PERSONAL BIG DATA

A privacy-centred selective cloud computing approach to progressive user modelling on mobile devices.

Sebastian Meier (Dipl.-Designer, MA)

Univ.-Diss.

zur Erlangung des akademischen Grades
'doctor rerum naturalium'
(Dr. rer. nat.)
in der Wissenschaftsdisziplin Geographie - Geoinformatik

eingereicht an der
Mathematisch-Naturwissenschaftlichen Fakultät
Institut für Geographie
der Universität Potsdam

Ort und Tag der Disputation: Potsdam, 7.12.2017

Hauptbetreuer: Prof. Dr. Hartmut Asche

weitere Gutachter: Prof. Dr. Frank Heidmann Prof. Dr. Till Nagel

Sebastian Meier: *Personal Big Data,* A privacy-centred selective cloud computing approach to progressive user modelling on mobile devices. ©2017

This thesis was written in LATEX, making use of André Miede's delightful dissertation style template.

Published online at the
Institutional Repository of the University of Potsdam:
URN urn:nbn:de:kobv:517-opus4-406696
http://nbn-resolving.de/urn:nbn:de:kobv:517-opus4-406696

ABSTRACT

Many users of cloud-based services are concerned about questions of data privacy. At the same time, they want to benefit from smart data-driven services, which require insight into a person's individual behaviour. The modus operandi of user modelling is that data is sent to a remote server where the model is constructed and merged with other users' data. This thesis proposes *selective cloud computing*, an alternative approach, in which the user model is constructed on the client-side and only an abstracted generalised version of the model is shared with the remote services.

In order to demonstrate the applicability of this approach, the thesis builds an exemplary client-side user modelling technique. As this thesis is carried out in the area of Geoinformatics and spatio-temporal data is particularly sensitive, the application domain for this experiment is the analysis and prediction of a user's spatio-temporal behaviour.

The user modelling technique is grounded in an innovative conceptual model, which builds upon spatial network theory combined with time-geography. The spatio-temporal constraints of time-geography are applied to the network structure in order to create individual spatio-temporal action spaces. This concept is translated into a novel algorithmic user modelling approach which is solely driven by the user's own spatio-temporal trajectory data that is generated by the user's smartphone.

While modern smartphones offer a rich variety of sensory data, this thesis only makes use of spatio-temporal trajectory data, enriched by activity classification, as the input and foundation for the algorithmic model. The algorithmic model consists of three basal components: locations (vertices), trips (edges), and clusters (neighbourhoods).

After preprocessing the incoming trajectory data in order to identify locations, user feedback is used to train an artificial neural network to learn temporal patterns for certain location types (e.g. work, home, bus stop, etc.). This Artificial Neural Network (ANN) is used to automatically detect future location types by their spatio-temporal patterns. The same is done in order to predict the duration of stay at a certain location. Experiments revealed that neural nets were the most successful statistical and machine learning tool to detect those patterns. The location type identification algorithm reached an accuracy of 87.69%, the duration prediction on binned data was less successful and deviated by an average of 0.69 bins. A challenge for the location type classification, as well as for the subsequent components, was the imbalance of trips and connections as well as the low accuracy of the trajectory data. The imbalance is grounded in the fact that most users exhibit strong habitual patterns (e.g. home > work), while other patterns are rather rare by

comparison. The accuracy problem derives from the energy-saving location sampling mode, which creates less accurate results.

Those locations are then used to build a network that represents the user's spatio-temporal behaviour. An initial untrained ANN to predict movement on the network only reached 46% average accuracy. Only lowering the number of included edges, focusing on more common trips, increased the performance. In order to further improve the algorithm, the spatial trajectories were introduced into the predictions. To overcome the accuracy problem, trips between locations were clustered into so-called spatial corridors, which were intersected with the user's current trajectory. The resulting intersected trips were ranked through a k-nearest-neighbour algorithm. This increased the performance to 56%. In a final step, a combination of a network and spatial clustering algorithm was built in order to create clusters, therein reducing the variety of possible trips. By only predicting the destination cluster instead of the exact location, it is possible to increase the performance to 75% including all classes.

A final set of components shows in two exemplary ways how to deduce additional inferences from the underlying spatio-temporal data. The first example presents a novel concept for predicting the 'potential memorisation index' for a certain location. The index is based on a cognitive model which derives the index from the user's activity data in that area. The second example embeds each location in its urban fabric and thereby enriches its cluster's metadata by further describing the temporal-semantic activity in an area (e.g. going to restaurants at noon).

The success of the client-side classification and prediction approach, despite the challenges of inaccurate and imbalanced data, supports the claimed benefits of the client-side modelling concept. Since modern data-driven services at some point do need to receive user data, the thesis' computational model concludes with a concept for applying generalisation to semantic, temporal, and spatial data before sharing it with the remote service in order to comply with the overall goal to improve data privacy. In this context, the potentials of ensemble training (in regards to ANNs) are discussed in order to highlight the potential of only sharing the trained ANN instead of the raw input data.

While the results of our evaluation support the assets of the proposed framework, there are two important downsides of our approach compared to server-side modelling. First, both of these server-side advantages are rooted in the server's access to multiple users' data. This allows a remote service to predict spatio-temporal behaviour that a user has not exhibited but another user has. The same accounts for the imbalance in the user-specific data, which represents the second downside. While minor classes will likely be minor classes in a bigger dataset as well, for each class, there will still be more variety than in the user-specific dataset. The author emphasises that the approach presented in this work holds the potential to change the privacy paradigm in modern data-driven services. Finding combinations of

client- and server-side modelling could prove a promising new path for data-driven innovation.

Beyond the technological perspective, throughout the thesis the author also offers a critical view on the data- and technology-driven development of this work. By introducing the client-side modelling with user-specific artificial neural networks, users generate their own algorithm. Those user-specific algorithms are influenced less by generalised biases or developers' prejudices. Therefore, the user develops a more diverse and individual perspective through his or her user model. This concept picks up the idea of critical cartography, which questions the status quo of how space is perceived and represented.

WISSENSCHAFTLICHE ZUSAMMENFASSUNG

Die Nutzung von modernen digitalen Diensten und Cloud-Services geht häufig einher mit einer Besorgtheit um die Sicherheit der eigenen Privatsphäre. Gleichzeitig zeigt sich, dass die Nutzung eben dieser Dienste nicht rückläufig ist. Dieses Phänomen wird in der Wissenschaft auch als Privacy-Paradox bezeichnet (Barnes, 2006). Viele digitale Dienste bauen einen Großteil ihrer Funktionalitäten auf NutzerInnendaten auf. Der Modus Operandi bei diesen Diensten ist bisher, die Daten der NutzerInnen an einen Server zu schicken, wo diese verarbeitet, analysiert und gespeichert werden. Die vorliegende Doktorarbeit schlägt ein alternatives Konzept vor: *Selective Cloud Computing*. Kern dieses Konzeptes ist die Verlagerung der NutzerInnen-Modellierung auf die privaten Endgeräte, wodurch für weitere Services nur ein abstrahiertes Daten- und NutzerInnenmodel mit den externen Diensten geteilt wird.

Um dieses Konzept auf seine Machbarkeit und Performanz zu überprüfen wird im Rahmen dieser Arbeit ein beispielhafter Prozess für die nutzerInnenseitige Modellierung von raumzeitlichen Informationen entwickelt. Da raumzeitliche Informationen mit zu den sensibelsten persönlichen Daten gehören, bietet die Verortung der vorliegende Arbeit im Bereich der Geoinformatik für das Anwendungsfeld der NutzerInnen-Modellierung einen passenden disziplinären Rahmen.

Die NutzerInnen-Modellierung fußt auf einem innovativen konzeptuellen Modell, welches Theorien zu räumlichen Netzwerken und Hägerstrands Theorie der Zeitgeographie miteinander kombiniert (Hägerstrand, 1970). Hierbei werden die von Hägerstrand entwickelten raumzeitlichen Einschränkungen (Constraints) auf das Netzwerkmodel übertragen, wodurch individuelle Aktionsräume konstituiert werden. Dieses Model wird schließlich in ein algorithmisches Computermodel übersetzt, dessen Operationen ausschließlich die Daten verarbeiten und nutzen, die auf den Smartphones der NutzerInnen generiert werden.

Moderne Smartphones bieten für die Datengenerierung gute Voraussetzungen, da sie den Zugriff auf eine ganze Bandbreite an Sensoren und anderen Datenquellen ermöglich. Die vorliegende Arbeit beschränkt sich dabei jedoch auf die raumzeitlichen Informationen, welche über die Ortungsfunktionen des Geräts produziert werden (Trajectories). Die Trajektorien werden angereichert durch Aktivitätsklassifikationen (z.B. Laufen, Radfahren, etc.), welche von der App, die diese Daten aufzeichnet, zugeordnet werden. Das Computermodel basiert auf diesen Daten und gliedert diese in drei grundlegende Komponenten: 1) Orte (Knotenpunkte) 2) Trips (Kanten) und 3) Cluster (Nachbarschaften).

Zu Beginn der algorithmischen Verarbeitung werden die eingehenden Daten optimiert und analysiert, um in einem ersten Schritt geographische Orte zu identifizieren. Um diese Orte nun mit semantischen Informationen anzureichern wird ein automatisierter Algorithmus über User-Feedback trainiert, welcher die Orts-Typen selbstständig erkennt (z.B. Zuhause, Arbeitsplatz, Haltestelle). Der Algorithmus basiert auf einem künstlichen neuronalen Netz, welches versucht, Muster in den Daten zu erkennen. Die Entscheidung, neuronale Netze in diesem Prozess einzusetzen, ergab sich aus einer Evaluation verschiedener Verfahren der statistischen Klassifizierung und des maschinellen Lernens. Das Verfahren zur Erkennung der Orts-Typen erreichte unter Zuhilfenahme eines künstlichen neuronalen Netz eine Genauigkeit von 87.69% und war damit das akkurateste. Eine weitere Einsatzmöglichkeit solcher neuronalen Netze ist bei der Vorhersage von Aufenthaltsdauern an bestimmten Orten, welche im Durschnitt 0.69 Klassen vom korrekten Ergebnis abwich. Eine große Herausforderung für alle Module war sowohl die Ungenauigkeit der Rohdaten, also auch die ungleichmäßige Verteilung der Daten. Die Ungenauigkeit ist ein Resultat der Generierung der Positionsinformationen, welche zugunsten eines geringeren Energieverbrauchs der mobilen Geräte Ungenauigkeiten in Kauf nehmen muss. Die ungleichmäßige Verteilung ergibt sich wiederum durch häufig wiederkehrende Muster (z.B. Fahrten zur Arbeit und nach Hause), welche im Vergleich zu anderen Aktivitäten vergleichsweise häufig auftreten und die Datensätze dominieren.

Die Orte, die in der ersten Phase identifiziert und klassifiziert wurden, werden im nächsten Schritt für die Konstruktion des eigentlichen räumlichen Netzwerks genutzt. Basierend auf den über einen bestimmten Zeitraum gesammelten Daten der NutzerInnen und im Rückgriff auf Hägerstrands Einschränkungsprinzip werden Vorhersagen über mögliche raumzeitliche Verhaltensweisen im nutzerspezifischen Netzwerk gemacht. Hierzu werden Methoden des maschinellen Lernens, in diesem Fall künstliche neuronale Netze und Nächste-Nachbarn-Klassifikation (k-nearest-neighbour), mit Methoden der Trajektorien-Analyse kombiniert. Die zugrundeliegenden Orts- und Bewegungsinformationen werden unter Anwendung von Netzwerk-Nachbarschafts-Methoden und klassischen räumlichen Gruppierungsmethoden (Clustering) für die Optimierung der Algorithmen verfeinert. Die aus diesen Schritten resultierende Methodik erreichte eine Genauigkeit von 75%

bei der Vorhersage über raumzeitliches Verhalten. Wenn man Vorhersagen mit einbezieht, bei denen der korrekte Treffer auf Rang 2 und 3 der Nächste-Nachbarn-Klassifikation liegt, erreichte die Methodik sogar eine Vorhersagen-Genauigkeit von 90%.

Um zu erproben, welche weiteren Schlussfolgerungen über die NutzerInnen basierend auf den zugrundeliegenden Daten getroffen werden könnten, werden abschließend zwei beispielhafte Methoden entwickelt und getestet: zum einen werden die Trajektorien genutzt um vorherzusagen, wie gut eine NutzerIn ein bestimmtes Gebiet kennt (Potential Memorisation Index). Zum anderen werden zeitlich-semantische Muster für Orts-Cluster extrahiert und darauf basierend berechnet, wann welche Aktivitäten und spezifischen Orte innerhalb eines Clusters für die NutzerIn potenziell von Interesse sind.

Trotz der Herausforderungen, die mit den unausgeglichenen Datensätzen und teilweise fehlerhaften Daten einhergehen, spricht die dennoch vergleichsweise hohe Präzision der nutzerseitigen Klassifizierungs- und Vorhersagemethoden für den in dieser Arbeit vorgestellten Ansatz der nutzerseitigen Modellierung. In einem letzten Schritt kontextualisiert die vorliegende Arbeit die erstellten Ansätze in einem realweltlichen Anwendungsfall und diskutiert den Austausch der generierten Daten mit einem datengestützten Dienst. Hierzu wird das Konzept der Generalisierung genutzt, um im Sinne des Schutzes der Privatsphäre abstrahierte Daten mit einem Dienst zu teilen.

Obgleich der positiven Ergebnisse der Tests gibt es auch klare Nachteile im Vergleich zur klassischen serverseitigen Modellierung, die unter Einbezug mehrerer aggregierter NutzerInnenprofile stattfindet. Hierzu zählt zum einen, dass unterrepräsentierte Klassen in den Daten schlechter identifiziert werden können. Zum anderen ergibt sich der Nachteil, dass nur Verhaltensweisen erkannt werden können, die bereits zuvor von der NutzerIn selber ausgeübt wurden und somit in den Daten bereits enthalten sind. Im Vergleich dazu besteht bei serverseitiger Modellierung auf der Basis zahlreicher Personenprofile der Zugriff auf ein breiteres Spektrum an Verhaltensmustern und somit die Möglichkeit, diese Muster mit dem der NutzerIn abzugleichen, ohne dass dieses Verhalten bereits in ihren nutzerseitig generierten Daten abgelegt ist. Nichtsdestotrotz zeigt die Arbeit, welches Potential die nutzerseitige Modellierung bereithält - nicht nur in Bezug auf den größeren Schutz der Privatsphäre der NutzerInnen, sondern ebenso in Hinsicht auf den Einsatz von Methoden des verteilten Rechnens (distributed computing). Die Kombination von beidem, nutzerInnen- *und* serverseitiger Modellierung, könnte ein neuer und vielversprechender Pfad für datengetriebene Innovation darstellen.

Neben der technologischen Perspektive werden die entwickelten Methoden einer kritischen Analyse unterzogen. Durch das Einbringen der nutzerseitigen Modellierung in Form von benutzerspezifischen künstlichen neuronalen Netzen trainieren die NutzerInnen ihre eigenen Algorithmen auf ihren mobilen Geräten. Diese spezifischen Algorithmen sind weniger stark von generalisierten Vorannahmen, Vorurteilen und möglichen Befangenheiten der

EntwicklerInnen beeinflusst. Hierdurch haben NutzerInnen die Möglichkeit, vielfältigere und persönlichere Perspektiven auf ihre Daten und ihr Verhalten zu generieren. Dieses Konzept setzt Ideen der kritischen Kartographie fort, in welcher der Status Quo der Wahrnehmung und Repräsentation des Raumes hinterfragt werden.

ALLGEMEINVERSTÄNDLICHE ZUSAMMENFASSUNG

Moderne digitale Dienste basieren immer häufiger auf tiefgehenden datenbasierten Einblicken in das Verhalten ihrer NutzerInnen. Von personalisierten Empfehlungen in Online-Shops bis hin zu sogenannten *intelligenten persönlichen AssistentInnen*. Letztere sammeln beispielsweise detaillierte Informationen über das gesamte speicherbare Verhalten ihrer NutzerInnen. Hierzu zählen unter anderem Bewegungsinformationen, welche dafür genutzt werden interessante Orte zu empfehlen, Navigationsfunktionen zu personalisieren und orts- und kontextabhängig möglicherweise relevante Informationen anzuzeigen. Diesem Trend zum Datensammeln steht ein öffentlicher Diskurs um die Privatsphäre der NutzerInnen gegenüber. Dieser Diskurs stellt allerdings nur eine Seite der Medaille dar. Denn Studien zeigen, dass die meisten NutzerInnen – obgleich sie Datenschutz und Privatsphäre wichtig finden – ihr digitales Verhalten nicht entsprechend anpassen. Dieses Phänomen ist bekannt als Privatsphären-Paradox (Barnes, 2006). Allerdings bietet die Gesetzgebung der Bundesrepublik Deutschland eine gute Grundlage dafür, die Ausgestaltung dieser digitalen Dienste zu überdenken. Die entsprechenden Gesetze zu Datensparsamkeit und Datenvermeidung sind daher auch leitgebend für diese Arbeit. Die vorliegende Dissertation beschäftigt sich mit *der Frage, ob der Datenaustausch solcher digitalen Dienste unter Beibehaltung ihrer Funktionalitäten zu Gunsten der Privatsphäre der NutzerInnen umgestaltet werden können*.

Der Modus Operandi in den meisten datengestützten Anwendungen sieht vor, dass alle Daten der NutzerInnen an einen Server geleitet werden. Diese Arbeit stellt diesem Modell eine Alternative gegenüber, bei der die Daten der NutzerInnen auf deren Geräten gespeichert werden (z.B. einem Smartphone). Dort werden die Daten lokal verarbeitet und analysiert, bevor sie letztendlich in einem abstrahierten Format mit einem Dienstleister geteilt werden. Da räumliche Informationen über z.B. Aufenthaltsorte zu den sensibelsten Daten einer/s NutzerIn zählen, zeigt diese Arbeit exemplarisch am Beispiel von Bewegungsinformationen wie solch ein Konzept umgesetzt werden könnte, entwickelt entsprechende Algorithmen und testet diese auf ihre Gebrauchstauglichkeit und Genauigkeit. Kern der entwickelten Algorithmen sind automatisierte Verfahren zur Erkennung von Verhaltensmustern. Hierzu gehört die Erkennung, um welchen Ortstyp es sich handelt (z.B. Wohnort oder Arbeitsplatz) oder zum Beispiel die Vorhersage über die Dauer eines Aufenthaltes an einem bestimmten Ort. Der komplexeste dieser Algorithmus berechnet potentielle Ziele der NutzerInnen, während sie sich bewegen. All diese Berechnungen werden ausschließlich lokal auf Basis der NutzerInnendaten durchgeführt, ohne die Einbindung externer Dienste. Viele dieser Funktionen nutzen sogenannte Verfahren des maschinellen Lernens, wie z.B. künstliche neuronale Netze. Diese modernen Verfahren sind Alternativen zur klassischen Statistik und erlauben es, Muster in komplexen Daten zu erkennen. Wurde zum Beispiel das Ziel einer Reise mit großer Wahrscheinlichkeit vorhergesagt, können der /dem NutzerIn Empfehlungen gemacht werden. Lediglich an dieser Stelle im Prozess müssen Daten mit einem externen Dienst geteilt werden. Um die Privatsphäre der NutzerInnen dennoch zu wahren, wird nur ein abstrahiertes Model der NutzerIn geteilt und nicht die zugrundeliegenden Rohdaten.

Die Arbeit stellt eine Alternative zum Modus Operandi von datengestützten Systemen vor. Die Algorithmen wurden auf ihre Performanz und Genauigkeit getestet. Die vorliegende Dissertation zeigt somit das Potential von selektivem Cloud-Computing und liefert einen Beitrag zum Diskurs um Privatsphäre im Bereich digitaler Dienste.

PUBLICATIONS

Some ideas and figures have appeared previously in the following publications:

- Meier, Sebastian: **The Marker Cluster: A Critical Analysis and a New Approach to a Common Web-based Cartographic Interface Pattern**. – *International Journal of Agricultural and Environmental Information Systems (IJAEIS), Volume 7, Issue 1*, January - March 2016
- Meier, Sebastian: **Location based Applications (LBA)**. – *Interfaces, Journal (UK), Issue 77*, 2008
- Meier, Sebastian; Glinka, Katrin: **Psychogeography in the Age of the Quantified Self - Mental Map Modelling with Georeferenced Personal Activity Data**. – *LNG&C - Advances in Cartography and GIScience*, Springer, 2017
- Meier, Sebastian: **Enhancing Location Recommendation Through Proximity Indicators, Areal Descriptors, and Similarity Clusters**. – *LNG&C - Progress in Location-Based Services*, Springer, 2016
- Meier, Sebastian: **Visualizing Large Spatial Time Series Data on Mobile Devices: Combining the HeatTile System with a Progressive Loading Approach**. – *LNG&C - Cartography - Maps Connecting the World, 27th ICC*, Springer, 2015
- Meier, Sebastian; Heidmann, Frank; Thom, Andreas: **Heattile, a New Method for Heatmap Implementations for Mobile Web-based Cartographic Applications**. – *Thematic Cartography for the Society*, Springer, 2014
- Meier, Sebastian; Heidmann, Frank: **Too Many Markers, Revisited: An Empirical Analysis of Web-Based Methods for Overcoming the Problem of Too Many Markers in Zoomable Mapping Applications**. – *Computational Science and Its Applications (ICCSA)*, 2014, Guimarães, Portugal
- Meier, Sebastian; Landstorfer, Johannes; Werner, Julia; Wettach, Reto; Knörig, Andre; Cohen, Jonathan; Sommerwerk, Andreas: **A Real-world Mobile Prototyping Framework for Location-and Context-based Services**. – *Published at the Wireless Communication and Information (WCI)*, 2012, Berlin

PREFACE

The first academic publication I wrote, back in 2008, was centred around novel concepts for location-based application distribution (Meier and Hirt, 2008). The paper was written in the advent of the iPhone, before *Apple's* own App store was introduced. The paper described a technical and conceptual framework for using Wireless Local Area Network (WiFi) fingerprinting to distribute applications with a local relevance, like a public transport Application (App) when one stands close to a bus stop. This method is now integrated in most smartphones today. Since this first encounter with Location-Based Applications (LBAs) I have repeatedly returned to the topic of LBAs and Location-Based Services (LBSs). In 2012, while working for the company *IxDS* I developed a prototyping framework for LBSs (Meier et al., 2012). Later in 2013 I was employed as a research associate working for the Interaction Design Lab (IDL), where I was part of a research conglomerate exploring the opportunities of LBSs for tourists in the metropolitan area of Berlin. The project resulted in several publications centred around Location Recommendation (LR) (Meier, 2016; Meier et al., 2014b; Meier et al., 2014a). Especially the latter project also led me to my final dissertation project. For me, LBSs and LBAs were one of the first successfully adopted ubiquitous computing applications for consumers, which attempted to adapt their computational behaviour to the location or rather context of the user. In so doing, they combined real-world interactions with digital user experiences.

For those who have met me during my doctoral journey (or depending on who you ask, the doctoral rollercoaster as a perhaps more suitable metaphor), the final focus of the thesis might come as a surprise. While contextuality has remained an important factor, the focus shifted. I started out exploring the opportunities available through context- and location-based data visualisations. I realised that in order to apply context models to visualisations that that would also require me to solve several rudimentary problems of defining contexts and connecting the data space with the user's real-world space. This in turn led me deeper into the algorithmic modelling of contexts and users. Due to my background in Human Computer Interaction (HCI) this led me to the question of how to let people interact with their behavioural models and how to help users regain control over their personal data by controlling their modelled data.

This shift in focus of my doctoral thesis was in sync with my general research focus. I started out with a strong focus on data-driven interfaces and visualisations. The deeper I drilled into data-driven human computer interfaces, the more important algorithms became. This continued to the point that, like an iceberg visible above water, the visual output came to represent only a small percentage of the substantial algorithmic workings underneath.

I believe this to be a trend in the design of interfaces, visualisations, and user experiences in general. This represents a challenge to traditional design and user experience designers as well as their education. In order to design novel digital user experiences, we need to acquire a deep understanding of the data-driven algorithmic processes underneath. In this sense, this thesis presents a new approach for regaining control over our personal data by exploring algorithmic techniques for user modelling.

ACKNOWLEDGMENTS

I would like to express my gratitude to my supervisors Prof. Dr. Harmut Asche and Prof. Dr. Frank Heidmann for supporting me on my academic journey. Prof. Dr. Asche's comments helped me remain focused and not lose sight of the goal ahead. Prof. Dr. Heidmann provided rich academic guidance and sincere motivation along this at-times arduous long haul towards the finishing line.

I want to thank my colleagues and students for all the exchange of ideas and thoughts over the years. I am especially grateful for the inspiration provided by the works of Prof. Dr. Till Nagel and the critical and helpful thoughts of Prof. Dr. Jason Dykes during my visit at the GiCentre in London.

Such a long journey cannot be undertaken without financial support. Therefore I would like to thank the DFG's graduate program on „Visibility and Visualisation" for inviting me to take part in their scholarship program. I remain thankful as well for the DAAD, which supported my conference activities, and the University of Applied Sciences Potsdam for employing me as a research associate.

The interdisciplinary graduate program as well as my colleagues from the Humanities helped me broaden my perspective on the often so technology-driven Computer Sciences. In particular the discussions and publications with my colleague Katrin Glinka helped me refine my critical perspective.

I want to thank Christina Chiknas for coping with my English and helping me refine the written language of my dissertation.

A heartfelt thank you to my friends and family for supporting me on this long journey. I am especially thankful to my parents Angelika and Dieter who supported my curiosity in the mysteries of life. I am most grateful to Katrin for supporting me on this challenging undertaking, and especially for standing with me during the last few months of finishing this project.

Last but not least, this thesis wouldn't have been possible without making use of many open source tools and open data. It would be too long a list to highlight every library used. But beyond the references cited in this thesis, I would also like to dearly thank everyone who put work into D3, Node.js, SQLite & Spatialite, Turf.js, Moment.js, TensorFlow, Weka, Convnet.js, and LaTeX. In order to commemorate all the effort put into these tools, all code related to this thesis will also be published under an open license on GitHub.

CONTENTS

Abstract iii
Preface xiii
Acknowledgements xiv
List of Figures xvii
List of Tables xxii
Listings xxii
Acronyms xxiii

1 INTRODUCTION 1
1.1 Motivation 1
1.2 Thesis Objectives 4
1.3 Discourse & Critical Perspectives 5
1.4 Thesis Structure 6

2 FRAMING & FOUNDATION 8
2.1 Application Domain 8
2.1.1 Location Recommendation 8
2.1.2 Context 11
2.2 Technological Foundation 13
2.2.1 Spatial Data from Smartphones 14
2.2.2 Data Refinement through User Feedback 16
2.2.3 Client-side Computing vs. Cloud computing 18
2.2.4 Technology Stack 22
2.2.5 Related Work 23

3 CONCEPTUAL MODEL 27
3.1 Modelling 27
3.2 Urban Network Theory 30
3.3 Conceptual Conclusion 38

4 INTEROPERABILITY & GENERALISATION 39

5 THE INDIVIDUAL IN THE DATA 43
5.1 Individual Perspectives 43
5.2 Critical Reflection on Data & Modelling 45

6 COMPUTATIONAL MODEL 48
6.1 Test Datasets 48
6.2 Spatio-temporal Data Preprocessing 50
6.3 Location Type Identification 53
6.3.1 Artificial Neural Networks 55

6.3.2 Time Use Survey 61
6.3.3 User-specific Model 63
6.3.4 Predicting Duration of Location Events 65
6.3.5 Conclusion 68
6.4 From Locations to the Network 68
6.4.1 Building the Network 70
6.4.2 Predicting Movement in the Network I 71
6.4.3 Refining the Network 75
6.4.4 Predicting Movement in the Network II 77
6.4.5 Transit Locations 82
6.4.6 Origin / Destination Clusters 83
6.4.7 Predicting Movement in the Network III 85
6.4.8 Conclusion of Network Analysis 86
6.5 From Locations and Networks to Areas 86
6.5.1 *Areal Descriptors* 88
6.5.2 *Potential Memorisation Index* 90
6.5.2.1 Temporal Data Clustering & Network Analysis 91
6.5.2.2 Evaluation of Clusters & Model 93
6.5.2.3 Conclusion 93
6.5.3 Areal Conclusion 94
6.6 Modelling Conclusion 94

7 IMPLEMENTATION 96
7.1 Event-based Recommendation 96
7.2 Generalisation & Date Exchange 98
7.2.1 Generalisation of Data for improved Privacy 98
7.2.2 Sharing trained Machine Learning Models 100

8 OUTCOME 102
8.1 Expert Evaluation 102
8.2 Envisioning Future Applications 105
8.3 Current Developments 107
8.4 Discussion 108

Bibliography 114
Appendix 130
Declaration 134

LIST OF FIGURES

Figure 1 Thesis structure. 7

Figure 2 Exemplary assisted Global Positioning System (GPS) workflow. (Adapted from Google, 2017). 15

Figure 3 First row: *Moves* Application process of improving location detection. Second row image 1 and 2: *Foursquare* asking the user to confirm or deny the detection of location visits. Last image on the lower right: *Google* location history asking the user to confirm or deny a location visit. 17

Figure 4 From left to right: 1. *Foursquare* collecting additional metadata through their users. 2. The same done by *Google* 3. *Google Now* acquiring user confirmation on content recommendation. 18

Figure 5 *Nike Training Club*. 21

Figure 6 *Geopaparazzi*. 21

Figure 7 *Foursquare*. 22

Figure 8 Set. 22

Figure 9 Technology Stack, the foundation of the conceptual Framework. In red the central components to be explored and developed in this thesis. 23

Figure 10 Modelling process, adapted from (Banks, 2010, p. 9). 29

Figure 11 Left: Public Conveyances on the Irish rail network (Griffith et al., 1838), Right: Migration patterns in the UK (Ravenstein, 1885). 30

Figure 12 Basic network structures. 30

Figure 13 Left: French wine exports (Minard, 1865), Right: Traveller network between Dijon and Mulhouse (Minard, 1845). 31

Figure 14 Cotton and wool imports to Europe (left 1858, right 1861) (Minard, 1862). 32

Figure 15 Left: Degree of centrality (darker more central), Right: Link Analysis. 33

Figure 16 From top left to bottom right: 1) *space time cube* with location L and a radius for the individual constraint limited action space 2) one location L with three prisms with varying timespan (*T1*, *T2*, *T3*) and varying constraints 3) a trip between two locations (*L1* and *L2*) 4) the same trip with a bigger timespan and therefore bigger spatial prism. 35

Figure 17 Transformation of isochrones from a purely spatial representation to a spatio-temporal representation. . . 36
Figure 18 Example isochrones for the city of London, taken from the Isoscope prototype developed by Gortana et al. (Gortana et al., 2014). 36
Figure 19 Applying the isochrone concept onto the abstracted trajectory network (left to right): 1) isochrone based on time, and historic travel information; 2) taking into account the current mode of transport, one can apply filters based on mode of transport derived from historic trip information, and reduce the network to matching trips; 3) in a similar manner the probability of a route being taken under current conditions (e.g. day of the week, hour of the day) can be added. 37
Figure 20 Hägerstrand's time-space prism as a tool for intersecting possible destination locations (DL1-5) from a given origin location (OL), within a timespan (T). . . . 37
Figure 21 Spatial relationship preserving generalisation. 39
Figure 22 Construction of the conceptual model. 42
Figure 23 Original line (dotted) and the resulting line after applying the median filter. 51
Figure 24 Spatial inactivity threshold for identifying a location. . 52
Figure 25 Simplified example for a progressive line simplification. 52
Figure 26 Location identification process. 54
Figure 27 Example of an ANN setup. 55
Figure 28 Input tensors for a 5x5 pixel grey-scale image and an RGB image. 56
Figure 29 A highly abstracted illustration adapted from Pyle and Jose, 2015, to discuss the difference between machine learning and linear statistics. 60
Figure 30 UK time use survey data for weekday (left) and weekend (right). Each area represents one location type. The y-axis shows the percentage of participants currently in a location type. The x-axis highlights 10-minute intervals of a 24-hour day, starting at 04:00. As an example the area for transport is highlighted in black. 62
Figure 31 Performance development over n-month of training data. x represents the number trained month and y the accuracy. The thick line indicates the average performance. 65

Figure 32 The y-axis shows days of the week, the x-axis on the left the starting point of an event on a 24 scale, the x-axis on the right the duration of the event. Both show clear cluster-distinctions between the days 0 and 1 and the rest of the week. 69

Figure 33 The y-axis represents the location types, the x-axis are the same as in the figure above. The left graphic shows that the location home shares clusters with all other locations (0), as it exhibits a broad variety of uses. The right graphic shows that the categories doctor, transport, and food (1,3,5) have similarly short durations and belong accordingly to similar clusters. . 69

Figure 34 Network of locations, each edge in the visualisation represents one or multiple edges. 70

Figure 35 Network of locations, each edge represents a group of edges in the same direction, the weight (number of edges) is visualised through the width of the lines. . . 71

Figure 36 The x-axis in both graphs represents the number of visits to a location. The y-axis shows the number of edges connected to a location. The radius of each circle indicates the average time spent at each location. The colour highlights the degree of centrality. The upper graph is built on a linear scale and the lower uses a linear scale. The difference between the graphs highlights the problem of this imbalanced nature. 72

Figure 37 The visualisation shows the author's spatial network for the metropolitan area of Berlin. The colour of the vertices is derived from their (top to bottom) closeness centrality, degree centrality, between centrality, and page rank (in 2 zoom levels) (black = strong to light grey = weak). 73

Figure 38 Visual explanation of temporal enrichment of edges (from left to right): 1) directed edge bundles 2) bundles split up by their temporal distribution between the two vertices 3) space-time cube of two exemplary edges between the two vertices. 74

Figure 39 Prediction accuracy increase as data variation decreases. X-axis shows the threshold of minimum edge-counts that make it into the training set and the y-axis represents the accuracy. Size of the points represents the number of trip-groups in the training data using a logarithmic scale. 74

Figure 40 Visualising the problem between a simplified trip network and the actual underlying spatial network between three given locations A, B, and C. The blue dot represents the user's current position. 76
Figure 41 Creating temporary locations at activity intersections. . 76
Figure 42 Process of creating the trajectory corridors. Top to bottom: 1) Raw trajectories between the three locations. 2) Each trajectory is extended through a buffer of 100 meters. 3) Trajectories that share 60% overlap are combined into one corridor, which can then be used to intersect with the user's current location and trajectory (blue). 78
Figure 43 Corridor examples from the author's *Moves* dataset. The two highlighted examples (circles) show how minor differences between trajectories (GPS inaccuracy) are ignored due to the tolerance. 79
Figure 44 The iterations for each test trajectory (of one test subject) are mapped to 1-100 (x-axis). For each step in this iteration the knn is performed. The stacked barchart shows the result of the knn. Blue highlights the number of correct classified instances. Dark green is used when the correct location was second in the knn-results, with bright green third, yellow forth and orange fifth. The red gradient indicates matches beyond rank five. Black visualises cases in which no resulting corridor matched the expected outcome. The y-axis thereby represents the percentage of classified and unclassified cases. 80
Figure 45 Example of iterations from a corridor trip prediction (only every second iteration is shown). The darker the polygon, the higher the probability calculated by the KNN algorithm. The longer the trajectory becomes, the less viable intersections are produced. 81
Figure 46 Identifying transit locations through intersections with similar trips. 82
Figure 47 Process of building a sequence of potential destination location predictions. 83
Figure 48 Clusters of locations that are connected with one another but also across the clusters. 84
Figure 49 Graph generalisation through clustering, while maintaining graph connections. 84
Figure 50 Distance-based *DBSCAN* algorithm with a network (connectivity) filter (location in the lower right is within the distance buffer, but not connected and therefore not considered for the cluster). 85

Figure 51 The iterations for each test trajectory (for one of the test subjects) are mapped to 1-100 (x-axis). The K-Nearest-Neighbour (KNN) is performed for each step in this iteration. The stacked bar chart shows the result of the KNN. Blue highlights the number of correct classified instances. Dark green when the correct location was second in the knn-results, bright green third, yellow forth and orange fifth. The red gradient indicates matches beyond rank five. Black visualises cases where no resulting corridor matched the expected outcome. The y-axis thereby represents the percentage of classified and unclassified cases. 87

Figure 52 The four examples above show four location types and their temporal patterns clustered by days (Sunday to Saturday) and by hour (4am to 3am) in 10 minute intervals. The examples highlight that some locations exhibit very specific patterns. Home locations are for example most occupied during evening, night, early mornings, and the weekends, while work locations exhibit the exact opposite and nightlife locations are most visited during Fridays and Saturdays. The gradients nicely illustrate how temporal behaviours shift slightly and are not the exact same every day. Using those insights one can still predict the outcomes. 89

Figure 53 PMI, per transport mode (motorised transport = 1, cycling = 2 and walking = 3. 91

Figure 54 Trajectories mapped onto buildings. On top of the buildings, the network structure between the walking islands is visualised. Colour-scale: from white unknown to black well known buildings. 92

Figure 55 On the left, the same network represented in a Situationist fashion with a manual projection and weighted links (arrows). On the right, using a force-directed graph method, the city layout is rearranged and distorted by the suppression of distance. 92

Figure 56 The x-axis represents the memorisation response from participants. On the left and in the middle, the y-axis represents the algorithmically-calculated *potential memorisation index*. The figure on the left shows the raw data points with averages (red) and the figure in the middle shows an analogous box-plot with medians (red). The graph on the right visualises the percentage for modes of transport for each memorisation response value. Green: walking, purple: cycling, and blue: motorised transport. 94

Figure 57 Technology Stack, the foundation of the conceptual Framework. In red the central components to be explored and developed in this thesis. 97
Figure 58 Example for the generalisation of the spatial trajectory network. 99
Figure 59 Example for the generalisation of temporal attributes. . 99
Figure 60 Example for the generalisation of semantic attributes. . 100
Figure 61 Removing the sensitive start- and end-segments of the trajectory. 103
Figure 62 Overcome the problem of minor class predictions through an interface solution. In this scenario the location *home* received the highest probability, but for the same spatial area and the same time threshold other locations might be viable destinations as well. Brain graphic by Sergey Patutin, Knife by Mello, Groceries by Chiara Galli, all from the Noun Project. 112

LIST OF TABLES

Table 1 Comparing the performance of the time-use survey ANN against other machine learning techniques. 63
Table 2 Extract from the unbalanced example evaluation. . . . 64
Table 3 Extract from the unbalanced example evaluation. . . . 65
Table 4 Comparing the performance of the user-specific ANN against other machine learning techniques. 66
Table 5 Transformation of duration. 66
Table 6 Comparison of machine learning approaches on the location event duration dataset. 67
Table 7 Correctly classified instances for the dataset with at least 6 edges per network. 75
Table 8 Correctly classified instances for the dataset with at least 6 instances per edge group. 75

LISTINGS

Listing 1 Querrying corridor matches 77
Listing 2 Identifying possible transit locations 82

Listing 3 Example event registrations 97
Listing 4 Full list of labels in the UK time use survey for locations and their reassigned location label 0 d̄ropped . . 132

ACRONYMS

ANN	Artificial Neural Network
API	Application Programmable Interface
App	Application
CF	Collaborative Filtering
CSV	Comma-Separated-Values
DP	Differential Privacy
EU	European Union
GIS	Geographic Information System
GPS	Global Positioning System
GSM	Global System for Mobile Communication
HCI	Human Computer Interaction
IDL	Interaction Design Lab
IR	Information Retrieval
KNN	K-Nearest-Neighbour
LBA	Location-Based Application
LBS	Location-Based Services
LBSN	Location-Based Social Networks
LR	Location Recommendation
MDA	Mobile Digital Assistant
ML	Machine Learning
OS	Operating Systems
PDA	Personal Digital Assistant

PMI	Potential Memorisation Index
SaaS	Software as a service
WiFi	Wireless Local Area Network

1 INTRODUCTION

Many modern digital services require us to reveal large amounts of personal information in order to use them. Our data are the foundation of those services' algorithms. In this thesis an alternative approach is suggested, moving the sensitive information from the cloud onto the user's device. Instead of sharing a person's data in its full extent, only an abstracted model is shared. Therefore, several layers of client-side modelling and abstraction are applied. This thesis takes the entirety of the underlying system into perspective, while it applies the developed framework to a specific use case in order to demonstrate its feasibility and its applicability.

This first chapter begins with an introduction to the domain of recommender systems, a popular domain for such personal data-driven algorithms. This will serve as the application domain for the developed approach. This chapter concludes with the research question and objectives of this thesis, before outlining the structure of the remaining book.

1.1 MOTIVATION

Why do we share so much personal information with digital services? Or rather: why do services collect so much personal information? We need to share them because many services and their functionality are built on their user's data and heavily rely on that information. A common example are services built on so-called recommender systems. Recommender systems are nothing new. They have been around for over two decades. *Tapestry* was one of the first recommender systems, designed to optimise the management of emails (Goldberg et al., 1992). Since it was developed at the *Xerox Palo Alto Research Center* in 1992 much has changed. The biggest change has been the turn from manual recommendations to fully automated recommendations based on the user's behaviour (Sharma and Singh, 2016). An important driver of this development was the fast diffusion of the world wide web. In order to foster the discussion and innovation, the academic and business community has intensified their research on data collection, behaviour analysis, user modelling, and the actual prediction or recommendation (Silveira et al., 2001; Aggarwal, 2016; Bobadilla et al., 2013).

Most people who are using a smartphone or the world wide web are exposed to such systems on a daily basis. Most common examples are shopping websites like *Amazon*, which analyse the shopping behaviour of their userbase in order to recommend related products of interest to their customers. With the rise of audio- and video-streaming services, which provide their users with access to thousands of songs, TV-shows, and films (e.g. *Net-*

flix, *Amazon*, *Spotify*, *Last.fm*, *Audioscrobbler*, and *Apple's iTunes*), the need for smarter recommendation services grew. *Netflix* did even start a competition back in 2009, granting one million dollars to the team who was able to improve their recommendation algorithm (Netflix, 2009). Even technologies that on the surface appear to be 'objective' information retrieval technologies, like for example most search engines, are optimizing their results to their users' preferences and previous browsing behaviour (Hannak et al., 2013; Simpson, 2012).

The introduction of smartphones brought about the opportunity to collect even more detailed behavioural information, especially spatial information. This trend was also picked up by recommender services. Early examples were working in small refined spatial areas and used such user data to recommend information on conferences (Chen et al., 2003). Today, many applications track their users' spatial behaviour in order to optimise their user experience. LBSs like for example *Foursquare* analyse their users' behaviour. On the one hand, this data is used to recommend locations[1] to their users, and on the other hand, the data is used to provide businesses with insights into their customers' behaviour, a service known as *Foursquare Location Intelligence* (Foursquare, 2017a). In a similar manner, companies like *Uber* are using the spatial behaviour of their users to optimise the experience and efficiency of their transportation services (Belmonte, 2016).

The latest addition to this domain of applications are so-called *intelligent personal assistants* (e.g. *Siri* by *Apple*, *Google Now* and *Allo*, *Hound*, *Microsoft's Cortana*, *Amazon Alexa* and *Samsung's Bixby*). These applications try to gather information about every aspect of a person's life, as they aim to be ubiquitous problem solvers. One issue they are attempting to solve is the question of providing or rather recommending the right information at the right time and within the right context. Therefore, the applications track the user's spatio-temporal behaviour (in addition to many other aspects). Companies have accumulated extensive amounts of data on each user for the purpose of creating highly detailed models of their users in order to be able to create techniques of prediction, forecasting, and recommendation.

In parallel to these technological developments, we have seen a rising discourse on privacy and the regaining of control over our digital selves (Steinebach et al., 2015). Data has become a commodity — a commodity that is being traded and used for profit. Data trading has become a process that happens behind the scenes and usually without the user's knowledge. While some argue that the user's profit is optimised services (e.g. better search results), others argue that the users should be more closely integrated into this process. In many cases, users lose the right over their own data as soon as they register with a company. To empower users the European Union (EU) has established a law that forces companies (located in the EU) to

[1] In this thesis the word *position* refers to a spatial coordinate (x,y); the words *location* and *place* refer to a semantically described entity like a restaurant or a park, which usually has a position or a spatial area defined for providing spatial context.

grant users access to (all) their personal data (upon request) (Ireland, 2003). The German law has an even more strict idea of how personal data should be handled by companies or any other type of organisation, which is specified in two basic principles. The first principle is based on §3 of the Federal Data Protection Act (Bundesdatenschutzgesetz), which holds the data collecting party accountable for collecting only as much information as required for a certain task. These principles are known as *data-avoidance and data-austerity* (Datenvermeidung und Datensparsamkeit) (BMJV, 2009). Furthermore, the German Federal Constitutional Court has ruled that individuals have the right to decide for themselves how to disclose and use their personal data. In this context the concept was called *informational self-determination* (Informationelle Selbstbestimmung) (BVerfG, 1983).

Beyond the issue of rights over our data, personal data has also become a liability. For one thing, leaks and hacks are exposing personal information of users. For another thing, companies are able to use personal information against their users. The company *Uber*, for example, was suspected of identifying policemen and -women through their spatial behaviour and denying them access to their service in order to avoid fines (Isaac, 2017). While people might simply avoid such data-driven services, it has to be acknowledged that many of those services are deeply interwoven into social interactions and business activities. It has to be recognised that many of those tools and services help in daily routines, which makes them so enticing.

This creates a discrepancy between the utility of the service for the user and that user's privacy. On the one hand, the user has a need or wish to use a service, for example, to partake in social interactions that require them to make use of certain digital tools and services. On the other hand, the user may wish to share less information about himself or herself (Barkhuus and Dey, 2003). While those privacy concerns have been acknowledged in academic publications (e.g. Musumba and Nyongesa, 2013; Hong et al., 2009; Barkhuus and Dey, 2003), it has not seen much attention in recent years. Now, this divide has even made it onto the government's agenda. The ruling parties in Germany are discussing whether to change the *data protection act*, specifically the aforementioned principle of *data-austerity*. The reason for this change of mind is the pressure of the digital industry and the fear of falling behind in digital innovations (Krempl, 2016b; Krempl, 2016a; Briegleb, 2016; Borchers, 2016; Krempl, 2015). While we could change the law and allow organisations and companies to gather more information on individuals, the opposing position would be to ask the question of whether **we can make use of smart data-driven services while sharing less data or rather only as much data as is really needed**. On a technological and research basis, this question opens a space of action, which this thesis will further explore.

