An algorithmic approach for collaborative-based prediction of user contexts in ubiquitous environments under consideration of legal implications

Dissertation

zur Erlangung des akademischen Grades eines Doktors
der Ingenieurwissenschaften (Dr.-Ing.)
im Fachbereich Elektrotechnik/Informatik der
Universität Kassel

von
Herrn M.Sc. Christian Voigtmann
aus
Kassel

Datum der Einreichung: Januar 2014
Datum der mündlichen Prüfung: 29. Januar 2014

Erster Gutachter: Prof. Dr. Klaus David
Zweiter Gutachter: Prof. Dr. Bernhard Sick

Acknowledgement

I am grateful for all the help and advices I got the last 3 1/2 years during my work for ComTec. First, I would like to thank Prof. Dr. Klaus David, who enabled me to work and participate in very interesting projects and research works and who always asked the right questions.

In particular, I would like to thank Dr. Sian Lun Lau who gave me a lot of support in the beginnings of my research work and more importantly who helped me to structure my ideas in the first place. Further, I would like to thank Immanuel König and Dr. Bernd Niklas Klein who always offered a sympathetic ear for discussions with regard to my research topics.

In addition, I am deeply thankful to my beloved wife Annegret, who always read my publications.

Finally, I would like to express my heart-felt gratitude to my friends and my family and I would also like to thank them for their support.

Abstract

Context prediction is used to proactively adapt, e.g. services in a ubiquitous environment to users' needs. Due to the fact that context prediction enables proactiveness, the significance for ubiquitous computing systems is high. To the best of our knowledge, well-known approaches in context prediction only focus on a user's history as a database whose next contexts have to be predicted. In case a user suddenly changes her behaviour in an unexpected way and does not follow her routine anymore, the context history of the user does not contain appropriate context information to provide reliable context predictions. Hence, context prediction algorithms that only rely on the user's context history whose context has to be predicted, might fail. To overcome the gap of missing context information in a user's context history, the Collaborative Context Prediction (CCP) approach is proposed. CCP takes advantage of existing direct and indirect relations, which may exist among the context histories of various users and therefore provides the possibility to forecast a user's next context even if the user suddenly changes her expected routine. CCP is based on the Higher-order Singular Value Decomposition, which has already successfully been applied in existing recommendation systems. To provide an evaluation of CCP, it is assessed in three different experiments. In these experiments, results are carried out with respect to prediction accuracy. These results are compared to the results received by three state of the art context prediction approaches: the Alignment predictor, the StatePredictor and the ActiveLeZi prediction approach. In all three experiments, collaborative data sets are used as a basis for evaluation.

Moreover, CCP is applied to a realistic collaborative use case, the proactive protection of pedestrians. CCP is used to proactively detect pedestrians that might be at risk to collide with a car nearby, using real movement data, measured by smartphones the pedestrians carried in their trouser pocket.

Due to the fact that context prediction approaches primarily use personal contexts such as location data or users' behaviour patterns, legal evaluation criteria are derived considering the principles of a user's right to informational self-determination. Based on the derived legal evaluation criteria, the CCP approach and the state of the art context prediction approaches are examined. The evaluation results outline the compatibility of different context prediction approaches to a user's right to informational self-determination. Finally, an approach for distributed and collaborative context prediction is presented in this thesis. This approach presents a possibility to overcome the identified legal problems caused by context prediction, especially by collaborative-based context prediction.

Zusammenfassung

Mit Hilfe der Vorhersage von Kontexten können z. B. Dienste innerhalb einer ubiquitären Umgebung proaktiv an die Bedürfnisse der Nutzer angepasst werden. Aus diesem Grund hat die Kontextvorhersage einen signifikanten Stellenwert innerhalb des 'ubiquitous computing'. Nach unserem besten Wissen, verwenden gängige Ansätze in der Kontextvorhersage ausschließlich die Kontexthistorie des Nutzers als Datenbasis, dessen Kontexte vorhersagt werden sollen. Im Falle, dass ein Nutzer unerwartet seine gewohnte Verhaltensweise ändert, enthält die Kontexthistorie des Nutzers keine geeigneten Informationen, um eine zuverlässige Kontextvorhersage zu gewährleisten. Daraus folgt, dass Vorhersageansätze, die ausschließlich die Kontexthistorie des Nutzers verwenden, dessen Kontexte vorhergesagt werden sollen, fehlschlagen könnten. Um die Lücke der fehlenden Kontextinformationen in der Kontexthistorie des Nutzers zu schließen, führen wir den Ansatz zur kollaborativen Kontextvorhersage (CCP) ein. Dabei nutzt CCP bestehende direkte und indirekte Relationen, die zwischen den Kontexthistorien der verschiedenen Nutzer existieren können, aus. CCP basiert auf der Singulärwertzerlegung höherer Ordnung, die bereits erfolgreich in bestehenden Empfehlungssystemen eingesetzt wurde. Um Aussagen über die Vorhersagegenauigkeit des CCP Ansatzes treffen zu können, wird dieser in drei verschiedenen Experimenten evaluiert. Die erzielten Vorhersagegenauigkeiten werden mit denen von drei bekannten Kontextvorhersageansätzen, dem 'Alignment' Ansatz, dem 'StatePredictor' und dem 'ActiveLeZi' Vorhersageansatz, verglichen. In allen drei Experimenten werden als Evaluationsbasis kollaborative Datensätze verwendet.

Anschließend wird der CCP Ansatz auf einen realen kollaborativen Anwendungsfall, den proaktiven Schutz von Fußgängern, angewendet. Dabei werden durch die Verwendung der kollaborativen Kontextvorhersage Fußgänger frühzeitig erkannt, die potentiell Gefahr laufen, mit einem sich nähernden Auto zu kollidieren. Als kollaborative Datenbasis werden reale Bewegungskontexte der Fußgänger verwendet. Die Bewegungskontexte werden mittels Smartphones, welche die Fußgänger in ihrer Hosentasche tragen, gesammelt.

Aus dem Grund, dass Kontextvorhersageansätze in erster Linie personenbezogene Kontexte wie z.B. Standortdaten oder Verhaltensmuster der Nutzer als Datenbasis zur Vorhersage verwenden, werden rechtliche Evaluationskriterien aus dem Recht des Nutzers auf informationelle Selbstbestimmung abgeleitet. Basierend auf den abgeleiteten Evaluationskriterien, werden der CCP Ansatz und weitere bekannte kontextvorhersagende Ansätze bezüglich ihrer Rechtsverträglichkeit untersucht. Die Evaluationsergebnisse zeigen die rechtliche Kompatibilität der untersuchten Vorhersageansätze bezüglich des Rechtes des Nutzers auf informationelle Selbstbestimmung auf. Zum Schluss wird in der Dissertation ein Ansatz für die verteilte und kollaborative Vorhersage von Kontexten vorgestellt. Mit Hilfe des Ansatzes wird eine Möglichkeit aufgezeigt, um den identifizierten rechtlichen Probleme, die bei der Vorhersage von Kontexten und besonders bei der kollaborativen Vorhersage von Kontexten, entgegenzuwirken.

Contents

Chapter 1

Introduction

The surroundings, users of ubiquitous computer systems live and work in, are constantly changing. Beginning with the vision of Mark Weiser [1] in the early nineties that predicts that the most profound technologies will be those that disappear, the basic idea and motivation to begin the journey to ubiquitous computing has been started. In 1994, Schilit et al. [2] proposed first ideas how context aware applications and techniques that utilise contexts of users, gained from inconspicuous technologies in intelligent spaces, can look like. In 1999, Dey and Abowd [3] defined the meaning of context to be "any information that can be used to characterise the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves." This definition extends the term context-aware, first outlined by Schilit and Theimer [4], who understand the term context in a way that it describes locations, identities of nearby people and objects and changes to those objects. Based on the definition of Dey and Abowd, GPS information was used by Asbrook and Starner [5] to represent context as location of a user in 2002. In 2003 the increasing use of personal context information by location-aware applications inspired researches such as Beresford et al.

[6] for the first time, to start investigations how pervasive and ubiquitous computing affects the user's privacy. In the same year, research in smart homes using environmental context information to support the user by letting the house act as an intelligent agent became more and more popular. On the basis of the idea of the neural network house, Cook et al. [7] presented an agent-based smart home using contexts to automatically assist their inhabitants. To extend the possibilities of ubiquitous environments to derive user context information, sensors became wearable. For instance, through wearable acceleration sensors to derive a user's movement activity even if the user is not inside a smart home [8], outlined by Bao and Intille in 2004. An extension of wearable sensors has been provided by smartphones. In contrast to wearable sensors that are placed on a user's body for a specific purpose, a smartphone is carried by the user mostly all the time. Additionally, a smartphone provides access to contexts that describe the current situation of a user such as her calendar. In 2005 Mika Raento et al. proposed a platform [9] to provide access to these contexts by context-aware mobile applications. Through the continuous improvement of smartphones they have become more and more important for context-aware applications that run in ubiquitous environments. Smartphones are not limited to only provide access to software-based context information but can also provide access to hardware-based context information gained from built-in hardware sensors such as an accelerometer, a gyroscope, or a barometer. Using hardware sensors, context information, such as a user's movement behaviour [10] as shown by Sian Lun Lau et al. or the floor a user is currently located [11] presented by Salvatore Vanini et al., can be automatically inferred. Moreover, today's smartphones are equipped with sufficient power that context data can be processed directly on the smartphone without transmitting the context data to an external server system. Nowadays smartphones find applications in nearly every ubiquitous computing environment because they enable the user to be always available and online, which allows applications to continuously access contexts of a user. Hence, the

vision of ubiquitous computing outlined nearly 20 years ago emerged a wide range of real and ubiquitous implementations that strongly influence our daily routines.

Examples of ubiquitous computing at the present time are systems that automatically adapt to a user's need without requiring a user's explicit input. These can be environments like intelligent and smart spaces such as meeting rooms, smart homes, cars, or public places like hospitals, shopping malls, transit stations or airports. Depending on a user's perspective, even a first person shooter can be considered as a ubiquitous environment. The automated adaption to the needs of the users is basically achieved by deriving, processing and utilising user related contexts collected by a variety of unobtrusive sensor technologies. Sensors can be mounted hardware-based sensors (infrared, motion, temperature, wlan, etc.), wearable hardware-based sensors (RFID, GPS, sensors provided by smartphones, etc.) or software-based sensors such as calendars, notes or digital phone books, etc. According to the high presence of ubiquitous environments and the high amount of sensors used in them, it would be true to say that ubiquitous computing becomes more and more ubiquitous.

To apply user related information that has been gathered by unobtrusive sensors to assist the user in a ubiquitous environment, approaches for context recognition and context prediction are required. Context recognition is used to interpret raw sensor data by transforming them to a higher abstraction level. An example is the transformation of acceleration values of the x-, y-, and z-axis, derived from an acceleration sensor integrated in a smartphone carried by a user, to the current movement behaviour of the user (sitting, standing, walking) [10]. The recognised contexts can be used to automatically adapt an application or a service that run in the ubiquitous environment to the user's needs. With the usage of context prediction approaches it is possible to proactively adapt an application or a service. This enables an application or a service to be adapted before the appropriate context of the user has been recognised by context recognition approaches. Hence, it grants a certain time advantage to the system. An application example is the proactive adaption of energetic consumers like the PC, the lighting or the heater based on the predicted movement behaviours of a user [12]. To enable the prediction of a user's next context, knowledge, represented by previously recognised contexts, is needed. The knowledge is stored in the context history of the user. Subsequently, the next context of a user can be predicted by appropriate context prediction approaches using a user's context history and a user's last recognised contexts. Well known context prediction approaches like Alignment [13] or ActiveLeZi [14] utilise explicit knowledge of a user which is stored in the context history of the user. If the history of the user whose next context has to be predicted does not provide sufficient context data, this is the case, if the user suddenly changes her behaviour patterns, current prediction approaches will fail to predict the next context. The goal of this thesis is to examine a collaborative-based context prediction approach to provide valid context prediction results even if the user's own context history does not provide sufficient context data. The objective of the collaborative-based context prediction approach is not only to use the context history of the user whose next context has to be predicted but to use context histories of additional users whose context histories show sufficient correlations. Due to the implicit usage of correlated and personal context in the collaborative-based context prediction process, the thesis also discusses and evaluates resulting legal consequences to existing context prediction approaches and the proposed colloborative-based context predictor.

1.1 Problem statements

In contrast to existing state of the art context prediction approaches, the research in this thesis concentrates on collaborative-based context prediction. In this thesis the term collaboration is defined as follows: Collaboration means the explicit utilisation of contexts of multiple users who are located in the same ubiquitous environment to provide a more robust context prediction to an individual user. Further, collaboration differs to the term cooperation. In our opinion cooperation defines a process whereby a user's individual context does not necessarily depends from contexts of other users but from aggregated or fused context information from local sensors.

Until now, present approaches neglected the integration of context histories of additional users into the context prediction process for a user that behaves in a ubiquitous environment. Thereby, it might be obvious that users in the same ubiquitous environment or users using the same ubiquitous computing application may assist each other because they have similar objectives and therefore behave the same way. Assistance can be achieved by providing own context information to the context prediction process of another user. The usage of additional information provided by other users is already commonly used in existing crowed sourcing approaches used for example for tag recommendation in social systems like Flickr, Last.fm, delicious, amazon, etc. Hence, mostly interests of a user are compared to the interest of other users to identify additional information a user might also be interested in but has not taken into consideration so far. To evolve context prediction to collaborative-based context prediction, indirect and latent information of other users has to be utilised. This is can be achieved by comparing context data stored in a user's context history to context data of other users stored in their context histories using appropriate mechanisms.

In ubiquitous computing not only technical considerations like availability, accuracy, scalability or for example efficiency are important to gain a user's trust. Also legal implications caused by ubiquitous computing systems should be taken into consideration. That privacy in ubiquitous computing systems is not only a legal or social issue, due to the reason that privacy and technology are closely intertwined, has already been outlined by Marc Langheinrich in [15, 16]. Rene Mayrhofer has also called attention to privacy issues caused by context prediction processes [17]. Even Mark

Weiser has already identified a user's privacy as one of the biggest challenges in the realisation of his vision of ubiquitous computing in 1991 [1]. In particular, the unobtrusiveness of getting personal context data, provided by inconspicuous sensors placed all over in a ubiquitous environment and the automated usage of the received context data by services or applications, prevents existing solutions to become publicly reality. This is because the German privacy law simply prohibits the implicit or explicit usage of personal data without the consent of the user the data belongs to. Hence, the only possibility would be to ask the user for her consent if her private data has to be collected, which would drastically limit the power of context awareness. Due to the reason that privacy is a complex issue in ubiquitous computing this thesis focuses on the analysis of how current context prediction approaches and the extension to collaborative-based context prediction affects the German privacy law. In detail the right to informational self-determination is considered and used to evaluate context prediction approaches with respect to their compatibility.

1.2 Contributions

This thesis contributes to the motivation and understanding of collaborative-based context prediction. Further, it raises the reader's awareness for challenges regarding privacy and trust that come along with collaborative-based context prediction. Besides, the thesis outlines a possible solution that tries to bring collaborative-based context prediction in line with the right to informational self-determination.

Based on existing approaches to context prediction, the collaborative-based context prediction (CCP) is motivated and its mathematical definitions are outlined and discussed. Next, CCP is practically illustrated and evaluated in three different experiments. The used data sets in the experiments differ with regard to their sizes, their use cases and the way they have been collected in.

Apart from the evaluation of context prediction approaches, the basic motivation of context prediction is discussed and examined with respect to its compatibility to the right to informational self-determination. Consequently, legal evaluation criteria are derived from the right to informational self-determination. The criteria are used to legally assess existing context prediction approaches. Further, KORA, a method to integrate and consider legal requirements in the design process of informational technology, is used. As a result of KORA, technical design proposals are derived. These technical design proposals are further used to produce a context prediction approach that can be considered as legally acceptable.

To provide an example of the usefulness of CCP, it is applied to a real world use case. In this use case CCP is utilised to proactively filter endangered pedestrians out of potentially many pedestrians. Therefore, realistic movement data have been collected using smartphones. Later, the derived movement behaviours of the pedestrians have been used to proactively predict their next step on the pavement. Collision avoidance systems can profit of the time advantage gained by proactively detecting endangered pedestrians.

Based on the derived design proposals received by using the KORA method, an approach for distributed and collaborative context prediction has been developed. This approach consists of two architectures. The first architecture enables users to collect their own context data using their own smartphones. Based on the collected context data the second architecture enables the users to collaboratively predict their next contexts, without disregarding their right to informational self-determination.

The results of this thesis have been tested and published in two national projects: Pervasive Energie durch internetbasierte Telekommunikationsdienste (Pinta) [18] and Gestaltung technisch-sozialer Vernetzung in situativen ubiquitären Systemen (Venus) [19].

1.3 Outline of the thesis

This thesis includes the content of all the research work I have submitted and presents their contributions within a logical context. The thesis is organised in seven chapters. The first chapter provides the introduction of the thesis and gives a motivating overview of the problem definition, Chapter 2 outlines the background of context prediction and summarises the state of the art. First, existing context prediction approaches are outlined and discussed with regard to location-based context prediction. Second, existing frameworks for context prediction are presented. Finally, research works with focus on trust and privacy in ubiquitous computing systems are introduced. Results outlined in this chapter are partially based on [20].

Chapter 3 introduces the approach of collaborative-based context prediction. Findings presented in this chapter are based on [21, 22]. In the beginning of the chapter the mathematical fundamentals of the approach are presented. Next, the approach is illustrated and afterwards it is evaluated in three different experiments.

Chapter 4 introduces the right to informational self-determination. Further, legal evaluation criteria are derived and used to legally evaluate existing context prediction approaches. Later, design principles for a compatible context prediction approach are created using the KORA method. Results and findings presented in this chapter are based on [23, 24].

In the next chapter the collaborative-based context predictor is applied in a real use case, which is based on [25]. The predictor is used to proactively filter pedestrians whose next step brings them close to the street, to provide collision avoidance systems, e.g. installed in cars, with an extra time advantage.

Finally, in Chapter 6 an approach for distributed and collaborative context prediction is presented. The presented approach is based on [26]. It addresses the legal implications caused by existing context predictors, before Chapter 7 concludes the findings of this thesis.

1.4 Publications

The publications conducted within the scope of this thesis have been published in conferences and workshops. These publications are as follows:

- C. Voigtmann, K. David, J. Zirfas, H. Skistims, and A. Roßnagel, "Prospects for Context Prediction Despite the Principle of Informational Self-Determination," in Advances in Human-Oriented and Personalized Mechanisms, Technologies and Services (CENTRIC), Nice, 2010, pp. 89–92.
- C. Voigtmann, S. L. Lau, and K. David, "An approach to Collaborative Context Prediction," in Pervasive Computing and Communications Workshops (PerCom Workshops) CoMoRea, Seattle, USA, 2011, pp. 438–443.
- C. Voigtmann, S. L. Lau, and K. David, "A Collaborative Context Prediction Technique," in Vehicular Technology Conference (VTC2011-Spring) 73rd, Budapest, Hungary, 2011, pp. 1–5.
- H. Skistims, C. Voigtmann, K. David, and A. Roßnagel, "Datenschutzgerechte Gestaltung von kontextvorhersagenden Algorithmen," vol. Datenschutz und Datensicherheit 36, pp. 31–16, 2012.
- C. Voigtmann, and K. David, "A Survey To Location-Based Context Prediction", Workshop on recent advances in behavior prediction and pro-active pervasive computing (AwareCast) in conjunction with the 10th International Conference on Pervasive Computing (Pervasive 2012), New Castle, UK, June 19th, 2012.
- C. Voigtmann, S. L. Lau, and K. David, "Evaluation of a collaborative-based filter technique to proactively detect pedestrians at risk," in Vehicular Technology Conference (VTC2012-Fall), Quebec City, QC, 2012, pp. 1–5.

- C. Voigtmann, K. David, H. Skistims, and A. Roßnagel, "Legal assessment of context prediction techniques," in Vehicular Technology Conference (VTC Fall), Quebec City, QC, 2012, pp. 1–5.
- C. Voigtmann, C. Schütte, A. Wacker, and K. David, "A new approach for distributed and collaborative context prediction," at the 10th IEEE Workshop on Context Modeling and Reasoning (CoMoRea), San Diego, USA, March 2013.

References

[1] M. Weiser, "The computer for the 21st century," *Scientific American*, vol. 265, pp. 66–75, Sept. 1991.

[2] B. Schilit, N. Adams, and R. Want, "Context-aware computing applications," in *Proceedings of the 1994 First Workshop on Mobile Computing Systems and Applications*, WMCSA '94, (Washington, DC, USA), pp. 85–90, IEEE Computer Society, 1994.

[3] A. K. Dey and G. D. Abowd, "Towards a better understanding of context and context-awareness," in *In HUC'99: Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing*, pp. 304–307, Springer-Verlag, 1999.

[4] B. N. Schilit and M. M. Theimer, "Disseminating active map information to mobile hosts," *Network, IEEE*, vol. 8, no. 5, pp. 22–32, 1994.

[5] D. Ashbrook and T. Starner, "Learning significant locations and predicting user movement with GPS," in *Wearable Computers, 2002. (ISWC 2002). Proceedings. Sixth International Symposium on*, pp. 101–108, 2002.

[6] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *Pervasive Computing, IEEE*, vol. 2, no. 1, pp. 46–55, 2003.

[7] D. Cook, M. Youngblood, I. Heierman, E.O., K. Gopalratnam, S. Rao, A. Litvin, and F. Khawaja, “MavHome: an agent-based smart home,” in *Pervasive Computing and Communications, 2003. (PerCom 2003). Proceedings of the First IEEE International Conference on*, pp. 521–524, Mar. 2003.

[8] L. Bao and S. S. Intille, “Activity recognition from user-annotated acceleration data,” in *Pervasive Computing*, pp. 1–17, Springer, 2004.

[9] M. Raento, A. Oulasvirta, R. Petit, and H. Toivonen, “ContextPhone: a prototyping platform for context-aware mobile applications,” *Pervasive Computing, IEEE*, vol. 4, no. 2, pp. 51–59, 2005.

[10] S. L. Lau and K. David, “Movement recognition using the accelerometer in smartphones,” in *Future Network and Mobile Summit, 2010*, pp. 1–9, 2010.

[11] S. Vanini and S. Giordano, “Adaptive context-agnostic floor transition detection on smart mobile devices,” in *10th IEEE Workshop on Context Modeling and Reasoning 2013 (CoMoRea 2013)*, (San Diego, USA), IEEE, Mar. 2013.

[12] “pinta-it2green: Startseite.” `http://www.pinta-it2green.de/`, last accessed: 2013-02-11.

[13] S. Sigg, S. Haseloff, and K. David, “An alignment approach for context prediction tasks in UbiComp environments,” *Pervasive Computing, IEEE*, vol. 9, pp. 90–97, Dec. 2010.

[14] K. Gopalratnam and D. J, “Active lezi: An incremental parsing algorithm for sequential prediction,” in *In Sixteenth International Florida Artificial Intelligence Research Society Conference*, pp. 38–42, 2003.

[15] M. Langheinrich, “Privacy by design - principles of privacy-aware ubiquitous systems,” in *Ubicomp 2001: Ubiquitous Computing*, pp. 273–291, 2001.

[16] J. Krumm, *Ubiquitous computing fundamentals*. Boca Raton: Chapman & Hall/CRC Press, 2009.

[17] R. Mayrhofer, "Context prediction based on context histories: Expected benefits, issues and current state-of-the-art," *Cognitive Science Research Paper-University of Sussex csrp*, vol. 577, p. 31, 2005.

[18] R. Kusber, K. David, and B. N. Klein, "A novel future internet smart grid application for energy management in offices," in *Future Network & Mobile Summit 2013*, July 2013.

[19] "Gestaltung technisch-sozialer Vernetzung in situativen ubiquitären Systemen (VENUS)." `http://www.uni-kassel.de/eecs/iteg/venus/`, last accessed: 2013-04-11.

[20] C. Voigtmann and K. David, "A Survey To Location-Based Context Prediction," in *Proceedings of the First Workshop on recent advances in behavior prediction and pro-active pervasive computing (AwareCast)*, June 2012.

[21] C. Voigtmann, S. L. Lau, and K. David, "An approach to collaborative context prediction," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, pp. 438–443, march 2011.

[22] C. Voigtmann, S. L. Lau, and K. David, "A collaborative context prediction technique," in *Vehicular Technology Conference (VTC Spring) 73rd*, (Budapest, Hungary), pp. 1–5, IEEE, May 2011.

[23] C. Voigtmann, K. David, J. Zirfas, H. Skistims, and A. Roßnagel, "Prospects for context prediction despite the principle of informational self-determination," in *Advances in Human-Oriented and Personalized Mechanisms, Technologies and Services (CENTRIC)*, (Nice), pp. 98–92, IEEE, Aug. 2010.

[24] C. Voigtmann, K. David, H. Skistims, and A. Roßnagel, "Legal assessment of context prediction techniques," in *Vehicular*

Technology Conference (VTC Fall), (Quebec City, QC), pp. 1–5, IEEE, 2012.

[25] C. Voigtmann, S. L. Lau, and K. David, “Evaluation of a collaborative-based filter technique to proactively detect pedestrians at risk,” in *Vehicular Technology Conference (VTC Fall)*, (Quebec City, QC), pp. 1–5, IEEE, 2012.

[26] C. Voigtmann, C. Schütte, A. Wacker, and K. David, “A new approach for distributed and collaborative context prediction,” in *10th IEEE Workshop on Context Modeling and Reasoning 2013 (CoMoRea 2013)*, (San Diego, USA), pp. 20–24, IEEE, Mar. 2013.

Chapter 2

Context prediction in ubiquitous computing systems

First, this chapter motivates context prediction and its usefulness to ubiquitous computing systems. Second, the basic functionalities of context prediction and its components are presented. Following, the most common state of the art in context prediction is introduced by the visualisation and the discussion of present prediction approaches. In addition, research works with focus on location-based context prediction are outlined in this chapter. The approaches are evaluated with regard to various aspects concerning the data sets used by the authors to evaluate their applied prediction approaches. At the end of this chapter, existing frameworks for context prediction as well as existing considerations of trust and privacy in ubiquitous computing systems are presented.

2.1 Background to context prediction

To motivate the basic idea and the usefulness of predicting a user's next context in a ubiquitous computing system, background information to context prediction is provided in this section. In the beginning, context prediction is introduced as a component of a ubiquitous computing system. Finally, the first presence of context prediction in ubiquitous computing systems is outlined and a short introduction to the functionality of context prediction is provided.

2.1.1 Components of a ubiquitous computing system

In our view, a ubiquitous computing system consists of several components. First, a ubiquitous computing system consists of an environment in which the interactions between users and computers are handled implicitly. Implicitly means that the user provides information, so called contexts, without giving attention to it. Examples for environments are smart spaces like smart homes, meeting rooms or even cars. To receive information of a user located in a ubiquitous environment, sensors are needed. These sensors are placed unobtrusively without distracting the users in the environment. A sensor can be every entity that can be used to collect any kind of information to characterise a user's behaviour. Examples for sensors are phidgets [1] or enocean sensors [2] for smart spaces as well as an on-board computer for a car. Furthermore, a communication infrastructure that can be Wi-Fi-based, cable-based or based on Bluetooth is needed to transmit the collected sensor data to a database structure. Subsequently, processing tasks are needed to interpret the gathered sensor data of the user. Tasks can be the fusion of sensor data [3, 4] to gain information with a lower entropy. The fusion of sensor data can be better than using the gathered sensor information individually. Another task can be the abstraction of gathered sensor data to a higher abstraction level, to shrink the underlying information space [5]. Further tasks that can be applied on the received data are the extraction of a user's current context [6] using appropriate context recognition approaches and the predicting of a user's next context based on previously recognised contexts [7], applying context prediction algorithms. Finally, a ubiquitous computing system consists of services or applications. These services and applications utilise the inferred context information to adapt their behaviour to the contexts of the user, located in the respective ubiquitous computing system.

An example for a ubiquitous computing system, is given by the application scenario outlined in Figure 2.1. The application scenario has been developed and implemented during the Venus project [8] at the University of Kassel. The scenario outlines a ubiquitous computing system used to support elderly people to live independently. The system provides a user, who is in charge of taking care of an elderly person, the opportunity to open an information window from her current position straight into the apartment of the elderly person, using a smart device. Using this information window, the user is able to get a quick overview of whether the elderly person is doing well or if the person needs medical support [9].

The concept to exchange context information between people that already know each other and live apart, is called SoLin (Social Link) [10]. To fulfil the task to support an elderly person to live independently, the flat of this person (the kitchen and the living room) has been equipped with unobtrusive sensor technology. These sensors, whose positions are highlighted in cyan colour in the figure, collect data in order to receive information about the current status of the person and the flat. The data are continuously transmitted and stored to a database using a cable-based connection. Afterwards, the received sensor information are interpreted using algorithms for sensor data fusion and context recognition to infer contexts that characterise the current situation of the person and the flat. Finally, the inferred contexts such as "did the person move sufficiently today", "are the hotplates still switched on although the person sits in front of the TV" can be displayed by the application. Also context prediction algorithms such as the

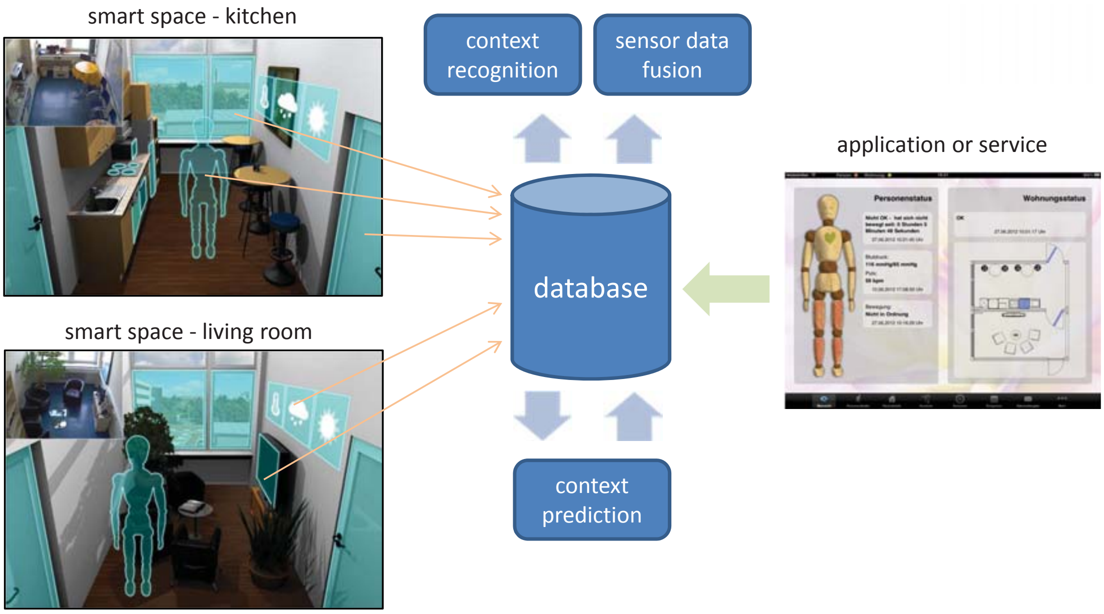

Figure 2.1: Ubiquitous computing system developed during the Venus project [8].

Alignment predictor [11] are integrated to predict the next action of the elderly person, utilising her already inferred context data stored at the database. Related to Support-U, the usage of a context prediction approach offers the possibility to proactively infer an upcoming critical situation that affects the elderly person.

For this reason the application of context predictors provides a time advantage for ubiquitous computing systems, which is a fact that cannot be underestimated.

2.1.2 Context prediction in ubiquitous computing systems

Context awareness is one of the main features of ubiquitous computing systems. Predicting a user's upcoming context even enhances context awareness in a way that applications or services can adapt their behaviour proactively to the user's needs. To the best of our knowledge, the first project that considered context prediction to proactively adapt a ubiquitous environment using previously seen contexts, was the adaptive house project [12] conducted by Micheal C. Mozer in 1998. Figure 2.2 illustrates a picture of the so called neural network house in Boulder, Colorado.

Figure 2.2: The Adaptive House at Boulder, Colorado [13].

In contrast to standard computerised homes, the neural network house used an adaptive control home environment system, called ACHE. This system monitored the environment, collected specific information of the inhabitants' lifestyle (e.g. adjusting the thermostat; turning on a particular configuration of lights; preferred sound levels or the inhabitant motion activity) and attempted to find regular behaviour patterns of the inhabitants. These contexts were used by ACHE to anticipate the inhabitants' needs to save energy costs by proactively adapting light or air temperature or by heating rooms in advance that are likely to be occupied in the near future. The used predictors to predict the next actions of the inhabitants were implemented as feed forward neural networks.

