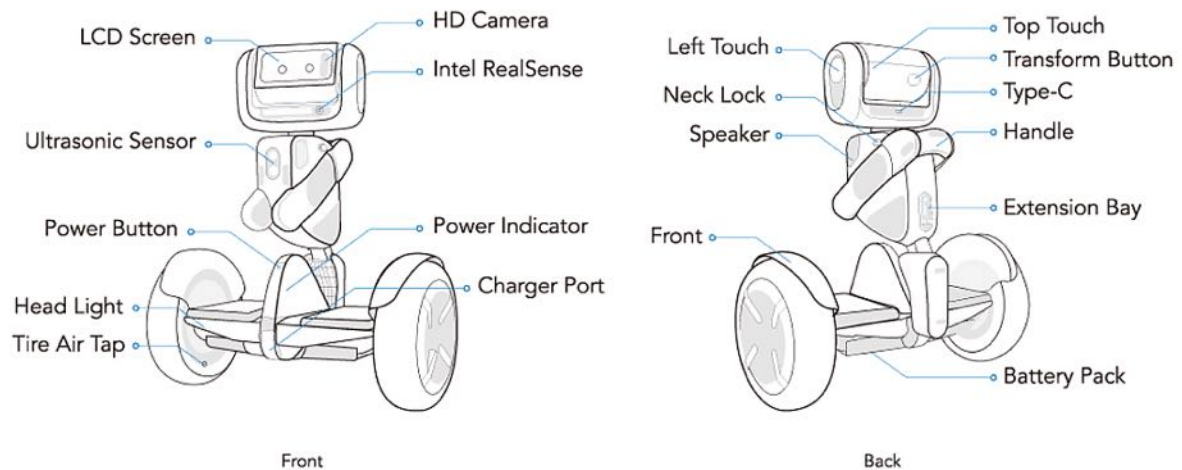


Abbildung 1.4: Hardware Komponenten des Loomo-Roboters [Seg20b]

1.3 Zusammenfassung

Nach diesem Kapitel sollten die grundlegenden Begriffe und Konzepte, die für den weiteren Verlauf der Arbeit relevant sind, geklärt sein. Das Gebiet des AAL wurde umrissen und gezeigt, welche Bereiche es abdeckt, was damit erreicht werden soll und wie es sich im Laufe der Zeit weiterentwickelt hat. Anschließend wurde im Detail betrachtet, was ADLs auszeichnet und wie sich diese kategorisieren lassen. Es wurde erläutert, was Sensoren sind und was die zwei gängigsten Funkprotokolle Z-Wave und ZigBee auszeichnet. Die Verwaltung der Sensoren und anderer smarter Geräte soll mittels OH erfolgen, weshalb diese Open-Source-Plattform mit ihren Besonderheiten, wie Bindings, Things und Items erklärt wurde und welche Vorteile diese für den Endnutzer bringt. Zuletzt wurde der Loomo von Segway vorgestellt, welcher als Repräsentant für die Assistenzroboter stehen soll, für die das Potenzial eines Aktivitätsprofils aufgezeigt werden soll. Im nächsten Kapitel werden Arbeiten analysiert, welche ähnliche Ziele wie diese Arbeit verfolgten.

2 Beschreibung verwandter Arbeiten aus dem Bereich der Aktivitätserkennung

In diesem Kapitel geht es um aktuelle verwandte Arbeiten zum Thema Aktivitätserkennung und wie sich diese Arbeit von den bereits bestehenden Arbeiten abgrenzt. Der größte Teil der Forschungsarbeiten im Bereich des AAL beschäftigt sich mit der Erkennung von Aktivitäten, gefolgt von der Kontrolle des Vitalstatus und der Positionsbestimmung [Cal+17].

2.1 Erkennung von Aktivitäten in Zusammenspiel mit einem Assistenzroboter

Viele dieser Arbeiten beschäftigen sich mit der Erkennung von abnormalen Verhaltensweisen bei bereits dementen Personen oder um die ersten Anzeichen einer Demenz zu erkennen. Fernandez-Carmona et al. wollten dies auch in ihrer Arbeit erreichen, indem sie über mehrere Sensoren globale Aktivitäten und über eine einzelne RGB-D Tiefenkamera lokale Aktivitäten erkennen wollten [Fer+17]. Die Tiefenkamera, welche bei dem Versuch an einer fixen Position befestigt war, sollte dabei einen Roboter darstellen, da die meisten Modelle eine solche Kamera besitzen. Es wurden Sensoren für Präsenz, Temperatur, Licht, Energieverbrauch und Türkontakt installiert, welche ebenfalls über OH verwaltet wurden. Dort wurden allerdings nur Sensoren mit dem Z-Wave Protokoll verwendet. Die Aktivitäten, die über die Sensorik und die Tiefenkamera erkannt wurden, sollten anschließend von einem medizinischen Experten ausgewertet werden, um Symptome erkennen zu können. Die lokalen Aktivitäten haben sich auf die Bewegung des von der Tiefenkamera erkannten Körpers bezogen und auch die globalen Aktivitäten nahmen Bezug auf Bewegungen in den durch die Sensoren überwachten Bereichen. Es wurden also keine spezifischen ADLs erkannt, sondern die allgemeine Aktivität im Haushalt und die Abweichung von dieser. Darüber hinaus wurde auch nur die Dauer einzelner Aktivitäten untersucht und nicht das genaue Zeitfenster, sodass sich daraus kein Tagesablauf konstruieren lässt. Trotzdem ist die Nutzung von OH als Middleware und der Austausch mit der Tiefenkamera bzw. dem Roboter nennenswert für diese Arbeit.

Schwiegelshohn et al. berichten in ihrer Arbeit über das AAL-Programm *RADIO*, welches Assistenzroboter in Hausautomatisierungssysteme integrieren soll [Sch+17]. Das erste Ziel des Projektes ist es, Daten zu sammeln und zu analysieren, um die Stimmung und ADLs des Nutzers herauszufinden. Dafür werden Bild- und Tonaufnahmen eines Roboters ausgewertet, welcher dem Nutzer, ohne diesen zu stören, assistieren soll. Dieser Roboter soll den Patienten bei ADLs unterstützen, soziale Interaktion und Inklusion bieten, Alarme auslösen und permanent den Gesundheitsstatus überwachen können. Um die nötigen ADLs zu erkennen, wurden sechs Kategorien identifiziert, die durch den Audiosensor und die Kameras des Roboters erkannt werden können: Eigen-Performance (Hygiene, Ankleiden, Gehen), Fortbewegung, Aktivitätslevel, Hobbies, Schlafrythmus und Psyche. Zusätzlich bewegt sich der Roboter in einer SH-Umgebung mit einer Vielzahl an Sensoren, welche ihm bei seiner Aufgabe mit weiteren Daten unterstützen sollen. *RADIO* verfolgt also genau den umgekehrten Weg, den diese Arbeit einschlagen will. Bei *RADIO* sammelt der Roboter hauptsächlich alle Daten und soll Aktivitäten erkennen, um den Nutzer dabei zu unterstützen. In dieser Arbeit soll die SH-Umgebung die Routinen des Nutzers erkennen und die Informationen an einen Roboter oder weitere AAL-Systeme übermitteln.

2.2 Aktivitätsbestimmung durch Nutzung von tragbarer Sensorik und Standortdaten

Jiménez und Seco untersuchten die Erkennung verschiedener persönlicher Aktivitäten zur Laufzeit anhand von Events [JS18]. Hier wurden verschiedene binäre Sensoren an verschiedenen Alltagsgegenständen befestigt, sowie Bluetooth Beacons in der Testumgebung platziert. Ein Smartfloor wurde ebenfalls integriert, wie in Abbildung 2.1 zu sehen ist. Binäre Sensoren sind meist Bewegungsmelder, die nur zwei Status haben, je nachdem ob gerade eine Bewegung erkannt wird oder nicht. Die Bluetooth Beacons kommunizieren mit einer Smartwatch am Arm des Nutzers und der Smartfloor besitzt mehrere Sensoren im Boden. Dieser aufwändige Testaufbau soll dabei helfen, die genaue Position des Nutzers herauszufinden, um dessen aktuelle Aktivität zu bestimmen. Mit Hilfe eines rekursiven Bayes Filter, welcher auf dem naiven Bayes Klassifizierer basiert, wurde bestimmt, welche Aktivität in welchem Zeitfenster mit der höchsten Wahrscheinlichkeit als nächstes stattfindet. Der Datensatz umfasste einen Zeitraum von einer Woche und so konnte untersucht werden, welche Aktivitäten öfters und zu bestimmten Zeiten in diesem Intervall stattgefunden hatten. Die Sensoren im Fußboden sollten Aufschluss darüber geben, wo sich der Nutzer in der Versuchsumgebung befindet und somit zur Bestimmung der Aktivität beitragen, wie beispielsweise der Schlussfolgerung, dass der Nutzer schläft oder sich ausruht, wenn keiner der Sensoren im Fußboden etwas erkennt. Die erkannten Aktivitäten wurden in drei verschiedene Zeitfenster kategorisiert, um die unterschiedlichen Tageszeiten darzustellen und beispielsweise morgendliche Aktivitäten in der Küche dem Frühstück zuordnen zu können. Insgesamt wurde eine durchschnittliche Genauigkeit von 68 Prozent erreicht, was bei 24 verschiedenen erkannten Aktivitäten vergleichsweise gut ist. Trotzdem ist der Aufbau sehr aufwändig und die Zeitfenster könnten präziser sein. Lauterbach et al. haben ebenfalls versucht über ein Sensorenetz im Fußboden ungewöhnliche Aktivitäten von Senioren zu erkennen [Lau+18]. Dort sollte über längere Zeit erkannt werden, wie sich Personen in ihrer Umgebung bewegen und wann sie von ihren täglichen Routinen abweichen. Die Sensoren im Fußboden sind zwar sehr unauffällig, aber die Installation ist aufwändig und teuer, da die Sensoren unter dem Bodenbelag platziert werden müssen. Die Idee könnte für Pflegeheime recht interessant sein, aber für den privaten Gebrauch im Eigenheim ist sie eher unpraktisch.

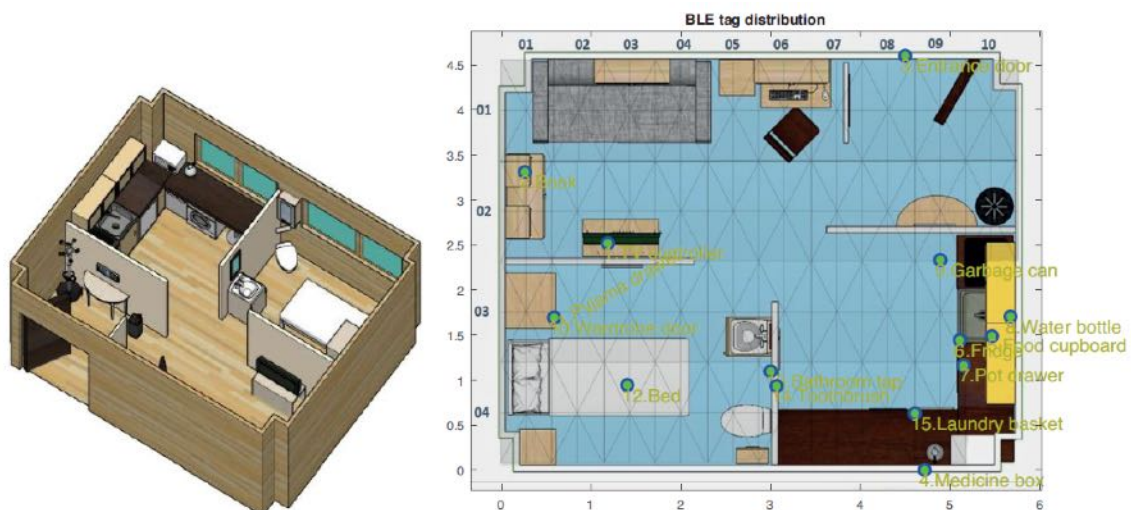


Abbildung 2.1: Versuchsaufbau von Jiménez und Seco [JS18]

Martin Franke behandelte in seiner Dissertation ein generisches Verfahren zur Klassifikation von Aktivitäten, mithilfe eines dynamischen Sensornetzes und einer selbst entwickelten semantischen Middleware [Fra16]. Ziel war es, den Tagesrhythmus einer Person herauszufinden, um

diesen gegebenenfalls zu verbessern, falls ungesunde Tendenzen zu erkennen sind. Dafür wurden größtenteils tragbare Sensoren, Web Services und eine Wetterstation mit zusätzlichen Sensormodulen verwendet, um obendrein Lichtverhältnisse und den CO₂-Gehalt in der Luft messen zu können, wobei alle Geräte dem Bereich der Konsumenten-Hardware zuzuordnen sind. Ein wesentlicher Bestandteil der Arbeit war die Entwicklung der semantischen Middleware, mit der neue Sensoren automatisch erkannt und deren Daten dem System zugeführt werden sollten, was in der vorliegenden Arbeit durch OH realisiert wird. Da der Fokus auf Rhythmen lag, mit denen chronobiologische Metriken abgeleitet wurden, beschränkte sich die Auswahl von Aktivitäten auf 10 verschiedene Kategorien. Wenn Aktivitäten erkannt wurden, sind diese einer Kategorie zugeordnet, mit einer Zeit versehen und verkettet worden um einen Tagesablauf zu bilden. Die Zeitfenster für eine Routine waren stundenweise gewählt, sodass möglichst wenig Abweichungen vom eigentlichen Ergebnis vom Algorithmus zurückgeliefert wurden. Die Nutzerschnittstelle war ein Command Line Interface, was die Bedienung für die meisten Nutzer unhandlich gestaltet, aber die Entwicklung eines solchen Interfaces war auch kein Bestandteil der Zielstellung. Auch dort wurde ein kostengünstiger Ansatz verfolgt und die Aktivitäten wurden in kleinere Zeitfenster eingeteilt, sodass eine genauere Bestimmung eines Tagesablaufs möglich war. Allerdings wurde viel mit tragbarer Sensorik und Messungen außerhalb der eigenen Wohnung gearbeitet und mehr Fokus auf die Vitalwerte gelegt, was für diese Arbeit eher weniger wichtig ist, da sich hier auf stationäre Indoor Sensorik sowie einen Assistenzroboter im Haushalt konzentriert wird.

2.3 Bestimmung von Aktivitäten mit Hilfe weniger stationärer Sensoren

Dimitrievski et al. haben in ihrer Arbeit versucht, möglichst unaufdringliche Sensoren zu verwenden, aber haben zusätzlich auch darauf geachtet, dass diese möglichst günstig sind und der Nutzer kein Gerät am Körper tragen muss, wie beispielsweise eine Smartwatch [Dim+16]. Sie kritisieren, dass die meisten Lösungen zur Aktivitätserkennung zu sehr in das Leben des Nutzers eingreifen oder zu kompliziert sind, um sie in einem durchschnittlichen Haushalt umzusetzen. Daher wurden dort ganz einfache Sensoren direkt mit einem Mikrocontroller zur Datenerfassung verkabelt. Die Gehäuse für die Sensoren wurden mit einem 3D Drucker erstellt und über mehrere Experimente wurden die optimalen Standorte für die Sensoren in der Testumgebung ermittelt. Die Sensoren und ihre Standorte sind in Abbildung 2.2 zu sehen, wodurch die Einfachheit der genutzten Sensoren sichtbar wird. Dadurch haben diese zwar immer die passenden Daten geliefert, aber die richtigen Standorte müssen zunächst erkannt und die Sensorik individuell für jeden Nutzer erstellt werden, obwohl es schon marktfähige Produkte gibt. Die Idee einer kostengünstigen Variante ist gut und wird auch in dieser Arbeit verfolgt, aber trotzdem wären Produkte, die bereits auf dem Markt etabliert sind, für private Zwecke im kleinen Rahmen praktikabler.

Händler-Schuster et al. haben über eine geplante Pilotstudie berichtet, bei der ADLs über den Energieverbrauch erkannt und anhand der gefundenen Aktivitäten benötigter Unterstützungsbedarf ermittelt werden sollte, um im Zweifelsfall Pflegepersonal einzubeziehen [Hän+16]. Diese Studie richtete sich vor allem an Personen über 70 Jahren, welche alleine zu Hause lebten. Bei diesen Personen sollte der Energieverbrauch erfasst werden, um anschließend daraus Aktivitäten abzuleiten. Zusätzlich sollten die Personen in der Lage sein, alleine zu kochen, zu haushalten und über einen separaten Wasser- bzw. Energiezähler verfügen. Der Versuch war auf ein halbes Jahr ausgelegt und wurde dabei regelmäßig von Studienmitarbeitern begleitet, damit die teilnehmenden Personen nicht von technischen Problemen beeinträchtigt waren. Die Teilnehmer wählten über ein Tablet aus, mit welcher Aktivität sie sich beschäftigten, wobei es eine Auswahl aus 12 verschiedenen Kategorien gab. Die Auswahl sollte immer erfolgen, wenn

die Aktivität gerade durchgeführt wurde, konnte aber noch nachträglich getätigt werden. Strom- und Wasserverbrauchsdaten wurden über Geräte gemessen, welche direkt an den Zählern angebracht und über das Internet zur Laufzeit übermittelt wurden. In dieser Arbeit sollen keine solchen Zähler genutzt werden, aber der Stromverbrauch könnte auch über im Handel erhältliche smarte Steckdosen ermittelt werden, was eine mögliche Erweiterung zu dieser Arbeit sein könnte. Interessant ist allerdings die Idee einer App zur Auswahl der Aktivitätskategorie, da eine Zuordnung der erkannten Aktivitäten durch den Nutzer sehr hilfreich für eine gute Datenqualität ist und auch hier umgesetzt werden könnte.

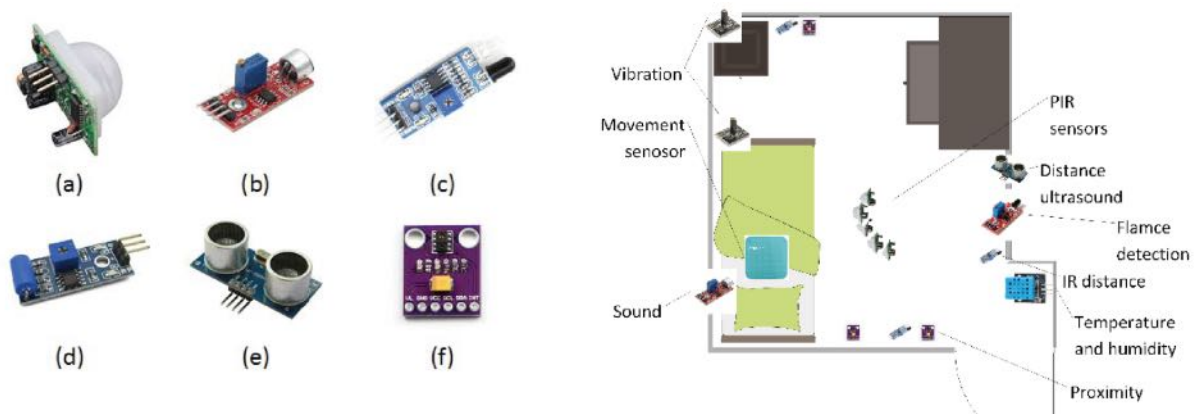


Abbildung 2.2: Versuchsaufbau und Sensorauswahl bei Dimitrievski et al. [Dim+16]: (a) Bewegungsmelder, (b) Mikrophon, (c) Infrarot Hindernissensor, (d) Vibrationssensor, (e) Ultraschall Distanzsensoren, (f) Sensor für Nähe

2.4 Zusammenfassung

In diesem Abschnitt wurden ähnliche Arbeiten beschrieben, welche sich auch mit der Erkennung von Aktivitäten mittels verschiedenster Sensorik oder Assistenzrobotern beschäftigen. Im Bereich der Assistenzrobotik hat sich gezeigt, dass oft Ton- und Videoaufnahmen verwendet werden, um Aktivitäten zu bestimmen. Diese Aktivitäten sind aber keine ADLs, wie sie in Abschnitt 1.1.2 erklärt wurden, sondern allgemeine körperliche Aktivitäten, welche durch Bewegungen der Gliedmaßen erkannt werden. Die Bestimmung von ADLs mit Standortdaten oder tragbarer Sensorik ist bereits sehr fortgeschritten in der Erkennung, aber aufgrund des aufwändigen Versuchsaufbaus oder der Notwendigkeit immer ein Gerät am Körper tragen zu müssen, wurden diese Ansätze hier nicht weiter verfolgt. Die Nutzung weniger Sensoren, welche in der Wohnung des Nutzers platziert werden, ist für diese Arbeit der interessanteste Ansatz und wird hier auch weiter verfolgt. Aber im Gegensatz zu den beschriebenen Ansätzen, wird hier marktreife SH-Sensorik verwendet. Im nächsten Kapitel werden die Anforderungen an das System zur Erkennung von Aktivitäten ermittelt, indem die Bedürfnisse aller relevanten Nutzergruppen analysiert werden.

3 Analyse der Anforderungen an ein System zur Erkennung von Aktivitäten

Nachdem das Hintergrundwissen für diese Arbeit vermittelt und verwandte Arbeiten beschrieben wurden, werden in diesem Kapitel konkrete Anforderungen an das System aufgestellt. Dafür wurden zuerst die Bedürfnisse der Nutzer an AAL-Systeme und anschließend ein genereller Anwendungsfall ermittelt. Mittels dieser Informationen wurden Anforderungen mit messbaren Erfolgskriterien aufgestellt.

3.1 Bedürfnisse der primären Nutzergruppe

Wie in Abschnitt 1.1.1 bereits erwähnt wurde, zielt AAL unter anderem darauf ab, dass ältere Menschen unterstützt werden und diese länger zuhause leben können. Diese Gruppe der älteren Menschen ist daher auch die primäre Nutzergruppe des Systems, welches in dieser Arbeit entwickelt werden soll. Gemeint sind hier alle Personen ab circa 65 Jahren, welche nicht mehr regelmäßig arbeiten gehen und alleine leben.

Werner et al. fassten in ihrer Arbeit einige Bedürfnisse dieser Gruppe bezüglich AAL-Systemen zusammen und zeigten, dass diese stark im Zusammenhang mit der Erfüllung der ADLs stehen [WPW15]. Zwischen 10–20 Prozent der Zielgruppe haben Probleme bei der Erfüllung dieser Aufgaben und bräuchten Unterstützung. Größere Schwierigkeiten bestehen bei instrumentellen ADLs, bei denen 10–30 Prozent der primären Nutzergruppe nicht mehr alleine zurechtkommen, wobei hauptsächlich Männer Probleme mit Tätigkeiten wie Wäsche waschen oder der Zubereitung von Mahlzeiten haben [WPW12]. Allerdings gestehen sich viele Senioren auch nicht ein, dass sie Hilfe benötigen. Deshalb denken hilfsbedürftige Rentner oft, dass sie keine Assistenztechnologie benötigen würden, obwohl ihr Pflegepersonal eine gegenteilige Meinung vertritt [Cou+08]. Bei den Personen, die gerne AAL-Systeme nutzen wollen, werden vor allem Systeme als nützlich empfunden, welche im häuslichen Umfeld zur Erfüllung von (instrumentellen) ADLs genutzt werden können. Allerdings fühlen sich die Nutzer unwohl, wenn ein Roboter bzw. ein Assistenzsystem gänzlich die Rolle einer Pflegekraft einnehmen würde [Hal+19]. Obwohl angenommen wird, dass sich jüngere Menschen eher mit Technologie auseinandersetzen, können auch ältere Personen motiviert sein, solche AAL-Technologien zu nutzen. Dafür müssen diese kompatibel mit deren aktuellen Lebensstil und Tagesabläufen sein [RKL09].

Ebenso besteht ein Bedürfnis nach Sozialisierung und somit eine Angst vor Isolation und fehlenden Beziehungen zu anderen Personen. Die Zielgruppe weist ein ausgeprägtes Bedürfnis an Sicherheit auf, dieses wird primär durch professionelle Pfleger und barrierefreie Umgebungen in Pflegeeinrichtungen erfüllt, als durch die eigene Wohnung. In den Pflegeheimen gäbe es ein soziales Netz für die älteren Personen, aber die meisten älteren Menschen nehmen einige Strapazen auf sich, um trotzdem weiterhin in der eigenen Wohnung bleiben zu können [Hof10]. Außerdem gibt es noch finanzielle Sorgen, die beachtet werden müssen, da eine sichere finanzielle Situation nicht immer gegeben ist und nicht jede Maßnahme von der Krankenkasse übernommen wird. In Abbildung 3.1 ist eine Zusammenfassung der Bedürfnisse der primären Nutzergruppe zu sehen.

3.2 Bedürfnisse weiterer Nutzergruppen

Weitere Gruppen, die Bezug zu dem System haben, sind einerseits die Pflegekräfte und andererseits jegliche Gruppen, die sich mit der Unterstützung und Finanzierung der primären Nutzergruppe beschäftigen.

Die primäre Nutzergruppe wird hauptsächlich von der sekundären Nutzergruppe gepflegt. Zu einem Teil geschieht dies durch professionelles Personal, aber zu einem größeren Teil, nämlich ca. 90 Prozent, durch die eigene Familie oder Freunde [NSS13]. Die Meinung von diesen Personengruppen über Assistenztechnologie spielt für viele Personen aus der primären Nutzergruppe eine entscheidende Rolle, wenn es um die Anschaffung eines solchen Systems geht [Cou+08]. Professionelle Pflegekräfte sind in den meisten modernen Ländern aufgrund des hohen Bedarfs knapp besetzt und somit oft stark überlastet. Meist haben sie nur wenig Zeit für jeden einzelnen Patienten und müssen in einigen Fällen, auch noch regelmäßig die Angehörigen über den Status informieren. Auch die nicht-professionellen Pfleger leiden unter Stress und Zeitdruck, da sie die Pflege ihrer Angehörigen in den meisten Fällen neben ihren eigentlichen Beruf wahrnehmen und auch selbst dadurch weniger soziale Kontakte pflegen können. Diese zusätzliche Belastung wirkt sich negativ auf die Gesundheit aus und führt oft zu chronischen Krankheiten und einer höheren Sterblichkeitsrate [Per+13]. Außerdem müssen sie für die Gesundheit und Sicherheit der pflegebedürftigen Person garantieren und oft auch finanzielle Unterstützung leisten. Die Unterbringung in einer Pflegeeinrichtung ist oft teurer als die Pflege im eigenen Heim durchzuführen und die finanzielle Belastung ist für viele Familien eine Bürde [RM15]. Darüber hinaus ist es für viele Menschen keine Option, die Angehörigen in ein Pflegeheim "abzuschieben". Meist werden die Familienmitglieder von einem schlechten Gewissen geplagt und möchten ihren Angehörigen die Möglichkeit geben, den Rest ihres Lebens in den eigenen vier Wänden zu verbringen [Hof10].

Die andere Gruppe an Interessenten, die tertiäre Nutzergruppe, ist heterogen und besteht aus Gesundheitsorganisationen, Pflegeeinrichtungen, Krankenkassen, Versicherungen und vielen Weiteren, die Interesse an dem Wohl der alternden Gesellschaft haben. Die Interessen dieser Stakeholdergruppe haben Nedopil et al. in ihrer Anforderungsanalyse bezüglich AAL-Systemen genauer erörtert [NSS13]. Für diese Gruppe stehen der Kostenfaktor und die Effizienz der

Bedürfnisse bezüglich eines AAL-Systems

Primäre Nutzergruppe	Weitere Nutzergruppen
<ul style="list-style-type: none"> • Unterstützung bei (instrumentellen) ADLs • Nutzung im häuslichen Umfeld • Ergänzung zu einer menschlichen Pflegekraft • Integration in den bestehenden Alltag • Vermittlung physischer Sicherheit • Verbesserung sozialer Interaktion • Möglichst geringe Kosten 	<ul style="list-style-type: none"> • Entlastung bei der Pflegearbeit (Überwachung, Kommunikation, Prävention) • Sicherheitsgarantie für die primäre Nutzergruppe • Nutzung im eigenen Zuhause • Möglichst geringe Kosten • Effizienz des Systems • Interoperabilität • Ansprechendes Design • Schutz der Privatsphäre

Abbildung 3.1: Bedürfnisse der verschiedenen Nutzergruppen bezüglich eines AAL-Systems

Lösungen im Vordergrund, da sie diese gegebenenfalls für die Gesellschaft bereitstellen und finanzieren wollen. Dabei spielt auch eine Nutzung bereits existierender Standards oder die Verknüpfung zu anderen Services in dem Bereich eine wichtige Rolle. Außerdem achten diese Gruppen auch darauf, dass der Datenschutz eingehalten wird und die Lösungen nicht zu sehr in das Privatleben der Patienten eindringen. Ein attraktives Design der Lösung ist für die tertiäre Gruppe ebenfalls von Bedeutung. Für die sekundären und tertiären Nutzergruppen ist eine Zusammenfassung der Bedürfnisse, ebenso in Abbildung 3.1 zu sehen.