As smartphones become ever more powerful in regards to their processing power and storage capacity, a concept has been proposed to present a partial solution to the problem. Most current services perform their data storage and processing in the cloud. The company *Set* (Set, 2017) and tools like *Geoparazzi* (Antonello, 2016) or research projects like *PersonisJ* (Gerber et al., 2010) propose to store and process the data on the device instead. This solution builds upon this concept of client-side data storage and processing by creating a data model on the client-side that allows to output datasets with multiple granularities. This option would overcome the challenge of sharing *just enough* data with a remote service by allowing the user to decide upon the level of granularity, which is to be shared with a remote service. This thesis explores this holistic and systematic alternative towards the modus operandi in current digital data-driven services. The previously mentioned LBSs, which makes use of spatio-temporal data was chosen as an application domain. The domain of LBSs or rather (location) recommender systems was not only chosen because this work is located in the research area of Geoinformatics, but also because spatio-temporal data belongs to the most sensitive information a user generates (Gambs et al., 2011). Spatio-temporal behaviour does not only allow a company to calculate the user's likes and interests, but also provides insights into the real-world's behaviour of the user (where he or she lives, works, eats, etc.). Furthermore, spatio-temporal behaviour represents a data-heavy and complex use case for such an alternative approach and is, therefore, a good testbed for the evaluation of the client-side approach.

1.2 THESIS OBJECTIVES

As introduced in the previous section, the focus of this thesis is a holistic perspective onto the client-side modelling of a user's personal (spatio-temporal) data in order to provide multi-granularity data-representations of the user's (spatio-temporal) behaviour. This behaviour can then be shared with a remote service. This dissertation has a Geoinformatics perspective. Therefore, it explores the potentials of Geoinformatics techniques and methods applied to the development of a multi-granularity spatio-temporal client-side data model. Furthermore, this work's approach is centred upon extending existing Geoinformatics techniques with machine learning approaches. First, individual spatio-temporal behaviour is analysed in order to build a user model. This part makes use of techniques from the domain of spatial behaviour analysis, specifically mobility analysis as well as (spatial) network analysis. Second, the dissertation creates a data representation of the model in multiple granularities. The second part builds upon methods of generalisation and clustering. In order to keep the models manageable and efficient, the two phases described above further focus on the use case of the previously described *intelligent personal assistants*. To be more precise, the focus lies on LBSs, which are further distilled to focus on the tasks of LR based on contextual (space and time) information. This focus on personal spatio-temporal data is also driven by the assessment, that such information is especially sensitive, as it is prone to allow for the identification of individuals based on their data. While human spatio-temporal activity encompasses a whole range of categories, this thesis' focus will primarily lie on urban mobility, specifically intra-urban mobility.

To encompass the modelling approach, the overarching objective of this thesis is to create a conceptual framework for the model that guides the selection of methods and techniques. Beyond the theoretical and methodological work, this thesis will also attempt to translate the acquired knowledge into applicable techniques. The following three foci will set the objectives for this thesis and will be the basis for the evaluation of the to be developed concept to follow:

Client-Side:

The most central aspect is developing a privacy-centred alternative to the status quo of user modelling in LBSs. Therefore, this thesis will explore the potentials of moving certain computations that involve sensitive user information onto the user's device.

User Modelling:

The primary component to be moved onto the user's device is the modelling of their spatio-temporal behaviour. Those models should meet the needs of LBSs and in the long run allow for the same user experience that existing server-side modelling approaches deliver.

Multi-Granularity:

As modern digital services heavily rely on data, the client-side user models need to be shared with the remote service. The sharing process should reproduce the user- and privacy-centred approach of the overall thesis. Therefore, this process entails only sharing a model of the user's activity and not the raw data. Moreover, this process ideally allows the user to manipulate the granularity of the shared data.

1.3 discourse & critical perspectives

The objectives above have a strong focus on the conceptual framework and on its technological foundation and algorithmic implementation. Beyond this technology-driven focus, the developed conceptual framework also contributes to the previously discussed discourse on data privacy in modern digital services. Most of the discourse on privacy is simply stating the fact that there is a problem, or it is suggested to ban the services that create those problems through laws and regulations. By contrast, this work plots an alternative solution. The framework proposed in this thesis takes on a holistic view and explores the interplay of requirements and technological solutions. Thereby, the results should help foster the discussion concerning data privacy and data-driven services. This critical perspective is not only applied to the modus operandi, but also to the proposed techniques and methods. It is variably applied then to the discussion of potential biases in data collection as well as to the exploration of possibilities of artificial neural networks for more personal and diverse classifications and predictions. These reflections and critiques should enrich the otherwise technology-driven discussion.

1.4 THESIS STRUCTURE

Chapter 2 contextualises this thesis in its application domain and related works. As a foundation for the subsequent chapters the technology stack is introduced and discussed.

Chapter 3 delivers the theoretical foundation for the thesis and develops the overarching conceptual framework, which will guide the following chapters, by providing a theoretical underpinning as well as goals and requirements.

Chapter 4 incorporates the conceptual model into the requirements of the application domain, specifically focussing on the data exchange between client and server.

Chapter 5 concludes the conceptual work by offering a critical, less technology-driven perspective onto the conceptual data modelling concept.

Chapter 6 focuses on the raw data and its analysis and processing. Therefore, it introduces and analyses the technological infrastructure used to gather the data. This is then followed by investigations of methods and techniques for analysing the spatial data in accordance with the conceptual framework developed in the previous chapter. The chapter concludes with a personal spatio-temporal data model.

Chapter 7 pulls everything together and discusses the implementation of the techniques in a real-world scenario.

Chapter 8 critically reflects upon the development and results of this thesis. After an in-depth discussion of the LBS use case in the previous chapters, the last chapter returns to the holistic view of the proposed approach and connects the insights from the use case onto the overall approach.

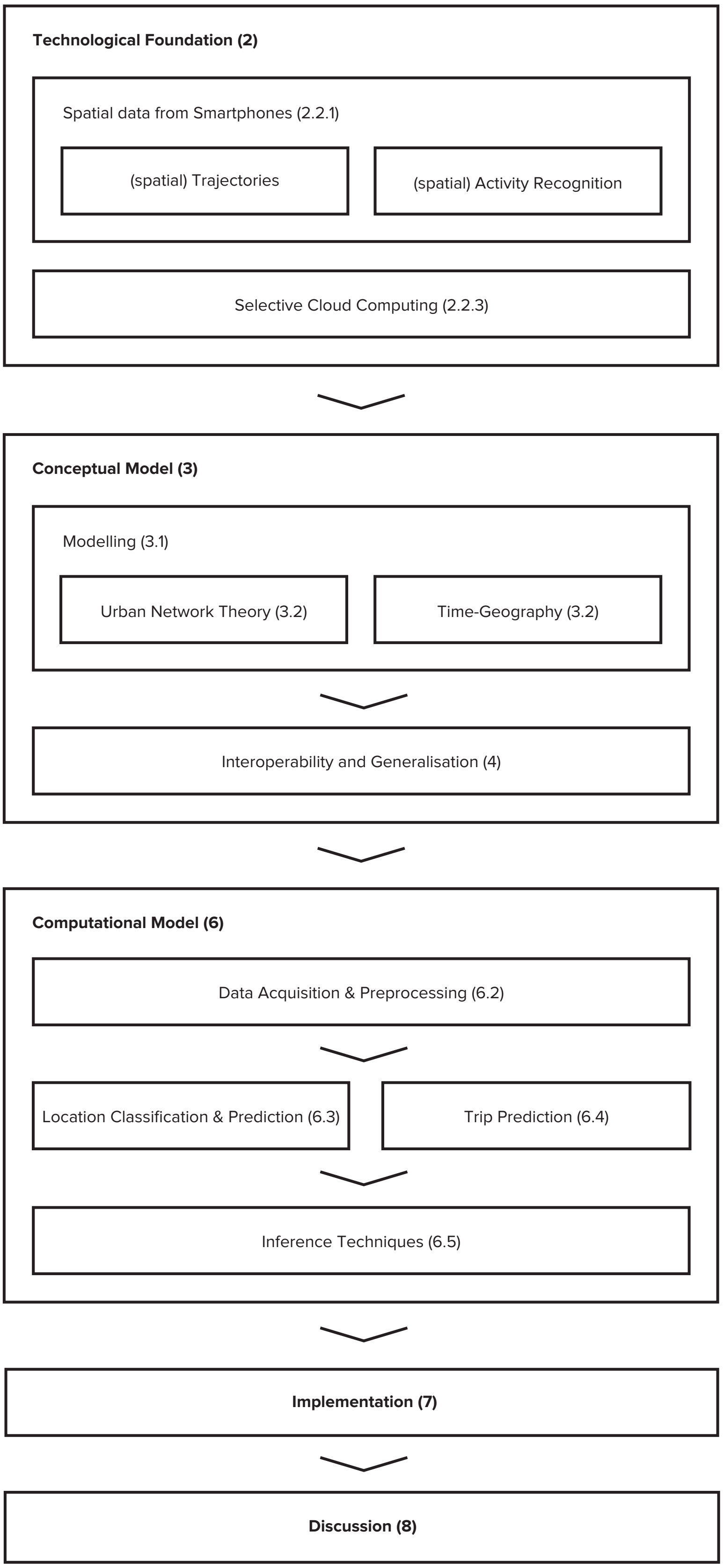

Technological Foundation (2)
Spatial data from Smartphones (2.2.1)
(spatial) Trajectories
(spatial) Activity Recognition
Selective Cloud Computing (2.2.3)
Conceptual Model (3)
Modelling (3.1)
Urban Network Theory (3.2)
Time-Geography (3.2)
Interoperability and Generalisation (4)
Computational Model (6)
Data Acquisition & Preprocessing (6.2)
Location Classification & Prediction (6.3)
Trip Prediction (6.4)
Inference Techniques (6.5)
Implementation (7)
Discussion (8)

Figure 1: Thesis structure.

2 FRAMING & FOUNDATION

2.1 APPLICATION DOMAIN

As introduced in the first chapter, the domain of LBSs and, to be more precise, LR within LBAs on mobile devices, will serve as an application domain for the privacy-preserving approach that will be developed. The focus of this thesis is to develop a broader conceptual model as well as the modelling approach to be used in such applications, not simply the development of such applications themselves. As the following sections will show, the development of models and the algorithms that create them entails precise requirements. The introduction to the application domain, therefore, serves as a construction of those requirements, which will be used in the sections and chapters hereafter. The requirements will provide a more realistic environment for the proof of concept of the client-side modelling approach, especially in regards to the evaluation of the models and algorithms. Therefore, this section will not go into depth in regards to LBSs, but will rather focus on the LR process, specifically the spatio-temporal components. Overviews and reviews of the LBS field, which also informed this thesis, are among others provided by Zontar et al., 2012,Tiwari et al., 2011 and Lee et al., 2006 (more specific overviews mentioned below).

2.1.1 *Location Recommendation*

The first chapter introduced recommender systems more generally. Modern recommender systems in the domain of LR make use of Information Retrieval (IR) techniques which "are generally divided in two types: algorithms that utilize collaborative filtering and algorithms that utilize content based filtering" (Savage et al., 2012, p. 45, see also Baudisch, 1999 and used in applications like Levandoski et al., 2012; Kuo et al., 2009). Content-based filtering analyses the properties of items and calculates the similarity between items. If a user, for example, likes a place that is a certain type of restaurant, the recommendation algorithm recommends similar restaurants. In addition, collaborative filtering includes the analysis of several users and their behaviour to arrive at recommendations. The most common example is the recommendation of locations based on commonly visited locations (e.g. users who visited this location also visited those locations). Even though the actual retrieval task is not the focus of this thesis, the data that is used to build those queries is at the heart of this thesis. Therefore, a further investigation into those data requirements will remain necessary. Across the literature a certain consensus exists, which categorises the data of the above-mentioned filtering approaches into three categories: personal, geographic, and social (Yu and Chen, 2015; Mokbel et al., 2011; Liu, 2014; Zhang and Chow, 2013).

Personal:

Personal filtering refers to approaches where traditional user profiles or rather preferences are used to filter locations, for example demographic filtering (Bobadilla et al., 2013). Those approaches can be combined with the latter two approaches to achieve hybrid filtering approaches, for example comparing the behaviour of users that belong to the same demographic group. Personal profiles or interests are acquired through questionnaires or by deducing them from visited locations, like an interest in a certain category of restaurants that are frequently visited.

Geographic:

Building upon Tobler's first law of geography (Tobler, 1970), geographic filtering approaches group locations based on their geographic relationships (e.g. distance, common region, etc.). Again, the geographic filtering can be combined with other approaches, for example screening locations in a shared neighbourhood which are used by a similar demographic group.

Social:

With the emergence of smartphones and their growing user groups, Location-Based Social Networks (LBSN) became popular and introduced new possibilities of filtering locations. Beyond comparing similarities like the two approaches above, social networks allow a filtering algorithm to analyse the social graph of a user. Analysing the behaviour of friends and connections of a user delivers more insights for new recommendations. An algorithm can, for example, analyse which locations are popular among a network of friends (Li et al., 2008; Seo and Ahn, 2013).

Like most modern LBSs this thesis follows a hybrid approach, combining data from the three areas described above in order to develop a LR algorithm. This will serve to "better the user experience [...] by inferring users' preferences and considering time geography and similarity measurements automatically." (Savage et al., 2012, p. 37). The modelling of spatio-temporal behaviour (geographic) will allow us to develop a personal model, which can highlight the preferences of the user (personal). The model can then in turn be used by the remote service to compare this personal model with other users' models (social). Based on this process, we can define the first set of requirements necessary to perform the IR tasks:

Spatial Behaviour

At the core of all requirements lies the task of the (future) recommendations of locations, built on behavioural predictions and information retrieval. In regards to a user's spatial behaviour, the system needs to acquire information on the *visited locations* (time, position, additional metadata derived from the locations). This will allow the algorithm to calculate patterns and preferences based on visited locations. Furthermore, the resulting location dataset should help identify spatial *areas of interest* (e.g. for finding similar areas). Ye et al. suggest in their work that certain locations or areas show specific temporal patterns (e.g. when people visit certain locations, unique visits vs. repetitive visits, etc.) (Ye et al., 2011a). Aggarwal further illustrates the importance of temporality:

> "1. The rating of an item might evolve with time, as community attitudes evolve and the interests of users change over time. User interests, likes, dislikes, and fashions inevitably evolve with time. 2. The rating of an item might be dependent on the specific time of day, day of week, month, or season. For example, it makes little sense to recommend winter clothing during the summer, or raincoats during the dry season"
>
> (Aggarwal, 2016, p. 21).

Therefore, all aspects mentioned above are not only observed in regards to their spatiality but also in regards to their temporality (Ye et al., 2011b; Yuan et al., 2013).

Personal / Semantic preferences

The focus of this thesis is the spatial aspects of user modelling. Therefore, personal characteristics are derived from the spatial behaviour instead of from traditional approaches that utilise user semantics like demographic information. Beyond the personal spatio-temporal behaviour, described above, one can use the resulting location and areal data from the previous step to acquire additional meta information. Location history, for example, can be used to gain semantic information about those locations and thereby calculate semantic similarities as well as derive patterns of the user's behaviour (Levandoski et al., 2012) (e.g. interest in a certain type of restaurant). In conclusion, the proposed system should be able to acquire personal preferences for further analysis from the spatial and temporal features identified in the spatial behaviour of the user. Based on those preferences and the spatio-temporal information, the system should, for example, be able to "consider activities (i.e., temporal preferences) and different user classes (i.e., Pattern Users, Normal Users, and Travelers) in the recommendation process [...] [thereby] generating more precise and refined recommendations to the users" (Leung et al., 2011, p. 305).

Social (collaborative) Filtering

Collaborative filtering is a powerful IR technique for filtering vast amounts of information and finding user-specific recommendations. In order to do so, the system needs information on a common set of parameters to compare users and find connections. Therefore, we need to share the gathered information from the two previous steps with a remote service, which has access to the aggregate of shared user profiles. As the focus of this thesis is to develop a privacy-preserving approach, the data shared with the remote service should allow the service to run collaborative filtering approaches while maintaining the users' privacy.

Based on the requirements outlined above, we could already build *pull* recommendation approaches. Pull, in the sense that the user sends a query request to the remote service and receives a result, the way most early LBAs worked (Tiwari et al., 2011). A user is, for example, searching for a restaurant and the service is combining filtering approaches to deliver a customised result. In contrast, newer approaches are often *pushing* information depending on the context of the user. Push, in the sense that the system autonomously decides when (and where) to display certain context-relevant information. Those techniques make full use of the mobile capabilities of smartphones and deliver a ubiquitous location and context-aware mobile service. To do so, they need to be able to model the user's context. In the following, the requirements above are extended to allow for context-aware filtering. In order to arrive at those extended requirements, the next section will explore context models and context awareness.

2.1.2 *Context*

Before we can identify further requirements, which will allow us to model and identify certain contextual aspects, we need to explore and define the concept of what 'context' is. Early work on context-aware computing mostly provided very broad definitions of context. One of the first notions of context-awareness within this thesis' application domain goes back to Schilit and Theimer's article from 1994, in which they understand context to be "the location of use, the collection of nearby people and objects, as well as the changes to those objects over time" (Schilit and Theimer, 1994, p. 22) (see also Schilit et al., 1994). Building upon Schilit's definition, the term 'identity' is often used in reference to nearby locations and objects (e.g. Dey, 2001; Zontar et al., 2012) highlighting their unique identification to be used in defining the current context. As Dey has pointed out, "[o]ther definitions have simply provided synonyms for context; for example, referring to context as the environment or situation [...] [which] are extremely difficult to apply in practice" (Dey, 2001, p. 6). Common important aspects highlighted in definitions of context are: *where*, *when* as well as *who and what is nearby* (Schilit and Theimer, 1994; Schilit et al., 1994; Pascoe, 1998; Dey, 2001; Zontar et al., 2012; Musumba and Nyongesa, 2013). Furthermore, environmental parameters extend the previous definition of context (e.g. temperature, humidity, lighting, noise, etc.). In addition to those external parameters the user themselves (e.g. current activity), as well as the user's device and its capabilities are often taken into account as additional parameters e.g. "computing environment, such as available processors, devices accessible for user input and output, network capacity, connectivity, and cost of computing" (Musumba and Nyongesa, 2013, p. 2). While those aspects above define a *whole* context, more application-specific definitions restrict those parameters to just those entities that are *relevant to the current task of the user*, allowing the application for example to provide context-relevant information.

Deducing from those definitions, this thesis defines context with reference to two layers.[2] In the first general layer, context is a point in (geographic) *space and time*, that refers to the where and when. In a second layer, additional parameters are acquired to enrich the context definition, revealing what is in the proximity of the user. Those parameters are of direct relevance to the current task at hand: location recommendation. The parameters refer to the previously outlined LR requirements, the *locations* (and their semantic metadata) in direct proximity of the user, as well as *areal descriptions* (type of area, e.g. the mix of location types), and are thus of primary relevance to the context. In addition, the *spatial relationships* of the current position / area must be considered. More relevant still, the user's current actions are taken into account and future actions are projected. As pointed out in the future works section of this thesis, additional parameters would help with more precise modelling of the context (e.g. weather information) but are excluded from this first approach due to the extent of this thesis.

In order to deal with this contextual definition, this work forwards a four-step process inspired by Chen et al. and their process of identifying activities (Chen and Nugent, 2009):

1. Create a computational model that allows the application to reason about the current context and its changes (to come).

2. Observe the user's behaviour as well as changes in his or her environment.

3. Digest the observed information and process it through the modelling algorithm.

4. Perform actions, deliver information or predictions based on pattern recognition.

[2] Other approaches to deal with the complexity of defining an application's relevant context have been done through numerous systematics, ontologies, and models (e.g. Baldauf et al., 2007; Strang and Linnhoff-Popien, 2004; Musumba and Nyongesa, 2013; Hong et al., 2009; Lee et al., 2006; Bobadilla et al., 2013; Chen et al., 2004).

On the basis of this process we can extend the previous requirements further:

Spatial Behaviour

The spatial context must be aggregated from the location history. In so doing, the system acquires more information about the location's spatial environment (e.g. which locations are near a location that is being visited by the user). In addition, again with a focus on spatio-temporal behaviour, the mobility between locations needs to be taken into account (Randell et al., 1992) Changes in the context need to be constantly observed in order to predict future events. This puts a stronger emphasis on mobility and spatial relationships.

Personal / Semantic Preferences

On the one hand, semantic information can be derived from the location history. On the other hand, this focus needs to shift towards areal descriptions in order to take the user's or a location's spatial context into account. This enables the service to compare contexts based on their common semantics and to derive recommendations.

Social (collaborative) Filtering

In line with the previous requirements, the approach needs to be able to share the resulting model with a remote service for Collaborative Filtering (CF). Again, the premise is to enable CF while maintaining the user's privacy.

In conclusion, the user's spatio-temporal behaviour needs to be analysed with regards to its interactions with locations (visiting a location) and the relationship between locations (mobility). It takes into account the context of the user and his or her locations in regards to space and time in order to create recommendations and predictions based on the user's behaviour. To do this, we need to model the users and their spatio-temporal behaviour as well as their contexts. The following sections will lay the technical foundations for the data that is required in order to create those models, concluding with a section exploring actual modelling from a conceptual perspective.

2.2 TECHNOLOGICAL FOUNDATION

Before discussing the modelling itself, this section will describe the technological foundation upon which the following sections and chapters will build. As described in chapter 1, with the introduction and rapid diffusion of smartphones and their continuous development, they have become of increasing interest for questions of data collection in the context of user modelling. In what follows, two key developments will be detailed further as they are essential to meeting the requirements outlined above. These requirements are fundamental for the development of the methods and techniques in chapter 6 as well as the implementation taken up in chapter 7. Of those two basal technologies, the first is spatial data production on smartphones, which fulfils the requirement of being able to observe the user's trajectories and their location visits. The second, the comparison of client-side computing versus cloud computing, serves the overall agenda of this thesis to create a privacy-preserving approach by moving certain sensitive components from server-side to the client-side.

2.2.1 *Spatial Data from Smartphones*

Today's generation of smartphones was brought about by the introduction of the *iPhone* in 2007. But before what is now referred to as a smartphone, there was another generation of mobile handheld computers. The first of these were introduced as Personal Digital Assistants (PDAs) in the mid-1990s (*Psion*, *Apple*, *IBM*, *Nokia*, *Palm*). During the early 2000s, a whole variety of handheld computers incorporating cellphone capabilities appeared on the market, like *HTC'c* Mobile Digital Assistants (MDAs). Most of the smartphone's early predecessors were not equipped with a GPS sensor and came mostly without any network access features in the early years. Still, the availability of those mobile computers sparked development and research on mobile applications for the display of and interaction with spatial data. One of the first of these devices, the *Simon Communicator* by *IBM* and *BellSouth*, already included a static map (Sager, 2012). Over time miniaturisation and other technological developments led to the integration of GPS sensors and network connections. This resulted in the forerunners of today's LBSs, at the time known as *wearable Geographic Information System (GIS)* (Zipf et al., 2000; Coors and Jasnoch, 1999). While the chips and sensor have become smaller, more efficient, and less power consuming, the fundamentals of acquiring the current location of the user haven't changed much since the early PDAs. Most locating technologies are built on the signal strength of external transmitters and either triangulation or fingerprinting based on the information of the transmitters' locations and time difference (Roxin et al., 2007; Chen, 2012). In the majority of cases, this is achieved through the Global System for Mobile Communication (GSM) network towers or the GPS satellite network (Rao and Minakakis, 2003). For short-range positioning, the same can be derived from WiFi or Bluetooth dongle networks. Modern smartphones combine all those systems in order to always provide a position quickly (e.g. *Android* (Google, 2017)), which is sometimes referred to as Assisted GPS or Synthetic GPS (See figure 2). Those alternative methods to GPS are of special interest if the user is moving through an urban environment (urban canyons), or is located indoors, where due to the building structure the GPS signal becomes weak.

Each of the previously mentioned technologies has its advantages and disadvantages. For our specific use case, two criteria are of further in-

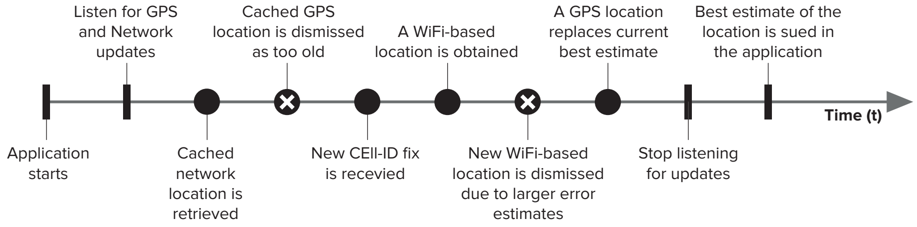

Listen for GPS and Network updates
Cached GPS location is dismissed as too old
A WiFi-based location is obtained
A GPS location replaces current best estimate
Best estimate of the location is sued in the application
Time (t)
Application starts
Cached network location is retrieved
New CEll-ID fix is recevied
New WiFi-based location is dismissed due to larger error estimates
Stop listening for updates

Figure 2: Exemplary assisted GPS workflow. (Adapted from Google, 2017).

terest: *power consumption and accuracy*. In order to model a user's spatio-temporal behaviour we need to constantly collect location information, thus enabling the generation of trajectories for the user's movements. Acquiring constant location information would, for one, create too much data for a mobile device to store. More importantly, the process would rapidly drain the battery of the user's handheld device. To overcome this, the location is only acquired in intervals, with the time between intervals depending on the user's speed and mode of transport. A user walking slowly requires fewer updates than a user sitting in a motorised vehicle. When users operate a navigational application to find a certain location, they use the application for several seconds or even minutes. This allows the application to refine the position and increase accuracy. For constant location tracking, we cannot afford to spend this much time on acquiring the location due to power efficiency. As a result of this middleground solution, inaccuracy and errors are introduced into the dataset. Errors and inaccuracy will vary as a result, depending the kind of network available (Zandbergen, 2009). This problem has to be taken into account in the spatial data preprocessing in chapter 3 when analysing the user's spatial trajectory data.

Activity recognition is a recent development that generates more detailed spatio-temporal information on a user and helps to further refine location acquisition. The enrichment of trajectories through additional user data is sometimes referred to as *semantic trajectories* (Hu et al., 2013). Within this context, activities include for example walking, running, cycling, riding a train, or travelling on an aircraft (Savage et al., 2012; Lau, 2012; Pennanen and Kyrölä, 2013). To recognise those activities, the smartphone analyses data from many different sensors (e.g. barometer, compass, accelerometer, GPS). Machine-learning algorithms interpret and analyse the resulting datasets, for example, by calculating speed and movement patterns and in turn identify the underlying activities. While older research on personal activity data was primarily reliant on GPS trajectories (Spek et al., 2009; Zheng et al., 2008b), this new generation of activity data allows more in-depth analysis of an individual's activity. This data cannot only be used for analyses. It also allows the previously mentioned trajectory generation to fine-tune the intervals at which a location is acquired. The application *Moves*, for example, reduces the interval when a user is riding a train (ProtoGeo, 2016).

Moves was one of the earliest applications deploying this technology while also granting the user access to their data. While Apps like *Moves* are using their own machine learning algorithms, modern smartphone Operating Systems (OSs) (Android 2.2+ (Android, 2017), iOS 7+ (Apple, 2017)) allows developers to access activity classification directly from the system's core Application programmable Interfaces (APIs).

Trajectories and the location history are essential to the requirements outlined above. Therefore, spatial trajectories with activity classifications, acquired from the technologies described in this section, will be the foundation for the data analysis and modelling described in the following sections and chapters.

2.2.2 *Data Refinement through User Feedback*

Creating models of real-world phenomena is difficult, not only because the physical world is complex, but also because data collection of those phenomena remains prone to error. GPS trajectories, for example, are often inaccurate or not accurate enough, which doesn't allow for an exact estimation of where the user is at a certain moment in time. This is particularly unfortunate if the systems try to map the user's position against a dense set of urban locations struggling to determine which location the user currently perambulates through. In order to overcome those uncertainties in the data and improve models and thereby deliver a better user experience, applications need feedback from their users. On a first level, one needs to differentiate between push and pull feedback. Some applications (e.g. *Foursquare*) have designed their user experience in such a way that users are invested in telling the application where they are (push). In the case of *Foursquare* this is achieved through a gamification approach. Pushed feedback on where a user currently is, is often referred to as *CheckIns*. In most other cases the information is explicitly requested from the service (pull). In the domain of LBS and intelligent personal assistants, three distinct pull feedback types can be classified: *improve*, *confirm* and *additional metadata*.

Improve and Confirm:

Through improve- and confirm-feedback functionalities, the system tries to increase its precision. In LBSs the most common feedback is the confirmation of visited locations (see the examples from *Foursquare* and *Google* in figure 3). The service tries to estimate the location with the highest probability and requests feedback from the user if those calculations are correct. If the system cannot arrive at a high probability result, the system requests an improvement from the user. Therefore, the system presents a list of possible locations for the user to choose from (see an example from *Moves* in figure 3).

The same is sometimes done to improve the recommendation algorithm. When the user is confronted with a new recommendation, the system

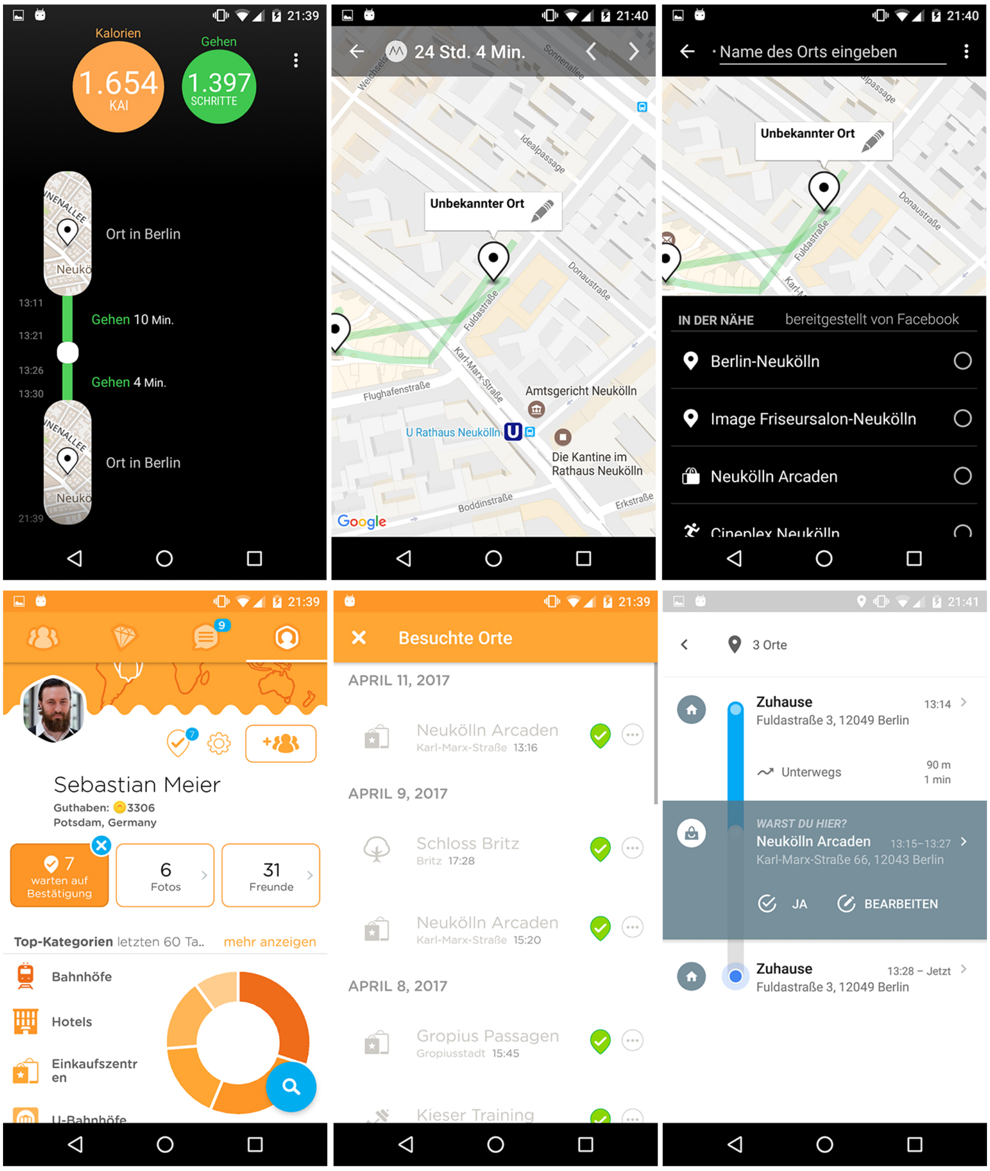
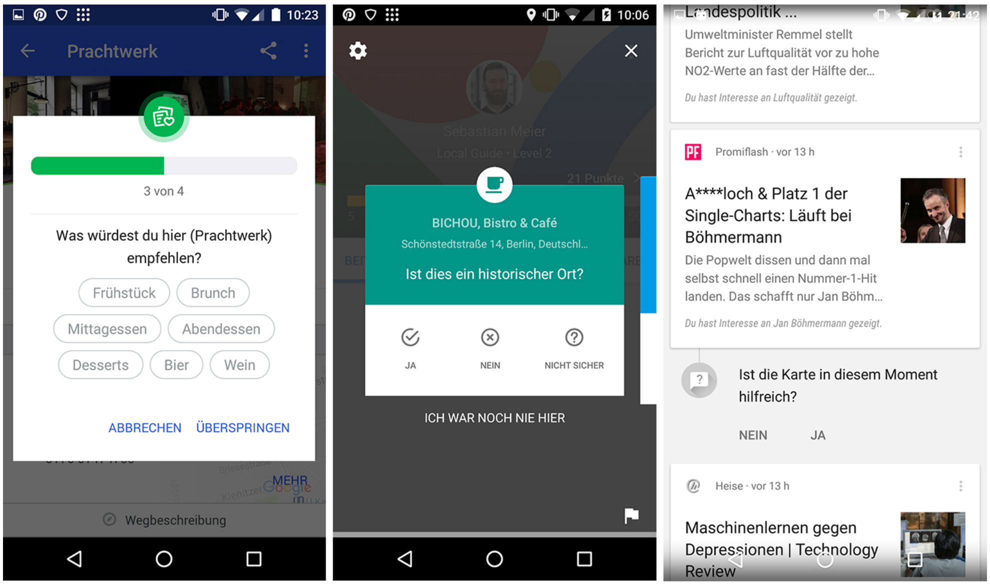


Figure 3: First row: *Moves* Application process of improving location detection. Second row image 1 and 2: *Foursquare* asking the user to confirm or deny the detection of location visits. Last image on the lower right: *Google* location history asking the user to confirm or deny a location visit.



Prachtwerk

3 von 4

Was würdest du hier (Prachtwerk) empfehlen?

BICHOU, Bistro & Café

Schönstedtstraße 14, Berlin, Deutschl...

Ist dies ein historischer Ort?

Landespolitik ...

Umweltminister Remmel stellt Bericht zur Luftqualität vor zu hohe NO2-Werte an fast der Hälfte der...

Du hast Interesse an Luftqualität gezeigt.

Promiflash · vor 13 h

A****loch & Platz 1 der Single-Charts: Läuft bei Böhmermann

Die Popwelt dissen und dann mal selbst schnell einen Nummer-1-Hit landen. Das schafft nur Jan Böhm...

Du hast Interesse an Jan Böhmermann gezeigt.

Ist die Karte in diesem Moment hilfreich?

Heise · vor 13 h

Maschinenlernen gegen Depressionen | Technology Review

Figure 4: From left to right: 1. *Foursquare* collecting additional metadata through their users. 2. The same done by *Google* 3. *Google Now* acquiring user confirmation on content recommendation.

asks the user to confirm if the presented information is helpful or not (see the example by *Google* in figure 4). By collecting such user feedback, the errors and inaccuracies in the datasets can be reduced.

Additional Metadata:

In order to perform IR tasks through collaborative or item-based filtering, a system needs common parameters within which items or users can be compared. Similar to volunteer geographic information systems, LBSs try to improve their location parameters and create a more homogeneous dataset by asking their users to complete the dataset and fill in the gaps (see the example by *Google* and *Foursquare* in figure 4).

Within this thesis' conceptual framework, the user feedback will be used for increasing the accuracy of the location history, as well as for improving the performance of the models.

2.2.3 *Client-side Computing vs. Cloud computing*

Most LBSs are cloud-based services. The first systems that are comparable to our modern cloud computing infrastructures were introduced in the 2000s by research institutes (Sotomayor et al., 2008) and companies like *Microsoft* (Hauger, 2010) and *Amazon* (Amazon, 2006). The main premise of cloud computing applications today is to move the processing of intensive procedures and data from the client's local machine to a network (or, today, rather a internet-connected array of servers, which can perform tasks more efficiently). Tsai et al. have suggested a classification of cloud-based services into four categories, which build upon one another (Tsai et al., 2010). Starting with the foundation, the *data centres (1)* at the bottom, followed by *Infrastructure as a service (2)*, *Platform as a service (3)* and *Software as a service (4)*. As this thesis develops a modelling and data exchange approach that resides between end-user and service, our focus lies on the last layer: *Software as a service (SaaS)*. The fundamental principle of SaaS is the *client-server model*, which goes back to the 1960s (Rulifson, 1969), when work on the ARPANET, the internet's predecessor, started.

In recent years, the term 'cloud' is more strongly associated with data storage than its processing capabilities. Services like *Google Drive*, *Dropbox*, or *Apple's iCloud*, which shaped the term cloud in the public discourse, are only one type of SaaS. This shifted perspective underlines the continuing trend of data-driven services, which resonate with most SaaS. In this sense, cloud-based systems allow their remote users to conveniently access terabytes of data and run algorithms on top. With the diffusion of mobile internet-connected devices, many Apps only serve as interfaces (client) to those cloud-based services (server). While, the mobile device only holds rudimentary functionality to interact with the cloud, the data and the actual services (algorithms) reside on remote servers.

In regards to personal information, the context of this thesis, it is the modus operandi in most SaaS to send all data gathered from the user's device into the cloud. The information is stored on remote servers and can then be used for analysis and calculations. While this thesis is exploring an alternative technique in part, one has to acknowledge that there are advantages concerning a user's personal information in an infrastructure that is completely cloud-based (Kim, 2009):

From a user perspective:

The biggest advantage to users is the relocation of data onto remote servers. Thereby, users can access the service when and where they need to. Multi-device usage can easily be achieved through cloud-based infrastructures, as all data is stored and modified on the server. Therefore it is not required to sync local changes between devices. If a device gets stolen or corrupted, a new device can easily be set up by simply signing into the service and returning to the last state. As most of data storage and processing is handled remotely, the user's devices need to be less powerful, saving cost on end-user hardware. Further costs can be saved through flexible scaling of the required resources on the service, as most cloud service providers only ask their clients to pay for the resources they actually need.

From a business perspective:

Updates to the main service, which do not involve changes in the display of information, can be incorporated without the users needing to update their application (or rather incorporate updates automatically without the user's knowledge). The company running the service has control over all their client's data. Therefore, they can easily analyse the data and for example build new algorithms or additional services (e.g. the company *Foursquare* is using their users' spatio-temporal data to run their business engine *Pilgrim*, which they allow third parties to access (Foursquare, 2017c)).

One of the biggest advantages of cloud-based services is the remote storage of data, which is at the same time a big liability. The concerns regarding personal information were introduced in the introduction (e.g. privacy and data leaks). This does not only account for private individuals, it also accounts for organisations and companies that switch to cloud-based services and need to entrust third-parties with critical data and processes (Marston et al., 2011). Some developers are suggesting a contrasting approach (see below). As smartphones have become more capable, developers have started performing some calculations directly on the device as well as storing certain sensitive information on the device instead of sending everything to a remote server. Such setups are often referred to as *distributed computing setups*. In this specific case, the decision to selectively distribute components based on privacy issues constitutes a new paradigm. Therefore, the term *selective cloud computing* is introduced to describe this divide of processing responsibilities between client and server. From a technical perspective, the concept is in line with distributed computing systems. The proposed concept has a stronger focus on providing a decisionmaking framework for determining which computations are to be made on either the client- or server-side based on privacy requirements. The reason we focus on this alternative is the resemblance of the underlying concept with the paradigm of *data-austerity* as introduced in chapter 1. To further elaborate on the differences between the two concepts, two traditional cloud approaches will be contrasted with two selective cloud computing approaches, each trying to accomplish similar tasks.

The first two applications that serve as an example in this context create spatial trajectories from the user's movements. One example is the cloud approach that serves the App *Nike Training Club* (Nike, 2017), an application developed by the company *Nike*. It allows its users to track their runs and share them on a platform. The App creates spatial trajectories from the smartphones location APIs and stores them locally until the run is completed. As soon as a sufficient network connection is established, the run is being synced with the server. The interesting aspect about the *Nike* App is the fact that it also stores a local copy of the run on the device. While some features of the app require you to sync your runs with the server, like sharing runs on *Nike's* website, most features could be achieved by only storing the runs on the device. The second App is *Geopaparazzi* (Antonello, 2016), an open source solution, which works completely without a cloud counterpart. The trajectory creation works similarly to *Nike's* App. Instead of uploading the

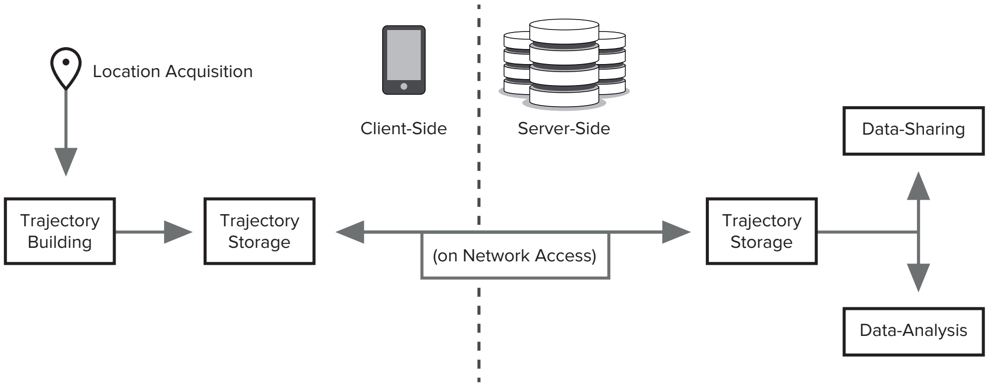


Figure 5: *Nike Training Club*.

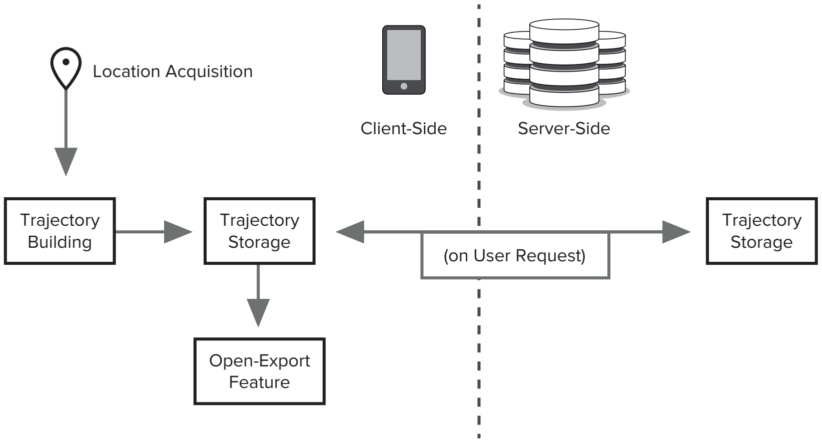


Figure 6: *Geopaparazzi*.

personal data to a server, the users can decide for themselves how they want to use the data. For example, they can share it via Email, upload to a file-sharing service, or simply keep it on the device. A visual comparison of the two approaches is presented in figure 5 and figure 6.

The second set of applications tries to model and thereby predict the behaviour of the user and use that information to build recommendations. On the traditional cloud approach side, we use *Foursquare* (Foursquare, 2017a) as an example. *Foursquare* is best known for its location directory and its LR service. But underneath, *Foursquare* collects and analyses its users' spatial behaviour in order to build better recommendations and to provide location owners with better insights into their customers' habits. To do so, *Foursquare* tracks its users, in particular their movement as well as their so-called *CheckIns* (Users can check into a location when they physically visit it). All this information is sent to the *Foursquare* servers, where it is stored and used for modelling and predicting users' behaviours. In contrast, the company *Set* (Set, 2017) does the same thing but on the user's device. Using *Set's* techniques, developers can collect information on the user directly on the device and, similar to *Foursquare*, generate predictions from the derived

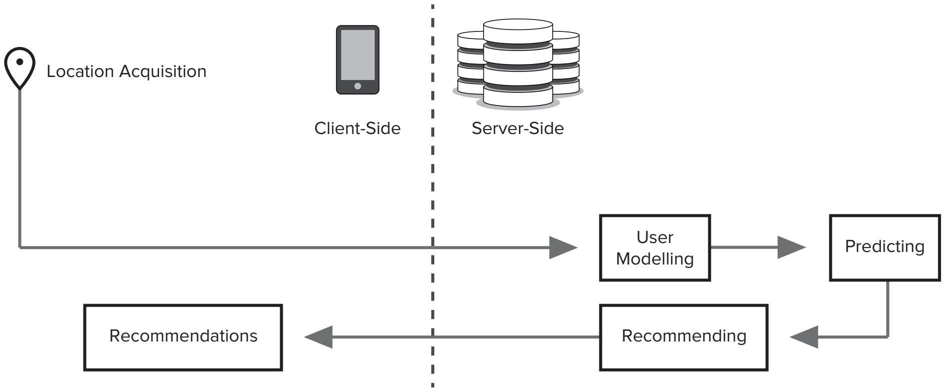


Figure 7: *Foursquare*.

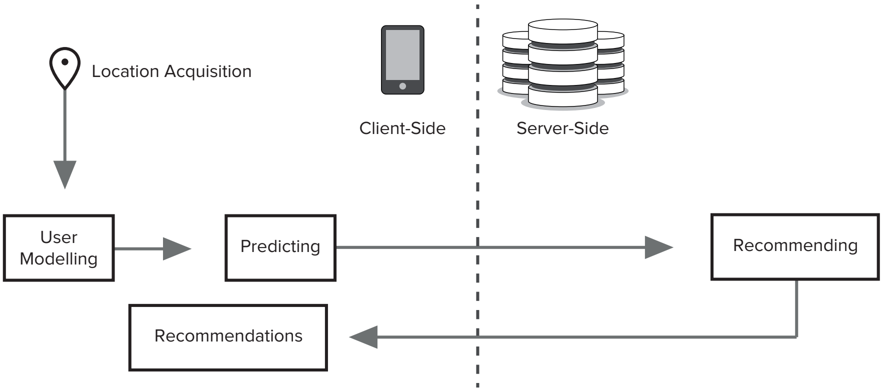


Figure 8: Set.

model, “all while providing better privacy” (Set, 2017). A visual comparison of the two approaches is presented in figure 7 and figure 8.

