SENSORDREVET OG SELVJUSTERENDE LOKATIONSBASERET MUSIKSTREAMING

SENSOR DRIVEN AND SELF-ADJUSTING LOCATION AWARE MUSIC STREAMING

MARTIN THOMSEN ØSTERGAARD, 201906178

BACHELORPROJEKT

Juni 2023

Vejleder: Niels Olof Bouvin

AARHUS UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE

SENSORDREVET OG SELVJUSTERENDE LOKATIONSBASERET MUSIKSTREAMING

Sensor driven and self-adjusting location aware music streaming

MARTIN THOMSEN ØSTERGAARD

Bachelorprojekt

Institut for Datalogi
Det Naturvidenskabelige Fakultet
Aarhus Universitet

Juni 2023

Martin Thomsen Østergaard: *Sensordrevet og selvjusterende lokationsbaseret musikstreaming,* Bachelorprojekt © Juni 2023

ABSTRACT

Dette projekt præsenterer et Wi-Fi baseret Indoor Positioning System (IPS) understøttet af Support Vector Machine (SVM) og neurale netværk, der kan afgøre i hvilket rum, en person befinder sig. Blandt de undersøgte modeller præsterer SVM modellerne bedst med en gennemsnitlig nøjagtighed på imellem 81,2% og 85,7% afhængigt af mængden af data. Til at demonstrere brugen af dette IPS er Android appen UbiMusic blevet udviklet, der giver brugeren en sømløs musikoplevelse. Dette opnås ved, at systemet skifter højtaler, når brugeren går ind i et nyt rum, hvormed musikafspilningen følger brugeren.

This project presents a Wi-Fi based IPS supported by SVM and neural networks, which can determine in which room a person is located. Among the examined models, SVM models perform the best with an average accuracy ranging from 81.2% to 85.7% depending on the amount of data. To demonstrate the use of this IPS, the Android app UbiMusic has been developed, which allows the user to have an ubiquitous music experience. This is achieved by the system changing speaker when the user enters a new room, allowing the music to follow the user.

INDHOLD

1 INTRODUKTION 1
- 1.1 Video der demonstrerer brugen af UbiMusic 3

2 RELATERET ARBEJDE 5
- 2.1 Bestemmelse af positioner 5
  - 2.1.1 Trilateration 5
  - 2.1.2 Proximity 5
  - 2.1.3 Fingerprinting 6
- 2.2 Oversigt over positioneringssystemers nøjagtighed 6
- 2.3 Musiksystemer 9
  - 2.3.1 Flere brugere 10
- 2.4 Machine Learning 10

3 ANALYSE 13
- 3.1 Positionering 13
  - 3.1.1 Håndtering af rum uden en højtaler 14
  - 3.1.2 Bevægelse fra rum til rum 14
- 3.2 Valg af service 15

4 DESIGN AF ARKITEKTUR OG EVALUERING 17
- 4.1 Fingerprinting 17
- 4.2 Machine learning modeller 20
- 4.3 Services 21
- 4.4 Flere brugere 21
- 4.5 Dataindsamling 22
- 4.6 Evaluering 22
  - 4.6.1 Nøjagtighed 24
  - 4.6.2 Dataanalyse 26

5 IMPLEMENTERING 29
- 5.1 Arkitektur 29
  - 5.1.1 Databehandling 29
  - 5.1.2 ModelForUbiMusic 29
  - 5.1.3 UbiMusic 30
- 5.2 Design 32
- 5.3 Dataindsamling 33
- 5.4 Træning af modellerne 34
  - 5.4.1 SVM modeller 34
  - 5.4.2 Neurale netværk 35
- 5.5 Brugbarheden af modellerne 37
- 5.6 Spotify 37
- 5.7 Wi-Fi scanninger 38

6 EVALUERING 39
- 6.1 Access points 39
- 6.2 Forskellige tidspunkter 39
- 6.3 Udenfor trænet område 40

6.4 Forskellige modeller 42
6.5 Rumskifte 44
6.6 Nøjagtighed 47
7 VIDERE ARBEJDE 49
7.1 Andre hjem 49
7.2 Antal Wi-Fi scanninger 49
7.3 UbiMusic 49
8 KONKLUSION 51
8.1 Modeller 51
8.2 Nøjagtighed 51
8.3 Mængden af data 52
8.4 Rum uden højtalere 53
8.5 Forskellige tidspunkter 53
8.6 Udenfor hjemmet 53
8.7 UbiMusic 53
8.8 Præcision eller lav latency 54
8.9 Videre arbejde og anvendelsesmuligheder 55
8.9.1 Adaptiv træning 55
8.9.2 Generelle muligheder 55
A APPENDIKS 57

REFERENCER 61

FIGURER

Figur 1 Sekvensdiagram for en bruger, der indsamler data. 18
Figur 2 Sekvensdiagram for en bruger, der træner modeller. 19
Figur 3 Sekvensdiagram over systemets design, når det bruges af to brugere. 23
Figur 4 Plantegning over rækkehuset, hvor dataene er indsamlet. 24
Figur 5 UbiMusics Graphical User Interface (GUI). 30
Figur 6 Konfidensintervaller for forskellige tidspunkter. 40
Figur 7 Konfidensintervaller for aktiveringsfunktioner. 41
Figur 8 Konfidensintervaller for forskellige modeller. 42
Figur 9 Konsekvensen af tærsklen for, hvornår der skiftes rum. 46

TABELLER

Tabel 1 Oversigt over positioneringssystemer. 8
Tabel 2 Oversigt over de fem modeller, der er evalueret på. 24
Tabel 3 Oversigt over variablene i træningen af neurale netværk. 36
Tabel 4 Antal på hinanden følgende prædiktioner for det samme rum inden der skiftes rum. 57
Tabel 5 Konfidensintervaller for Figur 6. 58
Tabel 6 Konfidensintervaller for Figur 8. 59

AKRONYMER

A3S Audio as a Service

AoA Angle of Arrival

BLE Bluetooth Low Energy

CA	Context-Aware
CEL	Cross Entropy Loss
GPS	Global Positioning System
GSM	Global System for Mobile Communications
GUI	Graphical User Interface
IPS	Indoor Positioning System
kNN	k-Nearest-Neighbor
ML	Machine Learning
MLP	Multi-Layer Perceptron
RBF	Radial Basis Function
RSS	Recieved Signal Strength
RSSI	Recieved Signal Strength Indication
SVM	Support Vector Machine
TDoA	Time Difference of Arrival
ToF	Time of Flight
ToA	Time of Arrival
UWB	Ultra-Wideband

1 INTRODUKTION

Dette projekt fokuserer på brugen af Machine Learning (ML) til at afgøre i hvilket rum, en person befinder sig. For at demonstrere anvendelsen er Android appen UbiMusic blevet udviklet, der kan skifte musikafspilning fra en højtaler til en anden, når en bruger skifter rum.

Som positioneringssystem er Global Positioning System (GPS) velkendt, nøjagtigt og yderst udbredt, når det gælder udendørs positionering. Som IPS er GPS dog ikke særlig brugbart, og der er ikke et IPS, der generelt set performer virkelig godt. En af de meget anvendte metoder indenfor IPS er fingerprinting, hvor der tidligere er blevet brugt forskellige ML modeller til at bestemme positioner baseret på modtagne radiosignaler.

Ifølge tidligere forskning har ML, og i særlig grad SVM, et stort potentiale til understøttelse af IPS [20]. I dette projekt er der derfor blevet evalueret på SVM modeller overfor neurale netværk med fokus på deres nøjagtigheder og brugbarhed i et IPS. Fokusset har været på ubiquitous computing, der dels omfatter usynlig brug af computere og sømløse interaktioner, idet computerne og interaktionerne er allestedsnærværende. Dette projekts forskningsspørgsmål er:

> I hvilket omfang kan der laves en sømløs musikoplevelse i hjem med højtalere i flere rum ved brug af et IPS understøttet af SVM og neurale netværk?

At dette er relevant skyldes i særlig grad det stigende brug af smarte enheder som eksempelvis styring af lyd, lys og temperatur. At kunne afgøre i hvilket rum en bruger befinder sig, giver mulighed for at tilpasse enhederne til brugeren. Det gælder både for musik, der kun afspilles i det relevante rum, men også for lys, der kun er tændt i relevante lamper og for termostater, der tilpasser temperaturen afhængigt af brugerens ønsker. Dette projekt har derfor også energimæssige muligheder ved kun at bruge den strøm, der er nødvendig.

For at brugere kan få gavn af forskellige smarte enheder, er det altafgørende med et nøjagtigt IPS. Derfor fokuserer projektet på mulighederne for et nøjagtigt IPS ved at bruge ML modeller. Musiksystemet UbiMusic er en demonstration af, hvordan et IPS kan bruges til at realisere en sømløs musikafspilning. Sidenhen kan systemet udvides til også at indeholde andre services, men det afgørende er, at positionerne er nøjagtige.

Når der bruges en eller flere services, kan bestemmelsen af et rum deles op i tre scenarier. Brugeren kan enten være i et rum med én tilgængelig service, i et rum uden en tilgængelig service eller i en ukendt position. En ukendt position skyldes, at brugeren er udenfor det trænede område, hvilket eksempelvis kunne være, hvis en bruger har forladt sit hjem.

I dette projekt er der blevet undersøgt mulighederne for at afgøre, hvornår en bruger er i et rum uden en tilgængelig service eller er udenfor sit trænede område, og dermed kan formodes at have forladt sit hjem. Tidligere forskning har vist, at ved musiksystemer som UbiMusic forventer nogle brugere, at musikken sættes på pause, når en bruger går ind i et lokale uden en højtaler [28]. Muligheder for at understøtte denne konkrete funktionalitet er derfor undersøgt, og det bidrager til evalueringen af forskningsspørgsmålet.

I dette projekt bruges udelukkende Wi-Fi signaler, der udsendes fra eksisterende access points. Dette gør, at modellerne og måden at afgøre den indendørs position kan bruges i langt de fleste hjem. Derudover skal der ikke opsættes noget hardware, og konfigureringen af systemet i det enkelte hjem foregår udelukkende via UbiMusics brugergrænseflade.

Ved at indsamle data om formiddagen og aftenen og evaluere på modellernes indbyrdes nøjagtigheder er det undersøgt, om der er forskel på Wi-Fi signalers styrke afhængigt af, hvor mange personer og enheder, der forventes at bruge Wi-Fi. Dette indgår også i evalueringen af forskningsspørgsmålet.

Til evaluering af forskningsspørgsmålet er lavet fem ML modeller. De er trænet og testet ud fra data indsamlet i et rækkehus, hvor der er blevet indsamlet data i fire rum fordelt på to etager. Ud fra det indsamlede data er det undersøgt, hvilken nøjagtighed forskellige mængder af indsamlet data giver. Dette er gjort for, at det i evalueringen af forskningsspørgsmålet vides, hvor meget data, der er nødvendig for at opnå forskellige nøjagtigheder.

UbiMusic giver brugeren mulighed for at aktivere sporing, hvorefter musikken følger brugeren i de konfigurerede rum. Konfigureringen laves selv af brugeren ved først at indsamle data i de ønskede rum og derefter koble de ønskede højtalere og rum sammen. Afspilningen foregår over Spotify og bruger Spotifys API, så valget af musik kan styres i Spotifys egen app, hvis det ønskes. Dette illustrerer det videre potentiale for dette projekt. At det er muligt at lave et nøjagtigt IPS, gør at flere forskellige services kan gøre brug af dette. Derved bli-

ver det ikke kun en sømløs musikoplevelse, men en sømløs oplevelse.

I kapitel 2 er der redegjort for relateret arbejde indenfor IPS samt musiksystemer, hvilket bruges i analysen i kapitel 3. I kapitel 4 er det beskrevet, hvordan arkitekturen og evalueringen er designet, hvorefter der i kapitel 5 beskrives den nøjagtige implementering af dataindsamlingen, træningen af modellerne, brugen af dem i UbiMusic samt implementeringen af UbiMusic. I kapitel 6 er konfidensintervaller på baggrund af indsamlet data beskrevet, hvorfra der er evalueret på forskningsspørgsmålet. I kapitel 7 er der forslag til videre arbejde, inden der i kapitel 8 fremgår den overordnede konklusion af projektet.

1.1 VIDEO DER DEMONSTRERER BRUGEN AF UBIMUSIC

https://youtu.be/I3WIKdzxNgU

2 RELATERET ARBEJDE

Relateret arbejde er opdelt i fire dele. Den første del handler om bestemmelse af positioner og metoder til at gøre dette indendørs. Derefter er der en gennemgang og oversigt over forskellige positioneringssystemers nøjagtigheder efterfulgt af nogle eksisterende musiksystemer og overvejelser omkring disse systemer. I den sidste del er der gennemgået forskellige ML modeller og deres nøjagtigheder.

2.1 BESTEMMELSE AF POSITIONER

For at kunne bestemme positioner bruges forskellige teknikker, hvoraf særligt GPS er velkendt. Til indendørs brug er GPS ikke optimal, da der ikke er en direkte linje imellem modtageren og satellitterne, og der ikke tages højde for flere etager [11]. GPS angiver en fysisk position, hvorimod der til IPS med fordel kan bruges en symbolsk position [11, 12]. En symbolsk position angiver, hvor noget er i forhold til noget andet. Det kan eksempelvis være i køkkenet eller stuen fremfor en positionsangivelse med længde- og breddegrader. Til indendørs positionering bruges flere forskellige metoder, hvoraf tre ofte brugte metoder er trilateration, proximity og fingerprinting [11, 18, 20].

2.1.1 *Trilateration*

Trilateration afgør en position ud fra afstanden til mindst tre punkter, hvis position allerede er kendt. Dermed kan en position relativt til punkterne bestemmes. Til trilateration bruges metoder som Recieved Signal Strength Indication (RSSI) Distance Estimation, Time of Flight (ToF) og Time of Arrival (ToA) [18]. En afstand til en enhed kan ændres proportionalt med enhedens udsendte RSSI værdi, hvorfor positionen kan afgøres ved tre eller flere enheder. ToF og ToA afgør positionen på baggrund af, hvor lang tid det tager for et signal at komme fra afsenderen til modtageren. Hertil kan bruges lyd, radiobølger og lys. Der vil som regel bruges en ekstern sensor til at registrere signalerne, da sensorer i mobiltelefoner ikke er konfigureret til at være præcise nok [18].

2.1.2 *Proximity*

Proximity angiver en position ud fra en eller flere kendte punkter, hvis position kendes nøjagtigt. Når der modtages signal fra et kendt punkt, angives positionen som værende nær dette punkt. Hvis der

modtages signal fra flere punkter, bruges der oftest det punkt med det stærkeste signal [20]. Denne form for IPS er velegnet til at afgøre, om en person er i et bestemt rum eller ej [11]. Ved infrarød stråling eller Radio Frequency Identification bruges proximity ofte som positioneringsmetode [20].

2.1.3 *Fingerprinting*

Fingerprinting afgør en position ud fra tidligere målte data. Den opdeles i en offline fase og en online fase [11]. I den offline fase laves en oversigt over positioner, og hvilke signalværdier, der modtages på de positioner. På den måde kan hver position karakteriseres ud fra unikke målte data. I den online fase sammenholdes de indkommende signalværdier med fingerprint dataene fra den offline fase, hvormed en position angives.

I den online fase kan en position bestemmes ud fra forskellige metoder herunder: k-Nearest-Neighbor (kNN), neurale netværk og SVM [20]. kNN bestemmer de k punkter, der er tættest på. Dette gøres ud fra den gennemsnitlige kvadrerede afvigelse imellem Recieved Signal Strength (RSS) og fingerprint dataene fra den offline fase. Ved brug af neurale netværk trænes netværket i den offline fase, så vægte og bias kendes, når det neurale netværk bruges i den online fase. Der bruges typisk et Multi-Layer Perceptron (MLP) netværk med et enkelt skjult lag [10, 20]. I SVM opdeles dataene i den offline fase i grupperinger, hvorved data indsamlet i den online fase bruges til at afgøre, hvilken gruppering de tilhører.

2.2 OVERSIGT OVER POSITIONERINGSSYSTEMERS NØJAGTIGHED

I Tabel 1 fremgår en oversigt over forskellige positioneringssystemer og deres nøjagtighed. Generelt beskrives i [8], at GPS har en nøjagtighed på imellem 10 og 20 meter. Til sammenligning er Wi-Fi nøjagtigt på imellem 1 og 5 meter, mens Ultra-Wideband (UWB) har en nøjagtighed på imellem 0,1 og 1 meter.

I tabellen er der angivet forskellige systemer, der gør brug af Wi-Fi, UWB, Bluetooth Low Energy (BLE) og Global System for Mobile Communications (GSM). RADAR er et system udviklet af Microsoft, der med en nøjagtighed på imellem 3 og 5 meter afgør positioner ved at bruge Wi-Fi signaler [2, 3, 20]. De var pionererne indenfor brug af RSS til at afgøre positioner indendørs, og det er siden blevet videreudviklet i systemet Horus [30].

RADAR bestemmer positioner ved at bruge trilateration og en metode, de kalder for Nearest Neighbours in Signal Space, som bestem-

mer den euklidiske afstand ud fra RSS [2]. Når systemet er i gang bruges en algoritme til at afgøre sandsynlighederne for den næste position ud fra tidligere positioner [3]. De beskriver denne algoritme som Viterbi-like, hvilket er en metode fra Hidden Markov Models, der hører under ML [2, 5]. Viterbi decoding bruges til at bestemme den mest sandsynlige sekvens af skjulte tilstande for en given række af observationer [5].