3.3 Anwendungsszenario

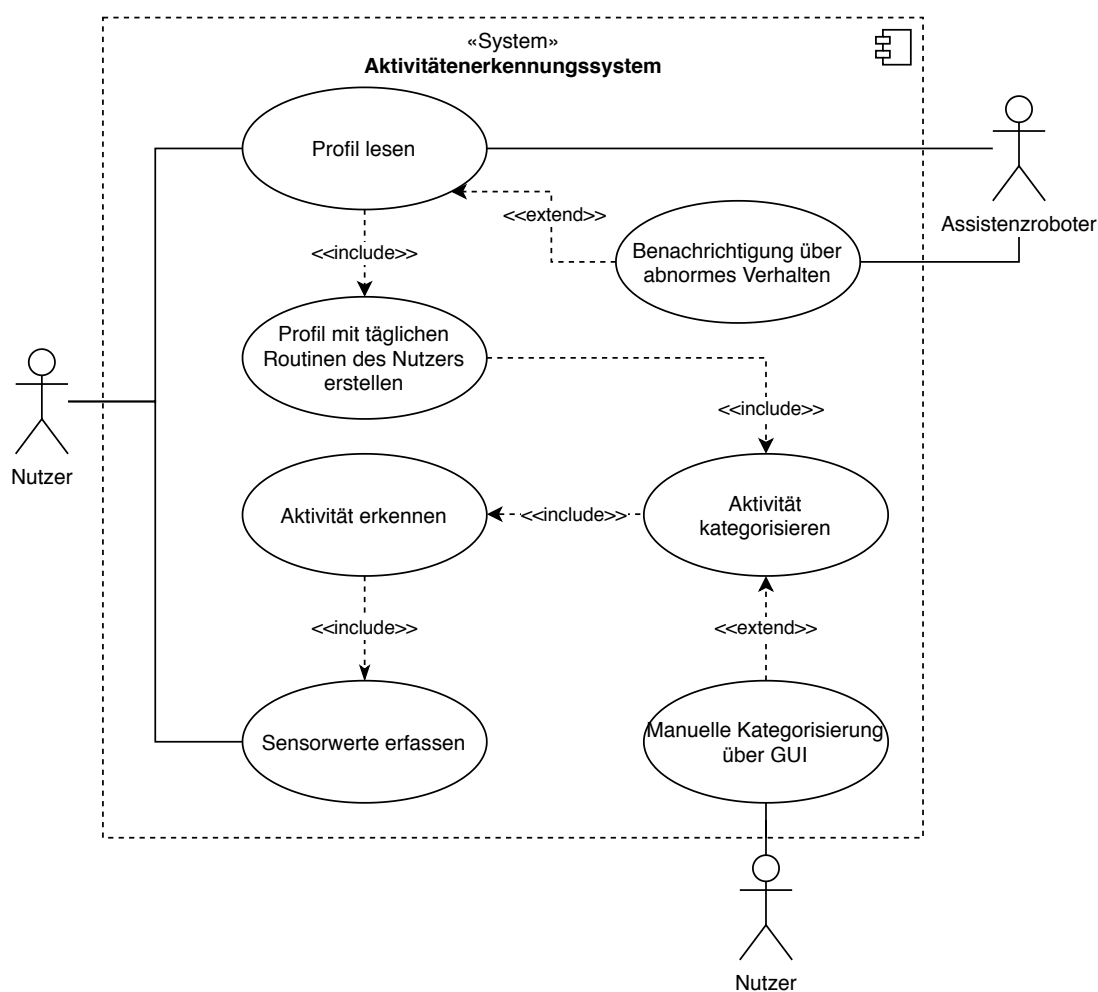


Abbildung 3.2: Anwendungsfalldiagramm für das erwartete Verhalten des Systems

Der grundsätzliche Anwendungsfall für das System ist in Abbildung 3.2 dargestellt. Der Nutzer muss keine speziellen Aufgaben erfüllen, sondern führt weiterhin seine täglichen Abläufe aus und wird dabei von Sensoren beobachtet, welche sich ändernde Werte erfassen, die vom System persistiert werden. Mit Hilfe der gespeicherten Daten klassifiziert das System Aktivitäten und fasst diese zu Abläufen zusammen, wobei dies auch durch den Nutzer über ein grafisches Interface durchgeführt werden kann. Die kategorisierten Aktivitäten werden dann in einen Tagesablauf des Nutzers eingetragen und dieser wird bei neu erkannten Aktivitäten aktualisiert. Das Nutzerprofil ist über eine visuelle Ausgabe einsehbar und wird über eine Schnittstelle für einen Assistenzroboter oder weitere AAL-Systeme zur Verfügung gestellt. Bei stark abweichendem Verhalten von der eingespeicherten Routine wird eine Benachrichtigung erstellt, welche

dem Roboter übermittelt wird, damit dieser entsprechend reagieren kann.

3.4 Definition der Anforderungen

Mit Hilfe der bisherigen Erkenntnisse dieses Abschnitts, lassen sich die Anforderungen an das System konkret definieren, um den Bedürfnissen der Nutzer so gut wie möglich gerecht zu werden.

- **Gebrauchstauglichkeit:**

Nach *DIN EN ISO 9241-11*¹ ist die Gebrauchstauglichkeit das Ausmaß, in dem das System durch bestimmte Nutzer in einem bestimmten Anwendungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und zufriedenstellend zu erreichen. Daher sollen diese Teilkriterien vom System erfüllt werden.

- **Effektivität:**

Alle Nutzer haben die Möglichkeit, eigene Aktivitäten einzutragen und diese im Tagesablauf einzusehen.

- Erfolgskriterium: Höchstens 10 % der Einträge schlagen fehl.

Alle Nutzer haben die Möglichkeit, die Einträge im Tagesablauf zu ändern oder zu löschen.

- Erfolgskriterium: Höchstens 10 % der Änderungen oder Löschungen schlagen fehl.

Sobald Sensorwertänderungen festgestellt werden, soll stets eine Aktivität dazu in der Datenbank eingetragen werden.

- Erfolgskriterium: Höchstens 5 % der Sensorwertänderungen werden nicht eingetragen.

- **Effizienz:**

Vom Nutzer eingetragene Aktivitäten sollen innerhalb weniger Sekunden nach dem Absenden des Formulars im Profil einsehbar sein.

- Erfolgskriterium: Eintrag ist zwei Sekunden nach Absenden des Formulars in der Tagesablaufansicht sichtbar.

Anfragen an die REST-Schnittstellen des Systems sollen innerhalb weniger Sekunden eine Antwort zurückgeben, um genaue Abläufe zu erfassen und den Nutzer nicht zu frustrieren.

- Erfolgskriterium: Antwort der Schnittstellen erfolgt innerhalb von fünf Sekunden.

- **Zufriedenheit und Akzeptanz:**

Die Nutzer sollten das System selber nutzen wollen.

- Erfolgskriterium: Maximal 10 % der Nutzer sollten das System nicht nutzen wollen, da nur knapp 10 % der Pflegebedürftigen von professionellen Personal gepflegt wird und dieses System wahrscheinlich weniger stark benötigt.

Eine transparente Übersicht aller gespeicherten Aktivitäten des Nutzers, soll jederzeit einsehbar sein, um die Akzeptanz des Systems zu erhöhen.

- Erfolgskriterium: Maximal 10 % der Nutzer denken, dass ihnen Daten vorenthalten werden.

¹ISO 9241 ist ein europäischer und internationaler Standard, welcher die Richtlinien der Mensch-Computer-Interaktion beschreibt.

- **Lernförderlichkeit und Selbstbeschreibungsfähigkeit:**

Die Nutzer sollen nach einer ersten Einführung durch einen geschulten Nutzer, das System selbstständig in vollem Umfang nutzen können.

 - Erfolgskriterium: Maximal 10 % der Nutzer sind nicht in der Lage, alle Funktionalitäten des Systems zu nutzen.
- **Zuverlässigkeit:**

Nach einem Systemausfall sind alle gespeicherten Daten weiterhin vorhanden sein.
- **Kosten:**

Bei allen Nutzergruppen ist der Kostenfaktor als bedeutsam eingestuft worden, vgl. Abschnitt 3.1 und Abschnitt 3.2, weshalb in der Anschaffung und Wartung keine hohen Kosten entstehen sollen. Ob die Kosten zu hoch sind, ist allerdings nutzerspezifisch, da dies von der finanziellen Situation abhängig ist. In Deutschland sind knapp 70 % der Bevölkerung bereit, bis zu 750 € für SH-Technologie inklusive Software und Installation zu bezahlen [Kun19], deshalb sollen dies die maximalen Kosten für das Basissystems, also SH-Hub und bis zu zehn Sensoren sein.
- **Datenschutz:**

Der Schutz der eigenen Daten und die Sorge, dass private Daten missbraucht werden, ist mittlerweile ein omnipräsentes Thema und soll hier vor allem wegen der Nutzung von kommerziellen Produkten beachtet werden. Deshalb soll das System ohne aktive Internetverbindung und herstellereigene Software funktionieren.
- **Interoperabilität:**

Das System soll bestmögliche Interoperabilität garantieren durch Nutzung standardisierter Protokolle und Datenformate.

 - Erfolgskriterium: Alle Schnittstellen sind durch HTTP-Methoden für REST-Services aufrufbar und geben eine Antwort im JSON-Format zurück.
- **Adaptivität:**

Das System soll sich automatisch an geänderte Verhaltensmuster des Nutzers anpassen, auch wenn abweichendes Verhalten protokolliert wird. Wöchentlich soll das Machine Learning (ML)-Modell von neu eingetragenen Aktivitäten des Nutzers lernen und diese Informationen bei der Klassifizierung mit einbeziehen.

 - Erfolgskriterium: Wöchentlich wird der Lernvorgang des ML-Modells angestoßen und neu hinzugefügte Aktivitäten werden vom Modell berücksichtigt.
- **Adaptierbarkeit:**

Nutzer haben die Möglichkeit bestehende Tagesabläufe zu ändern und eigene Abläufe zu erstellen, welche sich in den bestehenden abgespeicherten Tagesablauf integrieren lassen.

 - Erfolgskriterium: Modifizierte und selbst eingetragene Abläufe stehen an der korrekten Stelle im Tagesablauf und sind in der grafischen Schnittstelle sichtbar.

Nutzer können für Sensoren spezifische Standorte festlegen, welche die Art und Weise der Aktivitätserkennung direkt beeinflussen.

 - Erfolgskriterium: Die Änderung eines Sensorstandortes, ändert auch die Aktivitätsklassen, die dieser Sensor erkennen kann.

3.5 Zusammenfassung

Die konkreten Anforderungen an das System wurden nun anhand der Bedürfnisse der primären Nutzergruppe und den Bedürfnissen der Personen, die für deren Pflege verantwortlich sind, definiert. Ebenso wurde der allgemeine Anwendungsfall des Systems vorgestellt, welcher darin besteht, dass der Nutzer nichts an seinem Verhalten ändert und sein Tagesablauf vom System erkannt wird. Diesen kann der Nutzer einsehen oder aktiv mitgestalten, indem nicht erkannte Aktivitäten selbst einträgt. Hauptsächlich muss durch das System eine gute Erkennung der Tagesabläufe, bei geringen Kosten, gutem Datenschutz und hoher Gebrauchstauglichkeit gewährleistet werden. Im nächsten Kapitel werden ein technisches Konzept für ein System zur Erkennung von Aktivitäten vorgestellt, welches diese Anforderungen erfüllen soll und wie eine grafische Nutzungsschnittstelle für dieses System aussehen könnte.

4 Konzept eines adaptiven Smarthome-Systems zur Erkennung von Tagesabläufen

Nach der Klärung konkreter Anforderungen, wird in diesem Kapitel das Konzept für das System zur Erkennung der Aktivitäten und Erstellung des Nutzerprofils vorgestellt. Mit Hilfe der im vorherigen Kapitel aufgestellten Anforderungen entsteht ein technisches Konzept, welches die Methoden zur Persistierung der Sensordaten und des Nutzerprofils inklusive erkannter Aktivitäten festlegt. Außerdem werden die Möglichkeiten der Erkennung und Klassifizierung von Aktivitäten mit Vor- und Nachteilen erörtert.

4.1 Entwurf einer grafischen Benutzungsschnittstelle

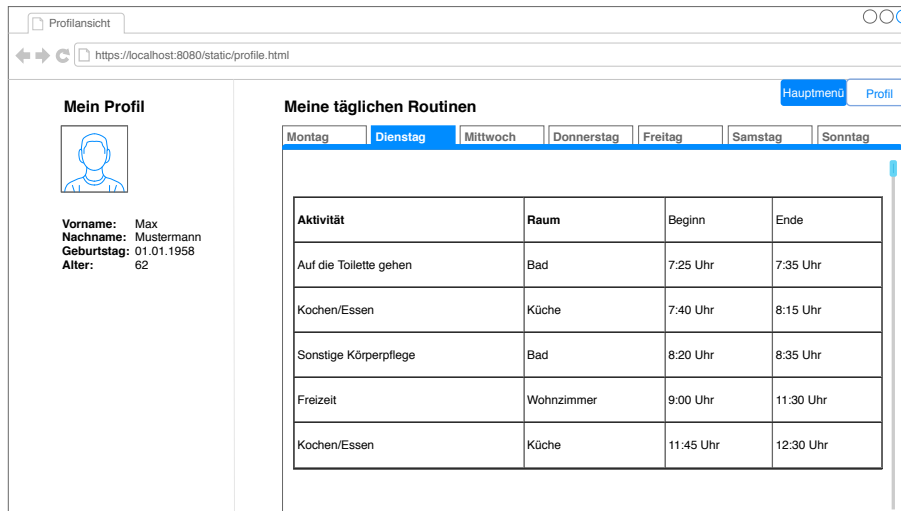
Um dem Nutzer eine Möglichkeit zur direkten Interaktion mit dem System zu geben und die Transparenz zu erhöhen, gibt es zwei relevante Ansichten für den Nutzer. Einerseits eine Profilsicht mit den täglichen Routinen des Nutzers und andererseits eine Ansicht, um Aktivitäten manuell einzutragen oder zu bearbeiten. Über OH lassen sich eigene HTML-Seiten und Skripte einbinden, welche auf dem System angelegt werden können, um diese auch ohne aktive Internetverbindung darzustellen. Im Profil, der primären Ansicht des Nutzers, stehen wenige personenbezogene Daten, sowie die nach Wochentagen eingeordneten erkannten Aktivitäten des Nutzers, wie in Abbildung 4.1 zu sehen ist. Abbildung 4.2 zeigt die Ansicht zur manuellen Bearbeitung von Aktivitäten durch den Nutzer, welche optional durchgeführt werden kann. Dadurch kann potenziell die Genauigkeit der Erkennung erhöht werden. Die Daten werden gekennzeichnet, um damit das ML-Modell zu trainieren. Der Nutzer soll sich aussuchen können, welche Aktivität er klassifizieren möchte und die Auswahl soll möglichst simpel durch eine Aktivierung der jeweiligen Schaltfläche geschehen, um den Lernaufwand zur Nutzung gering zu halten. Die Wahlmöglichkeiten orientieren sich an den Aktivitätsklassen aus Abschnitt 1.1.2, welche viele Aktivitäten zusammenfassen, damit der Nutzer nicht von einer Vielzahl an Optionen überwältigt wird. Die Ansichten sind über ein Hauptmenü miteinander verbunden, wodurch es die Möglichkeit gibt weitere Ansichten hinzuzufügen.

4.2 Technisches Konzept

Zur Erstellung des technischen Konzeptes wird geklärt, wie jegliche Daten abgespeichert werden und welche Daten zur Erstellung von Aktivitäten, Tagesabläufen und eines Nutzerprofils relevant sind. Außerdem wird eine Methode zur Klassifizierung von Aktivitäten entwickelt, wobei jeweils die Klassifizierung mittels ML und Aussagenlogik evaluiert wird. Außerdem wird geklärt, wie das Nutzerprofil mit den täglichen Routinen aussehen muss, damit es zur Informationsgewinnung an weitere AAL-Systeme übermittelt werden kann. Zusätzlich werden noch die adaptiven und adaptierbaren Mechanismen des Systems erarbeitet.

4.2.1 Persistierung der Daten

Um die Persistenz, also die dauerhafte Bereithaltung der Daten, sicherzustellen, nutzt das System eine Datenbank. Dafür stehen relationale oder nicht-relationale Datenbanken zur Auswahl,



The screenshot shows a web interface for a user profile. On the left, under 'Mein Profil', there is a placeholder for a profile picture and personal details: Vorname: Max, Nachname: Mustermann, Geburtstag: 01.01.1958, and Alter: 62. On the right, under 'Meine täglichen Routinen', there is a table with columns for 'Aktivität', 'Raum', 'Beginn', and 'Ende'. The table is filtered for 'Dienstag' (Tuesday).

Aktivität	Raum	Beginn	Ende
Auf die Toilette gehen	Bad	7:25 Uhr	7:35 Uhr
Kochen/Essen	Küche	7:40 Uhr	8:15 Uhr
Sonstige Körperpflege	Bad	8:20 Uhr	8:35 Uhr
Freizeit	Wohnzimmer	9:00 Uhr	11:30 Uhr
Kochen/Essen	Küche	11:45 Uhr	12:30 Uhr

Abbildung 4.1: Profilsicht mit täglichen Routinen des Nutzers

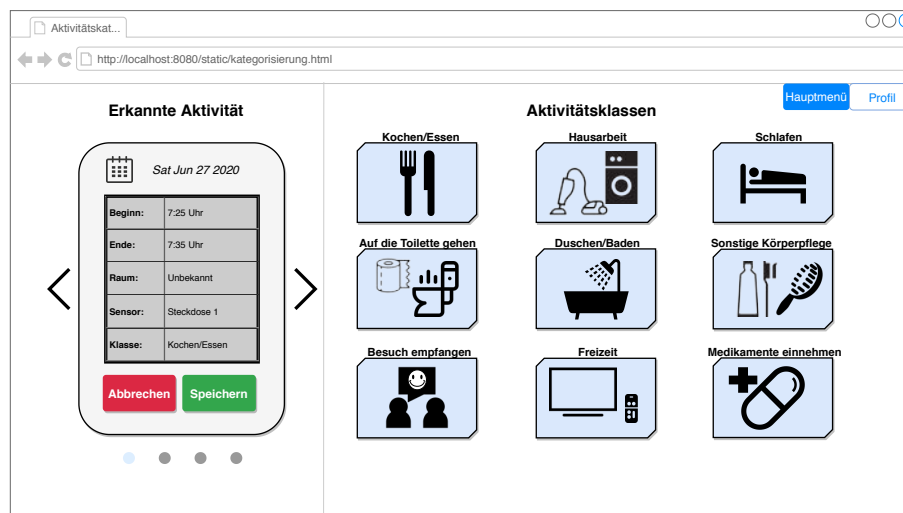


Abbildung 4.2: Ansicht der manuellen Klassifizierung durch den Nutzer

welche jeweils ihre Vor- und Nachteile besitzen. Relationale Datenbanken bestehen aus Tabellen mit Abhängigkeiten voneinander oder Beziehungen zueinander. Diese Tabellen haben eine fest vorgegebene Struktur in der die Daten abgespeichert werden und mittels SQL¹ manipuliert werden können. Nicht-relationale Datenbanken sind etwas flexibler bezüglich der Speicherung von Daten, da diese kein strukturiertes Schema besitzen. Stattdessen werden alle Daten in ein einziges Dokument und nicht in mehrere durch Beziehungen verknüpfte Tabellen geschrieben. Durch diese unstrukturierte Speicherung der Daten kann allerdings keine Konsistenz gewährleistet werden.

Für diese Arbeit werden beide Arten von Datenbanken genutzt. Aufgrund der Masse an unstrukturierten Sensordaten, die durch ein SH-System erfasst werden, bietet sich eine nicht-relationale Datenbank zur Persistierung an. Prädestiniert für eine solche Aufgabe sind Zeitreihendatenbanken, welche dafür optimiert sind, um Zeitreihendaten schnell abzuspeichern und abzurufen. Zeitreihendaten sind Paare von Zeitstempeln mit einem zugehörigen Wert in einem zeitlich geordneten Ablauf. Da die meisten SH-Sensoren zu jeder Zeit diskrete Sensorwerte besitzen, bietet sich eine Speicherung dieser Werte in einer Zeitreihendatenbank an, um möglichst

¹Structured Query Language

viele Sensordaten ohne ein vorgegebenes Schema persistieren zu können.

Zur Speicherung von Aktivitäten bzw. Routinen des Nutzers bietet sich eine relationale Datenbank an, da das Schema für diese Datenstrukturen für die Klassifizierung konsistent bleiben sollte. Allgemein ist es wünschenswert, dass die *ACID*²-Eigenschaften erfüllt werden, um Routinen verlässlich zu speichern und abzurufen, falls dies für einen Alarm benötigt wird. In Abbildung 4.3 ist ein Modell der Datenbank für Routinen und Aktivitäten zu sehen, welches die Beziehungen untereinander aufzeigt. Aktivitäten zeichnen sich durch eine Klasse und die Sensoren, mit denen diese erkannt wurden, aus. Zu welchem Zeitpunkt und in welchen Zyklen die Aktivitäten stattfinden, wird in nutzerspezifischen Routinen festgehalten.

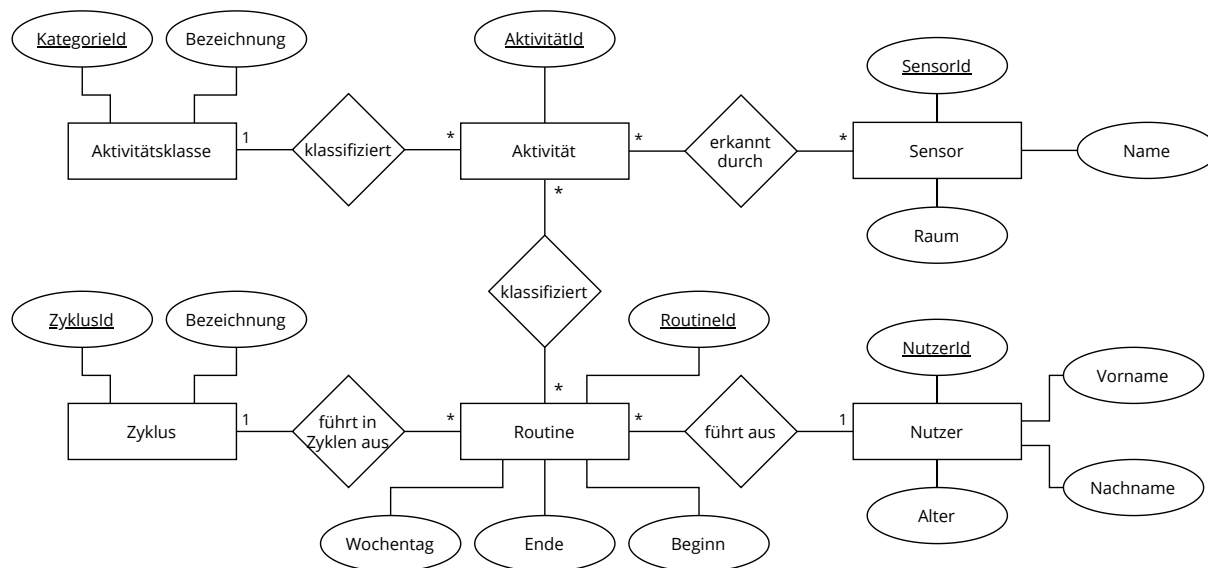


Abbildung 4.3: Entity-Relationship-Modell der Routinen- und Aktivitätsdatenbank

4.2.1.1 Technische Schemata von Aktivitäten und Routinen

Wie im vorherigen Abschnitt beschrieben, soll jeder Eintrag für eine Aktivität einen Verweis auf eine Klasse und einen oder mehrere Sensoren besitzen. Die Klassen orientieren sich in dieser Arbeit an denen aus Abschnitt 1.1.2, aber sie können je nach Bedarf erweitert werden. Bei den Sensoren sind nur die Bezeichnung und der Raum bzw. Ort, wo sich der Sensor befindet, relevant. Die gemessenen Sensorwerte werden in der zusätzlichen Zeitseriendatenbank abgespeichert und sind in dieser Arbeit nur zur Klassifizierung von ADLs bedeutend. Über die in der Aktivität involvierten Sensoren kann herausgefunden werden, wo die Aktivität stattfindet. Diese Information kann für einen Assistenzroboter hilfreich sein, um den Nutzer schneller zu finden. Routinen sind einem Nutzer zugeordnet und bestehen aus mindestens einer Aktivität mit Start-, Endzeit und dem Wochentag. Außerdem wird jeder Routine ein Zyklus zugeordnet, um auch Routinen zu erfassen, welche beispielsweise nur wöchentlich oder monatlich stattfinden. Die Trennung von Routine und Aktivität besteht hier, weil im Haushalt des Nutzers auch Aktivitäten stattfinden könnten, die nicht zu dessen Routinen gehören und eine Routine auch mehrere Aktivitäten enthalten kann.

Zur Übertragung an andere AAL-Systeme werden die angefragten Daten in das JSON-Format³ umgewandelt. Da bestehende Standards genutzt werden sollen, besteht hier die Wahl zwischen XML⁴ und JSON, da dies die gängigsten Formate zur Datenübertragung in Web Requests sind.

²In Deutsch: Atomarität, Konsistenz, Isolation und Dauerhaftigkeit

³JavaScript Object Notation

⁴Extensible Markup Language

Da für das Einlesen von XML-Dokumenten ein XML-Parser benötigt wird und JSON von jeder JavaScript Funktion eingelesen werden kann, wurde hier JSON als das präferierte Format gewählt.

4.2.1.2 Technisches Schema eines Nutzerprofils

Das Nutzerprofil soll sich vor allem durch die Routinen auszeichnen, da jeder Nutzer einen anderen Tagesablauf hat. Über die Routinen können andere Assistenzsysteme Informationen erlangen, die durch ein normales Profil nicht vermittelt werden. Trotzdem sind auch hier einige personenbezogene Daten, nämlich Name und Alter, über den Nutzer vorhanden, wie es in vielen anderen Anwendungen der Fall ist. Vorerst ist das System nur auf einen Nutzer ausgelegt, aber damit es erweitert werden kann, ist es wichtig, dass es ein Schema für Nutzer gibt, um zukünftig ebenfalls Mehrpersonenhaushalte abbilden zu können. Mit Hilfe von Abfragen ist es über das Nutzerprofil möglich herauszufinden, wo der Nutzer momentan ist und welche Aktivität gerade ausgeführt wird. Das Profil ist zwar aktuell einfach gehalten, aber es ist potenziell erweiterbar, damit auch bei Bedarf weitere Eigenschaften und Präferenzen des Nutzers ergänzt werden können. Beispielsweise könnte der Raum notiert werden, wo die meisten Aktivitäten des Nutzers stattfinden.

Genau wie bei Routinen und Aktivitäten, wird auch hier zur Übertragung von Informationen das JSON-Format genutzt. Für den Nutzer werden voraussichtlich die meisten Abfragen erstellt, da dieser mit hoher Wahrscheinlichkeit die größte Relevanz für weitere AAL-Systeme besitzt. Darüber hinaus kann das Profil in einer Weboberfläche dargestellt werden, um dem Nutzer die Möglichkeit zu geben, die abgespeicherten Daten einsehen zu können.

4.2.2 Klassifizierung der Aktivitäten

Bevor Aktivitäten oder Routinen in der Datenbank gespeichert werden können, müssen sie zuerst klassifiziert werden. Das bedeutet, dass zuerst ein Zeitfenster gewählt werden muss, welches man einer Aktivitätsklasse zuweisen kann. Dieses Zeitfenster kann fest vorgegeben sein und beispielsweise immer 30 Minuten betragen, sodass für diesen Zeitraum die Sensordaten ausgewertet werden können und dann eine Klasse dafür bestimmt wird. Denkbar wäre ebenfalls ein dynamisches Zeitfenster, welches anhand der Sensordaten bestimmt wird, sodass auch sehr kurze Aktivitäten beachtet werden können. Der Vorteil einer festgelegten Zeitspanne liegt darin, dass Aktivitäten, welche durch mehrere Sensoren in verschiedenen Räumen erkannt werden, auch beachtet werden. Zum Beispiel beim Staubsaugen bewegt sich die Person durch die ganze Wohnung und wird dabei von mehreren Sensoren erfasst, sodass dort die Erkennung eines dynamischen Zeitfensters anhand eines Musters in den Sensordaten schwieriger wäre, als ein statisches Zeitfenster zu wählen. Solche statischen Fenster haben auch in anderen Arbeiten zur ADL-Erkennung Verwendung gefunden [Fra16; JS18], da dies zu einer höheren Genauigkeit führte. Allerdings wurden die Zeitspannen relativ großzügig gewählt und das kleinste Fenster lag bei 30 Minuten. In dieser Arbeit soll anhand der Sensoraktivität für jede Minute die Aktivitätsklasse bestimmt werden, sodass aufeinander folgende Minuten mit der gleichen Aktivität zu einer Routine zusammengefasst werden können.

Zur Bestimmung der Klasse gibt es im Wesentlichen zwei Methoden, einerseits durch ML, andererseits anhand von Aussagenlogik. Diese zwei Ansätze werden im Folgenden untersucht und miteinander verglichen. Dafür wurden zwei öffentlich zugängliche Datensätze zur Analyse verwendet, welche in der Tabelle 4.1 aufgelistet sind. Der HTS-Datensatz besteht aus Zeitseriendaten von Gas-, Temperatur- und Feuchtigkeitssensoren, welche jeweils einer von drei verschiedenen Klassen zugeordnet wurden [Hue+16]. Beim ARAS-Datensatz wurden Zeitseriendaten von 20 Sensoren aus zwei verschiedenen Haushalten mit jeweils zwei Personen über einen

Zeitraum von 30 Tagen gesammelt [Ale+13]. Diese Daten wurden mit jeweils einer von 27 Klassen von Aktivitäten versehen. Beide Datensätze haben eine Grundwahrheit, da die Aktivitäten zu den gemessenen Sensorwerten von den Versuchspersonen selbst eingetragen wurden und sind somit äußerst wertvoll für eine Analyse.