To provide proactiveness using context prediction, previously sensed and stored context data are needed that describe a user's behaviours in the past. The data set that stores the context data of a person is called context history. Different types of abstraction levels of context data can be stored in a user's context history. Overall there exist three different abstraction levels of context data. Raw context data represent uninterpreted and directly gathered sensor values. Low-level context data represent a first abstraction of the gathered raw sensor data. High-level context data represent a higher interpretation level of low-level context data. High-level context data are mostly related to a person or even characterise a person (cf. Definition 1). An example for a transformation of raw data, received form a sensor, to high-level context data is illustrated in Figure 2.3. The measured voltage of the temperature sensor represents the raw sensor data. The low-level context is obtained by the conversion of the voltage into a temperature. The high-level context "warm" can, e.g. be defined by the subjective interpretation of the calculated temperature.

Definition 1 (*High-level context*)

Represents a high-level context element c that describes or characterises a person or a person's action at a certain point in time located in a ubiquitous environment.

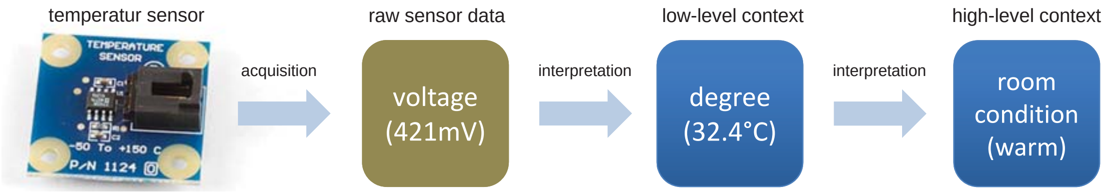

Figure 2.3: Different abstraction levels of contexts, illustrated using a phidget temperature sensor.

In our opinion, low-level and high-level context data are most suitable for the prediction of a user's next context. In contrast, raw data are more useful for context recognition approaches used to automatically infer higher interpretations of contexts that are based on the available raw sensor data. Examples are given in [14, 15].

In contrast to high-level context data, low-level context data offer more precise information [16]. Therefore, low-level contexts derived from different sensors are mostly stored separately. This, e.g. offers the opportunity to utilise existing relations between the contexts of the different sensor sources [17]. Furthermore, the order of gathered low-level contexts can be maintained, which can be important because different context sources most probably gather and interpret sensor data at different points in time [18]. The high-level representation of context data is the representation that is commonly used in literature related to context prediction. Examples can be found in [7, 19, 20, 21, 22]. Different high-level context data derived from various sensors in the same ubiquitous environment are mostly stored in one context history that belongs to a certain user (cf. Definition 2). In this thesis, we focus on high-level context data to predict a user's upcoming context. From now on, the terms context history and high-level context history are considered synonymously. An example of a user's context history with respect to the contexts derived by the ubiquitous computing system that is outlined in Section 2.1.1, is given in Figure 2.4. Basically, a context history of a user that consists of high-level contexts describes the lifestyle or the habits of a user in a ubiquitous environment. In Figure 2.4 the life habits of a person are stored using already interpreted high-level contexts. The high-level contexts are sorted by time.

Definition 2 (*High-level context history*)

Let H be a high-level context history of a user. H is called high-level context history if all contexts stored in this history are sorted by time and if all contexts $c \in H$ stored in this history are high-level contexts.

Contexts stored in a user's context history are mostly represented by a string that characterises the context at a certain point in time. The number of contexts that are stored in a history are only limited by the available storage space. The more contexts have been saved to the history of a user the more information can be utilised by a context prediction approach to predict a user's next context.

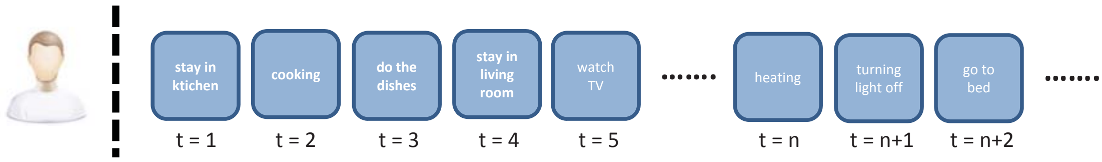

Figure 2.4: Example of a user's context history with respect to the scenario outlined in Section 2.1.1.

Therefore, a context history represents the knowledge base that is used by a context prediction approach to infer a user's next context, based on her current available contexts. To train and to validate a context prediction approach, a context history is divided into a training context data set and a test context data set (cf. Definition 3).

Definition 3 (*Training and test data set*)

A training data set TR and a test data set TS consist of high-level contexts $c \in TR \vee c \in TS$. The training and the test data set form a subset of the context history $TR \subset H \vee TS \subset H$ of the user. Therefore, $TR \cup TS = H$. The training and the test data set can be used to evaluate a context prediction approach with respect to its prediction accuracy, its needed prediction time or its needed memory consumption. The complete context history $TR \cup TS = H$ of a user is used to train a prediction model if a user's next contexts have to be predicted under real conditions in a ubiquitous environment.

For the application and evaluation of context prediction approaches the context history of the user is segmented. The segmentation of the context history is used to create various context parts that characterise the behaviour of a user during different time periods. A context part consists of a context pattern and a future context. The definition of a context pattern and a future context is given in Definition 4. The definition of the process of context prediction using the terms context patterns and future contexts is given in Definition 5.

Definition 4 (***Context pattern and future context***)

Let Cp be the context pattern that is used to identify the future context Fc of a user using her context history H. Cp consists of contexts $c \in H$. Whereby the size of Cp is determined by $2 \leq |Cp| \leq 7$ and the size of $fc \in H$ is determined by $|Fc| = 1$.

The size of the context patterns depend on the number of high-level contexts, the prediction of a user's next context is based on. In other words, the size of a context pattern characterises how much information from the past of a user is taken into account to provide a reliable prediction of a user's next context. In the experiments performed later in this thesis, the minimum size of a context patterns can be 2 and the maximum size of a context patterns can be 7.

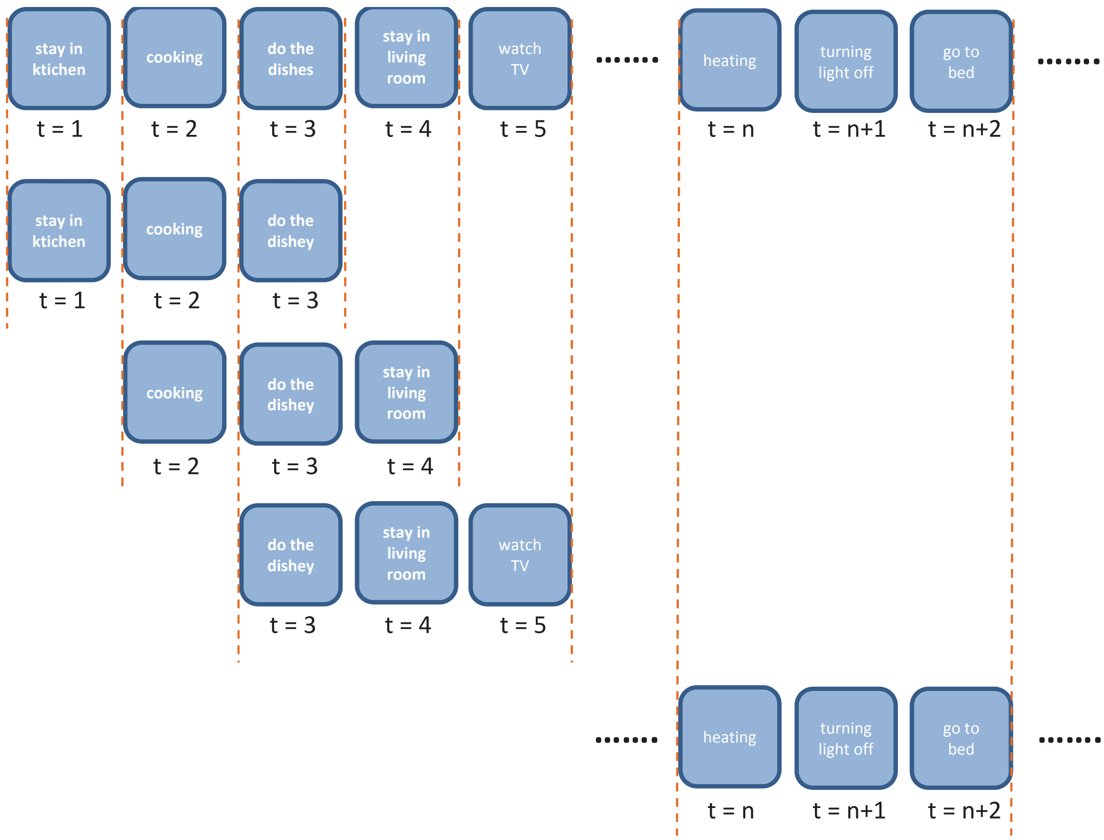

Figure 2.5: Segmentation of the user's context history using sliding window with $\frac{|Cp|-1}{|Cp|}$ overlapping.

The segmentation of a user's context history is performed using the sliding window approach [23]. The sliding window approach has already been used in previous investigations with concern to ubiquitous computing [24, 25, 26]. This algorithm is used in order to ensure that all available context patterns can be utilised in the prediction process. If a context history of a user is segmented without using the sliding window approach, information can get lost because the resulting context parts do not represent all possible context sequences gathered of the user in the past.

Definition 5 (*Context prediction*)

Context prediction is performed by algorithms that are trained using a given knowledge base, e.g. a training data set that has been extracted of a user's context history $TR \subseteq H$ or the context history H itself to generate a prediction function $f(Cp)$. Subsequently, $f(Cp) \rightarrow Fc$ is used to abstract from the given context pattern Cp to predict the future context $Fc \in TR$ or $Fc \in H$ of a user.

An example of a segmentation of a context history applying the sliding window approach, using an overlapping interval of $\frac{|Cp|-1}{|Cp|}$ is illustrated in Figure 2.5. In this figure the context history belonging to the person in the ubiquitous computing system outlined in Section 2.1.1 was split into several context parts. Each result window, respectively each context part consists of three contexts. That implies $|Cp| = 2$ and $|Fc| = 1$. The resulting context patterns are used to be matched either against context patterns of a test data set to evaluate a prediction approach or to be matched against a current context pattern to predict a user's next context in a ubiquitous environment. In both cases the context Fc that follows the context pattern Cp in the training data set or in the context history that most probably matches will be predicted as a user's next context.

2.2 Existing approaches in context prediction

In this subsection the most relevant context prediction techniques are presented. Furthermore, a detailed overview of location-based context prediction is given.

2.2.1 Context prediction techniques

In this thesis the Alignment predictor, the ActiveLeZi predictor and the StatePredictor approach are presented and discussed in more detail. Basically, most supervised learning approaches like, e.g., Markov Models, Support Vector Machines, Bayesian Networks, etc. can be used to predict a user's next context. An overall implementation of usable prediction approaches can, e.g. be found in the Weka framework [27] or in the rapid-i framework [28]. In this thesis the three above-mentioned context prediction approaches are used for evaluation because they are well known and have been developed for context prediction tasks in ubiquitous environments in particular. Further, the prediction accuracy of these approaches with regard to different data sets is compared to the Collaborative-based Context Predictor in Chapter 3.

Alignment Alignment is a context time series prediction algorithm that is inspired by algorithms with focus on computational biology. The Alignment prediction techniques have already been successfully proposed in [16, 18, 11, 29]. The algorithm is based on local Alignment techniques, such as the Smith and Waterman algorithm. Alignment compares two context sequences. Therefore, it belongs to the family of pattern-matching algorithms. The first sequence represents the context history H of the user whose next context has to be predicted. The second sequence represents the current context pattern of the user. During the matching process, a context pattern in H will be identified whose similarity to the given current context pattern is the highest and therefore results in the lowest penalty costs for a given cost matrix. As a result, the context that follows the identified context pattern in the history of the user will be predicted as the next context. For the calculation of the alignment of a given context pattern Cp and a user's context history H, the formula presented in 2.1 has been used. The formula, which has already been outlined in [11], has been adopted to our mathematical alphabet. Using the formula in 2.1, a matrix will be created that holds the penalty costs for the alignment of H and Cp. Finally, the context that is most probable to predict is given out using backtracking.

$$\begin{aligned}
&Cp_{1\ldots i}, H_{1\ldots j} = \\
&max(Cp_{1\ldots i-1}, H_{1\ldots j-1} + \delta(Cp_i, H_j), \\
&Cp_{1\ldots i-1}, H_{1\ldots j} + \delta(Cp_i, -), \\
&Cp_{1\ldots i}, H_{1\ldots j-1} + \delta(-, H_j), \\
&0)
\end{aligned} \tag{2.1}$$

Figure 2.6 outlines the matrix that contains the calculated penalty costs for the given example. The columns represent the contexts of H and the rows represent the contexts of Cp. To provide a better understanding, the context history outlined in Figure 2.4 is utilised. The context pattern is given by the sequence {watch TV, heating, turning light off}.

To calculate the overall penalty cost matrix, which is presented in Figure 2.6, a penalty cost of -1 is given if a context does not match, if a context has to be deleted or if a context has to be inserted. If a context matches, 1 is added. With regard to the matrix presented in Figure 2.6 for a given context history and for a given context pattern, the context *go to bed* is predicted.

ActiveLeZi The ActiveLeZi context predictor presented by Gopalratnam et al. in [30, 31, 32] and improved by Fang et al. in [33] is based on the Jacob Ziv and Abraham Lempel's LZ78 dictionary-based data compression algorithm that incrementally parses a given input sequence. ActiveLeZi further extends LZ78 by exploiting all the information in the context history of a user

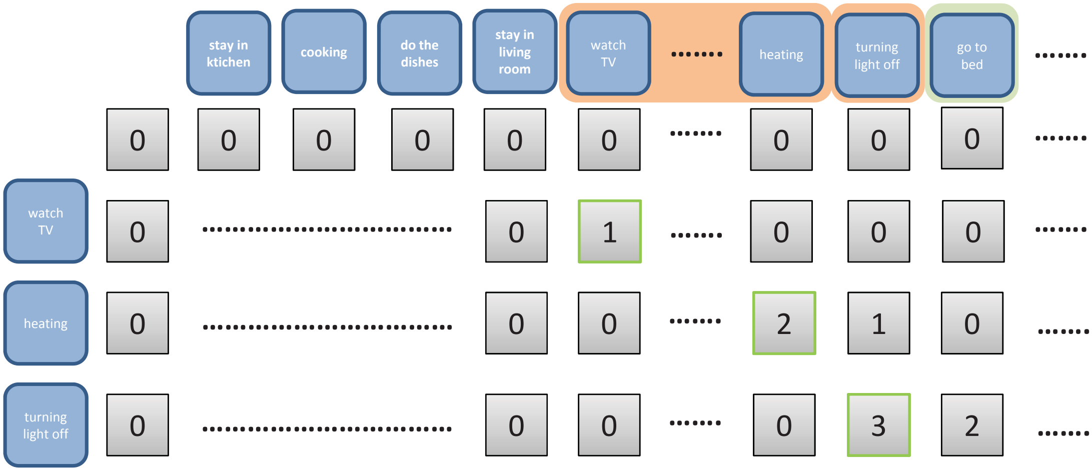

Figure 2.6: Matrix that holds the penalty costs. Less is worse.

using a sliding window approach. While ActiveLeZi parses the given context history of a user it forms a trie and calculates the probabilities for every possible context transition. The maximum depth of the trie corresponds to the length of the longest context pattern in the history of a user that has been found by ActiveLeZi. To predict a user's next context the generated trie receives the current pattern Cp as input and calculates the probability for all possible contexts that might follow after the given context pattern. The context with the highest probability will finally be predicted next.

To provide an illustrative example, a trie created by ActiveLeZi with regard to the context history presented in Figure 2.4 is outlined in Figure 2.7. For better reading, the contexts of the user such as *cooking* or *heating* have been converted into symbols. The context *stay in kitchen* is represented by the symbol "a", the context *cooking* is represented by the symbol "b" and so on. Furthermore, the history of the user has been expanded by duplicating the history three times, not to provide more information but to enlarge the context history. Finally, the history is represented by the following concatenated symbols "abcdefghabcdefghabcdefgh". According to the trie presented in Figure 2.7 the prediction of a user's next

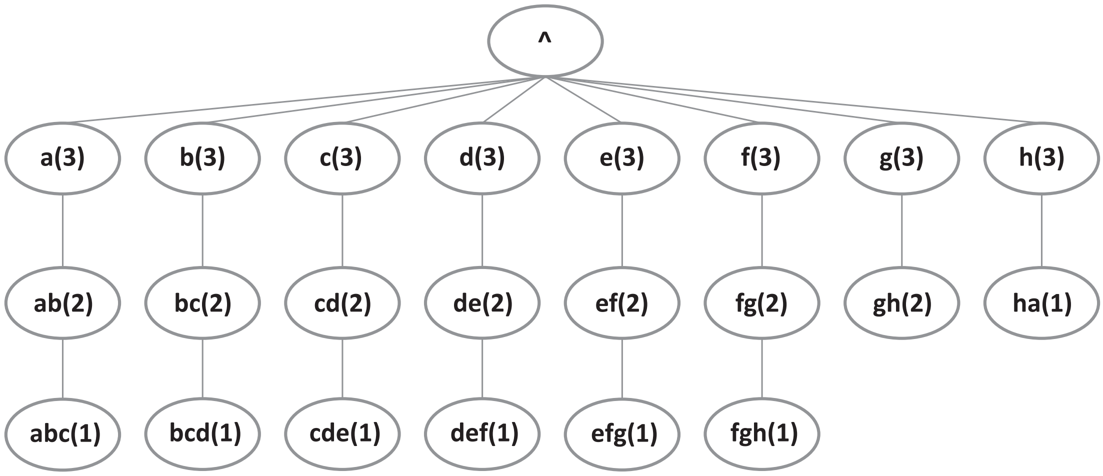

Figure 2.7: Trie formed by the ActiveLeZi approach parsing the string "abcdefghabcdefghabcdefgh", which represents the user's context history.

context can only be based on the last two recognised contexts of a user. This is because the size of the longest context pattern that was found by ActiveLeZi is three. If this trie is used to predict the next context e.g, of a given pattern "ab" that corresponds to the user's actions *stay in kitchen* and *cooking*, ActiveLeZi would predict "c", which corresponds to the context "do the dishes" with a probability of approximately 52%.

StatePredictor Jan Petzold at the University of Augsburg developed the StatePredictor approach. The approach was published in [34, 35, 36, 37, 38]. The StatePredictor is inspired by branch prediction techniques of microprocessors [39]. These techniques were transformed to handle context prediction tasks. Petzold distinguishes between a 1-state and a 2-state context predictor. The 1-state context predictor works the same way a one-bit branch predictor works.

Each possible context that can be predicted is presented by a state. According to each context there exists a 1-state prediction graph. The different states in the graph represent the different contexts that can be predicted after perceiving the contexts respectively the state the prediction graph is associated with. The state that is currently activated in the graph will be predicted. If the predicted state/context is correct, the graph remains in that state; otherwise it changes to the state that should have been predicted. To provide a simple example the history of a user presented in Figure 2.4 is limited to the following five contexts *watch TV, stay in kitchen, cooking, heating, go to bed.* The graph presented in Figure 2.8 outlines the prediction-graph associated with the context *stay in kitchen* of the user's history. The states indicate which future context of the user can be predicted after seeing the context *stay in kitchen.* The 2-state predictor presents a modification of the two-

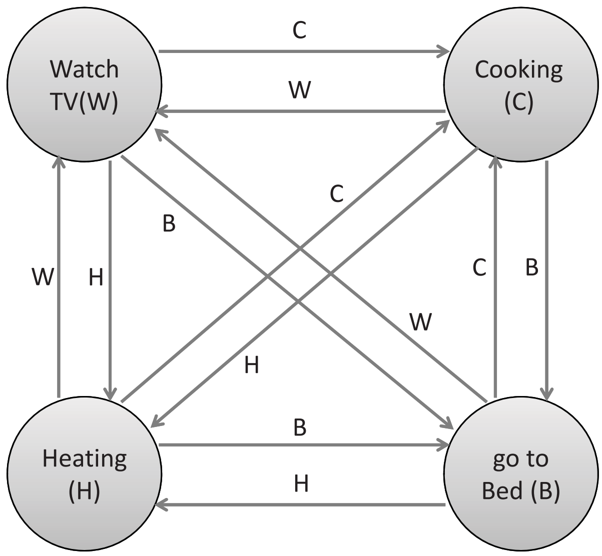

Figure 2.8: Prediction graph a of 1-state predictor or the context *stay in kitchen.*

bit branch predictor. Just like the 1-state predictor there exists one prediction graph for each different state in a user's context history that can be predicted. In contrast to a 1-state prediction graph, the 2-state prediction graph represents all contexts with two states. One weak state and one strong state. If the prediction of a user's next context is correct, the StatePredictor switches into the strong state. If a prediction is incorrect and it is in a strong representation of a state, it switches to the weak representation of the state. If it is already in the weak representation of a state and the prediction turned out to be wrong, it automatically switches to the weak representation of the state it should have been predicted. Figure 2.9 presents the 2-state modification of the prediction graph with regard to the history shown in Figure 2.4.

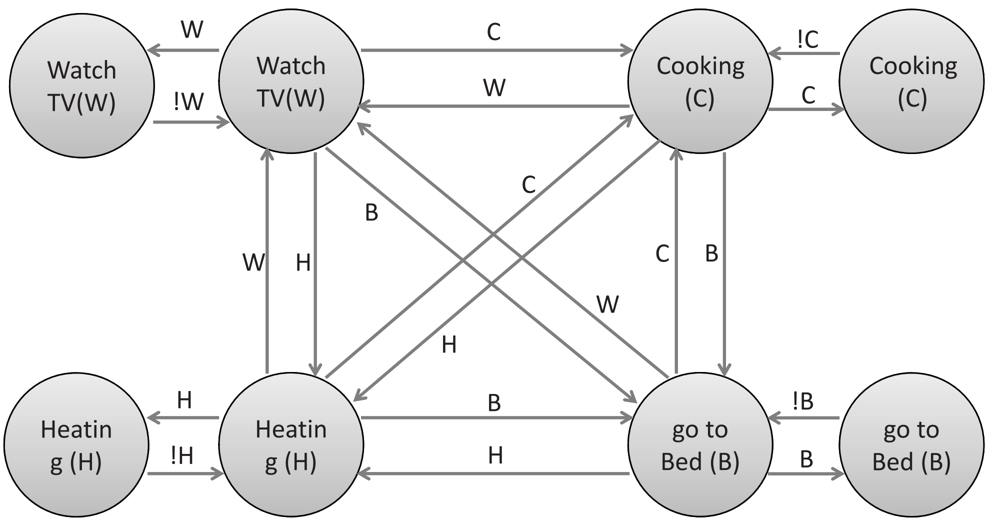

Figure 2.9: Prediction graph a of 2-state predictor for the context *stay in kitchen*.

Discussion To the best of our knowledge the three presented context prediction approaches have been solely used in literature to provide context predictions that are only based on the user's own context history so far. Therefore, the approaches have not been used with additional context histories of other users that show similar behaviours to the user whose next context has to be predicted. In our opinion, there are two possibilities to apply these algorithms to multiple context histories of several users. On the one hand, the different context histories can be concatenated to one big history. This history can then be used as an overall knowledge base for the different context predictors. On the other hand, the algorithms can be applied to the different context histories of the users separately. Subsequently, a majority voting is used to achieve the final prediction result. Both possibilities have the disadvantage that they do not take advantage of existing direct or indirect relations, which may exist among the users' context histories. Following to the idea presented in [40], the Higher-order Singular Value Decomposition is applied to utilise the direct and indirect information that exist between the context histories of the users. This is only possible, if the context histories of the users are put into relation to each other.

All three presented approaches have the substantial disadvantage that they will probably fail to predict a user's next context if her behaviour changes and no information is provided by the user's context history. In Section 3.3 the CCP approach is introduced, which tries to overcome this drawback of existing context prediction approaches that only take the user's own history into consideration during the prediction process.

2.2.2 Location-based context prediction

Over the past few years, a high number of interesting research work with focus on context prediction techniques has been published. These scientific works have covered a wide range of different application fields. The prediction of future vehicular traces [41], the prediction of pedestrians' next paths [42], the development and the investigation of new suitable approaches to predict next context information [30, 35, 11] and the examination and the development of so-called context prediction frameworks [20, 18]. Further, there exist several research works that discusses other aspects that are of concern for context prediction like, e.g. its legal effects [7]. This section describes the existing state of the art in context prediction with focus on the prediction of a user's next location. The section is divided into location-based context prediction using outdoor and indoor locations of a user as contexts. Furthermore, the presented research is evaluated with respect to the aspects presented in Table 2.1. These aspects focus on the data sets used by the authors to evaluate their prediction approaches. Most of the work outlined in this section has been published in [43] by the author of this PhD thesis.

Table 2.1: Describes the different aspects for the evaluation of the used data sets.

abbreviation	**aspect**
own data collected	did the authors collect their own data?
data set is extensive	did the authors specify how extensive their used data set is?
pub. data used	did the authors use publicly available data sets?
sim. data used	did the authors use simulated data?
prob.	did the authors mention problems they faced during the data collection process?
data is published	did the authors publish their data set?
approach compared	did the authors compare the approach to existing approaches?
approach published	did the authors make their approach available for the public?

Indoor Location Prediction

So-called smart homes represent a possible applicability for indoor location prediction. Smart homes are self-contained ubiquitous entities that offer the ideal space for observing and collecting persons' behaviours and environmental features. The Neural Network House project presented in Section 2.1.2 was one of the first smart home projects. The house included a device called ACHE to predict user actions based on their collected context information. The evaluation of the ACHE system showed that the collected behaviour patterns of the inhabitants did not show as much regularities as expected. Unfortunately, the authors did not outline an exact probability of correctness in their contributions.

Similar to the Neural Network House project is the MavHome project conducted by Diana J. Cook et. al [44, 45]. The goal of the project was to collect environmental context information of the inhabitants that lived in the smart home. These collected context data contain the movement behaviours of the inhabitants inside the house. Afterwards, these histories were used to predict the inhabitants' next location to minimize maintaining costs of the home and to maximize the comfort of the inhabitants. The next location predictions were made using the ActiveLeZi context prediction approach presented in Section 2.2. Evaluation results showed a prediction accuracy of 87%. Another approach developed in conjunction with the MavHome project was the Episode Discovery algorithm [46]. The Episode Discovery approach was used to pre-process the context data received of the inhabitants by filtering excessive noise.

In the field of indoor location prediction, a vision of smart doorplates within an office building were introduced in [47]. Smart doorplates were used to notify a visitor about the potential return of an absent office owner. Based on the smart doorplates a collection of movement data of four persons over a period of several months were collected and were published in [37]. The so called "Augsburger Indoor Location Tracking Benchmark" data set is publicly available at the institute for pervasive computing [48] together with additional context data sets. The data has been used in [37], [38] and [49] and to evaluate and compare several data mining techniques like, e.g., Multilayer Perceptrons, Bayesian Network or Markov Models with the StatePredictor approach presented in Section 2.2. The accuracy received by the StatePredictor showed that this prediction approach is a competitive technique compared to well-known data mining approaches. Furthermore, the Augsburger data set was used in [36] to evaluate a context prediction technique that bases on neuronal networks. The prediction results received by the proposed techniques were quite similar to the results presented in [37], [38] and [49].

An approach to infer a user's next position that additionally uses future knowledge derived from contextual sources such as a user's calendar was presented in [50]. The proposed approach extends a $O(k)$ Markov predictor that directly operates on states derived from past user movements by adding knowledge of a user's potential presence at a future location. The potential presence time has been extracted from the user's calendar. The extended Markov model was evaluated in comparison to Markov models that only used the user's movement history using the Dartmouth movement traces[1] data set. The gained results showed that the proposed extended Markov model could outperform classical Markov models by 6% to 30%.

A comparison of the different introduced indoor location prediction approaches with regard to the aspects outlined in Table 2.1 is presented in Table 2.2.

Outdoor Location Prediction

One of the first approaches that used GPS data in order to make reliable outdoor location predictions was presented in [19]. The data have been collected for a period of four months using an external GPS receiver. Afterwards, the authors used a modified k-means approach to cluster the data to meaningful locations. The location history of the user was used to infer the most likely place the user will go next.

Another approach that inferred high-level movement behaviours from tracked GPS data was outlined in [5]. The authors created a data file that contains 12 hours of GPS coordinates collected over a period of three months. This data were used to train a Bayes filter approach combined with an Expectation Maximization approach to learn the parameter of the Bayesian model. The trained model was used to recognise the current transportation mode (driving by bus, driving by car or walking) of a user. Afterwards, the information was used to predict the most likely path the user will go next.

One of the first approaches that collected location-based context data in form of GSM data using a mobile phone was developed in [21] during the Context Project. The main focus of the project was the examination and the understanding of the user's current context and the usage of these context data to provide automatic inferences. Kari Laasonen et al. developed two consecutively arranged approaches for the prediction of user movements within a GSM-Network. The first approach [21] described the automatic recognition of cell transitions, the learning of important locations

[1] http://crawdad.cs.dartmouth.edu

Table 2.2: Evaluation of the different aspects related to the state of the art in indoor location prediction.

ref.	own data collected	data set is extensive	pub. data used	sim. data used	prob.	data is published	approach compared	approach published
[12]	yes	no	no	no	no	no	no	no
[44]	yes	no	no	no	no	no	no	no
[45]	yes	no	no	yes	no	no	yes	no
[30]	yes	yes	no	yes	no	no	no	no
[46]	no	yes	no	yes	N/R	N/R	yes	no
[36]	no	yes	yes	yes	N/R	N/R	no	no
[37]	no	yes	yes	no	N/R	N/R	yes	no
[38]	no	N/R	yes	no	N/R	N/R	yes	no
[49]	no	yes	yes	no	N/R	N/R	yes	no
[50]	no	N/R	yes	yes	N/R	N/R	yes	no

and the prediction of possible important locations the user is going to next. Therefore, the proposed prediction approach took a sequence of recent cell transitions to find the most probable cell the user will enter next. The data have been collected for six months with software that runs continuously on a mobile phone. The second approach outlined in [22] extends the first one. Instead of only predicting the possible next important location (cell) the presented approach tries to predict the whole path that a user will probably go next. The gained prediction accuracy varied between 70% and 90%.

While cell-based location prediction is limited to the arrangement of radio cells of the cellular network and therefore cannot consider the exact geometry and the topology of the user's path, network-based location prediction using GPS can detect the user's position more precisely as outlined in [51]. In this paper two prediction approaches that use synthetic trajectory data sets, containing GPS information to predict a user's next path, were presented. The first approach adopts probabilistic information while the second approach adopts a regression-based classification technique for the trajectory prediction. The results showed that both approaches received better prediction accuracy than random predictions.

Not the prediction of a pedestrian's next movement or location was the objective in the following paper [41], but the prediction of a driver's possible next destination. Therefore, the authors collected GPS waypoints from about 200 drivers for a couple of weeks. Beyond the consideration of previously visited destinations, the proposed Bayesian algorithm, which was performed to run directly on a vehicle's navigation system, also considered trends in the data. The prediction accuracy of the algorithm improved, the closer the driver came to his desired destination. A comparison between different machine learning approaches for outdoor location prediction was presented in [52]. The authors compared a spatial context model with a Bayesian Network, a Decision Tree, a Rule-Induction and an Instance-based classification algorithm. Furthermore, the different approaches were combined using voting, bagging and boosting mechanisms. The best prediction result with regard to accuracy was achieved by applying the voting approach to the spatial context model.

In most cases, existing approaches to outdoor location prediction try to predict only the next behaviour or the next important place of a user. Therefore, they do not try to look further into the future. In [53] an approach called NextPlace is described that uses nonlinear time series not only to predict the next location but also to predict the user's arrival and residence time at the next location. To evaluate the NextPlace approach the authors used four different data sets. Two contain GPS-based data and two contain registration patterns of WiFi access points. The proposed approach extracted the significant locations from the GPS data and the WiFi data. Afterwards, two time series were derived, one that contains all start times and one that contains all duration times related to visited significant locations. Subsequently, these two histories were used to predict a user's next place, her arrival time and her residence time with an overall prediction accuracy up to 90%.

A comparison of the different introduced outdoor location prediction approaches with regard to the aspects outlined in Table 2.1 is presented in Table 2.3.