2.2.4 *Technology Stack*

Building upon the previous two sections, the following one will define the technology stack which is led by the requirements and serves as a foundation for the next section (see figure 9).

The core input data for our data-driven analysis approaches are the spatial trajectories generated through the methods described above. Those trajectories are enhanced through the activity recognition data, allowing us to categorise subsets of trajectories as certain activity classes. The precision of the trajectories will be refined through user feedback. Following the concept of *selective cloud computing*, the trajectories are stored on the user’s device. The trajectories are then used to build a user model which is continuously updated through new trajectory data, again on the user’s device. As mentioned in the first chapter, the overall concept is led by the fundamental idea of the *data-austerity* concept (BMJV, 2009). Our goal is to only share information with the remote service when it is necessary for performing a certain

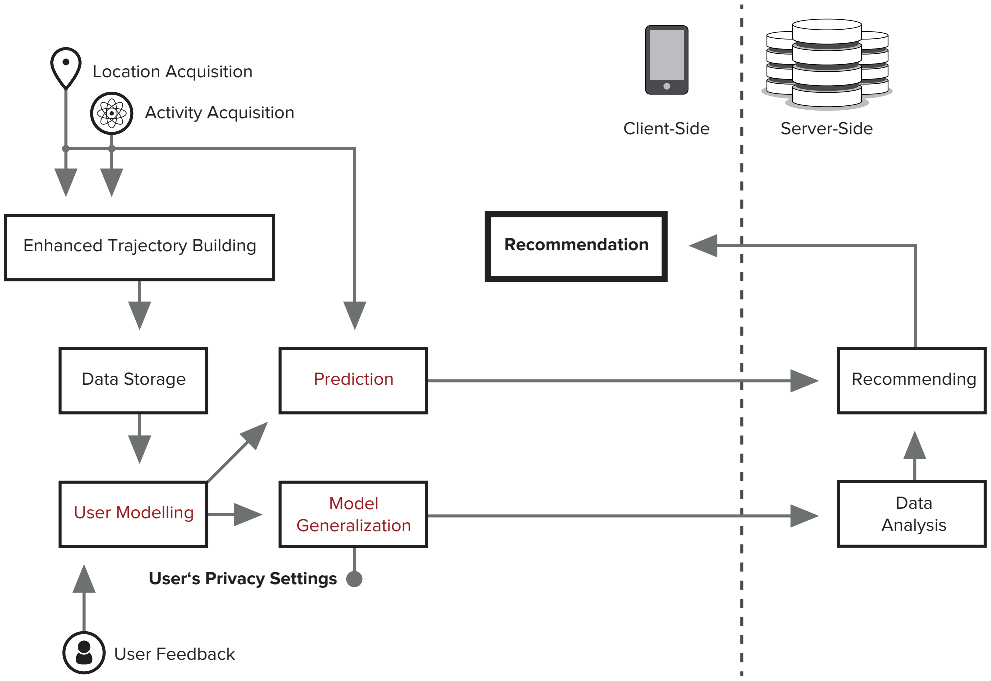


Figure 9: Technology Stack, the foundation of the conceptual Framework. In red the central components to be explored and developed in this thesis.

task, all other information on the user's device should be kept private. To this end, this thesis explores two conceptual and technical features. Firstly, the trained model on the user's device should be used to identify spatio-temporal patterns instead of using a remote service. Those patterns can be used to trigger the need for interactions with the remote service (e.g. the user's need for a location recommendation). Secondly, instead of sending the whole of the user's data to that remote service, only a generalised model is transferred. This leads to the challenge of allowing the user to control the granularity of the model that is shared with the remote service.

While this concept consists of numerous parts, as figure 9 indicates, this thesis will focus on user modelling — the prediction as well as the generalisation — in order to illustrate the overall feasibility and applicability of such an approach. To serve this purpose, the following section will use the technological foundation as well as the requirements of the previous sections to construct the conceptual modelling framework, which will guide the consecutive chapters.

2.2.5 *Related Work*

The previous section already introduced two applications that make use of approaches similar to the selective cloud computing approach. Before the modelling is introduced, the following paragraphs will further position the work in this thesis within the context of relevant research projects and publications.

When discussing privacy-preserving techniques in data-driven services, a particularly critical research field is that of anonymisation. Anonymisation, or more precisely data anonymisation, is a broad research field which takes up methods and techniques that either encrypt or modify information in order to render the persons or entities generating the data anonymous. The selective cloud computing approach was chosen over anonymisation for two reasons. 1) The main component of anonymisation processes is located on the server side, which requires the users to disclose their information in the first place. It requires them to put all their trust in the remote service to properly anonymise their data once it arrives. 2) A large body of research in the domain of anonymisation investigates concepts for de-anonymising data. A good overview of such techniques for spatial trajectories is found in the paper by Gambs et al. (Gambs et al., 2014), in which they perform a test that allows them to successfully identify 45% of users in a large-scale real dataset. This leads them to conclude that "the mobility behavior of an individual is far from being random [...] and tends to be unique thus acting as a signature of an individual" (Gambs et al., 2014, p. 19). This thesis strongly supports this statement. Otherwise the techniques for automated user-specific classification and prediction would not work. Therefore, this thesis acknowledges the importance of anonymisation while pointing out that it does not solve the problem of protecting the user's privacy. Accordingly, this thesis explores the concept of sharing only certain aspects of information with a remote service and moving the user modelling to the client side.

Similar approaches have for example been explored by Ceri et al., 2004; Paireekreng and Wong, 2009; Gerber et al., 2010, whose publications also inspired and influenced this work's approach. Overall, one can say that with the emergence of cloud-based services a strong focus on service-side modelling was established. Therefore, the literature on recent client-side modelling is not very rich. Gerber et al. point out that this server-side focus leads to a "lack of framework[s]" (Gerber et al., 2010, p. 3), which makes the development of client-side modelling approaches difficult.

While the main motivation for this thesis is to explore a privacy-preserving approach, the aforementioned projects also see a big potential for client-side modelling beyond privacy. For example, the

> "[c]lient-side solutions can reveal as being more dynamic, more adaptive, and protective for sensitive user data. They may be very effective for remembering the local context or being aware of the local peculiarities of the interaction. Also, a clear separation of concerns between the client and the server may lead to interesting business opportunities and models."
>
> (Ceri et al., 2004, p. 1).

But also here privacy remains a key concern. As Gerber et al. highlight:

> "there is a tension between such personalisation and privacy; the user model that drives personalisation is based upon the user's personal information. Moreover, there is evidence of considerable community concern about the proper protection of such information."
>
> (Gerber et al., 2010, p. 1).

The three publications have chosen similar paths to establish their client-side modelling approaches, upon which this thesis builds.

The oldest of the three projects by Ceri et al. (Ceri et al., 2004) builds upon client-side generated UML-Guides and server-side generated WebML content. In the example application, the team uses the approach in an e-learning tool. The application mainly modifies the interfaces; more precisely the system calculates more relevant content using the client-side model and highlights the content accordingly. Even though the exact process is not transparent, the team discusses the possibility of allowing the user to decide which information should be shared with the remote service. While the approach of keeping personal sensitive information on the client side and creating an adaptive frontend is interesting, it creates a significant obstacle in modern big data applications. While displaying a list of items in an e-learning environment is a manageable feat, the transfer of a location dataset of thousands if not millions of locations before deciding which to display is simply unfeasible. Therefore, the data sharing discussed by the authors needs further attention in order to customise the information query process and only transfer relevant information from the server-side.

The second project by Paireekreng et al. (Paireekreng and Wong, 2009), similar to the first project, also focuses on content personalisation. They take a more dynamic approach to creating the user model compared to the the static UML approach and instead discuss modern machine learning. But Paireekreng et al.'s choice for the clustering of user data ends up being a *K-Means* clustering, which is not very advanced in terms of machine learning. An important criterion for their choice at the time was the feasibility of implementing the technique on a mobile device. In the eight years since the publication of this paper, this aspect has clearly shifted. The approach by which *K-Means* and the classification is implemented uses unsupervised training in order to achieve a more dynamic user modelling. In regards to the communication of the client- and server-side, the specifications are more detailed than in the previous project by Ceri et al. (Ceri et al., 2004). As discussed in the context of the previous project, Paireekreng et al. (Paireekreng and Wong, 2009) suggest that the user profile and related data would be submitted to content providers for the next step of processing, in order to obtain personalised content (Paireekreng and Wong, 2009, p. 98). Still many aspects and details are left out.

The project *PersonisJ* builds on a similar premise but attempts to deliver a more sophisticated implementation (Gerber et al., 2010). *PersonisJ* is based upon *PersonisAD*, a framework for distributed context and user modelling

(Assad et al., 2007). On the one hand, the aspect of distribution presents an additional level of privacy assurance. *PersonisJ* is a third-party application installed on a mobile device, it is building and handling the user model. Other applications can then be granted access to *PersonisJ* and register to receive events. Thereby, the user model is secure in *PersonisJ* and acts as a gatekeeper to the user's information. On the other hand, this concept presents a big obstacle in regards to a real-world implementation. As the authors of *PersonisJ* indicate, a two-way communication is only possible if both *PersonisJ* and the other application are built under the same certificate. The author of this thesis tested this approach across android and iOS, ascertaining that communication between two applications is difficult if not impossible. Therefore, even though this concept is very interesting, it was not further explored in this thesis.

The overall concept in this thesis has been inspired and influenced by the related work described above. In contrast to the existing work, this thesis explores more dynamic approaches to user modelling. Beyond concepts, this thesis presents algorithmic implementations of those concepts in order to test and validate them through real-world datasets. As a foundation for those algorithms, the next chapter introduces and discusses the modelling, before becoming more precise in subsequent sections and chapters.

3 CONCEPTUAL MODEL

The previous chapter framed this thesis' focus by discussing related work and introducing the exemplary application domain. Furthermore, the technology stack was put forth to serve as the foundation for further investigations. Before the actual algorithms are explored, the following chapter constructs a theoretical foundation, beginning with the discussion of the modelling process, in order to guide the subsequent chapters.

3.1 MODELLING

> "[...] *In that Empire, the Art of Cartography attained such Perfection that the map of a single Province occupied the entirety of a City, and the map of the Empire, the entirety of a Province. In time, those Unconscionable Maps no longer satisfied, and the Cartographers Guilds struck a Map of the Empire whose size was that of the Empire, and which coincided point for point with it. The following Generations, who were not so fond of the Study of Cartography as their Forebears had been, saw that that vast Map was Useless, and not without some Pitilessness was it, that they delivered it up to the Inclemencies of Sun and Winters. In the Deserts of the West, still today, there are Tattered Ruins of that Map, inhabited by Animals and Beggars; in all the Land there is no other Relic of the Disciplines of Geography."*
>
> Borges, 1946, translated by Hurley, Andrew

> *" 'That's another thing we,ve learned from your Nation,' said Mein Herr, 'map-making. But we,ve carried it much further than you. What do you consider the largest map that would be really useful?'*
>
> *'About six inches to the mile.'*
>
> *'Only six inches!' exclaimed Mein Herr. 'We very soon got to six yards to the mile. Then we tried a hundred yards to the mile. And then came the grandest idea of all! We actually made a map of the country, on the scale of a mile to the mile!'*
>
> *'Have you used it much?' I enquired.*
>
> *'It has never been spread out, yet,' said Mein Herr: 'the farmers objected: they said it would cover the whole country, and shut out the sunlight! So we now use the country itself, as its own map, and I assure you it does nearly as well.' "*
>
> Carroll, 1993, p. 393

The two examples above illustrate a paradox which is a recurring theme in literature (Borges, 1946; Eco, 1963; Ende, 1973; Carroll, 1993). It describes the endeavour of cartographers to create a representation of physical space that is just as detailed and precise as the real world. In the most extreme case, in Michael Ende's 'Momo', the attempt of creating a life-size model of planet earth consumes the actual planet earth until only the model itself survives.

Although fictitious, the illustrations from the short stories and novels mentioned above help us to understand an essential principle of maps[3]: *abstraction* and *generalisation*. The process of abstraction is developed as an effect of the purpose of the map (e.g. hiking, car navigation, geological surveys, etc.). The purpose helps the cartographer focus on certain elements and defines the map's level of abstraction and generalisation. The map, in this instance, is a visual representation of a (conceptual) model of the physical world (see for example the cartographic communication model by Heidmann, 2013 or MacEachren, 2004). In the same way that a cartographer creates a conceptual model of the physical world in order to create a map, this thesis explores the modelling of an individual's spatio-temporal behaviour. It does this through abstraction into a (conceptual and computational) model, in order to provide insights to a recommendation algorithm.

As the literary examples figuratively illustrate, creating a model that is as exact and detailed as the real-world phenomenon is not only nearly impossible but also not helpful since "[...] [m]odels are approximations of the real world" (Banks, 2010, p. 1). By conceptualising a real-world phenomenon into a model and making it more abstract, we make it manageable. Through the abstraction of the real world and the conceptualisation into systematics, processes, etc., we gain the ability to utilise those models in order to, for example, communicate, discuss, simulate, compare, or compute aspects of the physical world (Banks, 2010). Depending on the task, models are often classified by their purpose in a development or research process these encompass the conceptual, communicative, programmed and experimental model (Banks, 2012; Banks, 2010; Balci, 1997). The latter two are at times also referred to as the executable model. This thesis specifically focuses on the conceptual model as the theoretical foundation and the programmed and experimental model as a means to validate the conceptual model and its implementation (see figure 10).

A model is built upon knowledge about the real-world phenomenon that is to be modelled. When it comes to spatio-temporal behaviour, the majority of current research focuses on the identification of patterns by using big datasets aggregated from multiple users. Through the analysis of vast sets of data across individuals, this big data approach aims to increase significance and find common behavioural patterns (e.g. identifying areas and their usage (Liu et al., 2014; Rösler and Liebig, 2013) or similar communities across city boundaries from LBSNs (Noulas et al., 2011a; Hannigan et al., 2013)).

By contrast, this thesis only works on one individual's data to serve the goal of user modelling on the client's device (selective cloud computing). As this data is quite large in and of itself due to the sensors and other input channels, it will furthermore be called *Personal Big Data*. As defined in the technology stack, the main input parameters are spatial positions and tra-

[3] While the term 'map' defines a variety of concepts and artefacts, in this case, it is used as MacEachran defined his *prototypic* map, on his abstractness continuum (MacEachren, 2004, p. 161).

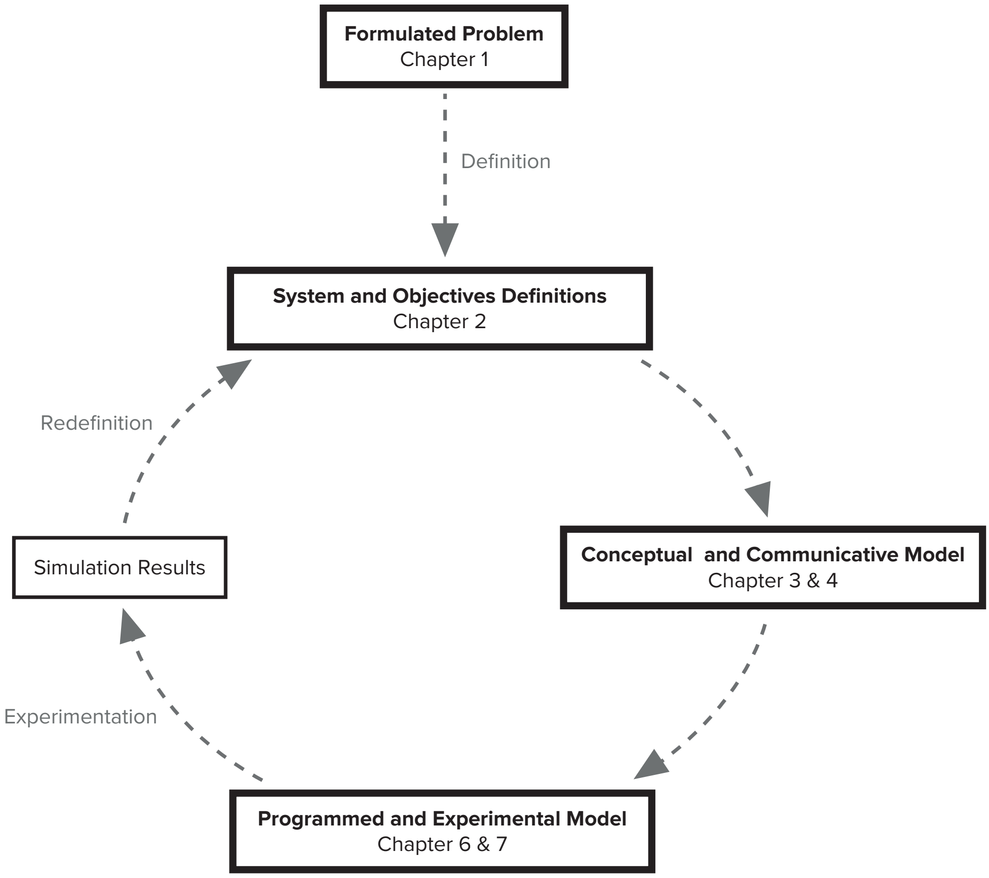


Figure 10: Modelling process, adapted from (Banks, 2010, p. 9).

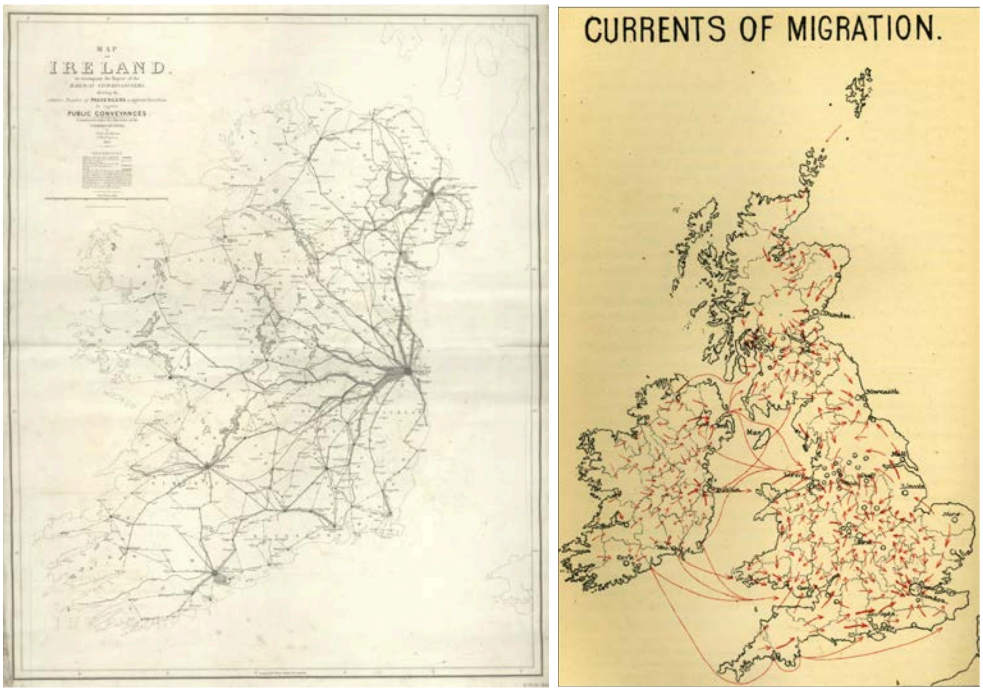


Figure 11: Left: Public Conveyances on the Irish rail network (Griffith et al., 1838), Right: Migration patterns in the UK (Ravenstein, 1885).

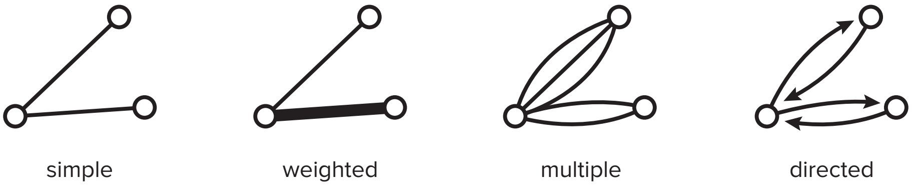


Figure 12: Basic network structures.

jectories enriched through activity classification data. This thesis relies on the concept of network theory as a fundamental concept for the modelling approach built on those datasets. This theory will thus further be explored in the next section.

3.2 URBAN NETWORK THEORY

> "[...] *Many different figures are exploring both the networking of space and the spatiality of the network, identifying a series of key conditions: the everyday superimposition of real and virtual space, the development of a mobile sense of place, the emergence of popular virtual worlds, the rise of the network as a socio-spatial model, and the growing use of mapping and tracking technologies. These changes are not simply produced by technology. On the contrary, the development and practices of technology (as well as the conceptual shifts that these new technological practices produce) are thoroughly imbricated in culture, society, and politics.*"
>
> Varnelis and Friedberg, 2008, p. 15

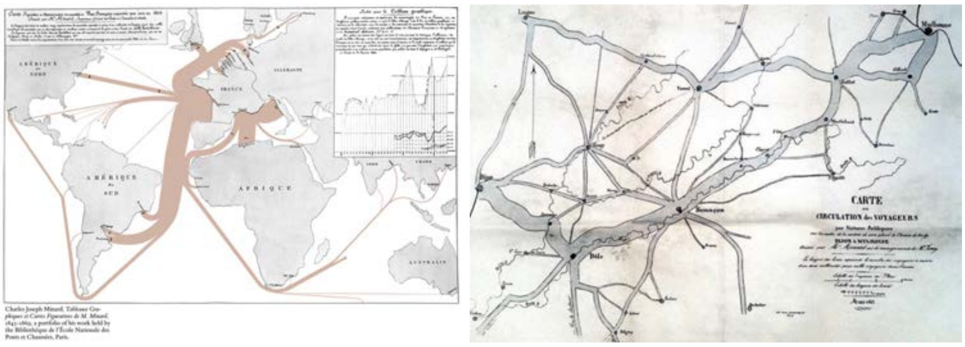


Figure 13: Left: French wine exports (Minard, 1865), Right: Traveller network between Dijon and Mulhouse (Minard, 1845).

Network theory is strongly connected to graph theory. Mathematical graphs that consist of vertices and edges[4] (see figure 12). The vertices represent entities (e.g. a location, a person, a neuron, or a virtual or theoretical entity), while the edges represent connections and relationships. Edges can carry additional information (e.g. a weight or a classification). In certain networks, two vertices can be connected through multiple edges (e.g. a two-way origin-destination dataset or edges of different classifications, see figure 12). Networks as models for (interconnected) real-world phenomena are nothing new and have become increasingly popular across academic disciplines. Explorations of networks have variously been seen from Latour's sociological actor-network theory (Latour, 2005; Latour, 2011) to neuroscience's network models of the human brain (Papo et al., 2014), just to name a few. As the quote above introduced, network theory is also frequently used to model spatial data (e.g. to model spatial or semantic relationships). One example close to this thesis' focus is the domain of transport and mobility. Its usage of network models dates back to the mid 19th century. Such uses for example spanned from a 1838 report by the *Rail Way Commissioners Ireland*, which contained a graphic of the rail network with a quantitative visualisation of the passenger flow, to a publication on migrations containing a flow map of migrant movement in the United Kingdom by Ravenstein in 1885 (see figure 11). In a similar way, Minard used network visualisations in many of his works (e.g. french wine export or travellers between Mulhouse and Dijon, see figure 13). Minard even incorporated the temporality of networks in his comparison of cotton and wool trade between 1858 and 1861 (see figure 14).

Since the 19th century, the process of developing network models has become more data-driven and formalised. Building upon those early examples from the domain of transport and mobility, cities have become a new focus for applying network theory. An early work was published by Christopher Alexander in 1965 in which he critically compared the structures of cities to networks (Alexander, 1965). In his work, he constructed the networks manu-

[4] Vertices are also often referred to as nodes, as well as edges are referred to as links.

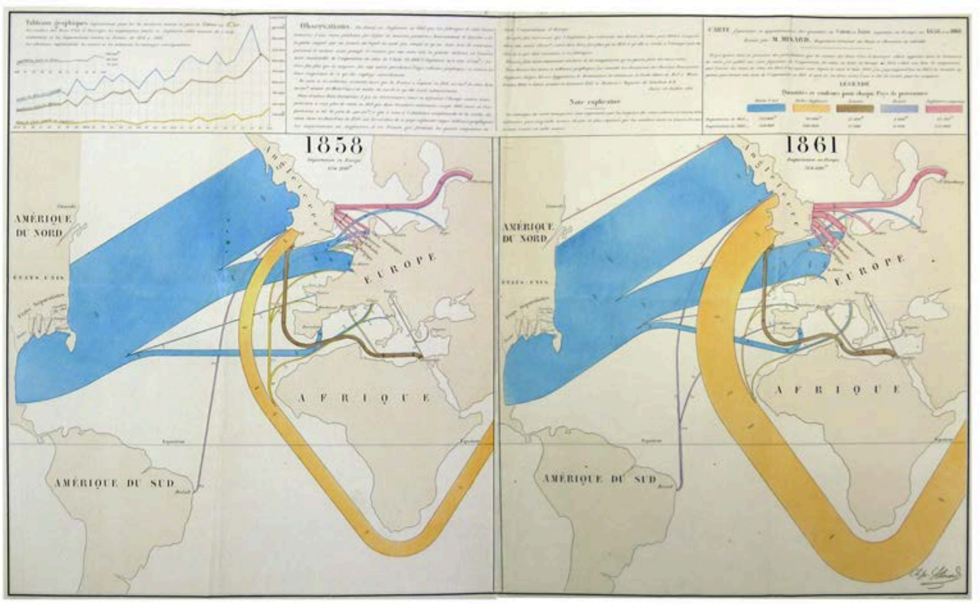


Figure 14: Cotton and wool imports to Europe (left 1858, right 1861) (Minard, 1862).

ally. 48 years later, Michael Batty published his book 'The new Science for Cities', which built upon Alexander's ideas (and many other thinkers since) but made such ideas more applicable to 21st-century contexts. Batty highlights the many layers of cities, that can be modelled as networks: from the physical road network to the social connections within a city (Batty, 2013). In addition to networks, he extends the concept with flows, describing the entities within the network. More importantly, he goes beyond theoretical approaches and transfers the concepts into computational models, allowing us to perform more complex analysis and simulations. Most of Batty's networks are subsumed within what is referred to as *spatial networks*.

Spatial networks are networks in which every vertex has a position in an n-dimensional coordinate system. Networks, in general, are not necessarily spatial; in many networks the spatiality is created through computational processes, which translate abstract data properties into a (mostly two-dimensional) spatial representation in which the inherent connections are rendered as the edges of the network. Nonetheless, all of the aforementioned examples, from the domain of mobility, transportation, and urbanity, are spatial networks in which the spatial coordinate system equals the geographic coordinate system. This means that on top of the general network analysis we can perform spatial analysis on the network.

The same accounts for the data generated through the technical processes outlined in chapter 2, which results in a set of locations (vertices) and trajectories between locations (edges).[5] Therefore, the topology of this thesis' data-

[5] An alternative to the approach outlined above could be to include the road network as a network with a higher granularity than our location and trajectory network - an approach

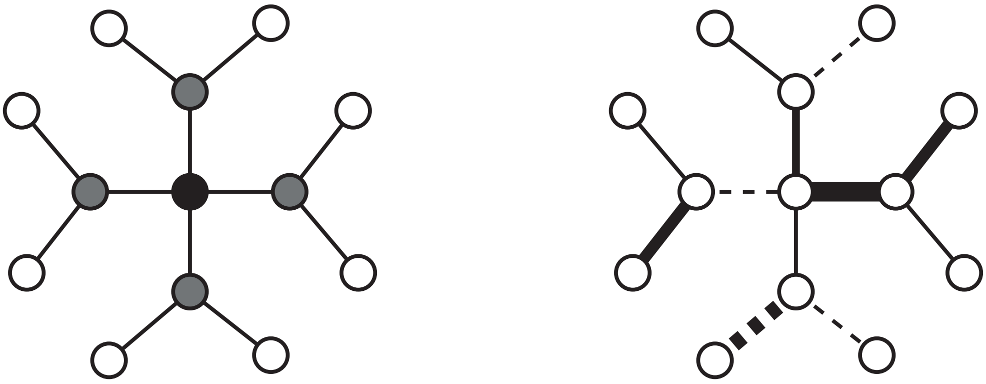


Figure 15: Left: Degree of centrality (darker more central), Right: Link Analysis.

set resembles a (spatial) network and has informed the decision to choose a network approach as a basis of modelling the spatio-temporal behaviour in this thesis. On the one hand, the role of each location within the network is of importance. In network analysis, this is known as *centrality*, which determines the relative importance of a vertex within the network. On the other hand, the relationships between locations must be analysed. In network analysis, this can be achieved through *link analysis* (see figure 15).

Most networks and their analysis represent one point in time. Therefore, integrating temporal aspects into a network presents a challenge. As already emphasised, temporality is important for the approach. As a first step to incorporate temporality, the vertices can be modelled for certain points in time (e.g. time of day, the day of the week, season, etc.), allowing the system to compare states and discover differences. These aspects can then be incorporated into recommendations and predictions. Realtime movement of the user within the network is more complex, therefore, we have to go beyond traditional network approaches. In geography, a common concept for dealing with such data is *time geography*. This concept was largely conceptualised in the 1960s and 70s by the social geographer Torsten Hägerstrand (Hägerstrand, 1970) and investigates social dynamics in spatio-temporal contexts. Spatio-temporal constraints are an essential component of time geography:

1. "'Authority constraints' are those which limit the activities of the individual because of [...] [their] biological construction and/or the tools [...] [they] can command"

that is commonly used in techniques of space syntax, a framework for the analysis of urban spaces using network theory. In defence of the higher abstracted network, it is argued that in order to create a functional model, one should only include relevant properties in a model and therefore abstract the model as much as possible. This is especially the case if that model needs to be transferred to a computational model which is grounded on parametrisation and strict rules. As the requirements of the approach developed within this thesis do not call for exact routeing or details of higher granularity, the abstracted network of locations and their trajectory connections fulfil those requirements.

(Hägerstrand, 1970, p. 12).

2. "[Coupling constraints] define where, when, and for how long the individual has to join other individuals, tools, and materials in order to produce, consume, and transact."

   (Hägerstrand, 1970, p. 14).

3. "[Coupling constraints describe] a time-space entity within which things and events are under the control of a given individual."

   (Hägerstrand, 1970, p. 16).

The first constraint describes the mobility capabilities of an individual. By having access to the past trajectories of a user, a model could allow the computation of such user-specific constraints. The second and third constraints describe (with two different perspectives) the duration of a spatial event, which - similar to the first - could be predicted through the historic user data.

This thesis forwards a novel approach for combining Hägerstrand's model with the previously introduced spatial network model. Hägerstrand accompanied his conceptual model with an abstract visualisation technique for describing a person's spatio-temporal trajectory and the influence of the above-described constraints. Hägerstrand used a three-dimensional space, in which x and y represent latitude and longitude and z represents time. This type of visualisation is today best know as the *space time cube* (see e.g. Bach et al., 2014; Kraak, 2003; Gatalsky et al., 2004; Kraak and Koussoulakou, 2005; Kraak, 2008). If a person is in a location *L* for a timespan *T*, their action space[6] is described as a circle. The radius of the circle is defined by the space that can be explored in time *T*, which is influenced by the spatial constraints (e.g. mode of transport) (see figure 16, top left). In a further abstraction, Hägerstrand reduces the dimensions to time and space. In doing this, he generates space-time prisms (see figure 16). In order to apply this to a sequence of spatio-temporal events, Hägerstrand mainly describes two types of events: 1) spending time *T* at location *L* and 2) travelling between a location *L1* and *L2*. In the first case, the action space increases until $\frac{T}{2}$, after which the space decreases. Therefore, the person needs to return to location *L* (see figure 16, top right). In a similar matter, the radius in case 2 is defined by the surplus of time during the trip between *L1* and *L2*, increasing the spatial reach beyond *L2*, before returning to *L2* (see figure 16, bottom).

Hägerstrand's prisms are abstract representations that help illustrate his concept. In what follows, these representations will demonstrate how this thesis combines time-geography with the network model described in the

[6] Hägerstrand also describes the prisms a person moves within as 'islands'. For this thesis the term *action space* was chosen as it puts more emphasis on the possibilities it offers in terms of movement and decisionmaking to the individual (for a definition of the term see Mayhew, 2009)

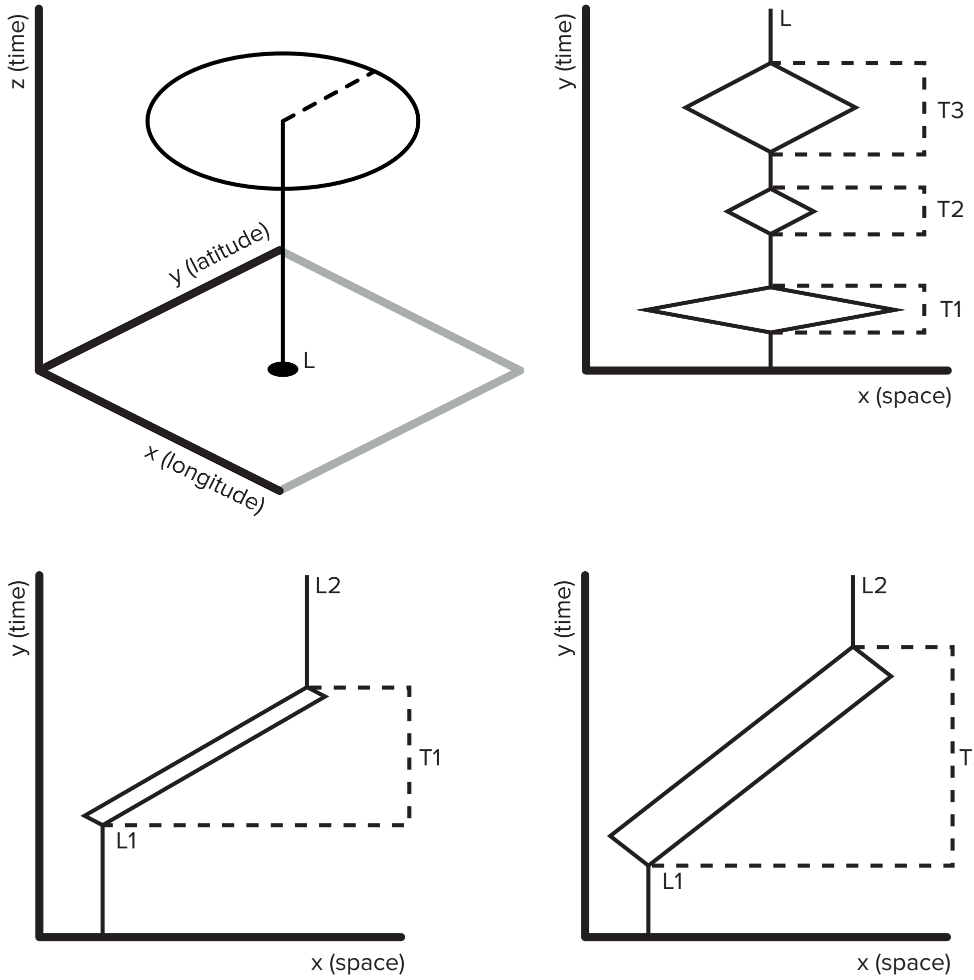


Figure 16: From top left to bottom right: 1) *space time cube* with location L and a radius for the individual constraint limited action space 2) one location L with three prisms with varying timespan (*T1*, *T2*, *T3*) and varying constraints 3) a trip between two locations (*L1* and *L2*) 4) the same trip with a bigger timespan and therefore bigger spatial prism.

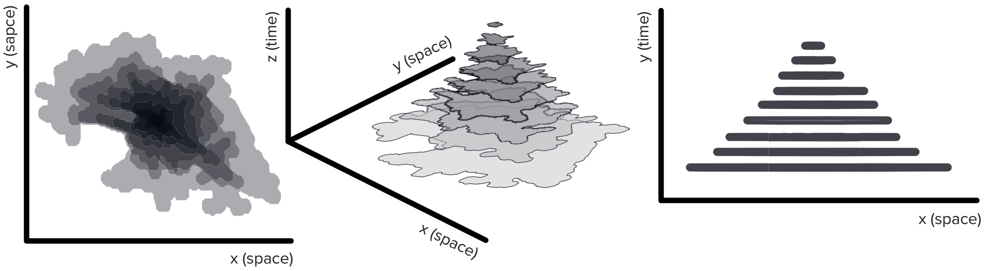


Figure 17: Transformation of isochrones from a purely spatial representation to a spatio-temporal representation.

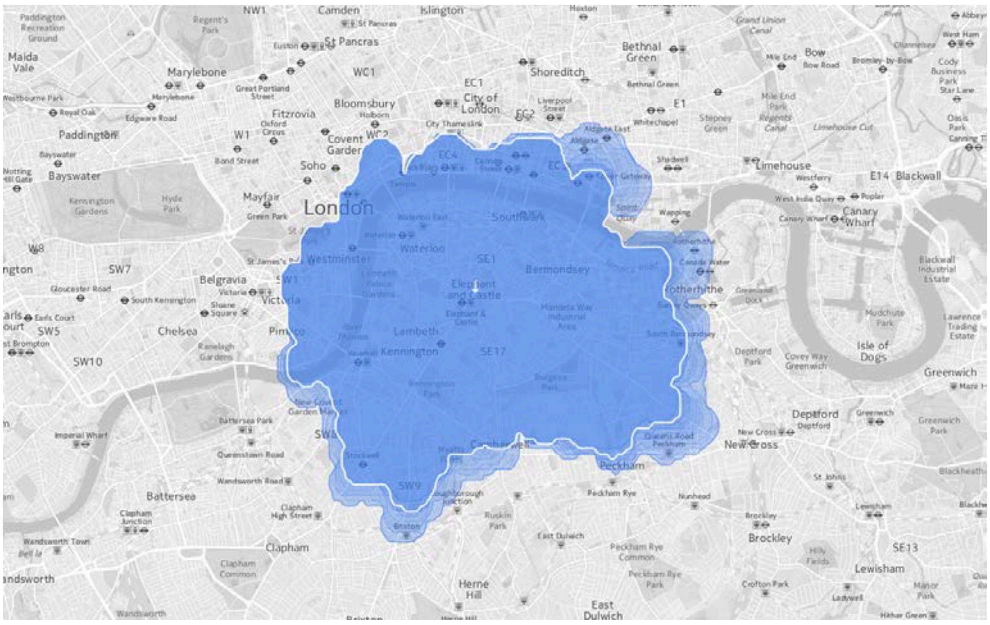


Figure 18: Example isochrones for the city of London, taken from the Isoscope prototype developed by Gortana et al. (Gortana et al., 2014).

previous section. While Hägerstrand is dealing with trips, similar to the edges in the network, his trips are not based on actual geographic (street) networks. As a means of bridging this gap, this thesis introduces a novel approach for building Hägerstrand's prisms from isochrones. Isochrones are an established technique for visualising the action space of an individual depending on their constraints (see e.g. (Street, 2006)). Each line in an isochrone map connects points at which an event happens at the same time (e.g. the arrival of a person who starts from a common origin point). If applied to mobility, one can sample isochrones for varying timesteps, which will result in a dataset describing Hägerstrand's prisms or at least one half of the prism (see figure 17).

Isochrone mobility maps are usually built on the basis of road network data, making use of routing algorithms that use historic traffic data in order to calculate the isochrones (see e.g. figure 18 by Gortana et al., 2014). The same approach can be applied to the personal trajectory network. The

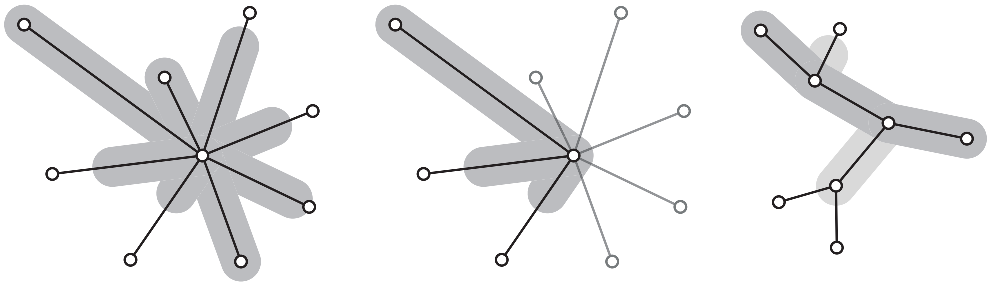


Figure 19: Applying the isochrone concept onto the abstracted trajectory network (left to right): 1) isochrone based on time, and historic travel information; 2) taking into account the current mode of transport, one can apply filters based on mode of transport derived from historic trip information, and reduce the network to matching trips; 3) in a similar manner the probability of a route being taken under current conditions (e.g. day of the week, hour of the day) can be added.

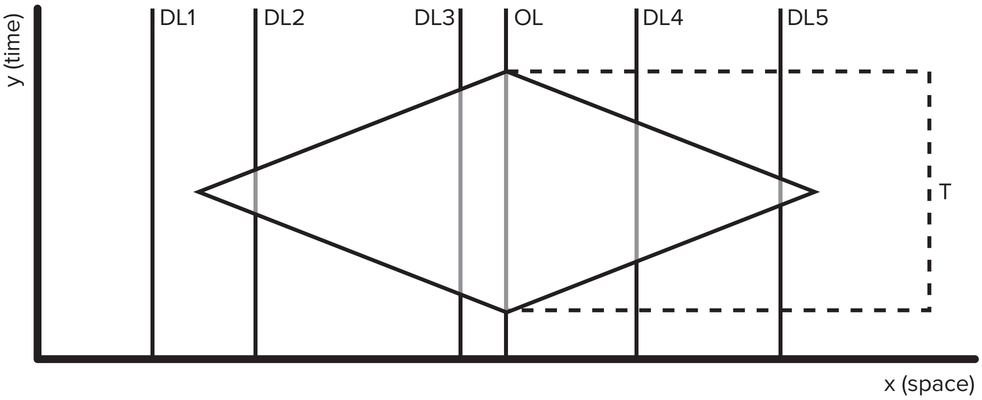


Figure 20: Hägerstrand's time-space prism as a tool for intersecting possible destination locations (DL1-5) from a given origin location (OL), within a timespan (T).

past trajectories should allow the model to build a location-specific set of constraints. Applying those constraints to the network should enable the modelling and thus predicting of movement for a specific user within their personal spatio-temporal network (see figure 19).

Using Hägerstrand's prism visualisation, a prediction for a trip opens a new prism, which can be constrained by the information aggregated from the user's trip history. Thereby, the resulting prism overlaps with a set of locations that represent potential destinations for the user (see figure 20). In conclusion, the novel combination of the spatial network model with the time geography constraints delivers a user-specific perspective from the current standpoint of the user in his or her network, perceived in a temporal continuum, which enables the prediction and analysis of potential destinations.

3.3 CONCEPTUAL CONCLUSION

In conclusion, the third chapter used existing concepts and models to conceive a new conceptual model for the algorithmic developments to be outlined in chapter 6. The model is built upon requirements that are derived from existing literature on LBSs as well as technological developments in the context of the application domain (chaper two). They relate to the research question of **how we can make use of smart data-driven services while sharing only as much data as is needed** (chapter one). The principle of *data-austerity*, introduced in chapter one, was used as a guideline. The technology stack provided the input data for the modelling (locations and trajectories) as well as a fundamental concept for structuring the computational setup (*selective cloud computing*). Based on the requirements and the technology stack, a modelling approach was selected. Central to this approach is a spatial network which incorporates temporal and semantic features. The proposed model should allow the client-side implementation to identify contexts and predict spatio-temporal behaviour. Therefore, the concept of time geography is incorporated into the network model. By including spatio-temporal constraints, a concept for predictions is constructed. This model represents the central user model of this thesis. In order to meet the requirements outlined in chapter 2, the following chapter extends the conceptual user model with an approach for sharing it with a remote service in order to enable, for example, collaborative filtering.

4 INTEROPERABILITY & GENERALISATION

The previous chapter outlined the framework for modelling spatio-temporal behaviour for tasks of prediction and recommendation. One requirement still unmet by the sections above is the server-side collaborative filtering, which requires the user to share certain aspects of his or her data. The requirements clearly outline that for effective collaborative filtering, a user needs to share features of his or her spatio-temporal behaviour and his or her derived preferences with the remote service. This undermines the general goal of this thesis to keep the shared data at a minimum, following the principle of *data austerity*. To overcome this dilemma, the thesis proposes two concepts: **1) sharing a model instead of the raw data** and in addition, it should be explored if it is possible to **2) allow the user to determine the granularity of the shared model**.

As outlined in the technology chapter, most modern LBAs send all their data to the remote servers, giving the providing company full access to their users' data. By moving the modelling onto the user's device (*selective cloud computing*), the imbalance is changed and the user is put back in control over his or her data. Collaborative filtering requires a set of user-specific preferences (e.g. location history), which can then be used to correlate users and find common user groups (e.g. based on interests). Therefore, an additional requirement for the client side is to provide data to the server side that allows for such comparisons. In order to maintain the users' privacy, this thesis proposes the use of generalisation. Generalisation here is meant in the sense as cartography applies it to map making.

Generalisation is the process of filtering and abstracting spatial information depending on map scale and map purpose. Over recent decades, several conceptual models of this process have been suggested (see e.g. McMaster and Shea, 1992 for an overview). Traditionally, a cartographer applies this process manually. With the introduction of GIS, researchers have worked on automating this process and thereby introducing semi-automatic generalisation concepts. *Semi* in such a way that most of the process becomes

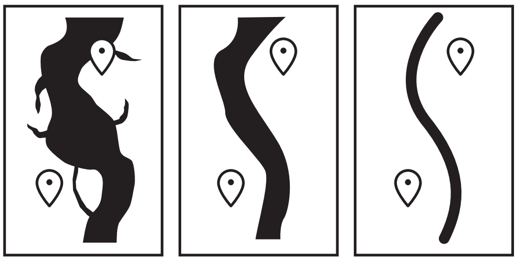


Figure 21: Spatial relationship preserving generalisation.

automated while still relying on human-made rules for the generalisation process. Consider the display of a river on a map for example: in a digital map creation process, the line or polygon data of the river can be progressively simplified automatically as the zoom level of a map decreases (see figure 21); A cartographer has to define those levels of simplification manually and, more importantly, decide when to show and hide the river. While the latter could be achieved by calculating visual complexity, the reason for the human-made decision is a focus on the purpose of the map. While a map of waterways needs to show rivers at all zoom levels, a highway map might only show rivers on high zoom levels. This makes all maps human-made products, as the early text from 1908 on generalisation already highlighted:

> "In generalizing lies the difficulty of scientific map-making, for it no longer allows the cartographer to rely merely on objective facts but requires him to interpret them subjectively. To be sure the selection of the subject matter is controlled by considerations regarding its suitability and value [...].[...] generalised maps and, in fact, all abstract maps should, therefore, be products of art clarified by science"
>
> (Eckert and Joerg, 1908, p. 347).