	Zeitreihendaten	Sensoren	Sensortypen	Labels
HTS [Hue+16]	919438	10	3	3
ARAS [Ale+13]	5184000	20	7	27

Tabelle 4.1: Umfang der untersuchten Datensätze

4.2.2.1 Machine Learning-Klassifizierung

Mithilfe von ML sollen in den großen Datenmengen aus Sensorwerten Muster erkannt werden, durch die sich die Aktivitäten klassifizieren lassen. Unter Nutzung der zwei Datensätze aus Tabelle 4.1 und der *Scikit-learn* Bibliothek für Python, wurden mehrere bereits implementierte Algorithmen auf ihre Eignung geprüft. Dafür wurde die k-fache stratifizierte Kreuzvalidierung mit einem Wert $k=10$ angewandt. Das bedeutet, dass die Menge an Daten in k gleich große Teilmengen aufgeteilt wird, wobei durch die Stratifikation darauf geachtet wird, dass jede der Teilmengen eine ähnlich große Verteilung besitzt. Anschließend werden k Durchläufe ausgeführt, bei denen jeweils die aktuell ausgewählte Teilmenge zur Validierung dient und die restlichen $k-1$ Teilmengen zum Training genutzt werden. Dies wurde für sechs verschiedene Algorithmen bei beiden Datensätzen durchgeführt, um eine ungefähre Genauigkeit dieser Ansätze zu ermitteln. In Abbildung 4.4 ist zu sehen, dass hier zwei Algorithmen wesentlich bessere Werte aufweisen als der Rest, nämlich k-Nearest-Neighbor (KNN) und Classification And Regression Tree (CART).

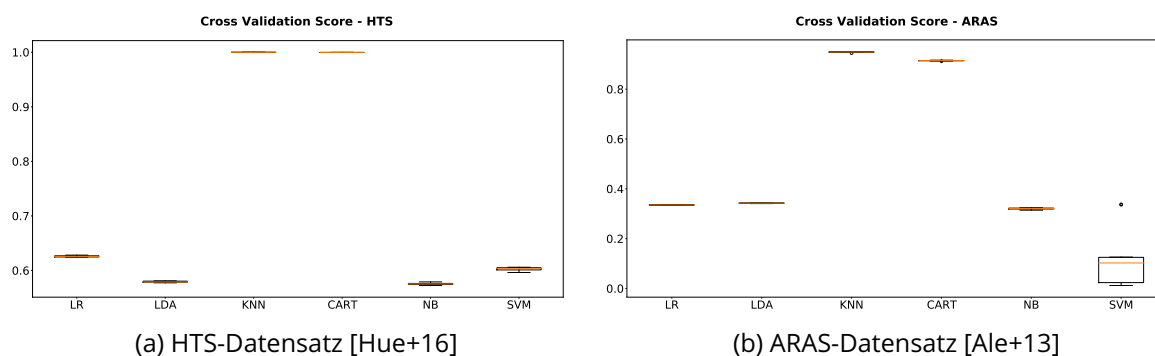


Abbildung 4.4: Geschätzte Genauigkeit der Klassifizierer für die zwei Datensätze

Für den HTS-Datensatz sehen die Werte insgesamt zwar besser aus, aber in Tabelle 4.1 ist zu sehen, dass es dort auch nur drei unterschiedliche Klassen gab, in welche die Daten zugeordnet wurden. Der ARAS-Datensatz hingegen besitzt 27 verschiedene Labels(Klassen) und ist damit realitätsnäher, weshalb ab diesem Punkt nur noch dieser Datensatz weiter untersucht wurde. Zuerst wurden die gemessenen und mit Labels versehenen Daten einer Zeitspanne von sieben Tagen zum Lernen genutzt. Um herauszufinden, welcher der Algorithmen besser abschneidet, wurden mit Hilfe der Daten der KNN- und der CART-Algorithmus trainiert. Dafür wurden 67 % der Daten zum Lernen genutzt und die restlichen 33 % um Vorhersagen zu treffen und die Genauigkeit zu bestimmen. Die Ergebnisse sind in Abbildung 4.5 als Konfusionsmatrizen dargestellt, d.h. es werden vom Algorithmus prognostizierte Aktivitäten und die echten Aktivitäten gegenübergestellt, um zu veranschaulichen, wie gut jede Klasse jeweils vom Algorithmus

bestimmt wurde. Je dunkler die Einfärbung der Kästchen ist, desto mehr Werte wurden dort prognostiziert. In den Kästchen steht die Wahrscheinlichkeit der Bestimmung der jeweiligen Klasse. In der Konfusionsmatrix für den KNN-Algorithmus sieht man eine sehr gute Verteilung, da die dunklen Kästchen wie eine gerade Linie von der oberen linken Ecke in die untere rechte Ecke verlaufen und somit die vorhergesagten Klassen des Algorithmus mit den echten Klassen größtenteils übereinstimmen. In der Konfusionsmatrix des CART-Algorithmus sieht man, dass die Verteilung etwas verstreuter ist und auch die Kästchen weniger stark eingefärbt sind. Für den Zeitraum von sieben Tagen, ergab die Klassifizierung mit den zwei Algorithmen die folgenden Werte:

KNN-Algorithmus:

- **Precision:** 95 %
- **Recall:** 95 %
- **F1-Score:** 95 %

CART-Algorithmus:

- **Precision:** 92 %
- **Recall:** 89 %
- **F1-Score:** 90 %

Um zu verstehen, ob diese Werte gut oder schlecht sind, müssen zunächst die Begriffe erläutert werden.

- Die *Precision*, oder auch Genauigkeit, steht für die Genauigkeit einer prognostizierten Klasse in der Konfusionsmatrix. Diese gibt an, wieviel Werte als wahr positiv bestimmt wurden in Relation zu allen positiv bestimmten Werten des Labels. In der Konfusionsmatrix wäre das beispielsweise für das Label "Sleeping" der Anteil der Aktivitäten, die wirklich der Klasse "Sleeping" angehörten, von allen Aktivitäten, die als "Sleeping" klassifiziert wurden.
- Der *Recall* ist die Trefferrate des Algorithmus. Diese gibt den Anteil der richtig bestimmten Werte von alle existierenden richtigen Werten an. Bei einem geringen Recall, werden viele richtige Werte einfach nicht bestimmt und somit fehlen am Ende eventuell relevante Informationen.
- Der *F1-Score* ist das harmonische Mittel von Precision und Recall, also werden dort beide Werte gleich behandelt.

Dadurch erschließt sich aus den Werten des Durchlaufs für sieben Tage, dass der KNN-Algorithmus dem CART-Algorithmus in diesem Anwendungsfall in allen Punkten überlegen ist. Trotzdem sind bei beiden Algorithmen die Werte im Bereich um 90 %, was für den Fall der Aktivitätsbestimmung außerordentlich gut ist. Schaut man in die Konfusionsmatrix, wird allerdings ersichtlich, dass bei der Klassifizierung mit KNN so gut wie alle Klassen eine Erkennungsrate von über 80 % aufweisen, was beim CART-Algorithmus nicht gegeben ist. Daher wurde ab diesem Punkt nur der KNN-Algorithmus weiter untersucht. Deshalb wird im Folgenden kurz erläutert, wie dieser Algorithmus funktioniert.

k-Nearest-Neighbor

Beim KNN-Algorithmus geht es, wie der Name schon sagt, um die nächsten Nachbarn. Und zwar um die Nachbarn des Datenpunktes, welcher vom Algorithmus bestimmt werden soll. Bei KNN werden alle vom Lernprozess bekannten Daten als Punkte mit ihrem jeweiligen Label in ein Koordinatensystem eingetragen. Ein neuer Punkt mit einem unbekanntem Label wird auch in dieses Koordinatensystem eingetragen und anschließend werden seine k-nächsten Nachbarn überprüft, wobei für die Nähe der euklidische Abstand zwischen den zwei Punkten berechnet wird. Die am häufigsten vorkommende Klasse unter den k gewählten Punkten, wird dann als Klasse für den neuen unbekanntem Punkt gewählt. Die Wahl eines passenden Wertes für k entscheidet, wie gut die Klassifikation durch den Algorithmus wird. Der Wert für k sollte

ungerade sein, damit es kein Entscheidungsproblem durch einen Gleichstand von zwei Labels gibt. Ist der Wert von k zu niedrig, so neigt der Algorithmus zum *Overfitting*, d.h. es wird sich zu sehr an den spezifischen Datensatz angepasst und zukünftige Vorhersagen mit zusätzlichen Daten werden eventuell gar nicht mehr erkannt. Ist der Wert für k wiederum zu groß, so steigt die Fehlerrate beim prognostizieren unbekannter Werte, obwohl man beim Erhöhen von k bis zu einem gewissen Punkt eine stetige Steigerung der Genauigkeit feststellen kann. Die Wahl eines geeigneten Wertes k ist dementsprechend von großer Bedeutung.

In dieser Arbeit wurde der Wert $k = 2 * \sqrt{n}$ gewählt, wobei n die gesamte Anzahl an Daten im Lerndatensatz ist und k immer zur nächsten ungerade Zahl aufgerundet wird. Ebenso wurde eine Gewichtung der Datenpunkte vorgenommen, sodass Punkte mit einer geringeren Distanz zum unklassifizierten Punkt stärker gewichtet werden, da dies insgesamt zu besseren Ergebnissen führt. In Abbildung 4.6 sind die Konfusionsmatrizen für einen Zeitraum von 14 Tagen zu sehen, wobei bei Abbildung 4.6b einige Klassen vor dem Lernen zusammengefasst wurden. Insgesamt gingen die Werte für Precision und Recall in diesem Zeitraum etwas herunter, was daran liegt, dass dort neue Aktivitäten hinzugekommen sind, welche in den ersten sieben Tagen nicht aufgetreten sind. Bei der Zusammenfassung einiger Labels, wie beispielsweise *Sleeping* und *Napping* zu *Sleeping*, wurden etwas bessere Werte erreicht, als bei der Verwendung aller vorgegebenen Labels. Für einen Zeitraum von 14 Tagen wurden mit dem KNN-Algorithmus folgende Werte erzielt:

KNN-Algorithmus alle:

- **Precision:** 90 %
- **Recall:** 89 %
- **F1-Score:** 90 %

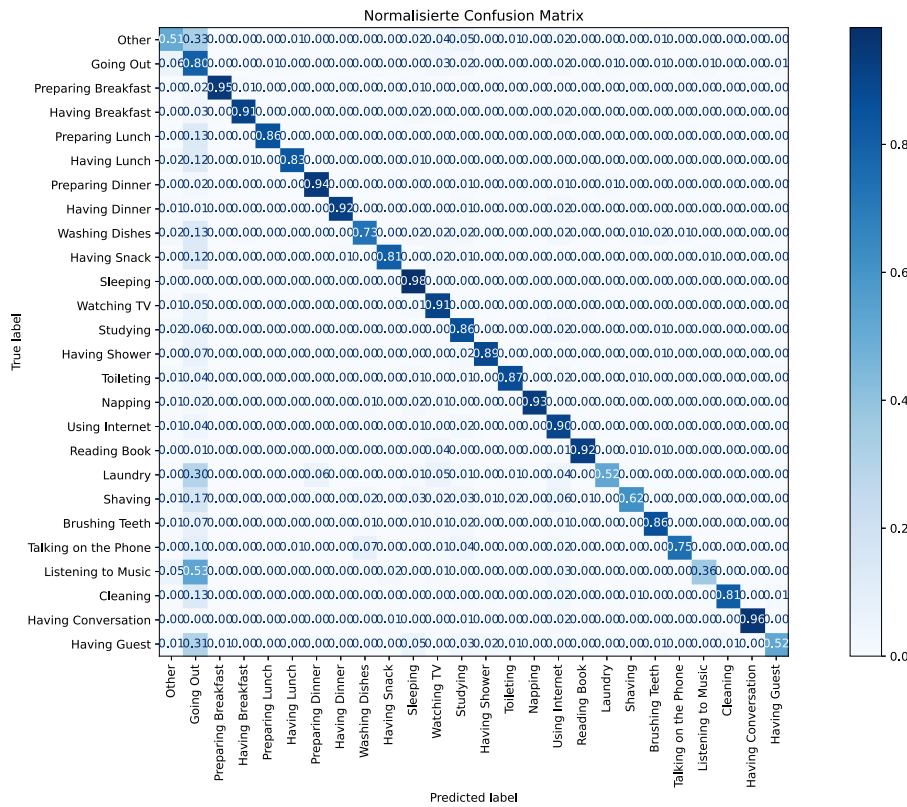
KNN-Algorithmus mit 20 Klassen:

- **Precision:** 90 %
- **Recall:** 90 %
- **F1-Score:** 90 %

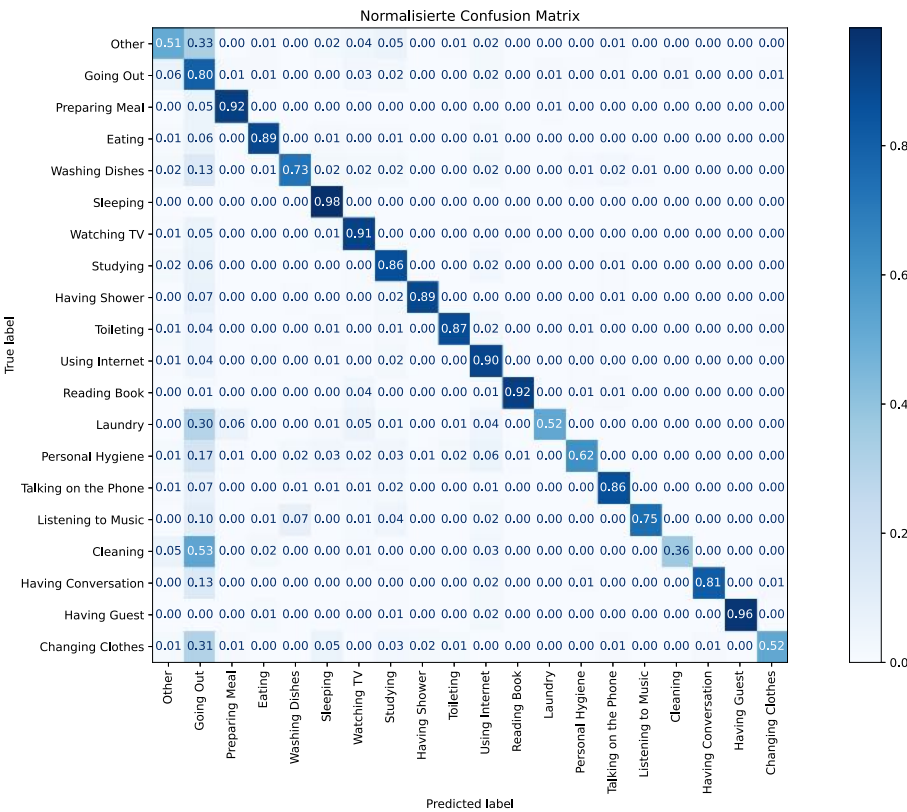
Obwohl die Werte etwas schlechter geworden sind, sind diese immer noch insgesamt in einem sehr guten Bereich, da die Verschlechterung nur durch die neuen Aktivitätsklassen zustande gekommen ist, welche durch ihr seltenes Vorkommen nicht besonders gut klassifiziert werden konnten. Daraus ergibt sich, dass eine Klassifizierung von Aktivitäten mittels ML zwar sehr gut funktionieren kann, aber zunächst eine große Menge an richtig gelabelten Daten zum Lernen braucht.

4.2.2.2 Aussagenlogische Klassifizierung

Neben der Klassifikation mittels ML könnten sich erkannte Aktivitäten auch mittels Aussagenlogik klassifizieren lassen. Das würde bedeuten, dass anhand fest definierter Regeln die Klassen von Aktivitäten bestimmbar sind. Ein großer Vorteil einer solchen Klassifizierung ist, dass sie nicht von gesammelten und gelabelten Daten abhängig wäre, die zur Klassifizierung mittels ML benötigt werden. Dafür müssen zuerst Regeln bzw. Aussagen erstellt werden, welche eine Zuordnung einer unklassifizierten Aktivität ermöglichen. Dieser Prozess ist allerdings weniger dynamisch als der ML-Ansatz und auf individuelle Nutzervorlieben, zum Beispiel ob morgens oder abends geduscht wird und ob in der Küche oder dem Wohnzimmer gegessen wird, kann nicht immer Rücksicht genommen werden. Ebenso ist die Zuordnung eines bestimmten Sensors zu einer spezifischen Aktivität nicht immer eindeutig, wie man auch in den zwei Datensätzen HTS und ARAS sehen kann [Hue+16; Ale+13]. Dort gibt es mehr oder gleich viele Labels als Sensortypen und wenn die Datensätze betrachtet werden, dann wird ersichtlich, dass viele der Sensortypen in mehreren Aktivitätsklassen aktiv waren, sodass dort keine eindeutige Zuordnung von Sensor und Label vorgenommen werden kann. Um so eine Zuordnung zu erreichen und dann eine Aktivitätsklasse anhand einer spezifischen Sensorwertänderung bestimmen zu



(a) KNN mit allen Klassen



(b) KNN mit reduzierter Anzahl an Klassen

Abbildung 4.6: Normalisierte Konfusionsmatrizen der Klassifikationen mit KNN für einen Zeitraum von 14 Tagen

können, benötigt es sehr viele Sensoren, die genau an den richtigen Stellen platziert sind, wie es beispielsweise Jiménez et al. in ihrer Arbeit durchgeführt haben [JS18].

Auch mit aussagenlogischer Klassifizierung sollen die Anforderungen aus Kapitel 3 erfüllt werden, d.h. eine sehr hohe Anzahl an Sensoren, welche an speziellen Stellen platziert werden müssen, sind kontraproduktiv bezüglich der Kosten und der Akzeptanz. Eine Möglichkeit wäre es die Aktivitäten nach Räumen in der Wohnung des Nutzers zu klassifizieren, also müssen die Sensoren mit Metainformation bezüglich ihrer Lokation ausgestattet und dann die Sensorwertänderungen eines Raumes untersucht werden. In Abschnitt 4.2.1 wird bereits erwähnt, dass Sensoren mit der Information abgespeichert werden, in welchen Raum sie sich befinden, damit diese Information von anderen AAL-Systemen genutzt werden kann. Daher kann dies auch direkt vom System genutzt werden, um herauszufinden in welchen Raum gerade Aktivitäten stattfinden und anhand dieser Information kann dieser Aktivität ein Label zugeordnet werden. Dafür muss es in jedem Fall eine fest definierte Menge an Aktivitätsklassen geben, wobei jeder Klasse ein Raum zugeordnet sein muss. Um diesem Ansatz noch ein gewisses Maß an Individualisierbarkeit zu verleihen, soll es möglich sein, dass der Nutzer den Sensoren selbst Räume zuweisen kann. Das System adaptiert dann diese Auswahl und bestimmt je nach Ort der Sensoraktivität die Klasse basierend auf den Präferenzen des Nutzers. Ebenso können sich aus zeitlichen Abläufen Informationen für die Klassifizierung ergeben, da beispielsweise nach dem Zubereiten einer Mahlzeit meist der Verzehr der Mahlzeit erfolgt.

In Abbildung 4.7 ist der Versuchsaufbau zur Erfassung einer Wohnung des ARAS-Datensatz abgebildet, auf dem sechs Bereiche zu erkennen sind. Der Versuchsaufbau enthält die gängigsten Komponenten einer Zweiraumwohnung und ist somit gut zum Aufstellen von generischen Regeln zur Aktivitätsbestimmung geeignet. Die Klassen der Aktivitäten pro Raum lassen sich wie folgt einteilen:

Küche:

- Frühstück zubereiten
- Mittagessen zubereiten
- Abendessen zubereiten
- Abwaschen
- Zwischenmahlzeit

Schlafzimmer:

- Schlafen
- Ankleiden

Flur:

- Wohnung verlassen
- Besuch empfangen

Bad:

- Duschen
- Auf die Toilette gehen
- Zähne putzen
- Rasieren

Wohnzimmer:

- Frühstück verzehren
- Mittagessen verzehren
- Abendessen verzehren
- Sich ausruhen
- Freizeitaktivitäten (Fernsehen, Lesen, ...)

Aus den gelabelten Daten ließen sich für die unterschiedlichen Testpersonen nur wenige übereinstimmende Muster bestimmen. Was sich feststellen ließ, war ein Zusammenhang zwischen der Zubereitung und dem Verzehr einer Mahlzeit, welcher immer unmittelbar auf die Zubereitung erfolgte. Ebenso fanden jegliche Freizeitaktivitäten fast ausschließlich im Wohnzimmer und alle Aktivitäten zur persönlichen Hygiene und Pflege meist im Badezimmer statt. Eine Unterscheidung der verschiedenen Aktivitäten im jeweiligen Raum, ist wiederum komplexer, da beispielsweise die Kontaktsensoren auf den Stühlen im Wohnzimmer schon Bestandteil von diversen Freizeitaktivitäten und dem Verzehr aller Mahlzeiten sind. Der Sensor in der Dusche ist andererseits nur beim Duschen ausgeschlagen, weshalb diese Aktivität auch beim ML-Ansatz eine gute Erkennungsrate hatte. Darum müsste man für eine ausführliche und genaue aussagenlogische Bestimmung eine Vielzahl an Sensoren platzieren, welche idealerweise nicht Bestandteil von mehreren Aktivitäten sein sollten. Der Verzicht auf einige Aktivitätsklassen ist mit

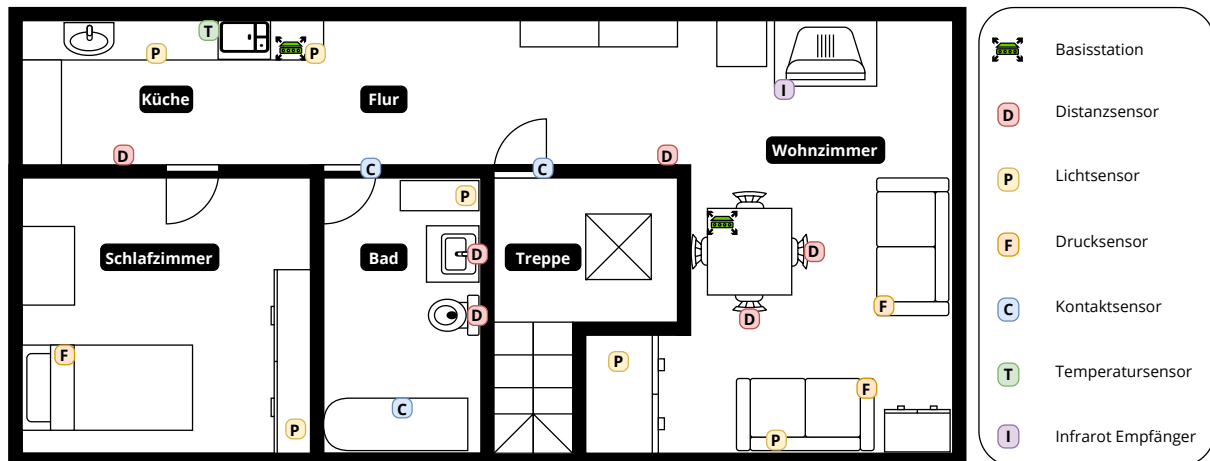


Abbildung 4.7: Aufteilung der Räume aus dem ARAS-Datensatz [Ale+13]

geringerem Aufwand verbunden, im Gegensatz zu einer Umsetzung mittels ausführlicher Sensorplatzierung, weshalb dieser Ansatz bevorzugt wurde. Der Nutzer sollte die Möglichkeit haben, dem System Informationen über vorhandene Räumlichkeiten zu übermitteln und welche Aktivitäten üblicherweise in dem jeweiligen Raum durchgeführt werden. Insofern die Sensoren die Information besitzen, in welchem Raum sie sich befinden, kann die Klassifizierung anhand der Sensoraktivitäten pro Raum geschehen und außerdem die Präferenzen des Nutzers berücksichtigen. Dies bedeutet allerdings, dass in jedem Raum Sensoren vorhanden sein müssen oder in Kauf genommen werden muss, dass bestimmte Aktivitäten nicht erkannt werden können. Anhand der Aktivitäten aus Kapitel 1.1.2 und den hier genannten Erkenntnissen aus der Analyse des ARAS-Datensatzes, sollen in dieser Arbeit für eine aussagenlogische Klassifizierung folgende Aktivitätsklassen an den jeweiligen Standorten betrachtet werden:

- Essen zubereiten (Küche)
- Essen (Esstisch)
- Persönliche Hygiene (Bad)
- Schlafen (Schlafzimmer)
- Freizeit (Wohnzimmer)
- Arbeiten (Arbeitszimmer)
- Duschen/Baden (Dusche | Badewanne)
- Ankleiden (Ankleide)
- Auf die Toilette gehen (Toilette)
- Haus verlassen (Haustür)

Spezielle Aktivitäten wie Duschen oder Fernsehen, können durch bestimmte Sensorik, wie einem Wassersensor oder Smart TVs, auch direkt erkannt werden, aber dafür müsste sichergestellt werden, dass diese Sensoren auch vorhanden sind. Diese Aktivitäten lassen sich allerdings auch in die vorher genannten Klassen einordnen. Diese Klassifizierung kann ohne einen bestehenden Datensatz durchgeführt werden und deshalb direkt von Beginn an verwendet werden.

4.2.2.3 Evaluation der Ansätze und Wahl des Algorithmus

Beide Ansätze haben Vor- und Nachteile. Es muss nun entschieden werden, welche Umsetzung die Bessere für das geplante System ist. Eine Gegenüberstellung, wie in Tabelle 4.2, zeigt, dass keiner der beiden Ansätze einen bedeutenden Vorteil hat und es letztendlich auch eine Frage des Kontextes ist. In einem Pflegeheim, wo viele Sensoren für viele unterschiedliche Personen befestigt werden können, wäre ein allgemeiner Regelsatz wahrscheinlich die bessere Wahl um Routinen zu erkennen, da sich die Einrichtung bzw. Umgebung nicht so häufig ändert und es für den ML-Algorithmus eventuell zu rechenaufwändig wird. In einem privaten Haushalt, wo sich öfters mal die Raumaufteilung oder ein Sensor-Standort ändern kann, wäre der dynamischere ML-Ansatz vorteilhafter. Da der aussagenlogische Ansatz das Kaltstartproblem⁵ umgeht, wäre dieser Ansatz optimal, solange es noch keine gelabelte Datenbasis gibt. Nachdem Daten gesammelt wurden und der Nutzer Aktivitäten eintragen konnte, kann ein ML-Modell erstellt werden. Aufgrund der Ergebnisse der Tests mit dem ARAS-Datensatz kann man den Lernvorgang ungefähr nach einer Woche starten. Sollte eine Genauigkeit von mindestens 80 % erreicht werden, könnte ab diesem Zeitpunkt der ML-Algorithmus verwendet werden. Da das Ganze auf einem ressourcenarmen System funktionieren soll, bieten sich dort vor allem ein ML-Algorithmus an, bei dem das Modell zu festen Zeiten mit neuen Daten angelernt und dies nicht regelmäßig zur Laufzeit berechnet wird. Glücklicherweise ist dies bei KNN der Fall, welcher bei der Untersuchung des ARAS-Datensatzes am besten abschnitt. Letztendlich soll also hier ein Hybrid-Lösung verwendet werden, um möglichst viele Vorteile beider Klassifikationsansätze vereinen zu können.

	ML-Klassifizierung	Aussagenlogische Klassifizierung
Vorteile	<ul style="list-style-type: none"> • Dynamisch • Ungewöhnliche Muster können erkannt werden • Erkennung von vielen Aktivitäten möglich 	<ul style="list-style-type: none"> • Transparent • Kein Kaltstartproblem • Ressourcensparend
Nachteile	<ul style="list-style-type: none"> • Benötigt gelabelte Daten zum Lernen • Rechenintensiv 	<ul style="list-style-type: none"> • Statisch • Erweiterung um neue Klassen ist sehr aufwändig

Tabelle 4.2: Vor- und Nachteile der Klassifizierungsansätze

4.2.3 Adaptive und adaptierbare Mechanismen des Systems

In der Anforderungsanalyse aus Kapitel 3 wurde festgestellt, dass das System adaptive und adaptierbare Mechanismen enthalten soll. Im Grunde genommen soll das System Präferenzen des Nutzers bei der Verarbeitung von Informationen berücksichtigen. Bei adaptierbaren Mechanismen gibt der Nutzer selbst seine Präferenzen an und das System ändert anhand dieser Informationen seine Funktionsweise, wie beispielsweise die Aktivierung eines Dunkelmodus, welche für den jeweiligen Nutzer eine veränderte Darstellung von Webseiten erzeugt. Bei adaptiven Mechanismen findet das System, anhand gesammelter Informationen über den Nutzer, dessen Präferenzen selbst heraus und passt daraufhin eigenständig betreffende Mechanismen an.