Interpretation of the aspects

The different indoor and outdoor location prediction approaches presented in this section have been examined with regard to the aspects outlined in Table 2.1. The results are presented in Table 2.2 for the indoor location prediction approaches and in Table 2.3 for the outdoor location prediction approaches.

The most interesting fact that can be noticed is that neither in the indoor prediction area nor in the outdoor prediction area a newly collected data set or a developed prediction approach have been made publicly available. Hence, it is quite difficult for interested researchers to evaluate and reproduce the presented results. Due to this, it is also hardly possible to compare own results with published

Table 2.3: Evaluation of the different aspects related to the state of the art in outdoor location prediction.

ref.	own data collected	data set is extensive	pub. data used	sim. data used	prob.	data is published	approach compared	approach published
[19]	yes	yes	no	no	yes	no	no	no
[5]	yes	yes	no	no	yes	no	yes	no
[21]	yes	yes	no	no	N/R	no	no	no
[22]	yes	yes	no	no	N/R	no	no	no
[51]	no	yes	no	yes	yes	no	no	no
[41]	yes	yes	no	no	N/R	no	no	no
[52]	no	N/R	no	yes	N/R	N/R	yes	no
[53]	no	yes	yes	no	N/R	N/R	yes	no

ones. The only data sets that are publicly available and that have been used in the presented research works are the Augsburger Location Tracking Benchmark data set, the data set created during the Context Project and the data sets used in [53]. An additional and extensive indoor data set containing automated recognised user activities that might be useful for context prediction, is described in [54].

2.3 Frameworks for context prediction

In this section two frameworks for context prediction are presented and compared with each other.

The first framework that enables context prediction was proposed by Mayrhofer in 2004 [20, 55]. The motivation of this work was to provide a basis for the evaluation of different context prediction approaches using the proposed context prediction framework. The approaches can be loaded at runtime due to a plugin-in concept. The proposed framework supports the context prediction task in an online and unsupervised manner and works directly on a Win32 system as well as on WinCE, POSIX and SymbionOS. Overall the architecture consists of four steps as presented in Figure 2.10. The first step combines the data acquisition phase and the feature extraction phase, which is closely coupled with the acquisition of the sensor data. Supported sensors of mobile devices are the window of an active application, the current audio stream, bluetooth, GSM, network connectivity, battery power, WiFi and the video signal from simple built-in CCD cameras.

Subsequently, in the second step, the extracted features of the derived sensor data are classified into a set of classes. Therefore, the "Lifelong Growing Neural Gas classification approach" first introduced in [56] was implemented in the framework. An extended version of the approach was published by Mayrhofer in [57]. The classes found by the classifier will be assigned to descriptive names given manually by the user in the labeling step. In the last step prediction algorithms like e.g. ARMA, MLP, HMM, SVM can be used to enable proactivity by predicting future contexts. A practical demonstration of the proposed context prediction framework was given in [55] and [58]. A data set used for the demonstration of the usefulness of the framework has been collected over a period of two months. The data set consists of real data received from the above-mentioned sensors of a mobile device. The focus was not to evaluate the received prediction results of the prediction approaches but to give a proof of concept that the proposed context prediction architecture is able to provide context prediction in an unobtrusive way and also performs well on devices with a weaker performance such as mobile devices.

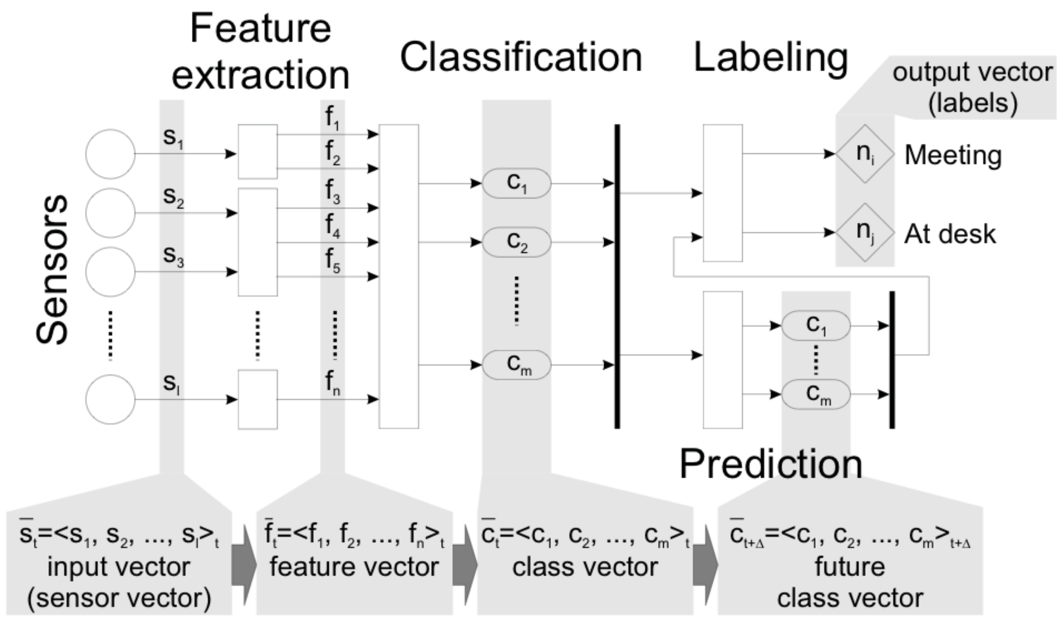

Figure 2.10: Context prediction architecture presented by Mayrhofer in [58].

The second context prediction framework was published by Sigg [16, 18] in 2007. Sigg introduced a modular context prediction architecture that could be applied to low-level contexts as well as to high-level contexts. The proposed context architecture outlined in Figure 2.11 consists of three different layers. The first layer is the acquisition layer, which is able to gather the data of different sensor types. Subsequently, the gathered sensor data are constantly saved to the user's context history. The prediction layer includes a learning component and a prediction component. The learning component extracts behaviour rules that base on the data stored in the context history of the user. Based on the extracted rules and the recently observed contexts, the prediction component predicts the user's context that most likely follows next. The third layer represents the interpretation step. In this step, noise from the predicted contexts is removed and reasoning, based on the predicted contexts, is performed. But the main focus of the author was on context prediction not on context reasoning. The prediction architecture

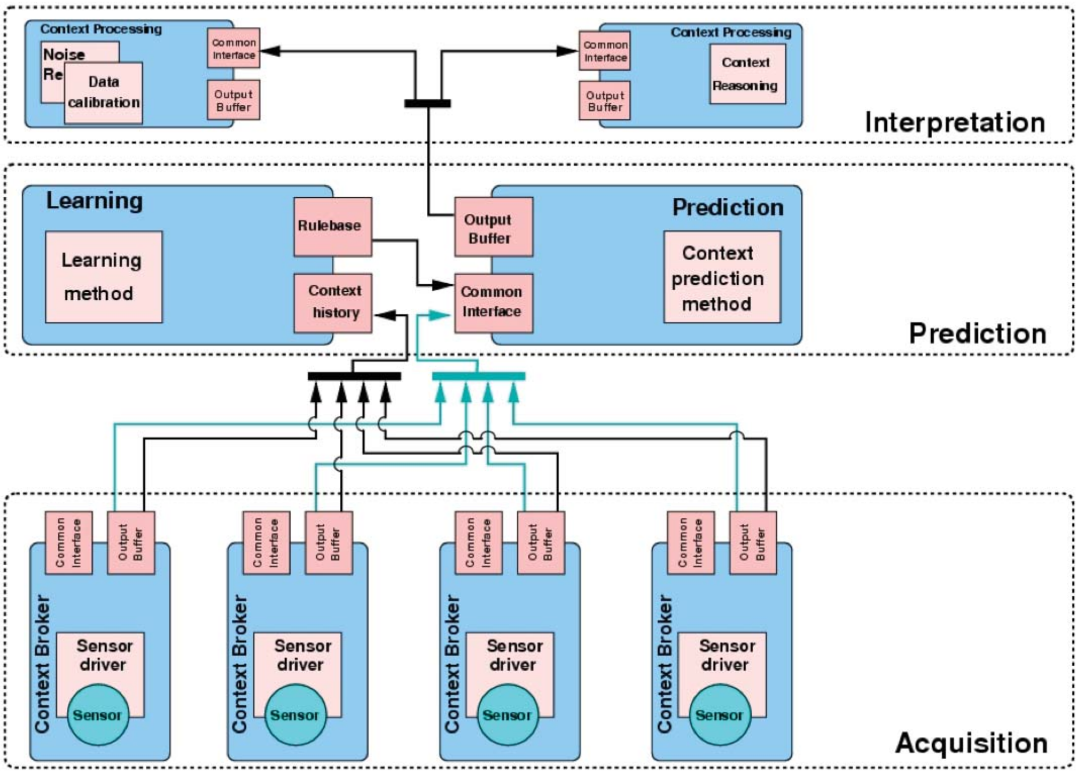

Figure 2.11: Context prediction architecture presented by Sigg in [16].

was used to examine whether high or low-level contexts are more suitable for predicting future contexts. Therefore, several studies have been provided that concern the error probability and the sizes of search spaces with respect to high-level and low-level contexts. The results demonstrate that the usage of low-level context resulted in better prediction accuracies [59, 18]. For the prediction of future contexts Sigg compared several approaches, e.g., ARMA, Markov models and the Alignment predictor using the proposed prediction architecture. The author applied the three approaches to two different context histories. The first history contains data from wind turbines, the second data set contains location based data [16]. The ARMA approach is the dominating prediction method when using low prediction horizons, while for longer prediction horizons the Alignment method becomes more competitive and outperforms the ARMA approach in context prediction studies.

The difference between the two proposed context prediction frameworks is the way they process the sensed context data. The architecture proposed in [20] aggregates low-level context data to high-level context data by using unsupervised methods before the context prediction process. In contrast to the aforementioned approach, the proposed framework in [16] aggregates low-level to high-level context data after the prediction process. Thus, the prediction process showed by Sigg can be less erroneous. First, the search space offers more information because it has not been shrunk by calculating high-level contexts. Second, it is obvious that clustering processes, which are used to derive high-level contexts, can also include errors, which negatively affect the overall prediction process. A further difference of the two prediction frameworks is that the architecture proposed in [20] by Mayrhofer can be executed directly on mobile devices. In contrast the architecture proposed in [16] by Sigg has been optimised for stationary systems like desktop or server computers. The idea of collaborativeness is not supported by any of the two context prediction frameworks. In this thesis a possible approach that addresses the integration of collaborativeness in an architecture for context acquisition and prediction is outlined in Chapter 6.

2.4 Collaboration in associated research fields

The aspect of collaboration is not only limited to the field of context prediction but also finds its application in associated research fields like activity recognition. In this section we focus on two applications of activity recognition where collaboration has already been applied to. These applications are, on the one hand, pedestrian dead reckoning (PDR) to allow, e.g. self-navigation without reference points and, on the other hand, research related to single user (SAR), multi user (MAR) and group activity recognition (GAR).

Kloch et al. considered collaborative PDR systems to improve the correctness of location estimation of users. In [60] the authors used additional location information of a second user to reduce the error rate of the determination of a user's indoor location derived by a PDR systems. The error results due to the double integration needed to recognise a user's location based on her acceleration data. Data have been derived utilising a foot-mounted inertial measurement unit that includes an acceleration sensor, a gyroscope and a compass. In the conducted experiments each time two users were less than 1m apart from each other the location information of both users were combined to correct their individual location estimation. The simulation results outlined by the authors showed the dependence between the collaborative location error and the number of users used to correct the location error. Compared to the location error derived by using raw PDR location recognition, which is 8m in average, the error rate could decrease to almost 2m using location information of 600 people in their simulations.

In a consecutive paper by Kloch et al. a smartphone-based PDR system, which uses collaboration in a real crowded public space to prevent unbounded PDR error, was presented [61]. As presented in the first paper the authors used the proximity information of two users if they are near to each other to improve their location estimation using PDR. The data set containing real movement traces used for evaluation were collected by 12 users who attended to a festival. As ground truth GPS coordinates of the attendees were used. The evaluation showed that the PDR error using raw data could be reduced from up to 100m to 25m using the collaborative estimation approach.

Another collaborative PDR approach using location information of clustered groups to adjust erroneous trajectories was presented in [62, 63]. In contrast to Kloch et al. who used location information of user pairs to correct their location estimation, the authors used group-based error correction whereby a group consists of several users who are in close distance to each other and whose moving traces are similar. In a real world experiment with 20 examinees a relative position accuracy of 3.51m could be demonstrated by correct PDR deviations using traces of other members belonging to the same group.

Compared to the way collaboration is used in this thesis and has been introduced in Section 1 the presented research works that utilise PDR focus on the recognition of present contexts instead of predicting upcoming contexts of a user. Furthermore, the work presented by Kloch et al. utilises information of several users but only combines information of two users at the same time to correct their location estimation errors. In contrast, the idea of using information gained by groups that include an unspecified number of users used to correct their location estimation error presented by Yamaguchi et al. corresponds to the way collaboration is used in this thesis.

In the research field of detecting single and multi user activities Gu et al. proposed an activity recognition system that is capable to determine both SAR and MAR simultaneously [64]. SAR defines the activity recognition of a single user whereas the activity is only inferred from the user's own contexts. MAR defines multiple activity recognitions of multiple users at the same time. The results, which base on contexts gathered from two users who acted in a smart home environment for two weeks, showed an overall recognition accuracy for both SAR and MAR of 89.72%.

Gordon et al. introduced group activity recognition GAR as the recognition of a single activity which is inferred from activities of multiple individuals [65]. Further, the authors present a definition of collaboration, which states that the activity recognition process fundamentally depends on information of multiple subjects and therefore data of multiple users have to be used to infer the activity, which corresponds with the definition of collaboration used in this thesis. To evaluate the proposed system for GAR an experiment using an office scenario was conducted. To detect resulting group activities common data mining techniques (C4.5, k-NN, Naive-Bayes) were applied to the collected data. Using locally extracted features a GAR accuracy of 96% was achieved.

MAR and GAR as outlined above differ in the way they utilise the recognised context from the way the predicted context is used in this thesis. In this thesis context information of the available users are used to provide information to a single user. In contrast to that MAR uses contexts of several users to simultaneously recognise separate activities of multiple users and GAR uses contexts of several users to detect a single context of a whole group.

2.5 Trust and privacy in ubiquitous computing systems

Privacy and trust are without no doubts two important challenges with respect to ubiquitous computing. Mark Weiser pointed out this opinion as he formulated his vision of ubiquitous computing in 1991. Privacy and trust are closely intertwined. It is obvious that if any kind of system respects a user's privacy, e.g. by providing transparency from a legal perspective, a user is most willingly to trust a system. If a system uses, stores and processes personal data without the knowledge of the user, it is probable that a user's trust in the ubiquitous computing system might decrease. Ubiquitous computing, according to the vision of Weiser, provides transparency to the user from the technical perspective. That implies that the aim of ubiquitous computing systems is to use mostly personal context data without explicitly informing the user about this process. The only way a user is able to recognise that her personal data have been utilised is by noticing the autonomous adaptions, reactions or decisions of a ubiquitous computing system according to her behaviours. Techniques in ubiquitous computing that particular rely on these personal context data are, e.g. context recognition or context prediction. Both techniques are used to accomplish autonomous adaptions, reactions or decisions of ubiquitous computing systems. Therefore, the unobtrusive collection of personal context data and the fact that users are not explicitly requested and informed that their data are collected and used are two major drawbacks that cause privacy issues. The third major drawback of ubiquitous computing systems is the fact that contexts are mostly processed on a server, e.g. due to performance aspects. This implies that the user loses control of her data.

In the following, principles and guidelines how trust and privacy could be fulfilled in ubiquitous computing are outlined. These principles have been published in [66] by Langheinrich first:

- **notice**, which means that users should receive a simple notification, if they are monitored or if their personal data are collected. Langheinrich, e.g. proposed some kind of announcement system similar to a radio traffic announcement system, RFID tags that passively announce data collection or the usage of the P3P developed at the World Wide Web Consortium.

- **choice and consent**, means that it is not sufficient to simply notify the user whose data are collected, but also her explicit consent for the data collection process is required. This can, e.g. be achieved by signing a contract. Langheinrich also mentioned the ineffectiveness of confirming a data transfer, e.g. via a user's smartphone because it always requires the attention of the user. Further, the author remarked that even if a single person declines a service it should be able to selectively disable certain functionality without disabling the whole ubiquitous computing system.

- **anonymity and pseudonymity**, means that a user can not be identified by her data because no link exists between a user and her personal context data. This would guarantee 100% security to the user. Langheinreich has already identified that anonymity prevents applications in ubiquitous computing to support the user based on her personal data because of the missing link between the user and her data. The author proposed that the pseudonymisation of the data could be an alternative.

- **proximity and locality**, means that data of other persons can only be collected by a sensor if the sensor owner is in close proximity to the sensor. This could prevent the intentional and unintentional collection of others personal data. Regarding locality, Langheinreich proposed the possibility that context data should be bound to the location the data were collected in.

- **adequate security**, means that the communication channel used to transmit personal context data of a person to the surrounding ubiquitous computing system should be secure. Langheinrich identified a bunch of constraints to security with respect to ubiquitous devices. An example is the power consumption of such devices, which could be quite insufficient to ensure a secure communication.

The presented principles outlined by Langheinrich offer good and visionary guidelines to establish and strengthen trust and privacy in ubiquitous computing in general. Actually, the investigation of direct consequences for techniques that are used in ubiquitous computing such as context prediction is still missing. An investigation how well existing algorithms used for context prediction satisfy existing legal requirements is presented in Chapter 4.

Another paper of Langheinrich [67] discusses the very interesting fact that there are situations, when users feel invaded in their privacy by ubiquitous computing systems. One situation occurs, if personal context information are constantly recorded and stored externally. These data can, e.g. easily be "played back" by third parties. This possibility might influence users in the way they behave. The second situation derives from the possibility to constantly store and access personal data. This enables the creation of personal user profiles. These profiles could result in a "glassy user" whose privacy can barely be protected. Today, there are plenty of services in the World Wide Web that are able to easily create user profiles. Another reason where users feel being invaded in their privacy is the fact that their personal information may cross social borders. This includes, e.g. information shared between a user and her doctor that are transmitted to the user's employer. A further interesting example is the possibility of inferring a user's next action based on her already collected context data. This reason urgently emphasises that context prediction, as one technique in ubiquitous computing system, affects a user's privacy.

A privacy and trust architecture for context aware systems was developed during the Awareness project [68]. The architecture ensures that a person will be informed if her personal data are sensed and transferred to an external party. But more important, a user has to give her explicit consent before personal data can be transmitted. Further, the architecture takes trust relationships into account. That basically means if user X trusts user Y and user Z trusts user X than user Z also automatically trusts user Y. One important aspect of privacy and trust, the pseudonymisation of contexts, to prevent third parties of creating profiles has not been taken into account. The resulting architecture finally offers an infrastructure for context-aware mobile applications while aspects of privacy and trust are considered. The overall aspects are user controlled privacy, context-aware security and context-aware trust management. The architecture was validated through prototyping with mobile health applications.

A conceptual framework of privacy management to enable social awareness in ubiquitous systems was presented by Raento et al. in [69]. The authors applied a descriptive instead of a normative definition of privacy and trust. Therefore, their ideas to trust mainly grounded on social psychology. The authors identified constituents for trust such as control, which is similar to giving consent, accountability, plausible deniability, reciprocity and utility. These constituents have been considered during the implementation of an application called ContextContacts. The application enriches mobile phone contacts with additional information e.g. the current and the last position of the contact, the phone usage activity of the contact, the presence status and the phone profile of the contacts [69].

Control as one identified constituent of trust has been implemented in the ContextContacts application by providing the user with a so called self-view, which shows every context information that has been sent to other users. Further, the user can decide whether she wants to be visible to other users or not. Accountability, the possibility to see the users who had access to a user's contexts, has been implemented to ContextContacts to overcome the feeling of being monitored by other users. The aspect of plausible deniability has not been implemented in the application but shows a really interesting and challenging idea: "how is it possible that a context aware system offers white lies, e.g. the automatic denial of a user's presence?". The aspect of reciprocity has been achieved by only showing the presence status of other contacts in the application, if the contact's own status is visible too, which is an important aspect because it automatically puts all contacts on the same level. The aspect of utility has been mentioned by the authors in theory: "Design of technical systems cannot rely on utility only. Requirements from privacy management should instead be seen the same way legal requirements are seen...".

In total the paper presents some motivating ideas how trust and privacy can be implemented in real-world applications. Further, it sensitises the reader that the consideration of trust constituents is as important as the implementation of technical functionalities.

To the best of our knowledge, the first who basically discussed possible privacy and trust implications regarding the user whose contexts have been used during the context prediction process were Nurmi et al. [7]. They identified the acquisition and the storage of user contexts as one possible step in the context prediction process, which raises privacy and security issues. As reasons they mentioned the scattered collection of data by various sensors in an heterogeneous environment and the possibility to access these collected data by third parties. As a possible solution they indirectly mentioned the framework of Mayrhofer, which is able to overcome these problems because it directly runs on a user's mobile device and it gathers only data from built-in sensors (cf. Section 2.3).

Furthermore, the authors proposed two interesting ideas how the prediction process can be extended. First, they proposed the usage of P2P communication not to provide trust but to overcome possible scalability concerns. Second, they proposed prediction sharing, which enables other users to use predicted information of friends or trustful persons. In our opinion, this indicates an initial idea to collaboration in the field of context prediction. Nevertheless, there still does not exist a solution for a prediction process that uses several histories of different users in a collaborative way using P2P communication to provide more trustfulness for the users. A first possible solution is presented and discussed in this doctoral thesis in Chapter 6.

An extensive overview of privacy and its challenges in ubiquitous computing is given in [70]. Despite understanding privacy in terms of ubiquitous computing, the authors of the book chapter present existing technical solutions for privacy in ubiquitous computing. Existing privacy solutions to common ubiquitous computing challenges are motivated using examples from smart spaces (Confab Toolkit [71], PawS system [72]), RFID (tags as identifier [73]) and location-based services (survey to location privacy [74]). Further, the authors try to provide an introduction for the reader respectively to the application provider of how privacy can be addressed. The conclusion of the authors is that there are no simple answers to this question, because so far, there do not exist algorithms or routines that fix the privacy issue. In this thesis this fact is adressed with respect to context prediction by providing a first evaluation in Section 4.4 of existing context prediction algorithms regarding their legal implications on the user introduced in Section 4.3. To create ubiquitous computing applications that address privacy, the author proposes three steps: understanding the impact of the application on the user, understanding how the users interact with the application and understanding the limits of current security technology used by the application. The proposed steps only reflect general guidelines, which indicate that application development in ubiquitous computing is quite challenging due to its far-reaching implications on the users.

Addressing trust and privacy in ubiquitous computing, applications are important points to enable software to become more socially acceptable. The Venus project [8] at Kassel University has the goal to explore a development method that allows to develop socially acceptable software in ubiquitous computing systems by design. User requirements in terms of usability, trust and legal regulations are considered in an interdisciplinary manner during the technical development process. In the following, publications from the Venus project are presented.

In [9] a ubiquitous computing application called Support-U is evaluated with respect to its social acceptability. Socially acceptable means whether the application development process considers, e.g. a user's right to informational self-determination respectively trust as a determinant of technology usage. Support-U addresses the field of Ambient Assisted Living (AAL) and combines it with the field of ubiquitous computing to enable elderly persons to live autonomously. The evaluation outlined in the paper basically shows that considerations of social aspects during the development process likely improve the acceptability of ubiquitous computing systems.

In [75] interdisciplinary patterns to address challenges in the development of context aware applications in ubiquitous environments with respect to their social acceptability are outlined. Two patterns the so called "TrustParency pattern" and the "Self-determination pattern" were presented in more detail. By utilising the Self-determination pattern it can be ensured that the user can decide whether she wants to provide her personal contexts for, e.g. a context prediction process, which is similar to the principle "choice and consent" mentioned by Langheinrich. If the user declines the functionality she can explicitly prevent the system from storing her personal data used for the prediction process. Therefore, the user may not lose control of her personal data. The application of the TrustParency pattern enables the user to receive information about the sensors installed in the ubiquitous environment that surrounds her. For this reason, the presented pattern primarily supports the transparency. It is not about the transparency from a technical point of view but it is about the transparency from a legal point of view, which enables the user to understand the system that utilises her personal contexts. By enabling transparency the TrustParency pattern encourages a user's trust in using context aware application. Both patterns were identified during the interdisciplinary development process of the Support-U application. Finally, an evaluation that included a user survey in form of a questionnaire was performed to determine whether the used patterns could improve a user's trust and privacy. The evaluation showed that the current version of Support-U, which has been developed by using the aforementioned interdisciplinary patterns, resulted in a stronger feeling of trust and privacy of the users compared to previous versions of Support-U.

A further example how to design socio-technical applications for ubiquitous computing was presented in [76]. First, requirements to design socio-technical ubiquitous computing applications, e.g. risks of data transmissions and the usage of personal data, were defined. Subsequently, a development proposal how socio-technical ubiquitous computing application can be designed was outlined. The proposed development approach consists of an iteration of analysis, conceptual and software design, as well as implementation and evaluation. A detailed description of the method is provided in the paper. To give a proof of concept, the authors applied their development proposal on the smart mobile application Meet-U. The case study outlines that the resulted version of Meet-U considers legal aspects, is more usable and more trustworthy than the first version.

A detailed introduction to a method supporting trust for socio-technical ubiquitous applications is given in [77]. So-called trust-supporting components were derived to encourage a user's trust in ubiquitous computing applications. To examine the effectiveness of the trust-supporting components, the authors integrated the components to an application for the recommendation of restaurants. Finally, the application, which has been developed being based on the derived trust components, was evaluated with the help of 166 test persons. The evaluation results show that a user's trust and a user's willingness to use the application could be increased significantly using trust-supporting components.

2.6 Conclusions

In this chapter the background to context prediction was introduced first. Subsequently, the Alignment predictor, the StatePredictor and the ActiveLeZi context predictor were presented, as well as an overview to existing location-based context prediction approaches was outlined. The presented approaches were evaluated with respect to different criteria. Furthermore, existing frameworks for context prediction and existing research works concerning trust and privacy in ubiquitous computing systems were presented and shortly discussed.

The wide diversity of research work focussing on trust and privacy with respect to ubiquitous computing applications shows the significance and the importance of this topic. This is even more the case if personal data of several users are combined and used in a collaborative manner. Therefore, in this thesis the state of the art context prediction approaches, as well as the collaborative-based context prediction approach that is presented in the following chapter, will be evaluated with regard to their compatibility to the right to informational self-determination. Later in this thesis, a possible solution to receive a prediction process that can be considered to be legally acceptable will be discussed.

References

[1] “Phidgets inc. - unique and easy to use USB interfaces.” `http://www.phidgets.com/`, last accessed: 2013-04-11.

[2] “Enocean, energy harvesting wireless sensor solutions and networks.” `http://www.enocean.com/en/home/`, last accessed: 2013-04-11.

[3] H. Wu, *Sensor Fusion for Context-Aware Computing Using Dempster-Shafer Theory*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, December 2003.

[4] H. Medjahed, D. Istrate, J. Boudy, J.-L. Baldinger, and B. Dorizzi, “A pervasive multi-sensor data fusion for smart home healthcare monitoring,” in *Fuzzy Systems (FUZZ), 2011 IEEE International Conference on*, pp. 1466–1473, 2011.

[5] D. J. Patterson, L. Liao, D. Fox, and H. Kautz, “Inferring high-level behavior from low-level sensors,” in *UbiComp 2003: Ubiquitous Computing*, pp. 73–89, 2003.

[6] E. Kim, S. Helal, and D. Cook, "Human activity recognition and pattern discovery," *Pervasive Computing, IEEE*, vol. 9, no. 1, pp. 48–53, 2010.

[7] P. Nurmi, M. Martin, and J. A. Flanagan, "Enabling proactiveness through context prediction," in *Proceedings of the Workshop on Context Awareness for Proactive Systems, Helsinki*, vol. 53, 2005.

[8] "Gestaltung technisch-sozialer Vernetzung in situativen ubiquitären Systemen (VENUS)." `http://www.uni-kassel.de/eecs/iteg/venus/`, last accessed: 2013-04-11.

[9] S. Hoberg, L. Schmidt, A. Hoffmann, M. Söllner, J. M. Leimeister, C. Voigtmann, K. David, J. Zirfas, and A. Roßnagel, "Socially acceptable design of a ubiquitous system for monitoring elderly family members," in *Sozio-technisches Systemdesign im Zeitalter des Ubiquitous Computing (SUBICO 2012)*, pp. 349–363, 2012.

[10] K. David, S. Lau, and B. Klein, "Social link APP," *IEEE Vehicular Technology Magazine*, vol. 6, pp. 80–87, Sept. 2011.

[11] S. Sigg, S. Haseloff, and K. David, "An alignment approach for context prediction tasks in UbiComp environments," *Pervasive Computing, IEEE*, vol. 9, pp. 90–97, Dec. 2010.

[12] M. C. Mozer, "The neural network house: An environment hat adapts to its inhabitants," in *Proc. AAAI Spring Symp. Intelligent Environments*, pp. 110–114, 1998.

[13] M. C. Mozer, "The adaptive house." `http://www.cs.colorado.edu/~mozer/index.php?dir=/Research/Projects/Adaptive%20house/`, last accessed: 2013-04-12.

[14] M. L. Blum, "Real-time context recognition," Master's thesis, Eidgenössische Technische Hochschule Zürich, 2005.

[15] K. Ouchi and M. Doi, “A real-time living activity recognition system using off-the-shelf sensors on a mobile phone,” in *Proceedings of the 7th international and interdisciplinary conference on Modeling and using context*, CONTEXT’11, (Berlin, Heidelberg), pp. 226–232, Springer-Verlag, 2011.

[16] S. Sigg, S. Haseloff, and K. David, “Prediction of context time series,” in *5th Workshop on Applications of Wireless Communications*, pp. 31–45, 2007.

[17] I. König, C. Voigtmann, B. Klein, and K. David, “Enhancing alignment based context prediction by using multiple context sources: Experiment and analysis,” in *Modeling and Using Context* (M. Beigl, H. Christiansen, T. Roth-Berghofer, A. Kofod-Petersen, K. Coventry, and H. Schmidtke, eds.), vol. 6967 of *Lecture Notes in Computer Science*, pp. 159–172, Springer Berlin Heidelberg, 2011.

[18] S. Sigg, *Development of a novel context prediction algorithm and analysis of context prediction schemes*. PhD thesis, University of Kassel, 2008.

[19] D. Ashbrook and T. Starner, “Using GPS to learn significant locations and predict movement across multiple users,” *Personal and Ubiquitous Computing*, vol. 7, pp. 275–286, Oct. 2003.

[20] R. Mayrhofer, “An architecture for context prediction,” in *Advances in Pervasive Computing* (A. Ferscha, H. Hörtner, and G. Kotsis, eds.), vol. 176, pp. 65–72, Austrian Computer Society (OCG), April 2004. part of the Second International Conference on Pervasive Computing (PERVASIVE 2004).

[21] K. L. Mika, M. Raento, and H. Toivonen, “Adaptive on-device location recognition,” in *In Proceedings of the Second International Conference on Pervasive Computing*, pp. 287–304, Springer Verlag, 2004.

[22] K. Laasonen, "Clustering and prediction of mobile user routes from cellular data," in *in PKDD. 2005*, pp. 569–576, Springer Verlag, 2005.

[23] E. Keogh, S. Chu, D. Hart, and M. Pazzani, "An online algorithm for segmenting time series," in *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pp. 289–296, 2001.

[24] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," in *Pervasive Computing*, pp. 1–17, Springer, 2004.

[25] S. L. Lau and K. David, "Movement recognition using the accelerometer in smartphones," in *Future Network and Mobile Summit, 2010*, pp. 1–9, 2010.

[26] S. L. Lau, I. Konig, K. David, B. Parandian, C. Carius-Dussel, and M. Schultz, "Supporting patient monitoring using activity recognition with a smartphone," in *Wireless Communication Systems (ISWCS), 2010 7th International Symposium on*, pp. 810–814, 2010.

[27] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *SIGKDD Explor. Newsl.*, vol. 11, pp. 10–18, Nov. 2009.

[28] I. Mierswa, M. Wurst, R. Klinkenberg, M. Scholz, and T. Euler, "Yale: Rapid prototyping for complex data mining tasks," in *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (L. Ungar, M. Craven, D. Gunopulos, and T. Eliassi-Rad, eds.), (New York, NY, USA), pp. 935–940, ACM, August 2006.

[29] S. Sigg, D. Gordon, G. v. Zengen, M. Beigl, S. Haseloff, and K. David, "Investigation of context prediction accuracy for different context abstraction levels," *IEEE Transactions on Mobile Computing*, vol. 11, pp. 1047–1059, June 2012.