Systemet ArrayTrack bruger også Wi-Fi signaler og hævder at være *"[...] the first time ubiquitous real-time, fine-grained location available on the mobile handset."*[30, s. 71] Ved at bruge Angle of Arrival (AoA) opnår de en median nøjagtighed på 23 cm og en reaktionstid på 100 ms. AoA er en trianguleringsmetode, hvor positionen udregnes på baggrund af signalernes vinkler [18]. Et system, der ikke bruger Wi-Fi signaler, men derimod BLE signaler, blev beskrevet i 2020 af blandt andet T. Dinh [9]. De brugte fingerprinting til at afgøre positioner og trilateration til at bestemme positioner i den offline fase. Deres mest nøjagtige metode havde en gennemsnitlig nøjagtighed på 0,8 meter.

Det sidste system, der er angivet i Tabel 1, bruger GSM signaler til at afgøre positioner ved at bruge fingerprinting. De siger selv, at de er det første system, hvis nøjagtighed kan måles med systemer, der bruger Wi-Fi signaler [26]. Deres system er testet i bygninger med flere etager og har en median nøjagtighed på imellem 2,48 og 5,44 meter.

Der findes mange andre systemer, hvor nogle eksempelvis gør brug af ultralyd eller kombinerer de forskellige typer af signaler [19, 24]. Her er fokuseret på signaler, der kan tilgås i de fleste telefoner.

TEKNOLOGI	METODE	NØJAGTIGHED	SYSTEMNAVN	REFERENCE
GPS	TDoA	10 - 20 m		[8]
Wi-Fi	Fingerprinting	1 - 5 m		[8]
UWB	ToA, TDoA	0,1 - 1 m		[8]
Wi-Fi	Fingerprinting, kNN	3 - 5 m	RADAR	[2, 3]
Wi-Fi	AoA	23 cm	ArrayTrack	[30]
BLE	Fingerprinting, trilateration	0,8 m		[9]
GSM	Fingerprinting, kNN	2,48 - 5,44 m		[26]

Tabel 1: Oversigt over forskellige positioneringssytemer. Tabellen angiver hvilke nøjagtigheder, der kan opnås ved positioneringssystemer, der bruger forskellige typer af signaler. Derudover er angivet nogen konkrete positioneringssystemer og deres nøjagtigheder.

2.3 MUSIKSYSTEMER

Der findes forskellige musiksystemer, der over et netværk kan afspille musik på en eller flere højtalere i flere forskellige rum. Eksempler på dette er Sonos, Airplay og Chromecast Audio. Alle tre har den begrænsning, at de ikke understøtter Context-Aware (CA) behandling af lyd [17]. CA er, når systemer tilpasser sig situationsbestemte ændringer for derved at forbedre interaktionen imellem brugeren og systemet [15].

Ændringer af musikvaner og integrationen af trådløse netværk i de fleste hjem har ifølge [28] åbnet for muligheder indenfor musikafspilning. Det beskrives som ubiquitous music af [28] og pervasive audio playback af [17]. De beskriver begge, hvordan musikken nu har mulighed for at følge med brugeren. I [17, s. 532] skriver de, at *"Pervasive audio playback aims to blur the boundary between the physical world and the content."*

Eksempler på musiksystemer, der gør brug af CA, er AirPlayer [28] og Audio as a Service (A3S) [17]. Derudover ser [23] på forskellige måder at justere lyd og billeder ud fra personers tilstedeværelse og afstand til objekter. Gennemgående for dem er, at de inddeler positioner i zoner, men måden, hvormed de bruger zonerne, er forskellige. I både A3S og [23] bruges zonerne til at afgøre en distance til en genstand, mens det i AirPlayer angiver et område som eksempelvis et rum, hvori der kan afspilles musik.

AirPlayer bygger ovenpå Apple Airplay platformen og giver brugeren mulighed for at lade musikken følge sig [28]. Systemet inddeler en bygning i zoner, hvor hver zone angiver en række positioner. Brugeren har to muligheder i forhold til ubiquitous music; location og movement. Location afgør den nuværende zone og afspiller musik i den zone. Det samme gør movement, men er den slået til, følger musikken også brugeren, hvis denne bevæger sig ind i en ny zone. AirPlayer afgør positioner ud fra modtagne Wi-Fi signaler på den enhed, hvor AirPlayer er installeret. De skriver, at *"The goal is to explore the use of proxemic interaction as a more implicit form of remote controlling a multi-room music system."*[28, s. 155]. Idet AirPlayer afgør i hvilken zone, brugeren er, og hvornår der skiftes zone, kan brugeren konstant interagere med højtaleren i samme rum som brugeren selv.

Ligesom AirPlayer skifter A3S også højtaler i takt med, at brugeren bevæger sig, men A3S mener, at et generelt problem med de eksisterende systemer er, at der ikke tages højde for højtalernes position [17]. Det resulterer i, at musikken somme tider er for høj og andre gange er for lav. Derved adskiller A3S sig fra AirPlayer ved at regulere

højtalernes lydstyrke ud fra brugerens afstand til højtalerne. Til dette inddeles afstanden til en højtaler i tre zoner. Derved stiger lydstyrken, når en bruger bevæger sig væk fra en højtaler for på et tidspunkt at aftage, når en ny zone er nået, og brugeren er kommet for langt væk fra højtaleren. I det andet forsøg beskrevet i [23] varieres lydstyrken også, men dette gøres ud fra en fastlagt model med 10% per halve meter. A3S bruger zoner, da afstandende er forskellige afhængigt af rummenes karakteristika [17].

2.3.1 *Flere brugere*

Til forskel fra A3S og AirPlayer fokuserer [23] på tilstedeværelse ved brug af kameraer, der kan afgøre, hvornår en person er til stede, og hvor mange personer, der er til stede. I deres tredje forsøg afgør de musikgenren, der skal afspilles, på baggrund af antallet af personer tilstede [23]. Hverken A3S eller AirPlayer tager højde for flere personer, hvilket ifølge Körber er forkert [16]. Ifølge ham skal intelligente miljøer understøtte flere brugere. Körber er medforfatter til A3S, og i [17] konkluderes, at de vil arbejde videre med understøttelse af flere brugere. De ønsker, at brugere skal have mulighed for både at lytte til det samme og forskelligt indhold. De adskiller sig derved fra [23], der afspiller det samme til alle brugere, men varierer genren ud fra antallet af brugere.

Dette er nogle forskellige systemer og overvejelser omkring musiksystemer, der bruger IPS til at afgøre brugerens position og afspille musik bestemte steder. Til fælles har de deres fokus på CA systemer, der giver brugeren en ubiquitous og pervasive lydoplevelse.

2.4 MACHINE LEARNING

SVM og neurale netværk er tidligere blevet brugt til understøttelse af IPS. I [4] blev der opnået en nøjagtighed på 85,6% for neurale netværk og 80,5% for SVM modeller, hvorimod [6] havde en nøjagtighed på 90,9% for SVM modeller og 94,4% for neurale netværk.

I [4] brugte SVM modellerne en polynomiel kernel, og de neurale netværk var implementeret som stacked autoencoders, der var trænet i to faser. Den første fase var unsupervised med Rectified Linear unit aktiveringsfunktioner, mens den anden fase var supervised med sigmoid aktiveringsfunktioner. [6] brugte derimod et supervised MLP neuralt netværk, hvor de valgte aktiveringsfunktioner ikke fremgår.

Mængden af data og antallet af rum varierede markant imellem de to, idet [4] havde 5.156 datapunkter til 162 rum fordelt på fire etager, hvorimod [6] havde 1.199 datapunkter fordelt på 14 rum på samme

etage. I [29] er der også undersøgt nøjagtigheden af IPS understøttet af SVM og neruale netværk. De brugte 18.480 datapunkter og opnåede en nøjagtighed på 96,0% for neurale netværk med stacked autoencoders og 95,5% for SVM modeller.

For de nævnte artikler er der udelukkende brugt Wi-Fi signaler, men ifølge [21] er dette også mere nøjagtigt end BLE signaler for SVM modeller, der bruger en Radial Basis Function (RBF) kernel. De har bestemt nøjagtigheden til 86,7%, når der bruges Wi-Fi signaler, og 84,8% når der bruges BLE signaler. Evalueringen er foretaget på baggrund af 106 datapunkter og tre rum, hvor rummenes placering ikke fremgår. I sektion 6.6 er dette projekts nøjagtigheder evalueret op imod nøjagtighederne angivet i denne sektion.

3 ANALYSE

Til at demonstrere brugen af et IPS til en sømløs musikoplevelse er appen UbiMusic blevet udviklet. Denne bruges på en telefon, der modtager Wi-Fi signaler fra eksisterende access points og beregner positionen herudfra. Ifølge [14] er UbiMusic dermed et terminal IPS ligesom både [17] og [28]. Det betyder, at hvis en bruger lægger telefonen fra sig, angives positioner stadigvæk ud fra telefonen. Dette blev hverken betegnet som et problem i [17] eller [28].

Selvom [16] mener, at alle CA lydsystemer skal håndtere flere brugere, som det er beskrevet i under-sektion 2.3.1, er der i dette projekt brugt samme tilgang som i [17]. Først fokuseres der på et system til en bruger, hvorefter det kan udvides til flere brugere.

3.1 POSITIONERING

Dette projekts IPS bruger fingerprinting og minder dermed om RADAR, men afviger i valg af metoder. Af sektion 2.2 fremgår det, hvordan RADAR i den online fase brugte ML og en Viterbi-like algoritme til at bestemme positioner. I dette projekt er der også brugt ML, men i stedet for en Hidden Markov Model er der brugt neurale netværk og SVM modeller. Årsagen til dette er de høje nøjagtigheder opnået i tidligere forskning, som det er beskrevet i sektion 2.4. Derudover beskrives SVM i [20] som en ny og lovende teknik indenfor fingerprinting, hvorfor dennes potentiale ønskes undersøgt.

ArrayTrack hævder, som det fremgår af sektion 2.2, at være den første nøjagtige lokaliseringsmetode, og de hævder dermed at være bedre end RADAR. Hvorfor der alligevel er valgt en tilgang nærmere RADARs i dette projekt skyldes, at det ønskes undersøgt, hvilket potentiale der er i at bruge ML indenfor fingerprinting.

Både ArrayTrack, RADAR og resten af Tabel 1 bruger afstandsnøjagtigheder. I dette projekt har fokus været på, hvorvidt en bruger er i det rigtige rum. Derfor er der ikke blevet evalueret ved at bruge afstandsnøjagtigheder, og der er heller ikke blevet sammenlignet med nøjagtighederne i Tabel 1 eller sektion 2.2. Til gengæld er en evaluering af modellernes nøjagtigheder i forhold til at angive de rigtige rum beskrevet i kapitel 6. Dette er i sektion 6.6 sammenholdt med nøjagtighederne, der fremgår af sektion 2.4.

3.1.1 *Håndtering af rum uden en højtaler*

I [28] blev det beskrevet, hvordan brugere oplevede og undrede sig over, at musikken fortsatte i det tidligere rum, når de gik ind i et rum uden en højtaler. Da outputtet fra et neuralt netværk med en softmax aktiveringsfunktion imellem det skjulte lag og outputlaget er sandsynligheder, er det blevet undersøgt, om problemstillingen kan løses ved at afgøre en mindste sandsynlighed, hvorunder musikken skal sættes på pause. Dette vil dermed svare til, at en person har bevæget sig ind i et rum, hvor der ingen højtaler er.

At neurale netværk oftest trænes ved at bruge både vægte og bias imellem samtlige noder, kan blive en udfordring for denne funktionalitet. Hvis en bruger har tændt sit system, men ingen kendte Wi-Fi access points bliver opfanget, vil et neuralt netværk, der er trænet med bias, returnere en prædiktion på baggrund af de trænede bias afhængigt af den valgte aktiveringsfunktion imellem inputlaget og det skjulte lag. En model med en identitets aktiveringsfunktion, der er trænet uden bias, har derimod nul i samtlige indgange til outputlaget og vil dermed prædiktere en lige stor sandsynlighed for alle klasser. Dette kan bruges til at afgøre, om et system er udenfor rækkevidde af sit trænede område ved eksempelvis at have bevæget sig udenfor hjemmet. En evaluering af dette fremgår af sektion 6.3.

Eftersom SVM modeller outputter en klasse og ikke en sandsynlighedsfordeling, kan det ikke løse udfordringerne, der er beskrevet i [28], på samme måde som neurale netværk. Hvis en bruger derimod både indsamler data i de rum, hvor musikken skal afspilles, og i rummene hvor der ikke skal afspilles, kan trænede SVM modeller bruges. Derfor ønskes neurale netværk med og uden bias samt SVM modeller med og uden rum uden højtalere sammenholdt for at afgøre deres indbyrdes fordele.

3.1.2 *Bevægelse fra rum til rum*

Hvis fokus udelukkende er at detektere, hvornår en bruger har bevæget sig fra et rum til et andet, giver det mening at skifte højtaler, så snart en Wi-Fi scanning påviser rumskite. Dette giver den lavest mulige latency, da højtalerne skifter hurtigst muligt. Dette projekts fokus er dog at give en sømløs oplevelse for brugeren. At skifte højtaler efter hver detektering giver en større sandsynlighed for, at der skiftes frem og tilbage imellem to højtalere, hvis systemet er i tvivl om, hvilket rum brugeren befinder sig i. Præcisionen af brugerens position mindskes derfor med en lavere latency.

En måde at imødekomme dette er ved først at skifte højtaler efter

to på hinanden følgende prædiktioner for det samme rum. Systemet har dermed mulighed for at være i tvivl imellem to rum og slet ikke skifte til det nye rum. Omvendt set når der først er skiftet til den nye højtaler, er der en lavere sandsynlighed for at skifte væk fra rummet, førend brugeren forlader det.

Hver Wi-Fi scanning tager omkring seks sekunder, men Android version 8.0 og opefter har en begrænsning på fire Wi-Fi scanninger indenfor hvert andet minut. Hvis det vælges at fokusere på lavest mulig latency, kan der derfor skiftes højtaler efter cirka seks sekunder. Fokuseres der derimod på en højere præcision, kan der ende med at gå over to minutter, før der skiftes rum, hvis der skal bruges mere end fire Wi-Fi scanninger.

Hvornår der skal skiftes rum og dermed højtaler, er derfor en opvejning af, om lav latency eller høj præcision vægtes højest. I dette projekt er der valgt først at skifte højtaler efter to på hinanden følgende prædiktioner for det samme rum. Dette skyldes, at det for musik vurderes at bidrage mest muligt til en sømløs oplevelse, hvis positionen er mere præcis, og der dermed ikke skiftes unødigt imellem højtalere fremfor at have den lavest mulige latency. I sektion 6.5 er der evalueret på konsekvenserne af, hvornår der skiftes højtaler.

3.2 VALG AF SERVICE

Ligesom AirPlayer brugte Apple Airplay platformen, er der i dette projekt taget udgangspunkt i Spotifys API[1] [28]. Det giver ligesom i Apple Airplay muligheden for at styre musikken og hvilken højtaler, der afspilles på. Spotifys API er valgt til dette projekt, da der er testet ved brug af Androids styresystem.

At Spotifys API bruges, betyder, at enheder tilføjes og fjernes automatisk igennem Spotifys system, og kun de online enheder kan tilgås via projektets system. Når højtalere vælges i Spotifys egen app, kan der tilgås forskellige typer af højtalere og forskellige mærker. Når der vælges højtalere over Spotifys API, er der nogle begrænsninger. Der kan hverken vælges højtalere fra Sonos eller Google Cast højtalere.

Ifølge [17] bør nye enheder opdages og integreres automatisk, da enheder kan komme online og offline på vilkårlige tidspunkter. I dette projekt opdages højtalere automatisk ligesom i AirPlayer og A3S, men brugeren skal angive, hvilken højtaler, der hører til hvilket rum. Når først højtaleren er angivet til et rum, bliver den automatisk opdaget og integreret via Spotifys API, når den kommer online.

1 https://developer.spotify.com/documentation/web-api/, besøgt 27. februar 2023

Ønskes der sidenhed andre services til systemet, kan dette tilføjes. Det kunne eksempelvis være andre højtalere, termostater eller lamper. Både Sonos højtalere, Philips Hue pærer og Google Home enheder kan tilgås via APIer. Eftersom fokus for dette projekt har været på en sømløs musikoplevelse, har de tilgængelige højtalere været tilstrækkelige for at evaluere dette.

4 DESIGN AF ARKITEKTUR OG EVALUERING

I kapitel 3 er nogle af designvalgene i dette projekt beskrevet, og de er uddybet her. Derudover er beskrevet designet af evalueringen.