Im Abschnitt 4.2.2.2 über aussagenlogische Klassifizierung wurde erwähnt, dass der Nutzer die Möglichkeit haben soll, für jede vom System vorgegebene Aktivitätsklasse einen Raum auswählen zu können, wo der Nutzer diese Aktivität üblicherweise durchführt. Die Wahl eines Raumes

⁵Dem System ist über einen neuen Nutzer noch nichts bekannt.

beeinflusst den Algorithmus bei der Klassifizierung einer Aktivität, da die Aktivitäten nach der Sensoraktivität im jeweiligen Raum bestimmt werden. Daraus ergibt sich, dass hier die Wahl eines Raumes ein adaptierbarer Mechanismus des Systems ist. Ebenso soll der Nutzer auswählen können, ob die Klassifizierung nur mittels Aussagenlogik durchgeführt oder ob der Hybrid-Ansatz mit ML genutzt werden soll. Die Bestimmung mittels Aussagenlogik ist transparenter, da man die Regeln zur Bestimmung besser nachvollziehen kann als bei ML und somit erzeugt dieser Ansatz bei manchen Nutzern mehr Vertrauen in das System [HB15].

Als adaptive Maßnahme soll das System den Routinenplan des Nutzers eigenständig anpassen, indem Daten über die Aktivitäten des Nutzers gesammelt werden und der geplante Tagesablauf verändert wird, falls neue Muster entdeckt oder neue Aktivitätsklassen und Sensoren hinzugefügt werden. In dem System gibt es einerseits die aktuell stattfindenden Aktivitäten, welche anhand der aktiven Sensorwerte bestimmt werden können, um zur Laufzeit Informationen über den momentanen Status des Nutzers bereitzustellen. Andererseits gibt es noch einen hinterlegten Tagesablauf des Nutzers mit den Routinen, die bereits vom System erkannt wurden. Dies ist wichtig, um beispielsweise zu erfahren, was der Nutzer normalerweise zu einer gewissen Uhrzeit machen würde, wenn schon eine gewisse Zeit keine Sensorwerte mehr erfasst wurden. Damit können weitere Assistenzsysteme beispielsweise herausfinden, ob dem Nutzer etwas passiert ist oder ein Roboter könnte nach dem Nutzer suchen. Dafür müssen die Routinen in diesem Plan aktuell gehalten und vom System selbstständig angepasst werden, wenn Verhaltensänderungen des Nutzers erkannt werden. Dafür werden regelmäßig die Werte für die Endzeit der jeweiligen Routine aktualisiert, indem die Zeitwerte für neu erkannte Aktivitäten in die Routine integriert werden, wodurch im Laufe der Zeit ein auf den Nutzer angepasstes Zeitintervall entsteht. Dies passiert nur wenn die aktuell erkannte Aktivität mit der zuletzt erkannten Aktivität und dem dazugehörigen Sensor übereinstimmt. Der Prozess beginnt mit dem Erkennen einer Routine, dann wird entweder eine neue Routine in der Datenbank angelegt oder falls es diese Routine schon geben sollte, wird die Zeitabweichung geprüft und gegebenenfalls eine Benachrichtigung erstellt, bevor die Zeiten aktualisiert werden, was in Abbildung 4.8 zu sehen ist. Ob die Routine schon existiert, wird anhand der involvierten Sensoren überprüft.

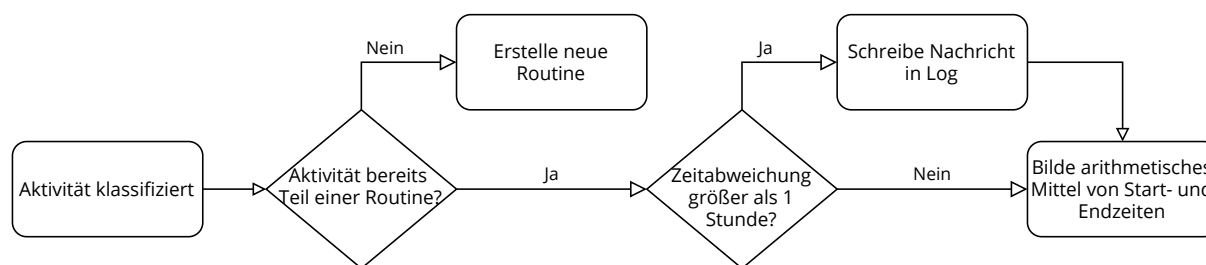


Abbildung 4.8: Ablauf der Routinenanpassung

4.3 Systemarchitektur Design

Für die konkrete technische Umsetzung muss noch die Architektur des Systems mit ihren Komponenten geklärt werden. Ein Anforderung der Analyse aus Kapitel 3 war, dass ein solches System möglichst kostengünstig sein soll, weshalb sich für den SH-Hub ein Einplatinencomputer, in diesem Fall ein Raspberry Pi 3, anbietet. Darüber hinaus bietet OH ein bereits vorkonfiguriertes Speicherkartenabbild für SD-Karten an, welches speziell für Einplatinencomputer ausgelegt ist und schon die meisten nötigen Schritte zur Installation vornimmt. Welche weiteren Vorzüge OH mit sich bringt, wurde bereits in Abschnitt 1.2.2 erläutert. OH wird mit einem Linux System bereitgestellt, auf dem auch die Services und Datenbanken laufen.

Die Services zur Klassifizierung von Aktivitäten und Benachrichtigung anderer AAL-Systeme

sind in Python umgesetzt, aufgrund der ML-Bibliothek *scikit-learn* und der Möglichkeit, REST-Services in Python zu entwickeln. Eine Umsetzung in Java wäre ebenfalls denkbar und auf dem OH-System umsetzbar, aber hauptsächlich wegen der genannten Bibliothek für maschinelles Lernen wurde Python gewählt. Als Standardvariante des Klassifizierungsalgorithmus ist die Hybrid-Lösung ausgewählt, welche eine aussagenlogische Klassifizierung durchführt bis das ML-Modell eine Genauigkeit von mindestens 85 % erreicht hat und anschließend die Klassifizierung von Aktivitäten übernimmt. Entscheidet sich der Nutzer dafür, eine reine aussagenlogische Klassifizierung zu wählen, wird ML zur Klassifizierung nicht verwendet. Eine durchgängige Klassifizierung mittels ML ist nicht möglich, da sonst die Kaltstartproblematik ungelöst blieb und keine angemessene Aktivitätserkennung durchgeführt werden kann. Die Kommunikation von OH mit den Services erfolgt über die dort integrierte *Rule Engine*, welche REST Aufrufe durchführen kann und die Services können ebenso über eine REST-Schnittstelle mit OH oder anderen AAL-Systemen kommunizieren.

Die Sensordaten werden in einer Zeitseriendatenbank persistiert, welche hier durch eine InfluxDB repräsentiert wird, da diese direkt bei OH als Persistenzdienst konfiguriert werden kann. Für alle weiteren Daten existiert eine relationale Datenbank, wie es in Abschnitt 4.2.1 beschrieben wurde. Der Benachrichtigungsservice kann die relationale Datenbank abfragen und der Klassifikationsservice kann beide Datenbanken abfragen.

Das Zusammenspiel der Komponenten ist in Abbildung 4.9 dargestellt und zeigt die vorangehend beschriebenen Teile des Systems. Die Sensoren geben Statusänderungen an den SH-Hub weiter, welcher diese in einer Zeitseriendatenbank abspeichert. Anhand der Werte aus dieser Datenbank klassifiziert der Klassifikationsservice die Aktivitäten, welche auch von diesem Service in die relationale Datenbank geschrieben werden. Der Nutzer hat über den SH-Hub die Möglichkeit Aktivitäten einzutragen, welche auf die dazugehörigen Sensordaten mit Hilfe des Zeitstempels abgebildet werden. Die gelabelten Daten werden dann zum Trainieren des ML-Modells genutzt, was einmal wöchentlich zu einer vom Nutzer gewählten Zeit oder sonntags um 01:00 Uhr nachts passiert. Das Training erfolgt auch nach dem Erreichen der 85% Genauigkeit, da sich die Muster des Nutzers im Laufe der Zeit ändern können. Der Benachrichtigungsservice bezieht Informationen aus der relationalen Datenbank und kann von anderen AAL-Systemen angefragt werden, um Informationen an diese weiterzugeben. Eine Authentifikation anderer AAL-Services ist noch nicht integriert, aber wäre eine sinnvolle Erweiterung. Das komplette System soll offline bzw. nur im lokalen Netzwerk funktionieren können.

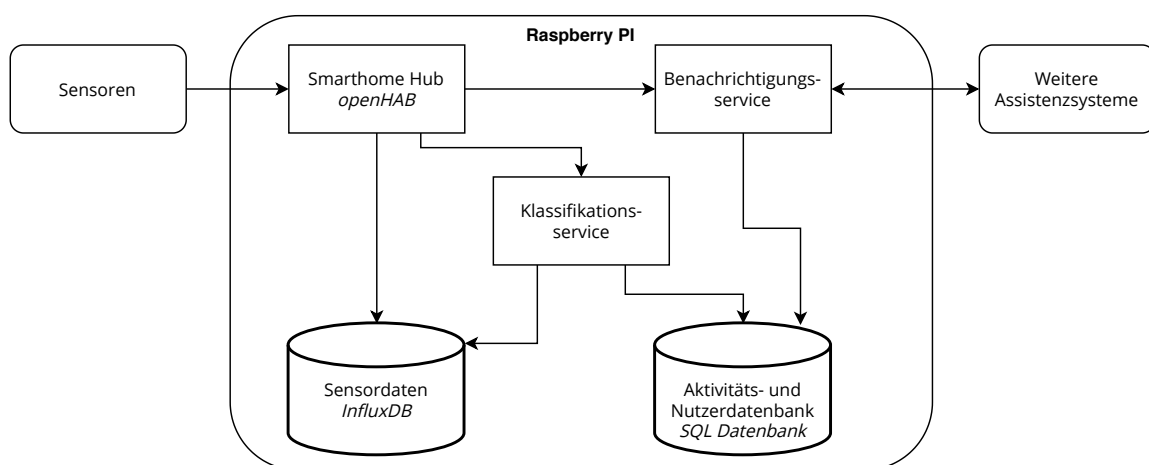


Abbildung 4.9: Übersicht der Bestandteile des Systems

4.4 Zusammenfassung

In diesem Kapitel wurde das technische Konzept für ein Aktivitätserkennungssystem mit adaptiven und adaptierbaren Mechanismen vorgestellt. Das System hat eine grafische Benutzerschnittstelle, um Nutzern einen Einblick in gespeicherte Daten zu geben und einige grundlegende Funktionen durchzuführen. Das Datenpersistierungskonzept wurde erläutert, in welchem die Sensordaten in einer Zeitreihendatenbank und die Aktivitäten und Nutzerdaten in einer relationalen Datenbank gespeichert werden. Für die Klassifizierung von Aktivitäten wurden unter Nutzung von Testdatensätzen die Ansätze mittels ML und Aussagenlogik untersucht und miteinander verglichen, was zu einer Hybrid-Lösung führte. Adaptive und adaptierbare Mechanismen des Systems wurden aufgezeigt und letztendlich die Systemarchitektur im Ganzen vorgestellt. Im nächsten Kapitel wird die Entwicklung eines Prototypen dieses Systems gezeigt.

5 Prototypische Umsetzung des Systems

Nachdem das technische Konzept erläutert wurde, wird in diesem Kapitel die Entwicklung und Konfiguration eines Prototypen des vorgestellten Aktivitätenerkennungssystems beschrieben. Dies beinhaltet die verwendete Hardware, sowie deren Einrichtung und die Konfiguration mit dem OH-System, die Erstellung der Persistenzdienste, sowie die Implementierung der Services und der grafischen Ausgabe der Daten.

5.1 Verwendete Hardware

Eine der Anforderungen aus Kapitel 3 betrifft die Hardware des Systems: den Kostenfaktor. Das System sollte die maximalen Kosten von 750 € nicht überschreiten und einen SH-Hub inklusive einiger Sensoren zur Aktivitätsbestimmung enthalten. Es wurde daher für den Hub ein Raspberry Pi 3 Model B+ mit einer 16 GB SD-Karte verwendet, welche das komplette System inklusive Datenbanken und Skripten enthält. Um den Sensoren mit dem ZigBee-Protokoll ansprechen zu können, wird ein CC2531 USB Stick benötigt, welcher an den Raspberry Pi angeschlossen wird, um mit ZigBee-Sensoren direkt zu kommunizieren. Ohne diesen Stick bräuchte es den herstellerspezifischen Hub inklusive dessen Software, was eine intransparente Verarbeitung der Sensordaten zur Folge hätte.



Abbildung 5.1: Raspberry Pi mit CC2531 Koordinator und Bewegungsmelder

Für die Sensorik wurden Produkte der Firma *Aqara*¹ genutzt, da diese alle das ZigBee-Protokoll nutzen, energiesparend und für SH-Technologie günstig sind. Für die visuelle Ausgabe wurden Computer und Smartphones mit einem gängigen Webbrowsern verwendet.

¹<https://www.aqara.com/>



(a) Türkontaksensor

(b) Temperatur- und Feuchtigkeitssensor

Abbildung 5.2: Zwei der eingebundenen Sensoren in der Versuchsumgebung

5.2 Installation und Konfiguration des Smarthome-Hubs

Wie bereits erwähnt wurde, läuft der Prototyp inklusive OH-Software auf einem Raspberry Pi. OH wurde mittels des hauseigenen Projekts *openHABian* installiert, welches bereits ein vorkonfiguriertes OH als SD-Karten Abbild anbietet. Ebenfalls ist das openHABian Konfigurationstool enthalten, welches mehrere Optionen bereithält, um OH und das darunterliegende Linux System zu konfigurieren. Nachdem die SD-Karte mit dem openHABian-Abbild überschrieben wurde, kann diese in einen mit dem Internet verbundenen Raspberry Pi eingelegt werden, um die Installation und Erstkonfiguration zu initialisieren. Ein Vorteil dieser Installation ist, dass bereits Nutzer angelegt, Netzwerkeinstellungen vorgenommen und das Java SDK installiert werden. Für die Ersteinrichtung muss das System mit dem Internet verbunden sein, aber nach der Installation kann die Internetverbindung auch getrennt und der Raspberry Pi direkt über eine Kabelverbindung angesprochen werden. Der Nutzer hat nach der Installation die Wahl zwischen mehreren grafischen Benutzungsschnittstellen, wobei in dieser Arbeit hauptsächlich das *Paper UI* genutzt wurde, welches die gleichen Möglichkeiten wie die anderen Schnittstellen bietet. Über das Paper UI wurden alle nötigen Bindings, Persistenzdienste und weitere Komponenten installiert.

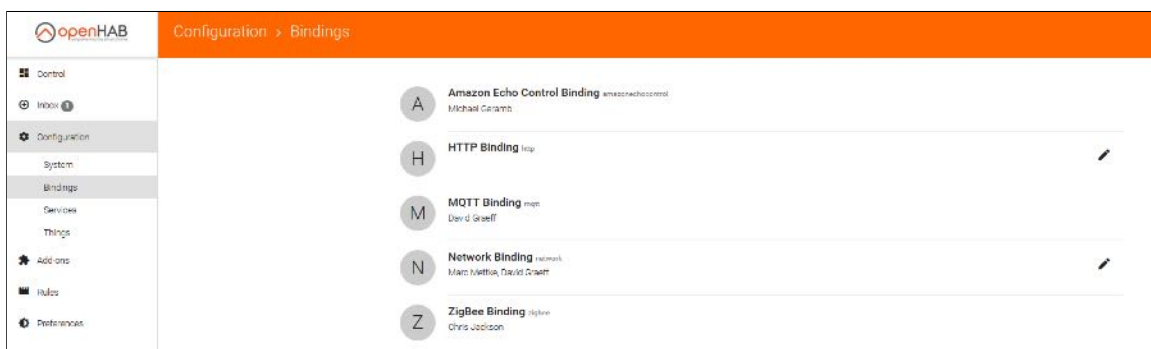


Abbildung 5.3: Paper UI Ausschnitt des Systems mit installierten Bindings

Einrichtung der Sensorik

Für die im Prototyp genutzten Sensoren wurde das ZigBee Binding benötigt, da mit diesem Binding die Sensoren erkannt und für das System nutzbar gemacht wurden. Dafür wurde zuerst der CC2531 USB Stick als Koordinator eingerichtet, d.h. mit der korrekten Firmware ausgestattet und über das ZigBee Binding als Thing konfiguriert. Der Koordinator steuert das Netzwerk und erkennt Sensoren mit dem ZigBee Protokoll, welche sich im Pairing-Modus befinden, sodass kein herstellerspezifisches Gateway benötigt wird. Je nach Konfiguration des Systems müssen die Sensoren manuell hinzugefügt werden oder sie werden automatisch hinzugefügt, sobald der Koordinator sie erkennt. Bereits hinzugefügte Geräte werden im Binding gespeichert, aber nicht im Koordinator, sodass dieser keine Sensordaten enthält, falls er ausgetauscht werden sollte. Die Kanäle, der als *Things* hinzugefügten Sensoren, wurden, wie in Abschnitt 1.2.2 beschrieben, als *Items* eingerichtet, wobei es nur die zwei Kanaltypen *Switch* und *Number* gab.

```
1 Switch Door_Sensor_1 "Door open" <door> {
2   channel="zigbee:device:49c320d0:00158d00045ce892:00158D00045CE892_1_switch"
3 }
```

Programmcode 5.1: Beispiel für ein Item des Prototypen

Einrichtung der Services

Um die Services nutzen zu können, wurde Python 3.7 inklusive der benötigten Bibliotheken auf dem System installiert. Für die Services sind die wichtigsten Bibliotheken *scikit-learn*, welche alle Funktionen für maschinelles Lernen bereitstellt, und *Flask*, welches den Aufbau einer REST-Schnittstelle und die Umsetzung des Routings für die visuelle Ausgabe ermöglicht. Die Skripte sind als Service im Betriebssystem des Raspberry Pi registriert und werden beim Start des Systems ausgeführt, sodass diese permanent im Hintergrund laufen und der Nutzer diese nicht manuell aktivieren muss. Zusätzlich zu den Python-Skripten wurde die OH REST API und die Rule Engine installiert, sodass einerseits über REST-Aufrufe die Daten des OH-Systems von den Skripten abgerufen und andererseits mit Hilfe der Regeln bei Sensorereignissen die REST-Schnittstellen der Python-Skripte aufgerufen werden können. Ein einfaches Beispiel für eine Regel, welche einen Request ausführt, sobald ein Türkontakt-Sensor seinen Wert ändert, ist in Abbildung 5.2 zu sehen.

```
1 import org.openhab.core.library.types.*;
2 import org.openhab.core.persistence.*;
3
4 val String time
5
6 rule "Door_Sensor_1 Start"
7 when
8   Item Door_Sensor_1 changed from OFF to ON
9 then
10  var String sensor = sendHttpRequest("http://localhost:5001/
11    classify_start/?sensor=Door_Sensor_1")
12  time = transform("JSONPATH", "$.time", sensor)
13  logInfo("Rules", "Aktivitaetsbeginn von Door_Sensor_1 erkannt: "+sensor)
14 end
15 rule "Door_Sensor_1 Ende"
16 when
17   Item Door_Sensor_1 changed from ON to OFF
18 then
19   var String sensor = sendHttpRequest("http://localhost:5001/
20     classify_end/?sensor=Door_Sensor_1&start="+time)
21   logInfo("Rules", "Aktivitaetsende erkannt: "+sensor)
22 end
```

Programmcode 5.2: Rule Engine Beispiel

Ebenso wurde die *JSONPath Transformation* installiert, damit es der Rule Engine möglich war JSON Strings auszulesen. Über den eingerichteten Persistenzdienst, also in diesem Fall InfluxDB, kann auch auf die Werte der Items zugegriffen werden. In der abgebildeten Regel ist zu sehen, dass eine *localhost*-Adresse aufgerufen wird, um die Klassifizierung zu starten. Das bedeutet allerdings nicht, dass nur durch den Raspberry Pi ein Aufruf der Dienste möglich ist, denn die Services sind so eingestellt, dass jeder im lokalen Netzwerk die Schnittstellen ansprechen kann. Zugriffe von außerhalb sind geblockt, da der Nutzer oder pflegende Personen über Geräte im lokalen Netz Zugriff auf die grafische Benutzeroberfläche haben sollen, aber Unbefugte von außerhalb nicht. Eine Authentifizierung und Zugriff über eine Remote-Verbindung waren in dieser Arbeit nicht vorgesehen und wurden daher nicht implementiert.

Einrichtung der Persistenzdienste

Wie im Konzept beschrieben, wurden eine InfluxDB als Zeitseriendatenbank für die Sensordaten und eine MariaDB als relationale Datenbank für alle weiteren Daten angelegt. Für OH wurde nur InfluxDB als Persistenzdienst registriert, da für die Interaktion mit OH nur die Sensorwerte relevant sind und dadurch die Rule Engine direkt auf die neuesten Sensorwerte zugreifen kann. Nachdem die Datenbank eingerichtet wurde, musste noch eine Persistenzstrategie für jedes Item festgelegt werden. Standardmäßig wird bei jeder Änderung ein Eintrag in die Datenbank geschrieben, aber hier wurde noch bewusst eine Strategie ergänzt, bei der minütlich die Sensorwerte eingetragen werden. Da Nutzer die Möglichkeit haben, selbst Aktivitäten einzutragen, sollen auch für Zeiträume, in denen keine Sensorwertänderung ausgelöst wurde, Daten für das ML-Modell verfügbar sein, um auch daraus Rückschlüsse ziehen zu können. Im Codebeispiel 5.3 ist eine Zuordnung der Strategien, welche selbst mit Hilfe von CRON²-Ausdrücken definiert werden, zu den jeweiligen Items zu sehen.

```

1 Strategies {
2     everySecond : "* / 1 * * * * ?"
3     everyMinute  : "0 * * * * ?"
4     everyDay     : "0 0 0 * * ?"
5 }
6
7 Items {
8     Door_Sensor_1 : strategy = everyMinute , everyChange
9     .
10    .
11    .
12    Humidity_Sensor_2 : strategy = everyMinute , everyChange
13 }
```

Programmcode 5.3: Persistenzregeln

Die relationale Datenbank ist eine MySQL-Datenbank bzw. eine MariaDB, welche ein Ableger von MySQL ist, mit dem Ziel weiterhin frei verfügbar zu bleiben. Die MySQL-Technologie wurde gewählt, da sie Teil des LAMP³-Stacks ist, welche gut geeignet für die freie Entwicklung von Webanwendungen ist. Die Datenbank wurde anhand des ER-Modells aus Abschnitt 4.2.1 erstellt und in dieser sind die Aktivitätsklassen, Sensoren, Zyklen und der Nutzer hinterlegt, welche bei der Einrichtung manuell direkt in die Datenbank eingetragen wurden. Aktivitäten und Abläufe bzw. Routinen werden während der Laufzeit vom System durch die Services oder Nutzereingaben ergänzt.

Für beide Datenbankdienste wurden extra Datenbanken für den jeweiligen Einsatzzweck erstellt und mit neuen Nutzern versehen, sodass die Nutzer mit Adminrechten nicht in den Services verwendet werden, um sich mit den Datenbankdiensten zu verbinden. Das dient der zusätzlichen Sicherheit und verhindert, dass die Services volle Kontrolle über alle Datenbanken

²Hintergrundprogramm zur zeitbasierten Ausführung von Prozessen in UNIX-Systemen

³Linux, Apache, MySQL, PHP/Perl/Python

auf dem System haben. Außerdem wurden beide Datenbanken als Services im System registriert, sodass diese bei einem Neustart des Raspberry Pi automatisch gestartet werden.

5.3 Klassifizierungsalgorithmus

Der Klassifikationsservice ist mit Python und der Erweiterung *Flask-RESTful* für Flask entwickelt worden. Wie in den Anforderungen in Kapitel 3 erläutert wurde, soll das System interoperabel sein, weshalb REST-Schnittstellen erstellt wurden. Die Schnittstellen liefern als Antwort auf Requests ein Objekt mit JSON-Body zurück, sodass jedes andere System, welches HTTP-Requests senden kann, im lokalen Netz mit dem Aktivitätenerkennungssystem kommunizieren kann.

Es gibt hauptsächlich zwei Schnittstellen, die für die Klassifikation von Aktivitäten relevant sind, nämlich "http://<hostname>:5001/classify_start/?sensor=<itemname>" für den Beginn und "http://<hostname>:5001/classify_end/?sensor=<itemname>&start=<HH:mm:ss>" für das Ende der Klassifikation. Über den eindeutigen Namen des Items und die Startzeit, die beim Aufruf des Klassifikationsbeginns ermittelt wird, lässt sich die Dauer einer Aktivität bestimmen. Im Code-Beispiel 5.2 wird die Schnittstelle für den Aktivitätsbeginn aufgerufen, sobald die Tür geöffnet wird, also wenn der Status des Items von *OFF* auf *ON* wechselt. Umgekehrt wird wiederum die Schnittstelle für das Ende aufgerufen und die Zeit übergeben, als das Ereignis begonnen hat. Anhand der Sensordaten und der Informationen über den Sensor, wird die Aktivität klassifiziert und in der relationalen Datenbank eingetragen. Wenn die klassifizierte Aktivität, der zuletzt klassifizierten Aktivität mit diesem Sensor entspricht, so wird nur die Zeit des bereits bestehenden Eintrags aktualisiert. Die Standortdaten der Sensoren sind in der Datenbank zu finden und die regelbasierte Klassifizierung kann für jeden Sensor mit eingetragenen Standort genutzt werden. Das ML-Modell muss zuerst trainiert werden, bevor es Aktivitäten klassifizieren kann, weshalb es erst einige Nutzereingaben benötigt. Diese vom Nutzer eingetragenen Aktivitäten werden in einer CSV-Datei eingetragen, wobei ein Eintrag das Label, den Wochentag, die Uhrzeit in Minuten und alle dazugehörigen Sensordaten zu der Zeit enthält, wie in Code-Beispiel 5.4 dargestellt. Ein Array in der gleichen Form, bloß ohne Label, wird in das trainierte Modell eingegeben, um die Aktivität bestimmen zu lassen. Mittels der Rule Engine wird der Lernprozess angestoßen und das Modell als separate Datei auf dem System gespeichert. Dies hat den Vorteil, dass das bestehende Modell zur Laufzeit abgefragt werden kann, ohne das aufwändige Berechnungen durchgeführt werden müssen, welche das System belasten würden. Da der Raspberry Pi nur ein Einplatinencomputer ist, wurde hier auf eine geringe Komplexität der Berechnungen geachtet, um ein funktionierendes System zu gewährleisten. Durch neue Eingaben des Nutzers lernt das Modell regelmäßig dazu, wohingegen bei der regelbasierten Klassifizierung durch den eingetragenen Standort des Sensors die Klassifizierung beeinflusst werden kann. Kurz nach dem Aufruf der Schnittstelle für das Ende der Klassifikation, wird auch schon der Eintrag in der Datenbank erstellt bzw. modifiziert. Da oft mehrere Sensoren direkt hintereinander ausgelöst werden können, muss die Klassifizierung schnell durchgeführt werden können, da sonst das Zusammenfassen bestehender Abläufe nicht korrekt funktionieren würde.

```
1 category,weekday,time,[Sensor Names]...
2
3 Essen zubereiten,0,1080,0.0,0.0,62.8, ...
4 Essen zubereiten,0,1081,0.0,0.0,62.8, ...
5 Essen zubereiten,0,1082,0.0,0.0,62.8, ...
6 ...
7 Außer Haus,6,872,1.0,0.0,53.1, ...
8 Außer Haus,6,873,1.0,0.0,53.1, ...
```

Programmcode 5.4: Auszug aus den Trainingsdaten

5.4 Visuelle Ausgabe

Ein Entwurf der grafischen Oberfläche des Systems wurde bereits in Abschnitt 4.1 vorgestellt. Mit Hilfe von Python und dem Microframework Flask wurden Backend und Routing realisiert und mittels Jinja wurden die Daten in HTML-Seiten eingetragen. Wie schon erwähnt, ist Flask ein Microframework, d.h. im Gegensatz zu anderen Frameworks ist es im Kern sehr einfach gehalten und jede Funktionalität, die man zusätzlich nutzen möchte, muss zunächst eingebunden werden. Dafür ist es beispielsweise nicht vorgegeben, welche Datenbank genutzt werden muss und im Gegensatz zu anderen Frameworks ist es recht schlank und ressourcensparend, da nur wirklich benötigte Bibliotheken genutzt werden.