[30] K. Gopalratnam and D. J., “Active lezi: An incremental parsing algorithm for sequential prediction,” in *In Sixteenth International Florida Artificial Intelligence Research Society Conference*, pp. 38–42, 2003.

[31] D. Cook, M. Youngblood, I. Heierman, E.O., K. Gopalratnam, S. Rao, A. Litvin, and F. Khawaja, “MavHome: an agent-based smart home,” in *Pervasive Computing and Communications, 2003. (PerCom 2003). Proceedings of the First IEEE International Conference on*, pp. 521–524, Mar. 2003.

[32] K. Gopalratnam and D. J. Cook, “Online sequential prediction via incremental parsing: The active LeZi algorithm,” *IEEE Intelligent Systems*, pp. 52–58, 2007.

[33] H. Fang and J. Ruan, “An improved position prediction algorithm based on active LeZi in smart home,” in *Computer Science & Service System (CSSS)*, pp. 1733–1736, IEEE, Aug. 2012.

[34] J. Petzold, F. Bagci, W. Trumler, and T. Ungerer, “Global and local state context prediction,” in *In Artificial Intelligence in Mobile Systems 2003 (AIMS 2003), Seattle, WA, USA*, 2003.

[35] J. Petzold, F. Bagci, W. Trumler, and T. Ungerer, “The state predictor method for context prediction,” in *In Adjunct Proceedings Fifth International Conference on Ubiquitous Computing, Seattle, WA, USA*, 2003.

[36] L. Vintan, A. Gellert, J. Petzold, and T. Ungerer, “Person movement prediction using neural networks,” in *In First Workshop on Modeling and Retrieval of Context*, 2004.

[37] J. Petzold, F. Bagci, W. Trumler, and T. Ungerer, “Next location prediction within a smart office building,” in *In Proceedings of 1st International Workshop on Exploiting Context Histories in Smart Environments at the 3rd International Conference on Pervasive Computing*, 2005.

[38] J. Petzold, A. Pietzowski, F. Bagci, W. Trumler, and T. Ungerer, “Prediction of indoor movements using bayesian networks,” in *In Proceedings of Location- and Context-Awareness (LoCA 2005)*, 2005.

[39] J. Silc, B. Robic, and T. Ungerer, *Processor architecture - from dataflow to superscalar and beyond.* Springer, 1999.

[40] P. Symeonidis, A. Nanopoulos, and Y. Manolopoulos, “Tag recommendations based on tensor dimensionality reduction,” in *Proceedings of the 2008 ACM conference on Recommender systems*, pp. 43–50, 2008.

[41] J. Krumm and E. Horvitz, “Predestination: Where do you want to go today?,” *Computer*, vol. 40, pp. 105–107, april 2007.

[42] T. Yoon and J.-H. Lee, “Goal and path prediction based on user’s moving path data,” in *Proceedings of the 2nd international conference on Ubiquitous information management and communication*, ICUIMC ’08, pp. 475–480, 2008.

[43] C. Voigtmann and K. David, “A Survey To Location-Based Context Prediction,” in *Proceedings of the First Workshop on recent advances in behavior prediction and pro-active pervasive computing (AwareCast)*, June 2012.

[44] S. Das, D. Cook, A. Battacharya, I. Heierman, E.O., and T.-Y. Lin, “The role of prediction algorithms in the mavhome smart home architecture,” *Wireless Communications, IEEE*, vol. 9, pp. 77–84, dec. 2002.

[45] D. Cook, M. Youngblood, I. Heierman, E.O., K. Gopalratnam, S. Rao, A. Litvin, and F. Khawaja, “Mavhome: an agent-based smart home,” in *Pervasive Computing and Communications, 2003. (PerCom 2003). Proceedings of the First IEEE International Conference on*, pp. 521–524, march 2003.

[46] I. Heierman, E.O. and D. Cook, “Improving home automation by discovering regularly occurring device usage patterns,” in

Data Mining, 2003. ICDM 2003. Third IEEE International Conference on, pp. 537–540, nov. 2003.

[47] W. Trumler, F. Bagci, J. Petzold, and T. Ungerer, "Smart doorplate," *Personal Ubiquitous Comput.*, vol. 7, pp. 221–226, July 2003.

[48] "Institut für pervasive computing." `http://www.pervasive.jku.at/Research/Context_Database/`, last accessed: 2013-04-17.

[49] J. Petzold, F. Bagci, W. Trumler, and T. Ungerer, "Comparison of different methods for next location prediction," in *Proceedings of the 12th international conference on Parallel Processing*, Euro-Par'06, pp. 909–918, Springer-Verlag, 2006.

[50] M. H. Sun and D. M. Blough, "Mobility prediction using future knowledge," in *Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*, MSWiM '07, pp. 235–239, 2007.

[51] X. Liu and H. A. Karimi, "Location awareness through trajectory prediction," *Computers, Environment and Urban Systems*, vol. 30, no. 6, pp. 741–756, 2006.

[52] T. Anagnostopoulos, C. Anagnostopoulos, S. Hadjiefthymiades, M. Kyriakakos, and A. Kalousis, "Predicting the location of mobile users: a machine learning approach," in *Proceedings of the 2009 international conference on Pervasive services*, ICPS '09, pp. 65–72, 2009.

[53] S. Scellato, M. Musolesi, C. Mascolo, V. Latora, and A. T. Campbell, "Nextplace: A spatio-temporal prediction framework for pervasive systems," in *Pervasive*, pp. 152–169, 2011.

[54] P. Lukowicz, G. Pirkl, D. Bannach, F. Wagner, A. Calatroni, K. Förster, T. Holleczek, M. Rossi, D. Roggen, G. Tröster, J. Doppler, C. Holzmann, A. Riener, A. Ferscha, and R. Chavarriaga, "Recording a complex, multi modal

activity data set for context recognition," in *ARCS Workshops* (M. Beigl and F. J. Cazorla-Almeida, eds.), pp. 161–166, VDE Verlag, 2010.

[55] R. Mayrhofer, *An Architecture for Context Prediction.* PhD thesis, Johannes Kepler University of Linz, Austria, October 2004.

[56] F. H. Hamker, "Life-long learning cell structures – continuously learning without catastrophic interference," *Neural Networks*, vol. 14, pp. 551–573, May 2001.

[57] R. Mayrhofer, "Extending the growing neural gas classifier for context recognition," in *Proc. EUROCAST 2007: 11th International Conference on Computer Aided Systems Theory*, vol. 4739 of *LNCS*, pp. 920–927, Springer-Verlag, February 2007.

[58] R. Mayrhofer, "Eine Architektur zur Kontextvorhersage," in *Ausgezeichnete Informatikdissertationen 2004*, vol. D-5 of *Series of the German Informatics society (GI)*, pp. 125–134, Lecture Notes in Informatics (LNI), May 2005.

[59] S. Sigg, S. Haseloff, and K. David, "The impact of the context interpretation error on the context prediction accuracy," in *Mobile and Ubiquitous Systems - Workshops, 2006. 3rd Annual International Conference on*, pp. 1–4, 2006.

[60] K. Kloch, G. Pirkl, P. Lukowicz, and C. Fischer, "Emergent behaviour in collaborative indoor localisation: An example of self-organisation in ubiquitous sensing systems," in *Proceedings of the 24th International Conference on Architecture of Computing Systems*, ARCS'11, (Berlin, Heidelberg), pp. 207–218, Springer-Verlag, 2011.

[61] K. Kloch, P. Lukowicz, and C. Fischer, "Collaborative pdr localisation with mobile phones," in *Wearable Computers (ISWC), 2011 15th Annual International Symposium on*, pp. 37–40, 2011.

[62] T. Higuchi, H. Yamaguchi, and T. Higashino, "Clearing a crowd: Context-supported neighbor positioning for people-centric navigation," in *Pervasive Computing* (J. Kay, P. Lukowicz, H. Tokuda, P. Olivier, and A. Krüeger, eds.), vol. 7319 of *Lecture Notes in Computer Science*, pp. 325–342, Springer Berlin Heidelberg, 2012.

[63] H. Yamaguchiy, T. Higuchi, and T. Higashino, "Collaborative indoor localization of mobile nodes," in *The Sixth International Conference on Mobile Computing and Ubiquitous Networking*, 2012.

[64] T. Gu, Z. Wu, L. Wang, X. Tao, and J. Lu, "Mining emerging patterns for recognizing activities of multiple users in pervasive computing," in *Mobile and Ubiquitous Systems: Networking Services, MobiQuitous, 2009. MobiQuitous '09. 6th Annual International*, pp. 1–10, 2009.

[65] D. Gordon, J.-H. Hanne, M. Berchtold, A. Shirehjini, and M. Beigl, "Towards collaborative group activity recognition using mobile devices," *Mobile Networks and Applications*, vol. 18, no. 3, pp. 326–340, 2013.

[66] M. Langheinrich, "Privacy by design - principles of privacy-aware ubiquitous systems," in *Proceedings of the 3rd international conference on Ubiquitous Computing*, UbiComp'01, (London, UK, UK), pp. 273–291, Springer-Verlag, 2001.

[67] M. Langheinrich, "Privacy invasions in ubiquitous computing," in *Workshop on Socially-Informed Design of Privacy-Enhancing Solutions (UBICOMP 2002)*, Springer, sep 2002.

[68] M. Wegdam, J. v. Bemmel, M. Bargh, R. Neisse, K. Sheikh, S. Iacob, B. v. Beijnum, and B. Hulsebosch, "An architecture for privacy and trust in context awareness infrastructures," Dec. 2006.

[69] M. Raento and A. Oulasvirta, "Privacy management for social awareness applications," in *Proceedings of 1st Workshop*

on Context Awareness for Proactive Systems (CAPS 2005), HIIT Publications 2005-1, pp. 105–114, Helsinki Institute for Information Technology, 2005.

[70] J. Krumm, *Ubiquitous computing fundamentals.* Boca Raton: Chapman & Hall/CRC Press, 2009.

[71] J. I. Hong and J. A. Landay, "An architecture for privacy-sensitive ubiquitous computing," in *Proceedings of the 2nd international conference on Mobile systems, applications, and services*, MobiSys '04, (New York, NY, USA), pp. 177–189, ACM, 2004.

[72] M. Langheinrich, "A privacy awareness system for ubiquitous computing environments," in *Proceedings of the 4th international conference on Ubiquitous Computing*, UbiComp '02, (London, UK, UK), pp. 237–245, Springer-Verlag, 2002.

[73] S. A. Weis, S. E. Sarma, R. L. Rivest, and D. W. Engels, "Security and privacy aspects of low-cost radio frequency identification systems," in *Security in Pervasive Computing*, vol. 2802 of *Lecture Notes in Computer Science*, pp. 201–212, Springer-Verlag, 2004.

[74] J. Krumm, "A survey of computational location privacy," *Personal Ubiquitous Comput.*, vol. 13, pp. 391–399, Aug. 2009.

[75] C. Voigtmann, D. Wilhelm, M. Söllner, K. David, and J. M. Leimeister, "Interdisziplinary development patterns from the use case: Support-u," in *Sozio-technisches Systemdesign im Zeitalter des Ubiquitous Computing (SUBICO 2013)*, 2013.

[76] D. E. Comes, C. Evers, K. Geihs, A. Hoffmann, R. Kniewel, J. M. Leimeister, S. Niemczyk, A. Roßnagel, L. Schmidt, T. Schulz, M. Söllner, and A. Witsch, "Designing socio-technical applications for ubiquitous computing: results from a multidisciplinary case study," in *Proceedings of the 12th IFIP WG 6.1 international conference on Distributed Applications*

and Interoperable Systems, DAIS'12, (Berlin, Heidelberg), pp. 194–201, Springer-Verlag, 2012.

[77] M. Söllner, A. Hoffmann, H. Hoffmann, and J. M. Leimeister, "Vertrauensunterstützung für sozio-technische ubiquitäre systeme," *Zeitschrift für Betriebswirtschaft*, vol. 2012, no. 4, pp. 109–140, 2012.

Chapter 3

Collaborative-based Context Prediction

Context prediction is used to proactively adapt, e.g. services to users' needs. Due to the fact that context prediction enables proactiveness, it has a high significance for ubiquitous computing systems. To the best of our knowledge, research literature on context prediction only focuses on the history of the user whose next context has to be predicted. In case a user suddenly changes her behaviour in an unexpected way, the context history of the user does not contain appropriate context information to provide reliable context predictions. Hence, context prediction algorithms that only rely on the user's context history whose context has to be predicted, will fail to predict the appropriate future context. To overcome the gap of missing context information in the user's context history, the Collaborative Context Prediction (CCP) approach is proposed. CCP takes advantage of existing direct and indirect relations, which may exist among the context histories of various users. Thereby, CCP bases on the

Higher-order Singular Value Decomposition, which has already been successfully applied in existing recommendation systems. To provide an evaluation of CCP the approach is compared to state of the art context prediction approaches with respect to prediction accuracy. For the evaluation collaborative data sets are used. These data sets consist of context histories of different users which are located in the same ubiquitous environment.

3.1 Motivation

One interesting research issue in the field of context-aware systems and environments is context prediction. Based on the available context data, such systems predict future contexts of a user. With the help of these predicted contexts, users in ubiquitous environments can be assisted to a greater extend in different ways. Taking for example, a research assistant who presents the progress of her work in the same room every week. Before she enters the room for her next presentation, the context-aware system automatically adapts the designated services in the room to be ready for her presentation using a prediction system spanning the whole university. Possible context information useful for the prediction are her movement patterns or devices she has already interacted with in the past. A common approach to enable the prediction of future context is to make use of the gathered and stored contexts related to the user's actions or to the user's environment. The information is needed by a context prediction algorithm to predict contexts for a given context pattern. Thereby, a context pattern is a sequence of contexts. However, if the research assistant gives her presentation in a room she has not been before, her present movement patterns might be unknown to the context prediction system. It can therefore be considered that, context prediction approaches that only rely on the context history of a single user might fail to predict the user's next context. Therefore, the proactive adaptations of the services would not take place [1]. For this reason the information space is expanded by also considering users that are located in nearby surroundings or show sufficient similarity to the user whose context pattern is currently unknown. Based on the upcoming collaborative relations among these users, the term ubiquitous environment is extended to the term *Collaborative* Ubiquitous Environment. As already successfully demonstrated in the work of recommender systems, existing user profiles in collaborative environments like, e.g. Last.fm[1] or Flickr[2] can be used to support other users in these environments. A recommendation of interesting items to buy, using for example similar user profiles is just one possibility. If this aspect is transferred to the field of context prediction, it can be can assumed that users, which are located in the same *Collaborative* Ubiquitous Environment as the user whose current context pattern is unknown, may have similar interests. For this reason their context histories might show similarities, too. The similarities in the context histories of the users are used to bypass the currently unknown context pattern of the individual user. A Collaborative-based Context Prediction (CCP) approach is proposed that increases the possibility to make a currently unknown context pattern of a user available to predict the next future context. The CCP approach utilises the Higher-order Singular Value Decomposition (HOSVD) technique, which is introduced in Section 3.3. To find similarities HOSVD is applied to the context histories of the user whose present context is unknown and to the context history of at least one additional user who is in the same Collaborative Ubiquitous Environment. The Higher-order Singular Value Decomposition method has already been successfully applied in the research field of tag recommendation [2]. CCP utilises HOSVD to enrich the context histories of the users with additional latent information. Latent information comprises new context parts in the context history of the user that were formally unknown and that can additionally be used to infer future contexts. A combination of a context pattern and a future context is

[1] http://www.last.fm, last accessed: 2013-04-20
[2] http://www.flickr.com/, last accessed: 2013-04-20

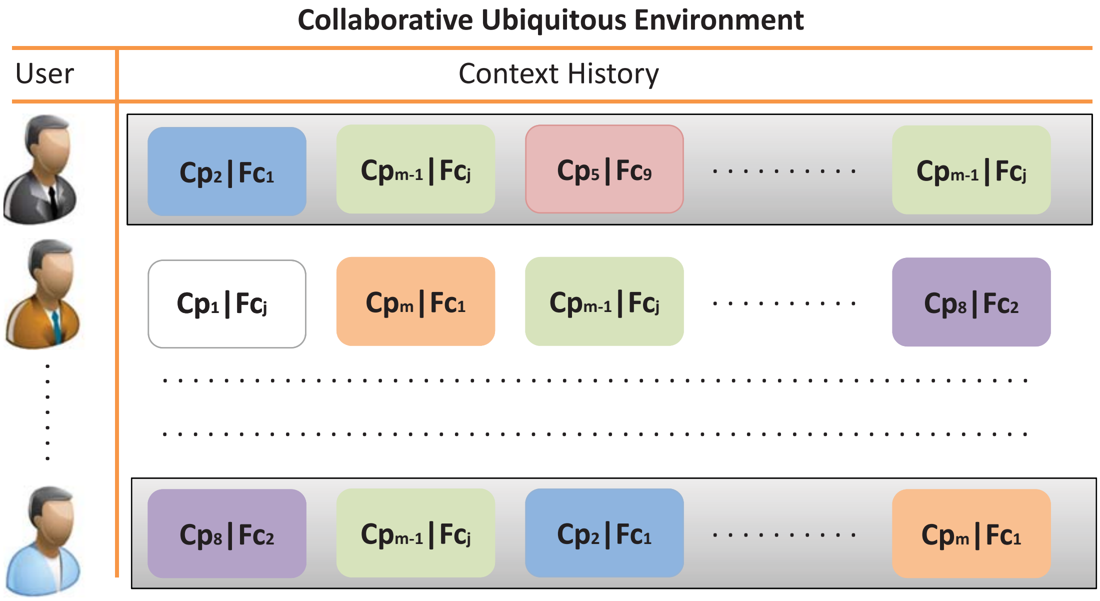

Figure 3.1: Presents n users of a Collaborative Ubiquitous Environment with n different context histories. Equal context parts are marked in the same colour. Every context part in the context history H_i of the user U_i consists of two elements. Cp $\in \mathcal{CP}$ indicates the context pattern and Fc $\in \mathcal{FC}$ indicates the future context that follows the previous context pattern.

called a context part. HOSVD uses existing relations (equal context parts) between the context histories of the users to find possible latent information. The Collaborative Ubiquitous Environment presented in Figure 3.1 that forms the foundation for the CCP approach consists of three different entities. The first entity is represented by the users U $\in \mathcal{U}$ of the *Collaborative* Ubiquitous Environment, the second entity by the set of possible context patterns Cp $\in \mathcal{CP}$ and the third entity by the set of predictable future contexts Fc $\in \mathcal{FC}$. Equal context parts in the collaborative ubiquitous environment are marked in the same colour. Further, a context history $H_i \in \mathcal{H}$ of a user U_i is described by $H_i \subseteq \mathcal{CP} \times \mathcal{FC}$. Altogether the Collaborative Ubiquitous Environment consists of n different users, m different context patterns and j different future contexts.

3.2 Mathematical derivation of HOSVD for collaborative-based context prediction

In this section the mathematical derivation of the Higher-order Singular Value Decomposition (HOSVD), which forms the basis for the Collaborative-based Context Prediction (CCP) approach is outlined. Therefore a short introduction to the Singular Value Decomposition (SVD) and to the tensor data structure is given. The presented mathematical derivations outlined and used in this section are based on the descriptions given in the master thesis [3] of the author of this PhD thesis.

The notation of the matrices and the tensors that are used to store the data for the prediction process which have been derived from a ubiquitous environment is determined by Kiers [4]. Hence, matrices are presented in capital and bold letters, e.g. $\mathbf{A}$. Vectors are presented in lower case letters, e.g. $\mathbf{a}$ and elements of a matrix are represented by a_{ij}. Tensors are illustrated by bold and capital letters with an underscore $\underline{\mathbf{T}} \in \Re^{I_1 \times I_2 \times \ldots \times I_n}$. An element of a third order tensor $\underline{\mathbf{T}} \in \Re^{|I_1| \times |I_2| \times |I_3|}$ is for example determined by $t_{i_1 i_2 i_3}$.

3.2.1 Singular Value Decomposition

By applying the Singular Value Decomposition, square matrices and non-square matrices can be decomposed in a product of three matrices. The decomposition of a given matrix $\mathbf{A}$ in the product of the matrices $\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}^T$ is given in Definition 6.

According to the unitary characteristic of the matrices $\mathbf{U}_{I_1 \times I_1}$ and $\mathbf{V}^T_{I_2 \times I_2}$ the column vectors of both matrices are orthogonal to each other. Furthermore, the length of the column vectors of both matrices is standardised. Therefore, the column vectors of both matrices are also orthonormal to each other. Based on the orthogonal characteristic of both matrices the simplified calculation outlined in equation 3.1 and 3.2 can be used, whereby $\mathbf{E}$ symbolises the unit matrix. HOSVD takes advantage of the simplified calculation of the inverse matrix to decompose a given higher-order tensor structure outlined in Section 3.2.3.

Definition 6 (*SVD*)

The Singular Value Decomposition is used to decompose a given matrix $\mathbf{A} \in \Re^{I_1 \times I_2}$ *into a product consisting of three matrices. The resulting decomposition of* $\mathbf{A}$ *is* $\mathbf{A} \in \Re^{I_1 \times I_2} = \mathbf{U}_{I_1 \times I_1} \mathbf{\Sigma}_{I_1 \times I_2} \mathbf{V}^T_{I_2 \times I_2}$.

- $\mathbf{U}_{I_1 \times I_1}$ *is an unitary matrix.*
- $\mathbf{\Sigma}_{I_1 \times I_2}$ *is a diagonal matrix whose values can not be negative.*
- $\mathbf{V}^T_{I_2 \times I_2}$ *is the transposed matrix of the unitary matrix* $\mathbf{V}_{I_2 \times I_2}$.

$$\mathbf{U}^T \mathbf{U} = \mathbf{U} \mathbf{U}^T = \mathbf{E} \tag{3.1}$$

$$\mathbf{V}^T \mathbf{V} = \mathbf{V} \mathbf{V}^T = \mathbf{E} \tag{3.2}$$

Both matrices $\mathbf{U}$ and $\mathbf{V}$ result from the calculation of the eigenvectors (cf. Definition 7) of $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$. The singular values stored in $\mathbf{\Sigma}$ represent the square root of the calculated eigenvalues (cf. Definition 8) of $\mathbf{U}$ and $\mathbf{V}$. $\mathbf{U}$ and $\mathbf{V}$ do have the same eigenvalues. The eigenvalues are stored on the diagonal of the matrix $\mathbf{\Sigma}$ according to their significance in a descending order, $\forall \sigma \in \Sigma \,|\, \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq 0$. The column vectors of $\mathbf{U}$ are referred to as left singular vectors and the column vectors of $\mathbf{V}^T$ are referred to as right singular vectors.

In general, the SVD can be used to weight data that has been stored in a matrix $\mathbf{A}$ according to their generality and their significance. Thereby, the column vectors of the matrices $\mathbf{U}$ and $\mathbf{V}$ encode the weighted data of the matrix in their column vectors. The first column vector $\mathbf{v}_1$ holds the most relevant data. The singular value σ_1 with the highest number accordingly represents the square root of the eigenvalue λ_1 which satisfies $\mathbf{A}\mathbf{v}_1 = \lambda_1 \mathbf{v}_1$.

Definition 7 (*Eigenvector*)

An eigenvector s a non-zero vector that is mapped by a given linear transformation of a vector space onto a vector that is the product of a scalar multiplied by the original vector.

Definition 8 (*Eigenvalue*)

Given a matrix $\mathbf{A}$ *and a corresponding eigenvector* $\mathbf{v}$ *the scalar* λ *is called eigenvalue of* $\mathbf{A}$ *if it satisfies* $\mathbf{Av} = \lambda\mathbf{v}$.

As a result that the SVD decomposes a given matrix into a product of three matrices where the data of the given matrix is stored in $\mathbf{U}$ and $\mathbf{V}^T$ according to their relevance, SVD can be used as a possible option for data reduction. This is possible by calculating an approximation $\mathbf{A}'$ of $\mathbf{A}$ by $\mathbf{A}' = \mathbf{U\Sigma V}^T$ by only considering the eigenvectors of $\mathbf{U}$ and $\mathbf{V}^T$ whose singular values are higher than a given threshold ξ. If $\mathbf{A}'$ is calculated by a reduced number of column vectors of $\mathbf{U}$ and $\mathbf{V}^T$ the rank of the matrix $\mathbf{A}'$ is also reduced to the number of singular values that exceeds the given threshold ξ (cf. Definition 9). Despite $\mathbf{A}'$ has been calculated only considering the most relevant data of $\mathbf{U}$ and $\mathbf{V}^T$ the dimensions of $\mathbf{A}'$ are still the same size as $\mathbf{A}$. Hence, it is possible to filter noise in the matrix $\mathbf{A}$ which can be caused by irrelevant data. A well known approach that utilises SVD to filter relevant data, e.g. to provide spam detection is Latent Semantic Indexing [5].

Definition 9 (*Rank of a matrix*)

The rank r *of a given matrix* $\mathbf{A}$ *is determined by its number of non zero singular values.*

Picture 3.2 visualises the decomposition of the matrix $\mathbf{A}$ with its dimensionality $I_1 \times I_2$ in its product $\mathbf{U\Sigma V}^T$ and the approximation of $\mathbf{A}'$ by reducing its rank.

3.2.2 Discussion of the tensor data structure

A tensor represents a multidimensional array and is defined as follows:

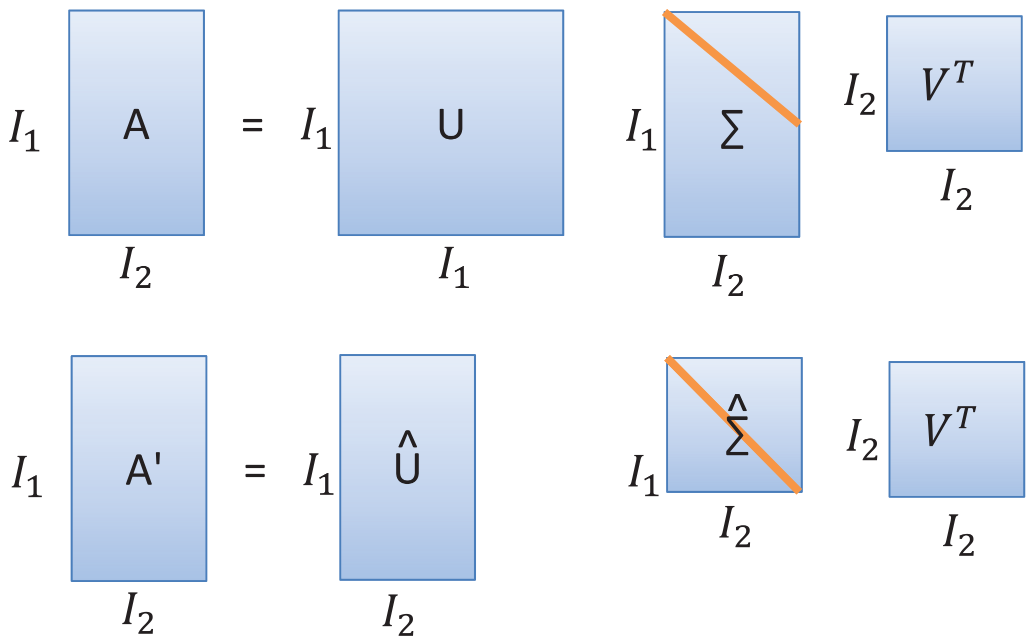

Figure 3.2: Decomposition of matrix **A** using SVD.

Definition 10 (*Tensor*)

A tensor with an order of n is an element of a tensor product that consists of n vector spaces. Each vector space has its own coordinate system [6].

In contrast to a matrix, a tensor can be used to represent existing relations with a higher order than 2. However, for further consideration, third-order tensor structures are used to store and represent the relations of the three relevant entities of a ubiquitous environment: the users $U \in \mathcal{U}$ their context patterns $Cp \in \mathcal{CP}$ and their possible next contexts $Fc \in \mathcal{FC}$ as outlined in more detail in Section 3.3. Picture 3.3 presents a third-order tensor $\underline{\mathbf{T}} \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|}$. The represented tensor consists of three dimensions and can be pictures as a cuboid.

The presented tensor $\underline{\mathbf{T}} \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|}$ consists of $|\mathcal{FC}|$ matrices $\mathbf{A}^{(1)}_{|\mathcal{U}|\times|\mathcal{CP}|} \ldots \mathbf{A}^{(|\mathcal{FC}|)}_{|\mathcal{U}|\times|\mathcal{CP}|}$ that hold the relations between the users $\mathcal{U}$ and the different context patterns $\mathcal{CP}$ of a ubiquitous environment.

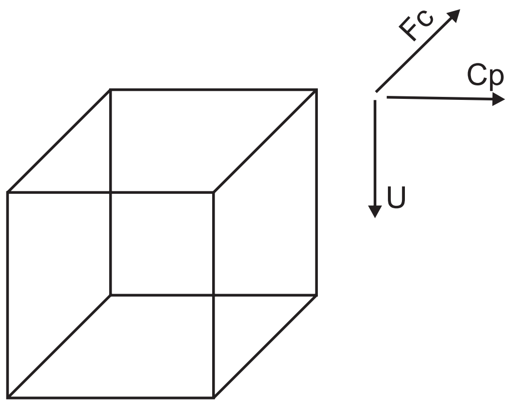

Figure 3.3: Third-order tensor $\underline{\mathbf{T}} \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|}$.

A tensor data structure can be decomposed in fibers and slices. Fibers can be determined as different vector representations. A third-order tensor can be decomposed in three different fiber representations as outlined in Figure 3.4. Column fibers can be determined by $\mathbf{t}_{:\,Cp\,Fc}$, row fibers can be determined by $\mathbf{t}_{U\,:\,Fc}$ and tube fibers can be determined by $\mathbf{t}_{U\,Cp\,:}$.

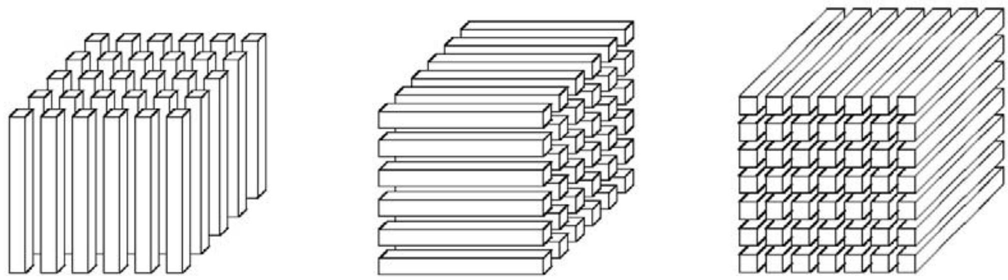

Figure 3.4: Different fiber representations of a third-order tensor [6].

Further, a given third-order tensor can be decomposed in so called slices. In contrast to slices, which represent matrices where two dimensions of the third-order tensor are fixed, fibers represent vectors and therefore one dimension has to be fixed. Overall there exist three different slice representations as outlined in Figure 3.5. Frontal slices are given by $\mathbf{T}_{::\,Fc}$, lateral slices by $\mathbf{T}_{:\,Cp\,:}$ and horizontal slices by $\mathbf{T}_{U\,::}$.

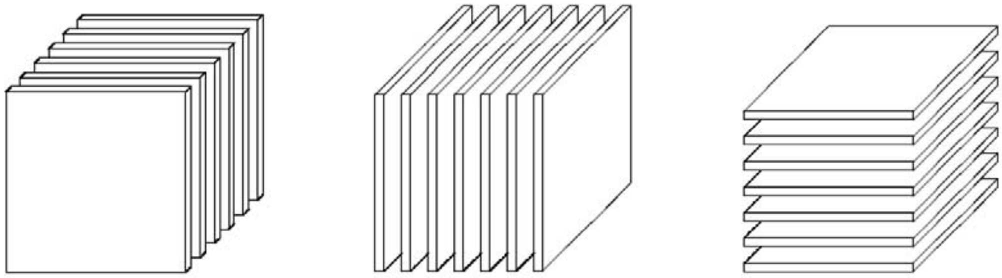

Figure 3.5: Different slice representations of a third-order tensor [6].