Two important aspects make generalisation a compelling conceptual model for being used as a foundation for sharing the spatio-temporal models in the proposed system. On the one hand, generalisation uses abstraction and simplification to render objects at various levels of detail. This process brings about data loss and inaccuracy. On the other hand, the same process attempts to maintain certain aspects of the topology of the spatial information (e.g. distance, orientation, etc.). Applying this for example to a user's spatial network could mean that the accuracy of the spatial vertices is decreased while the edges between vertices remain intact. Thereby, it allows us to change the accuracy of the shared data while maintaining general relationships and patterns. In this sense, the process of creating maps at various zoom levels of the same spatial information is applied to the process of creating data models at varying levels of granularity or rather levels of detail.

To guide this process of generalisation, Robinson and Morrison have proposed a further formalisation of the generalisation process on the basis of four steps simplification, classification, symbolisation, and induction as well as four conditionals objective, scale, graphic limits, and quality of data (Robinson et al., 1995). This formal framework provides a conceptual guideline for the generalisation process of the models and spatio-temporal networks to be developed in the next chapters.

The challenge in applying the conceptual model of generalisation to the shared data is the aspect of temporality and semantics. While generalisation in its cartographic sense is applied to spatial information, the models above also include temporal and semantic features. Therefore, the following chapters attempt to incorporate the above-outlined conceptual model of generalisation into temporal and semantic features. In addition, the level of generalisation in cartographic products is usually defined by the level of zoom or traditionally the scale. As previously discussed, an ideal framework would allow the user to define such a scale. This thesis thus explores ways of using generalisation to provide users with such novel capabilities.

In order to allow for collaborative filtering, the generated models need to be shared with a remote service. Therefore, the concept of generalisation is applied to the models in order to change the level of detail of the shared information (see figure 22 for an overview of the whole conceptual model from chapter 3 and 4).

In contrast to this individual user-specific perspective, which is represented by the conceptual models discussed in chapter 3 and 4, most related work in this area is concerned with deducing general patterns by aggregating data from multiple users. Such approaches de-emphasise the individual in favor of an aggregate. In those cases, the individual often only stands out as an *outlier*. The following chapter emphasises the egocentric perspective taken by the proposed techniques (for a discussion on egocentric geovisualisations see Meng, 2004). The chapter will take up a specific focus on time geographies and their capability of capturing a personal perspective in relation to urban space and its individual usage by that person.

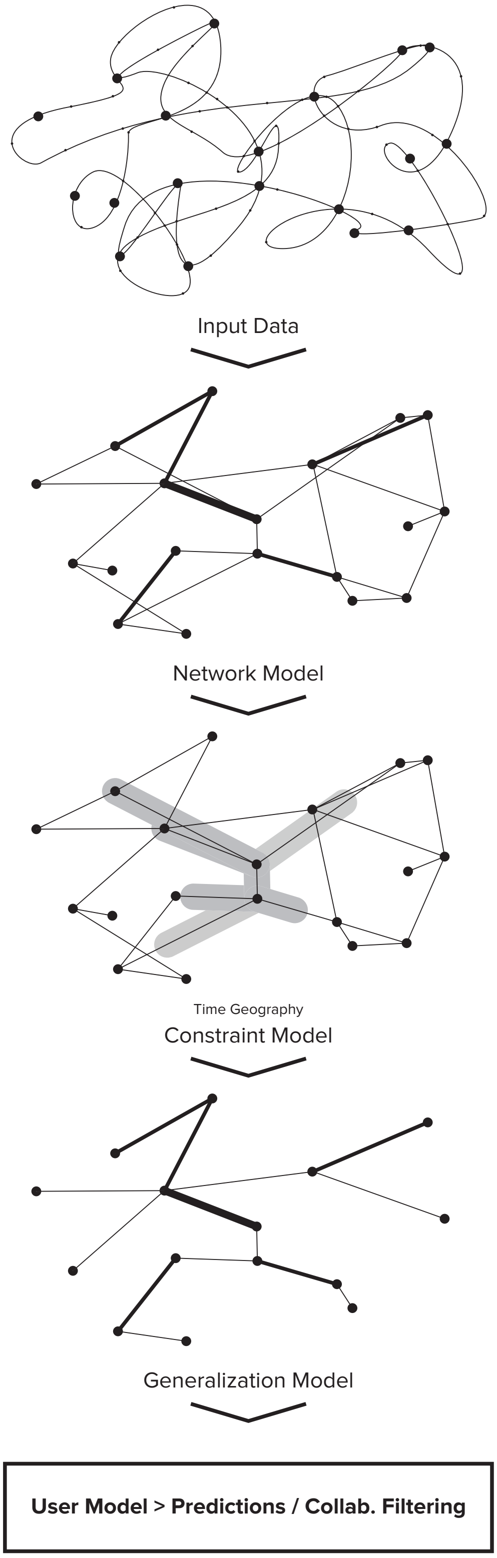


Figure 22: Construction of the conceptual model.

5 THE INDIVIDUAL IN THE DATA

Chapters three and four developed a conceptual model for capturing the user's spatio-temporal behaviour. In contrast to most existing examples, this modelling process is only built on the user's own data, and as a result, takes only a very individual user-specific perspective. The following section will explore the broader implications and potentials of such individual perspectives.

5.1 INDIVIDUAL PERSPECTIVES

The following section is in part adopted from Meier and Glinka, 2017, written by the author of this thesis.

Geography and cartography alike have a long history of critical perspectives within the respective disciplines. An important critique, often strongly influenced by post-colonial and feminist theory, is directed at the manifestation of power imbalances inherent to maps and cartographic products in general. One of the main counterarguments to cartographic practice emphasises its complicity in the reproduction, reification, and stabilisation of cultural or social imbalances. This is being most prominently discussed in relation to the infamous Mercator projection, which has become the standard projection for most web-based slippy maps (see e.g. Monmonier, 2004 and Cosgrove, 1999, p. 217). The underlying paradox is that cartographic representations could execute the power to question the status quo and establish alternative perspectives within and through established representations (Wood, 1992). Similarly, time geography has been accused of simply reproducing established power structures while remaining unreflective on that capability (Kwan, 2007; Kwan and Ding, 2008; Scholten et al., 2012; McQuoid and Dijst, 2012). This capability is sometimes referred to as *counter-mapping* (see e.g. Mitchell and Elwood, 2015; Lee Peluso, 2011; Hodgson and Schroeder, 2002). Prominent examples of counter-mappings can be seen in the works of the *French Situationist International* in the 1950s. They tried to visualise a concept referred to as psychogeography. The term 'psychogeography' was most prominently coined by Guy Debord as "the study of the [...] specific effects of the geographical environment, whether consciously organized or not, on the emotions and behavior of individuals" (Debord, 1955). Likewise, the approach presented in this thesis that is based on *personal big data* attempts to predict how the geographical environment affects the behaviour of the individual in particular. The approach of *personal big data* thus seeks to emphasise such perspectives of the individual over the aggregated *big data* patterns. It seeks to account for the complexity and diversity of urban space and life.

Many publications within this field of study cite the spatial knowledge modelling approach by Kuipers. Kuipers' concept of modelling spatial behaviour is inspired by the mental perception of space and the development of personal spatial knowledge (Kuipers, 1978). This thesis follows this line of thought and borrows from the domain of mental maps and personal perceptions of space in order to shape models of personal spatial behaviour. Several academic disciplines, ranging from sociology to psychology and philosophy, have explored such individual and subjective perceptions and representations of the physical world and its social and mental manifestations. The construction of mental maps as a personal vehicle of spatial perception and knowledge remains central to this discussion. The following paragraphs provide an overview of the discourse on personal perspectives on space and mental maps in order to guide the development of the executive models described in the previous sections.

A large body of research has emerged on the question of how people construct spatial knowledge and how this knowledge is used to make sense of the physical world, for instance when performing navigational tasks. A dominant image used in this discourse is the mental map. A general hypothesis within these discourses assumes that an aggregation of the entirety of our experiences form and influence our mental representations of space (Montello, 2013; Kitchin, 1994). Tversky extended the metaphoric concept of the mental map to mental collages, as those mental representations are not solely of map-like forms (Tversky, 1993). More generally, the correlation between map use and its impact on the shaping of mental maps has been a strong research focus, unravelling findings like the interdependency of orientation in maps and mental maps (mental rotation) (Tversky, 1981; Hintzman et al., 1981). Among these influencing factors, the mode of transport has been identified as one that has a significant impact on our perception of space and on how we interact with our environment (as will be discussed in the next chapter).

Scientists have tried to capture, model, and visualise mental representations of space, for instance by letting people draw or describe maps (Lynch, 1960; Vertesi, 2008). Research within neuroscience has produced valuable results in visualising neuronal maps that indicate where spatial information is stored in our brain (Maguire et al., 1997). However, visualising the resulting mental maps and making them usable (e.g. in computational models) has proven difficult. The task of the proposed modelling approach is not to create a model of a complete mental map, instead the modelling attempts to capture the user's personal behaviour and in turn also his or her relationship to the city (e.g. a bar (location) might be a place for spending leisure time for one person, while it is a workplace for another person). Therefore, the following chapters will use research on and concepts of mental maps, or rather personal individual maps, as an influential construct in the modeling of personal perspectives within user models. The notion of the importance of the personal perspective should help as an underlying guideline throughout the development process of the computational models.

The next section takes this individual perspective a step further by establishing a modelling critique. This critique should not undermine the previous chapters, but instead contextualise the modelling approaches in a broader framework.

5.2 critical reflection on data & modelling

Technological research and the natural sciences often take a positivist attitude towards their own research. The same stands for many publications written on modelling, which for example state that modelling "[...] allows for a precise abstraction of reality[...][, it allows] master complexity [...][, and is] validated by solid mathematical foundations" (Banks, 2010, p. 22)[7]. This contradicts the primary principle of models, which states that they "are *approximations* [emphasis added] of the real world" (Banks, 2010, p. 1), making them imprecise by nature. In the following, the downsides of models, especially data-driven approaches to modelling, are discussed.

A model is incomplete and inaccurate compared to the real-world entity or process it attempts to depict. In part, it is designed to be that way, and in part, it simply lacks data and insight. In regards to the research described in this thesis, some operations in creating the model are vulnerable to such objections. In order to create a model, one has to acquire information. For the purpose of creating a cartographic model, for example, one has to measure the physical world. In our case, we use data gathered through a smartphone. More precisely, we use the smartphone's GPS sensor for positioning and its accelerometer for identifying the activity type. The decision to use exactly this set of sensors and interpret their output in the way done in this publication is a (human-made) decision. It is a decision that influences the resulting data and thereby the model generation. Many such decisions accumulate when designing a data acquisition process, from the choice of sensors to the way the data is being stored in the database. Different choices will result in different data, data structures and finally different models. While we try to come to a decision using logical reasoning, we remain susceptible to preconditioned systemic influences. Therefore, the way we design our data acquisition techniques as well as the way our modelling algorithms are designed must be perceived as human-made and thus also as *cultural artefacts* and not as seemingly objective *technological artefacts*. By highlighting those

[7] The goal of this section is not to highlight misconceptions resulting from other research. Therefore, no further references for such attitudes are provided. The critique is more about a general mindset of the academic community than about individual researchers per se. One exception is the exemplary citation from Banks, whose work is cited throughout this thesis and whose foundational books on models and simulations have been an important and helpful resource for the development of this work.

influences on the data acquisition process it should be clear that data is not objective. People tend to trust numbers or rather quantitative data implicitly, as Porter illustrates in his 1995 book 'Trust in Numbers' (Porter, 1995). This even accounts for visualisations built from quantitative data, as in the ones created in this thesis, which help as visual interfaces to the models. Authors are in line with Porter's perception and speculate on the diffusion of this concept of trust into diagrammatic representations like maps and information visualisations since they are built upon quantitative information and as a result might inherit a certain aura of truth (Monmonier, 1991). Halpern, for example, calls this quality 'communicative objectivity' (Halpern, 2014). Similar downsides have also been pointed out in the academic discourse on data-driven journalism (Lewis and Lewis, 2014) and cartographic journalism (Green, 2013; Vujakovic, 2013). Beyond these systemic biases, which insinuate themselves into our technologies, our technologies are also prone to failure. Missing data or inaccuracies further falsify our models. In addition to these unwillingly included issues, we design our models to be an abstraction of reality. Real-world entities and processes in their full intricacy are too complex to be used in a model. Therefore, one has to create an abstraction. One must create an abstraction that summarises or combines properties as well as leaves out properties, in order to make it manageable and usable. I want to stress this point to make sure that when we talk about a user's spatio-temporal behaviour model, that we are discussing *one possible abstraction* of this user's behaviour. It remains a small extraction of the user's complete real-world behaviour. It is rather turned into an abstract model, inducing uncertainty and error, in order to make it usable and applicable.

As more and more areas adopt Machine Learning (ML) algorithms to deal with complex data analysis to the disadvantages of traditional statistical evaluations, the data which is used to train those ML models needs to be thoroughly investigated. This is because biases in the data will result in biases in the models and resulting predictions. Even though the models developed through the approaches in this thesis are user-specific, we also discuss the possibility of recombining those models into global models. In those cases, one also needs to be aware of biases in spatial trajectories (Johnson and Hecht, 2015). All the test-datasets used in this thesis, for example, stem from urban areas, and thus neglect rural ones.

A thesis in the area of geoinformatics would not be feasible if one could not build upon accepted conventions and conceptual models, like for example the geographic coordinate system. This thesis does not have the capacity to discuss each of the foundations that it thesis relies on. Therefore, it should be stated that the author is aware of the, for example, cultural bias, as introduced in the paragraph above, as among those foundations that shape and inform the overall knowledge constructed in this thesis.

Moreover, the above section acknowledges the incompleteness and uncertainty of the techniques developed in this thesis, as well as the potential bias introduced by the white Western male author of this thesis. Recognising that the modelling and prediction approaches developed in this work are at best good approximations of reality and at worst simply a solid hypothesis in the right direction, these approaches can never constitute an exact replica of reality itself. For the sake of writing style, the above is implied and will not be mentioned every time a model is described and discussed.

6 COMPUTATIONAL MODEL

The previous chapter introduced the conceptual model, which serves as the theoretical foundation for the computational model developed in this chapter (see figure 10). This chapter starts with the basis of the models: the real-world datasets that are being used to evaluate the algorithms, and the spatio-temporal data processing. The resulting data is then used in the third section to identify locations and finally, in the fourth section, to construct the network. The predictions derived from the behavioural models will be combined in the subsequent chapter in order to connect the client- with the server-side by introducing an event-based communication. This communication makes use of generalisation approaches for data anonymisation. Each algorithm presented in this chapter is exemplary for showing the applicability of the conceptual model from the previous chapters. While the decision-making that led to these specific examples is discussed, it should remain clear that the presented examples are only some of numerous viable solutions available. Throughout this chapter, visualisation acts as an interface for computational models. Thereby, the models can be inspected, validated, and improved[8].

6.1 TEST DATASETS

In order to test the computational models developed in the following sections, two datasets are used. The first dataset is the *GeoLife* dataset (Asia, 2012), which was produced by the *Microsoft Asia Research* team from April 2007 to August 2012 and published later that year (Zheng, 2007). The dataset includes GPS trajectories with classified activities from 182 users. The team around Zheng has collected and investigated the data in order to learn more about spatio-temporal behaviour in the area of Beijing (China) (Zheng et al., 2008a; Zheng et al., 2008b; Zheng et al., 2009). This first dataset is primarily used to test preprocessing and thereby simulates the process on an actual mobile device. Due to the short continuous observation spans of the *GeoLife* dataset, it can only be used for the location type predictions, but not for the trip predictions, as the dataset does not include enough trips to identical locations.

The second dataset is made up of six individual spatial datasets, collected from six individuals in the area of Berlin, constructed through the aforemen-

[8] The corresponding code for each of the experiments and developed solutions can be obtained from the following GitHub repository: https://github.com/sebastian-meier/Personal-Big-Data. As mentioned in the beginning of this chapter, some datasets are private and are therefore not included in the repository.

tioned *Moves* Application. Each individual spatial dataset is spread out over the time period of one year (four men, two women; ages ranging between 25 and 40 years; two students, three employees). While the *GeoLife* dataset is open source, the latter dataset was only collected for the development and testing of the models in this thesis. To ensure the participant's privacy the datasets are not available outside of this thesis. Due to privacy concerns, the second collection of datasets is neither included in the example visualisations. Only the thesis' author's trajectories are visualised. The two example datasets are structured as follows:

***Moves* Data**

The data from the *Moves* application is provided in GeoJSON format (Butler et al., 2008). On the upper-level, a FeatureCollection encloses all items in the dataset. Within the feature collection, the data is structured in what *Moves* calls a 'storyline'. The storyline is a continuous list of events ordered by time. The events are either trips or places. The trips contain the spatial trajectory accompanied by the detected activity as well as a temporal start- and endpoint for the trip. The places between trips contain the coordinate for the place, time of arrival and departure, semantic information like location type, as well as a location ID. In addition, the place item contains information on the activity within the location, which is usually walking activity (an example dataset excerpt is provided in the appendix). The latter place activity is not used in this thesis, as the *GeoLife* dataset does not contain this kind of information.

***GeoLife* Data**

The *GeoLife* data is in a more raw format than *Moves* and is hence transformed into the previously-cited *Moves* format. The *GeoLife* data is provided in folders per subject and divided into multiple *plt* files (An example dataset excerpt is provided in the appendix). Plt files contain a header with metadata followed by a Comma-Separated-Values (CSV)-formatted table. The plt tables contain one point per line from the user's trajectories (x, y, altitude, timestamp). The data has a very high resolution, usually multiple points per minute. While the *Moves* data for the collected participants contains continuous information for a year with only a few gaps, the *GeoLife* data contains many gaps, often over multiple days within the tracking timespan. Some of the datasets contain an additional *labels.txt* file. This file is a CSV formatted list of detected activities, where each activity has a start and an ending timestamp. The detected activity can be applied to the trajectory points through an additional processing step.

The following preprocessing approach is designed to be applied to raw input data that is collected by the user's device. In order to test it, it will be applied to the above-described datasets.

6.2 SPATIO-TEMPORAL DATA PREPROCESSING

The section on spatial data from smartphones in chapter 2 described how to acquire the current geographic position (P) that consists of a two-dimensional coordinate (x, y) from a user's device. On most devices, as in the example datasets described above, those coordinates are provided in the spherical Mercator projection (WSG84, EPSG:3857), where x is the longitude in the range of -180 to 180 degrees and y is the latitude in the range of -90 to 90 degrees. In addition, the activity recognition APIs, as described in the technology section of the previous chapter, are utilised. For each position in the recording, the activity (a) with the highest probability is stored with the position.

A spatial trajectory is defined as "a trace generated by a moving object in geographical spaces, usually represented by a series of chronologically ordered points." (Zheng, 2015, p. 1) (see also Spaccapietra et al., 2008). As such trajectories (T) are recordings of a certain timeframe (t_1 to t_n), a timestamp (ts) must be attached to each position. Depending on the device, additional parameters are added to the position, for example, altitude or, based on interpolations from the previous position, speed and acceleration.

$$P = (x,\ y,\ a,\ ts) \tag{1}$$

$$T = [P_1,\ P_2,\ \dots,\ P_n] \tag{2}$$

Before those trajectories can be used for further calculations they need to be processed for two reasons. On the one hand, such processing improves the dataset and reduces errors and, on the other hand, it reduces data reduction, so that we can save storage on the device.[9] For error reduction, a weighted median filter is used instead of a mean filter as "it is less affected by outliers" (Lee and Krumm, 2011, p22). Since the dataset includes activity classification, the median filter is activity-specific (see figure 23). The datasets used in this thesis include sparse positions due to the battery-saving approach. Hence, only the next position can be corrected, using the next positions normalised vector (V) and the weighted median speed of the previous positions. For a new position in the trajectory (P_p), the previous step's speed (S) of the same activity with a limit (l) are taken into account (S_{p-1},...,S_{p-l}), and a linear weight (W) is assigned to it in order to account for changes in e.g. speed. Having a trajectory history at hand, a threshold for the specific activity can be calculated. This threshold is used to define a tolerance for each new position. If a step exceeds the tolerance threshold, the median filter is applied. By doing so, the median filter process can be sped up.

[9] An overview of existing techniques on the preprocessing of spatial trajectories that strongly informed this section is provided by Lee and Krumm, 2011.

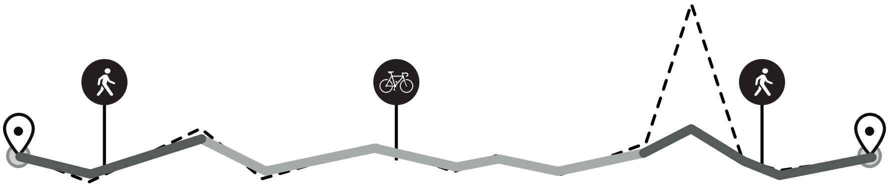


Figure 23: Original line (dotted) and the resulting line after applying the median filter.

$$W_{weight} = \frac{(l+1)-n}{l} \tag{3}$$

$$S_{speed} = \frac{|(P_{p-n}\ -\ P_{p-n-1})|}{t_{p-n}\ -\ t_{p-n-1}} \tag{4}$$

$$weighted\_median\{(S_{p-1}, W_{p-1}),\ ...,\ (S_{p-l}, W_{p-l})\} \tag{5}$$

$$if(S\ >\ activity\_threshold\ >\ S)\{P_p = P_{p-1}\ *\ V\ *\ weighted\_median\} \tag{6}$$

To reduce the positions in the trajectory, in order to reduce memory usage and improve processing speed, two techniques are used. First, locations are detected by recognising spatial inactivity. This is done by finding consecutive positions which do not exceed a predefined spatial error threshold (et) (see figure 24). The threshold has to be introduced to balance the errors in the dataset. While one would assume that a user's stay in a location is represented as one and the same coordinate, the inaccuracy of the GPS signal will result in many slightly offset coordinates. Therefore, for a series of positions which do not exceed this threshold, only the first and last position are stored. The geographic coordinates of the two edge positions are calculated from the median of all positions (including the dropped positions). Beyond the inactive positions, the *Douglas Peucker Simplification* algorithm is applied (see figure 25) for reducing the positions of the trajectories (Douglas and Peucker, 1973). The *Douglas Peucker* algorithm "produce[s] the most accurate generalization" (Shi and Cheung, 2013, p) when the difference between original and optimised path is the quality criterion[10].

The previously described preprocessing steps, which would normally be applied directly on the user's device, are applied to the *GeoLife* and *Moves* dataset. The *Moves* dataset format serves as an intermediary format, therefore, the *GeoLife* data needs to be transformed into the *Moves* storyline format.

[10] In evaluations with a different focus, for example generalisation, other algorithms outperform the *Douglas Peucker* approach (Visvalingam and Williamson, 1995).

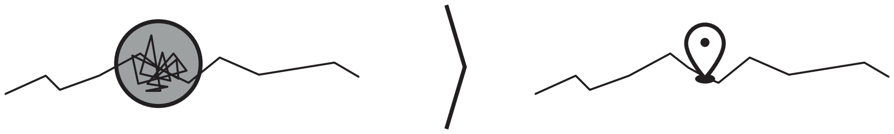


Figure 24: Spatial inactivity threshold for identifying a location.

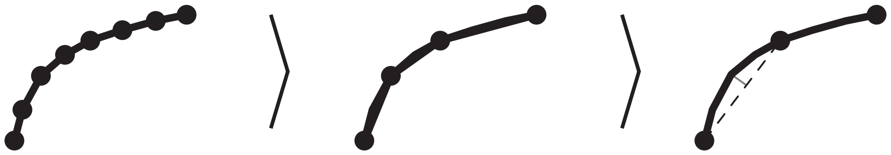


Figure 25: Simplified example for a progressive line simplification.

Using the median filter, the dataset is cleaned and filtered. Then, temporal gaps and location visits (spatio-temporal threshold) are detected and a unique ID is created for each location. The overall dataset is then split into trajectories between locations. For those trajectories where activity labels exist, the trajectories are further refined through the activity data. In so doing, this process splits the trips between locations by activity type into segments. The previously-mentioned *Douglas Peucker* algorithm is applied in order to reduce the data while maintaining the topology of the trajectories. The processed data is then stored in the *Moves* GeoJSON format.

In order to increase performance for subsequent queries on the trajectory dataset, the data is stored in a database. One of the most advanced databases that are available for mobile operating systems (*iOS* as well as *Android*) under an open source license is *SQLite* (SQLite, 2017). *SQLite* can furthermore be extended through the *SpatiaLite* extension (Furieri, 2017), which makes it the best fit for the use case in question. *SpatiaLite* adds spatial capabilities to the SQL-based database. *SpatiaLite* is similar to *PostgreSQL's* (PostgreSQL, 2017) *PostGIS* extension (PostGIS, 2017). *SpatiaLite* is a relational database. Anticipating the need for locations (vertices) as well as connections between locations (edges), the database is structured accordingly[11]. One table contains locations. Locations are the previously-mentioned consecutive positions with spatial inactivity. Locations with a distance smaller than the predefined error threshold are assumed to be the same location. Connected to the location table is a table containing events at a location (user visiting / staying at a location). The third table contains trips between location A and location B. A fourth table contains trajectory segments for each trip. A trajectory is split by activity type (e.g. bike + train + car) into such segments. The next two sections will use those four tables to classify locations that the user visits in the future and to build the spatial network.

[11] Other database formats that are optimised to represent network structures are graph databases that, in contrast to *SQLite* + *SpatiaLite*, do not support spatial data optimised structures and queries.

6.3 location type identification

As described in the previous section, the definition of a location is a geographic position within an additional spatial error threshold which the user occupies for a certain time threshold (see figure 24). A two-step process attempts to enrich the location with additional metadata, more precisely the location type (category). In a first step, the system contacts a location API. A location API is an interface to a location database, examples for such are *Foursquare* (Foursquare, 2017b), *Google Places* (Google Inc., 2017a), *Facebook Graph Search* (Facebook, Inc., 2017), or *Yelp Fusion* (Yelp, 2017). One feature that those APIs have in common is the capability to query their location database on the basis of a geographic location (x,y). The result is a list of nearby locations with their geographic position as well as additional metadata (e.g. category). The system can then request feedback from the user to confirm the new location. The system suggests either the location list delivered by the API (see 2.2.2) or one of the predefined location types (Home, Work, School/University, Other Home, Leisure, Shopping, Eating/Drinking, Transport, Transit[12]) or the user can define their own location type. If a location from the suggested API locations is selected, the upper-level category (e.g. restaurant) is added as a location type.

As outlined in the previous chapter, temporality is an important aspect of the user's spatio-temporal behaviour. The same accounts for locations (see e.g. Noulas et al., 2011b; Yuan et al., 2013; Biagioni and Krumm, 2013 for temporal pattern recognition in locations). In order to improve the manual location identification process, a second layer is introduced. When a user confirms a location type, the temporal patterns of that location (time of arrival, time of departure, time spent at the location, the day of week and month) are stored in the location event table. The data is used to train an ANN, in order to identify the type of future locations with similar behavioural patterns. Each user starts with an untrained neural net. To compensate for the time it takes to train the personal neural net, a pre-trained net is running in the background. The pre-trained ANN is using statistical data from time use surveys (see 6.3.2), which give insight into where people are and what they do at certain times of the day. The decision to use this second ANN only as a backup goes back to the individual perspectives and the emphasis on diversity introduced in the previous chapter. Any machine learning approach that builds upon a dataset, that is aggregated from a large set of users, generalises in order to classify or predict. Therefore, edge cases are difficult to detect. In this particular case, an edge case might be the behaviour of a specific user that does not fit in the overall behavioural patterns of the aggregated mass of users. By allowing the users to train their own classification system instead of using a system which is creating aver-

[12] The set of predefined location types is a simplified version of the categories selected in the time use surveys, presented later in this section.

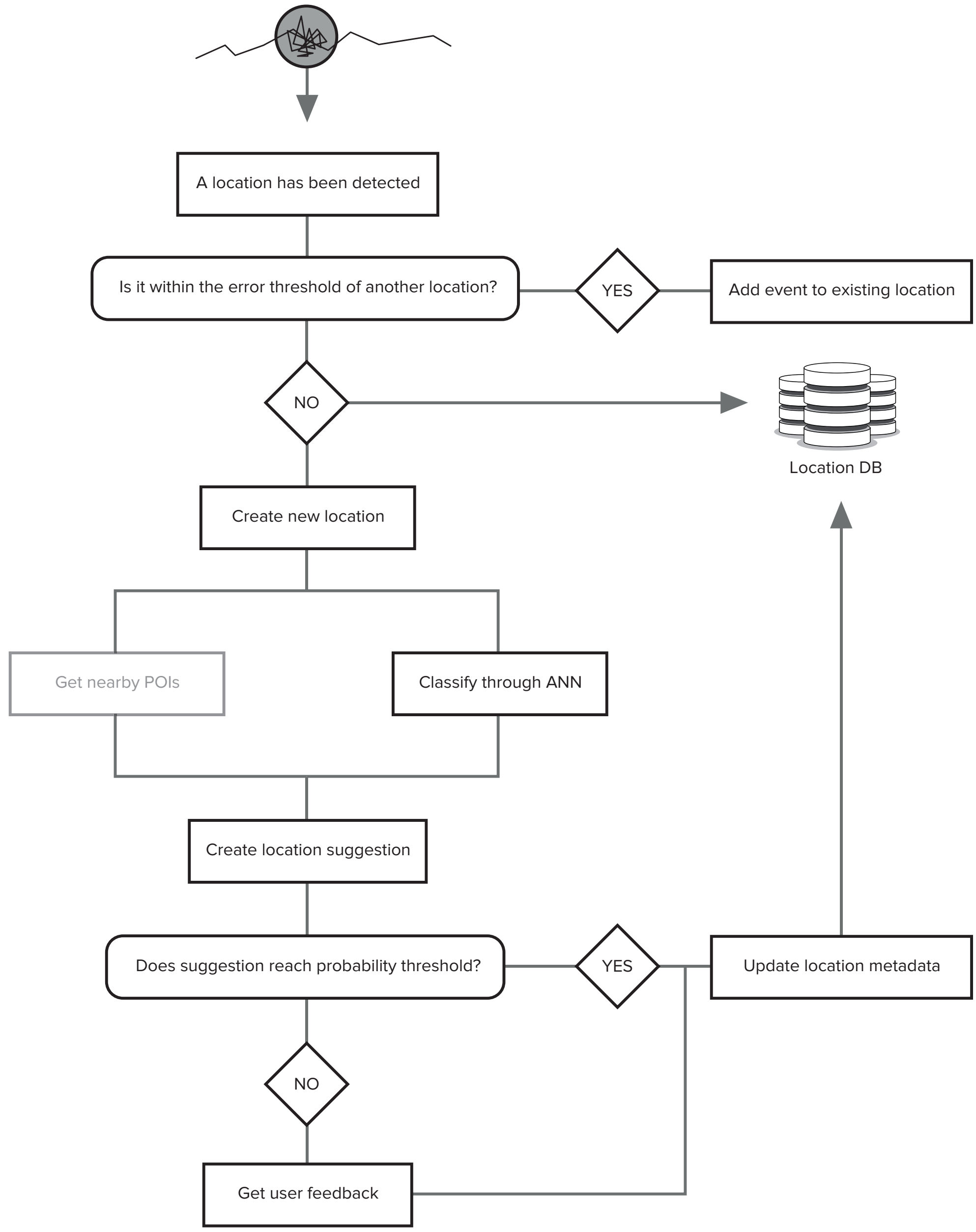

A location has been detected
Is it within the error threshold of another location?
YES
Add event to existing location
NO
Location DB
Create new location
Get nearby POIs
Classify through ANN
Create location suggestion
Does suggestion reach probability threshold?
YES
Update location metadata
NO
Get user feedback

Figure 26: Location identification process.

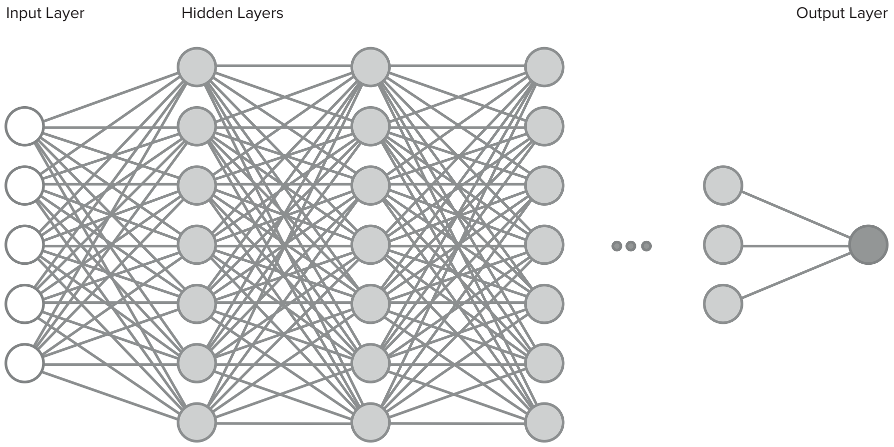


Figure 27: Example of an ANN setup.

age patterns across a larger userbase, the proposed system seeks to account for the individual perspectives of users. The following sections will dissect the previously-introduced approach and shed further light on the individual components. They will also evaluate the proposed system in regards to accuracy and test the individual versus the generalised perspective.

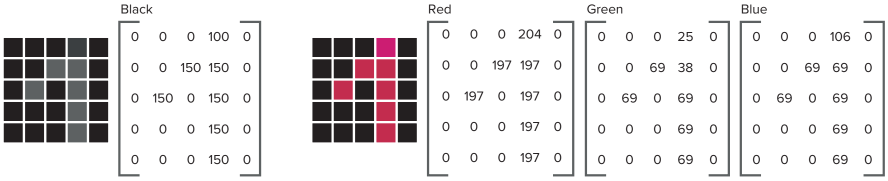
6.3.1 *Artificial Neural Networks*

A brief Introduction

ANNs belong to the group of machine learning techniques. While there is a variety of possible fields of application for ANNs, the most common application is data analysis and processing, including classification and pattern recognition. This is why the actual techniques are also often referred to as 'classifiers' or 'estimators'. The following describes the setup of an ANN to be used as a classifier[13]. The structure of ANNs is inspired by the nervous system of the human brain[14]. They are made up of so-called 'neurons' or 'nodes' which are organised in layers (see figure 27). Because of their layout, networks with many layers are sometimes referred to as 'deep nets' and their usage 'deep learning' respectively. Each layer contains neurons that are interconnected with the preceding and subsequent layer. Most important are the input (first) and the output (last) layer. The layers in between are called hidden layers. In order to process information, the information needs to be broken down into parameters. Depending on the system in use, they need to be further refined into continuous or discrete numeric values, which are then transformed into matrices or rather tensors. To give an example for added clarification, we will describe image classification, a common task for ANNs.

[13] The setup of ANNs for other purposes differs slightly.

[14] While the image of the human brain is a metaphor widespread in the ANN literature, lately some began challenging this concept, e.g. Gomes, 2014.



Black

$$\begin{bmatrix} 0 & 0 & 0 & 100 & 0 \\ 0 & 0 & 150 & 150 & 0 \\ 0 & 150 & 0 & 150 & 0 \\ 0 & 0 & 0 & 150 & 0 \\ 0 & 0 & 0 & 150 & 0 \end{bmatrix}$$

Red

$$\begin{bmatrix} 0 & 0 & 0 & 204 & 0 \\ 0 & 0 & 197 & 197 & 0 \\ 0 & 197 & 0 & 197 & 0 \\ 0 & 0 & 0 & 197 & 0 \\ 0 & 0 & 0 & 197 & 0 \end{bmatrix}$$

Green

$$\begin{bmatrix} 0 & 0 & 0 & 25 & 0 \\ 0 & 0 & 69 & 38 & 0 \\ 0 & 69 & 0 & 69 & 0 \\ 0 & 0 & 0 & 69 & 0 \\ 0 & 0 & 0 & 69 & 0 \end{bmatrix}$$

Blue

$$\begin{bmatrix} 0 & 0 & 0 & 106 & 0 \\ 0 & 0 & 69 & 69 & 0 \\ 0 & 69 & 0 & 69 & 0 \\ 0 & 0 & 0 & 69 & 0 \\ 0 & 0 & 0 & 69 & 0 \end{bmatrix}$$

Figure 28: Input tensors for a 5x5 pixel grey-scale image and an RGB image.

The MNIST[15] dataset, for example, is used for classifying hand-written digits in images. Taken a set of images, each 5 x 5 pixels in grey-scale (0 = black to 255 = white), the input tensor would be a two-dimensional tensor with 5 rows and 5 columns, one for each pixel in the image, each containing an integer between 0 and 255. If the dataset would include RGB images instead, it would be a three-dimensional tensor with 5 rows and columns times 3 for each colour (red, green, blue) value (see figure 28). Those input tensors can then be used in an ANN, which tries to assign a class (0-9) to each image.

To conclude, any input data that needs to be analysed also needs to be transformed into the above-described format. The same applies for the output. The output value, sometimes referred to as 'target', defines the desired output for each input data item. When ANNs are used for classification, one has to differentiate between two approaches: supervised and unsupervised. In a supervised mode, one needs to define the expected output classes, while unsupervised approaches allow the network to determine classes itself, by trying to identify patterns in the data. In the following, only supervised approaches are discussed. The creation and usage of such an ANN consists of three phases:

1) **The definition of the network**, specifies the number of layers and neurons, as well as further optimisation techniques, which are applied to the classification process (e.g. optimisers, loss function, etc.). As for most adjustments of ANNs, they strongly depend on the underlying data. This process of adjusting ANNs is guided by a number of rules: The number of nodes and hidden layers is proportional to the input and output nodes. More layers generally improve the accuracy of a net, until, at a certain threshold, the network begins to overfit, which will make the accuracy decrease at a certain stage. "Much the same behavior can be observed for decision trees as the number of nodes increases, or production rules, as the rule length increases." (Weiss and Kapouleas, 1989, p. 786). More and more sub-types of neural networks are introduced by current developments. In this thesis, only the very simple form of so-called 'feedforward networks' are used. In a 'feed-forward network', input parameters are passed to the network through the input layer and passed down until the information reaches

[15] Documentation of the dataset: LeCun et al., 1998b and the initial usage by LeCun and his team: LeCun et al., 1998a

the output layer. In order to improve the accuracy of the network, backpropagation is used. Backpropagation entails the comparing of the output of the network to the desired output and back-propagates the error to the nodes. Such simple 'feedforward networks' are sometimes also referred to as 'multi-layer Perceptrons'. In this thesis, these 'feedforward networks' trained through a supervised training approach (further explained for each use case). As the datasets in question are very sparse, an 'Adam Optimizer' is used.[16]

2) The training of the network with a training dataset. In this phase, the network is learning which inputs should result in what kind of output. During this phase, pre-classified training data is fed into the system. Thereby, neurons learn how the received input information from preceding neurons has to be processed and handed down to the next neurons. In a very abstracted and simplified manner, this leads to a weighted decision-making network through which the input data is transformed.

3) The deployment of the network is the final stage. The trained network can then be used to classify new information that it has not processed in the training phase. A common procedure for testing the accuracy of a network is to divide the pre-classified data into two parts, one for training and one for testing (e.g. 20% for testing). Once trained, the model of the network can be exported and reused.

While the structure of ANNs is different from traditional machine learning approaches (e.g. random forest), the three phases remain very similar.[17]

[16] While some statistically or empirically grounded guidelines exist for setting up and fine-tuning machine-learning algorithms like ANNs, in reality, the individual case strongly depends on the underlying data. Therefore, various combinations of optimisations need to be tested against those datasets. Taking into account the number of possible variations, this process can be quite time-consuming. An alternative approach, which has seen a lot of attention recently is *automated machine learning*. The concept behind automated machine learning is, to use an algorithm to work out the most successful variation of parameters on a specific dataset. One of the prototyping tools used in this thesis, *Convnet.js*, offers a magic class, "which performs fully automatic prediction given [...] arbitrary data. Internally, the MagicNet tries out many different types of networks, performs cross-validations of network hyper-parameters across folds of [...] [the] data, and creates a final classifier by model averaging the best architectures" (Karpathy, 2014). This idea is even picked up by startups like *DataRobot* (DataRobot, 2017), which provide fully automated infrastructures to their users in order to pick the best prediction techniques for the dataset in question. This development is in line with a trend for higher level APIs for machine learning techniques. *Google's TensorFlow*, for example, offers functionalities where the user simply needs to provide a set of classified images and the rest will be done by the underlying algorithms. Taking into account the ongoing developments in this area, this trend is likely to continue. Therefore, methods of machine learning will become available to a broader audience and more research communities. This will, at least for experiments and simple cases, make the fine tuning described in these paragraphs to a certain extent obsolete.

[17] ANNs are a very complex topic, this thesis can therefore only provide a short overview and more details on the exact implementation. Overviews and introductions to this topic are

ANNs for Spatio-Temporal Classifications

Over the last ten to twenty years, ANNs have experienced a peak in development and interest across research communities. In the geospatial sciences, common use cases are "classification, change detection, clustering, function approximation, and forecasting or prediction" (Gopal, 2016, p. 5). One example is the land use classification of data from aerial or satellite photography (see e.g. Carpenter et al., 1997). While image analysis and classification is a prominent example for supervised ANNs, they can be used for a variety of classification purposes, as long as they can be broken down into the previously outlined input and output parameters. This thesis explores their usage as classification algorithms for classifying and predicting spatio-temporal behaviours. The decision to use ANNs was guided by technological and data-driven statistical requirements. Similar to the underlying database, as described above, the classification technique needs to be implemented on the user's device. With the growing popularity of machine learning, several systems have been extended to be compatible with smartphone frameworks. One of the most advanced systems is *Google's TensorFlow* system (Google Inc., 2017b). The library is Open Source and available for Android as well as iOS applications. The mobile version is streamlined to work efficiently on a smartphone. Furthermore, the library does not only allow developers to implement ANNs, but also other machine learning approaches like, for example, random forest[18].

Beyond the technical requirements, the data or rather statistical requirements were even more important. Therefore, the following paragraphs discuss why machine learning was chosen over traditional statistical prediction methods and why ANNs in particular were chosen. As machine learning algorithms are not only more complex than traditional statistical methods for prediction, but also more cost-intensive (in terms of processing power, memory, time, etc.), there needs to be a good reason to favour them.

provided by Zeiler and Fergus, 2013; Wang and Raj, 2017; Schmidhuber, 2014. Probably typical for a cutting-edge technological development, many additional resources that are essential to not only understanding but implementing ANNs are web articles and discussions by the research and development community e.g. Kurenkov, 2015; Beam, 2017; Horowitz, 2017; Kogan and Tseng, 2016; Nielsen, 2015; Karpathy, 2017; Ericson et al., 2017; Stackoverflow, 2010.

[18] For prototyping, the JavaScript library *ConvNet.js* as well as the Java-based *WEKA* system were used in addition to the *TensorFlow* system (Karpathy, 2014; Machine Learning Group at the University of Waikato, 2016).

Carnahan et al. describes one downside of discriminant analysis and logistic regression:

> "Principal among [the limitations of discriminant analysis and logistic regression] is their dependence on a fixed, underlying model or functional form. Discriminant analysis uses linear summation of independent variables to differentiate one category from another (Huberty & Lowman, 1997). Logistic regression also makes use of linear summations of independent variables, incorporated into a logistic function (Myers, 1990). Koza (1992) made the observation that both techniques use regression merely to discover numerical coefficients for predetermined models."
>
> Carnahan et al., 2003, p. 409

This leads to the problem that in many cases "functional form cannot be established a priori. Such circumstances would necessitate the use of alternative classification modeling approaches."Carnahan et al., 2003, p. 409 This problem is also true for the approach in this thesis. In contrast, most techniques of machine learning belong to those methods, which" do not rely on predetermined models using linear summations of independent variables."Carnahan et al., 2003, p. 409. Here, Carnahan et al. summarise a general argument for machine learning over traditional linear statistics. A report by *McKinsey* illustrated this concept by comparing the risk prediction from two drivers (A, B) in a two-dimensional plot (see figure 29), highlighting the advantage of machine learning over, in this case, regression analysis (Pyle and Jose, 2015).

But, as figure 29 also illustrates, this effect obviously highly depends on the underlying data (compare e.g. Weiss and Kapouleas, 1989). Many of the above-mentioned papers, books, and websites even argue that strict generalisable conventions on when to use what kind of machine learning or statistical technique are difficult, especially due to the dependency on the data and its structure. Therefore, one should test various tools and find the best solution for the dataset in question. But beyond this general observation, Carnahan et al. (2003) mentions an important aspect of this thesis: the ability to establish a functional form *a priori*. This functional form includes, for example, the investigation of correlations between variables and building the statistical model a priori. In this thesis' case, the data is user-specific and generated on the user's device. Therefore, the prediction and classification methods need to adapt to each user's specific demands. Gopal summarises exactly this quality by stating that

> "[ANNs] have the ability to learn from experience in order to improve their performance and dynamically adapt themselves to changes in the environment. In addition, they are able to deal with fuzzy or incomplete information and noisy data, and can be very effective, especially in situations where it is not possible to define the rules or steps that lead to the solution of a problem. Hence they are fault tolerant"
>
> Gopal, 2016, p. 1.

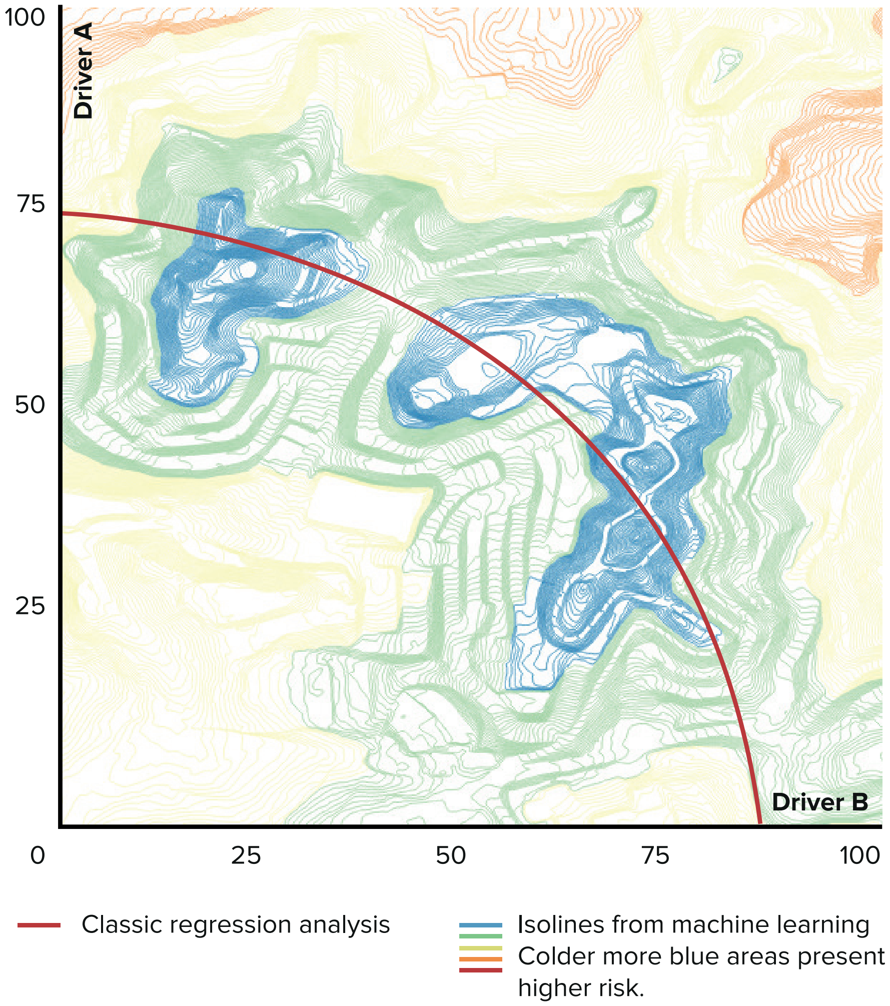


Figure 29: A highly abstracted illustration adapted from Pyle and Jose, 2015, to discuss the difference between machine learning and linear statistics.