4.1 FINGERPRINTING

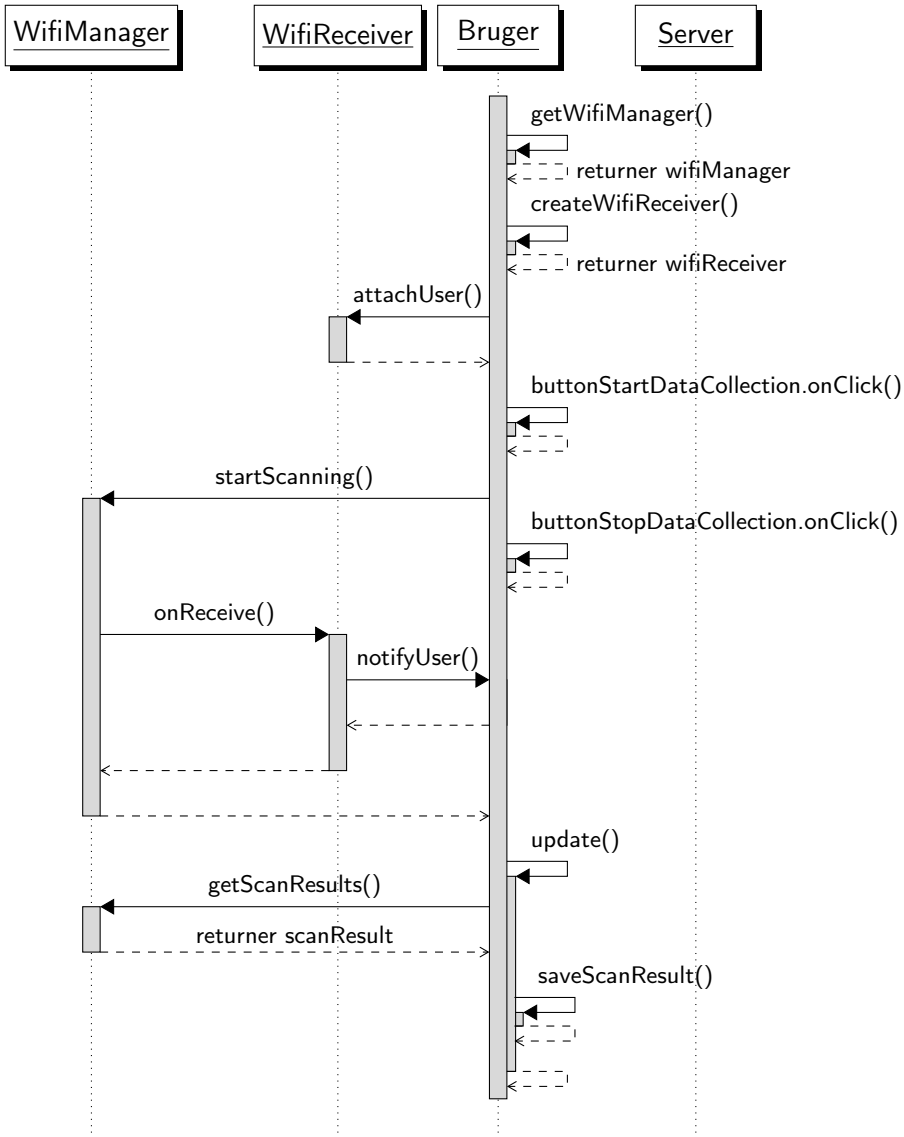
Der er i projektets IPS en offline fase og en online fase. I den offline fase indsamler brugeren data. Ud fra disse data trænes to typer af ML modeller, der bruges i den online fase. I denne fase har brugeren taget systemet i brug, og modellerne prædikterer i hvilket rum, brugeren befinder sig, så der afspilles musik i det rigtige rum.

Efter indsamlingen af data i den offline fase har brugeren en oversigt over samtlige tilgængelige højtalere og samtlige tilgængelige rum. Et rum svarer til zoner i [28]. Derved afgør brugeren, hvilken højtaler der skal afspille i de trænede rum. Dette kan altid ændres i den online fase, hvis brugeren ønsker det.

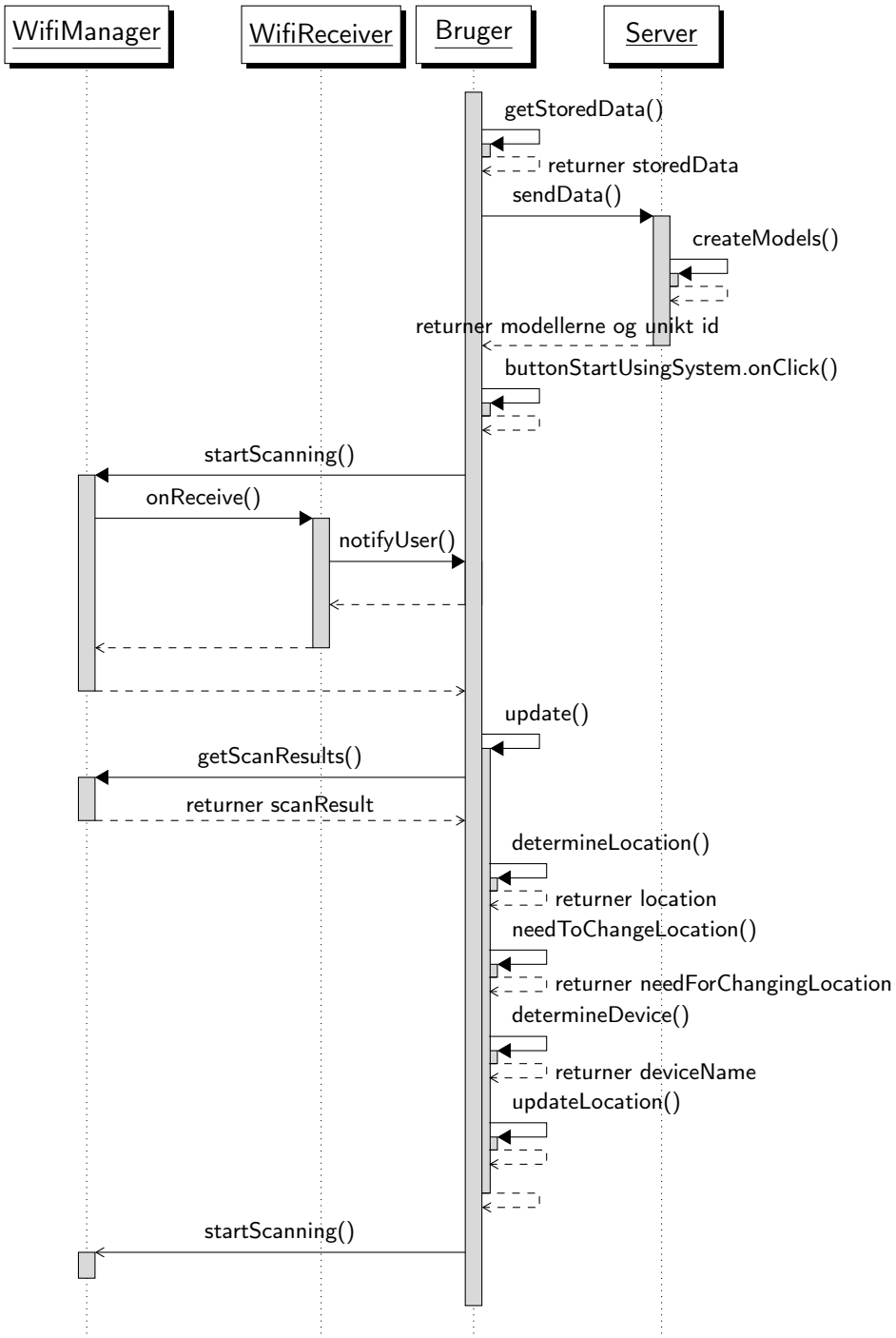
Træningen af modellerne foretages på en ekstern server, så brugeren ikke skal vente på, at telefonen gør dette. Når modellerne er færdige, sendes de tilbage til telefonen over netværket, så systemet kan bruges i den online fase. Udover at træne modellerne bliver de også testet, hvorfor serveren kan melde tilbage til brugeren, at der er brug for mere data, hvis nøjagtigheden af de trænede modeller er for lav. Dette er for at give brugeren den bedst mulige oplevelse.

Sammen med modellerne sendes et unikt id. Dette er synligt for brugeren, så en anden bruger i samme hjem kan angive dette id i systemet, hvorefter det sendes til serveren, og de allerede trænede modeller fås fra serveren. Den nye bruger kan også vælge at indsamle sine egne data og træne sine egne modeller.

I Figur 1 og Figur 2 ses to sekvensdiagrammer, der viser den offline fase for en bruger, der indsamler data for derefter at sende det til serveren og modtage trænede modeller. I dette projekt er der hverken implementeret featuren, der inkluderer brugen af et unikt id, eller featuren, hvor systemet informerer brugeren om en for lav nøjagtighed.



Figur 1: Sekvensdiagram for en bruger, der indsamler data. Umiddelbart efter indsamlingen er startet, trykker brugeren på knappen STOP DATAINDSAMLING, hvorfor der kun laves en scanning.



Figur 2: Sekvensdiagram for en bruger, der træner modeller ved at bruge en server, og som efterfølgende bruger modellerne til at afgøre i hvilket rum, der skal afspilles musik.

4.2 MACHINE LEARNING MODELLER

I dette projekt er der fokuseret på, hvilke muligheder et IPS, der gør brug af SVM modeller og neurale netværk, giver for en sømløs musikoplevelse. De to metoder er begge supervised learning, da de korrekte labels er kendt under træningen. Metoderne har dog forskellige karakteristika, og her er uddybet deres individuelle design.

Når SVM bruges til multiklasse klassifikation, hvor der er mere end to klasser, er der to forskellige muligheder[1]. Enten en-til-en tilgang eller en-til-resten tilgang. Forskellen er antallet af binære klassifikationer, da der i en-til-en tilgang for hver klasse trænes mod hver enkelt af de andre klasser, mens der i en-til-resten tilgang trænes mod resten af klasserne på en gang. I dette projekt bruges en-til-en tilgang, da dette øger antallet af binære klassifikationer.

I under-sektion 2.1.3 er det beskrevet, hvordan neurale netværk ofte laves som et MLP netværk med et enkelt skjult lag, hvilket også er gjort i dette projekt. Antallet af inputnoder er antallet af forskellige Wi-Fi access points, der er tilgængelige i de trænede data. De kan derfor variere selv for den samme bruger, hvis denne træner modeller af flere omgange. Antallet af outputnoder er antallet af forskellige rum og varierer derfor ikke for den samme bruger, medmindre vedkommende ændrer på de brugte rum.

Imellem inputlaget og det skjulte lag bruges en sigmoid aktiveringsfunktion, mens der imellem det skjulte lag og outputlaget bruges en softmax aktiveringsfunktion. Softmax bruges, så outputnoderne er en sandsynlighedsfordeling, der summerer til en. Det rum, der svarer til noden med den højeste sandsynlighed, bliver valgt som modellens prædiktion. Som loss function bruges Cross Entropy Loss (CEL), da denne ofte bruges ved multiklasse klassifikation for neurale netværk[2].

$$\begin{aligned}
\text{Sigmoid:} \quad & \sigma(x) = \frac{1}{1+e^{-x}} \\
\text{Softmax:} \quad & S(z)_i = \frac{e^{z_i}}{\Sigma_{j=1}^{k} e^{z_j}}, \quad \text{for k klasser} \\
\text{Cross Entropy Loss:} \quad & L_{CE} = -\Sigma_{i=1}^{n} t_i \log(p_i)
\end{aligned}$$

For n klasser hvor t_i er det korrekte label, og p_i er sandsynligheden givet af softmax for den i'te klasse.

1 https://www.baeldung.com/cs/svm-multiclass-classification, besøgt 11. april 2023

2 https://github.com/kaspergl/ML22/blob/main/notes/CrossEntropy.ipynb, Professor Kasper Green Larsen, besøgt 27. februar 2023

4.3 SERVICES

Som det fremgår af sektion 3.2, bruges Spotifys API i dette projekt til håndtering af musik og online højtalere. Det bruges til afspilning af musik og til at håndtere skift af højtalere, når brugere bevæger sig imellem rum. Funktionaliteten er designet, så den ligner movement fra AirPlayer, der er beskrevet i sektion 2.3. Derved har brugeren mulighed for at slå funktionen til, når musikken skal følge brugeren fra rum til rum, og slå funktionen fra når afspilingen skal blive i det pågældende rum.

I sektion 3.2 er forskellige services beskrevet, som efterfølgende kan tilføjes til systemet. For at opdele de forskellige services så de fungerer isoleret fra hinanden, kan der bruges en publish-subscribe arkitektur. Dette giver også den fordel, at de forskellige services nemt kan kobles til og fra. Arkitekturen gør det muligt, at flere maskiner kan kommunikere over en broker. De kan publicere til brokeren, og alle, der har subscribet til nøjagtig den broker, modtager det publicerede. Dette gør det fleksibelt i forhold til at koble services til og fra.

Hvis der bruges flere services, kan de hver især være koblet op til en bestemt broker ved at subscribe til denne. Som en del af konfigureringen af en bruger, kan denne i systemets brugergrænseflade blive bedt om at angive de ønskede services. Herefter kan systemet automatisk koble op til de ønskede services ved at subscribe til de brokere, de ønskede services har subscribet til.

4.4 FLERE BRUGERE

Tages systemet i brug af flere brugere i samme hjem, kan det med fordel udvides til at kunne håndtere, når disse brugere går ind i samme rum. Til dette kan publish-subscribe arkitekturen også med fordel bruges. Hvad der skal gøres i dette scenarie, afhænger af den pågældende service, men det giver muligheden for, at brugernes system kan reagere på, at der er flere brugere i samme rum.

Figur 3 viser et sekvensdiagram over interaktionen imellem to brugere, når de ender i samme rum. Dette diagram tager udgangspunkt i, at begge brugere har trænede modeller og er i den online fase. Beskederne, der er publiceret til brokeren, har den samme struktur, og ud fra elementerne ved brugernes system, hvordan de skal reagere.

Ved at bruge en publish-subscribe arkitektur kan de enkelte services nemt tilføjes, udskiftes eller helt kobles fra afhængigt af den ønskede funktionalitet. Flere brugere og services, der gør brug af en sådan

arkitektur, har ikke været fokus for dette projekt. Det er derfor udelukkende overvejelser til videre arbejde.

4.5 DATAINDSAMLING

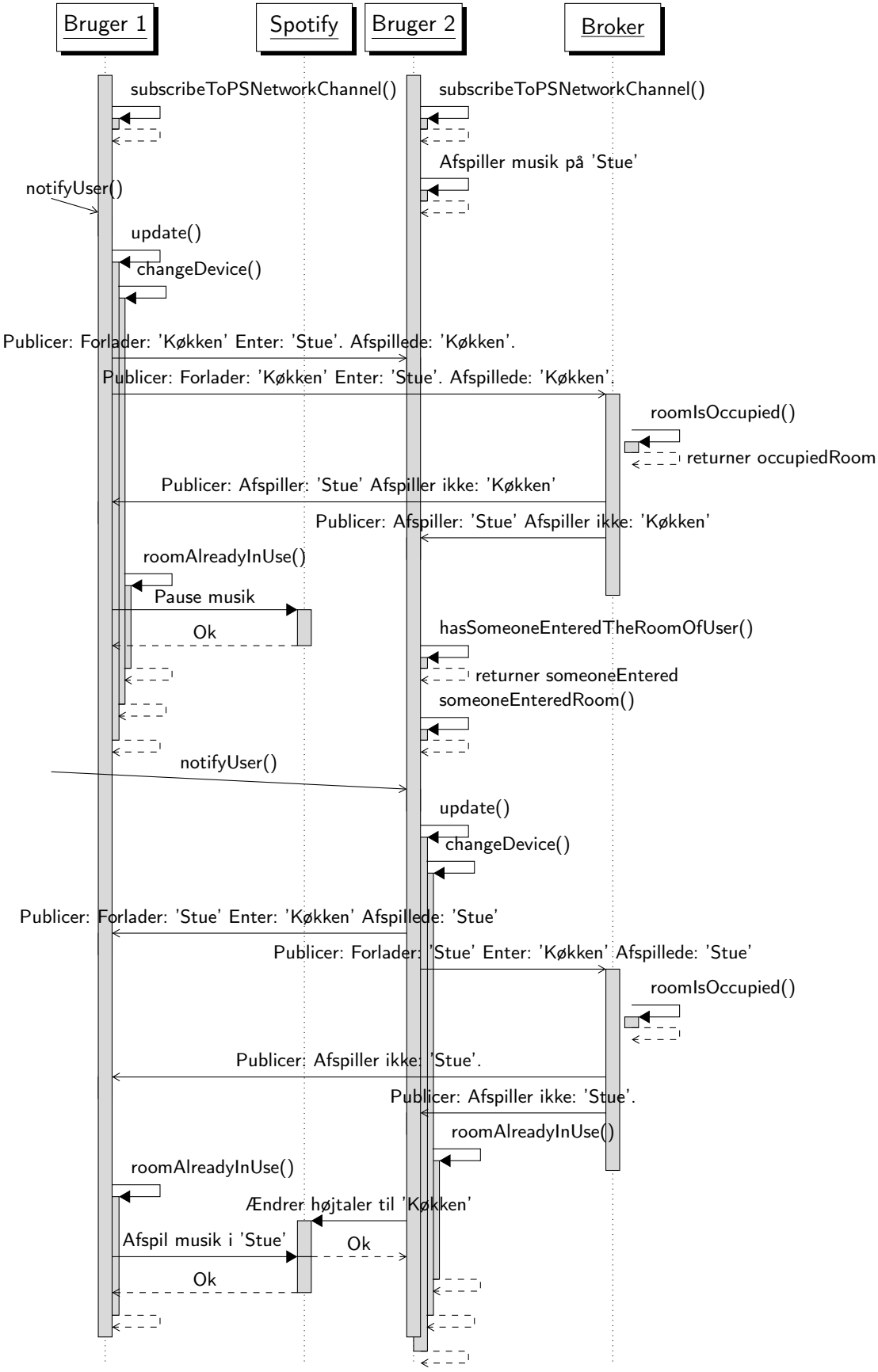
Dataindsamlingen foregik via samme telefon, som systemet er testet på. Dataene er indsamlet i samme højde, som det kan forventes for en telefon, der holdes i hånden, samt på steder hvor det kan forventes, at telefoner bliver lagt fra sig. Der er testet i et rækkehus med fokus på fire rum: Kontor, stue, køkken og entré. Kontoret er på første sal, mens stue og køkken til sammen udgør et åbent køkken-alrum, der ligger i stueetagen ligesom entréen. Kontoret er placeret omtrent midt over stuen og entréen. Figur 4 viser en plantegning over rækkehuset og rummenes inddeling. Den blå stiplede linje indikerer adskillelsen imellem stue og køkken. Entréen er det eneste rum, hvor der ikke er en højtaler i.