The presented decompositions of a tensor are relevant because the SVD can only by applied to matrices. As presented, a third-order tensor can be decomposed in three different matrix representations also called modes. Therefore, a tensor $\underline{\mathbf{T}} \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|}$ can be transformed into the following three modes $\mathbf{T}_U$ $\mathbf{T}_{Cp}$ and $\mathbf{T}_{Fc}$ which are presented in the equations 3.3, 3.4 and 3.5:

$$\underline{\mathbf{T}} \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|} \longrightarrow \mathbf{T}_U \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|*|\mathcal{FC}|} \tag{3.3}$$

$$\underline{\mathbf{T}} \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|} \longrightarrow \mathbf{T}_{Cp} \in \Re^{|\mathcal{CP}|\times|\mathcal{U}|*|\mathcal{FC}|} \tag{3.4}$$

$$\underline{\mathbf{T}} \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|} \longrightarrow \mathbf{T}_{Fc} \in \Re^{|\mathcal{FC}|\times|\mathcal{U}|*|\mathcal{CP}|} \tag{3.5}$$

A further prerequisite to apply the HOSVD presented in Section 3.2.3 to reduce data stored in a tensor structure is the n-Mode product outlined in [7]. The n-Mode product is used to multiply an n-order tensor structure with a given matrix. The n-Mode product is defined as follows:

Definition 11 (*n-Mode Product*)

Is $\underline{\mathbf{A}}$ a tensor with a dimensionality $I_1 \times I_2 \times \ldots I_n \times \ldots I_N$ and $\mathbf{U}$ a matrix with a dimensionality of $J \times I_n$, the n-Mode Product of $\underline{\mathbf{A}}$ and $\mathbf{U}$ is defined by $\underline{\mathbf{A}} \times_n \mathbf{U}$.

The result of the n-Mode product is given by tensor $\underline{\mathbf{A}}'$. It has a dimensionality of $I_1 \times I_2 \times \ldots J \times \ldots I_N$. Using the n-Mode product the matrix $\mathbf{U}$ has been multiplied with the n^{th} mode $\mathbf{A}_n^{I_n \times I_1 * \ldots I_{n-1} * I_{n+1} * \ldots I_N}$ of the tensor $\underline{\mathbf{A}}$. The usage of the n-mode

product illustrates that it can be used to reduce or to enlarge the size of the n^{th} mode of the given tensor $\underline{\mathbf{A}}$. In order to be more precise the the size of n^{th} dimension of the tensor $\underline{\mathbf{A}}$ has been changed to the size of the first dimension of the matrix $\mathbf{U}$.
In the following the n-Mode product is applied to the 3-order tensor $\underline{\mathbf{T}} \in \Re^{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|}$. Thereby, the first mode of $\underline{\mathbf{T}}$ is multiplied with a given matrix $\mathbf{A}_1 \in \Re^{I_1 \times |\mathcal{U}|}$, the second mode with a given matrix $\mathbf{A}_2 \in \Re^{I_2 \times |\mathcal{CP}|}$ and the third mode is multiplied with a given matrix $\mathbf{A}_3 \in \Re^{I_3 \times |\mathcal{FC}|}$. The resulting 3-Mode product looks as follows:

$$\begin{aligned} &\underline{\mathbf{T}} \in \Re^{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|} \times_1 \mathbf{A}_1 \in \Re^{I_1 \times |\mathcal{U}|} \\ &\times_2 \mathbf{A}_2 \in \Re^{I_2 \times |\mathcal{CP}|} \times_3 \mathbf{A}_3 \in \Re^{I_3 \times |\mathcal{FC}|} \end{aligned} \tag{3.6}$$

For the calculation of the 3-mode product the first mode of $\underline{\mathbf{T}}$ is multiplied with the matrix $\mathbf{A}_1$. Subsequently, the second mode of the resulting tensor $\underline{\mathbf{T}}'$ is multiplied with the matrix $\mathbf{A}_2$. Finally, the third mode of $\underline{\mathbf{T}}''$ is multiplied with $\mathbf{A}_3$. 3.7 and 3.8, 3.9, 3.10 presents the calculation of the 3-mode product in detail.

$$\begin{aligned} &\Bigg(\Bigg(\Bigg(\underline{\mathbf{T}} \in \Re^{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|} \times_1 \mathbf{A}_1 \in \Re^{I_1 \times |\mathcal{U}|}\Bigg) \\ &\times_2 \mathbf{A}_2 \in \Re^{I_2 \times |\mathcal{CP}|}\Bigg) \times_3 \mathbf{A}_3 \in \Re^{I_3 \times |\mathcal{FC}|}\Bigg) \end{aligned} \tag{3.7}$$

$$\underline{\mathbf{T}}' \in \Re^{I_1 \times |\mathcal{CP}| \times |\mathcal{FC}|} = \mathbf{A}^{(1)}_{I_1 \times |\mathcal{U}|} * \underline{\mathbf{T}} \in \Re^{|\mathcal{U}| \times |\mathcal{CP}| * |\mathcal{FC}|} \tag{3.8}$$

$$\underline{\mathbf{T}}'' \in \Re^{I_1 \times I_2 \times |\mathcal{FC}|} = \mathbf{A}^{(2)}_{I_2 \times |\mathcal{CP}|} * \underline{\mathbf{T}} \in \Re^{|\mathcal{CP}| \times I_1 * |\mathcal{FC}|} \tag{3.9}$$

$$\underline{\mathbf{T}}''' \in \Re^{I_1 \times I_2 \times I_3} = \mathbf{A}^{(3)}_{I_3 \times |\mathcal{FC}|} * \underline{\mathbf{T}} \in \Re^{|\mathcal{FC}| \times I_1 * I_2} \tag{3.10}$$

$\underline{\mathbf{T}}''' \in \Re^{I_1 \times I_2 \times I_3}$ in 3.10 represents the result of the 3-mode product. With respect to $\underline{\mathbf{T}} \in \Re^{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|}$ the size of the dimensions of the resulting tensor changed accordingly to the size of the first dimension of $\mathbf{A}^{(1)}$, $\mathbf{A}^{(2)}$ and $\mathbf{A}^{(3)}$. The possibility to adapt the sizes of the dimensions of a given tensor applying the n-Mode product is utilised by the Higher-order Singular Value Decomposition, which is shortly introduced next.

3.2.3 Higher-order Singular Value Decomposition

As basis for the Higher-order Singular Value Decomposition (HOSVD) the SVD and the tensor decomposition have been already discussed. Both techniques are utilised by the HOSVD to approximate data in a given tensor structure. The HOSVD has been first published in [7]. HOSVD represents a generalisation of the proposed decomposition method for three-way arrays [8]. In contrast to the proposed method presented by Tucker the HOSVD approach can also be applied to n-order tensors. Essential components of HOSVD are the SVD, the tensor decomposition and the n-Mode product. Assumed a tensor $\underline{\mathbf{T}} \in \Re^{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|}$ is given, it is shown that the approximation of $\underline{\mathbf{T}}$ calculated with the HOSVD (cf. Equation 3.11) is defined by the n-Mode product of $\underline{\mathbf{T}}$ with the transposed of the left-singular matrices calculated from its n different modes using the SVD.

$$\underline{\mathbf{\Sigma}} = \underline{\mathbf{T}} \times_{n=1}^{N} \mathbf{A}^{(n)} \tag{3.11}$$

It is assumed that a matrix $\mathbf{T} \in \Re^{I_1 \times I_2}$ is represented as a tensor $\underline{\mathbf{T}} \in \Re^{I_1 \times I_2 \times 1}$ with order 2. Further, it is assumed that $\underline{\mathbf{T}}$ is decomposed in its two modes $\mathbf{M}^{(1)}$ and $\mathbf{M}^{(2)}$ as shown in equation 3.12 and 3.13.

$$\mathbf{M}^{(1)} = \underline{\mathbf{T}} \in \Re^{I_1 \times I_2 * 1} \tag{3.12}$$

$$\mathbf{M}^{(2)} = \underline{\mathbf{T}} \in \Re^{I_2 \times I_1 * 1} \tag{3.13}$$

Subsequently, the two modes $\mathbf{M}^{(1)}$ and $\mathbf{M}^{(2)}$ will be decomposed in a product of the three matrices $\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}^T$ using the SVD as outlined in Section 3.2.1. The decompositions are outlined in 3.14 and 3.15.

$$\mathrm{SVD}\left(\mathbf{M}^{(1)}_{I_1 \times I_2}\right) = \mathbf{U}^{(1)}_{I_1 \times I_1} \mathbf{\Sigma}^{(1)}_{I_1 \times I_2} \mathbf{V}^{(1)T}_{I_2 \times I_2} \tag{3.14}$$

$$\mathrm{SVD}\left(\mathbf{M}^{(2)}_{I_2 \times I_1}\right) = \mathbf{U}^{(2)}_{I_2 \times I_2} \mathbf{\Sigma}^{(2)}_{I_2 \times I_1} \mathbf{V}^{(2)T}_{I_1 \times I_1} \tag{3.15}$$

If $\boldsymbol{\Sigma}^{(1)} \in \Re^{I_1 \times I_2}$ is represented as a tensor $\underline{\boldsymbol{\Sigma}}^{(1)} \in \Re^{I_1 \times I_2 \times 1}$ and if $\underline{\boldsymbol{\Sigma}}^{(1)}$ is further inserted with the two left-singular matrices $\mathbf{U}^{(1)}_{I_1 \times I_1}$ and $\mathbf{U}^{(2)}_{I_2 \times I_2}$ in the n-Mode product presented in definition 11 the following equation can be outlined:

$$\underline{\mathbf{T}}_{I_1 \times I_2 \times 1} = \underline{\boldsymbol{\Sigma}}^{(1)}_{I_1 \times I_2 \times 1} \times_1 \mathbf{U}^{(1)}_{I_1 \times I_1} \times_2 \mathbf{U}^{(2)}_{I_2 \times I_2} \tag{3.16}$$

Due to the orthogonality of $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ the equation in 3.16 can be transposed into the equation outlined in 3.17.

$$\underline{\boldsymbol{\Sigma}}^{(1)}_{I_1 \times I_2 \times 1} = \underline{\mathbf{T}}_{I_1 \times I_2 \times 1} \times_1 \mathbf{U}^{(1)\mathrm{T}}_{I_1 \times I_1} \times_2 \mathbf{U}^{(2)\mathrm{T}}_{I_2 \times I_2} \tag{3.17}$$

Equation 3.17 shows that the approximation $\underline{\boldsymbol{\Sigma}}$ of a given tensor $\underline{\mathbf{T}}$ can be calculated by using the n-Mode product whereby $\underline{\mathbf{T}}$ is multiplied with the transposed left-singular matrices calculated from its different modes. According to the 3-order tensor structure used to store the data of a ubiquitous environment, the following equation results:

$$\begin{aligned}\underline{\mathbf{T}}_{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|} = \underline{\boldsymbol{\Sigma}}_{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|} \times_1 \mathbf{U}^{(1)}_{|\mathcal{U}| \times |\mathcal{CP}| * |\mathcal{FC}|} \times_2 \\ \mathbf{U}^{(2)}_{|\mathcal{CP}| \times |\mathcal{U}| * |\mathcal{FC}|} \times_3 \mathbf{U}^{(3)}_{|\mathcal{FC}| \times |\mathcal{U}| * |\mathcal{CP}|}\end{aligned} \tag{3.18}$$

To calculate the approximation of the tensor $\underline{\mathbf{T}}_{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|}$, the equation presented in 3.18 is finally transposed into the equation outlined in 3.19.

$$\begin{aligned}\underline{\boldsymbol{\Sigma}}_{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|} = \underline{\mathbf{T}}_{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|} \times_1 \mathbf{U}^{(1)\mathrm{T}}_{|\mathcal{U}| \times |\mathcal{CP}| * |\mathcal{FC}|} \times_2 \\ \mathbf{U}^{(2)\mathrm{T}}_{|\mathcal{CP}| \times |\mathcal{U}| * |\mathcal{FC}|} \times_3 \mathbf{U}^{(3)\mathrm{T}}_{|\mathcal{FC}| \times |\mathcal{U}| * |\mathcal{CP}|}\end{aligned} \tag{3.19}$$

A visualisation of the HOSVD is given in Figure 3.6. The picture represents a visualisation of the equation outlined in 3.18. The cuboid T represents the tensor $\underline{\mathbf{T}}_{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|}$ which is reduced to the tensor $\underline{\boldsymbol{\Sigma}}_{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|}$ by only considering the column vectors of the left-singulars matrices $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, $\mathbf{U}^{(3)}$ whose corresponding singular values exceed a given threshold ξ, which is symbolised by dashed lines.

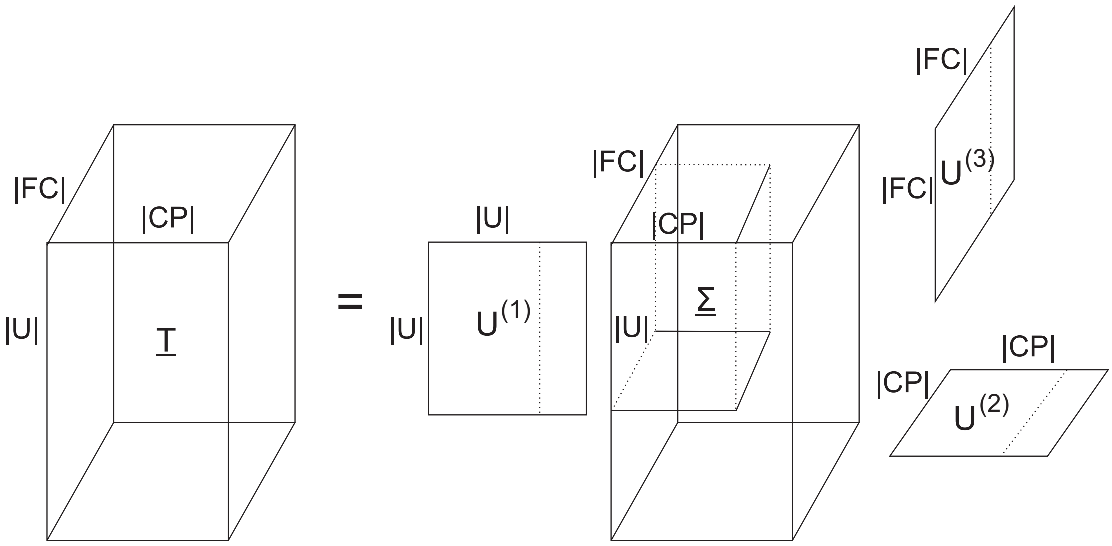

Figure 3.6: Visualisation of the decomposition of the tensor $\underline{\mathbf{T}}_{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|}$ using the HOSVD.

A complete calculation of $\underline{\mathbf{\Sigma}}$ of a given tensor $\underline{\mathbf{T}}$ using the reduced left-singular matrices $U^{(1)}_{c_1} \in \Re^{|U|\times c_1}$, $U^{(2)}_{c_2} \in \Re^{|CP|\times c_2}$, $U^{(3)}_{c_3} \in \Re^{|FC|\times c_3}$ is provided as follows:

Using the 3-mode product outlined in the equation 3.19 results in the equation 3.20.

$$\begin{aligned}\underline{\mathbf{\Sigma}}_{c_1\times c_2\times c_3} = \underline{\mathbf{T}}_{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|} \times_1 \mathbf{U}^{(1)\mathrm{T}}_{|\mathcal{U}|\times c_1} \times_2 \\ \mathbf{U}^{(2)\mathrm{T}}_{|\mathcal{CP}|\times c_2} \times_3 \mathbf{U}^{(3)\mathrm{T}}_{|\mathcal{FC}|\times c_3}\end{aligned} \tag{3.20}$$

In 3.21 the stepwise calculation of $\underline{\mathbf{\Sigma}}$ is presented.

$$\left(\underline{\mathbf{T}}_{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|} \times_1 \mathbf{U}^{(1)\mathrm{T}}_{|\mathcal{U}|\times c_1}\right) \in \Re^{c_1\times|\mathcal{CP}|\times|\mathcal{FC}|}$$

$$\left(\left(\underline{\mathbf{T}}_{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|} \times_1 \mathbf{U}^{(1)\mathrm{T}}_{|\mathcal{U}|\times c_1}\right) \times_2 U^{(2)\mathrm{T}}_{|\mathcal{CP}|\times c_2}\right) \in \Re^{c_1\times c_2\times|\mathcal{FC}|}$$

$$\underline{\mathbf{\Sigma}} = \left(\left(\left(\underline{\mathbf{T}}_{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|} \times_1 \mathbf{U}^{(1)\mathrm{T}}_{|\mathcal{U}|\times c_1}\right) \times_2 U^{(2)\mathrm{T}}_{|\mathcal{CP}|\times c_2}\right) \times_3 U^{(3)\mathrm{T}}_{|\mathcal{FC}|\times c_3}\right) \in \Re^{c_1\times c_2\times c_3} \tag{3.21}$$

In 3.22 the stepwise calculation of the tensor $\underline{\mathbf{T}}'$ is presented. $\underline{\mathbf{T}}'$ holds the original relations plus additional latent relations between $\mathcal{U}$, $\mathcal{CP}$ and $\mathcal{FC}$.

$$\left(\underline{\mathbf{\Sigma}}_{c_1\times c_2\times c_3} \times_1 \mathbf{U}^{(1)}_{|\mathcal{U}|\times c_1}\right) \in \Re^{|\mathcal{U}|\times c_1\times c_2}$$

$$\left(\left(\underline{\mathbf{\Sigma}}_{c_1\times c_2\times c_3} \times_1 \mathbf{U}^{(1)}_{|\mathcal{U}|\times c_1}\right) \times_2 U^{(2)}_{|\mathcal{CP}|\times c_2}\right) \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times c_3}$$

$$\underline{\mathbf{T}}' = \left(\left(\left(\underline{\mathbf{\Sigma}}_{c_1\times c_2\times c_3} \times_1 \mathbf{U}^{(1)}_{|\mathcal{U}|\times c_1}\right) \times_2 U^{(2)}_{|\mathcal{CP}|\times c_2}\right) \times_3 U^{(3)}_{|\mathcal{FC}|\times c_3}\right) \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|} \tag{3.22}$$

As can be seen, the sizes of the three dimensions of $\underline{\mathbf{T}}'$ are the same as the sizes of the dimensions of the original tensor $\underline{\mathbf{T}}$. Therefore it can be ensured that the approximated tensor $\underline{\mathbf{T}}'$ still holds the original relations and therefore the application of the HOSVD does not result in losing original relations that have been stored in $\underline{\mathbf{T}}$.

3.2.4 Efficient calculation of latent relations using HOSVD

The equation to calculate the approximated tensor $\underline{\mathbf{T}}'$ presented in 3.22 always calculates all possible latent relations between the entities $\mathcal{U}$, $\mathcal{CP}$, $\mathcal{FC}$ of a ubiquitous environment. This has two major drawbacks. First the calculation of $\underline{\mathbf{T}}'$ might be time consuming depending on the sizes of the three entities. Second the needed memory storage of $\underline{\mathbf{T}}'$ might be quite high because the resulting tensor is dense and contains solely 64-bit floating values. Assumed that the size of $|\mathcal{U}| = |\mathcal{CP}| = |\mathcal{FC}| = 1000$, then the overall size of $\underline{\mathbf{T}}'$ will be $10^9 * 8$ byte ≈ 7.45 GB. The algorithm presented in pseudocode snippet 1 shows a possibility to overcome the above mentioned draw-backs by only calculating all possible latent relations that exist between exactly one $U \in \mathcal{U}$ and $Cp \in \mathcal{CP}$ and $\forall\, Fc \in \mathcal{FC}$. In other words, the result of the algorithm is a tube fiber $\mathbf{t}_{U\,Cp\,:}$ as presented in Figure 3.4. The presented algorithm 1 has six input parameters. The core tensor $\underline{\mathbf{\Sigma}} \in \Re^{c_1 \times c_2 \times c_3}$ whose dimension sizes have been reduced to c_1, c_2 and c_3, the reduced left-singular matrices $\mathbf{U}_{c_1} \in \Re^{|\mathcal{U}| \times c_1}$, $\mathbf{U}_{c_2} \in \Re^{|\mathcal{CP}| \times c_2}$ and $\mathbf{U}_{c_3} \in \Re^{|\mathcal{FC}| \times c_3}$ calculated from the three different modes of the original tensor $\underline{\mathbf{T}}_{|\mathcal{U}| \times |\mathcal{CP}| \times |\mathcal{FC}|}$ and used to calculate $\underline{\mathbf{\Sigma}}$. Further parameters are the user $U \in \mathcal{U}$ and her context pattern $Cp \in \mathcal{CP}$. The output parameter of the algorithm is a tube fiber of $\underline{\mathbf{T}}'$ that holds all possible relations of $U \in \mathcal{U}$ and $Cp \in \mathcal{CP}$ and $\forall\, Fc \in \mathcal{FC}$.

In line 1 of the proposed algorithm the core tensor $\underline{\mathbf{\Sigma}}$ is decomposed in its first mode $\mathbf{\Sigma}_{c_1}$. Lines 6 to 12 show the first two steps of the 3-Mode product outlined in equation 3.22 in a simplified form. Thereby, the while loop iterates over the column vectors of the first mode. Line 11 ensures that only the column vectors of the first mode will be considered that contain information about the given context pattern $Cp \in \mathcal{CP}$. Line 7 represents the first calculation step of the proposed 3-Mode product. Thereby only these parts of $\mathbf{U}_{c_1}$ that are relevant for the calculation of the latent relations with respect to U are considered. The result is given by $\mathbf{a}_1 \in \Re^{1 \times c_2}$. Line 8 represents the second calculation step of the proposed 3-Mode product. Again, only the parts of the matrix $\mathbf{U}_{c_2}$ are considered that are relevant for the calculation of the latent relations with respect to Cp. The result is given by $\mathbf{a}_2 \in \Re^{1 \times 1}$ and stored in $\mathbf{b}$. After the while loop $\mathbf{b}$ has the form $\Re^{c_3 \times 1}$. In lines 14 to 16 the final latent relations for a given U and Cp with $\forall Fc \in \mathcal{FC}$ will be determined. Thereby, the for loop iterates $|\mathcal{FC}|$ times and calculates at each run the possible value of the relation for a given U and Cp with the respective $Fc \in \mathcal{FC}$. The result res_i has the form $\Re^{1\times 1}$ and holds the value of the relation. The final tube fiber $\mathbf{res}_{U\,Cp:}$ has the form $\Re^{1\times|\mathcal{FC}|}$. Compared to the size of $\underline{\mathcal{T}}'$ which is 7.41 GB for an assumed size of $|\mathcal{U}| = |\mathcal{CP}| = |\mathcal{FC}| = 1000$ the resulting size of $\mathbf{res}_{U\,Cp:}$ is $10^3 * 8$ byte ≈ 7.81 kB.

Data: $\underline{\mathbf{\Sigma}} \in \Re^{c_1\times c_2\times c_3}$, $\mathbf{U}_{c_1} \in \Re^{|\mathcal{U}|\times c_1}$, $\mathbf{U}_{c_2} \in \Re^{|\mathcal{CP}|\times c_2}$, $\mathbf{U}_{c_3} \in \Re^{|\mathcal{FC}|\times c_3}$, U, Cp
Result: $\mathbf{res} \in \Re^{\mathcal{FC}\times 1}$

$\mathbf{\Sigma}_{c_1} \in \Re^{c_1\times c_2 * c_3} \leftarrow \underline{\mathbf{\Sigma}} \in \Re^{c_1\times c_2\times c_3}$;
$\mathbf{b} \leftarrow [\,]$;
$start \leftarrow 1$;
$end \leftarrow c_2$;
$counter \leftarrow 0$;
while $end \leq c_2 * c_3$ **do**
  $\mathbf{a}_1 \leftarrow \mathbf{U}_{c_1}(U,:) \times \mathbf{\Sigma}_{c_1}(:, start : end)$;
  $\mathbf{a}_2 \leftarrow \mathbf{U}_{c_2}(Cp,:) \times \mathbf{a}_1^{\mathrm{T}}$;
  $\mathbf{b}[counter] \leftarrow \mathbf{a}_2$;
  $start \leftarrow start + c_2$;
  $end \leftarrow end + c_2$;
end
$\mathbf{res} \leftarrow [\,]$;
for $i \leftarrow 1$ **to** $|\mathcal{FC}|$ **do**
  $\mathbf{res}[i] \leftarrow \mathbf{U}_{c_3}(i,:) \times \mathbf{b}$;
end

Algorithm 1: Algorithm for the calculation of all possible relations between exactly one $U \in \mathcal{U}$ and $Cp \in \mathcal{CP}$ and $\forall\, Fc \in \mathcal{FC}$ given a tensor $\underline{\mathbf{\Sigma}} \in \Re^{c_1\times c_2\times c_3}$.

3.3 Illustration of the CCP approach

For the storage of the data that is used to calculate possible new context pattern in the history of a user, a 3-order tensor is used for its representation as outlined in Section 3.2.2. The tensor consists of three dimensions and stores the relations between the users $U \in \mathcal{U}$ their context patterns $Cp \in \mathcal{CP}$ and their possible future contexts $Fc \in \mathcal{FC}$. Figure 3.7 illustrates an example how data from the collaborative ubiquitous environment as presented in Figure 3.1 can be mapped to a 3-order tensor data structure $\underline{\mathbf{T}} \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|}$.

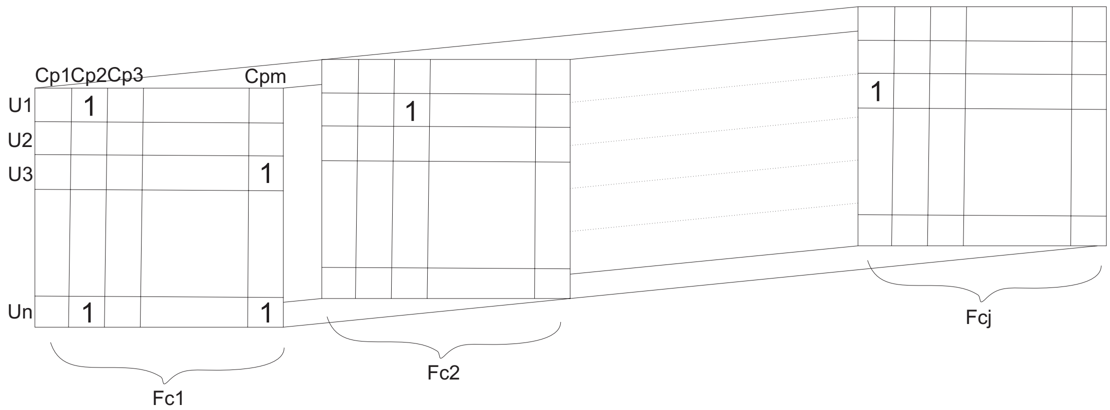

Figure 3.7: 3-order tensor $\underline{\mathbf{T}} \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|}$ to store data of a collaborative ubiquitous environment [9, 1].

The existing relations between $\mathcal{U}$, $\mathcal{CP}$, $\mathcal{FC}$, that are stored in the tensor, are given by $\mathcal{S} \subseteq \mathcal{U} \times \mathcal{CP} \times \mathcal{FC}$. Every existing relation in $\mathcal{S}$ is marked with a 1 in the tensor structure. All relations that do not exist between the users, the context patterns and the future contexts are given by $\mathcal{R} := (\mathcal{U} \times \mathcal{CP} \times \mathcal{FP}) \setminus \mathcal{S}$. All possible relations that do not exist are treated as zeros and are not displayed in the tensor data structure. As can be seen in Figure 3.7 every user has only stored a few context patterns in her history. That is because users have different and a limited amount of behaviour patterns that they follow periodically. In the context history of, e.g. the user U_3, two relations in the tensor $\underline{\mathbf{T}}$ can be seen. Context pattern Cp_m leads to the future context Fc_1 and context pattern Cp_1 leads to the future context Fc_j. HOSVD is applied to use existing relations between the context histories of the users to minimize the size of $\mathcal{R}$ the number of unknown context pattern in the user's history by finding latent relations. Therefore, the presented equations in 3.21 are used to calculate the 3-order core tensor $\underline{\mathbf{\Sigma}} \in \Re^{c_1\times c_2\times c_3}$ whose three dimension are reduced to the information that span the space that contains the most relevant information, i.e., the data whose corresponding singular values are higher than a freely chosen threshold for each dimension. Figure 3.8 shows the core tensor -

painted in red - as it can look like. The core tensor symbolises the approximation of the tensor $\underline{\mathbf{T}} \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|}$. The size of the dimension $\mathcal{U}$ is reduced to c_1, the size of the dimension $\mathcal{CP}$ is reduced to c_2 and the third dimension $\mathcal{FC}$ is collapsed to the size of c_3.

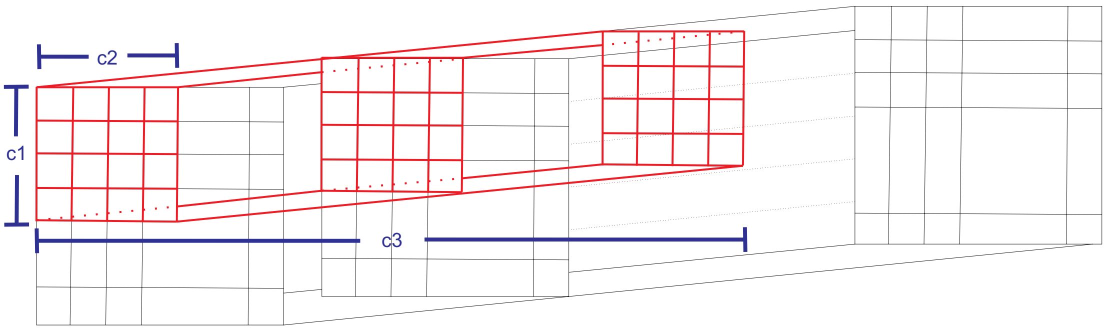

Figure 3.8: 3-order core tensor $\underline{\mathbf{\Sigma}}^{c_1\times c_2\times c_3}$ structure that contains the c_i most relevant data of every dimension [9, 1].

After calculating the core tensor $\underline{\mathbf{\Sigma}}$, HOSVD retransforms the core tensor to the initial dimension size of the tensor $\underline{\mathbf{T}}$ by reusing the n-mode product outlined in 11. Afterwards, the resulting tensor $\underline{\mathbf{T}}' \in \Re^{|\mathcal{U}|\times|\mathcal{CP}|\times|\mathcal{FC}|}$ concludes new information in terms of new relations between the three different entities $\mathcal{U}$, $\mathcal{CP}$, $\mathcal{FC}$.

In order to visualise the Collaborative Context Predictor, an illustrative example is given in this section. Please note that the algorithm presented in 1 is not used to partially calculate $\underline{\mathbf{T}}$' in this example. To provide a better understanding $\underline{\mathbf{T}}$' is completely calculated. This example consists of three users of a collaborative ubiquitous environment. The gathered high-level contexts of a user are respectively stored in her own context history. In total, there exist five different contexts: W = *W*alking, S = *S*itting, T = s*T*anding, D = going stairs *D*own, U = going stairs *U*p. The context histories represent parts of real gathered high-level contexts and characterise a user's movement behaviour. Section 3.4.1 provides detailed information on the data set.

Figure 3.9 presents the collaborative ubiquitous environment which consists of the three context histories of the users. As outlined, the collaborative ubiquitous environment consists of three entities: the users, their context patterns and their possible future contexts. Equal context parts in the histories of the users are marked the same colour. As illustrated in Figure 3.1, every context part consists of exact one context pattern $Cp \in \mathcal{CP}$ and exact one future context $Fc \in \mathcal{FC}$. The size of the context parts is equally determined to a window size of four in this example. In general the chosen window size depends on the number of contexts stored in a user's context history and on the number of different contexts. The length of $|Cp| = 3$ and the length of $|Fc| = 1$. Overall, there are five different contexts, five different context parts and each context history includes eight context parts in this example.

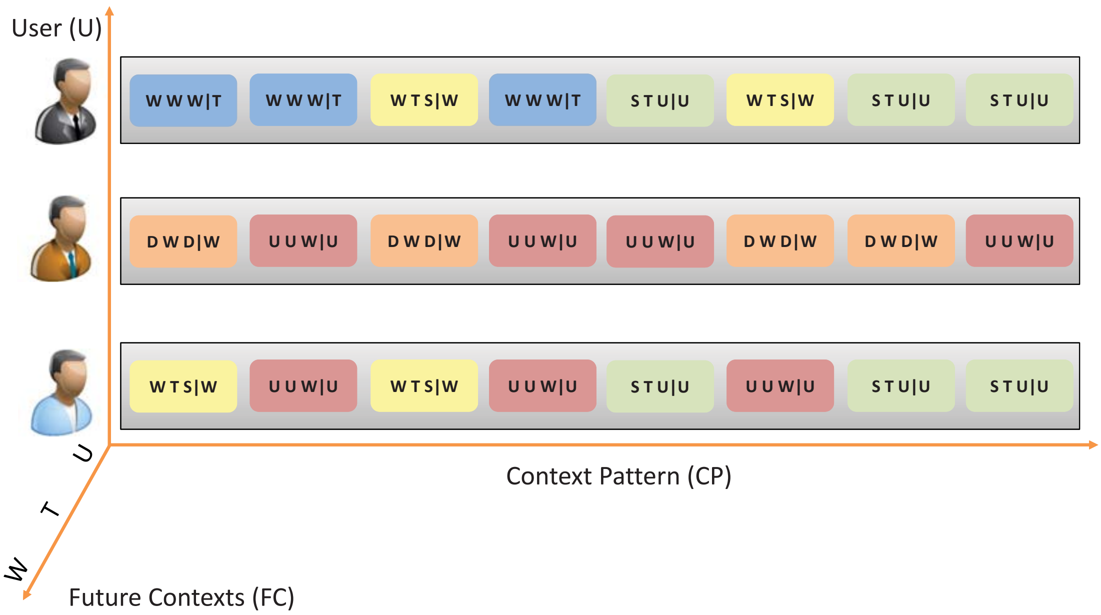

Figure 3.9: Collaborative Ubiquitous Environment consisting of three different users. Each history of a user comprises contexts that describe a user's different movement behaviours.