This means that the input is predefined, while the output is defined by the user and the connections within the network are left for the ANN to be built.

The model develops on the user's device using the user's personal spatio-temporal behaviour. This method connects this choice of technique with the individual perspectives introduced in the previous chapter, as every model is, even if it is built on the same input parameters, highly user-specific.

The last feature that made ANNs a good solution for this thesis is their ability to be shared, recombined, and merged. The implementation in *Google's TensorFlow* allows the combination of multiple models in a process called 'ensemble learning', sometimes referred to as 'training ensembles'. This possibility will be explored as a way of sharing the model with the remote service, instead of sharing the raw input data.

The discussion above led to the decision to use an ANN for the classification process.[19]

Throughout the previous paragraphs, the choice for using ANNs as well as their basic functionality has been established. In what follows, this technique will be applied to the case of location type classification. In the case of the location type classification, two competing neural networks are implemented. One network builds on custom user-specific data and one is based on the time use survey data. Both are based on the assumption that the temporal usage of certain location types (e.g. home or work) generates user-specific patterns (see e.g. Noulas et al., 2011b; Yuan et al., 2013).

6.3.2 *Time Use Survey*

The properties that are extracted for classifying a location type are the time of arrival and departure, the duration of the stay as well as weekday (to account for differences between workdays and non-workdays[20]) and month (to account for seasonal changes).

The backup classifier is trained through time use surveys. Those surveys expose the above-mentioned properties of a set of participants. Time use surveys are conducted in many countries[21]. They use diary studies to gain insights into the activities and whereabouts of a representative sample of the inhabitants of a country over several days or even weeks. For the Beijing dataset, the author tried to obtain a copy of the Chinese 2008 time use survey (Fisher, 2017a), which was not available. The 2012/2013 time use survey from Germany was also not available at the granularity required for this thesis (Länder Forschungsdatenzentrum, 2016). Therefore, as an alternative, the time use survey from the UK was used. According to the results produced by Winquists's report, Germany and the UK share many similarities, especially the overall temporal patterns of weekends and weekdays (Winquist, 2004).

[19] The author acknowledges that many if not all of the above requirements are also met by techniques like random forest or other machine learning techniques. Overall it can be said that the performance strongly depended on the underlying training data. As every user's dataset showed different patterns they also performed differently. The choice to explore ANNs was based on their overall good performance as well as their novelty in the field of spatio-temporal behaviour prediction. As the focus of this work is not to compare machine learning techniques, which could easily fill another book, there is no further evidence provided by comparing all available machine learning techniques. The author clearly states that ANNs are one viable solution but not the only solution. To provide some comparison, for every ANN evaluation, a short list of comparable approach and their performance is offered through subsequent sections.

[20] In order to produce more diverse and individual models, workday and non-workdays are not predefined, but establish themselves through the user's behaviour.

[21] A very comprehensive international list of such studies is maintained by the *Centre for time use research* at the sociology department at Oxford University (Fisher, 2017b)

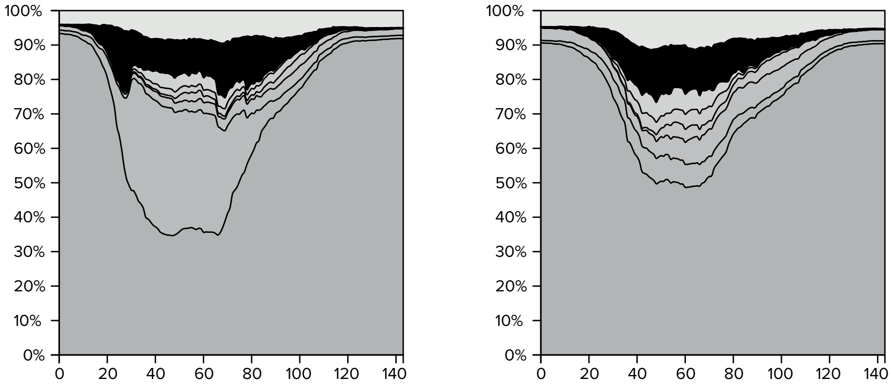


Figure 30: UK time use survey data for weekday (left) and weekend (right). Each area represents one location type. The y-axis shows the percentage of participants currently in a location type. The x-axis highlights 10-minute intervals of a 24-hour day, starting at 04:00. As an example the area for transport is highlighted in black.

The raw data from the 2014-2015 UK time use survey (Oxford, 2015), is organised in a tab-separated spreadsheet format. Each line represents one entry from a person for one timeslot. Each entry contains an extensive set of additional metadata, although for this thesis only the location and the day of the week (weekend, weekday) is relevant (a summary of the data is shown in figure 30). The granularity of available location types in the dataset is very high. Therefore, the location types have been grouped and reclassified (the classification is available in the appendix).

Each item in the dataset is used as one training item for the neural network and contains: location type, start time, end time, duration, and weekend/weekday. The property month is not used, as the study was not conducted over a long enough period to account for seasonal changes. After removing events with missing data, the dataset resulted in 92.627 items. Two alternative input approaches were tested: 1) using the previously introduced parameters start time, end time, and duration; 2) instead of start and end time and the duration, the three parameters were combined into 144 binary parameters, each for a 10 minute interval of a 24 hour day[22]. This approach was inspired by the process by which two-dimensional images are mapped onto tensors. Based on the two datasets, two ANNs were set up. Taking into account the guidelines for layers and neurons the according setup was 1: 10-20-10 and 2: 100:50:10. For optimisation purposes, the *SGD* algorithm was used (see for an overview discussion of gradient descent optimisation algorithms see Ruder, 2016).

In a first run, the datasets were split into 83364 items for training and 9263 items for testing (10% randomly picked testing items from the original set of items). The variation between the two input approaches was not significant

[22] The day in this dataset starts at 04:00 in the morning. According to the data, four o'clock is the time with the least activity and therefore a good breakpoint.

Technique	Correctly Classified Instances
Random Forest	66.29%
Multi-Layer Perceptron (ANN 5-10-20)	66.99%
Multi-Layer Perceptron (ANN 5-10)	66.67%
Multi-Layer Perceptron (ANN 5)	66.78%
Logistic Regression	65.00%
TensorFlow ANN	67.01%

Table 1: Comparing the performance of the time-use survey ANN against other machine learning techniques.

and ranged between one and two percent. As the input approach with fewer parameters was obviously faster, it was favoured for the following tests. The overall accuracy of the ANN was between 63% and 70%, averaging at 67%. A more in-depth analysis of the accuracy revealed that the location types which occurred more often received higher accuracies (home and work). A reduced set of location types (removing minor classes) reached an accuracy of 87.27%. The problem of unbalanced datasets for training is further discussed in the next section.

In addition, in order to highlight the choice of ANNs a comparison of the same training data applied to different machine learning techniques was performed using *WEKA* (see table 1).

In a second run, the trained dataset from above was tested against 400 randomly selected pre-classified location events from the *Moves* Application datasets. The ANN achieved an average of 77.78% accuracy on these locations. When reducing the location types to home and work, the accuracy reached even 98.41%. The increase in performance on the *Moves* dataset is likely due to less variation in the individual's behaviour, compared to the time-use survey.

6.3.3 *User-specific Model*

When locations are identified through the spatio-temporal threshold, they are classified by the user. In order to automatically classify future locations, an ANN is trained on the temporal patterns for each location type (the day of the week, time of arrival, duration, and location type). The *Moves* dataset was used for an initial testing, as it already contained user-classified locations. The test was conducted for four individuals. For each individual, the dataset contained 500 classified location events, for 3 to 7 different location types. The small number of location types was the result of dropping location types which had less than 10 events or where locations were not classified. A problem that arose early during the test was that all individuals had strongly unbalanced location events data. This means that a small

a	b	c	d	e	f	g	<– classified as
354	38	24	7	42	28	7	a = home
0	27	23	0	1	0	9	b = work
0	8	104	11	1	0	36	c = transport
12	5	47	93	1	0	2	d = food
68	24	8	2	47	8	3	e = shopping
24	2	1	2	7	124	0	f = other_home
0	7	40	0	0	0	113	g = doctor

Table 2: Extract from the unbalanced example evaluation.

number of location types accounted for most of the events. In all cases, these were the events for home and work locations. To account for this imbalance, a three-step approach inspired by the *SMOTE* algorithm (Chawla et al., 2002) was incorporated. The general concept behind *SMOTE* is the oversampling of minority classes. Therefore, KNN item-clusters of the minority classes are being generated, which are then used to build new synthetic items. Incorporating this concept, a histogram of events per location type is created. Through a *Sturges threshold*, the optimal number of bins is calculated (nb) (Sturges, 1926). The bin index (bi) (1 for least events to nb for most events) is assigned to each group of location type events and is used as a weight to even the number of overall location events, as a reference the maximum (m) from the highest bin is used as a reference. Thereby, the needed number (n) of events for a location type within bi is defined by

$$n = \frac{m}{nb} * bi \tag{7}$$

If the actual number of events in a bin for an event type is smaller than n, the missing items are synthetically generated through the SMOTE concept. In order to evaluate the resulting ANN, a test set (t) was extracted (before the SMOTE oversampling) where the distribution of events per location type represent the overall distribution in the complete dataset. The SMOTE approach helped identify the minor classes (compare table 2 to table 3). At the same time, it reduced the success rate of the major classes. Due to the high number of major classes, this led to an overall decrease in accuracy. Therefore, the unbalanced dataset was used for further testing, as it resulted in a higher overall success rate.

To simulate the learning process of the ANN (more data being generated by the user over time), the data was randomly divided into 10 parts, each subsequent part containing the data from the previous in addition to the next set of items (t_length*0.1, t_length*0.2, ... , t_length*1). In order to test the performance for each individual, the test data was run against each of the 10 training sets. As the randomisation during the test data retrieval

a	b	c	d	e	f	g	<– classified as
527	4	0	2	19	2	0	a = home
8	12	0	0	0	0	0	b = food
40	0	0	1	1	0	0	c = shopping
11	0	0	1	3	0	0	d = transport
25	0	0	0	30	0	0	e = work
8	0	0	0	4	0	0	f = doctor
16	0	0	0	0	0	0	g = other_home

Table 3: Extract from the unbalanced example evaluation.

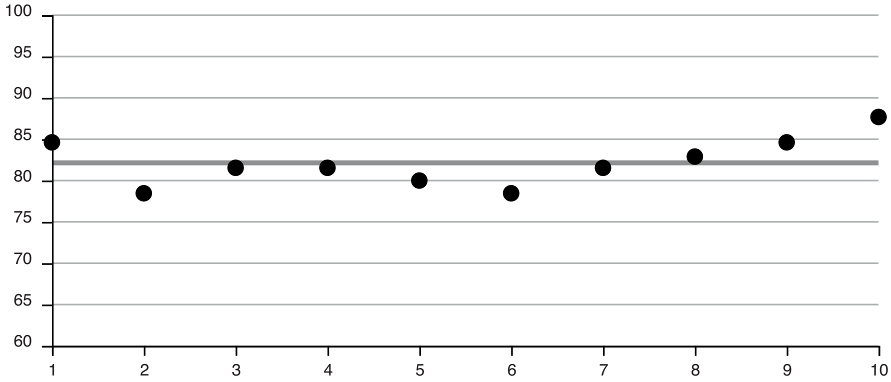


Figure 31: Performance development over n-month of training data. x represents the number trained month and y the accuracy. The thick line indicates the average performance.

changes the composition of the training and test data, each individual was tested 20 times to generate an average. The success rate came down to an average of 82.7% (median: 82%, max: 91%, min:72%). Although, as more data was introduced for training, a growth of success was expected over time, the growth was not significant and only varied between 78.46% and 87.69% (see graph 31). In order to see the effect of the minor classes in the dataset, an additional test was performed only for the major classes. Using only classes with at least 10 trips, the ANN reached up to 93.22% and using only home and work (the most common classes) even 94.74%. After all, the imbalance of the data is still not sufficiently solved and needs further attention. To again show the superiority of the ANN, the approach was tested against other machine learning techniques using the WEKA framework (see table 4).

6.3.4 *Predicting Duration of Location Events*

Building upon the results of the location type classification and prediction, this second approach takes the reverse approach and predicts the duration

Technique	Correctly Classified Instances
Random Forest	83.1%
Multi-Layer Perceptron (ANN 5-10-20)	81.69%
Multi-Layer Perceptron (ANN 5-10)	80.28%
Multi-Layer Perceptron (ANN 5)	80.28%
Logistic Regression	83.1%
TensorFlow ANN	87.69%

Table 4: Comparing the performance of the user-specific ANN against other machine learning techniques.

Minutes	0-15	16-80	81-255	256-624	625-1295	1296 - n
Hours	0-0.25	0.27-1.33	1.35-4.25	4.27-10.4	10.42-21.58	21.6 - n
Continuous Class	0-1.9	2-2.9	3-3.9	4-4.9	5-5.9	drop
Discrete Class	1	2	3	4	5	drop

Table 5: Transformation of duration.

of a stay at a location based on the time of arrival, the day of the week, and the location type. An exploratory data analysis revealed that the duration parameter included numerous outliers. Those outliers range between 24 and sometimes up to more than 72 hours. Those outliers are likely to be generated if the app crashes or is turned off. At the moment the app resumes, the missing time is categorised as a location, leading to these long periods. To counter this problem, the duration was transformed (see equation and table 5).

$$\mathit{trans\_duration} = \mathit{duration}^{0.25} \tag{8}$$

Durations for longer than 21.6 hours were removed from the training data. Based on the revised training data, the same neural network from the previous section was used to build a prediction system. The concept of continuous classes worked best in regards to the accuracy of predictions. The ANN in this case only achieved the second best results after the random forest approach, which is a similar result to the experiments above. The mean absolute error of the predictions was 0.6994 (Random Forest 0.61), which means that the predictions were on average not more off than one class. As in the previous cases, a comparison to other machine learning techniques was performed (see table 6).

ANN	
Correlation coefficient	0.6009
Mean absolute error	0.6994
Root mean squared error	1.0324
Relative absolute error	65.1349%
Root relative squared error	82.3324%
Random Forest	
Correlation coefficient	0.7114
Mean absolute error	0.6197
Root mean squared error	0.9119
Relative absolute error	57.7105%
Root relative squared error	72.7269%
Linear Regression	
Correlation coefficient	0.5931
Mean absolute error	0.7804
Root mean squared error	1.0212
Relative absolute error	72.6716%
Root relative squared error	81.4448%

Table 6: Comparison of machine learning approaches on the location event duration dataset.

6.3.5 *Conclusion*

The evaluation of the two competing ANNs showed that the user-specific algorithm outperforms the generic approach that is based on the time use survey data. We will now strengthen the argument for the implementation of the user-specific algorithm. The test design for the user-specific ANN anticipated a gradual learning process, which means that as more data beomes available, the accuracy in turn improves. This was only true for a very small increase, which was not significant. The imbalanced nature of the data presents an issue for the modelling approach. A solution to overcome this could be the combination of the synthetically balanced algorithm, which performed better on the under-represented classes and the imbalanced algorithm, which performed better on the over-represented classes. Other extensions of the approaches described above could include the incorporation of additional parameters e.g. activity at a location, previous location or even external influences like weather data. Another approach, which was not explored in this thesis, is the application of unsupervised machine learning for data analysis. While unsupervised training will not improve the identification of location types, it could help identify common temporal patterns across location types (see figure 33). As an example for unsupervised classification, the *canopy* algorithm was applied to the location event duration dataset (McCallum et al., 2000). *Canopy* specialises in unsupervised cases, as it does not require a predefined number of clusters, other than for example, k-Means-Clustering (Lloyd, 1982). The clustering in the example case in figure 33 identified overlaps and distinctions between classes, for example, a difference between weekend and weekday patterns.

To conclude, the supervised approach described in this section can be used to identify similar location types based on their temporal usage patterns. Those locations, in combination with spatial behaviour information, can be used to identify changes in the user's behaviour among other things. As an example, a user who switches his or her job will show up as a new location with the location type work assigned to it, if the user exhibits the same temporal patterns as in his or her previous job. Such events could trigger location recommendation processes that are discussed in the next chapter.

The process developed throughout this chapter is linked to the conceptual model's *coupling constraints*, as it applies the user's location-specific temporal behaviour in order to derive the constraints specific to a location.

Having identified and classified locations within the spatio-temporal datasets, the next section describes the process of building the network from the trips between those locations.

6.4 FROM LOCATIONS TO THE NETWORK

In regards to the user's spatial behavioural network, the preceding section described how to acquire information on the vertices (locations). In this next

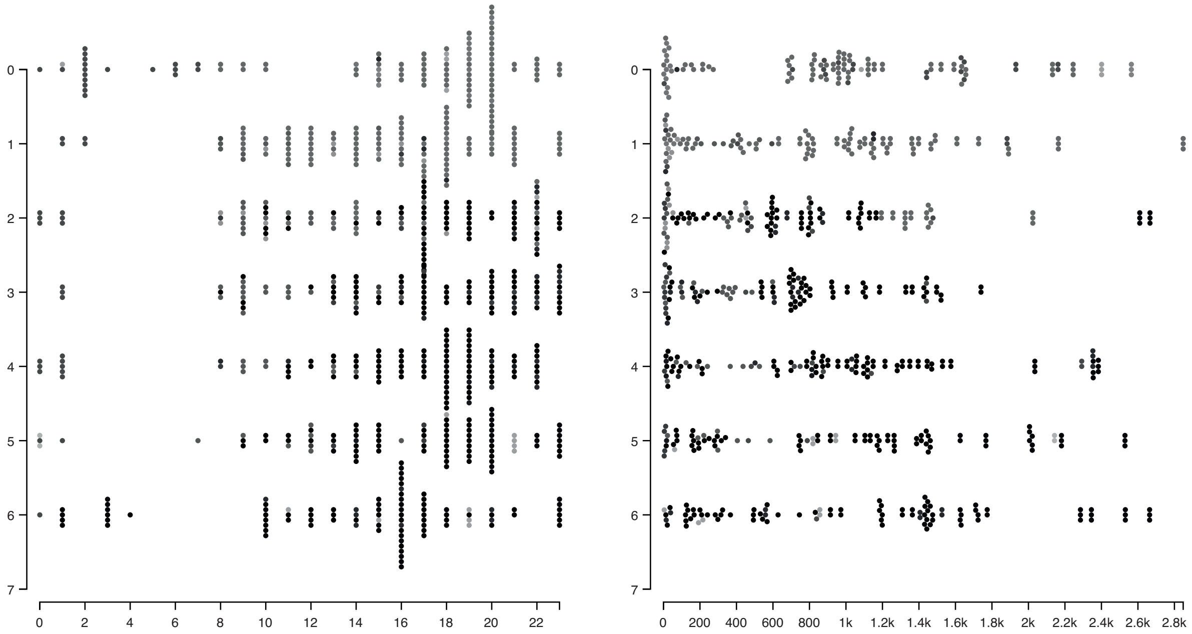


Figure 32: The y-axis shows days of the week, the x-axis on the left the starting point of an event on a 24 scale, the x-axis on the right the duration of the event. Both show clear cluster-distinctions between the days 0 and 1 and the rest of the week.

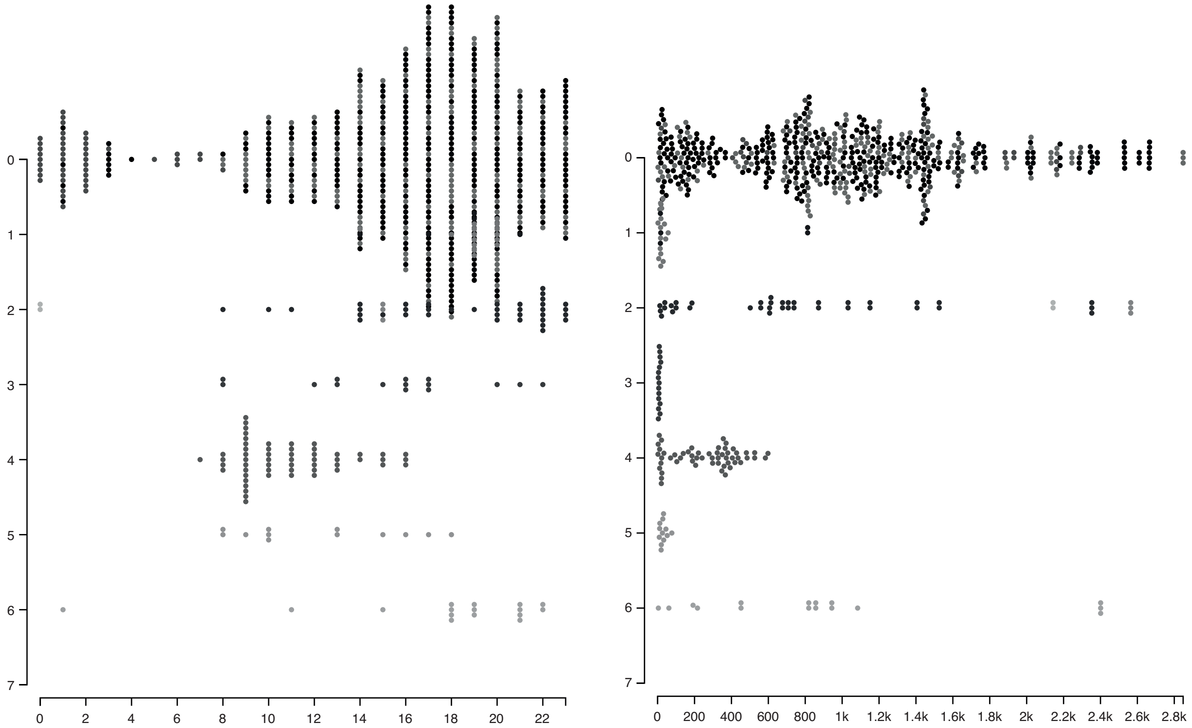


Figure 33: The y-axis represents the location types, the x-axis are the same as in the figure above. The left graphic shows that the location home shares clusters with all other locations (0), as it exhibits a broad variety of uses. The right graphic shows that the categories doctor, transport, and food (1,3,5) have similarly short durations and belong accordingly to similar clusters.

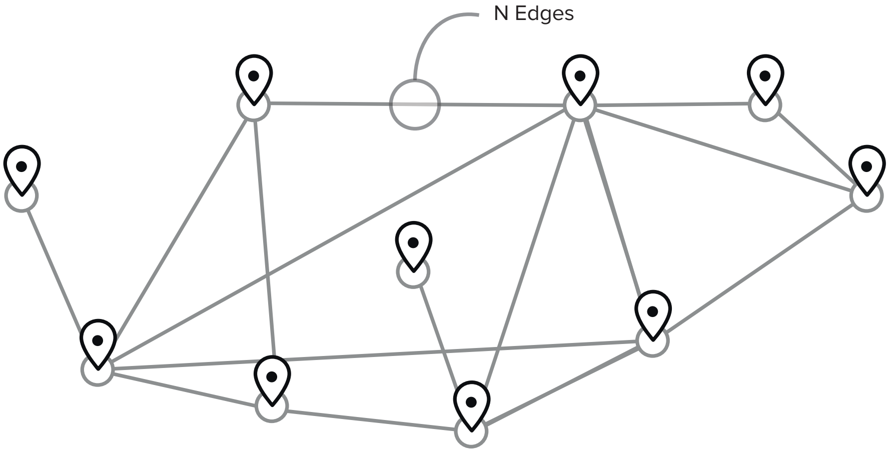


Figure 34: Network of locations, each edge in the visualisation represents one or multiple edges.

step, those vertices are connected through edges, which are the user's trips between those locations. The conceptual model's *capability constraints* are thus further explored by taking spatial and temporal patterns into account.

6.4.1 *Building the Network*

The section above on spatio-temporal data preprocessing detailed how the input data from the device is preprocessed and stored in a relational database. For each connection between location A and B from the location table, we have trips $\overline{AB}$ and $\overline{BA}$ in the trip table. Two locations can be connected by zero, one, or multiple trips. Those trips consist of n (at least one) trip segments (TS). A trip is either divided into segments by multiple activity types (e.g. A > walking + bike + train + walking > B) or by gaps during the tracking process (e.g. A > walking ? walking > B). This process results in a network of locations (see figure 34).

In order to improve efficiency for further analysis, a clustering is applied to the edges to reduce the complexity of the network. In a first step, trips with the same start and end location are clustered into one trip group. Therefore, a new table is created that holds the groups and references the trips. The result of this initial clustering phase is a network with a maximum of two directed edges per vertex-pair. Directed means either $\overline{AB}$ or $\overline{BA}$. Each edge is weighted by the amount of trips it represents ($edge\_weight = count(\overline{AB})$), in order to account for the clustering during the following network analysis (see figure 35). In network analysis, the weight of the edge is also sometimes referred to as strength.

The resulting network can then be used for network analysis. As discussed in the previous section, the network can for example be used to analyse the importance of vertices through centrality analysis. This provides the model with insight into the importance of nodes with respect to their connectivity

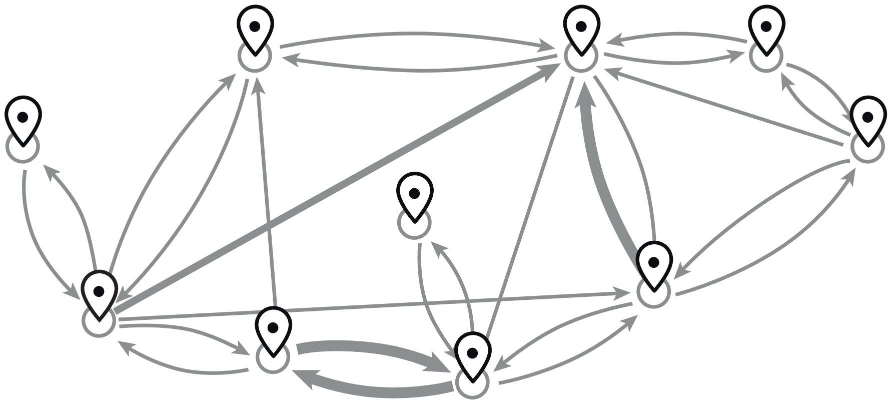


Figure 35: Network of locations, each edge represents a group of edges in the same direction, the weight (number of edges) is visualised through the width of the lines.

to the overall net (see figure 37). The results from the network analysis can be used to further enrich the location metadata as well as to inform the subsequent route predictions. As mentioned before, the imbalanced nature of the data also presents a challenge to the network analysis (see figure 36).

6.4.2 *Predicting Movement in the Network I*

Continuing the theme of predictions from the previous section, the remaining paragraphs of this section concentrate on using the above-generated network for the prediction of the user's spatio-temporal behaviour. While centrality and similar measures provide insights into the network structure, most of them ignore the spatiality and temporality of the network, which are essential for the prediction of the user's behaviour. Therefore, the subsequent steps incorporate the spatiality into the prediction process. The general approach is inspired by *hidden markov models* for finding patterns in trajectory datasets (Jeung et al., 2007), which try to identify reoccurring sequential patterns of spatio-temporal events, as well as the work by Tiakas et al. (Tiakas et al., 2009) and Chen et al. (Chen et al., 2013). Due to the inaccuracy of the dataset, an exact implementation of neither hidden markov chains nor Tiakas' or Chen's method was possible and thus themethods had to be adjusted. By following the general ideas of the hidden markov models, the paragraphs that follow attempt to predict the user's movement in the network based on the user's historic trajectory data.

In a first step, the edges of the network are enriched with a temporal dimension which depicts the occurrence of trips on each edge (see figure 38).

The basis for this second prediction or rather classification approach is again ANNs, as discussed in section on location type detection. The goal of this prediction is to determine where a user is travelling (output layer) by taking into account the start location as well as temporal features (input layer). During the tests, the performance in predicting the destination purely

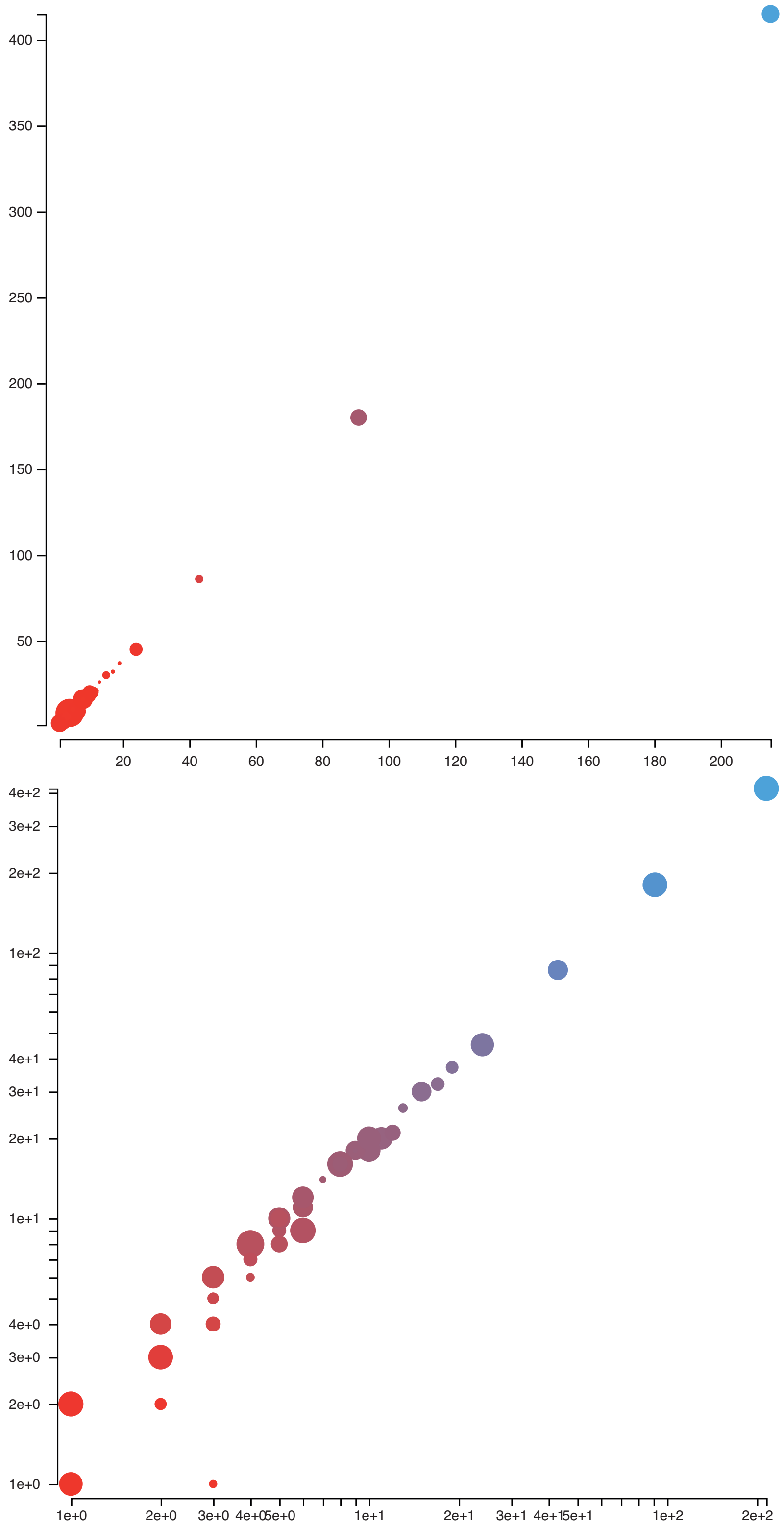


Figure 36: The x-axis in both graphs represents the number of visits to a location. The y-axis shows the number of edges connected to a location. The radius of each circle indicates the average time spent at each location. The colour highlights the degree of centrality. The upper graph is built on a linear scale and the lower uses a linear scale. The difference between the graphs highlights the problem of this imbalanced nature.

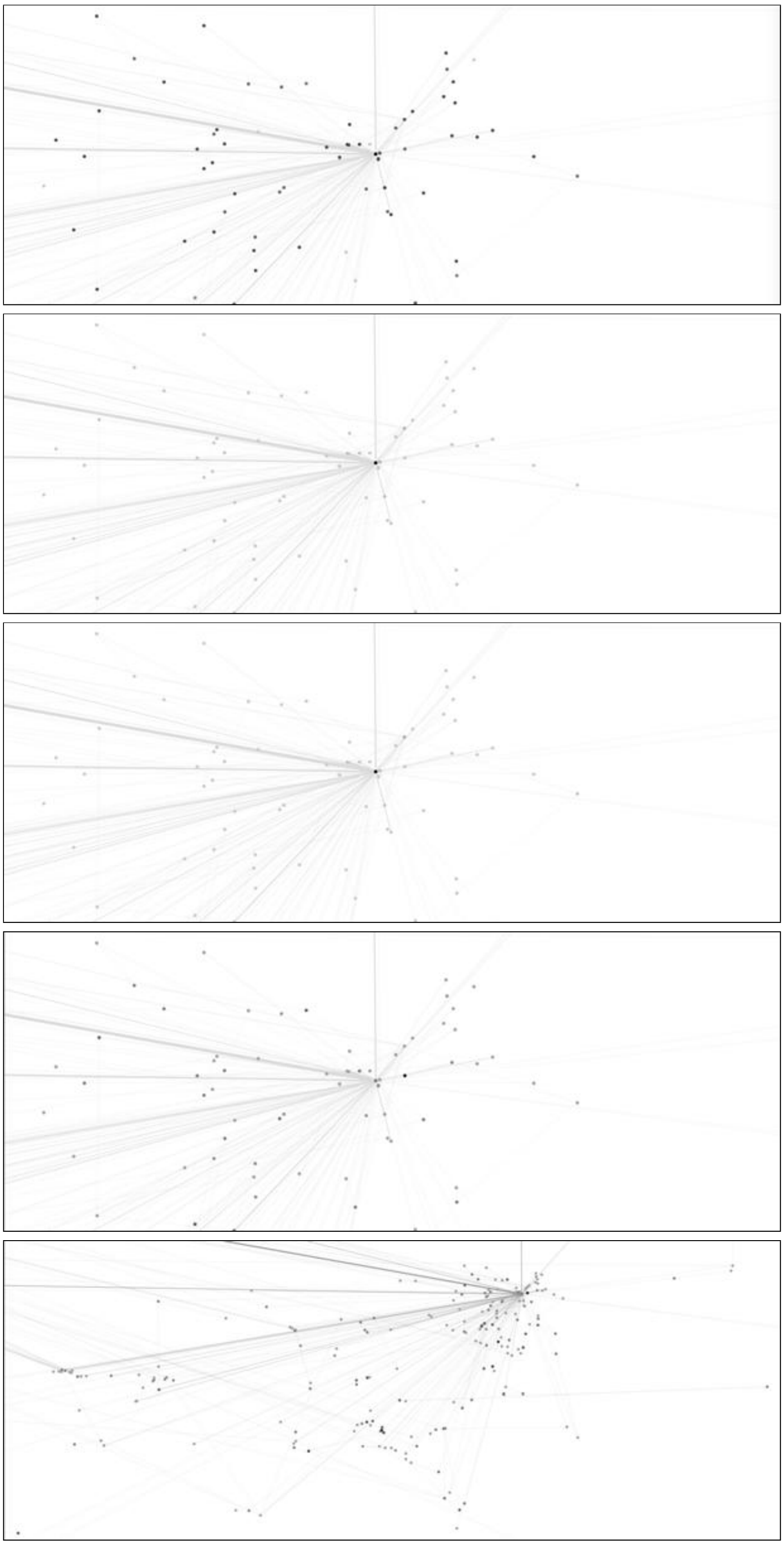


Figure 37: The visualisation shows the author's spatial network for the metropolitan area of Berlin. The colour of the vertices is derived from their (top to bottom) closeness centrality, degree centrality, between centrality, and page rank (in 2 zoom levels) (black = strong to light grey = weak).

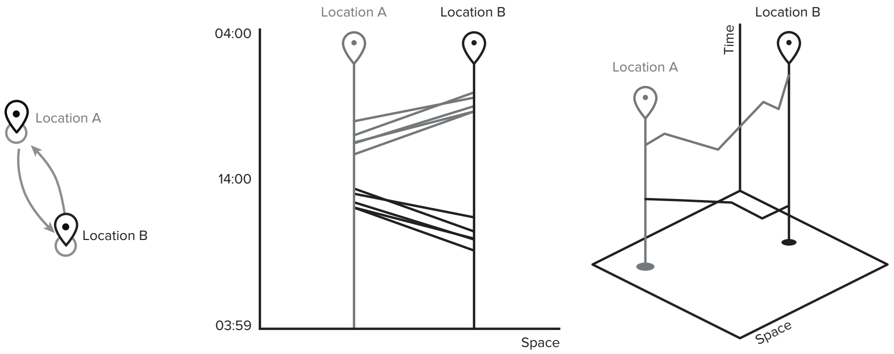


Figure 38: Visual explanation of temporal enrichment of edges (from left to right): 1) directed edge bundles 2) bundles split up by their temporal distribution between the two vertices 3) space-time cube of two exemplary edges between the two vertices.

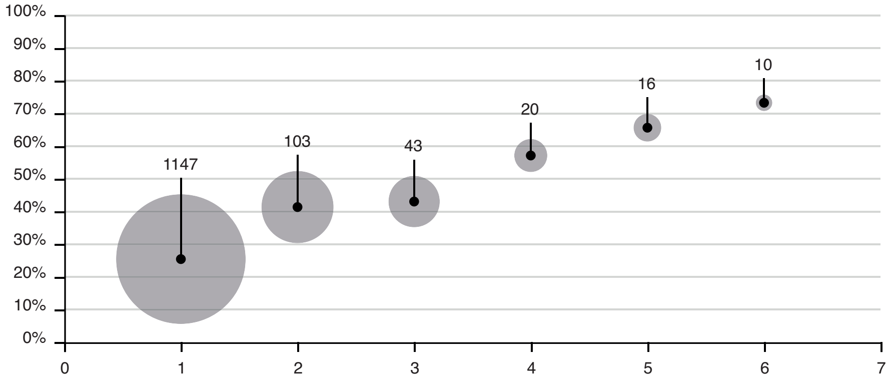


Figure 39: Prediction accuracy increase as data variation decreases. X-axis shows the threshold of minimum edge-counts that make it into the training set and the y-axis represents the accuracy. Size of the points represents the number of trip-groups in the training data using a logarithmic scale.

based on the start location and temporal information was only able to reach an accuracy of about 30%. An exploration of the data highlighted that the majority of trips only occurred once in the dataset. As soon as a threshold is implemented, limiting the trips included in the training data to trips occurring at least n-times, the prediction rate increases (see figure 39)[23]. At the same time, this obviously also decreases the variety of potential destinations that the algorithm can predict. To overcome these limitations, the following sections work to improve this initial prediction approach.

[23] Similar to the previous cases, the most successful attempt was repeated with random forest and linear regression to highlight the difference between those techniques and the ANN (see table 7).

Artificial Neural Network	73.27%
Random Forest	69.31%
Linear Regression	70.30%

Table 7: Correctly classified instances for the dataset with at least 6 edges per network.

a	b	c	d	e	f	<– classified as	
43	1	3	2	4	0	a = pl_96792719	81.13%
1	2	3	0	3	0	b = pl_2477990	22.22%
0	3	3	0	2	0	c = pl_46793809	37.5%
1	0	0	11	0	0	d = pl_171721792	91.67%
2	0	2	0	5	0	e = pl_188291457	55.56%
0	0	0	0	0	10	f = pl_96792910	100%

Table 8: Correctly classified instances for the dataset with at least 6 instances per edge group.

6.4.3 *Refining the Network*

The approaches applied above only accounts for the temporal dimension as well as the origin and destination of a trip, but they ignore the actual spatial trajectory between the two locations. Therefore, in the following, the spatial trajectories between locations will be included in the calculation. This will allow the technique to include an estimation of the actual route that the user might take to a destination, in the calculations and lead to better predictions. In order to discuss and illustrate the challenge of this analysis, the dataset is reduced to three locations and trips between those. The underlying approach will in the end be scaled to the overall network. While the abstracted network allows discussion of general relationships between locations as well as trips between locations, the underlying spatial trajectories are ignored. In the abstracted network the trip $\overline{AB}$ can be clearly distinguished from the trip $\overline{AC}$, whereas this proves difficult in the actual trajectory data, as trajectory segments overlap (see figure 40).

The following steps will inform a method that incorporates the abstracted network as well as the underlying spatial trajectories. The first step towards this is to further cluster the underlying trajectories. The trips are clustered by direction ($\overline{AB}$ vs $\overline{BA}$) as well as their inherent activity types (e.g. walking, transport, car, bike). In order to differentiate between trips with varying activities, temporary transition locations (TL) are created for the spatial position at which the activity type switches (see figure 41). Those transitional locations are stored in a new table for transition locations. Based on the transit locations and the new trips between those and the end and start location, new clusters are created which are made up of the same sequence

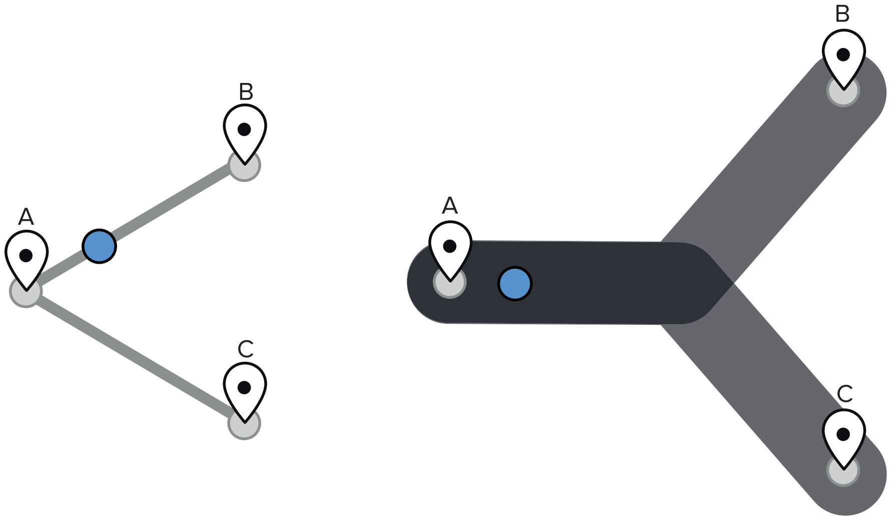


Figure 40: Visualising the problem between a simplified trip network and the actual underlying spatial network between three given locations A, B, and C. The blue dot represents the user's current position.

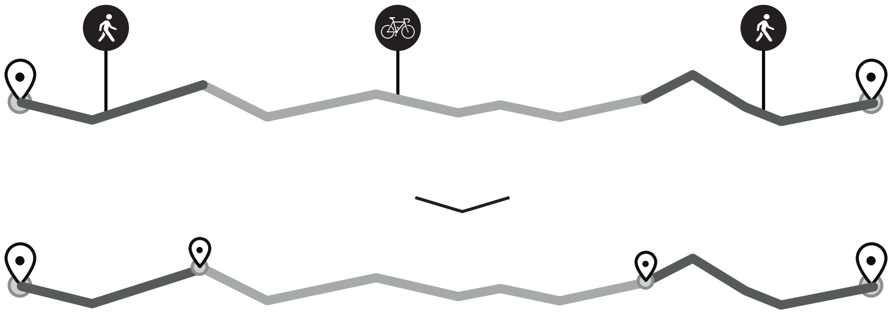


Figure 41: Creating temporary locations at activity intersections.

of activities and transition locations. Those clusters are stored again in a new table referencing the split-up trips. In a high precision dataset, the next step would include a map-matching procedure or a trajectory clustering (see Lou et al., 2009 for map matching and Kharrat et al., 2008 for trajectory clustering), followed by an analysis of finding common trajectory segments between trips (Mao et al., 2017). As the resolution of the dataset in question is not high enough, this is not possible. Therefore, an intermediary solution is integrated: *trip corridors*.

Trip corridors aggregate uni-directional trips between two locations. As a common trajectory segment analysis is not possible, each trajectory is extended through a buffer (100 meters) and then overlaps of at least 60 percent[24] are required in order for a trajectory to be grouped together. The resulting clusters are combined and extended with the additional buffer to create a polygon around the trajectories (see figure 42 and figure 43). Those cor-

[24] The number of 60 percent was calculated by trial and error, with the goal of achieving distinctive but not too sparse clusters.

ridors can then be used to detect which trip the user *might* embark on by comparing the user's current trajectory with the corridors in the database (see query and figure 42). Due to the inaccuracy of the dataset, an exact WITHIN or CONTAIN query cannot be used, therefore the overlap between the current trajectory and corridors is calculated. Matches with more than 70% overlaps are included in the result set.