Jinja ist eine Template-Engine, welche für Python entwickelt und optimiert wurde. Variablen aus dem Python-Code werden direkt an das Template übergeben und dort weiterverarbeitet. Dabei kümmert sich die Engine automatisch darum, dass die Seite vor Cross-Site-Scripting geschützt und somit sicher vor dem Einfügen boshafter Skriptcodes ist. Ebenso können einfache Kontrollstrukturen, wie Verzweigungen und Schleifen verwendet werden. In Code-Beispiel 5.6 ist dynamische Darstellung der verfügbaren Aktivitätsklassen mittels Jinja, anhand der aus der relationalen Datenbank ausgelesenen Daten zu sehen. Das Auslesen der Daten über eine MySQL-Query und der Aufruf der Template Engine ist in Code-Beispiel 5.5 zu sehen.

```

1 @app.route('/activity_log/')
2 def activity_log(name=None):
3     categories = []
4     # Open database connections
5     conn = helper_functions.connect_to_db()
6     cur = conn.cursor()
7     cur.execute(
8         "SELECT `name` FROM `activity_class` WHERE 1 ",
9     )
10    for (name) in cur:
11        categories.append(name[0])
12    conn.close()
13    # Return website
14    return render_template('category_page.html', categories=categories)

```

Programmcode 5.5: Auslesen der Datenbank und Routing zur HTML-Seite mittels Flask

```

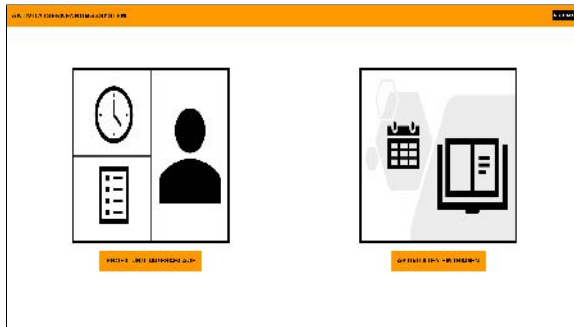
1 <html lang="de">
2     ...
3     <div class="flex-content">
4         {% for category in categories %}
5             <div>{{ category }}</div>
6         {% endfor %}
7     </div>
8     ...
9 </html>

```

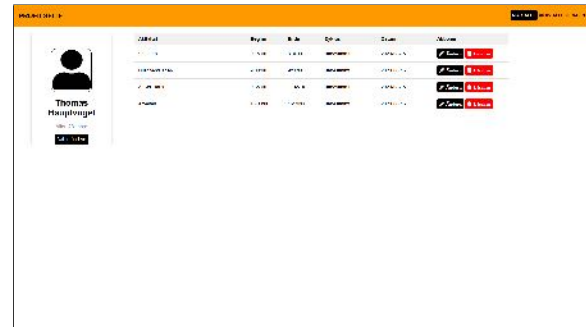
Programmcode 5.6: Jinja HTML-Template

Die Ausgabe des Prototypen beschränkt sich auf grundlegende Funktionen: eine Startseite, Profil, Tagesablauf und ein Formular zum Aktivitäten eintragen. Außerdem ist es noch möglich, eigene Profildaten zu ändern, sowie Routineneinträge zu ändern oder zu löschen. Es wurden wenige, aber dafür kontrastreiche Farben verwendet, welche sich am Farbschema von OH orientieren, um ein zusammengehöriges System zu suggerieren. Es wurden zusätzlich zu Textbeschreibungen noch Icons von *Font Awesome*⁴ verwendet, um das Auffinden von Funktionalitäten zu vereinfachen. Die Umsetzungen der verschiedenen Ansichten, dargestellt in einem Webbrowser, sind in Abbildung 5.4 zu sehen.

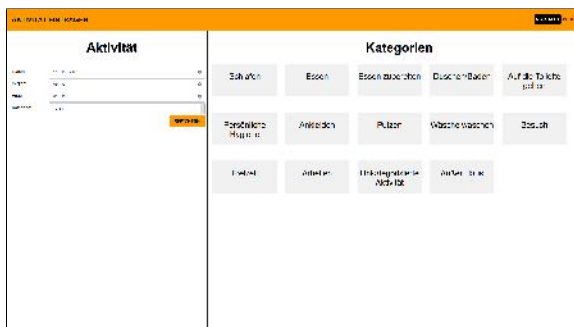
⁴<https://fontawesome.com/>



(a) Startseite



(b) Profil und Tagesablauf



(c) Aktivitätslogbuch



(d) Daten ändern

Abbildung 5.4: Visuelle Ausgabe des Systems in einem Webbrowser

5.5 Zusammenfassung

In diesem Kapitel wurde erläutert, welche Hardware für den Prototyp des Systems genutzt wurde und warum diese gewählt wurde. Auf der gewählten Hardware für den SH-Hub, dem Raspberry Pi 3, wurde das SH-Verwaltungssystem openHAB installiert. Es wurde gezeigt, wie man das System und die nötigen Bindings für die Sensoren mit dem ZigBee-Protokoll installiert. Die Sensordaten wurden mit dem Persistenzdienst für InfluxDB gespeichert, indem eine geeignete Strategie auf dem System eingestellt wurde, welche minütlich und bei Wertänderungen die Daten persistiert. Mittels der Rule Engine von OH wird der Service zur Klassifizierung von Aktivitäten aufgerufen, welcher mit Python und Flask umgesetzt wurde. Mit den gleichen Technologien wurde auch die grafische Oberfläche umgesetzt, über die grundlegende Funktionen, wie das Einsehen oder Eintragen von Aktivitäten durchgeführt werden können. Dieser Prototyp wurde von mehreren Versuchspersonen getestet, was im nächsten Kapitel betrachtet wird, inklusive der Auswertung der Ergebnisse.

6 Evaluation des prototypischen Systems

Mit dem entwickelten Prototypen aus Kapitel 5 wurden Versuche durchgeführt, um zu ermitteln, ob die Anforderungen aus Kapitel 3 erfüllt wurden. Diese Versuche wurden mit mehreren Personen aus verschiedenen Altersgruppen in unterschiedlichen Umgebungen durchgeführt, wobei der grundlegende Versuchsaufbau gleich blieb. In diesem Kapitel werden der Versuchsaufbau und die Durchführung näher beleuchtet, sowie die Ergebnisse der Versuche ausgewertet.

6.1 Aufbau und Ablaufplan des Versuchs

Um eine möglichst realitätsnahe Situation zu erzeugen, wurden für den Versuch mehrere verschiedene ADLs ausgewählt, welche von den Nutzern in der Versuchsumgebung durchgeführt werden sollten. Die Versuche wurden in privat genutzten Wohnungen durchgeführt, was die Versuche realitätsnaher machte. Dabei wurde aber immer darauf geachtet, dass die Aufteilung der Sensoren in allen Umgebungen gleich blieb. Nach dem Absolvieren der praktischen Aufgaben, sollten die Nutzer noch mit der grafischen Oberfläche des Systems interagieren und die grundlegenden Funktionsmöglichkeiten ausführen. Der genaue Ablauf für den Test der grafischen Oberfläche ist im Testplan in Tabelle 6.1 dargestellt. Bevor die Nutzer die Aufgaben durchführen sollten, wurden die Versuchsumgebungen mit Sensoren ausgestattet, wie in Abbildung 6.1 zu sehen ist. Die Platzierung der Sensoren wurde so gewählt, dass diese nicht sofort sichtbar sind, aber trotzdem die Erfassung nicht beeinträchtigt wurde. Dies hatte den Zweck, dass die Nutzer nicht absichtlich versuchen, durch die Sensoren erkannt zu werden.

Als praktische Aufgaben sollten die Versuchspersonen folgende Aktivitäten durchführen:

- Einen Tee kochen
- Den Tee trinken
- Den Müll entsorgen
- Zähne putzen
- Am Computer arbeiten oder Formulare ausfüllen
- Duschen oder Baden
- Fernsehen schauen

Die Reihenfolge der Durchführung war frei wählbar, da die Probanden die Aktivitäten möglichst natürlich ausführen und nicht durch einen festgelegten Ablauf eingeschränkt werden sollten. Gleichzeitig war es den Nutzern freigestellt, die Aktivitäten etwas abzukürzen, um das Risiko zu verringern, dass diese durch einen zu langen Test frustriert werden.

Zur Erfassung der Eindrücke durch die Nutzer wurde ein Fragebogen bereitgestellt, welcher Aufschluss über den technischen Hintergrund, die Gebrauchstauglichkeit der grafischen Oberfläche, sowie den Gesamteindruck der Nutzer vom System geben sollte. Neben der Erfassung von Alter und Geschlecht, wurde noch erfragt, wie die Probanden ihr technisches Interesse einschätzen würden, sowie ob und was für eine SH-Technologie sie besitzen. Die Nutzer hatten auch die Möglichkeit, Bedenken zu äußern, Verbesserungen vorzuschlagen und sie sollten einschätzen, wie wahrscheinlich es wäre, dass sie das System selbst nutzen würden.

Allgemeines Szenario	Testhypothese	Erfolgskriterien	Mögliche Probleme	Konkrete Aufgabe
Führe eine Aktivität in der Testumgebung durch und sieh dir das Resultat im Profil des Systems an.	Eine unauffällige Integration des Systems in den Alltag und eine sofortige transparente Ansicht der durchgeführten Aktivitäten, erhöhen die Akzeptanz des Systems durch die Nutzer.	Mindestens 90 % der Nutzer sollen in der Lage sein, nur mit der Hilfe des Systems, ihre durchgeführte Aktivität einzusehen. Die durchgeführte Aktivität ist im Profil sofort einsehbar.	Falsche Erwartung und Verständnis der Index- und der Profiseite.	Der Nutzer soll im Webbrowser die grafische Schnittstelle des Systems öffnen und seinen Tagesablauf öffnen, um dort die durchgeführten Aktivitäten einzusehen.
Trage eine vergangene Aktivität in das System ein.	Eine gute Gebrauchstauglichkeit und höheres Vertrauen wird erzielt, wenn der Nutzer mit geringem Aufwand die Aktivitäten auch selbst eintragen kann, welche zum Anlernen des Maschine Learning-Modells genutzt werden.	Mindestens 90 % der Nutzer sollen in der Lage sein, nur mit der Hilfe des Systems, eine Aktivität einzutragen. Es sollen keine Frustramente beim Eintragen einer Aktivität entstehen. Die selbst eingetragene Aktivität ist sofort im Tagesablauf einsehbar.	Falsche Erwartung und Verständnis der Index- und der Profiseite. Fehlende Orientierung und falsches Verständnis beim Wählen der Kategorie und der Zeitspanne. Toleranzgrenze bei der Zeit bis zur Rückmeldung des Systems zu niedrig.	Der Nutzer soll die Erkennung der Aktivitäten verbessern und daher mindestens eine Aktivität manuell eintragen. Dafür öffnet der Nutzer im Webbrowser das Aktivitätenlogbuch und trägt für einen selbst gewählten Zeitraum eine Aktivität ein.
Ändere die Kategorie einer bereits im Tagesablauf eingetragenen Aktivität.	Die Akzeptanz des Systems wird verbessert, wenn Nutzer die Möglichkeit haben, bereits eingetragene Aktivitäten zu ändern oder zu löschen.	Mindestens 90 % der Nutzer sollen in der Lage sein, nur mit der Hilfe des Systems, eine Aktivität zu ändern oder zu löschen. Es soll kein Frust oder Blocker entstehen. Änderungen sind sofort im Tagesablauf einsehbar.	Falsche Erwartung und Verständnis der Index- und der Profiseite. Fehlende Orientierung und falsches Verständnis beim Ändern der Kategorie und der Zeitspanne. Toleranzgrenze bei der Zeit bis zur Rückmeldung des Systems zu niedrig. Änderung oder Löschung wurde ungewollt durchgeführt.	Der Nutzer soll annehmen, dass eine Aktivität vom System nicht korrekt erkannt wurde. Diese soll er nun ändern oder löschen. Dafür öffnet der Nutzer im Webbrowser die Profiseite mit dem Tagesablauf und ändert oder löscht eine selbst gewählte Aktivität.

Tabelle 6.1: Testplan der grafischen Oberfläche des Systems



Abbildung 6.1: Die Versuchsumgebungen mit Sensorstandorten und ungefähren Sichtbereich der Bewegungsmelder

6.2 Versuchsdurchführung

Jeder Versuch wurde einzeln durchgeführt und mittels Audioaufnahme protokolliert. Wie in Abschnitt 6.1 beschrieben, wurde den Nutzern gesagt, dass sie die Aktivitäten in gewünschter Reihenfolge und ganz natürlich, wie im eigenen zuhause durchführen sollen. Wie die Aktivitäten durchgeführt werden, ist zwar für jedes Individuum anders, aber trotzdem wurde darauf geachtet, dass nicht nur eine bestimmte Gruppe an Personen befragt wird. Es wurden zu fast gleichen Teilen Männer und Frauen befragt, siehe Abbildung 6.2, wobei es etwas mehr Frauen als Männer und ungefähr 70 % der Testpersonen auch noch unter 30 Jahre alt waren. Dies entspricht zwar zum größten Teil nicht der primären Zielgruppe, aber es dafür wurde die se-

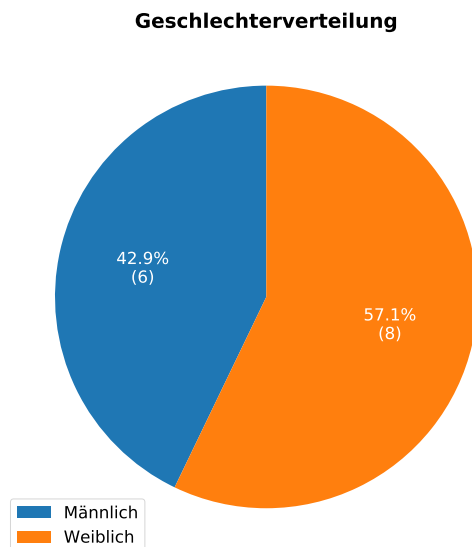


Abbildung 6.2: Versuchspersonen

kundäre Zielgruppe gut abgedeckt, was für diese Pilotstudie ausreichend sein soll.

Während der Durchführung des Versuchs hatte keiner der Probanden ein Problem mit der Lösung der praktischen Aufgaben. Es gab für jeden Probanden nur eine kurze Erklärung, wo sich jeder Raum in der Versuchsumgebung befindet und was für ein System getestet werden soll, aber nicht welche Sensorik in der Umgebung angebracht wurde. Dadurch sollte vermieden werden, dass die Nutzer sich unnatürlich verhalten, indem sie absichtlich probieren, von den Sensoren erkannt bzw. nicht erkannt zu werden. Außerdem wurden die Sensoren, wie im vorherigen Abschnitt beschrieben, unauffällig in der Umgebung platziert. Trotzdem probierten fast alle Probanden, die Sensoren zu entdecken, aber niemand von den Testpersonen schaffte es, alle Sensoren zu finden. Die Probanden verhielten sich alle natürlich in der Durchführung der Aktivitäten, wobei einige der Teilnehmer ihre Aktivitäten kommentierten und andere normale Gespräche nebenbei führten, wodurch sie weniger auf den Prozess der Durchführung achteten und die Aufgaben absolvierten, ohne weiter darüber nachzudenken.

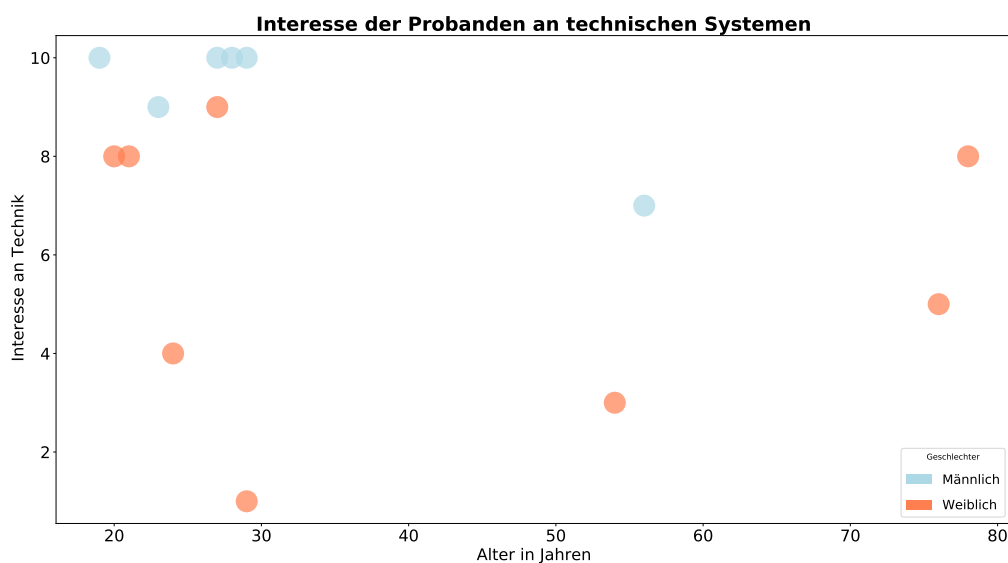
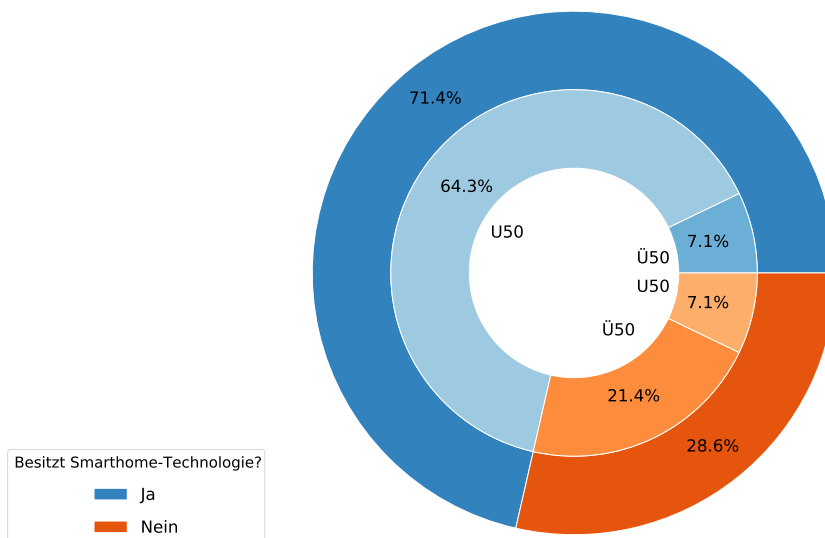
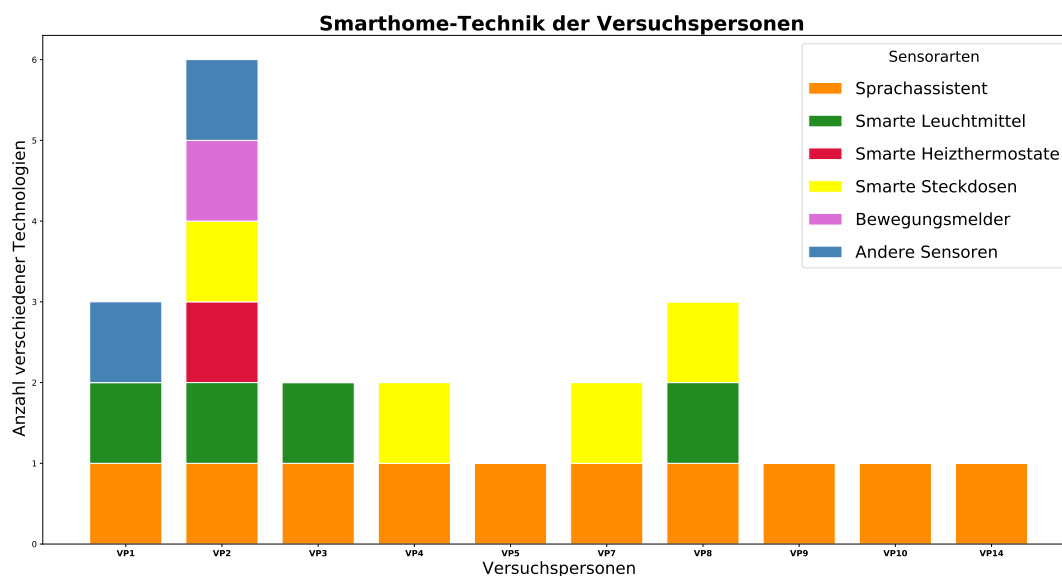


Abbildung 6.3: Technisches Interesse der Nutzer

Nutzerstatistik zum privaten Besitz von Smarhome-Technik



(a) Besitz von Smarhome-Technologie unterteilt in Altersgruppen



(b) Genutzte Sensorik der Nutzer

Das Interesse der teilnehmenden Personen an technischen Systemen reichte von fast gar keinem Interesse bis hin zu sehr hohen Interesse. Allerdings zeichnete sich bei den weiblichen Teilnehmern eine deutlich größere Varianz ab, als bei den männlichen Teilnehmern, siehe Abbildung 6.3. Am Alter konnte man das technische Interesse nicht feststellen, da die niedrigsten Werte eher im jungen und mittleren Alter erzielt wurden, sowie im hohen Alter die Tendenz auch wieder steigend war.

Wie in Abbildung 6.4a dargestellt, spielte jedoch das Alter beim Besitz von SH-Technologie eine Rolle. Denn obwohl sich fast alle Versuchspersonen, welche über 50 Jahre alt waren, mit einem mäßigen bis hohen technischen Interesse identifizierten, hatte nur eine Person dieser Gruppe solche Technologie zuhause. Ebenso gering wie der Anteil dieser Altersgruppe, welcher SH-Geräte besitzt, war der Anteil an unter 50-jährigen, welche keine SH-Technologie besitzen. Anzumerken ist, dass 80 % der Probanden, welche unter 50 Jahre alt waren, einen technisch orientierten Studiengang belegten oder einen technischen Beruf ausübten. In Abbildung 6.4b sind die Technologien zu sehen, welche von den Probanden mit SH-Technologien zuhause genutzt

werden. Besonders auffällig ist, dass die meisten Personen keine Bewegungsmelder oder andere binäre Sensorik besitzen, aber dafür hier jede Person, welche im Besitz von SH-Technologie ist, einen Sprachassistenten besitzt. Zusätzlich dazu sind auch smarte Leuchtmittel und Steckdosen häufig vertreten, aber im Schnitt beschränken sich die meisten Probanden auf zwei bis maximal drei verschiedene Technologien. Somit hatten zwar nur knapp 70 % der Probanden Erfahrung mit SH-Technologie aus ihrem privaten Umfeld, aber auch die restlichen 30 % konnten alle praktischen Aufgaben lösen, da sie durch das System und die genutzten Technologien nicht beeinträchtigt wurden.

Am Ende der Versuche sollten die Probanden noch die Funktionalitäten der grafischen Schnittstelle des Systems ausprobieren. Auch hier blieb die Reihenfolge der Durchführung und die Exploration der Oberfläche den Versuchspersonen überlassen. Die Oberfläche wurde in einem Webbrowser auf einem Laptop aufgerufen und die Bedienung erfolgte mit Maus und Tastatur. Nachdem die Probanden ihren Tagesablauf eingesehen, eine Aktivität gelöscht oder modifiziert und eine eigene Aktivität eingetragen hatten, sollten sie noch den Fragebogen, siehe Anhang A, ausfüllen. Die Ergebnisse aus der Befragung und eine Untersuchung der Genauigkeit des Systems, werden im nächsten Abschnitt präsentiert.

6.3 Interpretation der Ergebnisse

In diesem Abschnitt werden die Ergebnisse des Fragebogens ausgewertet, welcher von den Probanden nach dem Versuch ausgefüllt wurde. Damit wird ein Gesamteindruck der Nutzer vom System vermittelt und gezeigt wie gebrauchstauglich das System ist. Außerdem werden die Genauigkeit der regelbasierten und ML-Klassifizierung anhand der Versuchsdaten verglichen.

Bezüglich des Besitzes der SH-Technologie ist ersichtlich, dass im Versuch hauptsächlich Geräte verwendet wurden, welche nicht von den Probanden privat genutzt werden. Die mögliche Integration dieser häufig genutzten Technologien in das System wäre daher eine sinnvolle Erweiterung, um mehr Aktivitäten erfassen zu können, ohne Mehrkosten für den Nutzer zu schaffen. Allerdings gab es auch keine Notwendigkeit, direkt mit den Sensoren zu interagieren, da alle Aktivitäten im Haushalt auch ohne SH-Unterstützung durchführbar sind. In Abbildung 6.5 sind ein paar generelle Eindrücke zu sehen, die von den Probanden bezüglich des Systems erfasst wurden. Es ist ersichtlich, dass die Eindrücke größtenteils positiv waren und die meisten

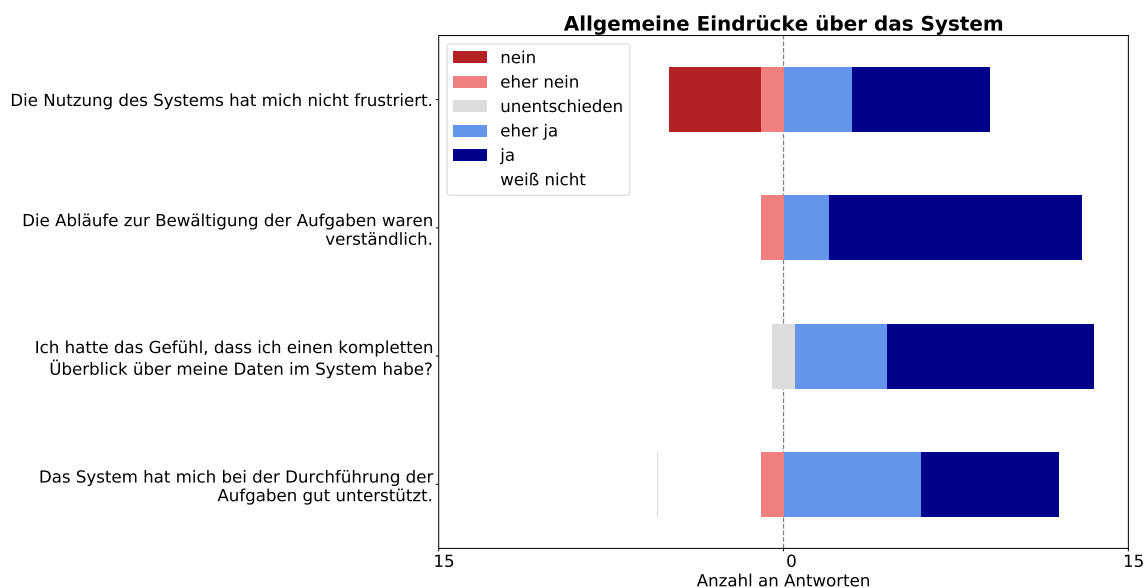
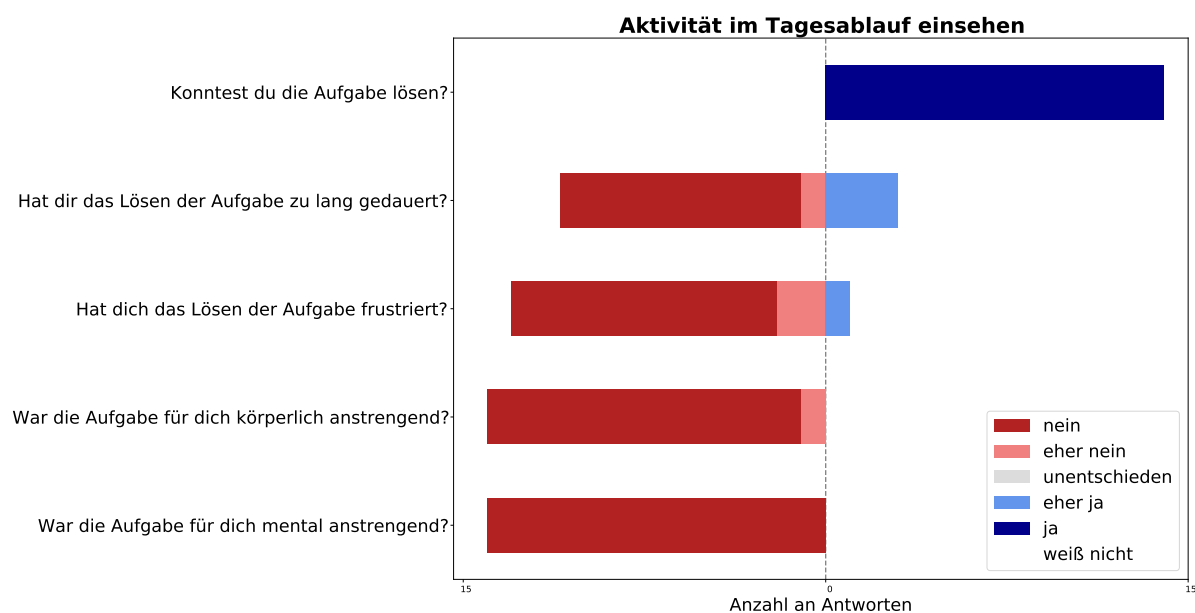


Abbildung 6.5: Allgemeine Eindrücke der Nutzer über das System

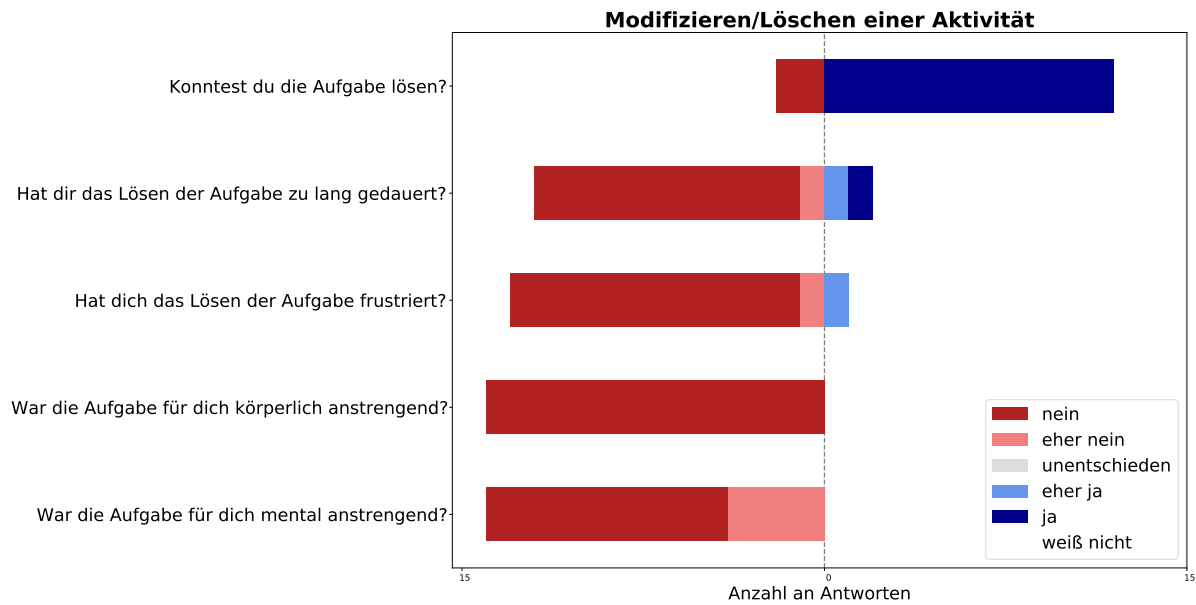
Nutzer sich vom System unterstützt gefühlt hatten. Die Abläufe waren für fast alle Probanden verständlich und das System vermittelte meist das Gefühl, dass es dem Nutzer auch tatsächlich alle gespeicherten Daten anzeigt. Anscheinend wurde aber ein Teil der Probanden durch die Nutzung des Systems frustriert. Der Grund zur Frustration ist aus den restlichen Antworten nicht direkt ableitbar, also lässt sich nur vermuten, dass die Probanden wahrscheinlich mit dem gesamten Ablauf des Versuchs oder dem Funktionsumfang des Systems nicht vollends zufrieden waren.