Indsamlingen af data er foretaget om formiddagen og aftenen. Det antages, at flere er hjemme om aftenen, hvorfor færre personer og enheder forventes at bruge Wi-Fi om formiddagen. Årsagen til indsamlingen på disse tidspunkter er for at undersøge, om antallet af enheder, der bruger Wi-Fi, har en indvirkning på Wi-Fi signalernes styrke og dermed modellernes nøjagtigheder. Evalueringen af dette fremgår af sektion 6.2.

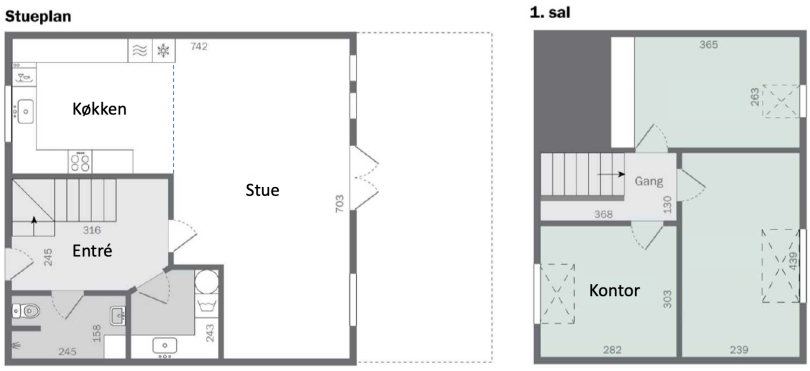
Konkret er der indsamlet data i 45 minutter i hvert rum imellem klokken 9 og 12 og igen imellem klokken 17.30 og 20.30. Under indsamlingen af data er der skiftet rum regelmæssigt, så eventuelle spidsbelastninger er fordelt imellem rummene. Der er indsamlet om formiddagen fire gange og tre gange om aftenen. Derudover er der indsamlet datapunkter i 90 minutter på Aarhus Universitet for at evaluere modellernes nøjagtigheder, når datapunkter ikke indeholder kendte netværk. Evalueringen af dette fremgår af sektion 6.3.

4.6 EVALUERING

Projektets evaluering har fokuseret på fem ML modeller, hvoraf to er SVM modeller, og tre er neurale netværk. En beskrivelse af modellerne fremgår af Tabel 2. For begge typer af ML modeller er der testet på både tre og fire rum. Som det fremgår af under-sektion 3.1.1, er modellerne lavet på denne måde for at evaluere, om det kan løse problemstillingen fra [28]. At der også er lavet et neuralt netværk med fire klasser, er for at evaluere dens nøjagtighed sammenlignet med både neurale netværk trænet på tre rum og SVM modellerne. Evalueringen af modellerne fremgår af sektion 6.4.



Figur 3: To brugere er i samme hjem og går ind i samme rum. Bruger 2 afspiller til at starte med i stuen, hvor bruger 1 kommer ind. Når bruger 2 forlader rummet, begynder bruger 1 at afspille musik. Bemærk at nogle metodekald fra de tidligere sekvensdiagrammer er undladt i dette diagram for at spare plads.



Figur 4: Plantegning over rækkehuset, hvor dataene er indsamlet. Den blå stiplede linje indikerer adskillelsen imellem køkken og stue.

MODELNAVN	TYPE ML	RUM	BIAS
SVM3	SVM	Stue, Køkken, Kontor	
SVM4	SVM	Stue, Køkken, Kontor, Entré	
NN3-B	Neuralt netværk	Stue, Køkken, Kontor	Ja
NN3-UB	Neuralt netværk	Stue, Køkken, Kontor	Nej
NN4-UB	Neuralt netværk	Stue, Køkken, Kontor, Entré	Nej

Tabel 2: Oversigt over de fem modeller, der er evalueret på.

NN3-B og NN3-UB er både evalueret i forhold til, hvor nøjagtige de er til at prædiktere tre rum, og hvor nøjagtige de er til at prædiktere fire rum. Dette er gjort for, at de kan sammenlignes med SVM modellerne. Når deres nøjagtigheder i fire rum er testet, er de angivet som NN3-B-P og NN3-UB-P. For dem begge er der bestemt en tærskel for, hvornår en bruger forventes at være udenfor et rum med højtalere. Dette angives som lokalet *Intet rum* og får musikken sat på pause.

4.6.1 *Nøjagtighed*

Alle modellerne er trænet med brug af cross validation i et forsøg på at hæve nøjagtigheden ved at bruge færre datapunkter. Dette er gjort for at mindske den tid, brugerne skal bruge på dataindsamling. Til dette er udvalgt 80% af dataene til træning og 20% til test. Testdataene er også brugt til at bestemme tærsklen for NN3-B-P og NN3-UB-P. Dette er gjort ved at tage de datapunkter, som modellen klassificerer korrekt og bestemme 5-fraktilen for den højeste sandsynlighed givet fra outputlaget. Denne værdi er valgt for, at modellen kun klassifice-

rer 5% af de korrekt klassificerede datapunkter som et rum uden en højtaler. Eftersom systemet er designet til først at skifte højtaler efter to på hinanden følgende prædiktioner for det samme rum, er den forventede sandsynlighed for, at musikken bliver sat på pause uden grund 0,25%.

Eftersom de indsamlede datasæt er stort set samme størrelse, er der ikke beregnet et vægtet gennemsnit i forhold til antal datapunkter i hvert datasæt. Den eneste undtagelse er for evalueringen af modellernes nøjagtigheder på data udenfor træningsområdet, der er beskrevet i sektion 6.3. Størrelsen på dette datasæt afviger markant fra resten, hvorfor modellernes gennemsnitlige nøjagtigheder i denne evaluering er beregnet som et vægtet gennemsnit for datasættene i forhold til antal datapunkter i hvert datasæt.

Samtlige modeller er evalueret på, hvor nøjagtige de kan blive afhængigt af mængden af data, en bruger indsamler i den offline fase. Det indsamlede data er derfor inddelt i mængder svarende til 5, 10, 15, 30 og 45 minutters dataindsamling i hvert rum. Den udvalgte mængde er valgt tilfældigt over al den indsamlede data for at få en repræsentativ måling for rummene. Det skyldes, at telefonen ikke blev flyttet konstant under dataindsamlingen men hvert andet minut. Evalueringen af dette er beskrevet i sektion 6.4.

De neurale netværks nøjagtigheder, ved brug af forskellige aktiveringsfunktioner, er også evalueret, og det er beskrevet i sektion 6.3. Der er evalueret på brugen af en identitets aktiveringsfunktion i forhold til en sigmoid aktiveringsfunktion imellem inputlaget og det skjulte lag. Ydermere er der evalueret på deres nøjagtigheder i forhold til at afgøre, hvornår en bruger har forladt det trænede område og dermed må forventes at have forladt sit hjem.

For modellerne, der bruger en identitets aktiveringsfunktion og ingen bias, er 5-fraktilen ikke brugt som tærskel. I stedet er der brugt tærsklen

$$T = \frac{1}{j} + \epsilon,$$

hvor j er antal klasser, og ϵ er machine epsilon. Bruges denne tærskel ved datapunkter udenfor det trænede område, har modellerne en nøjagtighed på 100%, fordi sandsynlighedsfordelingen, der outputtes fra netværket, er ligeligt fordelt imellem samtlige klasser. Hvorfor sandsynlighedsfordelingen er ligeligt fordelt, er beskrevet i undersektion 3.1.1.

4.6.2 *Dataanalyse*

Af sektion 4.5 fremgår det, at der er lavet syv forskellige dataindsamlinger på hver tre timer og halvanden times dataindsamling på Aarhus Universitet. Der er i alt indsamlet lidt over 13.000 datapunkter fordelt over en periode på seks dage i rækkehuset, der ses i Figur 4, samt knap 1.000 datapunkter på Aarhus Universitet.

Evalueringen af modellerne i kapitel 6 tager udgangspunkt i en række konfidensintervaller, der er lavet på baggrund af de indsamlede datapunkter. Datapunkterne udgør en stikprøve af den samlede mængde datapunkter, det er muligt at indsamle. Da der er en ukendt varians, er konfidensintervallerne udregnet ved at bruge en t-fordeling ud fra formlen [13].

$$\mu = \overline{x} \pm t_{n-1,\frac{\alpha}{2}} \cdot \frac{s}{\sqrt{n}} \tag{1}$$

I (1) er μ den nedre og øvre grænse af konfidensintervallet, α er signifikansniveauet, og n er stikprøvestørrelsen. $\overline{x}$ og s er henholdsvis gennemsnittet og standardafvigelsen for stikprøven, og de angives som.

$$\begin{aligned} \overline{x} &= \frac{\Sigma_{i=1}^{n} x_i}{n} \\ s &= \sqrt{\frac{\Sigma_{i=1}^{n} (x_i - \overline{x})^2}{n-1}} \end{aligned} \tag{2}$$

I (1) bruges en t-fordeling, hvorfor der i (2) divideres med $n-1$ og ikke blot n. Årsagen til at vælge en tosidet t-fordeling er, at variansen frit kan variere i intervallet $[0; 1]$, eftersom de forskellige modeller er trænet ud fra de indsamlede datasæt og er testet op imod de andre datasæt et af gangen.

Om modellernes nøjagtigheder følger en normalfordeling er uvist. Derfor skal den udvidede centrale grænseværdisætning være opfyldt for, at konfidensintervaller lavet med t-fordelingen kan forventes at være retvisende. Denne siger, at for en tilstrækkelig stor stikprøve, er gennemsnittet normalfordelt [13]. For at den er opfyldt, skal stikprøven være uafhængig og have en identisk fordeling. Dataene er simpelt tilfældigt udvalgt, og Wi-Fi data må formodes at være troværdige, hvorfor de må forventes at have en identisk fordeling. For at dataene er afhængige af hinanden, skal et access point stoppe med at fungere, så et netværk forsvinder imellem to dataindsamlinger. Det må forventes, at hvis det sker, vil netværket blive genstartet inden for samme dag. Eftersom dataindsamlingen foregik over flere dage, må dataene derfor forventes at være uafhængige.

Den udvidede centrale grænseværdisætning er dermed opfyldt, og

konfidensintervaller, der er lavet ud fra en t-fordeling med $n-1$ frihedsgrader, må forventes at være retvisende. Stikprøvestørrelsen n er antallet af forskellige datasæt, den pågældende model er testet på. Konfidensintervallerne i evaluering i kapitel 6 er lavet på et 5% signifikansniveau. Dermed kan det med en sikkerhed på 95% forventes, at den gennemsnitlige nøjagtighed for den pågældende model er indeholdt i intervallet. To konfidensintervaller siges at afvige signifikant, hvis der intet overlap er imellem dem.

5 IMPLEMENTERING

I kapitel 4 er designet af projektets system beskrevet, og på baggrund af projektets forskningsspørgsmål, der er beskrevet i kapitel 1, er der i implementeringen fokuseret på tre ting: Indsamling af data, træning af modeller og brugbarheden af disse modeller. Til dette er implementeret UbiMusic, DataBehandlingog ModelForUbiMusic.

5.1 ARKITEKTUR

Implementeringerne af både UbiMusic, DataBehandling og ModelForUbiMusic er lavet med fleksibilitet og vedligeholdelse som fokus. Dette er gjort for at gøre det lettere at tilføje ny funktionalitet ved udelukkende at tilføje software og ikke ændre i det eksisterende. I DataBehandling og ModelForUbiMusic er dette gjort for, at forskellige modeller nemmest muligt vil kunne tilføjes og blive testet. I UbiMusic har det særligt været et fokus for, at andre services eller opdateringer af musikservices nemmest vil kunne implementeres.

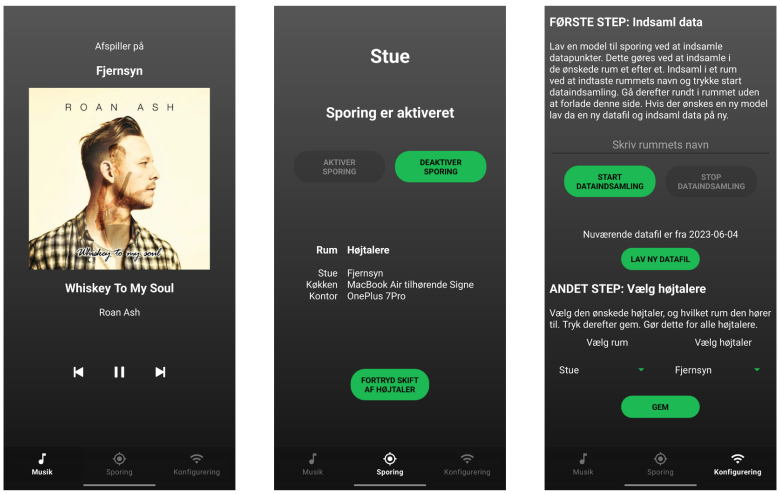
I implementeringerne er der fokuseret på SOLID principperne og særligt på, at klasser såvel som metoder udelukkende har et formål [22]. Dette kommer til udtryk i brugen af interfaces, der understøtter Open-Closed princippet og Changeability som en del af vedligeholdelse [7].

5.1.1 *Databehandling*

I databehandlingen er arkitekturen opbygget til træning og test af modeller. Der er både lavet SVM modeller og neurale netværk, og de har begge gjort brug af matrixmanipulation og udtrukket data fra filer. Undervejs i udviklingen er der fokuseret på variabilitetspunkter for at lade disse være i sin egen klasse. Dette gør sig gældende for både dataudtrækning fra filer, matrixmanipulation og ML modeller. Dette er gjort for at adskille ansvarsområder og gøre det nemmere at ændre på implementeringen af modellerne eller tilføje nye.

5.1.2 *ModelForUbiMusic*

I Figur 2 er der angivet en server, der returnerer modeller og et unikt id til en bruger. ModelForUbiMusic er implementeringen af en simpel form for denne funktionalitet. På baggrund af data træner den en SVM model ved at bruge libsvm biblioteket. Denne model gemmes i



Figur 5: Oversigt over de tre sider i appen UbiMusics Graphical User Interface (GUI). Under fanen *Musik* vises det musik, der afspilles. Under fanen *Sporing* aktiveres og deaktiveres sporing af brugeren, og under fanen *Konfigurering* konfigurerer brugeren systemet ved at indsamle data og vælge højtaler til de ønskede rum.

en fil, der efterfølgende flyttes over på den ønskede enhed og bruges i UbiMusic.

Undervejs i udviklingen er brugt 3-1-2 princippet til at bestemme variabilitetspunkter og derefter lave disse til et interface [7]. Dette kommer til udtryk i IFileSystem, IModel og IMatrixManipulation. Derved kan eksempelvis en ny modeltype nemt laves som en implementering af IModel og derefter bruges.

5.1.3 *UbiMusic*

UbiMusic er en Android app skrevet i Java i Android Studio med et minimum Software Development Kit på 26, hvilket svarer til Android 8.0 (Oreo). Dette er gjort for at kunne teste og bruge den på telefoner med et Android styresystem version 8.0 eller højere.

I Figur 5 ses de tre sider, UbiMusic indeholder, der er implementeret ved brug af fragmenter. Siden *Musik* viser den musik, der afspilles, mens siden *Sporing* bruges til at aktivere og deaktivere systemet samt se, når systemet skifter lokale. Den sidste side *Konfigurering* bruges til at guide brugeren igennem indsamlingen af data og valget af højtaler til de ønskede rum.

I udviklingen af UbiMusic er der tilsvarende brugt 3-1-2 princippet til at afgøre variabilitetspunkter. Dette kommer eksempelvis til udtryk i interfaces som IService, der implementeres af SpotifyService og IDa-

taManagement, der både implementeres af DataManagementSVM og DataManagementNN. Brugen af 3-1-2 princippet understøtter Interface Segregation princippet, idet der ikke er et generalt interface. Dette gør, at kun de nødvendige interfaces behøves at blive implementeret. I den nuværende implementering af UbiMusic bruges samtlige interfaces, men dette kan blive relevant i en videreudvikling af UbiMusic.

At SpotifyService er en implementering af IService, bidrager til at gøre UbiMusic fleksibelt for videreudvikling. Hvis der ønskes en anden musikservice eller en service, der håndterer eksempelvis lys, kan der laves en ny implementering af IService. Metoderne i interfacet er valgt, så de understøtter den nødvendige funktionalitet i UbiMusic, og hvad der må forventes at være nødvendigt, hvis en lignende app med en anden service ønskes implementeret. Det ses eksempelvis i form af metoder som changeLocation og stopService. Derved er det fleksibelt at bruge IService til forskellige services, da den ønskede funktionalitet udelukkende skal implementeres uden at ændre på anvendelsen af interfacet.