Listing 1: Querrying corridor matches

```
#To speed up the query, the users current trajectory is stored in a
    temporary table called temp_query

SELECT
  temp_query.id AS temp_id, corridors.id, corridors.from_id, corridors.
      to_id, corridors.from_cluster, corridors.to_cluster, COUNT(*) AS
      group_count, group_concat(trips.start_10_min) AS t_start_10_min,
      group_concat(trips.end_10_min) AS t_end_10_min, group_concat(trips
      .day_of_week) AS t_day_of_week, Area(Transform(Intersection(
      temp_query.the_geom, corridors.the_geom), 3857)) AS inter_area,
      Area(Transform(temp_query.the_geom, 3857)) AS trip_area
FROM
  temp_query,
  corridors
  LEFT JOIN corridor_trips ON corridors.id = corridor_trips.corridor_id
  LEFT JOIN trips ON corridor_trips.trip_id = trips.id
WHERE
  Intersects(temp_query.the_geom, corridors.the_geom) AND
  (inter_area / trip_area) > 0.7
GROUP BY
  temp_query.id, corridors.id
  ORDER BY temp_query.id, corridors.id ASC
```

6.4.4 *Predicting Movement in the Network II*

For further refining of the matched trajectories (if multiple), an approach like ANNs would not be efficient. This is the case not only due to processing requirements, but also due to a lack of data. Therefore, the temporal information on each corridor is used to run a KNN on the user's current temporal information and the matched corridors. As a result, this calculates the corridor with the highest probability. The prediction, as outlined above, is performed in intervals (2 minutes apart). While the user moves through space, the trajectory extends and, thereby, the accuracy is increased with every iteration (see figure 45). Due to the inaccuracy of the trajectories, at a certain step towards the end of the trajectory, some of them will not be able to match a correct corridor. This is due to the fact that too main points are outside of the corridors and therefore a big enough intersection cannot be computed

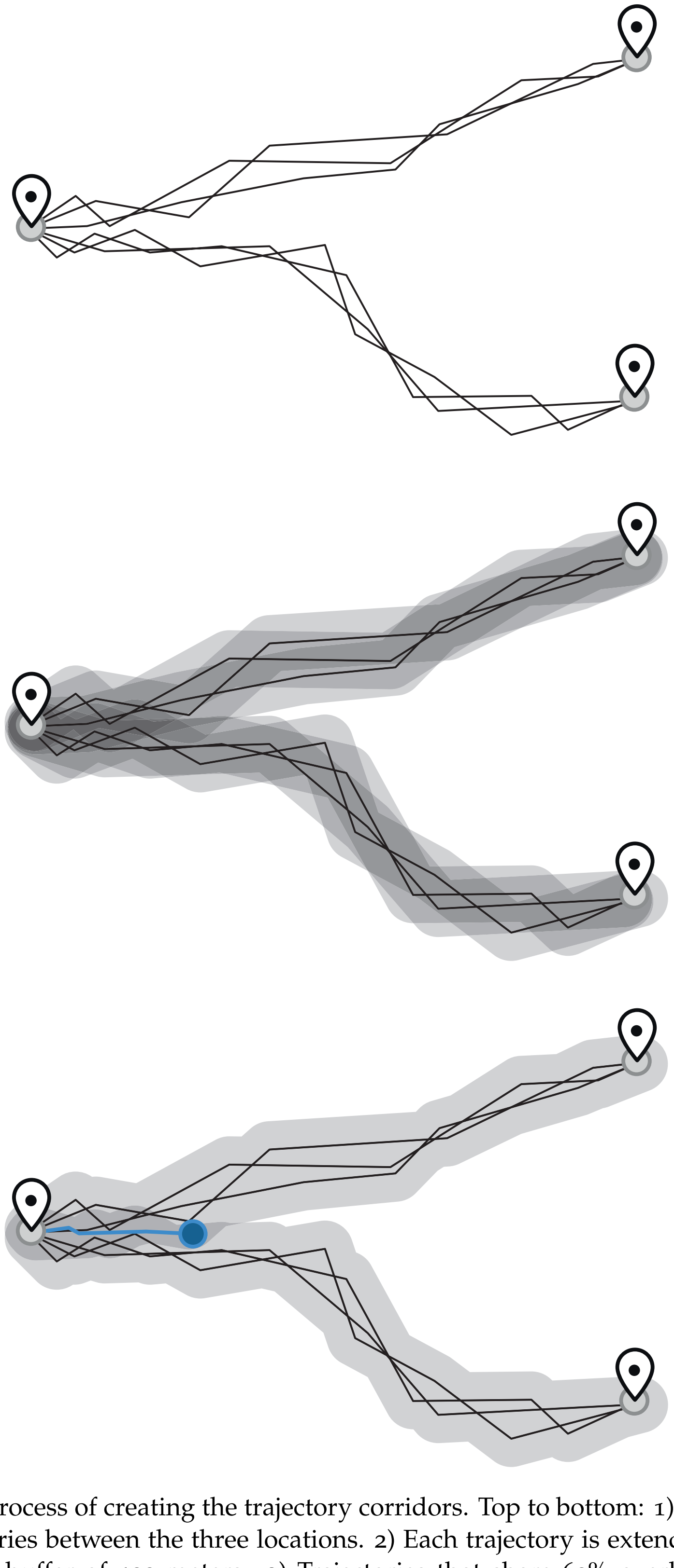


Figure 42: Process of creating the trajectory corridors. Top to bottom: 1) Raw trajectories between the three locations. 2) Each trajectory is extended through a buffer of 100 meters. 3) Trajectories that share 60% overlap are combined into one corridor, which can then be used to intersect with the user's current location and trajectory (blue).

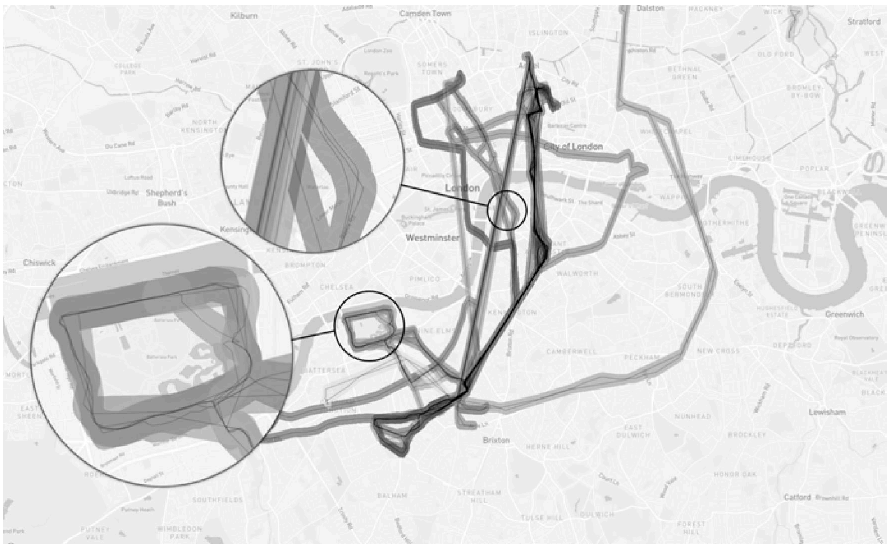


Figure 43: Corridor examples from the author's *Moves* dataset. The two highlighted examples (circles) show how minor differences between trajectories (GPS inaccuracy) are ignored due to the tolerance.

(see figure 44). As a result, the optimal prediction rate is reached towards the end of the process with a value of 56%. The overall likeliness of finding a match at all is at the highest 43% throughout the testing cycle. he correctly classified cases reach a value of 50% (mean: 46.7%, median:50%) towards the end of the journey. But even at the beginning of the journey, after 10% of its length, this number already reaches 40%. If the cases are included where the correct destination was in second and third place of the prediction, the rate rises to 85% (min: 70%, max: 85%, mean:77.5%, median: 80%).

To summarise, the previous steps have created a spatial network that consists of locations (vertices) and trips between locations (edges). These are refined into trip groups in which each trip is made up of trip segments that are combined into trip corridors. Those elements allow us to predict movement within the existing network.

In order for the clustering and prediction approach to work, one precondition must be satisfied for a trip to be considered of the same group: start and end location must be the same. In the previously described approach, so-called temporary transit locations were created (switches between two activity types, e.g. walking mode to car mode). As was revealed by an exploratory data analysis where the time threshold of the location identification was changed, there are two types of natural transit locations. First of all, there are actual transit locations that are detected as locations because the user stays there long enough (location time threshold) for it to be counted as a location (e.g. waiting for a connecting train). Secondly, users sometimes do not immediately proceed to their final destination but visit locations near start or stop locations (e.g. running quick errands near a location). Those two cases present challenges for the prediction algorithm. In order to over-

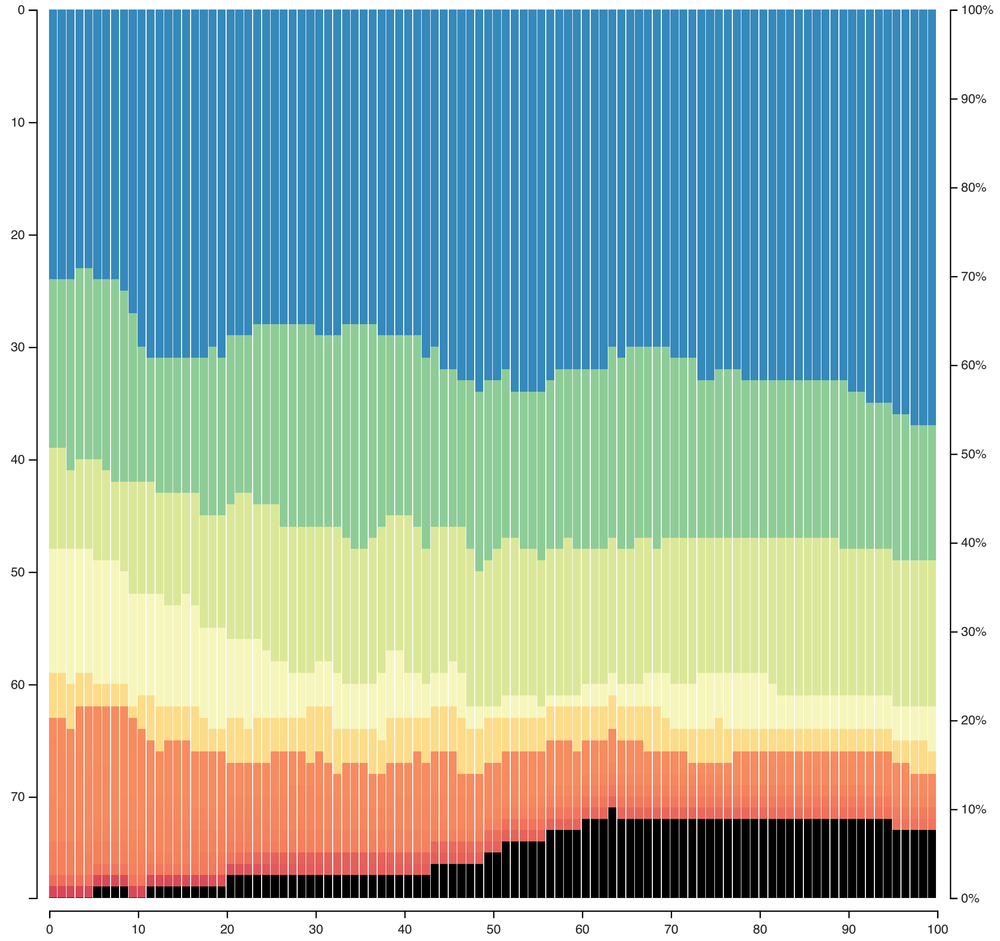


Figure 44: The iterations for each test trajectory (of one test subject) are mapped to 1-100 (x-axis). For each step in this iteration the knn is performed. The stacked bar-chart shows the result of the knn. Blue highlights the number of correct classified instances. Dark green is used when the correct location was second in the knn-results, with bright green third, yellow forth and orange fifth. The red gradient indicates matches beyond rank five. Black visualises cases in which no resulting corridor matched the expected outcome. The y-axis thereby represents the percentage of classified and unclassified cases.

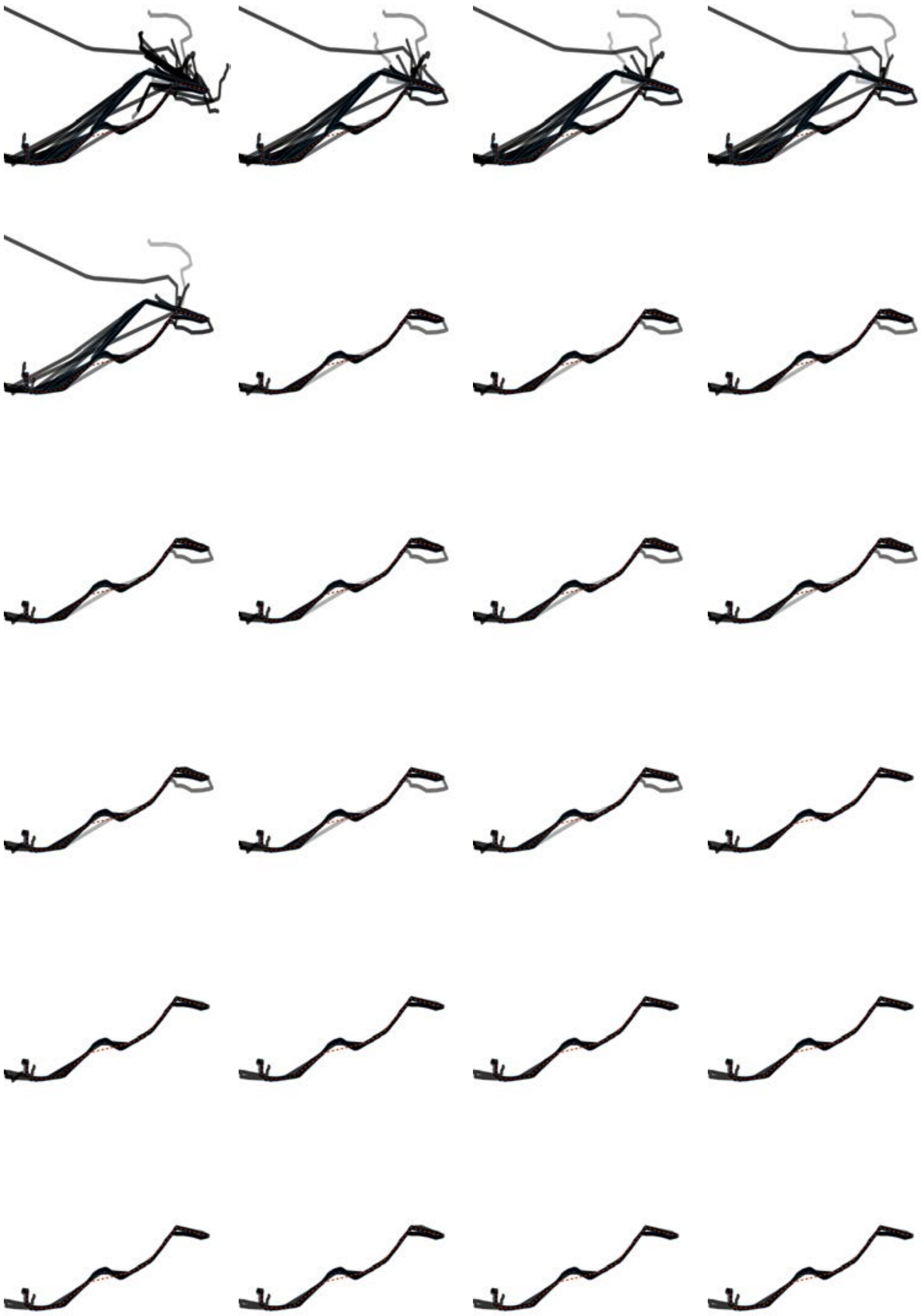


Figure 45: Example of iterations from a corridor trip prediction (only every second iteration is shown). The darker the polygon, the higher the probability calculated by the KNN algorithm. The longer the trajectory becomes, the less viable intersections are produced.

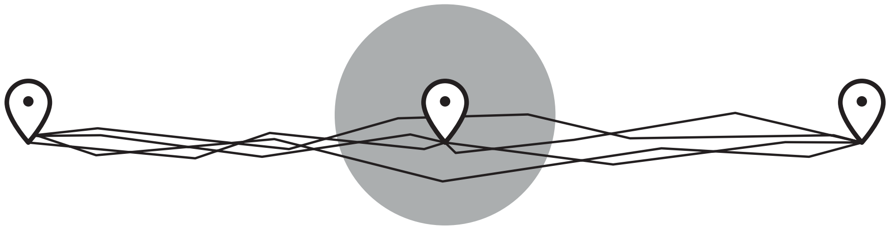


Figure 46: Identifying transit locations through intersections with similar trips.

come those, two extensions of the approach are presented: transit locations and origin / destination clusters.

6.4.5 *Transit Locations*

The destination prediction from the previous section only predicts the trip to the next location, but in some cases this might not be the actual final destination. To overcome this problem and improve the network, the approach of *Transit Locations* is proposed. Transit locations are locations along a user's trajectory, where the user spends enough time for it to be accounted as a location, but not being the actual final destination of the overall trip. Two identification approaches are combined to recognise transit locations. First a location is intersected with historic trajectory data in order to determine if there are trips with the same start and end location, that pass through that location without an intermediate stop. To do so, a simple buffer is used to collect nearby trajectories. The result is then filtered further to only include trajectories that start from the user's current trajectory origin (see figure 46).

Listing 2: Identifying possible transit locations

```
SELECT COUNT(*) FROM trips WHERE loc_1 = location_1 AND loc_2 =
    location_2 AND ST_Intersects(geom, ST_Buffer(transit_location, 2*
    location_spatial_buffer))
```

In addition to this intersection technique, the reverse location identification approach from the first section of this chapter is used to predict the duration of the stay at the location. If the stay is shorter than:

$$2 \text{ x location\_time\_threshold} \tag{9}$$

it is also classified as a potential transit location.

If a location is identified as a potential transit location, the destination prediction is executed, using the transit location as a starting point. By doing so, a chain of locations can be identified that leads to the potential final destination (see figure 47). One exception was identified while applying this approach: locations that the user only visits shortly before travelling directly back to the origin location. If it is a direct trip back to the origin location,

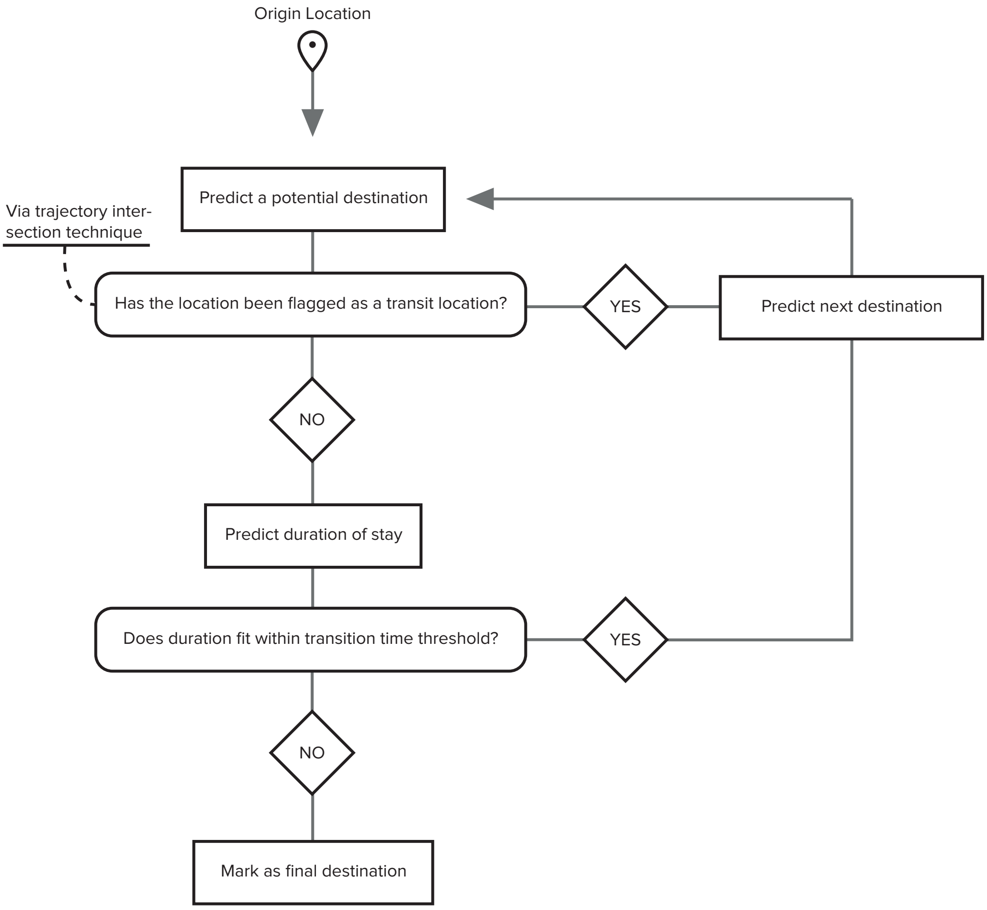

Origin Location
Predict a potential destination
Via trajectory intersection technique
Has the location been flagged as a transit location?
YES
Predict next destination
NO
Predict duration of stay
Does duration fit within transition time threshold?
YES
NO
Mark as final destination

Figure 47: Process of building a sequence of potential destination location predictions.

the location is not considered a transit location, since it is assumed that the user went to this location intentionally. Consequently, these locations are not reclassified as transit locations. A problem with this approach is that the uncertainty with each iteration stacks up. Therefore, the resulting predictions have to be handled carefully.

6.4.6 *Origin / Destination Clusters*

By visualising the resulting spatial network, another insight became clear. While there are many reoccurring trips between location pairs, another pattern are groups of nearby locations that are connected to another group of nearby locations in another area (see figure 48).

The problem illustrated in figure 48 is not yet covered by the outlined prediction approaches, as each trip would be identified individually but not as part of a common cluster. While the prediction approach still works without taking this fact into account, further clustering and refining of the network will increase the performance of the predictions. This is due to

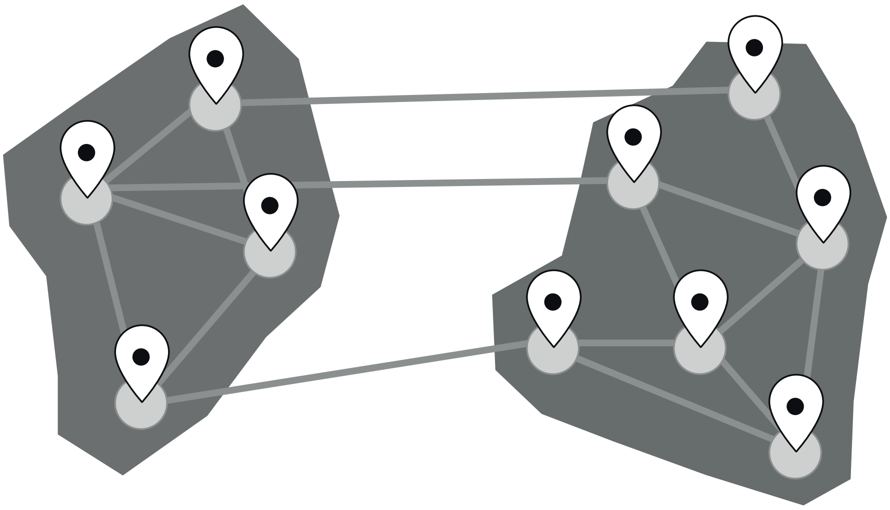


Figure 48: Clusters of locations that are connected with one another but also across the clusters.

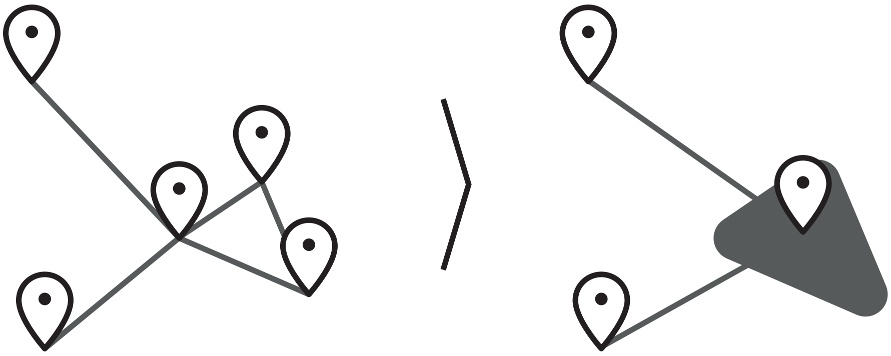


Figure 49: Graph generalisation through clustering, while maintaining graph connections.

the fact that the refined predictions only need to forecast trips to a cluster instead of an exact location (a discussion of a comparison of both approaches against one another will be presented at the end of this section).

Consequently, the goal of this clustering approach is to group related locations or vertices in the network into clusters. This reduces the overall complexity of the network. In the domain of graphs and networks, this can be achieved through graph generalisation or graph clustering. Similar to cartographic generalisation it continuously reduces the complexity of the network. In this specific case it is required to maintain the overall network relationships. This means if we cluster two vertices, the edges of both vertices need to merge and be present in the new vertex, which excludes many general network generalisation or clustering techniques (see figure 49).

One possible method to apply to the network is the *Louvain method* for identifying communities in the network (Blondel et al., 2008). Blondel et al. describe the potential of their algorithm, as "[t]he identification of these

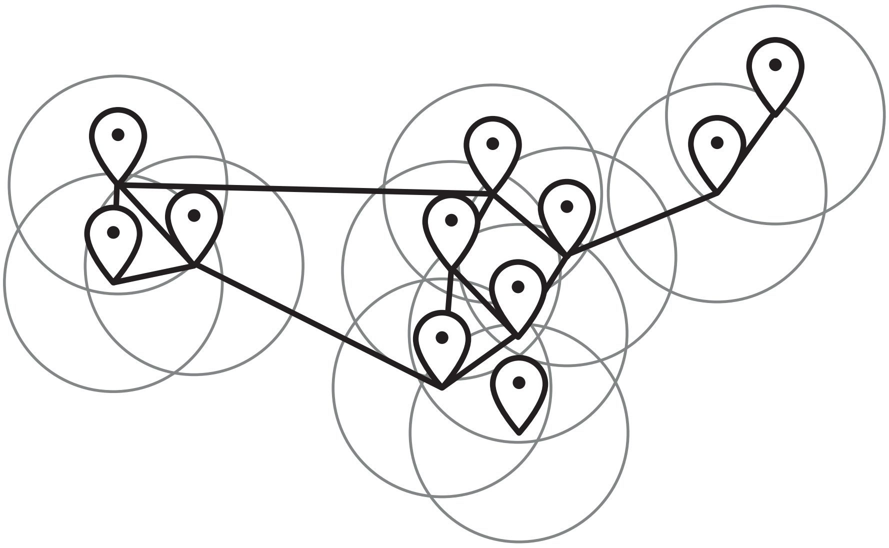


Figure 50: Distance-based *DBSCAN* algorithm with a network (connectivity) filter (location in the lower right is within the distance buffer, but not connected and therefore not considered for the cluster).

communities [, which] is of crucial importance as they may help to uncover a-priori unknown functional modules such as topics in information networks or cyber-communities in social networks." (Blondel et al., 2008, p. 2) The algorithm by Blondel et al. finds strong connected and, more importantly, interconnected sub-graphs in the overall graph by applying modularity optimisation. Thereby clusters (communities) are detected. While the results of the *Louvain method* and similar algorithms are of interest for exploring the transportational capabilities of the network, they ignore the spatiality of the network in question. In order to incorporate the spatiality of the network, the raw trajectories were used as an intermediary clustering technique. The technique is inspired by density clustering algorithms like *DBSCAN* and more hierarchical and connectivity-based clustering algorithms like the *Louvain method*. Similar to *DBSCAN*, the primary identification of locations belonging to the same cluster is the distance between *connected* locations. The difference of the method applied in this thesis and *DBSCAN* is the limitation of scanning for the distance between locations, only if they are also connected through a trajectory (see figure 50).

6.4.7 *Predicting Movement in the Network III*

Using the new cluster information, the corridors are updated and the prediction is repeated. As the algorithm now only needs to predict the correct cluster instead of the correct location, the performance increases compared to the previous attempt. The correctly classified cases at the end of the trip reach up to 90% (mean: 78.3%, median:75%), and even after only 10% of the journey this number is already at 68%. If also these cases are included where the correct destination was in second and third place of the prediction, the rate constantly remains at about 90% (min: 90%, max: 95%, mean:91.7%, median: 90%). At the same time, the number of cases that cannot be classified never reaches more than 5% (see as an example figure 51).

6.4.8 *Conclusion of Network Analysis*

Guided by the conceptual model which combined the network model with Hägerstrand's time-geography constraints, this chapter has used the processed spatio-temporal information to construct a user-specific network and predict the behaviour within that network. Through the improvements of corridors and clusters, the trip prediction reached up to 75% accuracy including minor classes. With this results, the primary requirement of analysing and predicting spatio-temporal behaviour is met. The imbalance and inaccuracy within the data proved to be challenging but could nonetheless be overcome by the proposed methods and techniques. Further general limitations and critiques will be discussed in the final chapter.

Future Works

The usage of ANNs in the area of aerial imagery classification has matured over the last few years. The processing and analysis of spatio-temporal behaviour through ANNs is still very much in flux. Promising research like the papers by Zhang et al. (Zhang et al., 2016b; Zhang et al., 2016a), will bring about more possibilities for prediction and classification of trajectories through ANNs. Furthermore, those approaches could be combined with more advanced implementations of the markov chain approaches in order to predict trip sequences.

While the idea of weighted learning was conceptually discussed when ANNs were introduced, the approach has not been implemented. Nevertheless, in a next step, weighted learning should be implemented in order to account for changes in the user's behaviour over time.

The focus of this section was put on predictions and classifications, but the underlying data and mechanics could also be used for creating navigational recommendations in a next step. If a user wants to travel from A to B, the algorithm could provide the route the user usually takes between those two locations. In this sense, the last section of this chapter uses the underlying data to derive further inferences on the user's behaviour to be used in location recommendation processes.

6.5 FROM LOCATIONS AND NETWORKS TO AREAS

Section 6.3 described a method for identifying and classifying locations through the user's spatio-temporal behaviour. Section 6.4 extended this to the movement between those locations. This last section focuses on the

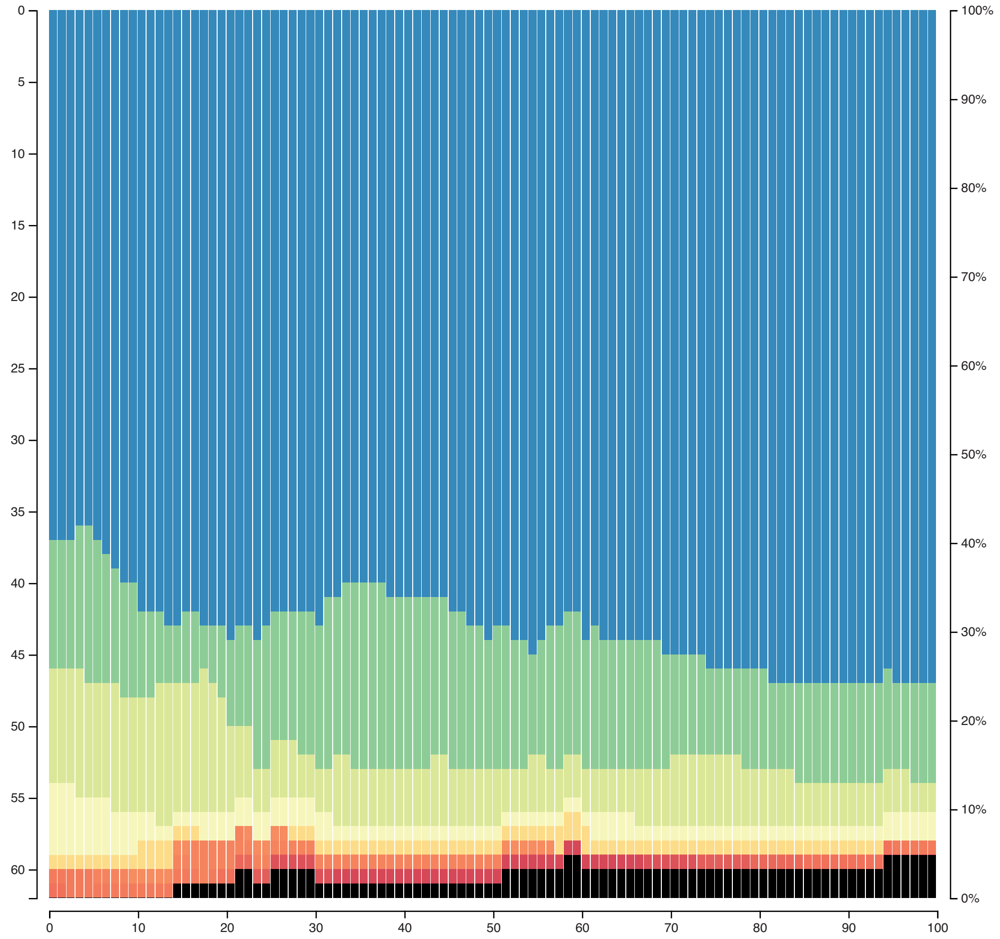


Figure 51: The iterations for each test trajectory (for one of the test subjects) are mapped to 1-100 (x-axis). The KNN is performed for each step in this iteration. The stacked bar chart shows the result of the KNN. Blue highlights the number of correct classified instances. Dark green when the correct location was second in the knn-results, bright green third, yellow forth and orange fifth. The red gradient indicates matches beyond rank five. Black visualises cases where no resulting corridor matched the expected outcome. The y-axis thereby represents the percentage of classified and unclassified cases.

second requirement, outlined in chapter 2: the inference of semantic and personal information. While there are numerous examples that use the underlying data to derive behavioural metadata, for example for general temporal patterns or location type preferences, this last section focuses again on two examples in the spatio-temporal domain. The two cases serve as examples for such inference techniques, which demonstrate the client-side capabilities. The two exemplary methods are *areal descriptors* and *Potential Memorisation Index (PMI)*.

6.5.1 *Areal Descriptors*

As cited throughout this section, these paragraphs strongly build upon this thesis' author's publication Meier, 2017, which introduced the concept of areal descriptors.
Through the network clustering approach, this thesis has proposed a method

for identifying connected neighbourhoods. Beyond their connectivity, the approaches above have not further analysed the inherent structure of those neighbourhoods. As already noted in the requirements, further insights on the areas occupied by the user are helpful for refining location recommendations and deriving inference-information. One such example that can be conducted on the location clusters are *areal descriptors* (Meier, 2017).

The fundamental concept behind this approach is the idea of connecting locations to their urban fabric. The location identification process from section 6.3 already enriches each location with a location type, which allows the system to categorise a location (e.g. restaurant). In regards to architecture and urban planning,

> "[...] it is a prevailing notion that a location is influenced by more than its own properties. Christopher Alexander describes this at many times in his 1977 book "A Pattern language" (Alexander et al., 1977), e.g. in his chapter on the mosaic of subcultures (page 42ff). The environment, the urban fabric in which a location is embedded, plays a crucial role in shaping the color, ambience, or rather atmosphere of a location, e.g. by the amount of nature it is surrounded by (Sullivan et al., 2004). One might think of restaurants in a picturesque part of town or a restaurant on the countryside. Following this line of thought, the land use, building types, natural features, the mix of locations, etc. of the surroundings affect the inherent locations. Especially the mix of locations, a feature requiring only a location dataset itself"
>
> Meier, 2017, p. 4

, presents a good starting point for enriching the location's metadata. Section 6.4 presented a method for clustering connected and nearby locations into what could be described as neighbourhoods. Those clusters present areas that the user occupies to visit certain location types. Applying the concept of *areal descriptors*, the locations within a cluster are analysed in regards to their location types as well as the temporal dimension of those location events. A resulting summary is stored alongside the cluster (see

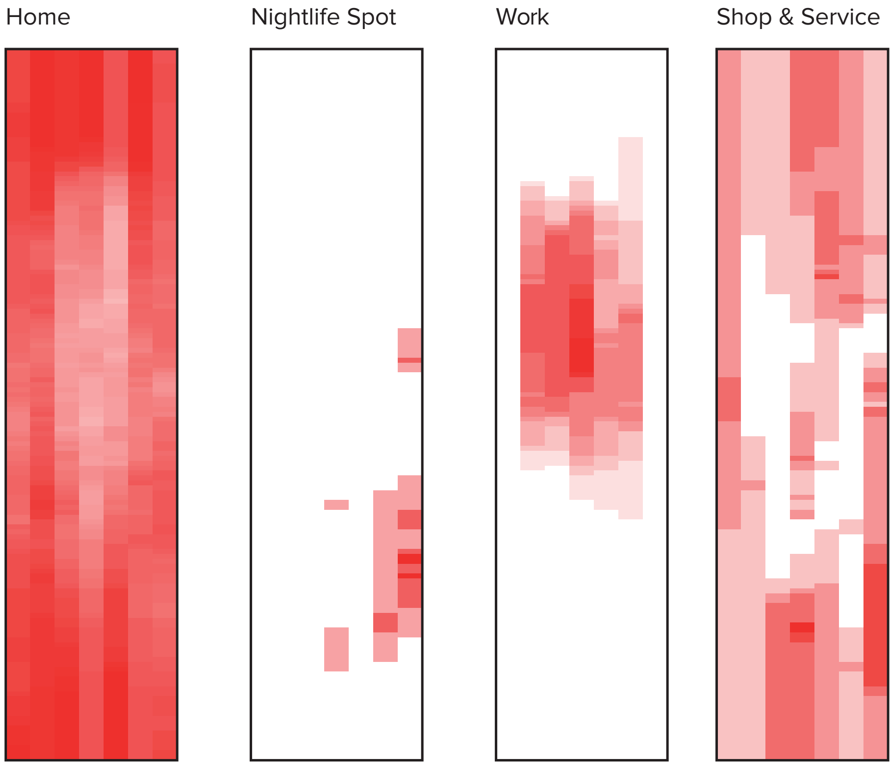


Figure 52: The four examples above show four location types and their temporal patterns clustered by days (Sunday to Saturday) and by hour (4am to 3am) in 10 minute intervals. The examples highlight that some locations exhibit very specific patterns. Home locations are for example most occupied during evening, night, early mornings, and the weekends, while work locations exhibit the exact opposite and nightlife locations are most visited during Fridays and Saturdays. The gradients nicely illustrate how temporal behaviours shift slightly and are not the exact same every day. Using those insights one can still predict the outcomes.

figure 52). This metadata can then, for example, be used for location recommendation (Example: A destination forecast predicts the user is travelling to cluster C, filtering the temporal summary of the cluster with the current time through KNN, the algorithm can suggest certain category types). Tests with the *Moves* test data showed that this approach achieved between 50 to 83% accuracy (median: 66%, mean: 66.8%) in correctly identifying location categories, including those where the correct match is only one off in the prediction ranking, it reaches even 61% to 95% (median: 84%, mean: 81.8%).

The *areal descriptors* made use of the user's identified and classified location data. The next approach shifts back to the trajectories and applies a novel approach for acquiring further user-specific insights on the spatial structures the user occupies.

6.5.2 *Potential Memorisation Index*

This section is adopted from Meier and Glinka, 2017. Most paragraphs have been slightly modified to fit the context of this thesis. The co-authors' (Katrin Glinka) main contribution was chapter 2.2, which was therefore excluded from this thesis.

While the previous approach relied on location data, the approach on *potential memorisation* shifts the focus back to the trajectories. The following approach addresses two issues: first, personal spatial relationships which are not present in the network structure (e.g. locations the users passes while travelling to a third location, which are therefore not connected by a trajectory). Second, it connects to the undertaking of improving location recommendation and the prediction of trips between locations, especially incorporating the trip's multimodality.

The construct of the mental map and the individual perception of space was introduced in the section on individual perspectives in chapter 2. The following section presents an algorithm, that highlights insights from the personal usage of space. To do so, the approach is based on the supposition that mental maps are influenced by many external stimuli, one of them being the way the users move through a city (Chorus and Timmermans, 2009; Mondschein et al., 2010). The users' movements change the way they experience the physical world and thus influence the shaping of mental maps. The presented approach uses the trip data discussed above and focuses on the mode of transport as one such experiential factor, which builds upon work by Mondschein et al., who reported that active or passive navigation influences the quality (richness of detail) of mental maps (Mondschein et al., 2010). Following this supposition, the algorithm attempts to calculate the personal level of *potential memorisation* or spatial knowledge of a certain area. This constitutes an additional metadata attribute to be used in a location recommendation algorithm. The calculated value is based on the user's trajectories and the mode of transport (activity type): from the highest value for walking, to a lower value for cycling, and finally to the lowest value for motorised transport. The term (potential) memorisation might at first sight seem ambiguous. Memorisation is often used in contexts of intentional learning as such, describing how information is moved from the working memory to the long-term memory. In this case, it rather describes the acquisition of knowledge of a certain area. Still, the repetitive interactions and the type of interactions with a certain area help us estimate the quality of spatial knowledge. Therefore, the term (spatial) memorisation is used. The previously discussed trajectories as well as their activity classification are analysed in order to gains insights from a single aspect of a person's mental map: the potential knowledge or rather memorisation of certain spatial areas.

$$\text{index} = \frac{\log(\text{times}_v\text{isited} + 1)}{0.1 * \text{transport}_m\text{ode}} \tag{10}$$

Figure 53: PMI, per transport mode (motorised transport = 1, cycling = 2 and walking = 3.

6.5.2.1 *Temporal Data Clustering & Network Analysis*

On the basis of the calculated locations and trips, the level of assumed spatial knowledge over certain areas is mapped by deriving a value for the potential level of memorisation based on the mode of transport. Mondschein et al. (Mondschein et al., 2010) did not develop a formula to calculate these levels, their findings rather indicate an order of memorisation potential for various transport modes. Their findings are used to develop a formula that translates visits of a certain part of town by a certain mode of transport into a *potential memorisation level* that can be used in algorithmic calculations. This formula serves only as a starting point for the research and has to be refined by more user studies as discussed in the end of this section. Based on the order introduced by Mondschein et al., a logarithmic curve that represents an exponential learning curve with a limit is used. At a certain level of knowledge, each additional spatial interaction will only increase the value of the PMI by a very small amount.

In order to test the approach, a 50-meter raster was created and then intersected with the trajectories, using a buffer of an additional 20 meters to overcome the precision issues of the trajectories, which resulted in a heat-map-like visualisation. Since cities are not made up of 50-meter grid cells, buildings were chosen as the final projection canvas, so that the PMI was calculated for each building. The building's geometries were sourced from *OpenStreetMap* (OpenStreetMap 2016).

Looking at the resulting visualisation (see figure 54), areas with high PMIs were identified, mainly resulting from walking activities, which were named *walking islands*. Those islands were located across the city and connected by other modes of transport (e.g. underground, car, or bike). To explore those connections of known areas, a new network graph was created on top of the islands, highlighting the connections between islands and their respective strength. Strength was calculated through the number of trips, as well as modes of transportation and time spent on trips between islands (see Figure 55 left). In order to visualise the inherent individual perspective, this network graph was then translated into a force-directed graph-like concept, allowing the reorganisation of the islands depending on their edges. As a result, (see Figure 55 right), a new city layout emerged that represents the movement and the associated personal experience, which is inspired by the visualisations of the Situationist International (Debord, 1955).

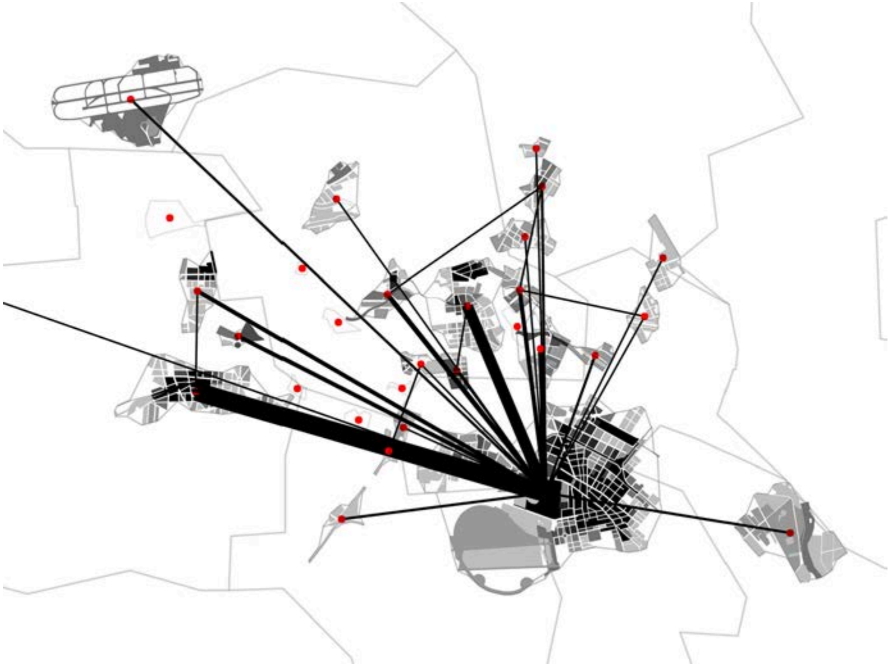


Figure 54: Trajectories mapped onto buildings. On top of the buildings, the network structure between the walking islands is visualised. Colour-scale: from white unknown to black well known buildings.

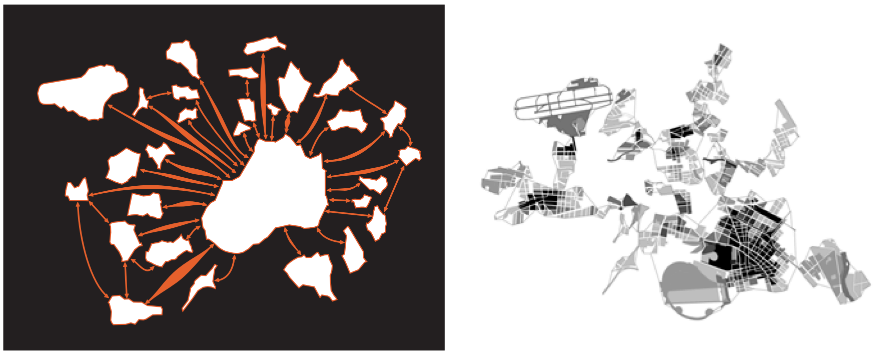


Figure 55: On the left, the same network represented in a Situationist fashion with a manual projection and weighted links (arrows). On the right, using a force-directed graph method, the city layout is rearranged and distorted by the suppression of distance.

6.5.2.2 *Evaluation of Clusters & Model*

As mentioned in the previous section, several uncertainties become apparent when implementing the concept of Mondschein et al. in an algorithmic approach. In order to improve the modelling approach, a preliminary user study with the initial algorithm that incorporates feedback on the assumed level of memorisation was conducted with the providers of the *Moves* datasets that has been used for this thesis. The *Moves* data was segmented on the time axis to only include the last 12 months. Since this part of the study was focused on the *potential memorisation* or knowledge of an area, the participants were also asked about how long they had been living in Berlin, which was shown to have no impact on the evaluation results.

For the test we selected 22 locations, with a wide range of algorithmically-generated *potential memorisation* values derived from the mix of transport modes. In addition, we added three locations presumably unknown to the user (at least based on the data we received). We created a web interface in which the user would receive a 360° image of each location, using *Google Street View*. The user could change the angle of the image but not the location. The participants were then asked if they knew the location. If so, they were asked to pin-point the location on a map. If the user did not know the location, the correct position was disclosed on the map. If the user recognised the area from the image, or knew the area indicated on the map, they were asked to rank their knowledge on a scale from 1 to 6. From user feedback we learned that in some cases the *Google Street View* images were quite old and therefore hard to identify. Thus, we allowed people to also rank their knowledge solely based on the position indicated on the map. This applied in cases when they were not able to identify the location based on the image, but later saw on the map that they indeed knew the location.