Bezüglich der Aufgaben, die von den Probanden mittels der grafischen Oberfläche gelöst werden sollten, gab es zwischen den Aufgaben einige Unterschiede in der Wertung. In Abbildung 6.6a sieht man, dass alle Probanden es schafften, im Tagesablauf ihre Aktivitäten einzusehen und nur ein Proband wegen der Aufgabe etwas frustriert war. Die Einsicht in den Tagesablauf ist für den Nutzer wahrscheinlich das relevanteste Feature, da es ihm erlaubt zu sehen, welche Aktivitäten das System erkannt hat und was anderen Assistenzsystemen oder Pflegepersonal potenziell über den Nutzer bekannt ist. Das Löschen oder Modifizieren, sowie das manuelle Eintragen von Aktivitäten soll dem Nutzer mehr Kontrolle über das System geben und somit das Vertrauen in das System steigern. In den Abbildungen 6.6b und 6.6c sieht man, dass diese Aufgaben nicht von allen Versuchspersonen gelöst werden konnten. Trotzdem kam bei den Probanden nicht wesentlich mehr Frustration und nur leicht höhere mentale Anstrengung auf, als bei der Einsicht in den Tagesablauf. Daher könnte man annehmen, dass es für die Nutzer eventuell eine niedrigere Relevanz hatte, diese Aufgaben lösen zu können. Denn obwohl alle Nutzer den Tagesablauf sehen konnten, kam dort Frust auf und es hat sogar einigen Nutzern zu lange gedauert.

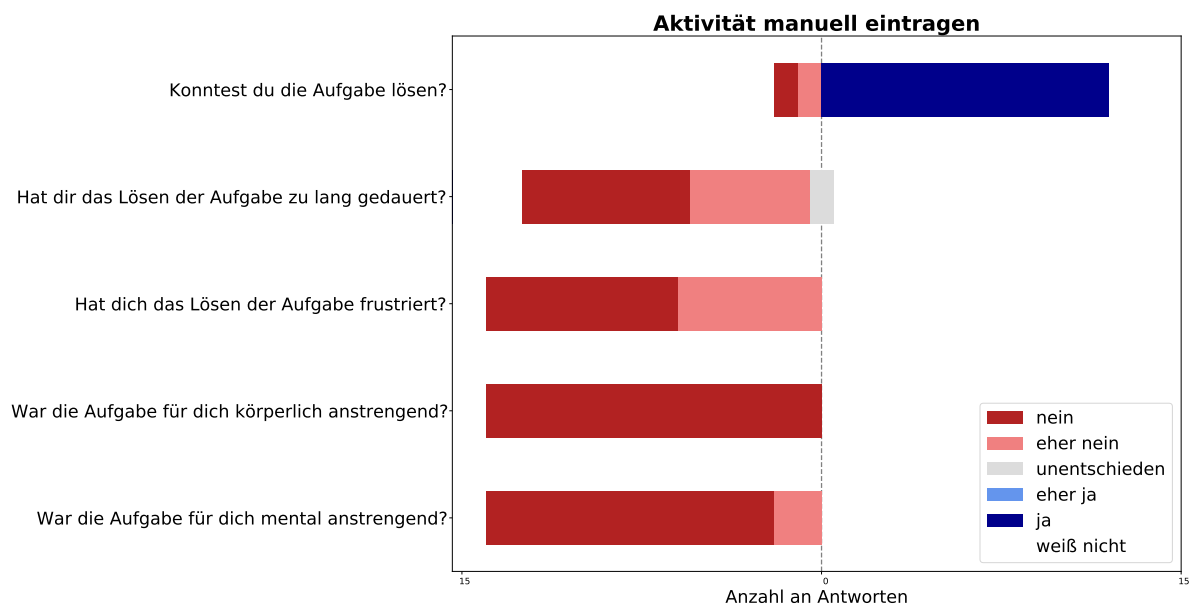
Im Vorfeld wurde den Probanden mitgeteilt, dass das Eintragen von Aktivitäten nur ergänzend ist und hauptsächlich dazu dient, dass das System dazulernen kann. Diese Aufgabe hat keinen sofort erkennbaren Mehrwert für den Nutzer, außer der eingetragenen Aktivität im Tagesablauf, weshalb es dort vermutlich weniger Frust bei Problemen, als bei der Modifizierung und dem Löschen von Aktivitäten gab. Für das Eintragen von Aktivitäten, gab es wiederum die meisten Verbesserungsvorschläge von den Nutzern. Das Erstellen eigener Aktivitätskategorien und das man in der Übersichtsseite auch Aktivitäten hinzufügen kann, waren beispielsweise Wünsche der Probanden. Von den älteren Probanden wurden sich auch andere Eingabemöglichkeiten gewünscht, da sie nicht sehr geschult im Umgang mit dem Computer waren. Insgesamt



(a) Nutzerwertung zum Einsehen des Tagesablaufs



(b) Nutzerwertung zum Modifizieren von Aktivitäten



(c) Nutzerwertung zum manuellen Eintragen von Aktivitäten

Abbildung 6.6: Ergebnisse des Fragebogens zu den Funktionalitäten der grafischen Oberfläche des Systems

samt wurden oft die einfache Bedienung und die übersichtliche Funktionalität der grafischen Oberfläche positiv hervorgehoben. Anhand der Eindrücke der Nutzer und den Antworten aus Abbildung 6.6 lässt sich sagen, dass die Oberfläche gut angenommen wurde und die meisten Probleme durch fehlende Computerkenntnisse entstanden waren, weshalb die Bereitstellung alternativer Schnittstellen angebracht wäre.

Neben den Eindrücken der Probanden, spielt auch die Genauigkeit der Aktivitätserkennung eine entscheidende Rolle. In Abschnitt 4.2.2.3 wurden schon Vor- und Nachteile der beiden Ansätze aufgelistet. Aufgrund des dort genannten Nachteils der ML-Klassifizierung, die notwendige Verwendung gelabelter Daten zum Lernen, konnte auch während der Versuche nur die

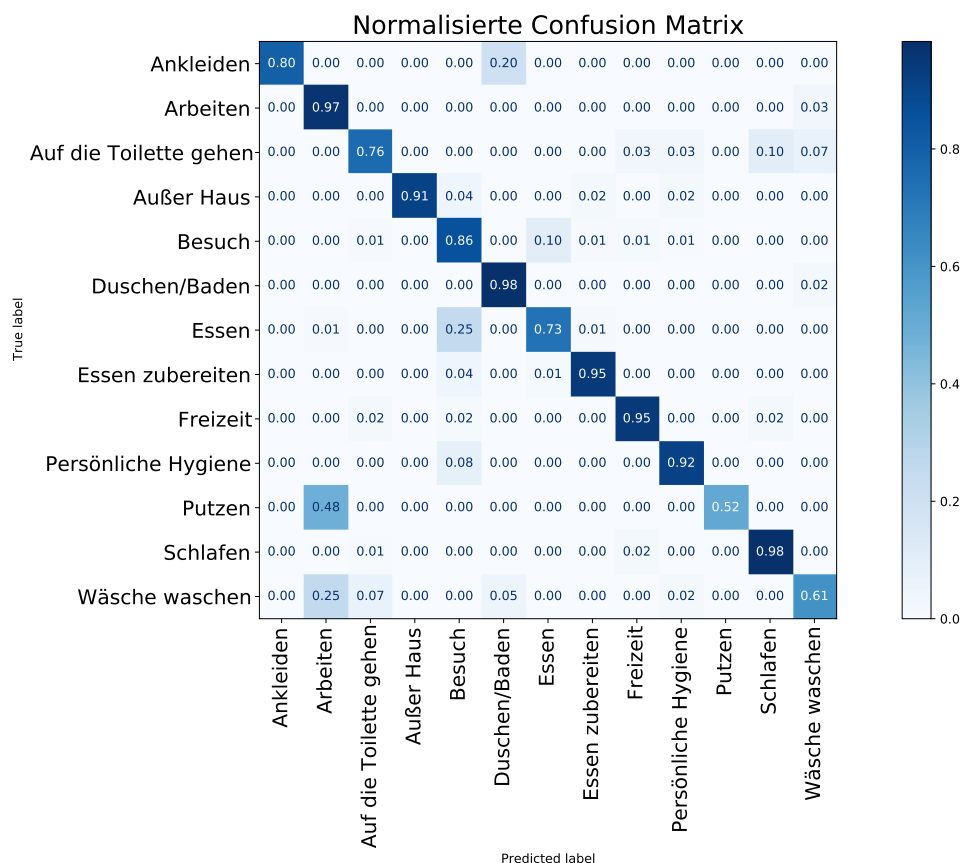


Abbildung 6.7: Konfusionsmatrix des ML-Modells zur Aktivitätenerkennung

aussagenlogische Klassifizierung verwendet werden. Neben den Versuchen wurden über einen Zeitraum von zwei Wochen Aktivitäten in das System eingetragen, während zwei Personen in Versuchsumgebung Nr.1 (Abbildung 6.1a) lebten. Mit diesen Daten wurde ein ML-Modell trainiert, sodass eine ähnliche Situation, wie in Abschnitt 4.2.2.1 beim ARAS-Datensatz entstanden ist. Wie auch für den ARAS-Datensatz, wurde hier eine Konfusionsmatrix erstellt, um die Genauigkeit des entstandenen Modells zu untersuchen. Zusätzlich wurden auch die erfassten Sensordaten aus den Versuchen an das Modell übergeben, um daraus die dazugehörigen Aktivitäten zu klassifizieren und diese mit den erkannten Aktivitäten aus der aussagenlogischen Klassifizierung und den eigentlichen Aktivitäten zu vergleichen. In der Konfusionsmatrix aus Abbildung 6.7 ist zu sehen, dass die meisten Kategorien häufig richtig erkannt wurden, dies ist ersichtlich an der Diagonalen. Bei manchen Aktivitäten, wie beispielsweise *Putzen* oder *Wäsche waschen* war die Zuordnung nur minimal besser als zu raten, da die Genauigkeit nur etwas größer als 50 % war. Das Modell des ARAS-Datensatzes, hatte eine ähnliche Genauigkeit und ebenso Probleme bei der Erkennung von Aktivitäten, wie Putzen und Wäsche waschen. Dort waren aber insgesamt noch mehr Aktivitätsklassen vertreten, sodass unser Modell aus Abschnitt 4.2.2.1 bei einer Reduzierung auf 13 verschiedene Klassen, eine ähnliche Genauigkeit, wie das Modell dieser Arbeit erreicht hätte. Wiederum wurden dort aber auch jede Sekunde Sensorwerte erfasst und es waren auch mehr Sensoren vorhanden, was in dieser Arbeit der Anforderung einer kostengünstigen Lösung widersprochen hätte.

Mit den Trainingsdaten aus zwei Wochen hat das Modell mit dem KNN-Algorithmus folgende Werte erzielt:

ML-Modell des Aktivitätenerkennungssystems:

- **Precision:** 93 %
- **Recall:** 93 %
- **F1-Score:** 93 %

Wie in Abschnitt 4.2.2.2 erklärt wurde, ist die Grundlage für die aussagenlogische Klassifizierung der Standort der Sensoren. In Abbildung 6.1a ist zu sehen, dass es in der Umgebung beispielsweise keinen Sensor im Schlafzimmer gab. Trotzdem konnte das ML-Modell die Aktivitäten *Schlafen* und *Ankleiden* erkennen. Der Vorteil gegenüber der regelbasierten Klassifizierung besteht hier darin, dass sogar Aktivitäten erkannt werden können, auch wenn es keine direkte Erfassung durch die Sensoren gibt. Die Aktivitäten aus den Versuchen mit den Probanden wurden mit dem ML-Modell ebenfalls korrekt klassifiziert, wobei es die gleichen Probleme, wie bei der regelbasierten Klassifizierung gab. Der Hauptgrund für falsche Klassifizierung waren fehlende Sensorwerte, welche vor allem bei den Temperatur- und Feuchtigkeitssensoren auftraten. Außerdem mussten zuerst viele verschiedene Aktivitäten eingetragen werden, um eine annehmbare Genauigkeit mit dem Modell zu erreichen und selbst dabei stach die Menge an gelabelten Daten zur Aktivität *Schlafen* hervor. Dies hängt damit zusammen, dass diese Aktivität täglich und lange am Stück durchgeführt wurde, weshalb es viele Daten dazu gibt. In Abschnitt 5.2 wurde erklärt, dass minütlich Sensorwerte persistiert werden, sodass man bei acht Stunden Schlaf auf 480 gelabelte Einträge kommt, wohingegen bei zehn Minuten Duschen nur zehn Werte eingetragen werden. Bei anderen Abläufen, wie der Einnahme von Medizin, dem Ankleiden oder ähnlich kurzen Aktivitäten, ist der Unterschied sogar noch größer.

Wenn ein Nutzer nicht genügend verschiedene Daten einträgt, wird das Modell nicht korrekt für alle Aktivitäten funktionieren. Daher wäre eine Hybrid-Lösung, wie schon in Abschnitt 4.2.2.3 angenommen, der beste Ansatz. Dadurch sind die Nutzer nicht dazu gezwungen, möglichst viele bzw. überhaupt Aktivitäten einzutragen und können das System trotzdem in vollem Umfang benutzen. Mit mehr Sensoren hätte die regelbasierte Klassifizierung, auch mehr verschiedene Aktivitäten erkennen können, wobei die Klassifizierung dann immer noch von einem korrekt gelabelten Sensor abhängt. Ein weiterer Vorteil des Hybrid-Modells besteht darin, dass neu installierte und dem System bekannte Sensoren sofort zur Klassifizierung genutzt werden können. Bei der Installation neuer Sensoren kann das ML-Modell die neuen Sensordaten nicht sofort korrekt einordnen, wohingegen die regelbasierte Klassifizierung anhand des Sensorstandortes sofort die Daten zur Klassifizierung verwenden kann. Bei der Erstellung des ML-Modells wäre es denkbar, für jede Aktivitätsklasse den F1-Score, sowie Precision und Recall in der Datenbank zu notieren. Es wären mehrere Ansätze möglich, wie ein hybrides Modell funktionieren könnte. Es könnten beide Klassifizierungen parallel laufen und wenn bei einer Aktivitätsklasse der F1-Score schlechter als beispielsweise 75 % ist, dann sollte stattdessen die regelbasierte Klassifizierung für diese Aktivität genutzt werden. In dieser Arbeit wurde nur das ML-Modell verwendet, sobald die Schwelle von 85 % bei der Genauigkeit überschritten wurde, aber das ist bei einer dauerhaften Nutzung nicht empfehlenswert, da bei neuen Sensoren das Modell nicht sofort auf die neuen Werte reagieren kann.

Auf lange Sicht ist es die beste Option, beide Ansätze parallel eine Klassifikation durchführen zu lassen und falls sich die klassifizierten Aktivitätsklassen voneinander unterscheiden, sollten die hinterlegten F1-Scores zu Rate gezogen werden. Wenn die regelbasierte Klassifizierung beispielsweise *Wäsche waschen* erkennt, aber das ML-Modell stattdessen *Arbeiten* erkennt, dann sollte die regelbasierte Klassifizierung Priorität haben, da der F1-Score für *Wäsche waschen* nur bei 61 % liegt, auch wenn der Wert für *Arbeiten* bei 97 % liegt. Beim F1-Score wird davon ausgegangen, dass Precision und Recall gleich gewichtet werden, daher kann als Schwellwert 75 %

gewählt werden, weil dann keiner der beiden Werte schlechter als 50 % sein kann. Auf diese Weise ist sichergestellt, dass immer eine Klassifizierung durchgeführt wird und wenn ausreichend gelernt wurde, auch außergewöhnliche Präferenzen berücksichtigt werden. Für das gesamte Modell wurde als Schwellwert 85 % gewählt, da es wünschenswert ist, dass fast alle Aktivitäten einen besseren F1-Score als 75 % haben. Da der Lernprozess wöchentlich angestoßen wird, wird auch erkannt, wenn sich die Qualität der Erkennung bei bestimmten Klassen verschlechtern sollte.

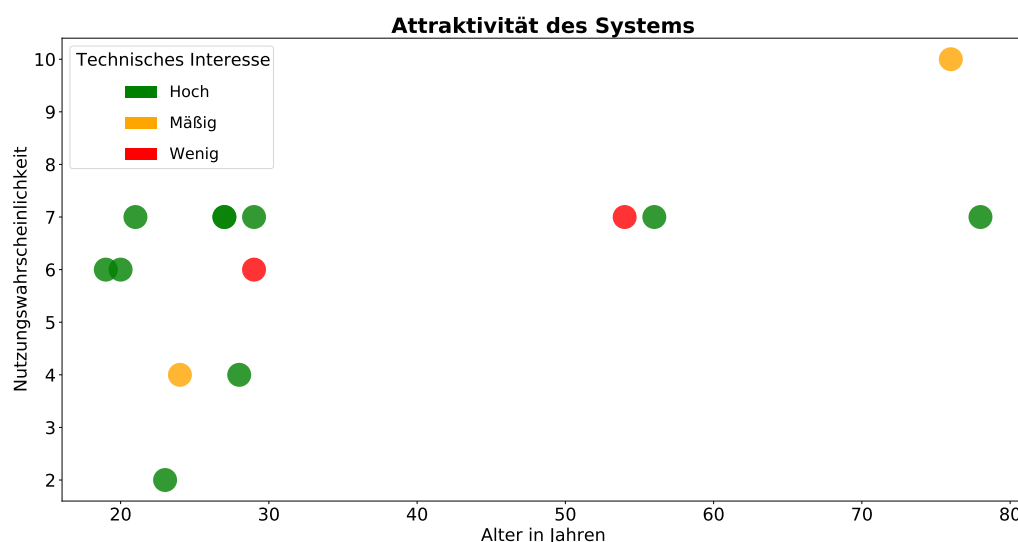


Abbildung 6.8: Nutzungswahrscheinlichkeit des Systems

Der Großteil der Probanden merkte im Fragebogen an, dass sie die schnelle und akkurate Erkennung der Aktivitäten gut fanden. Teilweise merkten die Versuchspersonen nicht einmal, wenn eine Aktivität nicht erkannt wurde. Es hat sich gezeigt, dass die Bereitschaft das System selbst zu nutzen, mit dem Alter ansteigt. In Abbildung 6.8 ist zu sehen, dass das technische Interesse eine nicht so große Rolle spielt, wie das Alter. Bei den jüngeren Probanden gab es eine wesentlich größere Varianz bei der Bereitschaft das System selbst zu nutzen, was an fehlender Notwendigkeit liegen könnte. Es gab nur eine Person, die komplett abgeneigt war, ein solches System zu nutzen. Die größten Bedenken der Nutzer, selbst bei Personen mit niedrigen technischen Interesse, lagen beim Datenschutz und der Sorge, dass Kriminelle die Informationen über den Tagesablauf ausnutzen könnten. Trotzdem lag die Bereitschaft zur Nutzung bei den meisten Probanden bei über 60 % mit einer steigenden Tendenz bei älteren Personen, welche die primäre Zielgruppe bilden. Um diese Tendenz zu bestätigen, müssten weitere Tests mit mehr Personen aus allen Altersgruppen durchgeführt werden. Im Gegensatz zu den jüngeren Probanden, sehen die älteren Probanden vermutlich einen konkreten für sie relevanten Anwendungsfall und Nutzen des Systems.

6.4 Zusammenfassung

Der Prototyp des in dieser Arbeit entwickelten Systems zur Erkennung von Aktivitäten wurde von mehreren Probanden aus verschiedenen Zielgruppen getestet. Dafür wurde der Prototyp in verschiedenen Testumgebungen aufgebaut, wobei darauf geachtet wurde, dass die Versuchsergebnisse nicht durch einen anderen Aufbau beeinträchtigt werden. In den Versuchen mussten die Probanden verschiedene praktische Aufgaben des täglichen Lebens absolvieren

und anschließend mit der grafischen Oberfläche des Systems interagieren. Die meisten Nutzer hatten schon Erfahrung mit SH-Technologie und hatten keine Probleme mit der Durchführung des Versuchs. Die einfache Bedienung und schnelle sowie akkurate Erkennung hatte einen positiven Eindruck bei den Probanden hinterlassen. Es stellte sich außerdem heraus, dass ein Hybrid-Ansatz bei der Klassifizierung die beste Option wäre, da die Nutzer sonst zwangsweise eine Zeit lang viele verschiedene Aktivitäten im System eintragen müssten, damit die Erkennung funktioniert. Das Eintragen zahlt sich aber auch trotz der regelbasierten Klassifizierung aus, denn mittels ML konnten Aktivitäten klassifiziert werden, welche nicht durch dazugehörige Sensorik abgedeckt waren. Im nächsten Kapitel werden die Erkenntnisse und Probleme dieser Arbeit diskutiert.

7 Fazit

In diesem abschließenden Kapitel der Arbeit werden alle wichtigen Aspekte noch einmal zusammengefasst vorgestellt. Im Anschluss werden die entstandenen Resultate dieser Arbeit diskutiert und bestehende Grenzen, sowie mögliches Potenzial aufgezeigt. Schließlich wird noch geklärt, ob die Forschungsfrage beantwortet werden konnte und welche zukünftigen Projekte mithilfe der hier erlangten Erkenntnisse entstehen könnten.

7.1 Zusammenfassung

In dieser Arbeit wurde das Konzept für ein System entwickelt, welches in einem mit SH-Technologie vernetzten Haushalt, die Aktivitäten der dort lebenden Personen erkennen kann. Dabei muss der Nutzer selbst keine Geräte am Körper tragen und sich auch nicht dem System anpassen, aber es besteht die Möglichkeit das System mit Informationen zu speisen, damit es von durchgeführten Aktivitäten lernen kann. Über eine REST-Schnittstelle können auch weitere Assistenzsysteme mit dem entwickelten System kommunizieren.

Dafür wurden am Anfang der Arbeit der Kontext und die dazugehörigen Grundlagen geklärt. Das System ist für allein lebende Senioren gedacht, welche länger in ihrem häuslichen Umfeld leben wollen und noch in der Lage sind, selbstständig ADLs durchzuführen. Als SH wurde in dieser Arbeit Ambient Assisted Living verstanden, also die Unterstützung von Senioren durch SH-Technologie, um diesen weiterhin ein selbstbestimmtes und sicheres Leben zu ermöglichen. Dort wurde gezeigt, dass sich die Technologien immer weiter entwickelten, um unaufdringlicher zu werden, sich besser in den Alltag integrieren und dem Nutzer assistieren. Diese Technologie unterstützt den Nutzer bei ADLs, also Aktivitäten des täglichen Lebens, welche alle Aktivitäten zur Selbsterhaltung und darüber hinaus zum sozialen Leben in einer Gemeinschaft beinhalten. Diese Aktivitäten wurden in dieser Arbeit auf Indoor-Aktivitäten beschränkt und auf einer hohen Ebene betrachtet, da es sonst zu viele fein-granulare Aktivitäten gäbe, welche nicht alle abgedeckt werden können. Es wurde erklärt, was Sensoren und Aktoren sind, wobei in dieser Arbeit nur Sensoren betrachtet wurden, welche mit dem ZigBee-Protokoll kommunizierten. Diese wurden mittels der Open-Source-Lösung openHAB verwaltet, welche es ermöglicht, dass die Sensoren ohne herstellerspezifischen Hub ausgelesen werden konnten. Außerdem wurde kurz der Roboter Loomo vorgestellt, welcher als Beispiel für ein Assistenzsystem diene, das potenziell mit dem hier entwickelten System kommunizieren könnte.

Anschließend wurde über ähnliche Arbeiten berichtet, welche auch ADLs erkennen wollten. Es wurde gezeigt, dass oft Video- und Tonaufnahmen genutzt wurden, wenn Aktivitäten mit Hilfe von Assistenzrobotern bestimmt wurden. Allerdings ging es in diesen Arbeiten oft um konkrete körperliche Aktivität und nicht um Aktivitäten des täglichen Lebens. Es wurden Ansätze mit tragbarer Sensorik und Standortdaten präsentiert, welche schon sehr gut funktionieren, aber oft sehr aufwändig im Aufbau waren. Die Notwendigkeit, ständig ein tragbares Gerät am Körper zu tragen, war auch ein Grund dafür, dass in dieser Arbeit gänzlich auf tragbare Sensorik verzichtet wurde. Die Ansätze mit einer geringen Anzahl an fest platzierten Sensoren, waren die relevantesten Ideen für diese Arbeit, wobei hier nur marktreife Sensorik verwendet und in einen Open-Source-SH-Hub integriert wurde.

Um ein Konzept für das System zu erstellen, wurden zuerst die Anforderungen analysiert. Dafür wurden die Bedürfnisse der betroffenen Nutzergruppen untersucht. Es wurde geklärt, dass die primäre Nutzergruppe aus den Senioren besteht, bei denen das System letztendlich installiert

wird. Die Personen aus dieser Gruppe erwarteten von einem AAL-System, dass es sie in ihrem häuslichen Umfeld bei ihren täglichen Aufgaben unterstützt. Ein solches System soll nur ergänzend zu menschlichen Pflegern sein und möglichst wenig kosten. Die weiteren Nutzergruppen bestehen aus Organisationen, welche in die Pflege älterer Personen involviert sind, also aus Pflegern bzw. Verwandten. Diese erwarten hauptsächlich Entlastung bei der Pflege und ein effizient arbeitendes System, was sich in bestehende Systeme integrieren lässt. Ebenso haben diese Gruppen den Wunsch, dass ein solches System im häuslichen Umfeld genutzt werden kann und dabei günstig bleibt. Der generelle Anwendungsfall des Systems wurde vorgestellt, welcher daraus besteht, dass die Nutzer ihr Leben normal weiterführen und währenddessen vom System beobachtet werden, damit es Aktivitäten im Tagesablauf des Nutzers erkennen kann. Der Nutzer kann diese dann einsehen oder modifizieren. Aus diesen Bedürfnissen und dem Anwendungsfall wurden mehrere Anforderungen an das System aufgestellt, welche wegweisend für das Konzept und die Entwicklung des Prototypen waren.

Es wurde ein Konzept erstellt, mit dem ein Aktivitätserkennungssystem umgesetzt und die Anforderungen beachtet werden konnten. Es wurde ein Entwurf für eine grafische Oberfläche des Systems konzipiert, welche dem Nutzer eine einfach handzuhabende Interaktion mit dem System ermöglichen sollte. Für die Sensor- und Aktivitätsdaten, die während der Nutzung des Systems entstehen, wurde ein Persistierungskonzept vorgestellt. Dieses sieht vor, dass Sensordaten in einer Zeitreihendatenbank und alle weiteren Daten in einer relationalen Datenbank gespeichert werden. Zur Klassifizierung von Aktivitäten, wurden bestehende Datensätze zum Trainieren von ML-Modellen genutzt. Dabei stellte sich heraus, dass der KNN-Algorithmus die geeignetste Lösung für das Erkennen von Aktivitäten ist. Weiterhin wurde erkannt, dass mittels der Sensorstandorte auch Aktivitäten bestimmt werden können. Diese Klassifizierung ist weniger dynamisch, benötigt aber keine gelabelten Daten und kann sofort angewandt werden. Es wurde geschlussfolgert, dass eine Hybrid-Lösung letztendlich die beste Idee ist, um die Nachteile des einen Ansatzes durch die Vorteile des jeweils Anderen auszugleichen. Mit diesen Informationen wurde eine Systemarchitektur erstellt, welche für die Entwicklung des Prototypen genutzt wurde.