Hvis der ønskes en anden musikservice, kan dette implementeres og bruges i stedet for, men ønskes der en anden service, bør der laves nogle ændringer. Eksempelvis bør musikfragmentet laves om, så det passer til den ønskede service, ligesom det bør overvejes, om det andet step i konfigureringen bør ændres. Konfigureringsfragmentets dataindsamling og ligeledes sporingen virker uafhængigt af servicen, og de kan derfor bruges i en eventuel videreudvikling. Tilsvarende kan de ønskede modeller ændres ved at lave en ny implementering af IDataManagement.

For at bruge færrest mulige ressourcer på telefonen, er både WifiService og SpotifyService implementeret ved brug af et observer pattern [7]. Derved informerer de MainActivity, med det samme en ændring er sket, i stedet for at MainActivity skal lytte på dem.

Ved at de tre fragmenter udelukkende har en relation til MainAcitivity, der står for den bagvedliggende logik, er der brugt et facade pattern. Dette er gjort for at mindske couplingen og derved øge maintainability [7]. Ifølge Law of Demeter er dette den mest hensigtsmæssige måde at lave arkitekturen på, idet der dermed udelukkende er metodekald til direkte objekter og ikke til indirekte objekter.

For at øge maintainability yderligere er der fokuseret på høj cohesion ved at inddrage clean code [7]. Dette kommer eksempelvis til udtryk ved, at metoder og klasser har et ansvarsområde, der er tydeligt angivet i navnet.

5.2 DESIGN

UbiMusic er så vidt muligt designet ud fra Normans principper og Shneidermans gyldne regler for at opnå den bedst mulige brugervenlighed [25, 27]. Derfor er al funktionalitet synlig for brugeren ved forklaringer og knapper på de tre sider, så det er i overensstemmelse med Normans visibility princip. For at være konsistent og dermed følge Shneidermans første gyldne regel og Normans sjette princip er elementer, der relaterer sig til hinanden, implementeret i samme stil med samme farver, skrifttyper og skriftstørrelser. Farverne, der er brugt, er i overensstemmelse med Spotifys foreskrifter for at være konsistent med Spotifys egen app og andre services, der forbindes med musikafspilning via Spotify[1].

UbiMusic er implementeret, så der gives mest mulig informativ feedback til brugeren, hvilket både er et af Normans principper og Shneidermans tredje gyldne regler. Der er blevet skildret imellem information, som brugeren forventes at være interesseret i at vide over en længere periode, og information brugeren bør oplyses om en enkelt gang. Den første slags information er implementeret, så det står i UbiMusic, og brugeren kan se dette ved at gå til den relevante side. Den anden slags information er implementeret, så brugeren bliver informeret via en pop up besked. Dette er gjort for at mindske mængden af information på siderne, så det er mest muligt overskueligt for brugeren.

At noget information er at finde i UbiMusic, er for at mindske belastningen på korttidshukommelsen og dermed følge Shneidermans ottende gyldne regel. Når brugeren lukker UbiMusic og starter den op igen, er alle relevante informationer gemt i en Settings fil, der indeholder et settings objekt. Dette er også gjort for at mindske mængden af information, en bruger skal huske, og for at minimere mængden af opgaver en bruger skal udføre for at bruge systemet. Derfor gemmes der blandt andet, hvilken højtaler, der hører til hvilket rum, så det ikke skal indstilles på ny. Filnavnet, der bruges til dataindsamling, bliver også gemt, så en bruger til hver en tid kan tilføje data til den eksisterende dataindsamling.

De forskellige funktionaliteter i UbiMusic er generelt set nemme at gøre om. Brugeren er eksempelvis informeret om, hvorvidt sporing er aktiveret dels i tekst, og ved at knapperne er grønne og aktive eller mørke og inaktive. Det samme gør sig gældende for dataindsamlinger. Det eneste, der ikke kan gøres om, er, når der laves en ny datafil. Derfor er der implementeret en dialogboks, der beder brugeren om at bekræfte, at der ønskes en ny datafil. Dette er gjort for at minime-

1 https://developer.spotify.com/documentation/design, besøgt 27. maj 2023

re risikoen for fejl og dermed følge Shneidermans femte gyldne regel.

Nederst på sporingssiden er implementeret en knap, der gør det muligt at fortryde et skift af højtaler. Knappen er kun aktiv, hvis det har været afspillet på en højtaler tidligere. Set fra et designperspektiv bidrager den både til, at brugeren er i kontrol over, hvor der skal afspilles musik, hvis der skiftes til et forkert rum, og den giver mulighed for at modvirke fejl. Knappen bidrager derfor både til at følge Shneidermans femte og syvende gyldne regel. I sektion 7.3 er det beskrevet, hvordan knappen også bidrager til muligheden for en konstant indsamling af data og træning af modeller, der vil bidrage yderligere til en sømløs musikoplevelse.

5.3 DATAINDSAMLING

Data indsamles på siden Konfigurering ved at skrive det ønskede rums navn og trykke på knappen til at starte en dataindsamling. Brugeren informeres herefter om, at en dataindsamling er startet og kan efterfølgende stoppe den. Hver gang systemet har modtaget en scanning, får brugeren en visuel information om, at data er gemt til den brugte datafil. At der på siden er angivet første step og andet step, er for at guide brugeren igennem konfigureringen. Dette er i overensstemmelse med Shneidermans fjerde gyldne regel, når konfigureringen er en længere proces.

Indsamlingen af datapunkter er implementeret ligesom, det er beskrevet i sekvensdiagrammet i Figur 1 ved at bruge en WifiManager og en WifiReceiver. MainActivity starter en Wi-Fi scanning ved brug af WifiManageren, og efter en fuldført scanning kaldes WifiReceiverens metode onReceive, der kalder en update metode i MainActivity. Årsagen til netop denne arkitektur er uddybet i under-sektion 5.1.3.

Om en ny Wi-Fi scanning skyldes, at systemet er taget i brug, eller en dataindsamling er i gang, håndterer MainActivity. Dette gør den ud fra to boolske variable. At det er implementeret ved brug af to variable, skyldes, at telefonen selv kan lave en Wi-Fi scanning. Hvis UbiMusic er åben på det tidspunkt, bliver WifiReceiveren notificeret og dermed også MainActivity.

En Wi-Fi scanning indeholder forskellige informationer, hvoraf kun nogle er relevante for dette projekt[2]. SSID er netværksnavnet, og BSSID er adressesen for det access point, hvorfra signalet kom. I dette projekt bruges BSSID, da denne er unik, selv hvis der er flere netværk med det samme navn, der har forskellig frekvens. Level er signalstyr-

2 https://developer.android.com/reference/android/net/wifi/ScanResult.html, Besøgt 12. marts 2023.

ken angivet i dBm, hvilket er det samme som RSS, og bruges som input til modellerne.

5.4 TRÆNING AF MODELLERNE

Til evalueringen af SVM modellerne og de neurale netværk er de implementeret i DataBehandling, der er skrevet i Python i Visual Studio Code. Dette er gjort på grund af skalerbarhed, da Python har bedre adgang til biblioteker, der understøtter udarbejdelsen af ML modeller. Hvis andre ML modeller ønskes evalueret senere, er det dermed bedre understøttet i Python end i Java.

Det neurale netværk, der bruges i UbiMusic, er trænet i DataBehandling, mens SVM modellen er trænet i ModelForUbiMusic, der er skrevet i Java. At træningen af modellerne ikke gøres i UbiMusic skyldes performance. Derved skal brugeren ikke vente på, at modellerne bliver trænet, og mængden af ressourcer brugt på telefonen mindskes. At SVM modellen er trænet i ModelForUbiMusic, giver mulighed for at bruge det samme bibliotek til træning og prædiktion. Derved er anvendelsen af modellen i UbiMusic mere pålidelig, da der bruges den samme beslutningsfunktion.

For dette projekt er de indsamlede data blevet overført til en computer. Derefter er modellerne blevet trænet og overført tilbage til telefonen og brugt. Da fokusset har været på mulighederne ved at bruge disse modeller, har denne fremgangsmåde været tilstrækkelig til evalueringen. Som en del af databehandlingen er datafilerne lavet om til matricer. Til dette er brugt tre værdier: BSSID, level og lokation. Ud fra lokationen for hvert indsamlet datapunkt er der lavet en vektor med de korrekte klassifikationer. Derudover er der lavet en matrice med træningsdata ud fra de indsamlede datapunkters level, hvor antallet af rækker svarer til antallet af datapunkter, og antallet af kolonner svarer til antallet af distinkte BSSIDer.

5.4.1 *SVM modeller*

Til at træne SVM modellerne er der brugt biblioteket sklearn i DataBehandling og biblioteket libsvm i ModelForUbiMusic. Disse gør det muligt at træne modeller ud fra matricer bestående af træningsdata og en vektor med korrekte klassifikationer. I træningen af modellerne er der brugt en balanceret vægtning af klasserne og en cache på 1.000 MB. Der er valgt en balanceret vægtning af klasserne for, at hver klasses vægtning er beregnet i forhold til antallet af den specifikke klasse i træningsdataene[3]. Derudover bruges en-til-en tilgang, som det er

3 https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html#sklearn.svm.SVC, besøgt 17. april 2023

beskrevet i sektion 4.2 samt en RBF kernel. Brugen af cross validation har givet mulighed for at vælge de C og γ parametre, der gav de højeste nøjagtigheder. De mulige værdier var.

$$C = \{0.1, 1, 10\}$$
$$\gamma = \{10^{-6}, 1.5 \cdot 10^{-5}, 10^{-5}, 10^{-4}, 0.001, 0.01\}$$

5.4.2 *Neurale netværk*

Som det er beskrevet i sektion 4.2, er de neurale netværk implementeret som et MLP netværk. Antallet af inputnoder er antallet af distinkte BSSIDer. Der er valgt fire skjulte noder og henholdsvis tre og fire outputnoder, der svarer til antallet af rum. For alle tre modeller bruges vægte imellem hver node, men kun for NN3-B bruges bias.

Til træningen af de neurale netværk er der først valgt vægte og bias tilfældigt ud fra en standard normalfordeling. Træningen er foregået over 5.000 iterationer, hvor der for hver 200 iterationer er bestemt tabet for modellen. Denne værdi er gemt sammen med de trænede vægte og bias. Efter alle iterationer er valgt de trænede værdier fra tidspunktet med det mindste tab. Tabet er udregnet med CEL funktionen, der kan ses i sektion 4.2, hvilket udtrykker den samlede afvigelse imellem modellens klassifikation og den korrekte klassifikation.

Som det er beskrevet i under-sektion 4.6.1, er den højeste sandsynlighed for hvert korrekt klassificeret datapunkt gemt under træningen af de neurale netværk. Ud fra disse er 5-fraktilen bestemt, som under evalueringen af netværkene er brugt til at afgøre, om brugeren er i *Intet rum*, hvis den højeste sandsynlighed er under 5-fraktilen.

Udregningerne for modeller med en sigmoid aktiveringsfunktion er beskrevet nedenfor, og en beskrivelse af variablene fremgår af Tabel 3. Udregningerne er bygget op af udregninger gående fremad fra inputlaget til outputlaget. Dette er udregningerne (3) til (6). Derefter udregnes de afledte værdier til loss funktionen i forhold til hver af variablene ved at gå tilbage fra outputlaget til inputlaget. Dette er udregningerne (7) til (11). I den første del gemmes de udregnede værdier, så de kan bruges i den anden del. Samtlige variable fratrækkes da den afledte værdi for loss funktionen i forhold til variablen multipliceret med en konstant indlæringshastighed. Dette er udregningerne (12) til (15). Indlæringshastigheden er valgt til 10^{-3} for at have en indvirkning på opdateringerne uden en stor risiko for at overskride værdierne, så de bliver være end før.

$$zh = wh \cdot X + bh \tag{3}$$

VARIABEL	BESKRIVELSE	DIMENSION
X	Træningsdata	$n \times m$
Y	Korrekte labels	$n \times 1$
wh	Vægte imellem inputlag og skjult lag	$m \times k$
bh	Bias imellem inputlag og skjult lag	$k \times 1$
wo	Vægte imellem skjult lag og outputlag	$k \times j$
bo	Bias imellem skjult lag og outputlag	$j \times 1$
lr	Indlæringshastigheden	
$cost$	Loss funktionen	

Tabel 3: Oversigt over variablene i træningen af neurale netværk. n er antal datapunkter, m er antal noder i inputlaget, k er antal noder i det skjulte lag, og j er antal noder i outputlaget.

$$ah = \frac{1}{1 + e^{-zh}} \tag{4}$$

$$zo = wo \cdot ah + bo \tag{5}$$

$$ao = \frac{e^{zo}}{\Sigma_{j=0}^{j-1} zo_j} \tag{6}$$

$$\frac{\partial cost}{\partial wo} = \frac{\partial cost}{\partial zo} \cdot \frac{\partial zo}{\partial wo} = (ao - Y) \cdot ah \tag{7}$$

$$\frac{\partial cost}{\partial bo} = \frac{\partial cost}{\partial zo} = ao - Y \tag{8}$$

$$\frac{\partial cost}{\partial ah} = \frac{\partial cost}{\partial zo} \cdot \frac{\partial zo}{\partial ah} = (ao - Y) \cdot wo \tag{9}$$

$$\begin{aligned} \frac{\partial cost}{\partial wh} &= \frac{\partial cost}{\partial ah} \cdot \frac{dah}{dzh} \cdot \frac{\partial zh}{\partial wh} \\ &= (ao - Y) \cdot wo \cdot \frac{1}{1 + e^{-zh}} \cdot \left(1 - \frac{1}{1 + e^{-zh}}\right) \cdot X \end{aligned} \tag{10}$$

$$\begin{aligned} \frac{\partial cost}{\partial bh} &= \frac{\partial cost}{\partial ah} \cdot \frac{dah}{dzh} \\ &= (ao - Y) \cdot wo \cdot \frac{1}{1 + e^{-zh}} \cdot \left(1 - \frac{1}{1 + e^{-zh}}\right) \end{aligned} \tag{11}$$

$$wh = wh - lr \cdot \frac{\partial cost}{\partial wh} \tag{12}$$

$$bh = bh - lr \cdot \frac{\partial cost}{\partial bh} \tag{13}$$

$$wo = wo - lr \cdot \frac{\partial cost}{\partial wo} \tag{14}$$

$$bo = bo - lr \cdot \frac{\partial cost}{\partial bo} \tag{15}$$

5.5 BRUGBARHEDEN AF MODELLERNE

Efter træningen af modellerne er de trænede værdier fra det neurale netværk indsat i UbiMusic, og SVM modellen er hentet fra en fil. Når systemet er taget i brug, bestemmes brugerens lokation ved, at det nye datapunkt gives til metoden determineLocation i ILocation interfacet, der ved at bruge implementeringer af IDataManagement afgør brugerens position.

For det neurale netværk i UbiMusic bestemmes brugerens lokation ved at bruge de trænede vægte samt en identitets aktiveringsfunktion. Dette giver en sandsynlighedsfordeling, der bruges til at afgøre, om brugeren er udenfor det trænede område. I evalueringen af de neurale netværk i DataBehandling, er brugerens position bestemt ud fra de trænede vægte, eventuelle bias samt modellernes aktiveringsfunktioner. SVM modellen i UbiMusic bestemmer brugerens lokation ved at opsætte et svm_node array og bruge svm_predict metoden fra libsvm. I DataBehandling er der brugt predict metoden fra sklearn.

Systemet tages i brug af brugeren ved at trykke på knappen AKTIVER SPORING på siden Sporing. Når en ny lokation er prædikteret, kontrolleres det, om denne er identisk med den eksisterende lokation. Hvis det er en ny lokation, gemmes denne. Når der er to på hinanden følgende prædiktioner for en lokation, der ikke er den eksisterende, skiftes der højtaler. For hver fuldført scanning startes en ny scanning, indtil brugeren har trykket på knappen DEAKTIVER SPORING, eller UbiMusic lukkes.

5.6 SPOTIFY

At der bruges Spotify i UbiMusic betyder, at brugeren skal logge ind med en Spotify bruger første gang, UbiMusic bruges. Det er implementeret sådan, at hvis Spotifys egen app er installeret på telefonen, henter UbiMusic automatisk informationerne om den bruger, der er logget ind. Hvis Spotifys egen app ikke er installeret, kommer en hjemmeside frem på skærmen, hvor brugeren bliver bedt om at logge ind. Dette gør det muligt at integrere musikken fra Spotifys egen app med UbiMusic. Det betyder også, at hvis brugeren har aktiveret sporing, så musikken følger vedkommende, men skifter sang i Spotifys egen app, over en computer eller ved brug af Google Assistent, fortsætter musikken med at følge brugeren. At andre interagerer med Spotify, har ingen indflydelse på, at UbiMusic stadigvæk sørger for at skifte til højtaleren i det rum, hvor brugeren er.

Når brugeren er logget ind, bliver brugerens adgangstoken hentet og gemt til brug ved senere kald til Spotifys API. Ved autorisation er

scoped sat til streaming, user-read-email, user-read-currently-playing og user-read-playback-state. Dette er nødvendigt for at kunne afspille musik samt se og skifte højtaler.

Valget af højtaler til de forskellige rum er implementeret som en del af siden Konfigurering. I takt med at brugeren indsamler data i forskellige rum, bliver rummene tilføjet til et string array. De forskellige rum er synlige for brugeren i en drop down menu ligesom en anden drop down menu med tilgængelige højtalere. Når brugeren vælger en højtaler og et lokale og trykker gem, gemmes lokalenavnet og højtalerens navn i en hashtabel. De tilgængelige højtalere er bestemt ved brug af Spotifys API, hvilket giver et navn, der vises til brugeren og et unikt id, der bruges til at skifte højtaler.