As it can be seen the context history of U_1 does not provide information for the context pattern {DWD, UUW}, the history of U_2 does not provide information for the context pattern {WWW, WTS, STU} and in the history of U_3 the context patterns {WWT, DWD} haven't been stored yet. Hence, e.g. it is not possible to provide a prediction for next sensed context pattern to U_1 if this context pattern is either DWD or UUW. To ensure the prediction of the users' unknown context patterns the CCP approach to find existing latent relations between the histories is applied. The approach takes advantage of direct and indirect relations between the histories of the users. Direct relations are characterised by equal context parts between two users. Indirect relations between two users, e.g. U_1 and U_2 exist if the following two conditions are fulfilled:

- U_1 and U_2 do not have the same context pattern,
- U_3 features similarities of both U_1 and U_2.

To be more precise:

- A direct relation $U_i \oplus U_j$ between two users $U_i \in \mathcal{U}$ and $U_j \in \mathcal{U}$ exists if $U_i \oplus U_j \Leftarrow \exists Cp_n \in \mathcal{CP} \vee \exists Fc_m \in \mathcal{FC} : \underline{\mathbf{T}}(U_i, Cp_n, Fc_m) \neq 0 \wedge \underline{\mathbf{T}}(U_j, Cp_n, Fc_m) \neq 0$.
- An indirect relation between two users U_i and U_j exists if $\neg(U_i \oplus U_j) \wedge (U_i \oplus U_k \wedge U_j \oplus U_k)$.

The direct and indirect relations between the three users are further visualised by Figure 3.10. The context parts located in the intersection between two users represent an existing direct relation between the two users. As a result, $U_1 \oplus U_3$ and $U_2 \oplus U_3$ and $\neg(U_1 \oplus U_2)$. The indirect relation between U_1 and U_2 implies that both users share different context parts with another user U_3, which can be understood as a friend of a friend relationship. These indirect respectively latent relations between the users can be used to additionally enrich the context histories of a user. If for example the current inferred context pattern of U_1 is DWD or UUW, both are unknown to U_1, the CCP approach can be used to automatically extract possible indirect relations between U_1 and the other users in a collaborative ubiquitous environment to enrich the context history of U_1. As a result, a possible predict of the next walking behaviour of U_1 based on a previously unknown movement pattern DWD can be provided.

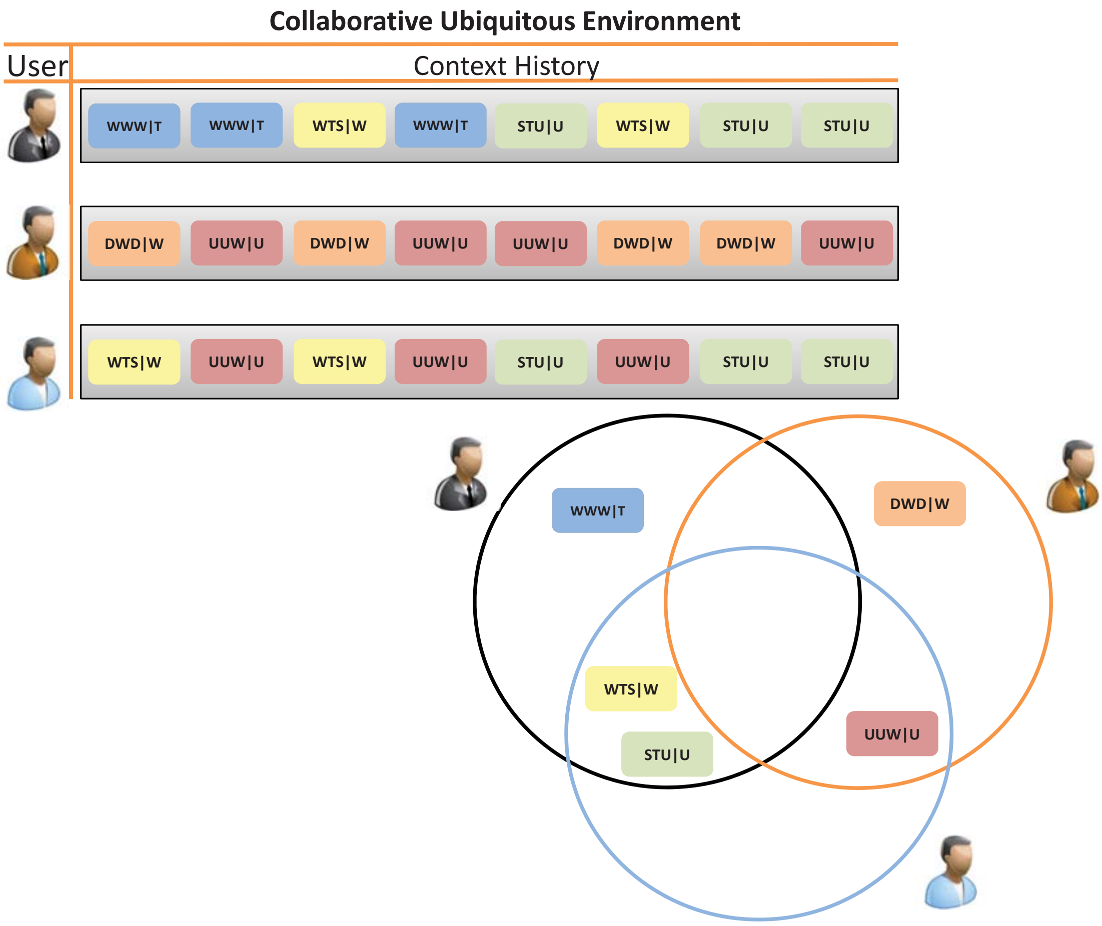

Figure 3.10: Exemplarily outlines the existing direct and indirect relations between three users in an collaborative ubiquitous environment.

To enrich the histories of the users with further information, that can be used to predict suitable future contexts for so far unknown context patterns, e.g. for U_1, the context histories of the users of an collaborative ubiquitous environment are mapped to a 3-order tensor structure. The tensor represents the existing direct relations between the different entities. With regard to the collaborative ubiquitous environment outlined in Figure 3.9 a 3-order tensor structure with size $\underline{\mathbf{T}} \in \Re^{3\times5\times3}$ results that represents the information of 3 users, five different context pattern and 3 different future contexts. The mapping of the information of the collaborative ubiquitous environment to the 3-order tensor structure is outlined in Figure 3.11. The first dimension of the tensor represents the users $\mathcal{U}$, the second dimension represents the different context patterns $\mathcal{CP}$ existing in the collaborative ubiquitous environment and the third dimension represents the different existing future contexts $\mathcal{FC}$.

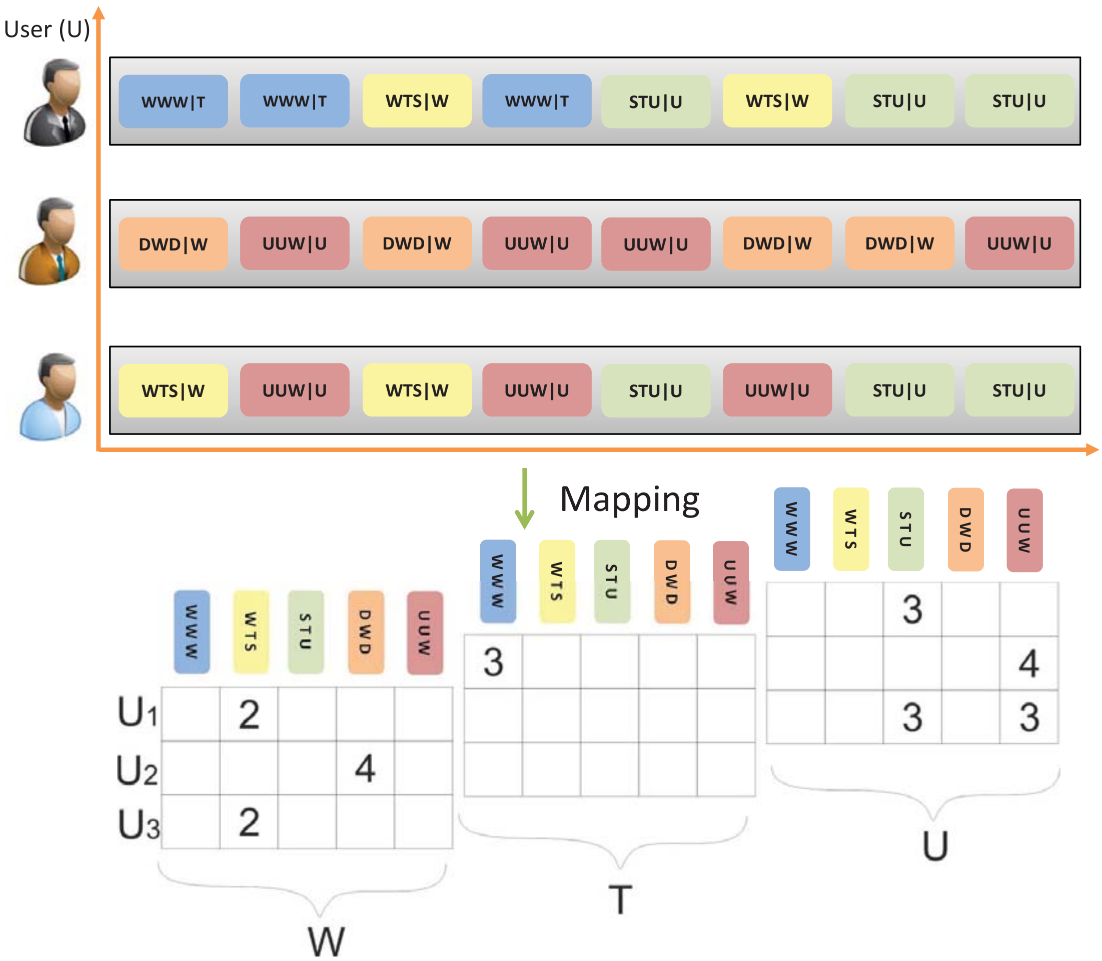

Figure 3.11: Mapping of the existing relations between the entities of a collaborative ubiquitous environment to a 3-order tensor structure.

Afterwards, the HOSVD is applied on the 3-order tensor to calculate the core tensor $\underline{\mathbf{\Sigma}}$ that spans the information space that only contains the most relevant information of the collaborative ubiquitous environment. The resulting core tensor structure is $\underline{\mathbf{\Sigma}} \in \Re^{1\times5\times3}$ in this example. Subsequently, based on the reduced information space, the tensor $\underline{\mathbf{T}}'$ that includes additional latent relations between the entities of the collaborative ubiquitous environment, is calculated. The latent relations in $\underline{\mathbf{T}}'$ are symbolised by new existing values. Because context patterns do not map to multiple future contexts in this example the relations between users, context patterns and future context in the resulted tensor $\underline{\mathbf{T}}'$ are unique. Otherwise, the future context with the highest value, that forms a relation to a given user and context pattern, forms the most probable relation. The transformation of tensor $\underline{\mathbf{T}}$ to $\underline{\mathbf{T}}'$ is presented in Figure 3.12. In the top of the picture the transformation of the underlying information space that changes during the calculation process is outlined for further visualisation. In the beginning, the original information space whose dimensions are weighted according to their corresponding singular values can be seen. Subsequently, the information space is reduced to the amount of eigenvectors that contain the most relevant information ($\underline{\mathbf{\Sigma}}$). Finally, the reduced eigenspace is expanded to the size of the original information space, whereby the calculation of the relations between the entities only bases on the reduced eigenspace. As a result an enriched information space is received that contains additional latent relations between the entities of the collaborative ubiquitous environment. The increase of information in tensor $\underline{\mathbf{T}}'$ can be seen by its additional values that represent new existing relations. To provide a prediction for the context pattern DWD and UUW to U_1, the new relations, provided by the resulted tensor $\underline{\mathbf{T}}'$, can now be utilised. For the pattern DWD, Walking can be predicted and for UUW, going stairs Up can be predicted. The resulting prediction for DWD is plausible, because U_1 shares the same context patterns STU, WTS with U_3 and therefore has a direct relation. The same applies to the prediction for pattern UUW, because U_1 has a direct relation to U_3 and U_3 has a direct relation to U_2. For this reason U_1 and U_2 have an indirect relation. The example shows that CCP can be used to enrich the context histories of users by using existing direct or indirect relations to context histories of other users.

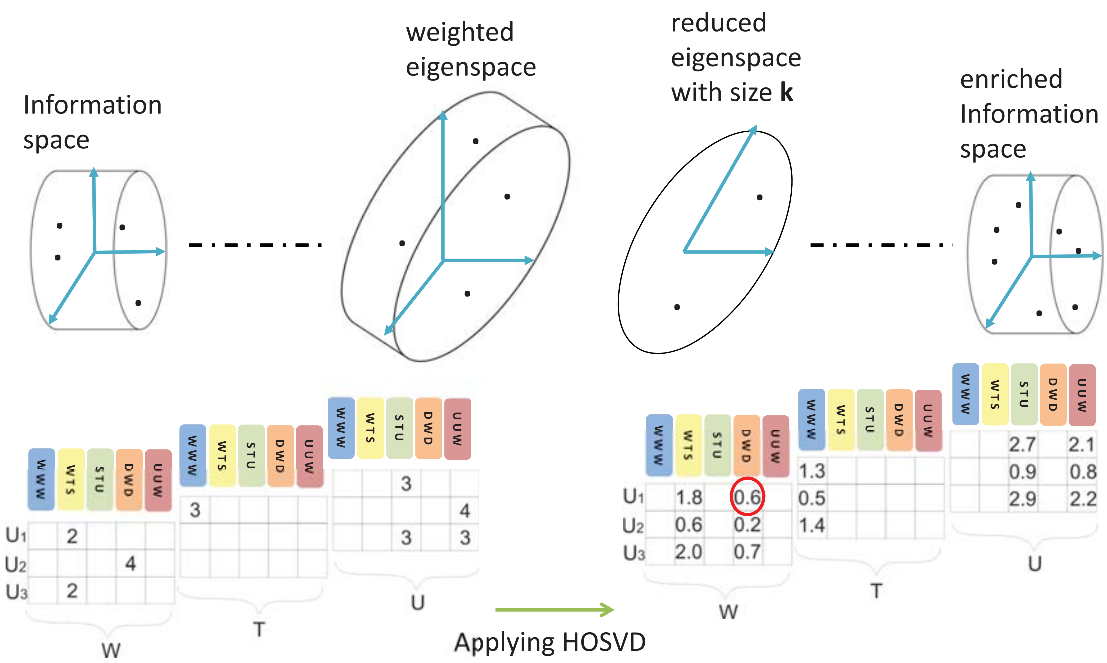

Figure 3.12: Outlines the tensor $\underline{\mathbf{T}}$ and the resulting tensor $\underline{\mathbf{T}}'$ after applying the HOSVD.

3.4 Evaluation

In this section a proof-of-concept of the CCP approach is given. CCP is evaluated in three different experiments. In the first experiment, which is outlined in Section 3.4.1, CCP is compared with the state of the art context prediction approaches introduced in Section 2.2. For the evaluation a data set that contains movement data of different users derived from a smartphone is used. In the second experiment (cf. Section 3.4.2) the number of algorithms that are compared to CCP is extended by three well-known data mining algorithms. For the evaluation the Augsburger data set [10] and a modification of the Augsburger data set is used. In the third experiment, which is presented in Section 3.4.3, a first person shooter is used to generate large synthetic and highly collaborative data sets including in-game movement behaviours of controlled characters. Subsequently, these data sets are used to evaluate the proposed context predictors.

3.4.1 Experiment 1

In this section the CCP approach is compared with the ActiveLeZi predictor, the Alignment predictor and the StatePredictor regarding the prediction accuracy using a real world context data set. The data set contains acceleration data, which has been recorded using the accelerometer in a smartphone. The acceleration data has been annotated with the movements performed by users, similar to the experiments performed in [11]. Smartphones used to record the accelerations of the users include a Nokia N95 8GB, a 5730 Xpressmusic and a N900. Each user was equipped with a smartphone, kept in the trouser pocket. The users performed five different movements {sitting, standing, walking, go upstairs, go downstairs}. The annotation of a user's movement behaviour was performed with a Nokia N800 Internet Tablet using a graphical logging tool. Afterwards, the recorded acceleration data and the performed annotations for each user were combined using a script. To transform the recorded time series (acceleration data) to a string representation, the Symbolic Aggregate approXimation (SAX) proposed by Lin et. al [12] was used. The transformation is necessary because the context prediction approaches utilised in the evaluation only work on contexts, which are represented as strings. The magnitude of the raw acceleration data was transformed into a sequence of alphabetical symbols, to transform the numerical data to data described by strings. In order to achieve this, each time series was normalised and split into windows with a length of four seconds. Next, each window was transformed into a piecewise aggregate approximation (PAA) representation of the given normalised time series. Finally, the PAA representation was converted into strings, consisting of different symbols, using the distance matrix defined in Lin's work. After applying SAX to the acceleration data, strings that represent the movement behaviour of a user with a length of 32 are obtained. Each string represents exactly one movement of a user, e.g. *sitting*. In total, the string consists of six different symbols {a, b, c, d, e, f}. To gain a higher number of possible different contexts the different movement behaviours of the users were subdivided. This was achieved by using a simple clustering approach that clusters each string sequence into three sub clusters. Sub clusters of the string representations for "go downstairs" are, e.g. represented by {B0, B1, B2}, sub clusters of the string representations for "go upstairs" are represented by {C0, C1, C2}, etc. To cluster the string representations gained by SAX, the applied clustering algorithm uses the distance matrix presented in [12]. As a result, each context history of a user consists of 15 different high-level contexts that finally characterise the movement behaviours of the respective user. Figure 3.13 illustrates an exemplary time series, each with a window size of four seconds. The left time series characterises the movement pattern "going upstairs" and the right time series "going downstairs". The red curves represent the time series of the recorded acceleration data. The blue curves indicate the transformation of the raw acceleration data into the alphabetical symbols. The data snippet below shows the label, the SAX transformation of the acceleration data and the respective sub cluster label of a user's recorded movement pattern.

For the experiments four context histories of different users were used. Every context history contains approximately 1000 sub-clustered and time ordered context information, resulting from the recorded movement behaviours of the users. The labels represent the contexts, which form a basis for a prediction approach to predict a user's next context. To provide a representative number of different evaluation sets, the context histories were split using different sizes three, five and seven for the length of the context parts (cf. Section 3.3). A context part with a size of three for example, consists of two sub cluster labels representing the context pattern $Cp \in \mathcal{CP}$ and one sub-cluster label representing the future context $Fc \in \mathcal{FC}$. That means that the prediction of a user's next context bases on the two last seen contexts of a user. Moreover, additional evaluation sets were generated by removing context parts that occurred successively. Data sets without successive context parts are called *single-mode* data sets, data sets with successive context parts are called *all-mode* data sets. Altogether, there are six different types of data sets generated from a user's context history used to evaluate the prediction accuracy of the different approaches.

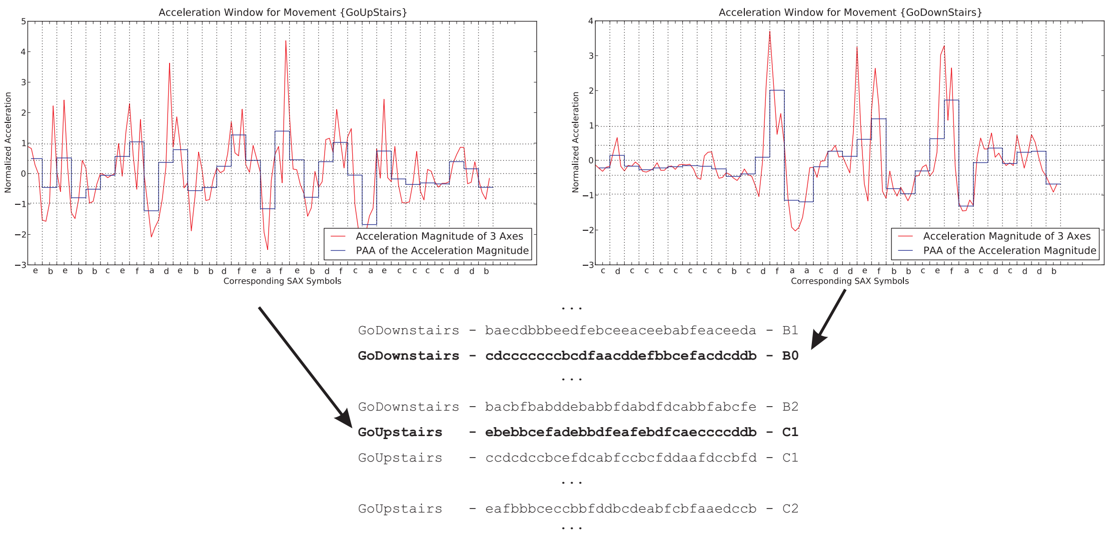

Figure 3.13: Transformation of the acceleration data into a symbolic representation using the SAX approach. Below, a snippet of a user's movement history is outlined [1].

The CCP, the Alignment predictor, the ActiveLeZi predictor and the StatePredictor approaches were applied to various test data sets for the evaluation. A test data set consists of the intersections of the context histories $H_i \cap H_j = \{H_i, H_j \,|\, H_i \in \mathcal{H} \vee H_j \in \mathcal{H}\}$ of two users U_i, U_j. Intersections represent equal context parts that occur in both histories H_i, H_j. For every test data set, there are three training data sets that are used to build the prediction models for the respective context prediction approach. The first training data set contains the information of the context histories of the two users; the test data set has been generated from. The second training data set extends the information of the first one by adding the context history of a third user. The third data set extends the second by adding the history of a fourth user. For the prediction process the "leave-one-out" strategy was used for the three context prediction approaches. For this reason, each single context part of the test data set is removed temporally one after the other from the context history of the corresponding user in the current training data set. Every time a single context part is removed temporally, the prediction model is constructed anew with the reduced training data set and the built model is used to predict the future context of the context pattern given by the current context part of the test data set. Finally, the predicted context is compared with the future context also given by the current context part afterwards.

Altogether, 24 different test data sets were generated. The test data sets result from the calculation of the intersections of different combinations of context history pairs. The intersection of the context histories of U_1 and U_2, U_2 and U_3, U_1 and U_3 and U_2 and U_4 were calculated. In addition, each combination was combined with the aforementioned three lengths of the context parts and the two different data set modes *single-mode* and *all-mode*. The prediction accuracy of CCP compared with the accuracy gained by the ActiveLeZi predictor, the Alignment predictor and the StatePredictor is presented in the following.

Interpretation of the results

In this section, the prediction results received by the different approaches are compared and interpreted. Figures 3.14, 3.15, 3.16, 3.17, 3.18, 3.19, 3.20 and 3.21 present and compare the gained prediction accuracy of CCP and ActiveLeZi. The accuracy results of CCP compared with the results of the Alignment predictor are presented in Figures 3.22, 3.23, 3.24, 3.25, 3.26, 3.27, 3.28 and 3.29. Finally, Figures 3.30, 3.31, 3.32, 3.33, 3.34, 3.35, 3.36 and 3.37 present the results of CCP compared to the StatePredictor approach. Each figure presents the gained accuracy of the approaches on the y-axis. On the x-axis the different sizes of the context parts are outlined. For each context part size three, five and seven, there exist three accuracy results for each prediction approach. In total, there are six bars for each context part size.

1. How does the accuracy of the approaches evolve depending on the number of context histories used to train the prediction model?

2. How does the accuracy of the approaches evolve according to the chosen context part size?

3. How do the different data set modes (single, all) affect the prediction accuracy of the approaches?

4. How does CCP perform compared with the respective state of the art prediction approaches?

The only exception can be found in the presentation of the accuracy results of the alignment predictor. Each figure contains a legend, which outlines the colour of the respective prediction approach according to the applied number of user context histories that are used to build the prediction model. The results of the CCP approach are presented in different shades of blue. The results of the ActiveLeZi approach in different shades of red. The results of the Alignment predictor in light red and the results of the StatePredictor approach are presented in different shades of green. The results gained by the CCP approach vary depending on the different used data sets. Compared to the different state of the art predictors they remain the same. Table 3.1 presents the different sizes of the tensor structures ($\underline{\mathbf{T}}_1, \underline{\mathbf{T}}_2, \underline{\mathbf{T}}_3$) used by the CCP approach to store the relations of the collaborative ubiquitous environment that consists of the users, their different movement behaviours and their possible next movements. The sizes of the dimensions vary because the number of used context histories differs from data set to data set. For the calculations of the core tensor $\underline{\mathbf{\Sigma}}$ only the dimensionality of the users has been reduced. A larger number of experiments, where all of the three dimension $\mathcal{U}, \mathcal{CP}, \mathcal{FC}$ have been reduced to different sizes, showed that the reduction of $\mathcal{U}$ to a size of one, while the sizes of $\mathcal{CP}, \mathcal{FC}$ remain constant, always leads to the best prediction results. Therefore, the calculation of $\underline{\mathbf{T}}'$ always bases on the core tensor $\underline{\mathbf{\Sigma}} \in \Re^{1 \times |\mathcal{CP}| \times |\mathcal{FC}|}$. Further, Table 3.1 outlines the different amounts Ψ of context patterns (intersections between the context histories of a user pair) stored in the respective test data set used to evaluate the prediction approaches. In the following, the prediction results of CCP compared with the three state of the

Table 3.1: Tensor dimensionality for the different context part sizes of the four test data sets.

	context part size = 3				context part size = 5				context part size = 7			
Fig.	$\underline{\mathbf{T}}_1$	$\underline{\mathbf{T}}_2$	$\underline{\mathbf{T}}_3$	Ψ	$\underline{\mathbf{T}}_1$	$\underline{\mathbf{T}}_2$	$\underline{\mathbf{T}}_3$	Ψ	$\underline{\mathbf{T}}_1$	$\underline{\mathbf{T}}_2$	$\underline{\mathbf{T}}_3$	Ψ
3.14, 3.15, 3.22, 3.23, 3.30, 3.31	2x44x11	3x63x11	4x68x11	54	2x104x11	3x136x11	4x183x11	34	2x123x11	3x184x11	4x250x11	18
3.16, 3.17, 3.24, 3.25, 3.32, 3.33	2x53x11	3x64x11	4x66x11	48	2x130x10	3x180x10	4x232x10	32	2x142x10	3x209x10	4x275x10	12
3.18, 3.19, 3.26, 3.27, 3.34, 3.35	2x54x11	3x59x11	4x67x11	44	2x125x11	3x160x11	4x205x11	28	2x125x11	3x176x11	4x239x11	10
3.20, 3.21, 3.28, 3.29, 3.36, 3.37	2x61x11	3x68x11	4x75x11	68	2x143x10	3x188x10	4x232x10	38	2x153x11	3x221x11	4x277x11	16

art context predictions will be discussed according to the following points:

CCP compared to ActiveLeZi predictor Figure 3.14 and Figure 3.21 compare the gained prediction results of CCP and the ActiveLeZi predictor. According to point 1 the prediction accuracy of the CCP and ActiveLeZi predictor increases in nearly all cases if the number of context histories, that can be used as knowledge bases, exceed the amount of context histories the test data set has been generated from. An exception to this rule is given by the outline of the prediction result of CCP for the context part size of three in Figure 3.21. In this case no improvement of the prediction accuracy was achieved. ActiveLeZi always increases its prediction accuracy if additional context histories are used. It can be recognised that the usage of four context histories does not automatically lead to better prediction results. Figures 3.18, 3.16 exemplarily show this effect. Some reasons for this effect are discussed later in this section. According to point 2 both, the CCP and the ActiveLeZi approach, received better prediction results the higher the size of the context parts is. This is due to the fact that the higher the size of the context part the less is the number of the available context parts that can be created from the context histories of the users. According to point 3 it can be recognised that the prediction of CCP for the all-mode data sets achieves better results than for the single-mode data sets. This is due to the fact that the repeated occurence of equal context parts can be utilised by CCP to make a more profound prediction decision. The prediction decisions of the ActiveLeZi approach, however, are only barely effected by the repeated occurence of equal context parts. The results presented in Figure 3.16 and Figure 3.17 even show no differences in the achieved prediction accuracy of ActiveLeZi. According to point 4 the results presented on the eight figures indicate that the CCP approach outperforms ActiveLeZi in predicting the future context of a user based on unknown context patterns by 20% to 50%, varying from the data set.

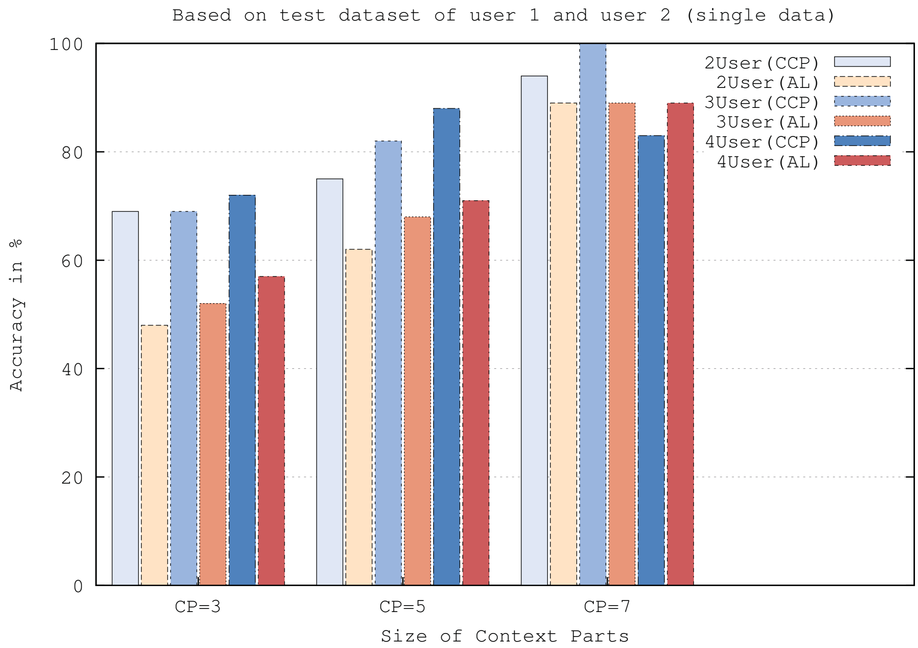

Figure 3.14: Test data sets generated from the intersections of U_1 and U_2. Single data mode used. Number of context parts is 54, 34 and 18.

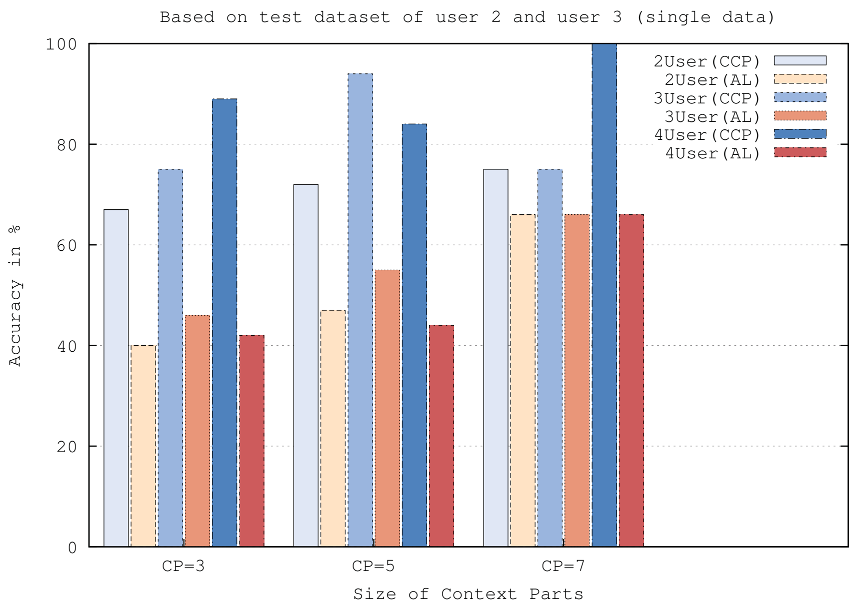

Figure 3.16: Test data sets generated from the intersections of U_2 and U_3. Single data mode used. Number of context parts is 48, 32 and 12.