Mit dem erstellten Konzept wurde ein Prototyp entwickelt, welcher auf einem Raspberry Pi 3 lief. Auf dem Gerät wurde OH installiert und die nötigen Bindings sowie Datenbanken eingerichtet. Es gab Bewegungsmelder, Kontakt-, Temperatur- und Feuchtigkeitssensoren von der Firma Aqara, welche über das ZigBee Protokoll mit Hilfe eines Koordinators mit dem System kommunizierten. Für die Sensoren wurde eine Persistenzstrategie erstellt, welche immer bei Sensorwertänderungen und auch minütlich die Sensorwerte in eine InfluxDB speicherte. Mittels der Rule Engine von OH wurden Sensorereignisse an den Klassifizierungsservice übermittelt, welcher die klassifizierte Aktivität in die relationale Datenbank eintrug. Sowohl die grafische Oberfläche als auch der Klassifizierungsservice wurden in Python unter Nutzung des Frameworks Flask implementiert. Um die Anforderung an das System zu überprüfen wurde im darauffolgenden Schritt der Prototyp mit Probanden getestet.

Für den Test des Systems wurde ein Plan aufgestellt, welcher Tests beinhaltete, die die Anforderungen überprüfen sollten. Dafür sollten die Probanden mehrere vorher festgelegte ADLs in selbst gewählter Reihenfolge durchführen und sich dabei möglichst natürlich verhalten. Anschließend sollten die Versuchsteilnehmer mit der grafischen Oberfläche des Systems interagieren und ihren Tagesablauf einsehen, eine Aktivität eintragen und eine Aktivität löschen oder modifizieren. Hauptsächlich waren unter den Probanden junge und technisch interessierte Personen, aber es konnten auch einige Probanden, welche über 50 Jahre alt waren oder kein Interesse an Technik hatten, getestet werden. Es bestätigte sich, dass eine Hybrid-Lösung zur Klassifizierung, die beste Option ist, da sehr viele Nutzeraktivitäten eingetragen werden mussten, damit das System eine gute Erkennung mittels ML gewährleisten kann. Diese Erkennung bietet vor allem den Vorteil der Dynamik und Adaptivität, aber hat mit dem Kaltstartproblem zu kämpfen, welches durch die regelbasierte Klassifizierung ausgeglichen werden kann. Insgesamt

samt waren die Probanden sehr zufrieden mit der Erkennung und der Oberfläche, hatten aber Sorge, dass ihr Tagesablauf von Kriminellen ausgelesen werden könnte. Bei den älteren Probanden zeigte sich trotzdem eine größere Bereitschaft das System zu nutzen, weil sie vermutlich einen direkten Nutzen im System für sich sehen können, welcher bei den jüngeren Probanden noch fehlt.

7.2 Diskussion

In diesem Abschnitt werden die Stärken und Schwächen des Systems gegenübergestellt und welches Potenzial noch nicht ausgeschöpft wurde. Dafür werden einige relevante Aspekte dieser Arbeit genauer betrachtet. Es wird erklärt, wie diese gelöst werden konnten, welche Probleme es gab und wie man diese durch Erweiterungen umgehen könnte.

Klassifizierung von Aktivitäten. Die Kernthematik dieser Arbeit war es, ein System zu erschaffen, welches den Nutzer unauffällig während seines Alltags beobachtet und dessen ADLs erkennen sowie klassifizieren kann. Dafür wurden zwei Ansätze verwendet, die Klassifizierung mittels ML und mittels vorher definierter Regeln. Die regelbasierte Klassifizierung basiert auf den Standorten der involvierten Sensoren. Aus der Analyse bestehender Datensätze hat sich ergeben, dass bestimmte Aktivitäten fast immer an den gleichen Orten ausgeführt werden. Dadurch kann jeder mit einem Standort gelabelte Sensor sofort zur Klassifizierung genutzt werden, womit das Kaltstartproblem, also das anfängliche Fehlen nutzerbezogener Daten zum Lernen, beim ML-Ansatz, ausgeglichen werden konnte. Die Klassifizierung durch die eingetragenen Standorte ist auch von den Nutzern besser nachvollziehbar und es ist einfacher möglich, die Klassifizierung um gewünschte Aktivitäten zu erweitern, indem ein Standort in den Regelsatz aufgenommen und dem Sensor hinzugefügt wird. Trotzdem werden individuelle Besonderheiten des Nutzers von dieser Klassifizierung nicht komplett beachtet, da die Regeln starr bleiben. Das ML-Modell beachtet die Präferenzen des Nutzers bei der Durchführung von ADLs. Allerdings muss der Nutzer dafür im Vorfeld genügend verschiedene Aktivitäten in das System eingetragen haben, um eine ausreichende Genauigkeit zu erreichen. Wie viele Aktivitäten eingetragen werden müssen, ist nicht genau zu sagen, aufgrund der stochastischen Natur des ML, sind die Ergebnisse bei einem erneuten Lerndurchlauf immer etwas anders. Es sollte aber jede Aktivitätsklasse mindestens einmal eingetragen worden sein, um die Genauigkeit des Modells über alle Klassen abschätzen zu können. Bis dahin wird die regelbasierte Klassifizierung verwendet. Auch neue Sensoren werden nicht sofort vom ML-Modell berücksichtigt, weshalb der Nutzer zum Eintragen von Aktivitäten stärker motiviert werden sollte, beispielsweise durch eine ansprechendere Schnittstelle. Das nachträgliche Hinzufügen einer Aktivitätsklasse, erweitert ebenfalls das ML-Modell, aber erst nach dem nächsten Lernprozess. Der Lernvorgang wird regelmäßig angestoßen, sodass das Modell aktuell bleibt, aber es wurde eine von der Bibliothek *scikit-learn* vorgegebene Implementierung des KNN-Algorithmus genutzt, welche möglicherweise noch optimiert werden kann. Daher könnte dort noch eine eigene Implementierung besser sein, aber auch mit der vorgegebenen Implementierung konnte eine Genauigkeit von über 90 % erreicht werden.

Hardware. Der Prototyp konnte auf einem Raspberry Pi umgesetzt werden, was der Anforderung kostengünstiger Hardware entgegen kommt. Außerdem wurde mit openHAB eine Open-Source-Lösung als Verwaltungssoftware für die SH-Sensorik verwendet, welche es ermöglichte, dass das System unabhängig von herstellerspezifischer Software der Sensorik benutzbar ist. Die Bewegungsmelder und Türkontaktsensoren sorgten mit ihren diskreten Wertänderungen für eine genaue Erkennung der Aktivitäten, wohingegen die Temperatur- und Feuchtigkeitssensoren zeitlich versetzt Werte übermittelten. Dies lag daran, dass Temperatur und Feuchtigkeit

stetig ansteigen und sinken, weshalb die Sensoren erst ein paar Minuten nach Aktivitätsbeginn getriggert wurden. Zusätzlich sorgten die Wetterbedingungen für verfälschte Werte, da die Werte bei schwülen und heißen Wetter langsamer sanken als üblich. Es können auch neue Sensoren in das System integriert werden, welche direkt von der regelbasierten Klassifizierung verarbeitet werden. Dafür muss der Sensor einerseits zu OH und andererseits mit Standort in die Datenbank hinzugefügt werden, was aktuell nur mit tieferehenden technischen Wissen vom System möglich ist. Ebenso wäre es denkbar, Roboter und tragbare Sensorik in das System zu integrieren, da mit diesen Technologien schon erfolgreich Aktivitäten klassifiziert wurden [Par+20; Fra16].

Nutzerakzeptanz und Gebrauchstauglichkeit. Die während des Versuchs getesteten Bestandteile des Systems konnten von fast allen Nutzern ohne Probleme in vollem Umfang benutzt werden. Dies beinhaltet die Nutzung der grafischen Oberfläche, als auch die Durchführung der ADL, während die Sensoren diese registrierten, aber nicht das Hinzufügen neuer Sensoren. Allerdings wurde das System hauptsächlich von jungen und technisch interessierten Probanden getestet, von denen ca. 70 % selbst im Besitz von SH-Technologie waren. Insgesamt kam trotzdem bei allen Nutzern kaum Frust auf und das System wurde überwiegend als transparent und unterstützend aufgefasst. Dabei war für die Nutzer nicht die Klassifizierung im Vordergrund, sondern die Ansicht ihrer Daten und Aktivitäten. Bei den älteren Probanden hat sich gezeigt, dass der PC als Eingabemöglichkeit nicht bevorzugt wird, da dort die Erfahrungen fehlen und die Angst vor falschen Eingaben besteht. Da alle Probanden mit SH-Technologie einen Sprachassistenten besaßen, würde sich dieser als weitere Schnittstelle eignen. Ebenso ist ein Roboter, beispielsweise Loomo, als Eingabemöglichkeit denkbar, da dieser bei der Durchführung der Aktivitäten präsent sein und somit direkt zum Loggen durchgeführter Aktivitäten genutzt werden könnte, welche zum Trainieren des ML-Modells genutzt werden.

Datenschutz und Privatsphäre. Zum Schutz der Privatsphäre wurde komplett auf Kameras und Tonaufnahmegeräte verzichtet. Die genutzte Sensorik konnte auf keine Weise bestimmen, welche Person sie aktuell erkennt. Sollte wiederum jemand in das System eindringen und die dort hinterlegten Nutzerinformationen auslesen, so könnten Rückschlüsse gezogen werden, welche Person die Aktivitäten ausgeführt hat. Der Datenschutz war die größte Sorge der Nutzer, da sie Bedenken hatten, dass der erkannte Tagesablauf von Außenstehenden für kriminelle Zwecke genutzt werden könnte. Das System funktioniert komplett ohne Internetanbindung, der Zugang erfolgt über SSH und die Datenbanken sind mit Passwörtern abgesichert, sodass das System grundlegende Sicherheitsmechanismen besitzt. Trotz dieser Informationen ließen sich die Bedenken der Nutzer nicht beseitigen. Wenn das System über das lokale Netzwerk betrieben oder für Remoteverbindungen erweitert wird, muss das Sicherheitskonzept verbessert werden.

Kosten. Die Hardware für den funktionsfähigen Prototypen war mit knapp 200 € Kosten weit unter dem gesetzten Limit von 750 €, sodass die Sorge der Zielgruppen, dass das System zu teuer sein könnte, beseitigt werden konnte. Das System bleibt trotzdem erweiterbar, was wiederum mit weiteren Kosten verbunden ist, da neue Hardware gekauft werden müsste. Es wurde kein Test für die laufenden Kosten durchgeführt, welche durch Robustheit der Sensoren, Energieverbrauch und Wartungskosten beeinflusst werden. In einem Monat Testzeit mussten keine Batterien gewechselt werden und es ist auch kein Sensor kaputt gegangen. Trotzdem haben zwei Sensoren regelmäßig ihre Verbindung verloren, da Kommunikation mit Koordinator nicht korrekt funktionierte. Dieses Problem gibt es mit offiziellen Bridges, wie Z-Wave-Sticks, nicht, aber dafür ist die Z-Wave-Sensorik kostenintensiver im Einkauf. Außerdem wird für die Einrichtung neuer Sensoren aktuell noch Expertenwissen benötigt, was in einem realen Szenario zu

weiteren Kosten führen könnte.

Interoperabilität und Robustheit. Das System hatte die Anforderung nach einem Ausfall sofort wieder zu funktionieren und alle bestehenden Daten zu behalten. Durch die Persistierung in Datenbanken konnten alle Daten beibehalten werden. Alle Funktionalitäten wurden als System-Services eingerichtet, sodass diese bei einem Neustart automatisch, ohne Eingreifen des Nutzers gestartet werden. Ein unregelmäßig auftretendes Phänomen war, dass die Temperatursensoren in den Offline-Status gingen und keine Signale mehr sendeten, bis der Nutzer den Sensor kurz manuell triggerte. Dies hing aber mit dem Sensormodell zusammen, da es offline ging, um Strom zu sparen. Mit dem originalen Hersteller-Hub passiert dies wahrscheinlich nicht, aber dafür wäre das hier konzipierte System dann nicht mehr unabhängig vom Hersteller der Sensoren. Die Nutzung kabelgebundener Sensorik, um sich vor Ausfällen abzusichern, ist keine Alternative, da dies teurer und aufwändiger einzurichten ist. Zur Kommunikation mit den Services werden REST-Schnittstellen genutzt, welche von jedem anderen webfähigen Gerät im lokalen Netz aufgerufen werden können. Die Schnittstellen entsprechen einem modernen Standard, sodass der Großteil aktueller Assistenzsysteme diese nutzen können. Allerdings sind die Schnittstellen beim aktuellen Prototypen noch ungesichert, sodass dort noch eine Sicherheitslücke besteht. Es können auch problemlos weitere Schnittstellen auf dem System bereitgestellt werden.

Adaptivität. Das System ist dynamisch und passt sich an bestimmten Stellen den Präferenzen des Nutzers an. Wenn der Nutzer regelmäßig seine durchgeführten ADLs in das System einträgt, dann ist durch das ML-Modell ein adaptives Verhalten des Systems gegeben. Da das Modell regelmäßig aktualisiert wird, passt sich die Klassifizierung, also die Kernfunktionalität des Systems, den Eingaben des Nutzers an und erkennt somit die individuellen Angewohnheiten bei den Aktivitäten. Der Nutzer hat auch die Möglichkeit, die Standortdaten der Sensoren zu ändern, was dazu führt, dass sich die regelbasierte Klassifizierung der Aktivitäten verändert, wenn dieser Sensor den Klassifizierungsalgorithmus startet. Momentan ist für jeden Standort die dazugehörige Aktivität vom System vorgegeben, sodass die Nutzer nur den Standort, aber nicht die Zuweisung von Standort und Aktivität ändern können. Das Hinzufügen und Entfernen von Sensoren aus dem System beeinträchtigt die Klassifizierung nicht und wird vom regelbasierten Algorithmus direkt beachtet. Das ML-Modell beachtet solche Änderungen ebenfalls, aber erst nach dem nächsten Lernprozess.

7.3 Beantwortung der Forschungsfrage

Am Anfang der Arbeit stand die Fragestellung, ob es möglich ist, mit SH-Technologie aus dem Konsumentenbereich ein System zu erschaffen, welches dazu fähig ist, mit anderen Assistenzsystemen zu kommunizieren sowie Aktivitäten des Nutzers zu erkennen und sich dabei an dessen Verhaltensweisen anzupassen. Im Laufe der Arbeit ist ein Konzept für ein solches System entstanden, welches als Prototyp umgesetzt und mit mehreren Probanden getestet werden konnte. Die Anforderungen, welche an das System gestellt wurden, konnten erfüllt werden, sodass von einer guten Gebrauchstauglichkeit des Systems ausgegangen werden kann. Das System ist über REST-Schnittstellen erreichbar und kann mit weiteren Schnittstellen erweitert werden, sodass es mit jedem internetfähigen Assistenzsystem kommunizieren kann. Es wurde auf Sensoren der Marke Aqara gesetzt, wobei jeder Sensor, welcher das ZigBee-Protokoll nutzt, funktioniert hätte. Über die Möglichkeit die Klassifizierung zu beeinflussen, indem der Nutzer Standorte für die Sensoren einstellt oder regelmäßig Aktivitäten einträgt, damit das System dazulernt, passt sich das System in seiner Kernfunktionalität den Präferenzen des Nutzers

an. Somit ließ sich die Fragestellung lösen und es ist ein funktionsfähiger Prototyp entstanden, welcher Potenzial für mehrere Erweiterungen besitzt.

7.4 Ausblick

- **Evaluation weiterer ML-Algorithmen:** In dieser Arbeit wurden nur von *scikit-learn* bereitgestellte ML-Algorithmen getestet und evaluiert, wobei der KNN-Algorithmus am besten abschnitt. Es ist möglich, dass andere ML-Algorithmen oder sogar eine eigene Implementierung die Aktivitäten genauer klassifizieren könnten. Daher wäre eine Evaluation weiterer Algorithmen eine gute Möglichkeit, um herauszufinden, ob das System genauer werden kann.
- **Ausarbeitung weiterer Anwendungsfälle:** Das System stellt den Tagesablauf eines Nutzers zur Verfügung, sodass untersucht werden sollte, welche Anwendungsfälle es im Zusammenspiel mit weiteren Assistenzsystemen geben könnte. Durch den Zugriff auf den regelmäßigen Tagesablauf, sowie die aktuellen Aktivitäten des Nutzers, könnten sich für andere Assistenzsysteme, vor allem für Roboter wie Loomo, neue Möglichkeiten ergeben. Es könnten beispielsweise Gespräche gestartet werden, wenn bestimmte Aktivitäten länger nicht mehr ausgeführt wurden oder es könnten auch Krankheitsmuster durch einen sich ändernden Tagesablauf erkannt werden.
- **Weitere Schnittstellen zur Kommunikation:** Aktuell können weitere Assistenzsysteme im selben lokalen Netzwerk das System ansprechen und über die REST-Schnittstellen die Klassifizierung starten oder über die OH-API Sensorwerte abfragen. Für einen Roboter, wie beispielsweise Loomo, wäre interessant, in welchem Raum aktuell Sensoraktivitäten stattfinden und welche Aktivität voraussichtlich als Nächstes stattfindet. Dadurch könnte der Roboter schneller den Nutzer in seiner Wohnung finden.
- **Verbesserte und zusätzliche Benutzungsschnittstellen:** Da es nur den PC als Eingabemöglichkeit gibt, wurde vor allem von älteren Nutzern bemängelt. Ein Roboter oder ein Sprachassistent könnten durch eine natürlichere Kommunikation, als bessere Schnittstelle zum Eintragen von Aktivitäten dienen, da diese direkt nach der Aktivität angesprochen werden können. Ebenso könnte auch das bestehende Interface motivierender gestaltet werden, beispielsweise durch Gamifizierung, also der Einbindung spieltypischer Elemente. Das Eintragen von Aktivitäten verbessert schließlich die Genauigkeit bei der Klassifizierung, weshalb die Nutzer auch gerne ihre Aktivitäten eintragen sollten.
- **Remoteüberwachung:** Eine Remoteverbindung zum System, beispielsweise durch einen Pflegedienst, wäre ein nützliches Werkzeug, um Pflegepersonal zu unterstützen. So könnten schon im Vorfeld Informationen gesammelt und abgeschätzt werden, ob vielleicht außerhalb des Terminplanes jemand den Pflegebedürftigen besuchen sollte. Mit einer Remoteverbindung müsste auch das Sicherheitskonzept überarbeitet werden, da dieser Fall momentan nicht abgedeckt ist.
- **Verarbeitung der Umweltdaten:** In der Diskussion wurde erwähnt, dass es bei Temperatur- und Feuchtigkeitssensoren verfälschte Werte aufgrund der Wetterbedingungen gab. Eine Möglichkeit wäre es eine Wetterstation einzubinden, wie es Martin Franke in seiner Arbeit tat [Fra16]. Anstatt die Wetterdaten für die Analyse des Wohlbefindens der Nutzer zu verwenden, könnte man diese nutzen, um die Schwellen in den Regeln anzupassen, welche als Auslöser für den Klassifizierungsalgorithmus dienen.

Literatur

- [AAL18] AAL Association. *AAL PROGRAMME - What we do*. AAL Programme. Library Catalog: www.aal-europe.eu. 5. Nov. 2018. URL: <http://www.aal-europe.eu/about/> (besucht am 29.04.2020).
- [Ale+13] Hande Alemdar u. a. „ARAS Human Activity Datasets in Multiple Homes with Multiple Residents“. In: 7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops. 2013, S. 4.
- [BHR09] J. Broekens, M. Heerink und H. Rosendal. „Assistive social robots in elderly care: a review“. In: *Gerontechnology* 8.2 (1. Apr. 2009), S. 94–103. ISSN: 1569-111X, 1569-1101. DOI: 10.4017/gt.2009.08.02.002.00. URL: <http://gerontechnology.info/index.php/journal/article/view/1011> (besucht am 29.04.2020).
- [Bla+16] Stephanie Blackman u. a. „Ambient Assisted Living Technologies for Aging Well: A Scoping Review“. In: *Journal of Intelligent Systems* 25.1 (1. Jan. 2016). ISSN: 2191-026X, 0334-1860. DOI: 10.1515/jisys-2014-0136. URL: <https://www.degruyter.com/view/j/jisys.2016.25.issue-1/jisys-2014-0136/jisys-2014-0136.xml> (besucht am 29.04.2020).
- [Boi+07] Patrick Boissy u. a. „A qualitative study of in-home robotic telepresence for home care of community-living elderly subjects“. In: *Journal of Telemedicine and Telecare* 13.2 (März 2007), S. 79–84. ISSN: 1357-633X, 1758-1109. DOI: 10.1258/135763307780096195. URL: <http://journals.sagepub.com/doi/10.1258/135763307780096195> (besucht am 29.04.2020).
- [Boo+08] Ann Bookman u. a. *Family Caregiver Handbook*. MIT Workplace Center, 2008. 154 S. URL: <http://web.mit.edu/workplacecenter/hndbk/> (besucht am 29.04.2020).
- [Cal+17] Davide Calvaresi u. a. „Exploring the ambient assisted living domain: a systematic review“. In: *Journal of Ambient Intelligence and Humanized Computing* 8.2 (Apr. 2017), S. 239–257. ISSN: 1868-5137, 1868-5145. DOI: 10.1007/s12652-016-0374-3. URL: <http://link.springer.com/10.1007/s12652-016-0374-3> (besucht am 29.04.2020).
- [Car+] Marco Carraro u. a. „An Open Source Robotic Platform for Ambient Assisted Living“. In: (), S. 5.
- [Cha+] C Chalmers u. a. „Detecting Activities of Daily Living and Routine Behaviours in Dementia Patients Living Alone Using Smart Meter Load Disaggregation“. In: (), S. 12.
- [Cou+08] Karen Courtney u. a. „Needing smart home technologies: the perspectives of older adults in continuing care retirement communities“. In: *Journal of Innovation in Health Informatics* 16.3 (1. Nov. 2008), S. 195–201. ISSN: 2058-4563, 2058-4555. DOI: 10.14236/jhi.v16i3.694. URL: <http://hijournal.bcs.org/index.php/jhi/article/view/694> (besucht am 29.04.2020).
- [CS09] D. J. Cook und M. Schmitter-Edgecombe. „Assessing the Quality of Activities in a Smart Environment“. In: *Methods of Information in Medicine* 48.5 (2009), S. 480–485. ISSN: 0026-1270, 2511-705X. DOI: 10.3414/ME0592. URL: <http://www.thieme-connect.de/DOI/DOI?10.3414/ME0592> (besucht am 20.05.2020).

- [DB13] Torbjørn Dahl und Maged Boulos. „Robots in Health and Social Care: A Complementary Technology to Home Care and Telehealthcare?“ In: *Robotics* 3.1 (30. Dez. 2013), S. 1–21. ISSN: 2218-6581. DOI: 10.3390/robotics3010001. URL: <http://www.mdpi.com/2218-6581/3/1/1> (besucht am 29. 04. 2020).
- [DBS18] Hugo H. Davila, Karisa M. Brown und Raul E. Storey. „When do we need to consider robotic-assisted radical nephrectomy?—repercussion to the health care cost between 2003 to 2015“. In: *Laparoscopic Surgery* 2 (März 2018), S. 5–5. ISSN: 26164221. DOI: 10.21037/ls.2018.02.02. URL: <http://ls.amegroups.com/article/view/4135/4739> (besucht am 29. 04. 2020).
- [Dim+16] Ace Dimitrievski u. a. „Towards application of non-invasive environmental sensors for risks and activity detection“. In: *2016 IEEE 12th International Conference on Intelligent Computer Communication and Processing (ICCP)*. 2016 IEEE 12th International Conference on Intelligent Computer Communication and Processing (ICCP). Cluj-Napoca, Romania: IEEE, Sep. 2016, S. 27–33. ISBN: 978-1-5090-3899-2. DOI: 10.1109/ICCP.2016.7737117. URL: <http://ieeexplore.ieee.org/document/7737117/> (besucht am 29. 04. 2020).
- [Do+18] Ha Manh Do u. a. „RiSH: A robot-integrated smart home for elderly care“. In: *Robotics and Autonomous Systems* 101 (März 2018), S. 74–92. ISSN: 09218890. DOI: 10.1016/j.robot.2017.12.008. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0921889017300477> (besucht am 29. 04. 2020).
- [Fer+17] Manuel Fernandez-Carmona u. a. „Entropy-based abnormal activity detection fusing RGB-D and domotic sensors“. In: *2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. 2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI). Dae-gu: IEEE, Nov. 2017, S. 42–48. ISBN: 978-1-5090-6064-1. DOI: 10.1109/MFI.2017.8170405. URL: <http://ieeexplore.ieee.org/document/8170405/> (besucht am 29. 04. 2020).
- [Fis+16] David Fischinger u. a. „Hobbit, a care robot supporting independent living at home: First prototype and lessons learned“. In: *Robotics and Autonomous Systems* 75 (Jan. 2016), S. 60–78. ISSN: 09218890. DOI: 10.1016/j.robot.2014.09.029. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0921889014002140> (besucht am 29. 04. 2020).
- [Fra16] Martin Franke. „Wissensbasierte Tagesrhythmenerfassung und -auswertung in ubiquitären Umgebungen“. Dissertation. Technische Universität Dresden, 2. Nov. 2016. 163 S. URL: [https://tud.qucosa.de/landing-page/?tx_dlf\[id\]=https%3A%2F%2Ftud.qucosa.de%2Fapi%2Fqucosa%253A29917%2Fmets](https://tud.qucosa.de/landing-page/?tx_dlf[id]=https%3A%2F%2Ftud.qucosa.de%2Fapi%2Fqucosa%253A29917%2Fmets) (besucht am 03. 05. 2020).
- [GH] Birgit Graf und Martin Hägele. „Dependable Interaction with an Intelligent Home Care Robot“. In: (), S. 7.
- [GHS04] Birgit Graf, Matthias Hans und Rolf D. Schraft. „Care-O-bot II—Development of a Next Generation Robotic Home Assistant“. In: *Autonomous Robots* 16.2 (März 2004), S. 193–205. ISSN: 0929-5593. DOI: 10.1023/B:AURD.0000016865.35796.e9. URL: <http://link.springer.com/10.1023/B:AURD.0000016865.35796.e9> (besucht am 29. 04. 2020).
- [Gun] Vaishnavi S Gunge. „Smart Home Automation: A Literature Review“. In: *International Journal of Computer Applications* (), S. 5.

- [Hal+19] Amanda K. Hall u. a. „Acceptance and perceived usefulness of robots to assist with activities of daily living and healthcare tasks“. In: *Assistive Technology* 31.3 (27. Mai 2019), S. 133–140. ISSN: 1040-0435, 1949-3614. DOI: 10.1080/10400435.2017.1396565. URL: <https://www.tandfonline.com/doi/full/10.1080/10400435.2017.1396565> (besucht am 29. 04. 2020).
- [Hän+16] Daniela Händler-Schuster u. a. „Den pflegerischen Unterstützungsbedarf durch Daten über den Energieverbrauch erkennen : eine Pilotstudie“. In: *Pflegezeitschrift* 69.4 (2016). Accepted: 2018-03-15T14:27:48Z Publisher: Springer, S. 1–5. ISSN: 0945-1129. DOI: 10.21256/zhaw-1807. URL: <https://digitalcollection.zhaw.ch/handle/11475/3939> (besucht am 03. 05. 2020).
- [Han+18] Jun Han u. a. „Do You Feel What I Hear? Enabling Autonomous IoT Device Pairing Using Different Sensor Types“. In: *2018 IEEE Symposium on Security and Privacy (SP)*. 2018 IEEE Symposium on Security and Privacy (SP). San Francisco, CA: IEEE, Mai 2018, S. 836–852. ISBN: 978-1-5386-4353-2. DOI: 10.1109/SP.2018.00041. URL: <https://ieeexplore.ieee.org/document/8418641/> (besucht am 29. 04. 2020).
- [Har+05] P. Harmo u. a. „Needs and solutions - home automation and service robots for the elderly and disabled“. In: *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems. Edmonton, Alta., Canada: IEEE, 2005, S. 3201–3206. ISBN: 978-0-7803-8912-0. DOI: 10.1109/IR0S.2005.1545387. URL: <http://ieeexplore.ieee.org/document/1545387/> (besucht am 29. 04. 2020).
- [HB15] Kevin Anthony Hoff und Masooda Bashir. „Trust in Automation: Integrating Empirical Evidence on Factors That Influence Trust“. In: *Human Factors: The Journal of the Human Factors and Ergonomics Society* 57.3 (Mai 2015), S. 407–434. ISSN: 0018-7208, 1547-8181. DOI: 10.1177/0018720814547570. URL: <http://journals.sagepub.com/doi/10.1177/0018720814547570> (besucht am 03. 07. 2020).
- [HCS19] Helen Harman, Keshav Chintamani und Pieter Simoens. „Robot Assistance in Dynamic Smart Environments—A Hierarchical Continual Planning in the Now Framework“. In: *Sensors* 19.22 (7. Nov. 2019), S. 4856. ISSN: 1424-8220. DOI: 10.3390/s19224856. URL: <https://www.mdpi.com/1424-8220/19/22/4856> (besucht am 29. 04. 2020).
- [HGS02] M. Hans, B. Graf und R.D. Schraft. „Robotic home assistant Care-O-bot: past-present-future“. In: *Proceedings. 11th IEEE International Workshop on Robot and Human Interactive Communication*. IEEE ROMAN 2002. 11th IEEE International Workshop on Robot and Human Interactive Communication. Berlin, Germany: IEEE, 2002, S. 380–385. ISBN: 978-0-7803-7545-1. DOI: 10.1109/ROMAN.2002.1045652. URL: <http://ieeexplore.ieee.org/document/1045652/> (besucht am 29. 04. 2020).
- [Hof10] Robert Hofwimmer. *Analyse der Heimeinzugsphase im Alten- und Pflegezentrum Vöcklamarkt*. März 2010. URL: <https://docplayer.org/145076-Analyse-der-heimeinzugsphase-im-alten-und-pflegezentrum-voecklamarkt.html> (besucht am 08. 05. 2020).
- [Hue+16] Ramon Huerta u. a. „Online Decorrelation of Humidity and Temperature in Chemical Sensors for Continuous Monitoring“. In: *Chemometrics and Intelligent Laboratory Systems* 157 (Okt. 2016), S. 169–176. ISSN: 01697439. DOI: 10.1016/j.chemolab.2016.07.004. arXiv: 1608.01719. URL: <http://arxiv.org/abs/1608.01719> (besucht am 27. 05. 2020).
- [Inf20] InfluxData Inc. *InfluxDB Open Source Time Series Database | InfluxDB*. InfluxData. Library Catalog: www.influxdata.com. 3. März 2020. URL: <https://www.influxdata.com/products/influxdb-overview/> (besucht am 03. 05. 2020).