Når der skiftes højtaler laves et json objekt bestående af den ønskede højtalers id, og at play skal være true. Dette betyder, at når der skiftes lokale, afspiller den nye højtaler altid musik. Dette er gjort for, at musikken altid begynder at spille, når en bruger aktiverer sporing. Det har dog den konsekvens, at hvis en bruger har pauset musikken og går ind i et andet rum, hvori der er en højtaler, begynder musikken at afspille der.

5.7 WI-FI SCANNINGER

I under-sektion 3.1.2 er det beskrevet, hvordan Android version 8.0 og opefter har en begrænsning på fire Wi-Fi scanninger indenfor to minutter. For at undgå denne begrænsning i indsamlingen af data og evalueringen af dette projekt, er telefonen, der blev brugt, sat til udviklertilstand uden en begrænsning på mængden af Wi-Fi scanninger. Dette giver mulighed for at lave Wi-Fi scanninger cirka hvert sjette sekund.

6 EVALUERING

Dette projekts forskningsspørgsmål er:

> I hvilket omfang kan der laves en sømløs musikoplevelse i hjem med højtalere i flere rum ved brug af et IPS understøttet af SVM og neurale netværk?

Som det er beskrevet i under-sektion 3.1.1, evalueres der også på muligheden for at sætte musikken på pause, når en bruger går ind i et rum uden en højtaler. I forlængelse af dette er der evalueret på muligheden for at registrere, når en bruger er udenfor rækkevidde af det trænede område og dermed formodes at have forladt sit hjem. Disse to særskilte evalueringer indgår i den samlede evaluering af forskningsspørgsmålet. I under-sektion 4.6.2 er det beskrevet, hvordan konfidensintervallerne, der bruges, er designet.

6.1 ACCESS POINTS

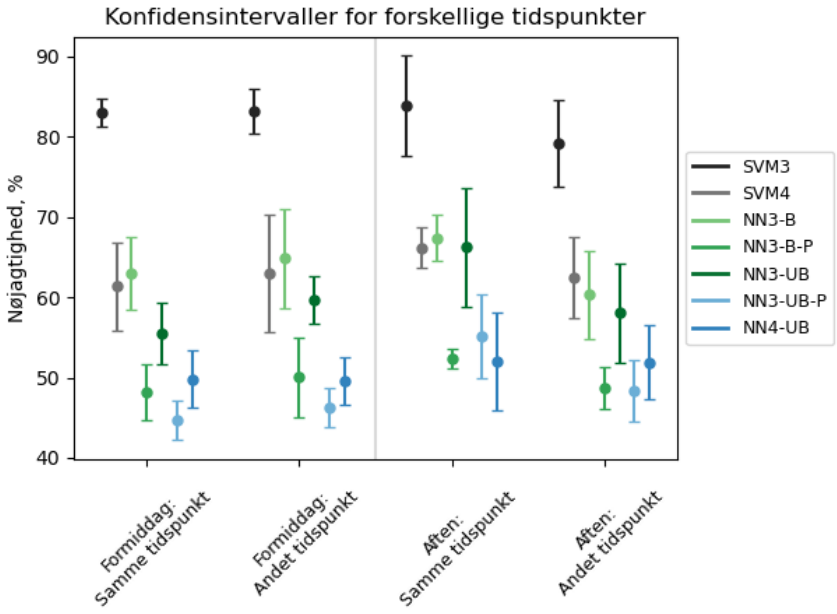
Til træningen af modellerne er der brugt 66 distinkte BSSID. Nogle access points udsender to netværk, da de udsender både 2.4 GHz og 5 GHz, og hvert af dem har et unikt BSSID. Det er væsentligt, at der udelukkende indsamles information om signalstyrken for Wi-Fi signalerne, og der derfor ikke behøves en adgangskode til netværkene.

Det kan blive en udfordring, at der kun bruges Wi-Fi signaler, hvis UbiMusic bruges i områder, hvor der ikke er mange tilgængelige netværk eller måske kun et. Dette kan eksempelvis være tilfældet for gårde, hvor der ingen naboer er tæt på. I et sådant tilfælde vil det være relevant at se på en videreudvikling, hvor der inddrages hardware.

6.2 FORSKELLIGE TIDSPUNKTER

Af sektion 4.5 fremgår det, at der er evalueret på modellernes nøjagtigheder, når de bruges på forskellige tidspunkter for at afgøre, om Wi-Fi signalers styrke varierer afhængigt af antallet af personer og enheder, der bruger netværket. I starten af projektet sås en indikation af en variation, hvorfor det var forventet, at der ville være forskel.

Figur 6 angiver 95% konfidensintervaller for modellernes nøjagtigheder, når de bruges på forskellige tidspunkter. Hvert punkt på førsteaksen angiver en model, der er trænet ud fra et datasæt indsamlet i et bestemt tidsrum og testet på de andre datasæt, der enten er ind-



Figur 6: Konfidensintervaller for modeller, der er trænet på data indsamlet enten om formiddagen eller aftenen. Modellerne er enten testet på data indsamlet på samme tidspunkt eller et andet tidspunkt. Konfidensintervallerne er lavet på et 5% signifikansniveau og kan ses i Tabel 5.

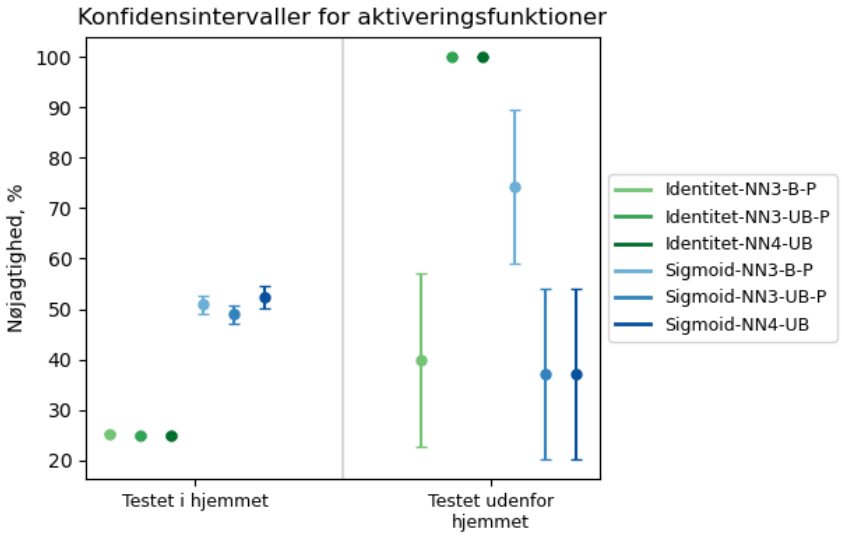
samlet i samme tidsrum eller det andet tidsrum. Samtlige modeller er trænet på et helt datasæt med en dataindsamling på 45 minutter i hvert rum og cross validation, som det fremgår af under-sektion 4.6.1.

For hver model fremgår det af figuren, at der ikke er signifikant forskel imellem nøjagtighederne afhængigt af, hvilket tidsrum de er testet på. Derfor kan det på det foreliggende grundlag ikke afvises, at de gennemsnitlige nøjagtigheder er de samme uafhængigt af, hvornår modellerne bruges.

6.3 UDENFOR TRÆNET OMRÅDE

I under-sektion 3.1.1 er det beskrevet, hvordan neurale netværk med en identitets aktiveringsfunktion, der er trænet uden bias, har mulighed for at outputte en lige stor sandsynlighed for samtlige klasser, hvis dataene er udenfor rækkevidde af det trænede område.

På Figur 7 ses modellerne NN3-B, NN3-UB og NN4-UB. For hver af dem er der to modeller med henholdsvis en identitets og en sigmoid aktiveringsfunktion. For de neurale netværk, der er trænet på tre rum, er der brugt versionerne NN3-B-P og NN3-UB-P, der angiver det fjerde rum ud fra 5-fraktilen, som det er beskrevet i under-sektion 4.6.1.

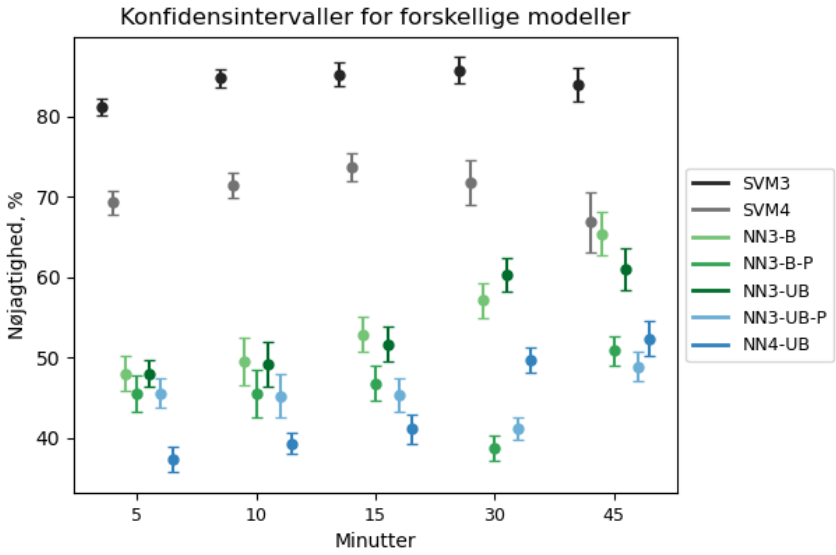


Figur 7: Tre forskellige neurale netværk hvor hver af dem er trænet med henholdsvis en identitets aktiveringsfunktion og en sigmoid aktiveringsfunktion imellem inputlaget og det skjulte lag. De angivne konfidensintervaller er lavet på et 5% signifikansniveau.

For at bestemme modellernes nøjagtigheder udenfor hjemmet er der testet på dataene, der er indsamlet på Aarhus Universitet.

Til venstre er konfidensintervallerne for modellerne, når de testes i hjemmet og dermed indenfor det trænede område. Her fremgår det, at modellerne, der bruger en sigmoid aktiveringsfunktion med 95% sikkerhed har en signifikant højere nøjagtighed end modellerne, der bruger en identitets aktiveringsfunktion. Til gengæld er der forskel, når modellerne testes på data, der er indsamlet på Aarhus Universitet og dermed udenfor det trænede område. Her har modellerne Identitet-NN3-UB-P og Identitet-NN4-UB begge et konfidensinterval på $[1,0; 1,0]$, fordi samtlige datapunkter er korrekt klassificeret. Til sammenligning har de andre modeller brede konfidensintervaller, og de har med 95% sikkerhed signifikant lavere nøjagtigheder.

I under-sektion 4.6.1 er det beskrevet, hvordan Identitet-NN3-UB-P og Identitet-NN4-UB bruger en særlig tærskel og derved opnår en nøjagtighed på 100%, når de bruges udenfor det trænede område. Derfor er konfidensintervallerne, der er opnået i dette projekt for disse modeller, ikke kun et udtryk for en høj nøjagtighed i dette tilfælde. Den samme nøjagtighed vil opnås i andre hjem, hvis de samme modeller bruges udenfor det trænede område.



Figur 8: Konfidensintervaller lavet på et 5% signifikansniveau for fem forskellige ML modeller. Ud af førsteaksen er angivet, i hvor lang tid der er indsamlet data i hvert rum til at træne modellerne på. De nøjagtige intervaller kan ses i Tabel 6.

På det foreliggende grundlag kan det dermed konkluderes, at det er en fordel at bruge en sigmoid aktiveringsfunktion sammenlignet med en identitets aktiveringsfunktion, når modellerne bruges indenfor det trænede område. Tilgengæld opnås en nøjagtighed på 100%, når modeller uden bias og med en identitets aktiveringsfunktion bruges udenfor det trænede område. Dette kan bruges til at afgøre, om en person har forladt sit hjem, og at en eventuel igangværende afspilning dermed skal sættes på pause.

For at afgøre om en bruger er udenfor sit hjem, kan der alternativt i den offline fase vælges et foretrukket access point eller en GPS position og en afstand. I den online fase kan dette bruges til at afgøre, om en bruger er for langt fra sit hjem eller ikke modtager signal fra sit foretrukne access point. Hvis det neurale netværk bruges, kan modellen trænes ud fra de indsamlede datapunkter, og brugeren behøver derfor ikke foretage sig yderligere.

6.4 FORSKELLIGE MODELLER

I Figur 8 er der konfidensintervaller for samtlige fem ML modeller samt ekstra test af NN3-B og NN3-UB, som det er beskrevet i sektion 4.6. Mængden af data, hver model er trænet på, er angivet ud af førsteaksen som antallet af minutter, der er indsamlet data i hvert

rum. Samtlige modeller er testet på resten af dataene fra samme datasæt samt samtlige andre datasæt, hvorfor antallet af frihedsgrader er omkring 50.

Ud fra Figur 8 er det muligt at fremhæve flere pointer, hvoraf den mest væsentlige er, at begge SVM modeller med undtagelse af SVM4, der er trænet på 45 minutters data, med 95% sikkerhed har en signifikant højere nøjagtighed end samtlige neurale netværk. At SVM4 med 95% sikkerhed har en signifikant lavere nøjagtighed end SVM3, kan skyldes rummenes placering i forhold til hinanden. Derudover er der ikke en datamængde, der er signifikant bedre end samtlige andre datamængder for hverken SVM3 eller SVM4. På det foreliggende grundlag kan det med 95% sikkerhed heller ikke afvises, at den gennemsnitlige nøjagtighed er ens ved indsamling af data i fem og 45 minutter i hvert rum.

De neurale netværk kan opdeles i to kategorier afhængigt af, om de testes på tre eller fire rum. NN3-B og NN3-UB er testet på tre rum, mens NN3-B-P, NN3-UB-P og NN4-UB er testet på fire rum. For 15, 30 og 45 minutter har modellerne, der er testet på fire rum, med 95% sikkerhed signifikant lavere nøjagtigheder end modellerne, der er testet på tre rum. Eftersom dette også ses for SVM modellerne, kan en forklaring som nævnt være rummenes placering i forhold til hinanden.

NN3-B modeller, der er trænet på 45 minutters dataindsamling i hvert rum, har med 95% sikkerhed en signifikant højere nøjagtighed end samtlige andre NN3-B modeller. Tilsvarende for NN3-UB og NN4-UB modeller, der er trænet på 30 og 45 minutters dataindsamling, har de med 95% sikkerhed signifikant højere nøjagtigheder end henholdsvis andre NN3-UB og NN4-UB modeller. Dette kan skyldes, at neurale netværk har brug for meget data for at være præcise, hvilket står i kontrast til SVM modellerne.

På det foreliggende grundlag kan det med 95% sikkerhed konkluderes, at SVM modellerne har en signifikant højere nøjagtighed end de neurale netværk. Ydermere kan det med 95% sikkerhed ikke afvises, at brugere får den samme nøjagtighed ved en dataindsamling på fem minutter sammenlignet med en på 45 minutter i hvert rum, når der bruges en af SVM modellerne. Derudover er der evidens for, at de neurale netværk skal bruge mere data for at opnå en højere nøjagtighed.

6.5 RUMSKIFTE

I under-sektion 3.1.2 er det beskrevet, hvordan UbiMusic er designet til først at skifte rum efter to på hinanden følgende prædiktioner for det samme rum. I Tabel 6 i appendiks fremgår det, at for SVM3 modeller med fem minutters dataindsamling er konfidensintervallet for den gennemsnitlige nøjagtighed [80,1%; 82,3%]. Dette betyder, at sandsynligheden for at have to forkerte prædiktioner lige efter hinanden forventes at være 3,53%. Som det fremgår af under-sektion 3.1.2, betyder dette, at det tager længere tid, førend et nyt rum bliver valgt, men det mindsker også risikoen for at vælge væk fra et korrekt rum markant. Alt afhængigt af hvilken service det ønskes at understøtte, kan det overvejes, om antallet af korrekte prædiktioner skal justeres.