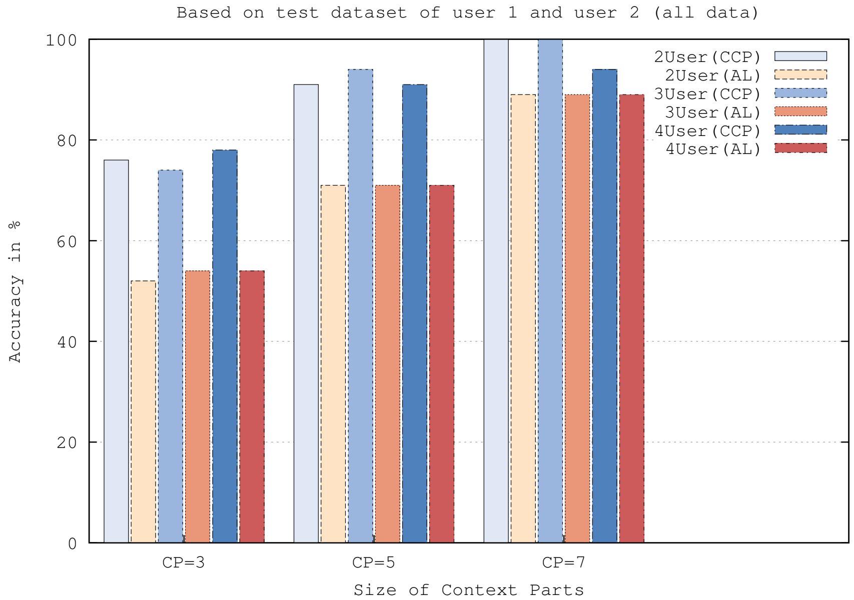

Figure 3.15: Test data sets generated from the intersections of U_1 and U_2. All data mode used. Number of context parts is 54, 34 and 18.

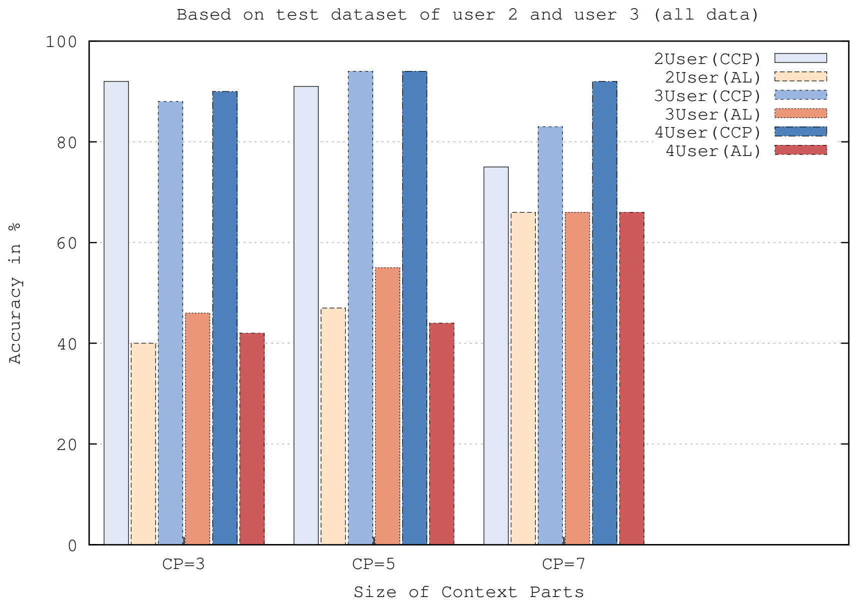

Figure 3.17: Test data sets generated from the intersections of U_2 and U_3. All data mode used. Number of context parts is 48, 32 and 12.

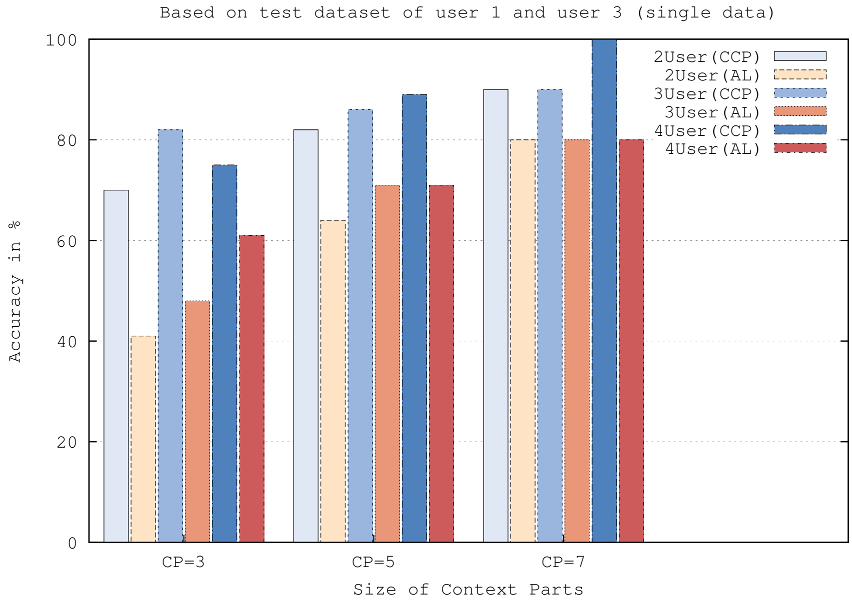

Figure 3.18: Test data sets generated from the intersections of U_1 and U_3. Single data mode used. Number of context parts is 44, 28 and 10.

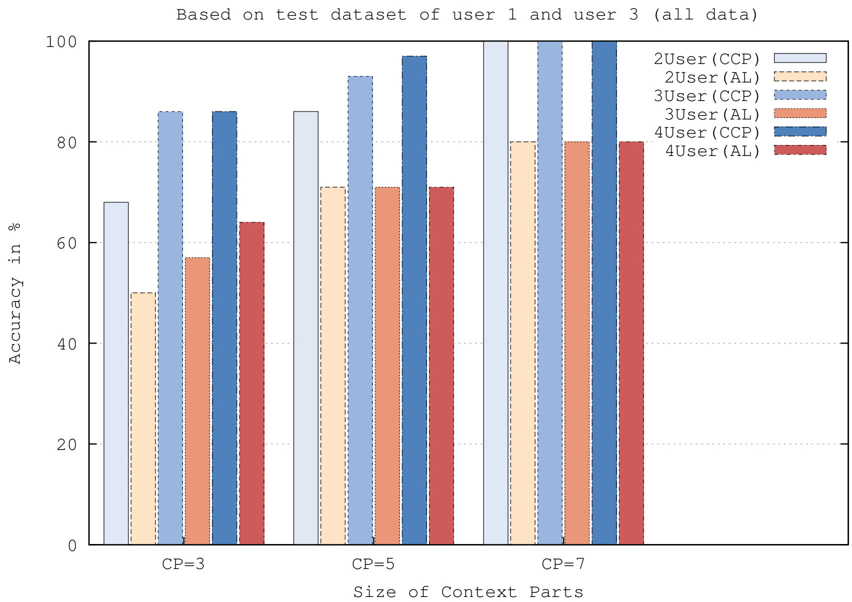

Figure 3.19: Test data sets generated from the intersections of U_2 and U_3. All data mode used. Number of context parts is 44, 28 and 10.

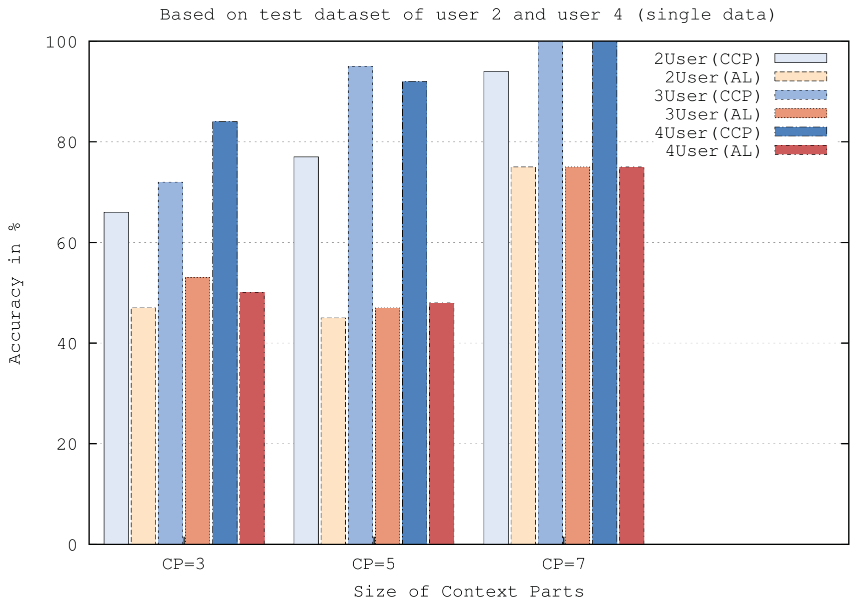

Figure 3.20: Test data sets generated from the intersections of U_2 and U_4. Single data mode used. Number of context parts is 68, 38 and 16.

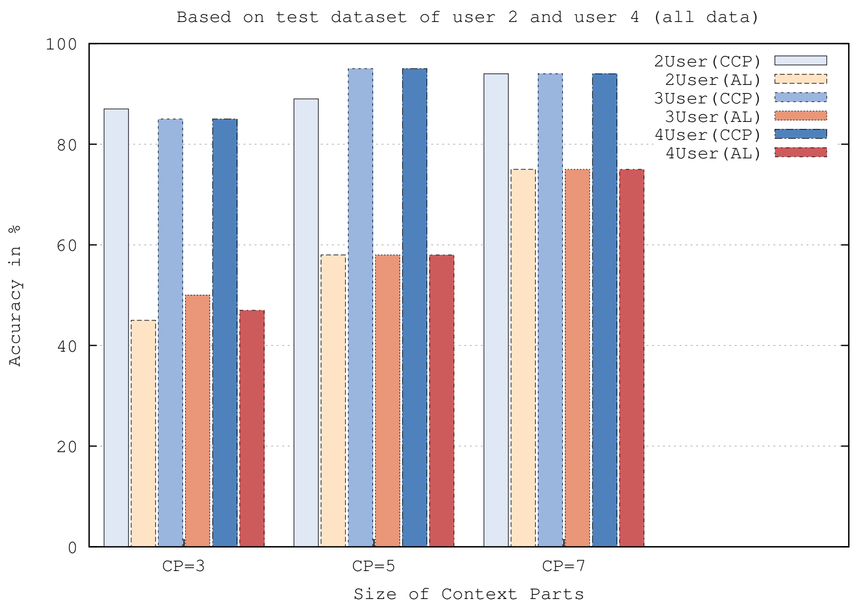

Figure 3.21: Test data sets generated from the intersections of U_2 and U_4. All data mode used. Number of context parts is 68, 38 and 16.

CCP compared to Alignment predictor Figure 3.22 to Figure 3.29 compare the gained prediction results of CCP and the Alignment predictor. In the following, only the results received by the Alignment predictor are discussed. The results of CCP have already been presented in Figures 3.14 to 3.21. According to point 1 the received results show that the Alignment predictor always gets the same prediction results, regardless whether it uses two, three or four context histories. This indicates that the Alignment approach gets most of the information it needs to make a reliable prediction from the two context histories the test data set has been generated from. Therefore, in this experiment, it does not benefit from additional context histories of the users. The light red bar in the figures represents the prediction accuracy of Alignment for all three different knowledge base sizes. According to point 2 the Alignment predictor continuously increases its prediction accuracy the higher the size of the context parts is. The distribution of the context histories in context parts with different sizes show that the prediction accuracy is less accurate for smaller context parts sizes and is getting more accurate for higher context part sizes. That is because the smaller the chosen context part size, the higher the number of entries in the test data set and the more ambiguous is the data in the data set. Point 3 shows that the single-mode data sets mostly lead to less accurate prediction results by the Alignment predictor in comparison to its prediction results received by using the all-mode data sets. CCP provides more accurate prediction results than the Alignment approach for all examined test data sets, as can be seen in point 4. CCP outperforms Alignment depending on the chosen data set by 20% to 40%.

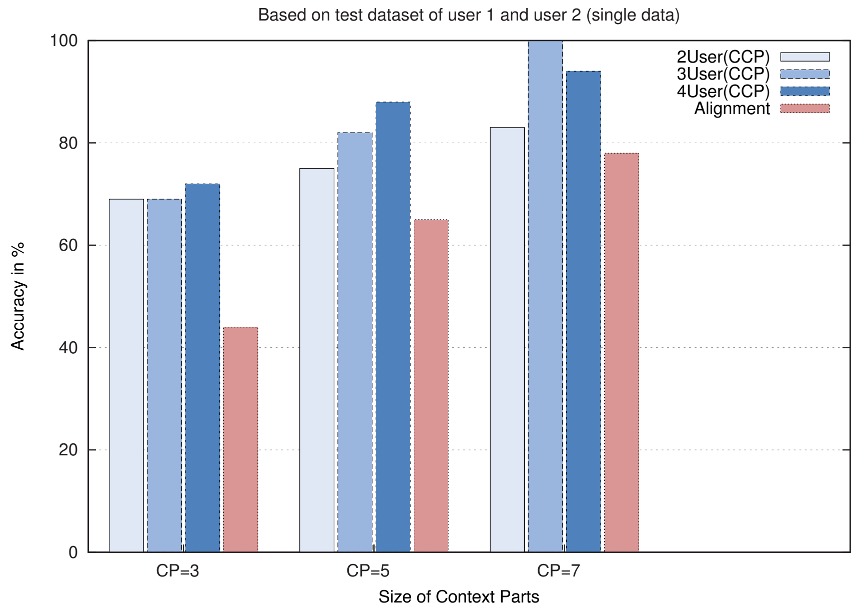

Figure 3.22: Test data sets generated from the intersections of U_1 and U_2. Single data mode used. Number of context parts is 54, 34 and 18.

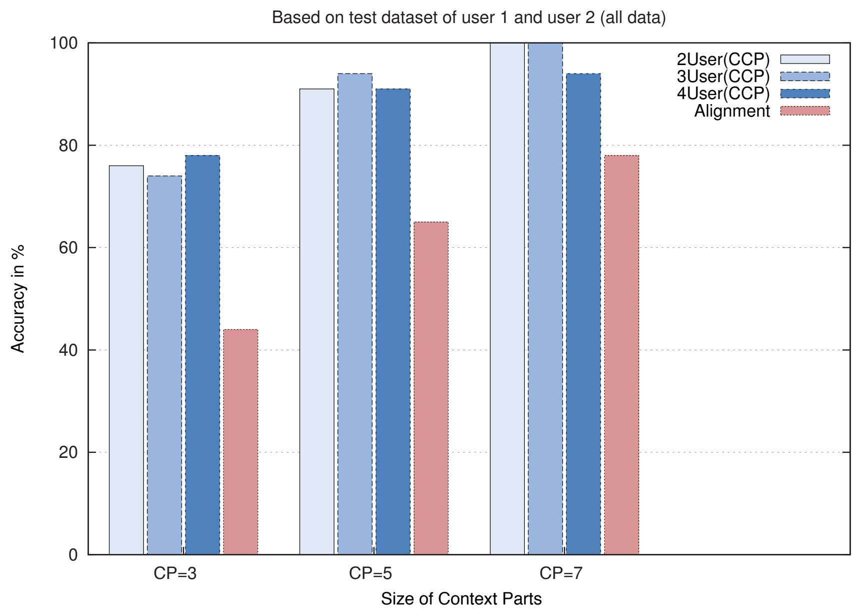

Figure 3.23: Test data sets generated from the intersections of U_1 and U_2. All data mode used. Number of context parts is 54, 34 and 18.

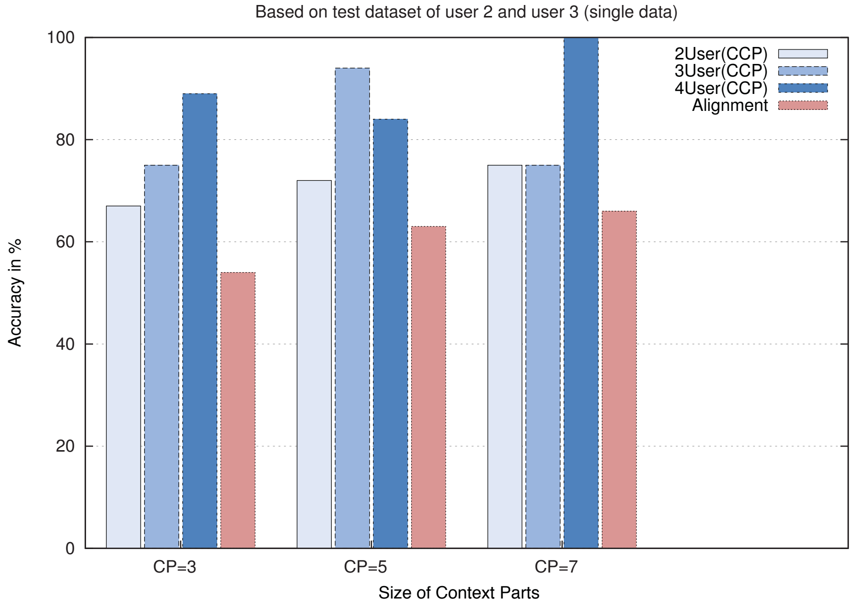

Figure 3.24: Test data sets generated from the intersections of U_2 and U_3. Single data mode used. Number of context parts is 48, 32 and 12.

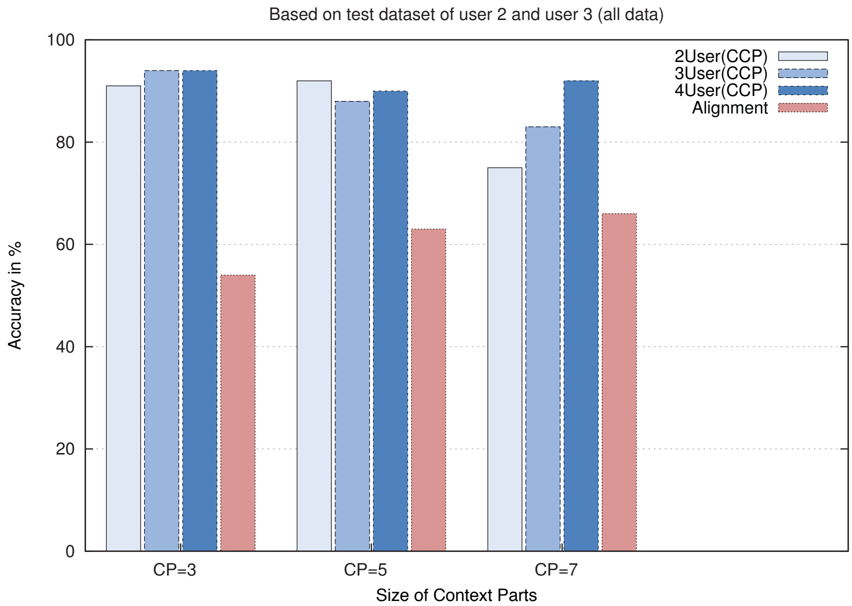

Figure 3.25: Test data sets generated from the intersections of U_2 and U_3. All data mode used. Number of context parts is 48, 32 and 12.

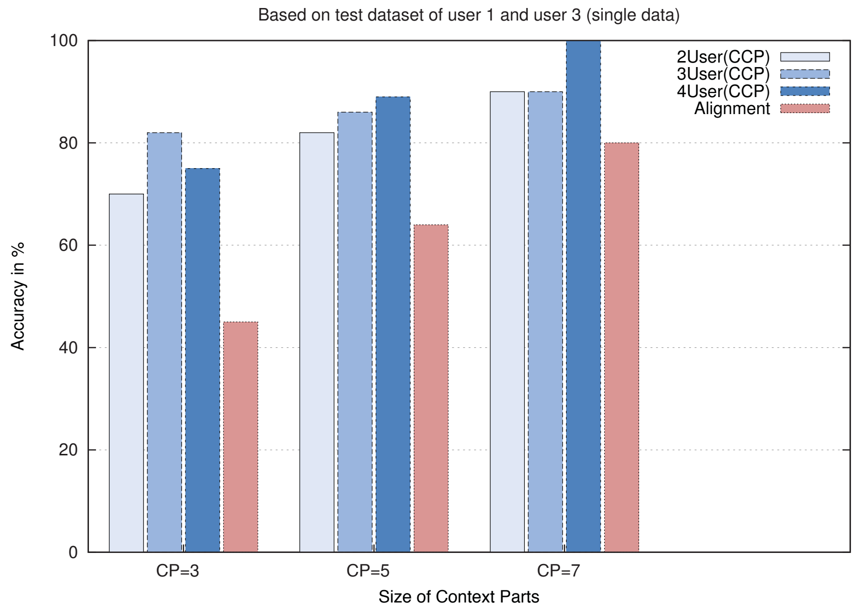

Figure 3.26: Test data sets generated from the intersections of U_1 and U_3. Single data mode used. Number of context parts is 44, 28 and 10.

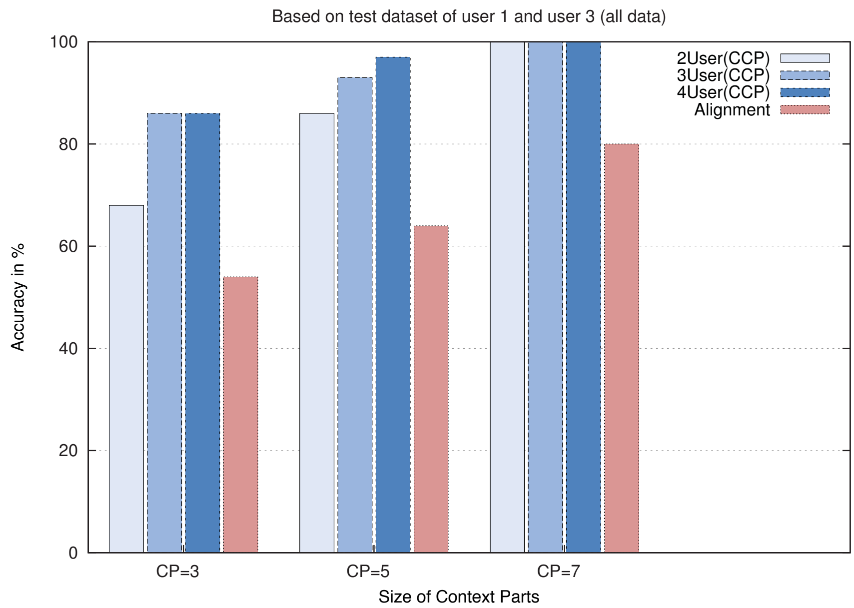

Figure 3.27: Test data sets generated from the intersections of U_1 and U_3. All data mode used. Number of context parts is 44, 28 and 10.

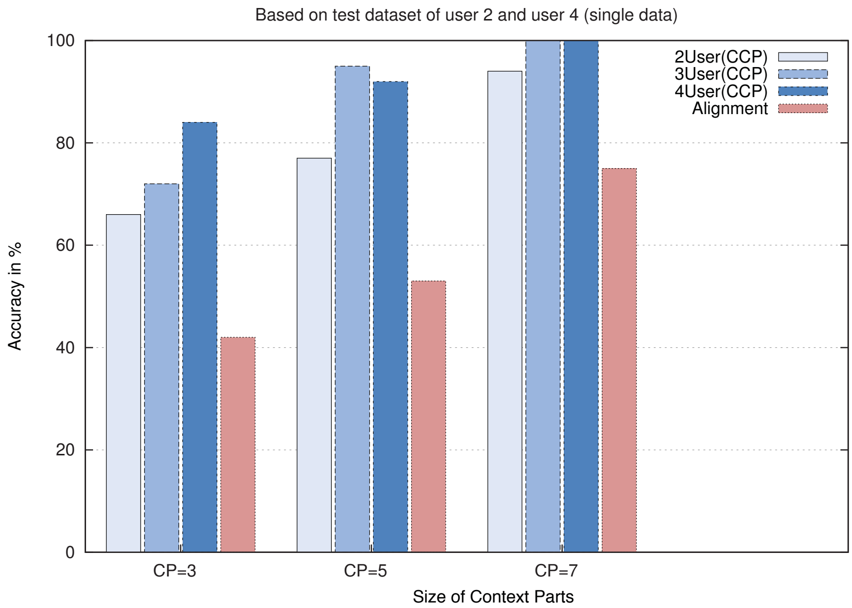

Figure 3.28: Test data sets generated from the intersections of U_2 and U_4. Single data mode used. Number of context parts is 68, 38 and 16.

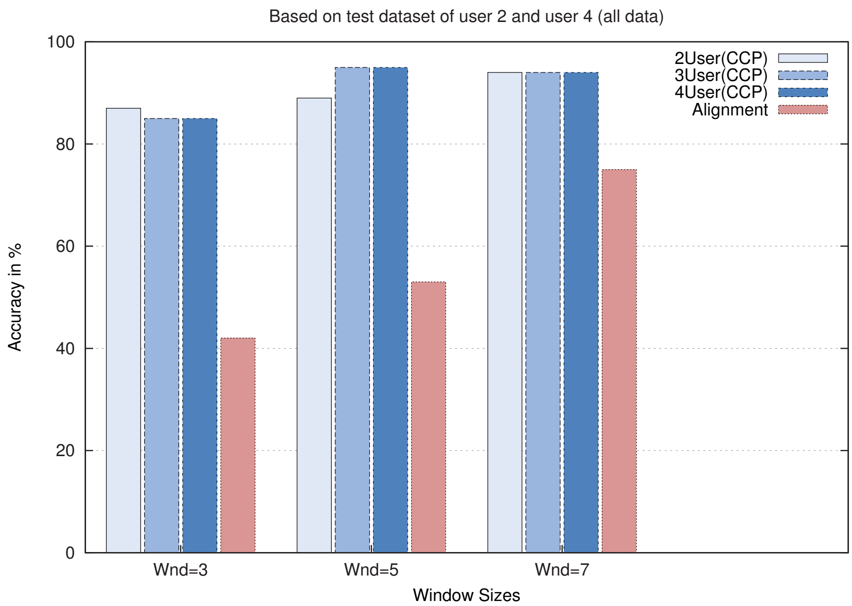

Figure 3.29: Test data sets generated from the intersections of U_2 and U_4. All data mode used. Number of context parts is 68, 38 and 16.

CCP compared to StatePredictor Figure 3.30 to Figure 3.37 compare the gained prediction results of CCP with the results received by the StatePredictor. In the following, only the results received by the StatePredictor are discussed.

According to point 1 the received results of the StatePredictor outline that in contrast to the CCP, the Alignment predictor and the ActiveLeZi predictor, that could mostly increase their prediction accuracy using additional context histories, the accuracy of the StatePredictor mostly decreases while using additional context histories. This effect can exemplarily be seen in Figure 3.30, 3.31 and 3.35.

Point 2 shows that the StatePredictor always receives the worst results while using the smallest size for the context parts compared with the Alignment, ActiveLeZi and the CCP approach. If a context part size of five or seven is used, the StatePredictor is competitive to Alignment and ActiveLeZi.

The single-mode data sets mostly lead to less accurate prediction results by the StatePredictor approach in comparison to its prediction results received using the all-mode data sets, as can be seen in point 3. An exception is given by the results in Figure 3.36. Here, the predictor receives better results on the single-mode data set than on the all-mode data set.

According to point 4 CCP provides more accurate prediction results than the StatePredictor approach for all examined test data sets except for the single-mode data set created of the intersections of U_2 and U_4 using a context part size of seven (cf. Figure 3.36). CCP outperforms the StatePredictor approach depending on the chosen data set up to 50% cf. Figure 3.33.

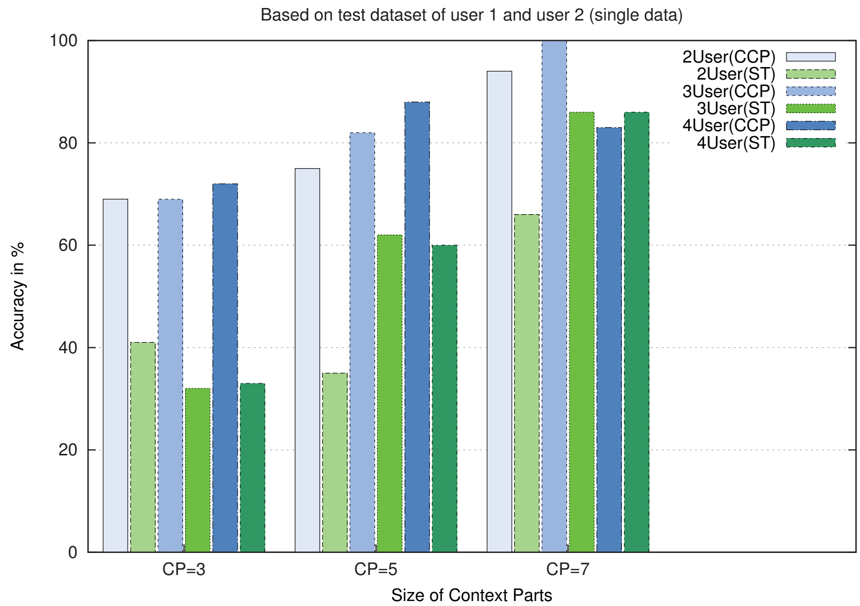

Figure 3.30: Test data sets generated from the intersections of U_1 and U_2. Single data mode used. Number of context parts is 54, 34 and 18.

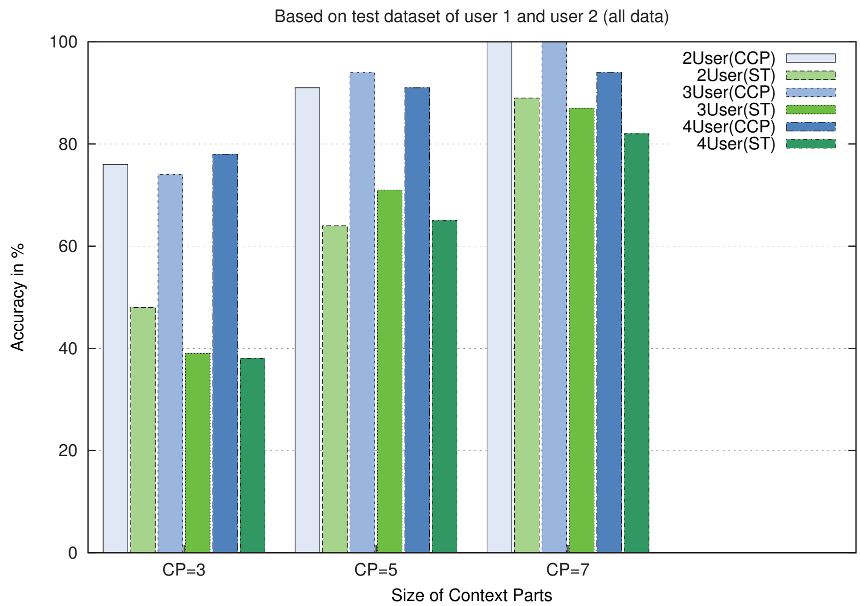

Figure 3.31: Test data sets generated from the intersections of U_1 and U_2. All data mode used. Number of context parts is 54, 34 and 18.