- [JS18] Antonio Jiménez und Fernando Seco. „Event-Driven Real-Time Location-Aware Activity Recognition in AAL Scenarios“. In: *Proceedings 2.19* (30. Okt. 2018), S. 1240. ISSN: 2504-3900. DOI: 10.3390/proceedings2191240. URL: <http://www.mdpi.com/2504-3900/2/19/1240> (besucht am 29. 04. 2020).
- [Kan+16] Byeongkwan Kang u. a. „Analysis of Types and Importance of Sensors in Smart Home Services“. In: *2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*. 2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS). Sydney, Australia: IEEE, Dez. 2016, S. 1388–1389. ISBN: 978-1-5090-4297-5. DOI: 10.1109/HPCC-SmartCity-DSS.2016.0196. URL: <http://ieeexplore.ieee.org/document/7828537/> (besucht am 29. 04. 2020).
- [Kas+08] Tim van Kasteren u. a. „Accurate activity recognition in a home setting“. In: *Proceedings of the 10th international conference on Ubiquitous computing - UbiComp '08*. the 10th international conference. Seoul, Korea: ACM Press, 2008, S. 1. ISBN: 978-1-60558-136-1. DOI: 10.1145/1409635.1409637. URL: <http://portal.acm.org/citation.cfm?doid=1409635.1409637> (besucht am 20. 05. 2020).
- [Kit+15] Ralf Kittmann u. a. „Let me Introduce Myself: I am Care-O-bot 4, a Gentleman Robot“. In: *Mensch und Computer 2015 - Tagungsband*. Hrsg. von Martin Pielot, Sarah Diefenbach und Niels Henze. Berlin, München, Boston: DE GRUYTER, 28. Jan. 2015. ISBN: 978-3-11-044392-9. DOI: 10.1515/9783110443929-024. URL: <https://www.degruyter.com/view/books/9783110443929/9783110443929-024/9783110443929-024.xml> (besucht am 29. 04. 2020).
- [Kle+07] Thomas Kleinberger u. a. „Ambient Intelligence in Assisted Living: Enable Elderly People to Handle Future Interfaces“. In: *Universal Access in Human-Computer Interaction. Ambient Interaction*. Hrsg. von Constantine Stephanidis. Bearb. von David Hutchison u. a. Bd. 4555. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, S. 103–112. ISBN: 978-3-540-73281-5. DOI: 10.1007/978-3-540-73281-5_11. URL: http://link.springer.com/10.1007/978-3-540-73281-5_11 (besucht am 04. 05. 2020).
- [Kun19] Alexander Kunst. *Eigene Ausgaben für Smart Home in Deutschland 2017*. Statista. Library Catalog: [de.statista.com](https://de.statista.com/prognosen/795404/umfrage-in-deutschland-zu-bisherigen-ausgaben-fuer-smart-home). 20. Dez. 2019. URL: <https://de.statista.com/prognosen/795404/umfrage-in-deutschland-zu-bisherigen-ausgaben-fuer-smart-home> (besucht am 10. 05. 2020).
- [Lau+18] Christl Lauterbach u. a. „AAL Functions for Home Care and Security“. In: *Current Directions in Biomedical Engineering* 4.1 (1. Sep. 2018), S. 127–129. ISSN: 2364-5504. DOI: 10.1515/cdbme-2018-0032. URL: <http://www.degruyter.com/view/j/cdbme.2018.4.issue-1/cdbme-2018-0032/cdbme-2018-0032.xml> (besucht am 29. 04. 2020).
- [LK18] Hyeong Suk Lee und Jeongeun Kim. „Scenario-Based Assessment of User Needs for Point-of-Care Robots“. In: *Healthcare Informatics Research* 24.1 (2018), S. 12. ISSN: 2093-3681, 2093-369X. DOI: 10.4258/hir.2018.24.1.12. URL: <https://synapse.koreamed.org/DOIx.php?id=10.4258/hir.2018.24.1.12> (besucht am 29. 04. 2020).
- [Loi18] Claudia Loitsch. „Designing Accessible User Interfaces for All by Means of Adaptive Systems“. Dissertation. Dresden: Technische Universität Dresden, 25. Juni 2018. 229 S. URL: [https://tud.qucosa.de/landing-page/?tx_dlf\[id\]=https%3A%2F%2Ftud.qucosa.de%2Fapi%2Fqucosa%253A31984%2Fmets](https://tud.qucosa.de/landing-page/?tx_dlf[id]=https%3A%2F%2Ftud.qucosa.de%2Fapi%2Fqucosa%253A31984%2Fmets).

- [Maj+17] Sumit Majumder u. a. „Smart Homes for Elderly Healthcare—Recent Advances and Research Challenges“. In: *Sensors* 17.11 (31. Okt. 2017), S. 2496. ISSN: 1424-8220. DOI: 10.3390/s17112496. URL: <http://www.mdpi.com/1424-8220/17/11/2496> (besucht am 29.04.2020).
- [Mar+08] Suzanne Martin u. a. „Smart home technologies for health and social care support“. In: *Cochrane Database of Systematic Reviews* (8. Okt. 2008). Hrsg. von Cochrane Effective Practice and Organisation of Care Group. ISSN: 14651858. DOI: 10.1002/14651858.CD006412.pub2. URL: <http://doi.wiley.com/10.1002/14651858.CD006412.pub2> (besucht am 29.04.2020).
- [Med+11] Hamid Medjahed u. a. „A pervasive multi-sensor data fusion for smart home healthcare monitoring“. In: *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*. 2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). Taipei, Taiwan: IEEE, Juni 2011, S. 1466–1473. ISBN: 978-1-4244-7315-1. DOI: 10.1109/FUZZY.2011.6007636. URL: <http://ieeexplore.ieee.org/document/6007636/> (besucht am 29.04.2020).
- [Mer+] Matthias Merten u. a. „A mobile robot platform for socially assistive home-care applications“. In: (), S. 6.
- [Mic] F Michaud. „Telepresence Robot for Home Care Assistance“. In: (), S. 6.
- [Mor+13] Elaine Mordoch u. a. „Use of social commitment robots in the care of elderly people with dementia: A literature review“. In: *Maturitas* 74.1 (Jan. 2013), S. 14–20. ISSN: 03785122. DOI: 10.1016/j.maturitas.2012.10.015. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0378512212003489> (besucht am 29.04.2020).
- [Neh+06] Jürgen Nehmer u. a. „Living assistance systems: an ambient intelligence approach“. In: *Proceeding of the 28th international conference on Software engineering - ICSE '06*. Proceeding of the 28th international conference. Shanghai, China: ACM Press, 2006, S. 43. ISBN: 978-1-59593-375-1. DOI: 10.1145/1134285.1134293. URL: <http://portal.acm.org/citation.cfm?doid=1134285.1134293> (besucht am 29.04.2020).
- [Ngu+17] Sao Mai Nguyen u. a. „Plug and Play your Robot into your Smart Home: Illustration of a New Framework: A Communication Protocol to Plug and Play any Robot or Device of a Smart Home“. In: *KI - Künstliche Intelligenz* 31.3 (Aug. 2017), S. 283–289. ISSN: 0933-1875, 1610-1987. DOI: 10.1007/s13218-017-0494-8. URL: <http://link.springer.com/10.1007/s13218-017-0494-8> (besucht am 29.04.2020).
- [Nov+15] Peter Novitzky u. a. „A Review of Contemporary Work on the Ethics of Ambient Assisted Living Technologies for People with Dementia“. In: *Science and Engineering Ethics* 21.3 (Juni 2015), S. 707–765. ISSN: 1353-3452, 1471-5546. DOI: 10.1007/s11948-014-9552-x. URL: <http://link.springer.com/10.1007/s11948-014-9552-x> (besucht am 29.04.2020).
- [NSS13] Christoph Nedopil, Cornelia Schauber und Glende Sebastian. *KNOWLEDGE BASE AAL STAKEHOLDERS AND THEIR REQUIREMENTS*. 17. Dez. 2013. URL: https://www.youse.de/documents/nYOUSE/AALA_Knowledge_Base_YOUSE_online.pdf (besucht am 09.05.2020).
- [ope20] openHAB Foundation e.V. *openHAB*. openHAB. Library Catalog: www.openhab.org. 2020. URL: <https://www.openhab.org/> (besucht am 29.04.2020).
- [Par+20] Stylianos Paraschiakos u. a. „Activity recognition using wearable sensors for tracking the elderly“. In: *User Modeling and User-Adapted Interaction* 30.3 (Juli 2020), S. 567–605. ISSN: 0924-1868, 1573-1391. DOI: 10.1007/s11257-020-09268-2. URL: <http://link.springer.com/10.1007/s11257-020-09268-2> (besucht am 05.08.2020).

- [Pas+] Max Pascher u. a. „Little Helper: A Multi-Robot System in Home Health Care Environments“. In: (), S. 8.
- [Pea+12] Alan J. Pearce u. a. „Robotics to Enable Older Adults to Remain Living at Home“. In: *Journal of Aging Research* 2012 (2012), S. 1–10. ISSN: 2090-2204, 2090-2212. DOI: 10.1155/2012/538169. URL: <https://www.hindawi.com/journals/jar/2012/538169/> (besucht am 29.04.2020).
- [Per+13] Martinique Perkins u. a. „Caregiving Strain and All-Cause Mortality: Evidence From the REGARDS Study“. In: *The Journals of Gerontology Series B: Psychological Sciences and Social Sciences* 68.4 (Juli 2013), S. 504–512. ISSN: 1079-5014. DOI: 10.1093/geronb/gbs084. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3674731/> (besucht am 07.05.2020).
- [Per+17] Thinagaran Perumal u. a. „IoT based activity recognition among smart home residents“. In: *2017 IEEE 6th Global Conference on Consumer Electronics (GCCE)*. 2017 IEEE 6th Global Conference on Consumer Electronics (GCCE). Nagoya: IEEE, Okt. 2017, S. 1–2. ISBN: 978-1-5090-4045-2. DOI: 10.1109/GCCE.2017.8229478. URL: <http://ieeexplore.ieee.org/document/8229478/> (besucht am 29.04.2020).
- [Pol05] Martha E. Pollack. „Intelligent Technology for an Aging Population: The Use of AI to Assist Elders with Cognitive Impairment“. In: *AI Magazine* 26.2 (15. Juni 2005). Number: 2, S. 9–9. ISSN: 2371-9621. DOI: 10.1609/aimag.v26i2.1810. URL: <https://www.aaai.org/ojs/index.php/aimagazine/article/view/1810> (besucht am 29.04.2020).
- [RKL09] Min-Ho Ryu, Seongcheol Kim und Euehun Lee. „Understanding the factors affecting online elderly user’s participation in video UCC services“. In: *Computers in Human Behavior*. Including the Special Issue: Enabling elderly users to create and share self authored multimedia content 25.3 (1. Mai 2009), S. 619–632. ISSN: 0747-5632. DOI: 10.1016/j.chb.2008.08.013. URL: <http://www.sciencedirect.com/science/article/pii/S0747563208001696> (besucht am 09.05.2020).
- [RM15] Erica L. Reaves und MaryBeth Musumeci. *Medicaid and Long-Term Services and Supports: A Primer*. The Henry J. Kaiser Family Foundation. Library Catalog: www.kff.org. 15. Dez. 2015. URL: <https://www.kff.org/medicaid/report/medicaid-and-long-term-services-and-supports-a-primer/> (besucht am 07.05.2020).
- [Sch+17] Fynn Schwiegelshohn u. a. „Tackling The New Health-Care Paradigm Through Service Robotics: Unobtrusive, efficient, reliable, and modular solutions for assisted-living environments.“ In: *IEEE Consumer Electronics Magazine* 6.3 (Juli 2017). Conference Name: IEEE Consumer Electronics Magazine, S. 34–41. ISSN: 2162-2256. DOI: 10.1109/MCE.2017.2685158.
- [Seg20a] Segway. *Segway Loomo*. Segway Online-Shop. Library Catalog: shop.segway.com. 2020. URL: <https://shop.segway.com/de-de/57/-segway-loomo> (besucht am 29.04.2020).
- [Seg20b] SegwayRobotics Inc. *Segway Robotics | Developer*. Segway Robotics | Developer. 2020. URL: <https://developer.segwayrobotics.com/developer/documents/segway-robot-overview.html> (besucht am 29.04.2020).
- [SML10] Jeannette E South-Paul, Samuel C Matheny und Evelyn L Lewis. *CURRENT Diagnosis and Treatment in Family Medicine (3rd Edition)*. OCLC: 958564483. New York, USA: McGraw-Hill Professional Publishing, 2010. ISBN: 978-0-07-176938-9. URL: <http://public.ebookcentral.proquest.com/choice/publicfullrecord.aspx?p=4668558> (besucht am 29.04.2020).

- [Spo+18] Alexandros Spournias u. a. „Experimental Evaluation of a Novel Personal Assistant in Greek Language for Ambient Assisted Living Environments employing home robots“. In: *2018 South-Eastern European Design Automation, Computer Engineering, Computer Networks and Society Media Conference (SEEDA_CECNSM)*. 2018 South-Eastern European Design Automation, Computer Engineering, Computer Networks and Society Media Conference (SEEDA_CECNSM). Kastoria: IEEE, Sep. 2018, S. 1–9. ISBN: 978-618-83314-1-9. DOI: 10.23919/SEEDA-CECNSM.2018.8544920. URL: <https://ieeexplore.ieee.org/document/8544920/> (besucht am 29. 04. 2020).
- [Spr+16] Gina Sprint u. a. „Detecting Health and Behavior Change by Analyzing Smart Home Sensor Data“. In: *2016 IEEE International Conference on Smart Computing (SMARTCOMP)*. 2016 IEEE International Conference on Smart Computing (SMARTCOMP). St Louis, MO, USA: IEEE, Mai 2016, S. 1–3. ISBN: 978-1-5090-0898-8. DOI: 10.1109/SMARTCOMP.2016.7501687. URL: <http://ieeexplore.ieee.org/document/7501687/> (besucht am 29. 04. 2020).
- [SS12] Amanda Sharkey und Noel Sharkey. „Granny and the robots: ethical issues in robot care for the elderly“. In: *Ethics and Information Technology* 14.1 (März 2012), S. 27–40. ISSN: 1388-1957, 1572-8439. DOI: 10.1007/s10676-010-9234-6. URL: <http://link.springer.com/10.1007/s10676-010-9234-6> (besucht am 29. 04. 2020).
- [Tam+04] T. Tamura u. a. „Is an Entertainment Robot Useful in the Care of Elderly People With Severe Dementia?“ In: *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences* 59.1 (1. Jan. 2004), S. M83–M85. ISSN: 1079-5006, 1758-535X. DOI: 10.1093/gerona/59.1.M83. URL: <https://academic.oup.com/biomedgerontology/article-lookup/doi/10.1093/gerona/59.1.M83> (besucht am 29. 04. 2020).
- [Tor+14] Elena Torta u. a. „Evaluation of a Small Socially-Assistive Humanoid Robot in Intelligent Homes for the Care of the Elderly“. In: *Journal of Intelligent & Robotic Systems* 76.1 (Sep. 2014), S. 57–71. ISSN: 0921-0296, 1573-0409. DOI: 10.1007/s10846-013-0019-0. URL: <http://link.springer.com/10.1007/s10846-013-0019-0> (besucht am 29. 04. 2020).
- [UDP19] United Nations, Department of Economic and Social Affairs und Population Division. *World population prospects Highlights, 2019 revision Highlights, 2019 revision*. OCLC: 1142478963. 2019. ISBN: 978-92-1-148316-1.
- [VNK15] Michalis Vrigkas, Christophoros Nikou und Ioannis A. Kakadiaris. „A Review of Human Activity Recognition Methods“. In: *Frontiers in Robotics and AI* 2 (16. Nov. 2015). ISSN: 2296-9144. DOI: 10.3389/frobt.2015.00028. URL: <http://journal.frontiersin.org/Article/10.3389/frobt.2015.00028/abstract> (besucht am 29. 04. 2020).
- [VSS10] Antoine A. J. van de Ven, Anne-mie A. G. Sponselee und Ben A. M. Schouten. „Robo M.D.: a home care robot for monitoring and detection of critical situations“. In: *Proceedings of the 28th Annual European Conference on Cognitive Ergonomics - ECCE '10*. the 28th Annual European Conference. Delft, Netherlands: ACM Press, 2010, S. 375. ISBN: 978-1-60558-946-6. DOI: 10.1145/1962300.1962391. URL: <http://portal.acm.org/citation.cfm?doid=1962300.1962391> (besucht am 29. 04. 2020).
- [Wad+04] K. Wada u. a. „Effects of robot-assisted activity for elderly people and nurses at a day service center“. In: *Proceedings of the IEEE* 92.11 (Nov. 2004), S. 1780–1788. ISSN: 0018-9219. DOI: 10.1109/JPROC.2004.835378. URL: <http://ieeexplore.ieee.org/document/1347458/> (besucht am 29. 04. 2020).
- [Wil+14] Brie A Williams u. a. *Current diagnosis and treatment*. OCLC: 885407197. 2014. ISBN: 978-0-07-179209-7. URL: <http://accessmedicine.mhmedical.com/book.aspx?bookid=953> (besucht am 29. 04. 2020).

- [WM17] Reiner Wichert und Beate Mand, Hrsg. *Ambient Assisted Living*. Advanced Technologies and Societal Change. Cham: Springer International Publishing, 2017. ISBN: 978-3-319-52322-4. DOI: 10.1007/978-3-319-52322-4. URL: <http://link.springer.com/10.1007/978-3-319-52322-4> (besucht am 29.04.2020).
- [WPW12] Petra Winkler, Elisabeth Pochobradsky und Charlotte Wirl. *Gesundheit und Krankheit der älteren Generation in Österreich*. Februar 2015. Bundesministerium für Gesundheit, Sektion III, Radetzkystraße 2, 1030 Wien, Juni 2012. 162 S. ISBN: 978-3-85159-169-9.
- [WPW15] Franz Werner, Sabine Payr und Katharina Werner. *Potential of Robotics for Ambient Assisted Living*. Juli 2015.
- [Zig20] Zigbee Allianz. *Zuhause - Zigbee Allianz*. Zigbee Alliance. 17. März 2020. URL: <https://zigbeealliance.org/de/> (besucht am 29.04.2020).
- [ZWa18] Z-Wave Alliance. *About Z-Wave Technology*. Z-Wave Alliance. 30. Mai 2018. URL: https://z-wavealliance.org/about_z-wave_technology/ (besucht am 29.04.2020).
- [ZWa20] Z-Wave Europe GmbH. *Was ist Z-Wave - Z-Wave Smart Home*. Was ist Z-Wave. 2020. URL: <https://zwave.de/was-ist-zwave/> (besucht am 03.05.2020).

Abkürzungsverzeichnis

EU	Europäische Union	7
AAL	Ambient Assisted Living	5
ADL	Aktivitäten des täglichen Lebens	7
SH	Smarthome	3
OH	openHAB	10
ML	Machine Learning	23
KNN	k-Nearest-Neighbor	29
CART	Classification And Regression Tree	29

Programmcode-Verzeichnis

5.1	Beispiel für ein Item des Prototypen	43
5.2	Rule Engine Beispiel	43
5.3	Persistenzregeln	44
5.4	Auszug aus den Trainingsdaten	45
5.5	Auslesen der Datenbank und Routing zur HTML-Seite mittels Flask	46
5.6	Jinja HTML-Template	46

Abbildungsverzeichnis

0.1	Voll vernetzte SH-Umgebung, welche einen Roboter bei ungewöhnlichen Aktivitäten benachrichtigt	4
1.1	AAL Technologien im Laufe der Zeit	8
1.2	Kategorisierung von Aktivitäten des täglichen Lebens	9
1.3	Zusammenspiel von Bindings, Things und Items in openHAB	12
1.4	Loomo Robot Komponenten	14
2.1	Versuchsaufbau von Jiménez und Seco [JS18]	16
2.2	Versuchsaufbau und Sensorauswahl bei Dimitrievski et al. [Dim+16]: (a) Bewegungsmelder, (b) Mikrophon, (c) Infrarot Hindernissensor, (d) Vibrationssensor, (e) Ultraschall Distanzsensor, (f) Sensor für Nähe	18
3.1	Bedürfnisse der verschiedenen Nutzergruppen bezüglich eines AAL-Systems	20
3.2	Anwendungsfalldiagramm für das erwartete Verhalten des Systems	21
4.1	Profilansicht mit täglichen Routinen des Nutzers	26
4.2	Ansicht der manuellen Klassifizierung durch den Nutzer	26
4.3	Entity-Relationship-Modell der Routinen- und Aktivitätsdatenbank	27
4.4	Geschätzte Genauigkeit der Klassifizierer für die zwei Datensätze	29
4.5	Normalisierte Konfusionsmatrizen der Klassifikationen mit KNN und CART	31
4.6	Normalisierte Konfusionsmatrizen der Klassifikationen mit KNN für einen Zeitraum von 14 Tagen	33
4.7	Aufteilung der Räume aus dem ARAS-Datensatz [Ale+13]	35
4.8	Ablauf der Routinenanpassung	37
4.9	Übersicht der Bestandteile des Systems	38
5.1	Raspberry Pi mit CC2531 Koordinator und Bewegungsmelder	41
5.2	Zwei der eingebundenen Sensoren in der Versuchsumgebung	42
5.3	Paper UI Ausschnitt des Systems mit installierten Bindings	42
5.4	Visuelle Ausgabe des Systems in einem Webbrowser	47
6.1	Die Versuchsumgebungen mit Sensorstandorten	51
6.2	Versuchspersonen	52
6.3	Technisches Interesse der Nutzer	52
6.5	Allgemeine Eindrücke der Nutzer über das System	54
6.6	Ergebnisse des Fragebogens zu den Funktionalitäten der grafischen Oberfläche des Systems	56
6.7	Konfusionsmatrix des ML-Modells zur Aktivitätenerkennung	57
6.8	Nutzungswahrscheinlichkeit des Systems	59

A Anhang

Fragebogen - Aktivitätserkennungssystem

Persönliche Daten

Zuerst ein paar persönliche Fragen über dich.

1. Was ist dein Geschlecht? *

- Männlich
- Weiblich
- Divers

2. Wie alt bist du? *

Technisches Vorwissen

3. Wie würdest du dein Interesse an technischen Systemen einschätzen? *

0	1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	---	----

Sehr gering

Sehr groß

4. Besitzt du selbst Smarthome-Technologie zuhause? (Sprachassistent, Smarte Leuchtmittel, Smarte Heizthermostate, ...) *

- Ja
- Nein

5. Wenn ja, welche Technologien besitzt du?

Sprachassistentz (Alexa, Google Home, ...)

Smarte Leuchtmittel

Smarte Heizthermostate

Smarte Steckdosen

Sensoren

Bewegungsmelder

Other

Praktische Aufgaben

Dieser Abschnitt beschäftigt sich mit der Lösung der praktischen Aufgaben.

6. Konntest du alle Aufgaben lösen?

Ja

Nein

7. Falls nicht, welche Aufgabe(n) haben dir Probleme bereitet?

Tee kochen

Wohnung verlassen

Bad einlassen

Zähne putzen

Am PC arbeiten

Fernsehen schauen

Tee trinken

8. Hat dich etwas bei der Lösung der Aufgaben gestört?

Enter your answer

Tagesablauf abrufen

9. Die folgenden Fragen beziehen sich auf die Aufgabe einen Tee zu kochen und diesen Eintrag im Nutzerprofil einzusehen: *

	ja	eher ja	unentschieden	eher nein	nein	weiß nicht
Konntest du die Aufgabe lösen?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Hat dir das Lösen der Aufgabe zu lang gedauert?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Hat dich das Lösen der Aufgabe frustriert?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
War die Aufgabe für dich körperlich anstrengend?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
War die Aufgabe für dich mental anstrengend?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Aktivität eintragen

10. Die folgenden Fragen beziehen sich auf die Aufgabe eine Aktivität im Logbuch einzutragen: *

	ja	eher ja	unentschieden	eher nein	nein	weiß nicht
Konntest du die Aufgabe lösen?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Hat dir das Lösen der Aufgabe zu lang gedauert?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Hat dich das Lösen der Aufgabe frustriert?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
War die Aufgabe für dich körperlich anstrengend?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
War die Aufgabe für dich mental anstrengend?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Aktivität modifizieren

11. Die folgenden Fragen beziehen sich auf die Aufgabe eine Aktivität im Tagesablauf zu ändern oder zu löschen: *

	ja	eher ja	unentschieden	eher nein	nein	weiß nicht
Konntest du die Aufgabe lösen?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Hat dir das Lösen der Aufgabe zu lang gedauert?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Hat dich das Lösen der Aufgabe frustriert?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
War die Aufgabe für dich körperlich anstrengend?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
War die Aufgabe für dich mental anstrengend?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Allgemeine Zufriedenheit

12. Wie sehr treffen folgende Aussagen über die Arbeit mit dem Aktivitätenerkennungssystem bei dir zu? *

	ja	eher ja	unentschieden	eher nein	nein	weiß nicht
Das System hat mich bei Durchführung der Aufgaben gut unterstützt.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ich hatte das Gefühl, dass ich einen kompletten Überblick über meine Daten im System habe.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Die Abläufe zur Bewältigung der Aufgaben waren verständlich.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Die Nutzung des Systems hat mich nicht frustriert.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

13. Wie wahrscheinlich ist es, dass du das System selbst nutzen würdest? *

0	1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	---	----

Sehr unwahrscheinlich

Sehr wahrscheinlich

14. Was hat dir besonders gut am System gefallen?

Enter your answer

15. Welche Bedenken hättest du bei der Benutzung des Systems? Siehst du eventuell sogar Risiken?

Enter your answer

16. Was würdest du am System verbessern wollen?

Enter your answer

Installationsanleitung

1. Installation von openHABian:
<https://www.openhab.org/docs/installation/openhabian.html>
2. Per SSH auf den Raspberry Pi verbinden
3. Nach der Installation *openhabian-config* ausführen und *Optional Components* auswählen
 - a. InfluxDB installieren
 - b. Datenbank *sensor_data* erstellen
 - c. Nutzer *activity_tracker* erstellen und auf *sensor_data* berechtigen
 - d. Passwort für *activity_tracker* notieren
4. In die PaperUI Oberfläche wechseln und den Menüpunkt Add-ons auswählen
 - a. Bindings installieren:
 - i. MQTT
 - ii. ZigBee
 - iii. http
 - b. MISC installieren:
 - i. Rule Engine
 - c. Persistence installieren:
 - i. InfluxDB
 - ii. MySQL
 - d. Transformations installieren:
 - i. JSONPath
 - ii. XSLT
5. Bereitgestellte MySQL-Datenbank importieren
 - a. Bsp.: phpMyAdmin installieren: *sudo apt-get install phpMyAdmin*
 - b. Oberfläche öffnen und neue Datenbank erstellen auswählen
 - c. Dort die beigelegte Datei *ActivityRecognizerSystem.sql* importieren
6. ZigBee Coordinator über das Binding einrichten
 - a. Über Inbox auf Plus-Symbol klicken, ZigBee-Binding und manuell hinzufügen auswählen
 - b. Vorhandenen Koordinator auswählen
7. Sensoren hinzufügen
8. Ordner *openhab2* in „/etc/“ durch bereitgestellten Ordner mit dem gleichen Namen ersetzen
 - a. Im Unterordner *services* die Datei *influxdb* öffnen und das Passwort für den Nutzer *activity_tracker* umändern
 - b. Im Unterordner *items* die Datei *activity_recognition* öffnen und ggf. die Channel-Strings anpassen, nachdem die Sensoren dem System hinzugefügt wurden
 - c. Im Unterordner *scripts* die Datei *helper_functions.py* öffnen und in Zeile 33 das Passwort für *activity_tracker* anpassen

9. Python3 auf dem System installieren:

sudo apt-get update

sudo apt-get install python3

10. Python Packages installieren

a. In den Ordner „*/etc/openhab2/scripts/*“ navigieren

b. Befehl ausführen: *pip3 install -r requirements.txt*

11. Raspberry Pi neustarten

12. Services starten

a. In den Ordner „*/etc/openhab2/scripts/*“ navigieren

b. Skript *flask_interface_startup.sh* starten

c. Skript *classification_service_startup.sh* starten

d. Alternativ die Skripte als System-Services einbinden, sodass diese immer beim Systemstart ausgeführt werden