Prædiktionerne kan anses som værende Bernoulli fordelt, da de enten er forkerte eller korrekte. Derved kan det forventede antal Wi-Fi scanninger inden k på hinanden følgende korrekte prædiktioner bestemmes ved den geometriske fordeling [1]. Formlen hertil er

$$E(X_k) = \Sigma_{i=1}^{k} \left(\frac{1}{p^i} \right),$$

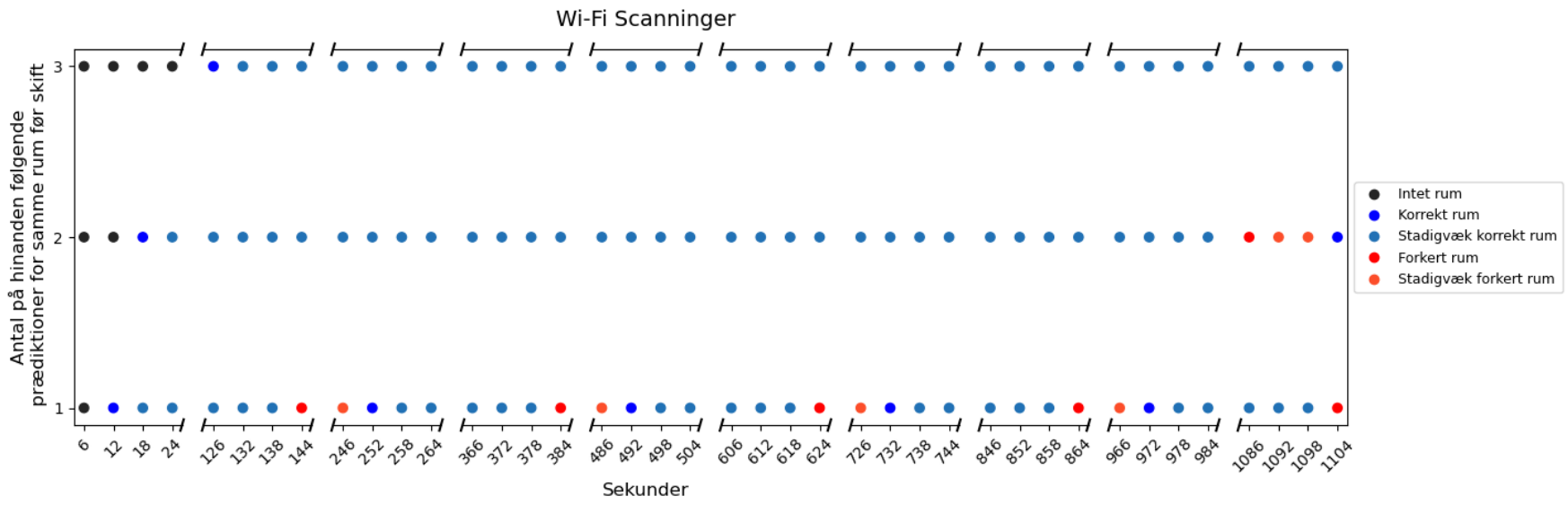
hvor p er sandsynligheden for et korrekt udfald. I Tabel 4 i appendiks fremgår de forventede antal afhængigt af, hvilken tærskel, der vælges. Når der kræves to på hinanden følgende prædiktioner for det samme rum, giver dette et forventet antal forsøg på 2,75. Efter den tredje Wi-Fi scanning forventes det dermed, at der er skiftet til det korrekte rum.

I sektion 5.7 er det beskrevet, hvilke begrænsninger, der er for Wi-Fi scanninger i Android. Eftersom det forventede antal scanninger, inden det korrekte rum er valgt, er tre, kan dette gøres på 18 sekunder uafhængigt af, om begrænsningen af scanninger er slået til eller fra. På det foreliggende grundlag forventes det derfor, at UbiMusic har skiftet til det rigtige rum efter 18 sekunder.

Konsekvenserne, ved at vælge forskellige tærskler for, hvornår der skiftes rum, fremgår af Figur 9. Hvis tærsklen vælges til tre, forventes der først at være skiftet til det korrekte rum efter 126 sekunder. Til gengæld forventes der først at blive skiftet væk fra et korrekt rum efter 185 Wi-Fi scanninger, hvilket svarer til lige over 92 minutter. Til sammenligning forventes det, at der skiftes væk fra et korrekt rum efter 34 Wi-Fi scanninger svarende til lige over 16 minutter, når tærsklen er valgt til to. Den laveste latency opnås ved at sætte tærsklen til en, men det vil medføre, at der oftere vil blive skiftet væk fra et korrekt rum.

At opnå den mest sømløse musikoplevelse afhænger derfor af, hvad

der vægtest højest af lav latency og høj præcision. Hvis der ønskes det hurtigst mulige skift af højtaler, bør tærsklen sættes til en, men ønskes der færrest mulige skift væk fra en korrekt højtaler, bør tærsklen sættes op. Dette er en afvejning, der bør gøres for hver service, og som muligvis bør gives som en indstilling til brugeren.



Figur 9: Konsekvensen af forskellige tærskler for hvor mange på hinanden følgende prædiktioner for det samme rum, der skal være, førend der skiftes rum. De er angivet ud fra de forventede værdier udregnet ud fra den geometriske fordeling, da prædiktionerne følger en Bernoulli fordeling. De forventede antal kan ses i Tabel 4. Sættes tærsklen til tre, forventes der at blive skiftet til et forkert rum efter 185 Wi-Fi scanninger svarende til lige over 92 minutter.

6.6 NØJAGTIGHED

Som det fremgår af sektion 3.1, er evalueringen af dette projekts nøjagtigheder sammenholdt med tidligere forskning, der er beskrevet i sektion 2.4. Tidligere forskning havde SVM modeller med nøjagtigheder på imellem 80,5% og 95,5% og neurale netværk med nøjagtigheder på imellem 85,6% og 96,0%. En gennemsnitlig nøjagtighed på 81,2% for SVM3 modeller, der er trænet på fem minutters dataindsamling i hvert rum, er derfor indeholdt i intervallet men i den lave ende. Til gengæld har NN3-B, der er trænet på 45 minutters dataindsamling i hvert rum, den højeste gennemsnitlige nøjagtighed for de neurale netværk på 65,4%, hvilket er markant lavere end tidligere forskning.

I den tidligere forskning bruges flere etager med undtagelse af [6], men i ingen af dem kendes rummenes nøjagtige placering i forhold til hinanden. I dette projekt er bevidst valgt et åbent køkkenalrum samt et rum, der ligger lige over to af de andre rum for at udfordre modellerne mest muligt. Dette kan derfor være en af årsagerne til den lavere nøjagtighed for både SVM modellerne og de neurale netværk. At der ikke er signifikant forskel i nøjagtighed imellem SVM3 trænet på fem og 45 minutters dataindsamling understøtter dette. Eftersom nøjagtigheden ikke stiger i takt med, at der bruges mere træningsdata, kan den lavere nøjagtighed skyldes andre faktorer, såsom rummenes placering eller placeringen af de brugte access points i forhold til de valgte rum.

I under-sektion 5.4.1 er det beskrevet, hvordan cross validation er brugt til at bestemme de bedste SVM modeller. Under træningen af modellerne sås en klar tendens til overfitting for modeller, der var trænet på 45 minutters dataindsamling, hvis en C-værdi på 100 blev inkluderet. Denne blev derfor fjernet som mulighed. At nøjagtigheden for SVM3 modeller falder fra 30 til 45 minutters dataindsamling, kan derfor skyldes en generel tendens til overfitting, når der indsamles en stor mængde data uafhængigt af den valgte C-værdi.

Ud fra de neurale netværks nøjagtigheder er der, som det fremgår af sektion 6.4, evidens for, at nøjagtigheden stiger ved indsamling af mere træningsdata. Dette gør sig også gældende i tidligere forskning. Dataindsamlingen i [4] svarer til 3,2 minutters dataindsamling i hvert rum, dataindsamlingen i [6] svarer til 8,6 minutter, og i [29] svarer dataindsamlingen til 21,5 minutter. Tilsvarende stiger nøjagtigheden, hvor [4] har den laveste nøjagtighed, og [29] har den højeste nøjagtighed. Dette understøtter derfor, hvad der også er set i dette projekt, at mere træningsdata giver en højere nøjagtighed for neurale netværk.

På trods af en større dataindsamling har de neurale netværk i dette projekt opnået væsentlig lavere nøjagtigheder. Det kunne derfor være relevant at se på opbygningen af de neurale netværk, da både [4] og [29] bruger stacked autoencoders. At dette ikke var valgt oprindeligt til dette projekt, skyldes dels rummenes opbygning, og at tidligere forskning havde angivet, at der oftest blev brugt MLP netværk til fingerprinting [10, 20]. I det åbne køkkenalrum var det væsentligt, at brugeren kunne angive, hvor skillelinjen imellem køkken og stue var, og det ikke blev klassificeret af en unsupervised autoencoder. Om der har været tilsvarende rum i [4] og [29], og de neurale netværk dermed kunne håndtere dette, fremgår ikke af artiklerne.

En anden væsentlig forskel imellem dette projekt og tidligere forskning i forhold til de neurale netværk er antallet af noder i det skjulte lag. Både [4] og [29] tester hvilket antal, der giver den højeste nøjagtighed, hvilket ikke har været et fokus i dette projekt. Det kan derfor også være relevant at undersøge den bedst mulige kombination af skjulte noder, hvis nøjagtigheden for de neurale netværk ønskes hævet.

Generelt er det bemærkelsesværdigt, at neurale netværk i tidligere forskning har en højere nøjagtighed end SVM modellerne, når SVM modellerne i dette projekt har signifikant højere nøjagtigheder end de neurale netværk. Det er dog væsentligt at bemærke, at SVM3 med kun fem minutters dataindsamling i hvert rum opnår en nøjagtighed lignende tidligere forskning. Ønskes der en højere nøjagtighed, opnås der en gennemsnitlig nøjagtighed på 85,7% ved 30 minutters dataindsamling, men det giver samtidig en markant længere dataindsamling.

7 VIDERE ARBEJDE

Et nøjagtigt IPS har et stort potentiale og med en gennemsnitlig nøjagtighed på 81,2% med kun fem minutters dataindsamling i hvert rum, er det nøjagtigt nok til at kunne bruges i dagligdagen. Alligevel er der mange muligheder for at arbejde videre med dette projekt.

7.1 ANDRE HJEM

Dette projekt har udelukkende testet forskellige ML modellers nøjagtigheder i et hjem, hvor der var signifikant forskel imellem tre og fire rum for SVM modeller. Det kunne derfor være relevant at undersøge, hvor nøjagtig de forskellige modeller er i andre hjem.

7.2 ANTAL WI-FI SCANNINGER

Eftersom Android har begrænsninger i forhold til antal Wi-Fi scanninger, vil det være relevant at arbejde videre med optimeringen af Wi-Fi scanninger eksempelvis ved at bruge telefonens accelerometer til at udløse Wi-Fi scanninger. Det er væsentlig at undersøge, hvornår der skal stoppes med at foretages Wi-Fi scanninger, så det sikres, at der er blevet skiftet til det korrekte rum.

7.3 UBIMUSIC

Som det er beskrevet i sektion 3.2, kan der arbejdes videre med mængden af tilgængelige højtalere, der bruges i UbiMusic. Derudover kan designet, der er beskrevet i sektion 4.3 og sektion 4.4 implementeres, så UbiMusic kan håndtere flere services og flere brugere.

For virkelig at kunne give brugeren en sømløs oplevelse er det nødvendigt, at modellerne hele tiden er trænet på det mest opdaterede data. Derfor kunne adaptiv træning være relevant at arbejde videre med. Med knappen, der er beskrevet i sektion 5.2, kan systemet få korrekt klassificeret træningsdata, der kan bruges til træning af nye modeller. Ud fra andelen af kendte netværk i nye Wi-Fi scanninger kan det afgøres, hvornår der skal trænes nye modeller. Dette kan eksempelvis gøres, når den gennemsnitlige andel er kommet under en fastsat tærskel ud fra et bestemt signifikansniveau.

8 KONKLUSION

Dette projekt har fokuseret på et Wi-Fi baseret Indoor Positioning System (IPS) ved at bruge fingerprinting, hvor der trænes og efterfølgende bruges Machine Learning (ML) modeller i form af Support Vector Machine (SVM) modeller og neurale netværk.

8.1 MODELLER

SVM modellerne bruger en Radial Basis Function (RBF) kernel, en-til-en tilgang samt balanceret vægtning af klasserne. De neurale netværk er et Multi-Layer Perceptron (MLP) netværk med et skjult lag bestående af fire noder. Loss funktionen er Cross Entropy Loss (CEL), og imellem det skjulte lag og outputlaget er der brugt en softmax aktiveringsfunktion. Imellem inputlaget og det skjulte lag er der testet på brugen af henholdsvis en identitets og en sigmoid aktiveringsfunktion.

For at teste ML modellerne er der indsamlet lige over 13.000 datapunkter i et rækkehus fordelt på fire rum og to etager. Dataindsamlingen var fordelt på fire formiddage og tre aftener. Derforuden er der indsamlet knap 1.000 datapunkter på Aarhus Universitet. Til indsamlingen er brugt en telefon, og hvert datapunkt er en Wi-Fi scanning.

8.2 NØJAGTIGHED

Dette projekts forskningsspørgsmål er:

> I hvilket omfang kan der laves en sømløs musikoplevelse i hjem med højtalere i flere rum ved brug af et IPS understøttet af SVM og neurale netværk?

Hvorvidt SVM og neurale netværk kan bruges til at lave en sømløs musikoplevelse i hjem med højtalere i flere rum, afhænger i høj grad af modellernes nøjagtigheder. Ud fra dette projekt kan det konkluderes, at en SVM model, der er trænet på tre rum med fem minutters dataindsamling i hvert rum, på det foreliggende grundlag med 95% sikkerhed har en nøjagtighed på imellem 80,1% og 82,3%. Dermed matcher det den tidligere forskning, der har nøjagtigheder på imellem 80,5% og 95,5%.

Til sammenligning opnår et neuralt netværk trænet på tre rum med 45 minutters dataindsamling i hvert rum med 95% sikkerhed en nøj-

agtighed på imellem 62,8% og 68,1%. De neurale netværk har generelt signifikant lavere nøjagtigheder end SVM modellerne, og der er evidens for, at de kræver en større mængde data for at hæve nøjagtigheden. Dette er uddybet i sektion 6.4.

Tidligere forskning indikerer også, at en større mængde træningsdata hæver nøjagtigheden for neurale netværk. De neurale netværks nøjagtigheder i dette projekt ligger dog markant under nøjagtighederne, der er opnået i tidligere forskning. Det er derfor relevant at overveje, om der bør arbejdes videre med en anden opbygning af dem. Det kunne eksempelvis være ved at bruge stacked autoencoders eller undersøge den bedste kombination af noder i det skjulte lag.

Ved at bruge hardware som eksempelvis bevægelsessensorer, vil det være muligt at angive en persons lokation mere nøjagtigt, men det vil kræve en mere omfattende konfigurering. Om der udelukkende ønskes at bruges Wi-Fi signaler fra eksisterende access points eller opsættes hardware, er derfor en afvejning imellem omfanget af konfigureringen i forhold til præcision. Eftersom dette projekt har fokuseret på brugbarheden i et hjem, har fokusset været på en minimal konfigurering. At SVM modellerne er nøjagtige med kun fem minutters dataindsamling i hvert rum, bekræfter muligheden for et nøjagtigt IPS med en begrænset konfigurering.

8.3 MÆNGDEN AF DATA

På det foreliggende grundlag kan det på et 5% signifikansniveau ikke afvises, at SVM modeller, der er trænet på 45 minutters dataindsamling i hvert rum, har samme nøjagtighed, som modeller, der er trænet på fem minutters dataindsamling i hvert rum. Dette indikerer, at den lavere nøjagtighed sammenlignet med noget af den tidligere forskning muligvis skal findes i andre faktorer end modellernes opbygning.

En årsag kunne være rummenes placering i forhold til hinanden eller det åbne køkkenalrum. I tidligere forskning fremgår rummenes placering i forhold til hinanden ikke, hvorfor det ikke vides, om der er vægge imellem alle rum eller åbne rum som køkkenalrummet i dette projekt. I dette projekt er der bevidst valgt rum, der må forventes at udfordre modellerne mest muligt. Dette er gjort ved både at have et åbent køkkenalrum, der udgør to rum, samt ved at have to etager, hvor et af rummene er lige ovenpå to andre rum.

8.4 RUM UDEN HØJTALERE

I under-sektion 3.1.1 er det beskrevet, hvordan brugere af et lignende system ønskede, at musikken skulle stoppe, når de forlod et rum med en højtaler og gik ind i et rum uden en højtaler. Selvom nøjagtighederne for både SVM modeller og neurale netværk er signifikant lavere for fire rum, er en SVM model trænet på fire rum det bedste valg til at understøtte denne funktionalitet. Med 95% sikkerhed vil modellen give en gennemsnitlig nøjagtighed på imellem 67,8% og 70,8% ved fem minutters dataindsamling i hvert rum.

8.5 FORSKELLIGE TIDSPUNKTER

Det er blevet undersøgt, om der er signifikant forskel i nøjagtighed afhængigt af, hvornår en model bruges. Om den bruges på samme tidspunkt, som dataene, den er trænet på, var indsamlet, eller om den bruges på et andet tidspunkt. På det foreliggende grundlag kan det på et 5% signifikansniveau ikke afvises, at modellernes nøjagtighed er den samme uafhængigt af tidspunktet, de bruges. En uddybning af dette fremgår af sektion 6.2.

8.6 UDENFOR HJEMMET

De neurale netværk, der er brugt i dette projekt, har haft en sigmoid aktiveringsfunktion imellem inputlaget og det skjulte lag. Der er også blevet testet på at bruge en identitets aktiveringsfunktion, men dette giver en signifikant lavere nøjagtighed. Til gengæld kan et neuralt netværk med en identitets aktiveringsfunktion bruges til at afgøre, hvornår en bruger har forladt det trænede område og dermed kan formodes at have forladt sit hjem. Dette kan gøres med en nøjagtighed på 100%, når det neurale netværk implementeres uden brug af bias. Dette er uddybet i sektion 6.3.

8.7 UBIMUSIC

Til at demonstrere brugbarheden af dette projekts IPS er Android appen UbiMusic blevet udviklet. I konfigureringen af UbiMusic vælger brugeren, hvilken højtaler, der skal afspille i hvilket rum. Derved følger musikken brugeren, når denne har aktiveret sporing og går ind i et andet rum med en højtaler.

UbiMusic bruger Spotifys API, hvormed det er muligt at styre musikken, der afspilles over Spotify samt skifte højtaler. Den er implementeret, så brugeren kan tilgå Spotify via andre enheder og eksempelvis skifte musik uden, at sporingen i UbiMusic påvirkes. UbiMusic opdaterer også automatisk informationerne om den afspillende sang, så

de altid er opdateret.