Our study investigated the performance of our algorithmic approach by comparing the PMI of our algorithm and the reported response from the users. With such a small sample it is difficult to discernibly implicate a correlation or a significant effect, but there is a clear trend in the data which indicates that our algorithm, even in this untrained phase, performs well. Figure 56 left and middle show that for a high memorisation response our algorithm also calculated a high PMI. Returning to Mondschein et al.'s theory, we see in Figure 56 on the right that areas with low memorisation responses show less walking and cycling and more motorised transport modes.

6.5.2.3 *Conclusion*

Informed by the theoretical perspective of mental maps, a data-driven and quantitative perspective on deducing further user-specific inference from the spatio-temporal data was presented in this section. The complexities of mental maps are acknowledged and the resulting algorithm therefore does not try to answer the question of how to visualise mental maps in general. Thus, the scope was limited to subjective experiences of the urban space on the

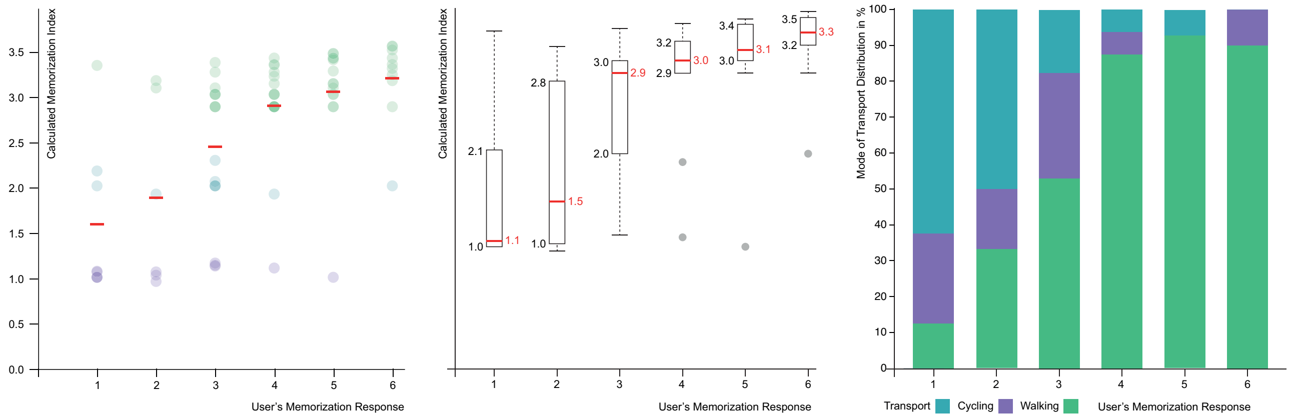


Figure 56: The x-axis represents the memorisation response from participants. On the left and in the middle, the y-axis represents the algorithmically-calculated *potential memorisation index*. The figure on the left shows the raw data points with averages (red) and the figure in the middle shows an analogous box-plot with medians (red). The graph on the right visualises the percentage for modes of transport for each memorisation response value. Green: walking, purple: cycling, and blue: motorised transport.

basis of movement through the city. Inspired by the works of Mondschein et al. the presented algorithm creates a PMI that is based on mode of transport. The algorithm showed promising results in the preliminary user study and therefore serves as a good example for complex insights generated from the user's underlying spatio-temporal data.

6.5.3 *Areal Conclusion*

This last section of the computational model focused on the second requirement from chapter 2, the deducing of inference from the users spatio-temporal data, especially insights that go beyond time and space. The first approach gave insights into temporal-semantic usage patters inside the cluster-areas. The second approach allowed the system to calculate a *potential memorisation index* for any given point in space. Those two techniques only serve as examples to highlight the potential of such client-side inference techniques.

6.6 MODELLING CONCLUSION

Chapter 6 focused on the user modelling as laid out by the requirement definitions in chapter 2. The process was guided by the conceptual model from chapter 3. The input data from the user's phone was transformed into vertices (locations) and edges (trips). These were used to build the network of the user's spatio-temporal behaviour. A novel combination of techniques for classifying and predicting the user's behaviour were presented, discussed, and evaluated. Even though the inaccuracy and imbalanced nature of the data presented a challenge throughout the applied techniques, the results from the evaluations are good. This proves the case that such spatio-temporal intelligence can be built on the client-side, only using the user's data without the need for aggregated multi-user approaches. Due to the imbalanced nature of the dataset, the biggest issue discovered throughout the exploration and development of approaches was the correct classification of minor-classes. Chapter 7 draws together the developed techniques in order to highlight how everything could be combined in a real-world use case. To this end, the last requirement of sharing the developed user model with a remote service will also be discussed.

7 IMPLEMENTATION

The overall focus of the thesis was put on the computational model for client-side modelling of a user's spatio-temporal behaviour, presented in chapter 6. This chapter will take a step back again and briefly highlight how this model could be incorporated into a real-world application. This endeavor builds upon the technology stack that was introduced in chapter 2 (see figure 57).

7.1 EVENT-BASED RECOMMENDATION

The previous chapter outlined models and techniques for classifying and predicting the user's behaviour. In the context of the application domain, those insights need to be turned into assistance or recommendations. A common developing paradigm in object-oriented programming are so-called *events*. In this example, events serve as the middlemen between client and server side. Events can connect two processes with one another and synchronously send information. To apply this to the use case in question, the external recommendation service registers to the client-side user model (examples are provided in list 3).[25]. If such an event is triggered and a recommendation is required, a connection to the remote service needs to be established. At this point, it is important to point out, that, if a user wants to make use of a modern data-driven service, some user-data needs to be exchanged at some point. Otherwise collaborative filtering and other IR techniques will not deliver sufficient results. Therefore, the next section will introduce several approaches on how the client-side model can be shared with a remote service without revealing all of the user's sensitive raw information.

[25] As previously mentioned, the exact technological development of an application making use of the client-side modelling approach is not the focus of this work. Therefore, the event-based process is only briefly introduced to explain a possible connection between client-side and server-side processes.

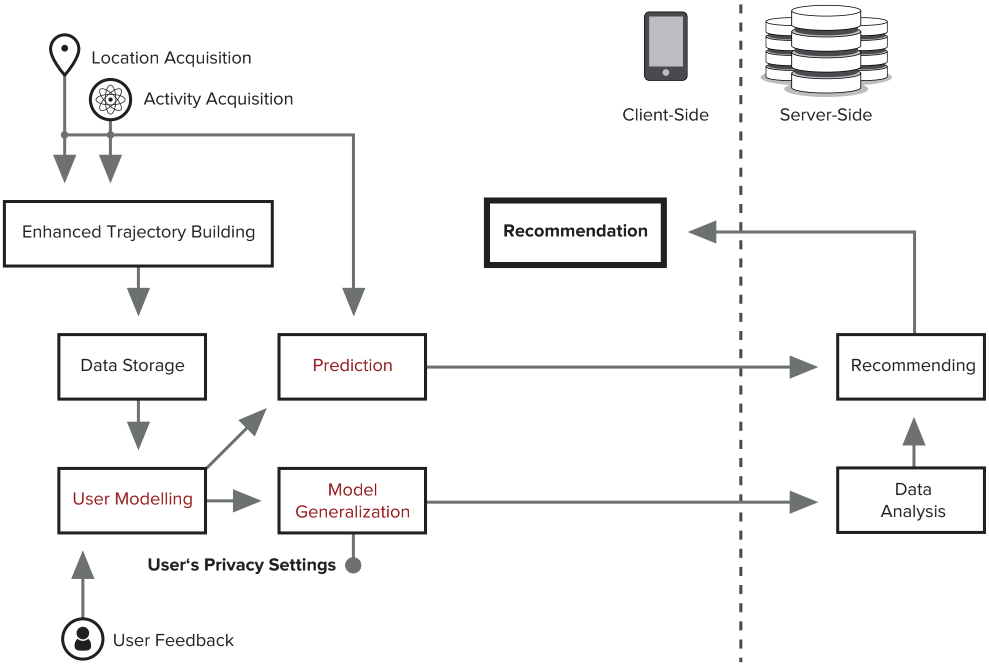


Figure 57: Technology Stack, the foundation of the conceptual Framework. In red the central components to be explored and developed in this thesis.

Listing 3: Example event registrations

```
let user_mode = UserModel()

user_model.on(
  {
    type:’destination_prediction’
  },
  function(user_object, destination, probability){}
)

user_model.on(
  {
    type:’destination_prediction’,
    filter:{
      key:’type’,
      value:’work’
    }
  },
  function(user_object, destination, probability){}
)

user_model.on(
  {
    type:’new_location_classification’,
    filter:{
      key:’type’,
      value:’home’
    }
  },
  function(user_object, location, probability){}
)
```

7.2 GENERALISATION & DATE EXCHANGE

Following the concept of *data-austerity* and *data-avoidance*, the goal is to share *only* the data that is truly required in order to perform a certain task. In the conceptual model, the principle of generalisation was introduced to build such a data sharing strategy. The following sections present two distinct conceptual examples for how such a process could be designed: Generalisation and Ensemble training.

7.2.1 *Generalisation of Data for improved Privacy*

Generalisation is always guided by certain (communication) goals. Similar to cartographic generalisation, the communication goal of the sharing process guides the selection and abstraction of the data. The proposed process uses GIS' concept of hierarchies for generalisation. "When going from the bottom up [of such a hierarchy] one encounters an increasing level of generalisation, i.e. the object description becomes less specific with each step up in the hierarchy"(Molenaar, 1998, p. 137). In order to achieve those levels of generalisation, Molenaar describes four operations:

> "1) The selection of objects to be represented at the reduced scale. This selection will be based on the attribute data of the objects. 2) The elimination from the database of objects that should not be represented. 3) The aggregation of [...] objects that should not be represented individually. 4) The reclassification of the generalised objects."
>
> (Molenaar, 1998, p. 167)

While the application to spatial (geometric) information is well known, this can also be applied to semantic information. The following examples will refer back to the initial application domain of LR in LBAs, as a goal for the exchanged data.

Spatial Network

Generalising trajectories, is a task that can, for example, be achieved through simplification algorithms. Generalising a spatial network is more complex. Traditional network generalisation approaches, do not account for the spatiality of the network. The clustering, which is already built into the approaches presented in the previous chapter, therefore presents an alternative. In this case, the specific start and end locations are hidden and only the connections between clusters are presented (see figure 58). By only sharing clusters and cluster connections, a remote service could still use the data to find common clusters and predict general patterns of trips between those clusters. In addition, this would require the submission of temporal information inherent to those trips (see next example). In a use case, where the goal is a route calculation, such a level of abstraction would obviously not be sufficient.

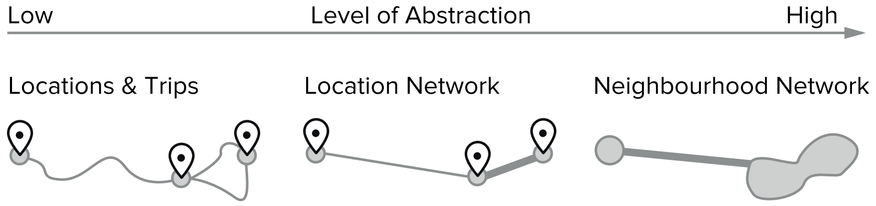


Figure 58: Example for the generalisation of the spatial trajectory network.

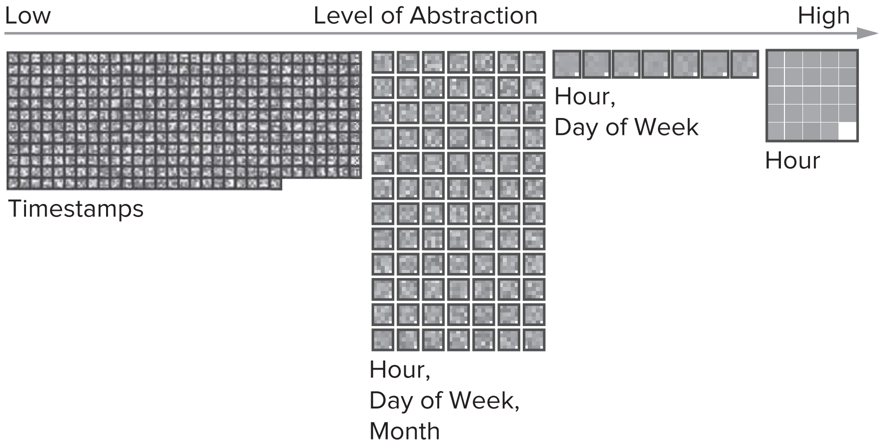


Figure 59: Example for the generalisation of temporal attributes.

Temporal Information

The generalisation of temporal information, even more so than the previous spatial networks, relates to the hierarchical generalisation of data. Taken a set of timestamps of visits in a certain location, a remote service wants to detect temporal patterns in the data of visitors. In this example, the exact time stamps represent the raw information with the highest level of detail. As in the techniques applied in chapter 6, the exact time can be generalised into 10-minute intervals, thereby introducing a first level of generalisation. Next, the exact date can be generalised into the day of the week and month. The day of the week can be used to identify patterns between workdays and weekends and the month can be used to identify seasonal changes. This can then be further generalised by getting rid of the day of the week and month, just leaving the 144 time slots of a 24 hour days. Each layer loses information while it increases the anonymisation and, as a result, the privacy of the user (see figure 59).

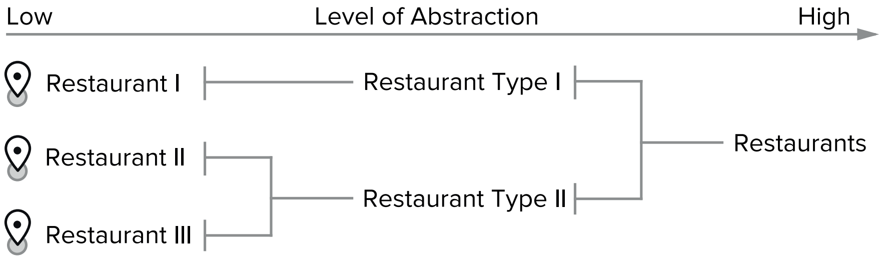


Figure 60: Example for the generalisation of semantic attributes.

Semantic Information Similar to the temporal information, the semantic information can be transformed into a hierarchical systematic. The location types, for example, are traditionally already organised into hierarchies. From top-level categories, like 'restaurant' in general, to a second level like *Asian restaurant* to a very specific classification like *Vietnamese restaurant* (see figure 60). Molenaar also uses an example in which he connects spatial generalisation to semantic descriptions: field > farm yard > farm > farm district (Molenaar, 1998), which is similar to how locations are generalised into neighbourhoods in this thesis.

Seeing the three examples above, it should be clear that every level of abstraction and generalisation reduces the level of detail and, therefore, also the level of detail for any inference built on top.

7.2.2 *Sharing trained Machine Learning Models*

At several instances throughout the computational model, ANNs are being trained and used for classification and prediction. One reason that led the decision to use ANNs is their ability to be shared and recombined. A trained ANN can be stored in a relatively small data format, compared to the training data that went into the training of that model. Those model states can then be shared without including the original training data. If a third party collected such states, they could use so-called *ensemble training* or *ensemble machines*[26] to combine a set of ANNs and create a global generalised ANN. A variety of techniques exists for combining those ensembles, one of the less complex techniques is ensemble averaging (see e.g. Naftaly et al., 1997; Zhou et al., 2002; Krogh and Vedelsby, 1994, an introductory overview of the topic is provided by Haykin, 1998). Most of the publications did not have this intended use case in mind but rather use the approach for creating better predictions, for example by overfitting individual models in the ensemble (Peter Sollich, 1996).

The resulting generalised ANN could also be used as a starting point for users who need to start from scratch with an untrained network, similar

[26] The technique of ensembles is sometimes also referred to as *committee machines*. As in a committee of different experts that help solve a problem (Haykin, 1998, chapter 7).

to how the time-use survey was used for the location type classification in chapter 6.

From models to systems

The conceptual data exchange approaches briefly outlined above connect the developed client-side user modelling approach to a larger context of applications and scenarios. In so doing it puts a specific emphasis on the privacy-preserving focus of this work. To further extend the pros and cons of the developed approach, the last chapter will close this thesis with a discussion of limitations and opportunities.

8 OUTCOME

This final chapter discusses upsides and downsides of the developed approach, starting with an expert evaluation. This is followed by an elaboration on possible future applications of the developed method. The chapter is completed by an overview of current developments in the field of client-side modelling and machine learning. The last section distils the most important insights of this thesis and lays out future works.

8.1 EXPERT EVALUATION

Each component of the computational model in chapter 6 has been technologically evaluated using the sample datasets. Beyond this technological evaluation, the work was presented to three experts from the domain of spatial analytics. Two of the experts work for companies that specialise in the analysis and visualisation of spatial data and one expert works for a research centre in the department of mobility analytics. Each interviewee was given a presentation of the system, followed by an open interview that centred around the three focus points outlined in the first chapter: Client-Side, User-Modelling, and Data Exchange. All interviewees stated that the developed techniques presented an interesting new approach with many possible applications, not only in regards to privacy but also in areas including urban planning and especially in the domain of distributed computing. As noted by one interviewee, a combination of the suggested client-side approach - that comes into effect especially in situations in which network access is weak - with a more powerful server-side approach, could present a promising use case. Besides the classification and prediction algorithms, the method for potential memorisation as an example for inference techniques was of particular interest for the interviewees. Even though the focus was put on the modelling and the theoretical and computational process, many questions went beyond the academic scope and took up real-world use cases and implementational questions. Throughout the interviews, a couple of questions or critical remarks were repeatedly raised, which will be briefly discussed in an aggregated manner in the following paragraphs.

Is the approach feasible in regards to processing limitations on mobile handsets?

The client-side modelling envisioned in this thesis is meant to be implemented on a smartphone. All technical frameworks used in the computational model were picked to be compatible with the inherent requirements of those mobile handheld devices. While the processing performance strongly de-

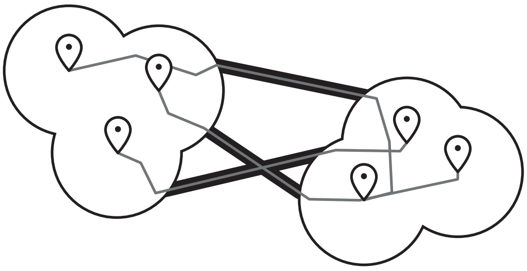


Figure 61: Removing the sensitive start- and end-segments of the trajectory.

pends on various environmental and data-specific requirements, the data-footprint was for all *Moves* test-subjects around 10 MegaBytes large. This includes the input trajectories, as well as clusters and corridors. To put this number into perspective, most of the mainstream smartphones from the last two years are backed by 32 or more gigabytes of storage. Accordingly, the approach stays well within the processing limitations of mobile handsets.

Could anonymisation be used alternatively?

In existing server-side implementations one of the standard solutions for ensuring a user's privacy is anonymisation. A problem with this approach is, however, that a large field of research is also investigating methods of de-anonymisation. Gambs et al., for example, showed that in a dataset similar to the one used in this thesis 45% of users could be correctly identified, although their data had been previously anonymised (Gambs et al., 2014). The problem lies within the distinct spatio-temporal patterns of every individual in the dataset. Consequently, anonymisation was not taken into account as a viable privacy technique that could be useful to the framework of this thesis. An interesting suggestion by one of the interviewees in regards to anonymisation was the removal of sensitive information from the trajectories, more particularly the start- and end-segments that could lead to the exact start- and end-location of the user. This could be achieved by starting the trajectories at the outskirts of the clusters, instead of at the exact start- and end-locations (see figure 61).

What influence does the progressive training of algorithms (more data over time) have on the presented approach?

Each module of the computational model presented in chapter 6 was evaluated on several real-world datasets. In those tests, the analysis and training of the data was conducted in batches. This means that after removing test-cases from the overall dataset, the remaining training data was used to train or build the prediction model. In a realistic setting, this would obviously not be the case. Every day that the user operates the application, more training data is generated, which needs to be used to update and further train the models. While this has not been discussed in the previous chapters, the methods and techniques chosen to create the models all allow for this to be implemented. In the domain of machine learning, such concepts are subsumed under the term *online machine learning* (see Fontenla-Romero et al., 2013 for an introduction). Those techniques allow for the continuous and thereby progressive training of machine learning algorithms, like the ANNs used in this work. The processing techniques applied to the spatial input data were already done in a progressive manner. While this progressive approach makes it in part slow for a large batch of data, new data is incorporated very quickly. As an example, the process for incorporating a new trajectory into a one-year dataset takes less than a second - which includes cleaning and inserting the data, calculating the intersections for corridors, and reclustering the network.

How does Big Data vs. Personal Big Data and Server- vs. Client-Side compare?

The downsides identified in this thesis, notably the difficulty in identifying minor classes in the unbalanced data, were also brought up by the interviewees. In this case, a Big Data approach holds the potential of having larger amounts of data at hand, aggregated from several individuals. Nonetheless, this stands against the advantages of distribution and privacy, which are the main motivations for this thesis. Another downside raised by one of the interviewees is the capability to exactly identify specific transport types. While this thesis relied on the transport types delivered through the smartphone's API, server-side approaches that integrate street network data for instance, can yield more detailed results. Depending on the use case, this would make a server-side solution more powerful. Especially in regards to business use cases, the sustainability of server-side solutions holds benefits for companies in terms of their ability to secure the longevity of data-driven services, as discussed in chapter 2. A particular downside of the client-side approach is the first training phase initiated by the user, where no existing data is available for training the model. While this thesis presented an approach for using a generalised model for the location identification, similar approaches would be required for the other classification and prediction methods. A minor remark came up in regards to cross-device scenarios, in which case a server-side approach would be a lot easier to implement.

Remarks on: Services Need Data

This last remark, regarding the service's need for data, was not mentioned in the interviews, but came up in discussion with more business focused experts. It dealt with the question of: *How much data does a remote service need to rise to its full potential?* The question is difficult to answer, as it depends on the specific service. The experts agreed that a certain process of modelling and abstraction is conducted on the server-side service anyway and it could, therefore, be viable to move this part to the client-side. On the downside, if it is decided to later on that the model needs to be changed and additional data needs to be acquired, this would then be a problem for the client-side models. Furthermore, collaborative filtering, which is normally based on having aggregated raw user data at hand, would need some rethinking. The focus of this work remained on the user modelling. Therefore, future research would require a focus on the data exchange and on tests that compared traditional approaches to the novel approach presented in this thesis.

8.2 envisioning future applications

> "The future cannot be predicted, but futures can be invented."
>
> Gabor, 1963, p. 207[27].

Before the discussion closes with final concluding remarks, this section will take a step back from the scientific and technology-driven course of this dissertation and shift towards envisioning further scenarios of applications for the developed approach. The first chapter highlighted that - beyond the technological and methodological development in this thesis - one goal is to foster and support the discourse on data privacy. As discussed in the first chapter, there are tendencies to loosen the regulations on protecting citizens' data privacy in favour of data-driven innovation by the digital economy. With the conceptual and technological developments in this thesis, an alternative approach for such innovative technologies is laid out and discussed that still enables innovative technologies. To deepen this discussion, the following examples will partially go beyond today's status quo and illustrates the way services are developed and implemented. By envisioning such future implementations of the thesis' approach, the examples try to further question the status quo and illustrate alternatives. The concepts of *selective cloud computing*, *client-side user modelling*, and *generalised data exchange* form the basis of all examples.

Shopping:

Recommendations in digital shopping experiences are a common example for personalised user-experiences. The proposed client-side modelling could easily observe the shopping behaviour of a user across shopping sites and build a user-specific interest model. This model could then be shared with shops in return for discounts or other services. This would not only allow the user to regain control over their

[27] The quote or variations of the quote often appeared in computer science related literature. Alan Kay is most associated with the quote, but Gabor was the first in line to say it according to Investigator, 2017

data but would furthermore allow shopping sites to get a broader perspective by being able to access the user's complete shopping behaviour model across all shopping sites, instead of one that is limited to their own platform.

News reading:

Recommendations in the context of online news has lately received some negative publicity through the discussion concerning filter bubbles[28]. While not completely removing the problem of filter bubbles, the ownership over the client-side behaviour model would make users independent from aggregation systems like Facebook. Users could thus take their model and visit news outlets directly and filter the content based on their interests.

Routing:

Exact routing was not part of the exemplary methods developed in this thesis, but it could easily be built on top of the methods discussed in this work. By incorporating the user-specific model, the system could for example predict previously used routes to a destination. Even more interestingly, a system could implement Hägerstrand's personal constraints within the calculation. Instead of computing a generalised duration for a bike trip, the algorithm could take the user's personal constraints into account and in so doing refine the predictions.

The public discourse on privacy indicates that there is a growing interest in personal information privacy and data protection. A study funded by the EU in 2012 even highlighted that there is an economic potential for companies to implement privacy-preserving functions (Jentsch et al., 2012). On the other hand, several studies and publications discuss what is known as the *privacy paradox* (Norberg and Horne, 2007; Kehr et al., 2014; Brown and Muchira, 2004). The paradox details the phenomena that people describe themselves as being 'privacy concerned' while their actual behaviour indicates otherwise. In his overview paper Preibusch described that finding significant insights about the influence of privacy concerns on users' behaviour is complex, as it involves interdependencies between concerns and other needs:

> "users can appropriate returns from disclosing personal data, such as better prospects of finding a job or a romantic relationship [...] [to] returns in the form of better product recommendations. [...] In both cases, users' benefits may well outweigh their privacy concerns."
>
> Preibusch, 2013, p. 5

In this context, the system discussed in this thesis offers a potential alternative, which does not force users to change their behaviour, but instead changes the underlying infrastructure.

[28] The term filter bubble describes the effect of intense personalisation, which can facilitate a news feed that only represents the user's own world views and ideologies.

8.3 CURRENT DEVELOPMENTS

In the final months of this dissertation, several technological developments emerged that align with the proposed framework in this thesis. The following paragraphs will shortly describe three examples and discuss them in the context of this thesis. The mobile development plugin by the company *Set* was already introduced in the technology section of chapter 2 (Set, 2017). *Set* was publicly announced in early 2017 and left private beta in early April. Its capabilities now include location detection and behaviour (trip destination) predictions. While their implementation looks very promising, the framework so far offers no approach towards sharing the user's data. As highlighted in the generalisation section of chapter 7, in order to provide recommendations or other data-driven services, data needs to be shared in some form. The second example is closer to delivering a solution for exactly that. In early April 2017, *Google* announced their upcoming technique *Federated Learning* (McMahan and Ramage, 2017). While their *TensorFlow* framework already allows for ensemble learning, federated learning takes this approach to the next level. The concept behind federated learning is to start with a global model that is installed on the user's device. The user then customises this model through their own data and thereby trains and creates their own model. This user-specific model resides on the user's device. If a good internet connection is available, the user-specific model is sent to the server, where it is recombined with other users' models into a new global model. Thereby, the user's training data is not shared with the remote service, instead only the abstracted model is shared. This is similar to the advantages outlined in this thesis. *Google* emphasises, that this is not only of interest in regards to privacy. It is also relevant in terms of saving data transmission costs, as the phone does not need to send continuous updates but only incremental changes of the abstracted model. This approach is very close to the framework outlined in this thesis. The new technique by *Google* should be made public in the next few months and is already implemented in the latest version of *Google's* smart keyboard (word prediction). *Apple's* approach to privacy-preserving data collection is Differential Privacy (DP). The technique of *DP* gained some attention lately as *Apple* is integrating it into their data aggregation infrastructure. The fundamental concept behind DP is adding noise to the collected data. In principle, the technique adds so much noise that general patters remain intact, while it is not possible to identify individual events or items. A recent news announcement by *Bloomberg Technology* (Gurman, 2017) reported that *Apple* is working on a mobile chip specifically designed for artificial intelligence, or rather machine learning. *Apple* was one of the earliest companies to introduce a smart assistant on their devices: *Siri*. This development of custom chip designs that are optimised for machine learning is not new. In mid-2016, *Google* announced their new chip designs (Jouppi, 2016), which are optimised for their *TensorFlow* platform. In contrast to *Apple's* technology, the so-called TPUs are developed for *Google's* cloud-based services and are, as of May 2017, already available in their second generation (Dean and Hölzle, 2017). *Apple's* move to optimise client-side machine learning through custom processors falls in line with other chip-manufacturers, for example, *ARM* (Windeck, 2017).

The developments outlined above show that this thesis is set in a technological domain which is in a state of flux. The trend of machine learning, especially with ANNs, is merging with the concept of distributed computing, as discussed in this thesis through the concept of *selective cloud computing*. While it feels like this trend is mainly driven by an economic interest in the optimisation of behavioural modelling (e.g. through distributed computing), an increase in more data privacy might be the by-product of this development - a development which needs to be assessed carefully while it unfolds.

8.4 DISCUSSION

This thesis set out to explore the potentials of client-side user modelling of spatio-temporal behaviour. As a proof of concept, a client-side modelling process for location- and context-based applications was developed. This development process was grounded on the requirements outlined in chapter 2. As a conceptual framework, the principles of network theory were combined with time-geography, that later defined the spatio-temporal network constraints in chapter 3. Those constraints were transferred into a computational model, which was tested and evaluated in chapter 6. This process was accompanied by a critical perspective on technology in chapter 5, highlighting the potentials and limitations of the presented approaches.

Scientific Contribution

This thesis explored the concept of *selective cloud computing* in order to achieve a stronger privacy-centred user-modelling approach. As a proof of the concept, a client-side user-modelling approach for spatio-temporal behaviour was developed and evaluated. The modelling was achieved by combining methods and techniques from the domain of machine learning and spatial data analysis. For this particular data type, the combination of the two fields within the limitations of a smartphone device, is a novel development based on all indications. Even the plugin by *Set* (discussed in chapter 2) which is neither published open source nor academically documented, does not go as far as the approaches in this thesis. Two specific challenges in this development were posed by the size of the datasets: the imbalance within datasets, and the spatial error thresholds in the input trajectories. Due to those restrictions, many traditional methods for trajectory analysis could not be used, like the hidden markov model, map matching, or common segment analysis. An additional restriction was the implementational focus on mobile devices, which emphasised feasibility in regards to performance and resource usage. In this specific setting, the presented conceptual and computational model revealed a novel approach with promising performance, as presented in the previous chapters. It thereby extends the discourse on spatial data processing through ANN and in conclusion illustrates the inherent potentials for applied use cases and scenarios. Beyond client-side user modelling, the presented approach could also be of interest for many other research fields that deal with the modelling of spatio-temporal behaviour and the limitation of sparse datasets. Those research fields span from transport research to human geography. Due to the small datasets required for the approach, the novel technique could also be helpful to prototyping and testing environments.

The exemplary proof of concept presents a feasible approach for a privacy-centred modelling while also taking into account recent discourses in adjacent fields. In this context, most discussions on privacy highlight the problems and try to force regulations onto businesses. As opposed to this, the approach developed within this thesis shows an alternative, which could change the underlying infrastructure instead of forcing business or users to change their behaviour.

Limitations

As mentioned in chapter 6 and the expert feedback above, there are also some downsides to the presented approach. For one, there is the limitation to only include data by one user, which limits predictions or the identification of classification patterns to those that have previously been exposed by the user and thus are represented in the data. Secondly, there is the problem of minor classes within the behaviour modelling. Those two limitations introduce an accuracy trade-off in favour of the client-side modelling. This contradiction has been noted by researchers who considered privacy in their discussion. Berkovsky et al., for example, point out that "the goals of the privacy-preserving mechanisms contradict the goals of the personalization systems, leading to privacy versus accuracy trade-off"(Berkovsky et al., 2009, p 16f). They continue by suggesting that "[o]ne possible compromise may be that the user will have a comprehensive representation of his/her model and allow parts of it to be provided anonymously to the service provider, if requested"(Berkovsky et al., 2009, p 17), which aligns with the concept presented in this thesis. Therefore, the best results will likely be achieved by combining server- and client-side approaches. This can be achieved for example through generalised global models and user-specific local models.

This thesis put the focus on modelling and did not elaborate on the challenge of abstracting or rather generalising and then sharing the data in the same depth. As will be highlighted in the future works section, this particular strand of research still requires attention.

Implementational limitations of the approach include problems in cross-device syncing, as the client-side user model resides on a specific device.

This and other implementational downsides are briefly discussed in chapter 2. They simply present obstacles for implementing the proposed model, but not for the modelling itself. As a result, they will not be further discussed.

Future Work

The work presented in this thesis explores an approach to privacy-preserving *selective cloud computing* for the domain of LR in LBSs. This work lays out frameworks and foundations for further explorations into privacy-focused mobile applications. While this work showed first examples, there are several things to further investigate. Three perspectives are believed to be especially fruitful for further discoveries: additional contextual information, increased privacy, and interfaces to the model.

Additional contextual information:

The models discussed in this thesis were primarily built on a user's spatio-temporal behaviour, extended through activity classifications and user feedback. As described in chapter 2, an important feature of modern LBAs is their location- and context-awareness. While the former is modelled and represented, additional context-features are not taken into account (due to the extent of this thesis). In a next step, the models need to be extended to include more contextual information, that could be acquired in reference to the space and time parameters. A first example for such an extension is weather data (Liu, 2014). To demonstrate a use case: The models developed in chapter 6 describe trips between locations as well as the machine learning predictions. They are used to estimate the highest probability for a possible destination when the user starts a trip. This model could be extended to also include weather data in the calculations. As a result, the trajectories would be extended through weather data (e.g. temperature, precipitation, and wind), which would allow the machine learning algorithm to include those parameters as additional input. Hypothetically, this would enable the prediction that the chances of the user taking public transport or bike on a sunny day might be 50-50. On a rainy day, on the other hand, this enhanced model would likely calculate a low probability for the user to choose the bicycle.

Long-term observations:

The data used to validate the approaches developed in this thesis covered one year of usage. This timespan is not long enough to discover changes in the user's spatio-temporal behaviour patterns. Such a discovery would require data collected over longer timespans. With long-term data at hand, one could furthermore extend the modelling approaches to systematically identify changes in the user's behaviour, for example by employing long-term and short-term models, which compare prediction results.

Increased privacy:

In chapter 7 the data exchange between client and server was mostly discussed on a conceptual level. In a next phase, this needs to be transferred into a more concrete computational model and put to a test. In addition, contemporary techniques for anonymisation should be explored in order to access their potential for the data exchange. The technique of *DP*, for example, as discussed in the previous section, could offer such a service. Applied to the approach presented in this thesis, this could mean adding certain trajectories or locations to the dataset that create enough noise that the exact home address of the user cannot be retrieved from the data while the home region can still be identified.

Interfaces to the model

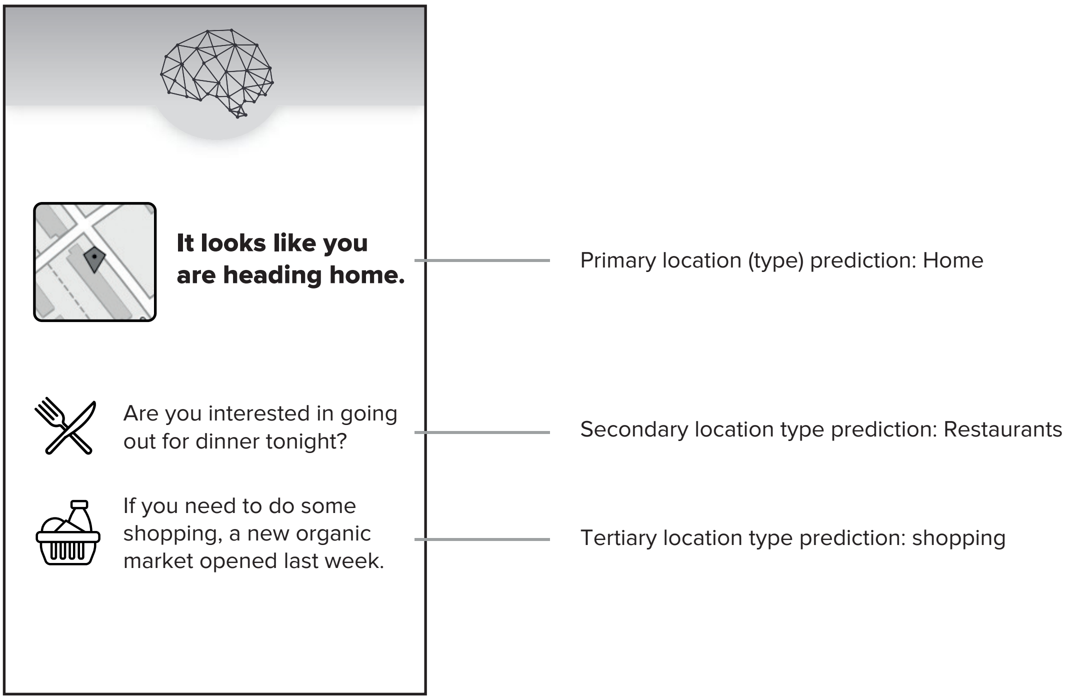
While it is unlikely that the client-side model will reach the precision of a big data server-side solution, there might be other ways of overcoming the inherent problems of the client-side modelling. Data-driven interfaces for instance, which include the user in the decision-making process, could do just that. For example, the problem of minor classes in the prediction of destinations and events could be overcome by including not only the prediction with the highest probability but also lower ranked results (see figure 62). In this way, the lack of accuracy might be overcome by including the user in the decision-making.

Critical Reflection

Beyond the focus on the technical innovations, this work also sought to take a humanistic perspective to the user modelling process. This thesis introduces individual and user-specific approaches that go beyond generalised aggregates and thereby conceptually allow more diversity in the modelling process. This puts emphasis on the user's individual perspective on spatio-temporal behaviour. In a next step, this could be extended to reflective practices, allowing users to visually analyse their behaviour (see for example Otten et al., 2015; Meier and Glinka, 2017; Thudt et al., 2013) or even build their own context- and location-based intelligence.

Personal Reflection?

The field of machine learning is a vibrant field of research. Many disciplines that need help in analysing and classifying data or more advanced machine intelligence have an interest in the upcoming sets of developments. While there is a broad conceptual discussion on machine learning in many



It looks like you are heading home.

Are you interested in going out for dinner tonight?

If you need to do some shopping, a new organic market opened last week.

Primary location (type) prediction: Home

Secondary location type prediction: Restaurants

Tertiary location type prediction: shopping

Figure 62: Overcome the problem of minor class predictions through an interface solution. In this scenario the location *home* received the highest probability, but for the same spatial area and the same time threshold other locations might be viable destinations as well. Brain graphic by Sergey Patutin, Knife by Mello, Groceries by Chiara Galli, all from the Noun Project.

academic communities, there is still a lack in reproducible implementations and their technical documentation. The *WEKA* tool and *Convnet.js* are two of the few examples where researchers turned their research into publicly available technology. This is a problem that goes beyond the field of machine learning and is, in my opinion, a fundamental flaw in the academic gratification system. While some publications present novel and innovative approaches to spatial data analysis, it is merely impossible to implement and validate those approaches without spending a lot of time rebuilding them. The reproduction and reimplementation of other researchers' work was one of the biggest challenges in developing this thesis. In hope of making a difference, all experiments conducted in this thesis and their code foundation are published under an open source license.

BIBLIOGRAPHY

Aggarwal, Charu C (2016). ‘An Introduction to Recommender Systems’. In: *Recommender Systems The Textbook*. Springer International Publishing, pp. 1–29. isbn: 978-3-319-29657-9.

Alexander, Christopher (1965). ‘A city is not a tree’. In: *Architectural Forum* 122.1, pp. 58–62.

Alexander, Christopher, Sara Ishikawa, Murray Silverstein, Max Jacobsen, Ingrid Fiksdahl-King and Shlomo Angel (1977). *A Pattern Language: Towns, Buildings, Construction*. 1386544400th ed. New York: Oxford University Press.

Amazon (2006). *Announcing Amazon Elastic Compute Cloud (Amazon EC2) - beta*. url: `https://aws.amazon.com/about-aws/whats-new/2006/08/24/announcing-amazon-elastic-compute-cloud-amazon-ec2---beta/`.

Android, Developers (2017). *Making Your App Location-Aware*. url: `https://developer.android.com/training/location/index.html`.

Antonello, Andrea (2016). *Geopaparazzi*. url: `http://geopaparazzi.github.io/geopaparazzi/`.

Apple, Developers (2017). *CMMotion Activity*. url: `https://developer.apple.com/reference/coremotion/cmmotionactivity`.

Asia, Microsoft (2012). *GeoLife GPS Trajectories*. url: `https://www.microsoft.com/en-us/download/details.aspx?id=52367&from=https%3A%2F%2Fresearch.microsoft.com%2Fen-us%2Fdownloads%2Fb16d359d-d164-469e-9fd4-daa38f2b2e13%2F`.

Assad, Mark, David J Carmichael, Judy Kay and Bob Kummerfeld (2007). ‘PersonisAD - Distributed, Active, Scrutable Model Framework for Context-Aware Services.’ In: *International Conference on Pervasive Computing*. Springer Berlin Heidelberg, pp. 55–72. isbn: 978-3-540-72036-2.

Bach, B, P Dragicevic, D Archambault, C Hurter and S Carpendale (2014). ‘A Review of Temporal Data Visualizations Based on Space-Time Cube Operations’. In: *Eurographics , STAR State of The Art Report*, pp. 1–19.

Balci, Osman (1997). ‘Verification, Validation and Accreditation of Simulation Models.’ In: *Winter Simulation Conference*, pp. 135–141.

Baldauf, Matthias, Schahram Dustdar and Florian Rosenberg (2007). ‘A survey on context-aware systems.’ In: *IJAHUC* 2.4, pp. 263–277.

Banks, Catherine M (2010). *Introduction to Modeling and Simulation*. Theoretical Underpinnings and Practical Domains. Hoboken, NJ, USA: John Wiley & Sons, Inc. isbn: 9780470590621.

— (2012). ‘Research and Analysis for Real-World Applications’. In: *Handbook of Real-World Applications in Modeling and Simulation*. Hoboken, NJ, USA: John Wiley & Sons, Inc., pp. 8–25. isbn: 9781118117774.

Barkhuus, Louise and Anind K Dey (2003). 'Location-Based Services for Mobile Telephony: a Study of Users' Privacy Concerns.' In: *INTERACT 2003*.

Barnes, Susan B (2006). 'A privacy paradox: Social networking in the United States'. In: *First Monday* 11.9.

Batty, Michael (2013). *The New Science of Cities*. MIT Press. ISBN: 0262019523.

Baudisch, P (1999). 'Joining collaborative and content-based filtering'. In: *Proceedings of the ACM CHI Workshop on . . .*

Beam, L Andrew (2017). *Deep Learning 101 - Part 1: History and Background*. URL: http://beamandrew.github.io/deeplearning/2017/02/23/deep_learning_101_part1.html.

Belmonte, Nicolas Garcia (2016). *Engineering Intelligence Through Data Visualization at Uber*. URL: https://eng.uber.com/data-viz-intel/.

Berkovsky, Shlomo, Dominikus Heckmann and Tsvi Kuflik (2009). 'Adressing Challenges of Ubiquitous User Modeling: Between Mediation and Semantic Integration'. In: *Advances in Ubiquitous User Modelling*. Ed. by Tsvi Kuflik, Shlomo Berkovsky, Francesca Carmagnola, Dominikus Heckmann and Antonio Krüger. Springer, pp. 1–19.

Biagioni, James and John Krumm (2013). 'Days of Our Lives: Assessing Day Similarity from Location Traces'. In: *LNCS 7899 - User Modeling, Adaptation, and Personalization*, pp. 1–13.

Blondel, Vincent D, Jean-Loup Guillaume, Renaud Lambiotte and Etienne Lefebvre (2008). 'Fast unfolding of communities in large networks'. In: *Journal of Statistical Mechanics: Theory and Experiment* 2008.10, P10008.

BMJV (2009). 'Datenvermeidung und Datensparsamkeit'. In: *Bundesdatenschutzgesetz* 3a.

Bobadilla, J, F Ortega, A Hernando and A Gutiérrez (2013). 'Recommender systems survey'. In: *Knowledge-Based Systems* 46, pp. 109–132.

Borchers, Detlef (2016). *Europäischer Polizeikongress: Kanzleramtsminister Altmaier fordert neues Datenbewusstsein*. URL: http://www.heise.de/newsticker/meldung/Europaeischer-Polizeikongress-Kanzleramtsminister-Altmaier-fordert-neues-Datenbewusstsein-3115502.html.

Borges, Jorge Luise (1946). 'Del rigor en la ciencia'. In: *Los Anales de Buenos Aires, año , no.* Pp. 1–1.

Briegleb, Volker (2016). *EU-Digitalkommissar Oettinger: "Wir sind beim Datenschutz hypersensibel"*. URL: http://www.heise.de/newsticker/meldung/EU-Digitalkommissar-Oettinger-Wir-sind-beim-Datenschutz-hypersensibel-3104183.html.

Brown, Mark R and Rose Muchira (2004). 'Investigating the Relationship between Internet Privacy Concerns and Online Purchase Behavior.' In: *J. Electron. Commerce Res.* 5.1, pp. 62–70.

Butler, Howard, Martin Daly, Allan Doyle, Sean Gillies, Tim Schaub and Christopher Schmidt (2008). *GeoJSON Specification*. URL: http://geojson.org/geojson-spec.html.

BVerfG (1983). 'Volkszählung, BVerfGE 65, Urteil des Ersten Senats, 1 BvR 209/83 u. a. ' In:

Carnahan, Brian, Gérard Meyer and Lois-Ann Kuntz (2003). 'Comparing Statistical and Machine Learning Classifiers - Alternatives for Predictive Modeling in Human Factors Research.' In: *Human Factors* 45.3, pp. 408–423.

Carpenter, Gail A, Marin N Gjaja, Sucharita Gopal and Curtis E Woodcock (1997). 'ART neural networks for remote sensing - vegetation classification from Landsat TM and terrain data.' In: *IEEE Trans. Geoscience and Remote Sensing* 35.2, pp. 308–325.

Carroll, Lewis (1993). *Sylvie and Bruno*. Project Gutenberg.

Ceri, Stefano, Peter Dolog, Maristella Matera and Wolfgang Nejdl (2004). 'Model-Driven Design of Web Applications with Client-Side Adaptation.' In: *International Conference on Web Engineering*. Berlin, Heidelberg: Springer, Berlin, Heidelberg, pp. 201–214. ISBN: 978-3-540-22511-9.

Chawla, Nitesh V, Kevin W Bowyer, Lawrence O Hall and W Philip Kegelmeyer (2002). 'SMOTE - Synthetic Minority Over-sampling Technique.' In: *J. Artif. Intell. Res.* 16, pp. 321–357.

Chen, H, F Perich, T Finin and A Joshi (2004). 'SOUPA: standard ontology for ubiquitous and pervasive applications'. In: *The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004*. IEEE, pp. 258–267. ISBN: 0-7695-2208-4.

Chen, Harry, Tim Finin and Anupam Joshi (2003). 'An ontology for context-aware pervasive computing environments.' In: *Knowledge Eng. Review ()* 18.3, pp. 197–207.

Chen, Liming and Chris D Nugent (2009). 'Ontology-based activity recognition in intelligent pervasive environments.' In: *IJWIS* 5.4, pp. 410–430.

Chen, P, J Gu, D Zhu and F Shao (2013). 'A dynamic time warping based algorithm for trajectory matching in LBS'. In: *International Journal of Database Theory and Application* 6.3, pp. 39–48.

Chen, Ruizhi (2012). *Ubiquitous Positioning and Mobile Location-Based Services in Smart Phones*. IGI Global. ISBN: 1466618280.

Chorus, C and HJP Timmermans (2009). 'Empirical study into influence of travel behavior on stated and revealed mental maps'. In: *88th Transportation Research Compendium*.

Coors, V and U Jasnoch (1999). 'Using Wearable GIS in outdoor applications'. In: *Computer Graphik TOPICS* 2, pp. 11–12.

Cosgrove, Denis (1999). *Mappings*. Reaktion Books. ISBN: 1861898363.

DataRobot (2017). *Machine Learning Platform for Predictive Modeling | DataRobot*. URL: `https://www.datarobot.com/`.

Dean, Jeff and Urs Hölzle (2017). *Build and train machine learning models on our new Google Cloud TPUs*. URL: `https://blog.google/topics/google-cloud/google-cloud-offer-tpus-machine-learning/`.

Debord, Guy (1955). 'Introduction to a Critique of Urban Geography'. In: *Les Lèvres Nues* 6.

Dey, Anind K (2001). 'Understanding and Using Context.' In: *Personal and Ubiquitous Computing* 5.1, pp. 4–7.

Douglas, D H and T K Peucker (1973). 'Algorithms for the reduction of the number of points required to represent a digitized line or its caricature'. In: *Cartographica: The International Journal for Geographic Information and Geovisualization* 10.2, pp. 112–122.

Eckert, Max and W Joerg (1908). 'On the Nature of Maps and Map Logic'. In: *Bulletin of the American Geographical Society* 40.6, pp. 344–351.

Eco, Umberto (1963). Diario minimo. Ed. by Arnoldo Mondadori. ISBN: 88-04-32169-5.

Ende, Michael (1973). *Momo*. Stuttgart: Thienemann.

Ericson, Gary, Larry Franks and Brandon Rohrer (2017). *How to choose algorithms for Microsoft Azure Machine Learning*. URL: `https://docs.microsoft.com/en-us/azure/machine-learning/machine-learning-algorithm-choice`.

Facebook, Inc. (2017). *Verwenden der Graph API - Dokumentation - Facebook for Developers*. URL: `https://developers.facebook.com/docs/graph-api/using-graph-api/#search`.

Fisher, Kimberly (2017a). *China Time Use Survey*. URL: `http://timeuse-2009.nsms.ox.ac.uk/information/studies/data/china-2008.php`.

— (2017b). *Time Use Studies*. URL: `https://www.timeuse.org/`.

Fontenla-Romero, Oscar, David Martinez-Rego, Bertha Guijarro-Berdinas, Beatriz Perez-Sanchez and Diego Peteiro-Barral (2013). 'Online Machine Learning'. In: *Efficiency and Scalablity Methods for Computational Intellect*. Ed. by Boris Igelnik and Jacek M Zurada, pp. 27–54.

Foursquare (2017a). *About*. URL: `https://foursquare.com/about`.

— (2017b). *Foursquare API*. URL: `https://developer.foursquare.com/`.

— (2017c). *Foursquare Location Intelligence for Enterprise*. URL: `https://enterprise.foursquare.com/how-it-works`.

Furieri, Alessandro (2017). *SpatiaLite*. URL: `http://www.gaia-gis.it/gaia-sins/`.

Gabor, Dennis (1963). *Inventing the Future*. Secker & Warburg.

Gambs, Sébastien, Marc-Olivier Killijian and Miguel Núñez del Prado Cortez (2011). 'Show Me How You Move and I Will Tell You Who You Are.' In: *Trans. Data Privacy*, pp. 103–126.

— (2014). 'De-anonymization attack on geolocated data'. In: *Journal of Computer and System Sciences* 80.8, pp. 1597–1614.

Gatalsky, P, N Andrienko and G Andrienko (2004). 'Interactive analysis of event data using space-time cube'. In: *2000 IEEE Conference on Information Visualization. An International Conference on Computer Visualization and Graphics*, pp. 145–152.

Gerber, Simon, Michael Fry, Judy Kay, Bob Kummerfeld, Glen Pink and Rainer Wasinger (2010). 'PersonisJ - Mobile, Client-Side User Modelling.' In: *18th international conference on User Modeling, Adaptation, and Personalization*, pp. 111–122.

Goldberg, David, David Nichols, Brian M Oki and Douglas Terry (1992). 'Using collaborative filtering to weave an information tapestry'. In: *Communications of the ACM* 35.12, pp. 61–70.

Gomes, Lee (2014). *Machine-Learning Maestro Michael Jordan on the Delusions of Big Data and Other Huge Engineering Efforts*. URL: `http://spectrum.ieee.org/robotics/artificial-intelligence/machinelearning-maestro-michael-jordan-on-the-delusions-of-big-data-and-other-huge-engineering-efforts`.

Google, Android (2017). *Location Strategies*. URL: `https://developer.android.com/guide/topics/location/strategies.html`.

Google Inc. (2017a). *Places API Web Service*. URL: `https://developers.google.com/places/web-service/?hl=de`.

— (2017b). *Tensorflow Mobile*. URL: `https://www.tensorflow.org/mobile/`.

Gopal, Sucharita (2016). 'Artificial Neural Networks in Geospatial Analysis'. In: *The International Encyclopedia of Geography*. Oxford, UK: John Wiley & Sons, Ltd, pp. 1–7. ISBN: 9780470659632.

Gortana, F, S Kaim, M von Lupin and T Nagel (2014). 'Isoscope-Visualizing temporal mobility variance with isochrone maps'. In: *IEEE VIS 2014*.

Green, David R (2013). 'Journalistic Cartography: Good or Bad? A Debatable Point'. In: *The Cartographic Journal* 36.2, pp. 141–153.

Griffith, Richard John, henry Drury Harness and Thomas Aiskew Larcom (1838). *Atlas to accompany 2d report of the Railway Commissioners Ireland 1838*. Irish Railway Commission.

Gurman, Mark (2017). *Apple Is Working on a Dedicated Chip to Power AI on Devices*. URL: `https://www.bloomberg.com/news/articles/2017-05-26/apple-said-to-plan-dedicated-chip-to-power-ai-on-devices`.

Hägerstrand, Torsten (1970). 'What about People in Regional Science?' In: *Regional Sciences* 24.1, pp. 7–24.

Halpern, Orit (2014). *Beautiful data*. Duke University Press.

Hannak, Aniko, Piotr Sapiezynski, Arash Molavi Kakhki, Balachander Krishnamurthy, David Lazer, Alan Mislove and Christo Wilson (2013). 'Measuring personalization of web search.' In: *22nd international conference on World Wide Web*, pp. 527–538.

Hannigan, Joseph, Guillermo Hernandez, Richard M Medina, Patrick Roos and Paulo Shakarian (2013). 'Mining for Spatially-Near Communities in Geo-Located Social Networks.' In: *AAAI Technical Report* 1309, arXiv:1309.2900.

Hauger, Doug (2010). *Windows Azure General Availability*. URL: `https://blogs.technet.microsoft.com/microsoft_blog/2010/02/01/windows-azure-general-availability/`.

Haykin, Simon S (1998). *Neural networks: a comprehensive foundation; 2nd ed.* Englewood Cliffs, NJ: Prentice-Hall.

Heidmann, Frank (2013). 'Interaktive Karten und Geovisualisierungen'. In: *Interaktive Infografiken*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 39–69. ISBN: 978-3-642-15452-2.

Hintzman, D L, C S O'Dell and D R Arndt (1981). 'Orientation in cognitive maps.' In: *Cognitive Psychology* 13.2, pp. 149–206.

Hodgson, Dorothy L and Richard A Schroeder (2002). 'Dilemmas of Counter-Mapping Community Resources in Tanzania'. In: *Development and change* 33.1, pp. 79–100.

Hong, Jong-yi, Eui-ho Suh and Sung-Jin Kim (2009). 'Context-aware systems: A literature review and classification'. In: *Expert Systems with Applications* 36.4, pp. 8509–8522.

Horowitz, Andreessen (2017). *AI Playbook*. URL: http://aiplaybook.a16z.com/.

Hu, Yingjie, Krzysztof Janowicz, David Carral, Simon Scheider, Werner Kuhn, Gary Berg-Cross, Pascal Hitzler, Mike Dean and Dave Kolas (2013). 'A Geo-ontology Design Pattern for Semantic Trajectories'. In: *COSIT 2013: Proceedings of the 11th International Conference on Spatial Information Theory - Volume 8116*. BBN Technologies. Cham: Springer-Verlag New York, Inc., pp. 438–456. ISBN: 978-3-319-01789-1.

Investigator, Quote (2017). *We Cannot Predict the Future, But We Can Invent It*. URL: http://quoteinvestigator.com/2012/09/27/invent-the-future/.

Ireland, Data Protection Commissioner of (2003). 'Right to establish existence of personal data '. In: 3.

Isaac, Mike (2017). *How Uber Deceives the Authorities Worldwide*. URL: https://www.nytimes.com/2017/03/03/technology/uber-greyball-program-evade-authorities.html?_r=0.

Jentsch, Nicola, Sören Preibusch and Andreas Harasser (2012). *Study on monetising privacy*. Tech. rep. URL: https://www.enisa.europa.eu/publications/monetising-privacy.

Jeung, Hoyoung, Heng Tao Shen and Xiaofang Zhou (2007). 'Mining Trajectory Patterns Using Hidden Markov Models'. In: *Data Warehousing and Knowledge Discovery*. Berlin, Heidelberg: Springer, Berlin, Heidelberg, pp. 470–480. ISBN: 978-3-540-74552-5.

Johnson, I and B Hecht (2015). 'Structural causes of bias in crowd-derived geographic information: Towards a holistic understanding'. In: *Proceedings of the Association for the Advancement of Artificial Intelligence*.

Jouppi, Norm (2016). *Google supercharges machine learning tasks with TPU custom chip*. URL: https://cloudplatform.googleblog.com/2016/05/Google-supercharges-machine-learning-tasks-with-custom-chip.html.

Karpathy, Andrej (2014). *ConvNetJS - Deep learning in the browser*. URL: http://cs.stanford.edu/people/karpathy/convnetjs/.

— (2017). *CS231n Convolutional Neural Networks for Visual Recognition*. URL: http://cs231n.github.io/convolutional-networks/.

Kehr, Flavius, Daniel Wentzel and Tobias Kowatsch (2014). 'Privacy Paradox Revised - Pre-Existing Attitudes, Psychological Ownership, and Actual Disclosure.' In: *35th International Conference on Information Systems*.

Kharrat, Ahmed, Iulian Sandu Popa, Karine Zeitouni and Sami Faïz (2008). 'Clustering Algorithm for Network Constraint Trajectories.' In: *SDH* Chapter 36, pp. 631–647.

Kim, Won (2009). 'Cloud Computing - Today and Tomorrow.' In: *Journal of Object Technology* 8.1, pp. 65–72.

Kitchin, Robert M (1994). 'Cognitive maps: What are they and why study them?' In: *Journal of environmental psychology* 14.1, pp. 1–19.

Kogan, Gene and Francis Tseng (2016). *Machine Learning for Artists*. URL: `http://ml4a.github.io/`.

Kraak, Menno-Jan (2003). 'The space-time cube revisited from a geovisualization perspective'. In: *21st International Cartographic Conference*, pp. 1988–1996.

— (2008). 'Geovisualization and Time– New Opportunities for the Space–Time Cube'. In: *Geographic Visualization*. Chichester, UK: John Wiley & Sons, Ltd, pp. 293–306. ISBN: 9780470515112.

Kraak, Menno-Jan and Alexandra Koussoulakou (2005). 'A Visualization Environment for the Space-Time-Cube'. In: *Developments in Spatial Data Handling*. Berlin/Heidelberg: Springer-Verlag, pp. 189–200. ISBN: 3-540-22610-9.

Krempl, Stefan (2015). *Merkel auf dem IT-Gipfel: Datenschutz darf Big Data nicht verhindern*. URL: `http://www.heise.de/newsticker/meldung/Merkel-auf-dem-IT-Gipfel-Datenschutz-darf-Big-Data-nicht-verhindern-2980126.html`.

— (2016a). *Staatssekretär schießt scharf gegen das Motto "Meine Daten gehören mir"*. URL: `http://www.heise.de/newsticker/meldung/Staatssekretaer-schiesst-scharf-gegen-das-Motto-Meine-Daten-gehoeren-mir-3099732.html`.

— (2016b). *Zwei Jahre digitale Agenda: "Cloud hört sich an wie Stehlen"*. URL: `http://www.heise.de/newsticker/meldung/Zwei-Jahre-digitale-Agenda-Cloud-hoert-sich-an-wie-Stehlen-3314846.html`.

Krogh, A and J Vedelsby (1994). 'Neural network ensembles, cross validation, and active learning'. In: *7th International Conference on Neural Information Processing Systems*. unknown, pp. 231–238.

Kuipers, Benjamin (1978). 'Modeling Spatial Knowledge.' In: *Cognitive Science ()* 2.2, pp. 129–153.

Kuo, Mu-Hsing, Liang-Chu Chen and Chien-Wen Liang (2009). 'Building and evaluating a location-based service recommendation system with a preference adjustment mechanism'. In: *Expert Systems with Applications* 36.P2, pp. 3543–3554.

Kurenkov, Andrey (2015). *A 'Brief' History of Neural Nets and Deep Learning, Part 1*. URL: `http://www.andreykurenkov.com/writing/a-brief-history-of-neural-nets-and-deep-learning/`.

Kwan, M P (2007). 'Affecting geospatial technologies: Toward a feminist politics of emotion'. In: *The Professional Geographer* 59.1, pp. 22–34.

Kwan, M P and G Ding (2008). 'Geo-Narrative: Extending Geographic Information Systems for Narrative Analysis in Qualitative and Mixed-Method Research*'. In: *The Professional Geographer* 60.4, pp. 443–465.

Länder Forschungsdatenzentrum, Statistische Ämter des Bundes und der (2016). *Datenangebot | Zeitverwendungserhebung / Zeitbudgeterhebung*. URL: `http://www.forschungsdatenzentrum.de/bestand/zve/index.asp`.

Latour, Bruno (2005). *Reassembling the Social - An Introduction to Actor-Network-Theory*. Oxford University Press.

— (2011). 'Network Theory| Networks, Societies, Spheres: Reflections of an Actor-network Theorist'. In: *International Journal of Communication* 5.0, p. 15.

Lau, Sian Lun (2012). *Towards a User-centric Context Aware System: Empowering Users Through Activity Recognition Using a Smartphone as an Unobtrusive Device*. kassel university press GmbH. ISBN: 3862192458.

LeCun, Y, L Bottou and Y Bengio (1998a). 'Gradient-based learning applied to document recognition'. In: *Eurographics Conference on Visualization*. unknown, pp. 2278–2324.

LeCun, Yann, Corinna Cortes and Christopher J C Burges (1998b). *The MNIST Database of handwritten digits*. URL: `http://yann.lecun.com/exdb/mnist/`.

Lee, Wang-Chien and John Krumm (2011). 'Trajectory Preprocessing '. In: *Computing with Spatial Trajectories*. Ed. by Yu Zheng and Xiaofang Zhou. New York, NY: Springer Science & Business Media. ISBN: 1461416299.

Lee, Yun-mi, Jong-yi Hong, Won-il Oh and Hyeon Kang (2006). 'A Review of Context-Aware Computing'. In: *11th Annual Conference of Asia Pacific Decision Sciences*. Hong Kong, pp. 546–548.

Lee Peluso, Nancy (2011). 'Whose Woods are These? Counter-Mapping Forest Territories in Kalimantan, Indonesia'. In: *The Map Reader*. Chichester, UK: John Wiley & Sons, Ltd, pp. 422–429. ISBN: 9780470742839.

Leung, Kenneth Wai-Ting, Dik Lun Lee and Wang-Chien Lee (2011). 'CLR: a collaborative location recommendation framework based on co-clustering'. In: *the 34th international ACM SIGIR conference*. New York, New York, USA: ACM, pp. 305–314. ISBN: 978-1-4503-0757-4.

Levandoski, Justin J, Mohamed Sarwat, Ahmed Eldawy and Mohamed F Mokbel (2012). 'LARS: A Location-Aware Recommender System'. In: *ICDE '12: Proceedings of the 2012 IEEE 28th International Conference on Data Engineering*. IEEE Computer Society, pp. 450–461. ISBN: 978-1-4673-0042-1.

Lewis, Seth C and Seth C Lewis (2014). 'Journalism In An Era Of Big Data. Cases, concepts, and critiques'. In: *Digital Journalism* 3.3, pp. 321–330.

Li, Quannan, Yu Zheng, Xing Xie, Yukun Chen, Wenyu Liu and Wei-ying Ma (2008). 'Mining user similarity based on location history.' In: *16th ACM SIGSPATIAL international conference on Advances in geographic information systems*.

Liu, Qihua (2014). 'Context-aware Mobile Recommendation System Based on Context History'. In: *TELKOMNIKA Indonesian Journal of Electrical Engineering* 12.4, pp. 1–10.

Liu, Yong, Wei Wei, Aixin Sun and Chunyan Miao (2014). 'Exploiting Geographical Neighborhood Characteristics for Location Recommendation'. In: *the 23rd ACM International Conference*. New York, New York, USA: ACM Press, pp. 739–748. ISBN: 9781450325981.

Lloyd, S (1982). 'Least squares quantization in PCM'. In: *IEEE Transactions on Information Theory* 28.2, pp. 129–137.

Lou, Yin, Chengyang Zhang, Yu Zheng, Xing Xie, Wei Wang and Yan Huang 0002 (2009). 'Map-matching for low-sampling-rate GPS trajectories.' In: *16th ACM SIGSPATIAL international conference on Advances in geographic information systems*. ACM, pp. 352–361. ISBN: 978-1-60558-649-6.

Lynch, Kevin (1960). *The Image of the City*. Cambridge: The Technology Press & Harvard University Press.

MacEachren, Alan M (2004). *How Maps Work*. Representation, Visualization, and Design. Guilford Press. ISBN: 9781572300408.

Machine Learning Group at the University of Waikato (2016). *Weka 3: Data Mining Software in Java*. URL: http://www.cs.waikato.ac.nz/ml/weka/.

Maguire, E A, R S Frackowiak and C D Frith (1997). 'Recalling routes around london: activation of the right hippocampus in taxi drivers.' In: *The Journal of neuroscience : the official journal of the Society for Neuroscience* 17.18, pp. 7103–7110.

Mao, Yingchi, Haishi Zhong, Xianjian Xiao and Xiaofang Li (2017). 'A Segment-Based Trajectory Similarity Measure in the Urban Transportation Systems'. In: *Sensors* 17.3, pp. 524–16.

Marston, Sean, Zhi Li, Subhajyoti Bandyopadhyay, Juheng Zhang and Anand Ghalsasi (2011). 'Cloud computing — The business perspective'. In: *Decision Support Systems* 51.1, pp. 176–189.

Mayhew, Susan (2009). *A Dictionary of Geography*. 4th ed. Oxford University Press. ISBN: 9780199231805.

McCallum, Andrew, Kamal Nigam and Lyle H Ungar (2000). 'Efficient clustering of high-dimensional data sets with application to reference matching.' In: *6th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 169–178.

McMahan, Brendan and Daniel Ramage (2017). *Federated Learning: Collaborative Machine Learning without Centralized Training Data*. URL: https://research.googleblog.com/2017/04/federated-learning-collaborative.html.

McMaster, Robert Brainerd and K Stuart Shea (1992). *Generalization in Digital Cartography*. Association of American Geographers. ISBN: 9780892912094.

McQuoid, Julia and Martin Dijst (2012). 'Bringing emotions to time geography: the case of mobilities of poverty'. In: *Journal of Transport Geography* 23, pp. 26–34.

Meier, Sebastian (2016). 'The Marker Cluster:' in: *International Journal of Agricultural and Environmental Information Systems* 7.1, pp. 28–43.

— (2017). 'Enhancing Location Recommendation Through Proximity Indicators, Areal Descriptors, and Similarity Clusters'. In: *Progress in Location-Based Services 2016*. Cham: Springer, Cham, pp. 273–291. ISBN: 978-3-319-47288-1.

Meier, Sebastian and Katrin Glinka (2017). 'Psychogeography in the Age of the Quantified Self - Mental Map Modelling with Georeferenced Personal Activity Data'. In: *LNG&C - Advances in Cartography and GIScience*. Springer International Publishing, pp. 507–522.

Meier, Sebastian and Thomas Hirt (2008). 'Location based applications'. In: *Interfaces* 77, pp. 1–3.

Meier, Sebastian, Johannes Landstorfer, Julia Werner, Reto Wettach, Andre Knörig, Jonathan Cohen and Andreas Sommerwerk (2012). 'A Real-world Mobile Prototyping Framework for Location- and Context-based Services'. In: *Wireless Communication and Information: Mobile Gesellschaft*.

Meier, Sebastian, Frank Heidmann and Andreas Thom (2014a). 'A comparison of location search UI patterns on mobile devices.' In: *Mobile HCI*.

— (2014b). 'Heattile, a New Method for Heatmap Implementations for Mobile Web-Based Cartographic Applications'. In: *Thematic Cartography for the Society*. Cham: Springer International Publishing, pp. 33–44. ISBN: 978-3-319-08179-3.

Meng, Liqiu (2004). 'About Egocentric Geovisualisations'. In: *12th Int. Conf. on Geoinformatics*, pp. 1–8.

Minard, Charles Joseph (1845). *Carte de la circulation des voyageurs par voitures publiques sur les routes de la contrée où sera placé le chemin de fer de Dijon à Mulhouse*.

— (1862). *Carte figurative et approximative des quantités de coton en laine importées en Europe en 1858 et en 1862*.

— (1865). *Carte figurative et approximative des quantités de vin français exportés par mer en 1864*.

Mitchell, Katharyne and Sarah Elwood (2015). 'Counter-Mapping for Social Justice'. In: *Politics, Citizenship and Rights*. Ed. by Kirsi Kallio, Sarah Mills and Tracey Skelton. Springer Singapore, pp. 207–223. ISBN: 978-981-4585-94-1.

Mokbel, M, J Bao and A Eldawy (2011). 'Personalization, socialization, and recommendations in location-based services 2.0'. In: *5th International Conference on Personal Data Bases*.

Molenaar, Martin (1998). *An Introduction To The Theory Of Spatial Object Modelling For GIS*. CRC Press.

Mondschein, Andrew, Evelyn Blumenberg and Brian Taylor (2010). 'Accessibility and Cognition: The Effect of Transport Mode on Spatial Knowledge'. In: *Urban Studies* 47.4, pp. 845–866.

Monmonier, Mark (1991). *How to Lie with Maps*. Chicago & London: University of Chicago Press. ISBN: 0-226-53414-6.

Monmonier, Mark S (2004). *Rhumb lines and map wars : a social history of the Mercator projection*. Chicago : University of Chicago Press. ISBN: 0226534316.

Montello, Daniel R (2013). 'Cognitive Map-Design Research in the Twentieth Century: Theoretical and Empirical Approaches'. In: *Cartography and Geographic Information Science* 29.3, pp. 283–304.

Musumba, George Wamamu and Henry O Nyongesa (2013). 'Context awareness in mobile computing: A review'. In: *International Journal of Machine Learning and Applications* 2.1, pp. 1–10.

Naftaly, Ury, Nathan Intrator and David Horn (1997). 'Optimal ensemble averaging of neural networks'. In: *Network: Computation in Neural Systems* 8.3, pp. 283–296.

Netflix (2009). *Netflix Prize*. URL: http://www.netflixprize.com/.

Nielsen, Michael A (2015). *Neural Networks and Deep Learning*. URL: http://neuralnetworksanddeeplearning.com/index.html.

Nike (2017). *Nike+ Training Club App*. URL: http://www.nike.com/de/de_de/c/nike-plus/training-app.

Norberg, P A and D R Horne (2007). 'The privacy paradox: Personal information disclosure intentions versus behaviors'. In: *Journal of Consumer Affairs* 41.1, pp. 100–126.

Noulas, A, S Scellato and C Mascolo (2011a). 'Exploiting Semantic Annotations for Clustering Geographic Areas and Users in Location-based Social Networks.' In: *The social mobile web* 11.2.

Noulas, Anastasios, Salvatore Scellato, Cecilia Mascolo and Massimiliano Pontil (2011b). 'An Empirical Study of Geographic User Activity Patterns in Foursquare.' In: *7th International Conference on Neural Information Processing Systems*, pp. 570–573.

Otten, H, L Hildebrandt, T Nagel, M Dörk and B Müller (2015). 'Are there networks in maps? An experimental visualization of personal movement data'. In: *IEEE VIS 2015*.

Oxford, Centre for Time Use Research at the University of (2015). *United Kingdom Time Use Survey*. URL: https://discover.ukdataservice.ac.uk/series/?sn=2000054.

Paireekreng, W and K W Wong (2009). 'Client-side mobile user profile for content management using data mining techniques'. In: *2009 Eighth International Symposium on Natural Language Processing (SNLP)*. IEEE, pp. 96–100. ISBN: 978-1-4244-4138-9.

Papo, David, Javier M Buldú, Stefano Boccaletti and Edward T Bullmore (2014). 'Complex network theory and the brain'. In: *Philosophical Transactions Royal Society B* 369.1653.

Pascoe, Jason (1998). 'Adding Generic Contextual Capabilities to Wearable Computers.' In: *2nd International Symposium on Wearable Computers*.

Pennanen, Juho and Aapo Kyrölä (2013). 'User activity tracking system'. Pat. URL: https://www.google.com/patents/US20150004998.

Peter Sollich, Anders Krogh (1996). 'Learning with ensembles: How overfitting can be useful'. In: *Advances in Neural Information Processing Systems* 8.

Porter, Theodore M (1995). *Trust in Numbers*. The pursuit of objectivity in science and public life. Princeton, New Jersey: Princeton University Press.

PostGIS (2017). *PostGIS*. URL: `http://postgis.net/`.

PostgreSQL (2017). *PostgreSQL*. URL: `https://www.postgresql.org/`.

Preibusch, Sören (2013). 'Guide to measuring privacy concern: Review of survey and observational instruments'. In: *International Journal of Human-Computer Studies* 71.12, pp. 1133–1143.

ProtoGeo (2016). *Moves - Activity Diary for iPhone and Android*. URL: `https://www.moves-app.com/`.

Pyle, Dorian and Christina San Jose (2015). *An executive's guide to machine learning*. URL: `http://www.mckinsey.com/industries/high-tech/our-insights/an-executives-guide-to-machine-learning`.

Randell, David A, Zhan Cui and Anthony G Cohn (1992). 'A Spatial Logic based on Regions and Connection.' In: *3rd international conference on knowledge representation and reasoning*, pp. 165–176.

Rao, Bharat and Louis Minakakis (2003). 'Evolution of mobile location-based services'. In: *Communications of the ACM* 46.12, pp. 61–65.

Ravenstein, E G (1885). 'The Laws of Migration'. In: *Journal of the Statistical Society* 48.2.

Robinson, Arthur Howard, Joel L Morrison, Muehrcke, P. C., A Jon Kimerling and Stephen C Guptill (1995). *Elements of Cartography*. 6th ed. New York: John Wiley & Sons. ISBN: 978-0471555797.

Rösler, Roberto and Thomas Liebig (2013). 'Using Data from Location Based Social Networks for Urban Activity Clustering'. In: *Geographic Information Science at the Heart of Europe*. Cham: Springer International Publishing, pp. 55–72. ISBN: 978-3-319-00614-7.

Roxin, A, J Gaber, M Wack and A Nait-Sidi-Moh (2007). 'Survey of Wireless Geolocation Techniques'. In: *IEEE Globecom Workshops*. IEEE, pp. 1–9. ISBN: 978-1-4244-2024-7.

Ruder, Sebastian (2016). 'An overview of gradient descent optimization algorithms'. In: *arXiv.org*. arXiv: `1609.04747v1 [cs.LG]`.

Rulifson, Jeff (1969). *DEL*. URL: `https://tools.ietf.org/html/rfc5`.

Sager, Ira (2012). *Before IPhone and Android Came Simon, the First Smartphone*. URL: `https://www.bloomberg.com/news/articles/2012-06-29/before-iphone-and-android-came-simon-the-first-smartphone`.

Savage, Norma Saiph, Maciej Baranski, Norma Elva Chavez and Tobias Höllerer (2012). 'I'm feeling LoCo: A Location Based Context Aware Recommendation System'. In: *Advances in Location-Based Services*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 37–54. ISBN: 978-3-642-24197-0.

Schilit, B, N Adams and R Want (1994). 'Context-aware computing applications'. In: *Workshop on Mobile Computing Systems and Applications*. IEEE Comput. Soc. Press, pp. 85–90. ISBN: 0-8186-6345-6.

Schilit, B N and M M Theimer (1994). 'Disseminating active map information to mobile hosts'. In: *IEEE Network: The Magazine of Global Internetworking* 8.5, pp. 22–32.

Schmidhuber, Juergen (2014). 'Deep Learning in Neural Networks: An Overview'. In: *arXiv.org*, arXiv:1404.7828–117. arXiv: `1404.7828 [cs.NE]`.

Scholten, Christina, Tora Friberg and Annika Sandén (2012). 'Re-Reading Time-Geography from a Gender Perspective: Examples from Gendered mobility'. In: *Tijdschrift voor economische en sociale geografie* 103.5, pp. 584–600.

Seo, Y and J Ahn (2013). 'Novel Method for Enhancing Contents Recommendation Accuracy Using LBS-based Users Viewing Path Similarity'. In: *International Journal of Multimedia and Ubiquitous Engineering* 8.4, pp. 217–226.

Set (2017). *Set - The Human Forecasting Platform*. URL: `https://www.set.gl/`.

Sharma, Richa and Rahul Singh (2016). 'Evolution of Recommender Systems from Ancient Times to Modern Era: A Survey'. In: *Indian Journal of Science and Technology* 9.20, pp. 1–12.

Shi, Wenzhong and ChuiKwan Cheung (2013). 'Performance Evaluation of Line Simplification Algorithms for Vector Generalization'. In: *The Cartographic Journal* 43.1, pp. 27–44.

Silveira, Giovani J C da, Flavio S Fogliatto and Denis Borenstein (2001). 'Mass customization: Literature review and research directions'. In: *International Journal of Production Economics* 72.1, pp. 1–13. ISSN: 1860-5168.

Simpson, Thomas W (2012). 'Evaluating Google as an Epistemic Tool'. In: *Metaphilosophy* 43.4, pp. 426–445.

Sotomayor, Borja, Ruben Santiago Montero, Ignacio Martin Llorente and Ian Foster (2008). 'Capacity leasing in cloud systems using the opennebula engine'. In: *Workshop on Cloud Computing and its Applications*.

Spaccapietra, Stefano, Christine Parent, Maria Luisa Damiani, Jose Antonio de Macedo, Fabio Porto and Christelle Vangenot (2008). 'A conceptual view on trajectories'. In: *Data & Knowledge Engineering* 65.1, pp. 126–146.

Spek, Stefan van der, Jeroen van Schaick, Peter de Bois and Remco de Haan (2009). 'Sensing Human Activity: GPS Tracking'. In: *Sensors* 9.4, pp. 3033–3055.

SQLite (2017). *SQLite*. URL: `https://www.sqlite.org/`.

Stackoverflow (2010). *When to choose which machine learning classifier?* URL: `http://stackoverflow.com/questions/2595176/when-to-choose-which-machine-learning-classifier`.

Steinebach, Martin, Christian Winter, Oren Halvani, Marcel SChäfer and York Yannikos (2015). *Chancen durch Big Data und die Frage des Privatsphärenschutzes*. Tech. rep. Stuttgart.

Strang, Thomas and Claudia Linnhoff-Popien (2004). 'A Context Modeling Survey'. In: *1st International Workshop on Advanced Context Modelling, Reasoning and Management at UbiComp*. unknown.

Street, Nicholas (2006). *TimeContours: Using isochrone visualisation to describe transport network travel cost*. Tech. rep. URL: `http://www.imperial.ac.uk/pls/portallive/docs/1/18619712.PDF`.

Sturges, Herbert A (1926). 'The Choice of a Class Interval'. In: *Journal of the American Statistical Association* 21.153, pp. 65–66.

Sullivan, W C, Frances E Kuo and Stephen F DePooter (2004). 'The Fruit of Urban Nature: Vital Neighborhood Spaces'. In: *Environment and Behavior* 36.5, pp. 678–700.

Thudt, A, D Baur and S Carpendale (2013). 'Visits: A spatiotemporal visualization of location histories'. In: *Eurographics Conference on Visualization*.

Tiakas, E, A N Papadopoulos, A Nanopoulos, Y Manolopoulos, Dragan Stojanovic and Slobodanka Djordjevic-Kajan (2009). 'Searching for similar trajectories in spatial networks'. In: *The journal of system and software* 82.5, pp. 772–788.

Tiwari, Shivendra, Saroj Kaushik, Priti Jagwani and Sunita Tiwari (2011). 'A Survey on LBS: System Architecture, Trends and Broad Research Areas'. In: *7th international conference on Databases in Networked Information Systems*. Springer Berlin Heidelberg, pp. 223–241. ISBN: 978-3-642-25730-8.

Tobler, W R (1970). 'A computer movie simulating urban growth in the Detroit region'. In: *Economic geography* 46, p. 234.

Tsai, Wei-Tek, Xin Sun and Janaka Balasooriya (2010). 'Service-Oriented Cloud Computing Architecture'. In: *2010 Seventh International Conference on Information Technology: New Generations*. IEEE, pp. 684–689. ISBN: 978-1-4244-6270-4.

Tversky, Barbara (1981). 'Distortions in memory for maps'. In: *Cognitive Psychology* 13.3, pp. 407–433.

— (1993). 'Cognitive Maps, Cognitive Collages, and Spatial Mental Models.' In: *Spatial Information Theory A Theoretical Basis for GIS. COSIT 1993. Lecture Notes in Computer Science* 716.Chapter 2, pp. 14–24.

Varnelis, Kazys and A Friedberg (2008). 'Place: The Networking of Public Space'. In: *Networked Publics*. Mit Press.

Vertesi, J (2008). 'Mind the Gap: The London Underground Map and Users' Representations of Urban Space'. In: *Social Studies of Science* 38.1, pp. 7–33.

Visvalingam, Mahes and Peter J Williamson (1995). 'Simplification and Generalization of Large Scale Data for Roads: A Comparison of Two Filtering Algorithms'. In: *Cartography and Geographic Information Science* 22.4, pp. 264–275.

Vujakovic, Peter (2013). 'The State as a 'Power Container': The Role of News Media Cartography in Contemporary Geopolitical Discourse'. In: *The Cartographic Journal* 51.1, pp. 11–24.

Wang, Haohan and Bhiksha Raj (2017). 'On the Origin of Deep Learning'. In: *arXiv.org*, arXiv:1702.07800. arXiv: `1702.07800 [cs.LG]`.

Weiss, S M and I Kapouleas (1989). 'An empirical comparison of pattern recognition, neural nets and machine learning classification methods'. In: *Readings in machine learning*, pp. 781–787.

Windeck, Christof (2017). *ARM DynamIQ: Kürzere Latenzen, mächtigere AI-Befehle für Cortex-Chips*. URL: https://www.heise.de/newsticker/meldung/ARM-DynamIQ-Kuerzere-Latenzen-maechtigere-AI-Befehle-fuer-Cortex-Chips-3660723.html.

Winquist, K (2004). *How Europeans spend their time. Everyday life of women and men*. Rapport de la Commission européenne.

Wood, Dennis (1992). *The Power of Maps*. New York: The Guilford Press. ISBN: 978-0-89862-493-9.

Ye, Mao, Dong Shou, Wang-Chien Lee, Peifeng Yin and Krzysztof Janowicz (2011a). 'On the semantic annotation of places in location-based social networks'. In: *KDD '11: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. Pennsylvania State University. ACM.

Ye, Mao, Krzysztof Janowicz, Christoph Mülligann and Wang-Chien Lee (2011b). 'What you are is when you are - the temporal dimension of feature types in location-based social networks.' In: *16th ACM SIGSPATIAL international conference on Advances in geographic information systems*. ACM Press, pp. 102–111. ISBN: 9781450310314.

Yelp (2017). *Yelp Fusion / Search*. URL: https://www.yelp.com/developers/documentation/v3/business_search.

Yu, Y and X Chen (2015). 'A survey of point-of-interest recommendation in location-based social networks'. In: *Proceedings of the Association for the Advancement of Artificial Intelligence*.

Yuan, Quan, Gao Cong, Zongyang Ma, Aixin Sun and Nadia Magnenat-Thalmann (2013). 'Time-aware point-of-interest recommendation.' In: *36th international ACM SIGIR conference on Research and development in information retrieval*. ACM Press, pp. 363–372. ISBN: 9781450320344.

Zandbergen, Paul A (2009). 'Accuracy of iPhone Locations: A Comparison of Assisted GPS, WiFi and Cellular Positioning'. In: *Transactions in GIS* 13.4, pp. 5–25.

Zeiler, Matthew D and Rob Fergus (2013). 'Visualizing and Understanding Convolutional Networks'. In: *arXiv.org*, arXiv:1311.2901. arXiv: 1311.2901 [cs.CV].

Zhang, Jia-Dong and Chi-Yin Chow (2013). 'iGSLR: personalized geo-social location recommendation: a kernel density estimation approach'. In: *SIGSPATIAL'13: Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. The University of Hong Kong. New York, New York, USA: ACM, pp. 334–343. ISBN: 978-1-4503-2521-9.

Zhang, Junbo, Yu Zheng and Dekang Qi (2016a). 'Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction.' In: *AAAI Technical Report* 1610, arXiv:1610.00081.

Zhang, Junbo, Yu Zheng, Dekang Qi, Ruiyuan Li and Xiuwen Yi (2016b). 'DNN-based prediction model for spatio-temporal data'. In: *the 24th ACM SIGSPATIAL International Conference*. ACM Press. ISBN: 9781450345897.

Zheng, Y, Y Chen, X Xie and W Y Ma (2009). 'GeoLife2.0: a location-based social networking service'. In: *Mobile Data Management: Systems, Services and Middleware*. IEEE, pp. 357–358. ISBN: 978-1-4244-4153-2.

Zheng, Yu (2007). *GeoLife: Building Social Networks Using Human Location History*. URL: https://www.microsoft.com/en-us/research/project/geolife-building-social-networks-using-human-location-history/.

— (2015). 'Trajectory Data Mining'. In: *Transactions on Intelligent Systems and Technology* 6.3.

Zheng, Yu, Longhao Wang, Ruochi Zhang, Xing Xie and Wei-ying Ma (2008a). 'GeoLife - Managing and Understanding Your Past Life over Maps.' In: *9th International Conference on Mobile Data Management*, pp. 211–212.

Zheng, Yu, Like Liu, Longhao Wang and Xing Xie (2008b). 'Learning transportation mode from raw gps data for geographic applications on the web.' In: *22nd international conference on World Wide Web*, pp. 247–256.

Zhou, Zhi-Hua, Jianxin Wu and Wei Tang (2002). 'Ensembling neural networks - Many could be better than all.' In: *Artificial Intelligence* 137.1, pp. 239–263.

Zipf, A, V Chandrasekhara and J Häussler (2000). 'GIS hilft Touristen bei der Navigation–ein erster Prototyp des mobilen Deep Map Systems für das Heidelberger Schloß'. In: *HGG-Journal*.

Zontar, Rok, Marjan Henricko and Ivan Rozman (2012). 'Taxonomy of context-aware systems'. In: *ELEKTROTEHNISKI VESTNIK* 79.1, pp. 41–46.

APPENDIX

MOVES SAMPLE GEOJSON

```
{
  "type" : "FeatureCollection",
  "features" : [ {
    "type" : "Feature",
    "geometry" : {
      "type" : "Point",
      "coordinates" : [ 13.43459, 52.48248 ]
    },
    "properties" : {
      "type" : "place",
      "startTime" : "20131231T010128+0100",
      "endTime" : "20140101T013834+0100",
      "place" : {
        "id" : 96792719,
        "name" : "Home",
        "type" : "user",
        "location" : {
          "lat" : 52.48248,
          "lon" : 13.43459
        }
      },
      "activities" : [ {
        "activity" : "walking",
        "group" : "walking",
        "manual" : false,
        "startTime" : "20140101T001222+0100",
        "endTime" : "20140101T001252+0100",
        "duration" : 30.0,
        "distance" : 35.0,
        "steps" : 71,
        "calories" : 2
      } ],
      "lastUpdate" : "20140408T121729Z",
      "date" : "20140101"
    }
  },
  [...]
  {
    "type" : "Feature",
    "geometry" : {
      "type" : "MultiLineString",
      "coordinates" : [ [ [ 13.43459, 52.48248 ], [ 13.41223, 52.47567 ], [...] ] ]
    },
    "properties" : {
      "type" : "move",
      "startTime" : "20140103T155125+0100",
      "endTime" : "20140103T164701+0100",
      "activities" : [ {
        "activity" : "transport",
        "group" : "transport",
        "manual" : false,
        "startTime" : "20140103T155125+0100",
        "endTime" : "20140103T160805+0100",
        "duration" : 1000.0,
        "distance" : 3616.0
      }, {
        "activity" : "walking",
        "group" : "walking",
        "manual" : false,
        "startTime" : "20140103T160805+0100",
        "endTime" : "20140103T163219+0100",
        "duration" : 1454.0,
        "distance" : 2398.0,
        "steps" : 2107,
        "calories" : 153
      }, {
        "activity" : "transport",
        "group" : "transport",
        "manual" : false,
        "startTime" : "20140103T163218+0100",
        "endTime" : "20140103T163548+0100",
        "duration" : 210.0,
        "distance" : 1573.0
      }, {
        "activity" : "walking",
        "group" : "walking",
        "manual" : false,
        "startTime" : "20140103T163548+0100",
        "endTime" : "20140103T164701+0100",
        "duration" : 673.0,
        "distance" : 890.0,
        "steps" : 1060,
        "calories" : 57
      } ],
      "lastUpdate" : "20140103T194321Z",
      "date" : "20140103"
    }
  } ]
}
```

geolife sample dataset

Example PLT file

```
Geolife trajectory
WGS 84
Altitude is in Feet
Reserved 3
0,2,255,My Track,0,0,2,8421376
0
39.921712,116.472343,0,13,39298.1462037037,2007-08-04,03:30:32
39.921705,116.472343,0,13,39298.1462152778,2007-08-04,03:30:33
39.921695,116.472345,0,13,39298.1462268519,2007-08-04,03:30:34
39.921683,116.472342,0,13,39298.1462384259,2007-08-04,03:30:35
39.921672,116.472342,0,13,39298.14625,2007-08-04,03:30:36
```

Example Labels Text-File

```
Start Time  End Time  Transportation Mode
2007/06/26 11:32:29 2007/06/26 11:40:29 bus
2008/03/28 14:52:54 2008/03/28 15:59:59 train
2008/03/28 16:00:00 2008/03/28 22:02:00 train
```

location labels from the time use survey

Listing 4: Full list of labels in the UK time use survey for locations and their reassigned location label 0 d̄ropped

```
Value = 0.0  nLabel = 0  Label = Unspecified location
Value = 10.0 nLabel = 0 Label = Unspecified location (not travelling)
Value = 11.0 nLabel = Home Label = Home
Value = 12.0 nLabel = Home Label = Second home or weekend house
Value = 13.0 nLabel = Work Label = Working place or school
Value = 14.0 nLabel = OtherHome Label = Other peoples home
Value = 15.0 nLabel = Food Label = Restaurant cafe or pub
Value = 16.0 nLabel = Leisure Label = Sports facility
Value = 17.0 nLabel = Leisure Label = Arts or cultural centre
Value = 18.0 nLabel = Leisure Label = Parks countryside seaside beach or
     coast
Value = 19.0 nLabel = Shopping Label = Shopping centres markets other
    shops
Value = 20.0 nLabel = 0 Label = Hotel guesthouse camping site
Value = 21.0 nLabel = 0 Label = Other specified location (not travelling
    )
Value = 30.0 nLabel = Transport Label = Unspecified private transport
    mode
```

```
Value = 31.0 nLabel = Transport Label = Travelling on foot
Value = 32.0 nLabel = Transport Label = Travelling by bicycle
Value = 33.0 nLabel = Transport Label = Travelling by moped motorcycle
    or motorboat
Value = 34.0 nLabel = Transport Label = Travelling by passenger car as
    the driver
Value = 35.0 nLabel = Transport Label = Travelling by passenger car as a
     passenger
Value = 36.0 nLabel = Transport Label = Travelling by passenger car -
    driver status unspecified
Value = 37.0 nLabel = Transport Label = Travelling by lorry or tractor
Value = 38.0 nLabel = Transport Label = Travelling by van
Value = 39.0 nLabel = Transport Label = Other specified private
    travelling mode
Value = 40.0 nLabel = Transport Label = Unspecified public transport
    mode
Value = 41.0 nLabel = Transport Label = Travelling by taxi
Value = 42.0 nLabel = Transport Label = Travelling by bus
Value = 43.0 nLabel = Transport Label = Travelling by tram or
    underground
Value = 44.0 nLabel = Transport Label = Travelling by train
Value = 45.0 nLabel = Transport Label = Travelling by aeroplane
Value = 46.0 nLabel = Transport Label = Travelling by boat or ship
Value = 47.0 nLabel = Transport Label = Travelling by coach
Value = 48.0 nLabel = Transit Label = Waiting for public transport
Value = 49.0 nLabel = 0 Label = Other specified public transport mode
Value = 90.0 nLabel = 0 Label = Unspecified transport mode
Value = 99.0 nLabel = 0 Label = Illegible location or transport mode
Value = -9.0 nLabel = 0 Label = No answer/refused
Value = -7.0 nLabel = 0 Label = Interview not achieved
Value = -2.0 nLabel = 0 Label = Schedule not applicable
```

DECLARATION

I hereby guarantee, that I have created this work by myself and have only used the referenced tools and sources. This work has not been published yet and has not been submitted to another university.

Potsdam, Germany, June 2017

Sebastian Meier