Til forskel fra [23] er der i UbiMusic ikke behov for opsætning af særlig hardware. Derved minder den om AirPlayer og Audio as a Service (A3S) [17, 28]. Dette gør UbiMusic nem at bruge for en bruger, der ikke har interesse for eller kendskab til opsætning af særlig hardware. Brugeren bliver guidet igennem konfigureringen, hvorefter sporingsfunktionaliteten kan benyttes.

UbiMusic er implementeret, så den bruger to modeller. Den ene er en SVM model, der bruges til at afgøre, hvilket rum brugeren er i, da denne med 95% sikkerhed har en signifikant højere nøjagtighed end de neurale netværk. Den anden model er et neuralt netværk uden bias og med en identitets aktiveringsfunktion, der bruges til at afgøre, om brugeren har forladt det trænede område. Denne model bruges udelukkende til at sætte musikken på pause, når en bruger forlader sit trænede område.

8.8 PRÆCISION ELLER LAV LATENCY

Som konklusion til projektets forskningsspørgsmål kan det konkluderes, at det er muligt at lave en sømløs musikoplevelse med højtalere i flere rum ved at bruge en SVM model. I hvilken grad oplevelsen er sømløs, afhænger blandt andet af, om der foretrækkes præcision og dermed et minimalt antal skift væk fra et korrekt rum, eller om der foretrækkes lav latency og dermed et hurtigt skift af højtaler, når der skiftes rum.

UbiMusic er implementeret, så den først skifter rum efter to på hinanden følgende prædiktioner for det samme rum. Dermed forventes den på det foreliggende grundlag at have skiftet til det rigtige rum efter tre Wi-Fi scanninger, hvis der er foretaget fem minutters dataindsamling i hvert rum. Dette forventes at være gjort indenfor 18 sekunder. Tilsvarende forventes den at skifte til et forkert rum 3,53% af Wi-Fi scanningerne, hvilket svarer til et forventet skift væk fra et korrekt rum efter lidt over 16 minutter.

Disse andele kan mindskes ved at hæve det nødvendige antal på hinanden følgende prædiktioner. Til gengæld vil det tage længere tid, førend systemet bestemmer det korrekt rum, når der skiftes rum, hvormed fornemmelsen af en sømløs musikoplevelse i overgangen fra et rum til et andet vil mindskes. Dette er en nødvendig afvejning af, hvad der ønskes at fokuseres på, og konsekvenserne er beskrevet i sektion 6.5.

8.9 VIDERE ARBEJDE OG ANVENDELSESMULIGHEDER

Ud fra den viden, der er opnået i dette projekt, kunne det have været relevant at vide fra starten, hvor stor forskellen imellem nøjagtighederne for SVM modellerne og de neurale netværk ville være i forhold til mængden af indsamlet data. Med projektets fokus på anvendeligheden i et hjem kunne de neurale netværk have været undladt. I stedet for kunne der have været fokuseret mere på opbygningen af SVM modellerne for at hæve deres nøjagtigheder. Med viden om begrænsningerne for antal Wi-Fi scanninger, kunne det også have været relevant at fokusere på at hæve den gennemsnitlige nøjagtighed. Dette vil kunne give mulighed for at hæve tærsklen for, hvornår der skiftes rum og stadigvæk forvente at bruge maksimalt fire Wi-Fi scanninger.

Med viden om, at det er muligt at lave et nøjagtigt Wi-Fi baseret IPS understøttet af SVM, der kan bruges til at lave en sømløs musikoplevelse, ville et videre arbejde fokusere på særligt to ting, hvis projektet skulle fortsætte.

8.9.1 *Adaptiv træning*

Ved at undersøge mulighederne for en konstant dataindsamling og en adaptiv træning, som det er beskrevet i sektion 7.3, vil nøjagtigheden kunne opretholdes og muligvis forbedres. Uafhængigt af andre features, videreudviklinger eller valg af services, vil en adaptiv trænet model kunne gøre brugerens oplevelse bedre ved konstant at give den højest mulige nøjagtighed.

8.9.2 *Generelle muligheder*

Det andet punkt, der særligt kunne være relevant at arbejde videre med, er de generelle muligheder, IPS giver, og som dette projekt er med til at understøtte. Et relevant forskningsspørgsmål i den sammenhæng kunne være.

> I hvilket omfang er det muligt at lave en sømløs oplevelse af smarte enheder i et hjem ved brug af SVM understøttet IPS?

I dette projekt er der evalueret på og demonstreret potentialet ved at bruge et IPS, der gør brug af SVM, til en sømløs musikoplevelse. Det kunne derfor være relevant at se på mulighederne for at inkludere samtlige smarte enheder i et hjem. Dette kunne være at undersøge, hvorvidt det er muligt at bruge IPS til at slukke og tænde alle smarte lamper, ændre temperaturen, fortælle morgennyhederne når man kommer ind i køkkennet om morgenen og bruge det i lås og alarme-

ring af hjemmet.

Et system, der inkluderer lås og alarmer, kræver en høj grad af sikkerhed, og sporing kunne derfor indgå som en komponent i lås og alarmering, hvis brugeren er tæt på hjemmet. I den sammenhæng er det væsentligt, at data ikke forlader hjemmet, og at datastrømme er krypteret for at opnå fortrolighed og autenticitet.

At dette projekt har demonstreret brugbarheden af et IPS baseret på fingerprinting uden at bruge særlig hardware og med udelukkende fem minutters dataindsamling i hvert rum, viser potentialet for SVM. I en tid hvor mængden af smarte enheder stiger, og flere og flere services tilgås via telefoner, har dette potentialet til at blive taget i brug i folks hjem. Ved at have en brugervenlig GUI kan dette være til glæde og gavn for personer, der ønsker en simpel løsning med minimal konfigurering.

A

APPENDIKS

	FORVENTET ANTAL WI-FI SCANNINGER INDEN	
TÆRSKEL	KORREKT RUM	FORKERT RUM
1	1,23	5,32
2	2,75	33,6
3	4,62	184,1

Tabel 4: Tærsklen angiver antal på hinanden følgende prædiktioner for det samme rum, inden der skiftes rum. De forventede antal er udregnet ved brug af den geometriske fordeling og formlen $E(X_k) = \Sigma_{i=1}^{k} \left(\frac{1}{p^i}\right)$. Her er p den gennemsnitlige nøjagtighed på 81,2% for SVM3 med fem minutters dataindsamling i hvert rum.

	NØJAGTIGHED, %											
	FORMIDDAG			FORMIDDAG			AFTEN			AFTEN		
	SAMME TIDSPUNKT			ANDET TIDSPUNKT			SAMME TIDSPUNKT			ANDET TIDSPUNKT		
MODEL	μ_-	$\bar{x}$	μ_+	μ_-	$\bar{x}$	μ_+	μ_-	$\bar{x}$	μ_+	μ_-	$\bar{x}$	μ_+
SVM3	81,2	83,0	84,8	80,3	83,1	86,0	77,6	83,8	90,1	73,8	79,1	84,5
SVM4	55,9	61,4	66,8	55,6	63,0	70,4	63,6	66,2	68,8	57,5	62,5	67,4
NN3-B	58,5	63,0	67,5	58,7	64,9	71,0	64,5	67,4	70,3	54,8	60,3	65,8
NN3-B-P	44,6	48,1	51,6	45,1	50,0	55,0	51,1	52,3	53,5	46,0	48,7	51,4
NN3-UB	51,7	55,5	59,3	56,7	59,7	62,6	58,8	66,2	73,6	51,9	58,0	64,2
NN3-UB-P	42,2	44,6	47,1	43,9	46,2	48,6	49,9	55,2	60,4	44,5	48,3	52,2
NN4-UB	46,2	49,7	53,3	46,6	49,6	52,6	45,9	52,0	58,2	47,3	51,9	56,5

Tabel 5: Konfidensintervallerne for Figur 6. $\mu_- = \bar{x} - t_{n-1,\frac{\alpha}{2}} \cdot \frac{s}{\sqrt{n}}$ og $\mu_+ = \bar{x} + t_{n-1,\frac{\alpha}{2}} \cdot \frac{s}{\sqrt{n}}$. Værdierne er afrundet til første decimal.

	nøjagtighed, %														
	5 minutter			10 minutter			15 minutter			30 minutter			45 minutter		
model	μ_-	$\overline{x}$	μ_+	μ_-	$\overline{x}$	μ_+	μ_-	$\overline{x}$	μ_+	μ_-	$\overline{x}$	μ_+	μ_-	$\overline{x}$	μ_+
SVM3	80,1	81,2	82,3	83,6	84,7	85,9	83,7	85,2	86,7	84,0	85,7	87,3	81,9	83,9	86,0
SVM4	67,8	69,3	70,8	69,9	71,5	73,0	72,0	73,7	75,5	69,0	71,8	74,5	63,2	66,9	70,5
NN3-B	45,9	48,0	50,2	46,6	49,6	52,6	50,7	52,9	55,1	54,9	57,1	59,4	62,8	65,4	68,1
NN3-B-P	43,3	45,5	47,8	42,7	45,5	48,4	44,7	46,8	49,0	37,2	38,8	40,4	49,0	50,9	52,7
NN3-UB	46,4	48,1	49,7	46,4	49,2	52,0	49,6	51,7	53,8	58,3	60,3	62,4	58,4	61,0	63,6
NN3-UB-P	43,9	45,6	47,4	42,6	45,2	47,9	43,4	45,4	47,5	39,8	41,2	42,5	47,1	48,9	50,7
NN4-UB	35,8	37,4	38,9	38,1	39,4	40,6	39,4	41,1	42,9	48,2	49,7	51,3	50,3	52,4	54,5

Tabel 6: Konfidensintervallerne for Figur 8. $\mu_- = \overline{x} - t_{n-1,\frac{\alpha}{2}} \cdot \frac{s}{\sqrt{n}}$ og $\mu_+ = \overline{x} + t_{n-1,\frac{\alpha}{2}} \cdot \frac{s}{\sqrt{n}}$. Værdierne er afrundet til første decimal.

REFERENCER

[1] S. Andersen. *Notesamling til statistik: HA og HD 1. del*. Aarhus Universitet, Business og Social Sciences, 2014.

[2] P. V. Bahl og V. N. Padmanabhan. *Enhancements to the RADAR User Location and Tracking System*. Tekn. rapp. MSR-TR-2000-12. 2000, s. 13. URL: https://www.microsoft.com/en-us/research/publication/enhancements-to-the-radar-user-location-and-tracking-system/.

[3] P. V. Bahl og V. N. Padmanabhan. „RADAR: an in-building RF-based user location and tracking system". I: *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*. Bd. 2. 2000, 775–784 vol.2. DOI: 10.1109/INFCOM.2000.832252.

[4] S. BelMannoubi og H. Touati. „Deep Neural Networks for Indoor Localization Using WiFi Fingerprints". I: *Mobile, Secure, and Programmable Networking*. Red. af Éric Renault, Selma Boumerdassi, Cherkaoui Leghris og Samia Bouzefrane. Cham: Springer International Publishing, 2019, s. 247–258. ISBN: 978-3-030-22885-9.

[5] C. M. Bishop og N. M. Nasrabadi. *Pattern recognition and machine learning*. Bd. 4. 4. Springer, 2006, s. 605–652.

[6] A. A. Careem, W. H. Ali og M. H. Jasim. „Wirelessly Indoor Positioning System based on RSS Signal". I: *2020 International Conference on Computer Science and Software Engineering (CSASE)*. 2020, s. 238–243. DOI: 10.1109/CSASE48920.2020.9142111.

[7] H. B. Christensen. *Flexible, reliable software: using patterns and agile development*. CRC Press, 2010.

[8] D. Dardari, P. Closas og P. M. Djurić. „Indoor Tracking: Theory, Methods, and Technologies". I: *IEEE Transactions on Vehicular Technology* 64.4 (2015), s. 1263–1278. DOI: 10.1109/TVT.2015.2403868.

[9] T. T. Dinh, N. Duong og K. Sandrasegaran. „Smartphone-based indoor positioning using BLE iBeacon and reliable lightweight fingerprint map". I: *IEEE Sensors Journal* 20.17 (2020), s. 10283–10294.

[10] I. Goodfellow, Y. Bengio og A. Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[11] Y. Gu, A. Lo og I. Niemegeers. „A survey of indoor positioning systems for wireless personal networks“. I: *IEEE Communications Surveys Tutorials* 11.1 (2009), s. 13–32. DOI: 10.1109/SURV.2009.090103.

[12] J. Hightower og G. Borriello. „Location systems for ubiquitous computing“. I: *Computer* 34.8 (2001), s. 57–66. DOI: 10.1109/2.940014.

[13] G. Keller. *Statistics for management and economics*. Cengage Learning, 2014.

[14] H. H. Kim, K. N. Ha, S. Lee og K. C. Lee. „Resident Location-Recognition Algorithm Using a Bayesian Classifier in the PIR Sensor-Based Indoor Location-Aware System“. I: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 39.2 (2009), s. 240–245. DOI: 10.1109/TSMCC.2008.2008099.

[15] R. Kirk og J. Newmarch. „A location-aware, service-based audio system“. I: *Second IEEE Consumer Communications and Networking Conference, 2005. CCNC. 2005*. 2005, s. 343–347. DOI: 10.1109/CCNC.2005.1405194.

[16] Y. Körber. „Device Selection for Speech Output in a Multi-User Scenario“. I: *Proceedings of the 16th International Conference on Mobile and Ubiquitous Multimedia*. MUM '17. Stuttgart, Germany: Association for Computing Machinery, 2017, 363–370. ISBN: 9781450353786. DOI: 10.1145/3152832.3156624. URL: https://doi.org/10.1145/3152832.3156624.

[17] Y. Körber, M. Feld og T. Schwartz. „Pervasive audio playback in cyber-physical environments“. I: *2017 Intelligent Systems Conference (IntelliSys)*. 2017, s. 531–541. DOI: 10.1109/IntelliSys.2017.8324346.

[18] C. Langlois, S. Tiku og S. Pasricha. „Indoor Localization with Smartphones: Harnessing the Sensor Suite in Your Pocket“. I: *IEEE Consumer Electronics Magazine* 6.4 (2017), s. 70–80. DOI: 10.1109/MCE.2017.2714719.

[19] P. Lazik, N. Rajagopal, O. Shih, B. Sinopoli og A. Rowe. „ALPS: A Bluetooth and Ultrasound Platform for Mapping and Localization“. I: *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*. SenSys '15. Seoul, South Korea: Association for Computing Machinery, 2015, 73–84. ISBN: 9781450336314. DOI: 10.1145/2809695.2809727. URL: https://doi.org/10.1145/2809695.2809727.

[20] H. Liu, H. Darabi, P. Banerjee og J. Liu. „Survey of Wireless Indoor Positioning Techniques and Systems“. I: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and*

Reviews) 37.6 (2007), s. 1067–1080. DOI: 10.1109/TSMCC.2007.905750.

[21] Z. P. Mabunga, J. C. dela Cruz, A. C. Samortin og R. R. Maaliw. „Utilization of Different Wireless Technologies' RSSI for Indoor Environment Classification Using Support Vector Machine". I: *2021 IEEE 12th Control and System Graduate Research Colloquium (ICSGRC)*. 2021, s. 233–237. DOI: 10.1109/ICSGRC53186.2021.9515242.

[22] R. C. Martin. „Design principles and design patterns". I: *Object Mentor* 1.34 (2000), s. 597.

[23] B. Mrazovac, M. Z. Bjelica, I. Papp og N. Teslic. „Smart audio/video playback control based on presence detection and user localization in home environment". I: *2011 Second Eastern European Regional Conference on the Engineering of Computer Based Systems*. IEEE. 2011, s. 44–53.

[24] T. L. N. Nguyen, T. D. Vy, K. Kim, C. Lin og Y. Shin. „Smartphone-Based Indoor Tracking in Multiple-Floor Scenarios". I: *IEEE Access* 9 (2021), s. 141048–141063. DOI: 10.1109/ACCESS.2021.3119577.

[25] D. Norman. *The design of everyday things: Revised and expanded edition*. Basic books, 2013.

[26] V. Otsason, A. Varshavsky, A. LaMarca og E. De Lara. „Accurate GSM indoor localization". I: *UbiComp 2005: Ubiquitous Computing: 7th International Conference, UbiComp 2005, Tokyo, Japan, September 11-14, 2005. Proceedings 7*. Springer. 2005, s. 141–158.

[27] B. Shneiderman og C. Plaisant. *Designing the user interface: Strategies for effective human-computer interaction*. Pearson Education India, 2010.

[28] H. Sørensen, M. G. Kristensen, J. Kjeldskov og M. B. Skov. „Proxemic Interaction in a Multi-Room Music System". I: *Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration*. OzCHI '13. Adelaide, Australia: Association for Computing Machinery, 2013, 153–162. ISBN: 9781450325257. DOI: 10.1145/2541016.2541046. URL: https://doi.org/10.1145/2541016.2541046.

[29] Z. Turgut, S. Üstebay, G. Z. G. Aydın og A. Sertbaş. „Deep Learning in Indoor Localization Using WiFi". I: *International Telecommunications Conference*. Red. af Ali Boyaci, Ali Riza Ekti, Muhammed Ali Aydin og Serhan Yarkan. Singapore: Springer Singapore, 2019, s. 101–110. ISBN: 978-981-13-0408-8.

[30] J. Xiong og K. Jamieson. „ArrayTrack: A fine-grained indoor location system". I: *Proc. 10th USENIX Symp. Netw. Syst. Design Implement. (NSDI)* (2013), s. 71–84.