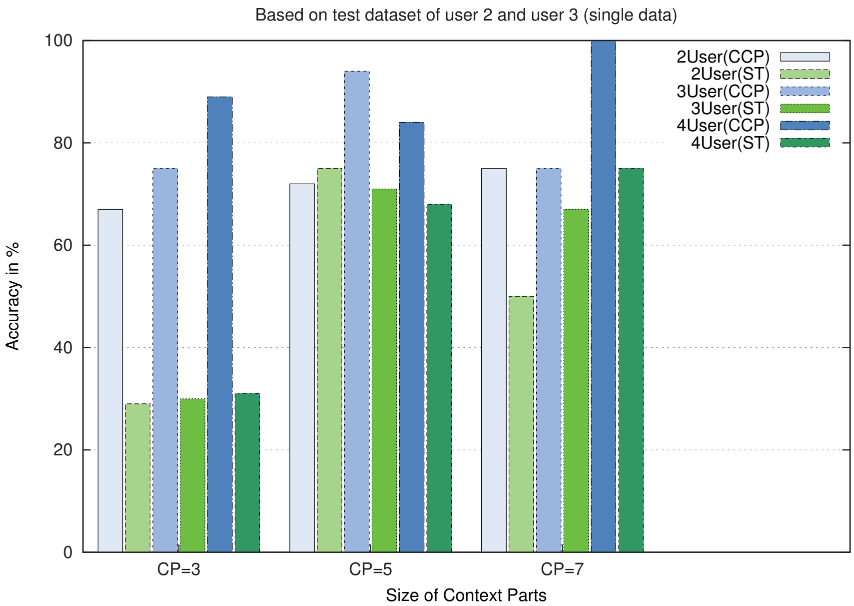

Figure 3.32: Test data sets generated from the intersections of U_2 and U_3. Single data mode used. Number of context parts is 48, 32 and 12.

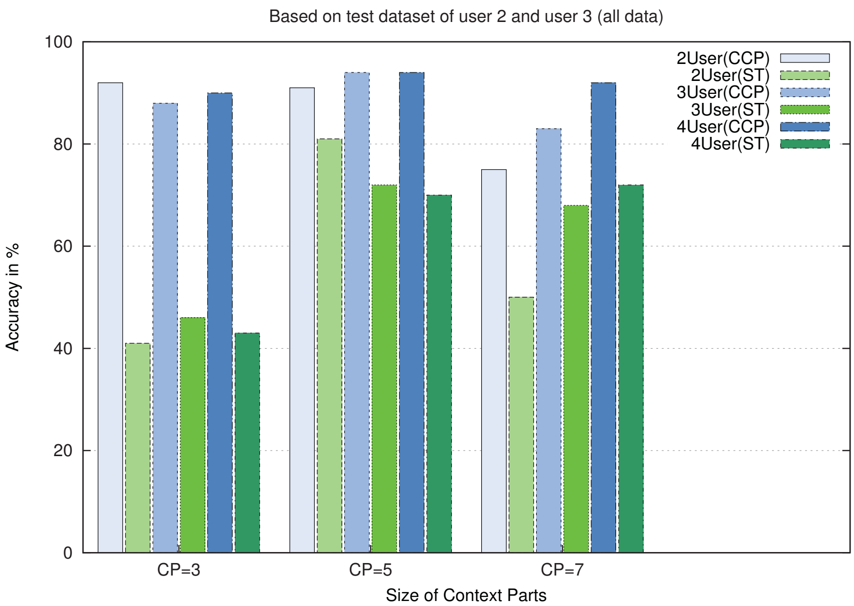

Figure 3.33: Test data sets generated from the intersections of U_2 and U_3. All data mode used. Number of context parts is 48, 32 and 12.

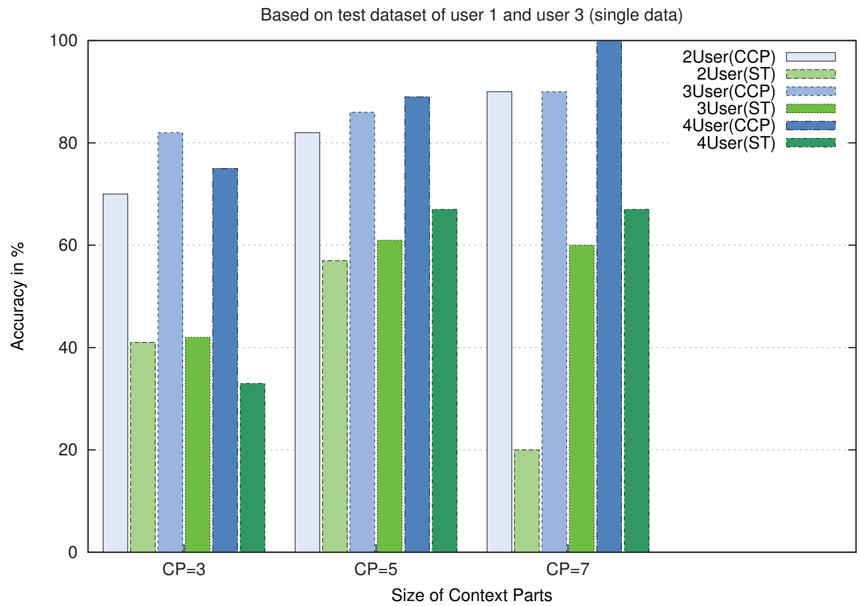

Figure 3.34: Test data sets generated from the intersections of U_1 and U_3. Single data mode used. Number of context parts is 44, 28 and 10.

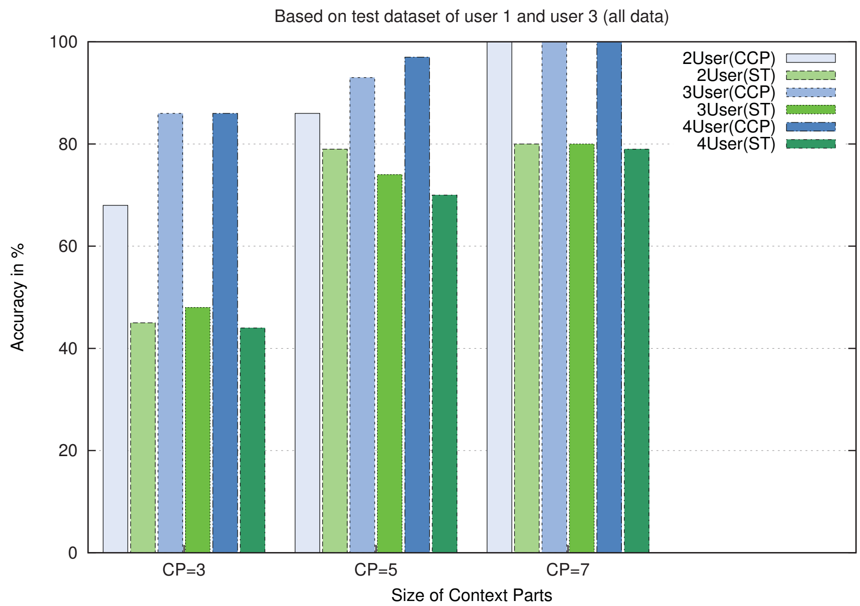

Figure 3.35: Test data sets generated from the intersections of U_1 and U_3. All data mode used. Number of context parts is 44, 28 and 10.

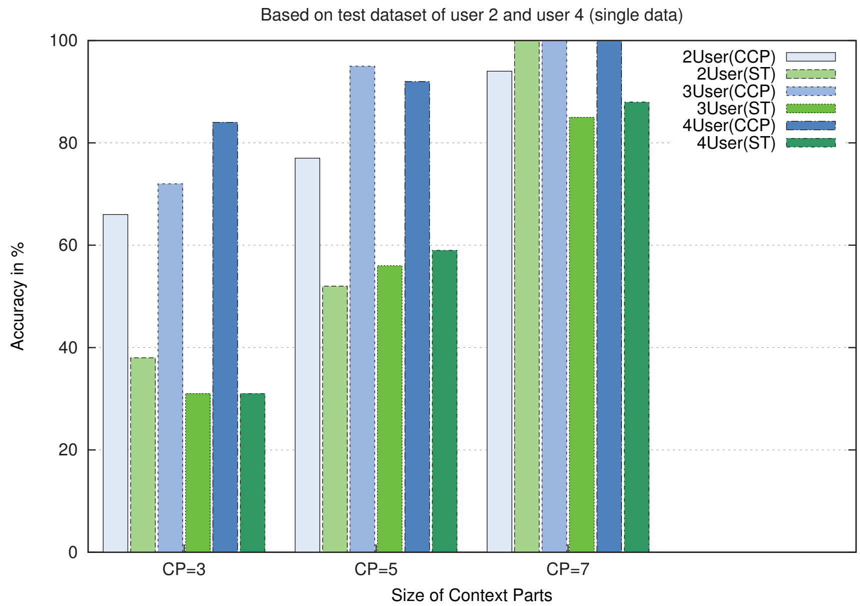

Figure 3.36: Test data sets generated from the intersections of U_2 and U_4. Single data mode used. Number of context parts is 68, 38 and 16.

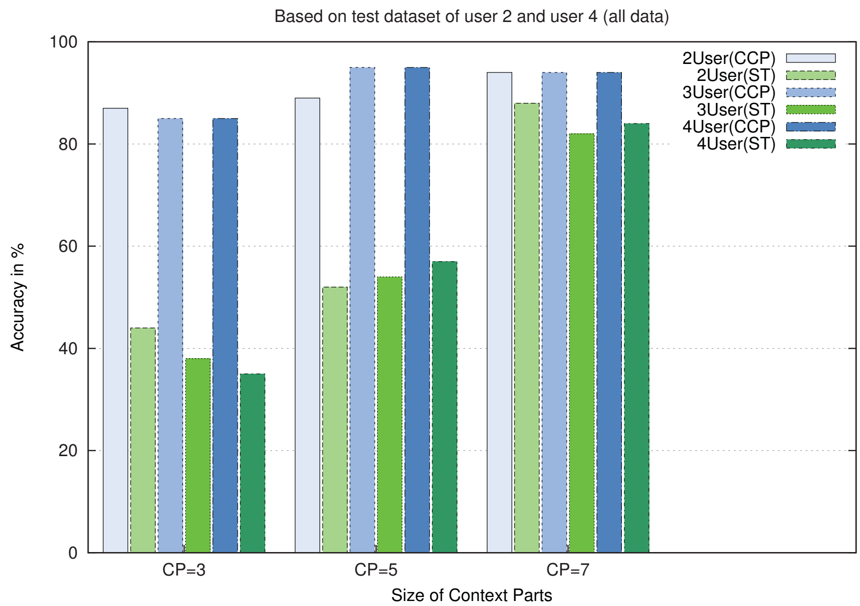

Figure 3.37: Test data sets generated from the intersections of U_2 and U_4. All data mode used. Number of context parts is 68, 38 and 16.

Summary of Experiment 1 In the presented experiment, the CCP approach has been compared to three state of the art context prediction approaches that have been introduced in Section 2.2. To compare the approaches, they have been used to predict a user's next movement based on her own context history and based on additional context histories of other users. The used movement data have been recorded using the acceleration sensor of a smartphone carried in the users' trouser pockets. The numerical acceleration data were pre-processed using the SAX approach to receive string representations of the contexts. The context histories have been segmented into context parts using different window sizes of three, five and seven. Afterwards, the intersections (equal context parts) between different pairs of context histories were determined and stored in different test data sets. Each context part of a test data set is removed temporally one after the other by the context history of the corresponding user in the appropriate training data set to simulate missing context information in a user's context history. Altogether, 24 different test data sets were generated. The experiment showed that CCP is able to obtain quite accurate prediction results, even if the underlying context information is missing in the context history of the respective user. Furthermore, the results indicate that CCP almost consistently receives better prediction accuracy than the ActiveLeZi predictor, the Alignment predictor and the StatePredictor approach.

Finally, it can be noticed that the prediction accuracy of the different approaches is not consistantly getting better if additional context histories are added. The reason for this can be the increase of ambiguity meaning that a context pattern induces to different future contexts, which makes a decision for a context prediction approach even more difficult.

3.4.2 Experiment 2

In this section the CCP approach and the three state of the art context prediction techniques presented in Section 2.2 are compared with three common data mining classification techniques. The reason is to evaluate, on the one hand, which technique performs best and, on the other hand, to evaluate if common and well-known classification techniques outperform algorithms specifically used in the field of context prediction. A further aspect of this experiment is the evaluation of the CCP approach using a publicly available and well-known context data set. As data mining classifiers a J48 tree classifier, a Bayesian Network classifier and a classifier based on Decision Tables have been used. The context prediction approaches have been implemented in Java. The data mining algorithms were used from the Weka Data Mining Software [13].

For the evaluation of the algorithms two data sets were used. The first data set is the Augsburger data set introduced in [10, 14]. The second data set is a slightly modified version of the Augsburger data set, called Augsburger_2 in this experiment. In this modified version a sliding window approach with a window size of four was used to add additional data to the data set. The aim was to increase the number of existing direct relations between the context histories of the different users that are encapsulated in this data set. The coefficient Θ outlined in Table 3.2 indicates the average similarity between the four context histories of the users that are encapsulated in the Augsburger data set and in its modified version.

The Augsburger data set basically consists of location information of four different persons. The location data were collected manually by using a graphical user interface implemented on a PDA. With the help of this graphical user interface, each user labels her current location. Locations are different rooms in an office building (cf. Figure 3.38). For each person two data sets have been recorded. One data set contains data collected in the summer period one data set contains the data that were collected in the fall period. In this experiment the summer and the fall data of a user were combined to one history. Hence, each user has exactly one data set respectively one context history that contains the user's location contexts.

Afterwards, each data set was segmented using a windows size of four. Consequently, every context history of a user consists of context parts with a size of four, whereby the fourth context

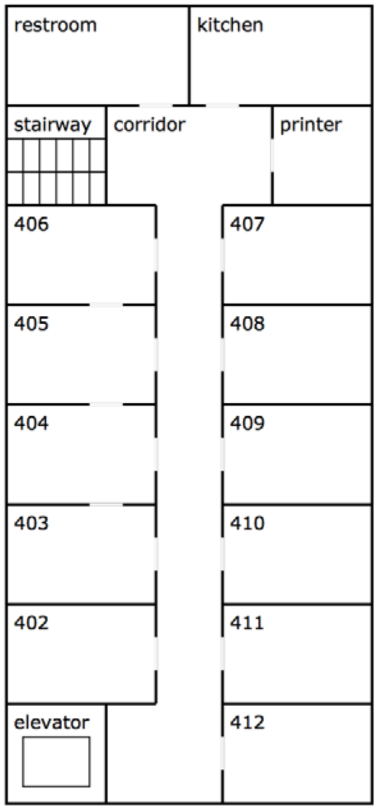

Figure 3.38: Graphical user interface used to label the locations of the users and a map of the office building the data were collected in [14].

symbolises the future contexts that should be predicted after seeing the first three contexts (context pattern). In contrast to the presented evaluation in [15], which used the summer data sets of the users to train the prediction models and the fall data set to evaluate the trained prediction models, in this experiment the test data were extracted from each context history to evaluate the prediction accuracy of the context prediction approaches and the data mining algorithms. In total, approximately 8% of the context data of each context history of a user were randomly picked and stored in a test data set. For the evaluation of the prediction accuracy of the different approaches the context histories of the users were concatenated to one history. All algorithms, except the CCP approach, used the concatenated histories as data basis to train or to create their prediction models. The final prediction accuracy of the different approaches result from calculating the average value of the four prediction accuracies received by applying the approaches to the four test data sets. The prediction results are outlined in Table 3.2.

The first three rows present the results of the data mining techniques. The second four rows the results of the context prediction approaches and Θ presents the similarity coefficient of the related data set. This coefficient specifies the similarity of the different user context histories of a data set. To calculate the similarity coefficient, the context histories of the users have been split into contexts parts using the sliding window approach outlined in Figure 2.5 in Chapter 2 first. Subsequently, the occurrences of the context parts in the different context histories were compared to each other. Finally, Θ results from average accordances of the context parts in the histories of the users.

Table 3.2: Prediction accuracies of the different evaluated prediction and data mining approaches.

	Augsburger	Augsburger_2
BayesNet	55.6%	60%
DecissionTable	44.9%	57.5%
J48 Tree	54%	58%
ActiveLeZi	55%	13%
Alignment	55%	11%
StatePredictor	61%	57%
CCP	28%	63%
Θ	0.6%	32%

The results show that the evaluated context prediction approaches ActiveLeZi, Alignment and StatePredictor achieved results that are slightly higher than those received by the well-known data mining approaches for the not modified version of the Augsburger data set. CCP received the lowest accuracy rate on the Augsburger data set. An explanation for this result can be given considering the similarity coefficient Θ. As outlined, the average similarity of the four context histories is approximately 0.6%, which is extremely low. With respect to the average number of context parts (approximately 132) that are stored in the context histories of the users in the Augsburger data set, an average similarity of 0.6% indicates that there exist only three to four equal context parts in a pair of context histories on average. Apparently, this is a not sufficient direct relations for the CCP approach to make reliable predictions. With respect to the Augsburger_2 data set the CCP approach received the best prediction accuracy. This is due to the fact that the Θ coefficient of the modified Augsburger_2 is higher than the Θ coefficient of the Augsburger data set. Therefore, the number of existing relations (equal context parts) between the context histories of the users is higher, because of adding data using a sliding window approach. The data mining approaches could also improve their prediction results. However, the prediction accuracy of the Alignment and the ActiveLeZi approach dropped drastically. The reason for this can be the fact that using the sliding window approach comes along with adding ambiguous information to the data set.

3.4.3 Experiment 3

In this section an evaluation of the CCP approach and the approaches presented in Section 2.2 using a synthetic data set, extracted from a first person shooter, is presented. One of the most challenging aspects in context prediction, besides the considerations of a user's trust and privacy, is the collection of context data, needed to evaluate context prediction approaches. This is due to different reasons. One reason is that public environments, private houses and working places become more and more ubiquitous but the installation of sensors and access to the data derived by these sensors is not easy and not always possible. This is the case, because personal data is mostly private and can not be accessed. And even if the collection is permitted, the infrastructure to collect these data often does not exist or is heterogeneous. Therefore, the collection of context data often takes place in controlled environments and is also limited to a certain use case. Examples can be found in [14], [16] and [17]. A further reason that complicates the acquisition of context data is the time and the manpower needed to collect sufficient data. A possibility to generate context data in a short period of time and with nearly no need of manpower is to simulate a certain use case. Therefore, e.g. Siafu [18], an open source context simulator tool, can be used. The disadvantage of using context simulation tools is the fact that the generated context data is simply simulated and provides no evidence that persons would act the same way if they were in the same situation in real life. The same problem may occur if real persons act in a controlled and closed environment, because people tend to behave unnaturally if they feel observed.

A possible solution to the above mentioned challenges can be given by computer games, e.g. by so-called first person shooters. These games have some characteristics that can be considered as advantages for the simple simulation of contexts or the acquisition of contexts by persons in controlled environments. The most significant advantages are the following:

- Computer games, e.g. first person shooter games are valid candidates to be considered as ubiquitous environments. First, a first person shooter provides an environment users can move around in, they perform actions and interact with other users. Second, first person shooters contain a large number of information that can be considered as users' contexts such as locations, behaviour patterns, scores, used items, locations of items, etc. One very prominent 'computer game' not a shooter game but a good example to motivate the consideration of computer games to be seen as ubiquitous environments can be "Second Life" [19].

- Gathered contexts such as, e.g. locations, velocity or used items in first persons shooter games are not simulated but correspond to the actual behaviour of a character in the game.

- Characters are intuitively and naturally controlled by users.

- Users follow realistic goals. These goals can be, e.g. to win against an opponent, to defend or to attack certain points in the game or to follow certain movement behaviours.
- Games, e.g. first person shooter games are highly collaborative. This includes group behaviours and tactics.
- People around the world play computer games, respectively first person shooter games. For this reason, a high amount of context data can be generated rapidly.

In [20] a first person shooter game called 'Quake III Arena' has already been used to demonstrate and test context-aware services. In detail, the authors used the first person shooter to overcome the problem of missing location-based context data to test their service during the development process. To broadcast the sensor values derived from the game to the service, the authors encapsulate their Quake III Arena modification as a sensor in the well-known Context Toolkit widget [21]. The broadcasted sensor values, in this specific case, the location of the character in the game, is used to simulate the current GPS coordinate of a user utilised by the service that is under development. To provide an evaluation of the CCP approach using a collaborative data set with a greater extend, a version of Quake III Arena has been modified to extract in-game locations of different characters as contexts during their gameplay [22]. Subsequently, the extracted contexts have been used to predict a character's next in-game position. One implementation has provided real time location-based context prediction of the characters in the game, based on their previously collected location contexts. A second implementation has used a higher number of collected in-game location contexts of the characters to predict their next in-game location offline. The second implementation has been used to provide an evaluation of the prediction accuracy of the different approaches.

For the collection of the context data using Quake III Arena, six users played two different maps. The maps did not include teleports or jump pads to ensure realistic movement behaviour of the characters controlled by the users. Table 3.3 presents the characteristics of the games on the two different maps.

Table 3.3: Characteristics of the two games.

properties	map 1	map 2
map name	am_lavactf	ps37ctf2
play time (train)	30 min.	37.5 min.
play time (test)	20 min.	15 min.
ratio train/test	3:2	5:2
# of different contexts	1242	1599
players	6	6

Each position, respectively context, extracted from the characters in the game consists of a x-, y-, and z-coordinate. In the game, the position of a character is determined nearly pixel wise. To gain a manageable number of different in-game location contexts, the raster that determines the different location points of the characters in the game has been increased. This is achieved by dividing the different coordinates by a variable factor. The higher the factor is chosen the coarser the division of the map. In the two experiments, each coordinate has been divided additionally by a factor of 300. As a result on map 1, in total, 1242 different location contexts were collected by the users and on map 2, in total 1599 different location contexts were collected by the users (cf. Table 3.3). The contexts derived from the training and the test game for each map were stored in the training and test data set of the respective user. The training data sets served as knowledge bases for the context prediction approaches and the test data sets were used to evaluate the context prediction approach with respect to their prediction accuracy.

For the evaluation of the prediction approaches the six context histories, containing the training data, were merged to one training history and the six histories containing the test data were also merged into one test history (cf. Table 3.4). Based on these two

Table 3.4: Number of instances in the training and test context data sets.

user	**map 1**		**map 2**	
	train	test	train	test
U1	1514	951	1592	592
U2	1552	975	1554	633
U3	1620	964	1728	689
U4	1638	966	1607	619
U5	1490	953	1476	627
U6	1518	967	1562	679
Σ	9332	5776	9519	3839

training and test data sets, the prediction results of the approaches were carried out. The prediction accuracies achieved by the context predictors on map 1 are outlined in Figure 3.39, the prediction accuracies on map 2 are outlined in Figure 3.40.

On both maps, the CCP approach outperforms the state of the art prediction approaches. The Alignment predictor and the ActiveLeZi predictor have the most difficulties with the high number of different possible future contexts that might follow a given context sequence. The same effect could have already be seen in the results of the second experiment after increasing the number of existing relations between the users in the Augsburger data set in Section 3.4.2. Remarkable are the reliable prediction results of the StatePredictor, which also outperform the Alignment and the ActiveLeZi predictors although the StatePredictor is the most simple prediction approach from all the evaluated approaches. The CCP approach is able to predict almost 80% of the characters' next steps correctly on map 1 and almost 68% on map2. The achieved results of the CCP approach demonstrate that it can take advantage of the existing relations between the movement behaviours of the six played characters in the game best. The results have also shown that the extraction and usage of synthetic in-game location data of first person shooters is a promising approach to evaluate context prediction approaches.

Further, the contexts in the data sets were split into instances of four contexts. That implies that the fourth location context of a user is predicted based on her three previously seen location contexts. Table 3.4 presents the number of instances of each training and test data base for each user depending on the map.

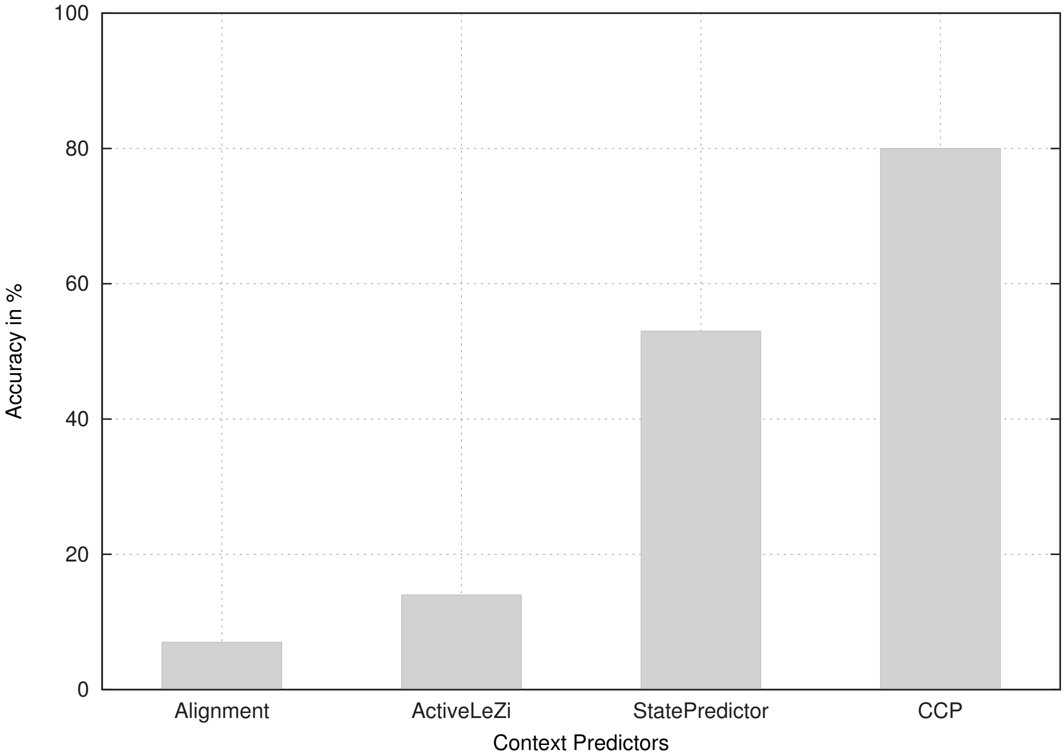

Figure 3.39: Prediction accuracies using the extracted location data of map 1 in Quake III Arena.

3.5 Conclusions

In this chapter, the Collaborative Context Predictor (CCP) method was introduced. This approach overcomes the problem of unknown or missing context information in a single user's context history. This is possible because CCP takes advantage of existing direct and

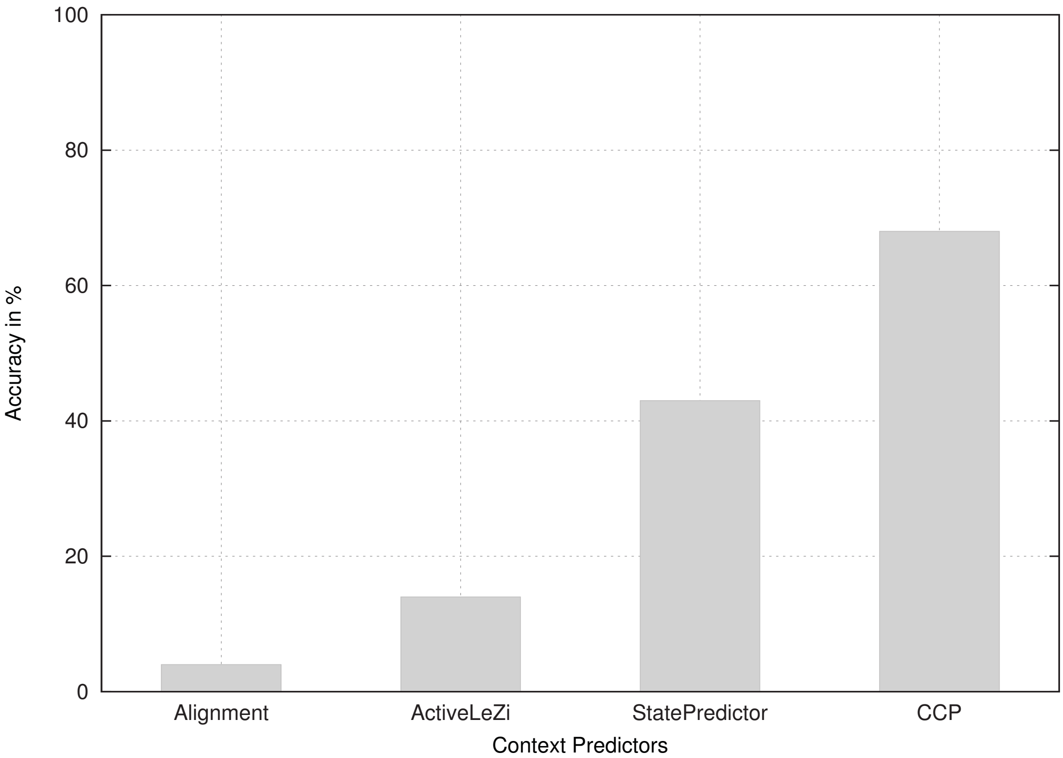

Figure 3.40: Prediction accuracies using the extracted location data of map 2 in Quake III Arena.

indirect relations between the context histories of several users in the same collaborative ubiquitous environment. Experiments on real world movement data gathered by smartphones showed that CCP is able to obtain accurate prediction results. Furthermore, CCP was evaluated using a publicly available data set, the Augsburger data set. The results showed that CCP outperforms all prediction approaches, if a sliding window approach has been applied to the data set (cf. the similarity coefficient Θ) to increase the number of existing relations between the context histories of the users. In the last experiment, a first person shooter was used to create large synthetic in-game location data. The data was retrieved by extracting the location data of six different characters controlled by users, playing the game Quake III Arena. Thus, a large and a highly collaborative data set have been created. The prediction results showed that the CCP, which obtained a next location prediction accuracy up to 80%, is the most promising approach.

References

[1] C. Voigtmann, S. L. Lau, and K. David, "An approach to collaborative context prediction," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, pp. 438–443, march 2011.

[2] P. Symeonidis, A. Nanopoulos, and Y. Manolopoulos, "Tag recommendations based on tensor dimensionality reduction," in *Proceedings of the 2008 ACM conference on Recommender systems*, pp. 43–50, 2008.

[3] C. Voigtmann, "Implementierung und Evaluation eines Tagempfehlers basierend auf der Tensorzerlegung und der HOSVD," Master's thesis, Universität Kassel, 2010.

[4] H. A. L. Kiers, "Towards a standardized notation and terminology in multiway analysis," *Journal of Chemometrics*, 2000.

[5] K. R. Gee, "Using latent semantic indexing to filter spam," in *Proceedings of the 2003 ACM symposium on Applied computing*, SAC '03, (New York, NY, USA), pp. 460–464, ACM, 2003.

[6] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, pp. 455–500, Aug. 2009.

[7] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, pp. 1253–1278, Mar. 2000.

[8] L. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.

[9] C. Voigtmann, S. L. Lau, and K. David, "A collaborative context prediction technique," in *Vehicular Technology Conference (VTC Spring) 73rd*, (Budapest, Hungary), pp. 1–5, IEEE, May 2011.

[10] W. Trumler, F. Bagci, J. Petzold, and T. Ungerer, "Smart doorplate," *Personal and Ubiquitous Computing*, pp. 221–226, 2003.

[11] S. L. Lau, I. König, K. David, B. Parandian, C. Carius-Düssel, and M. Schultz, "Supporting patient monitoring using activity recognition with a smartphone," in *The Seventh International Symposium on Wireless Communication Systems (ISWCS'10)*, (York, UK), 2010.

[12] J. Lin, E. Keogh, L. Wei, and S. Lonardi, "Experiencing sax: a novel symbolic representation of time series," *Data Mining and Knowledge Discovery*, vol. 15, pp. 107–144, October 2007.

[13] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *SIGKDD Explor. Newsl.*, vol. 11, pp. 10–18, Nov. 2009.

[14] J. Petzold, "Augsburg indoor location tracking benchmarks," Tech. Rep. 9, Institut für Informatik, D-86135 Augsburg, April 2004.

[15] J. Petzold, F. Bagci, W. Trumler, and T. Ungerer, "Next location prediction within a smart office building," in *In Proceedings of 1st International Workshop on Exploiting Context Histories in Smart Environments (ECHISE 05) at the 3rd International Conference on Pervasive Computing*, 2005.

[16] P. Lukowicz, G. Pirkl, D. Bannach, F. Wagner, A. Calatroni, K. Förster, T. Holleczek, M. Rossi, D. Roggen, G. Tröster, J. Doppler, C. Holzmann, A. Riener, A. Ferscha, and R. Chavarriaga, "Recording a complex, multi modal activity data set for context recognition," in *ARCS Workshops* (M. Beigl and F. J. Cazorla-Almeida, eds.), pp. 161–166, VDE Verlag, 2010.

[17] M. Jahn, M. Jentsch, C. Prause, F. Pramudianto, A. Al-Akkad, and R. Reiners, "The energy aware smart home," in *Future*

Information Technology (FutureTech), 2010 5th International Conference on, pp. 1–8, 2010.

[18] M. Martin and P. Nurmi, "A generic large scale simulator for ubiquitous computing," in *Mobile and Ubiquitous Systems - Workshops, 2006. 3rd Annual International Conference on*, pp. 1–3, 2006.

[19] M. Varvello, S. Ferrari, E. Biersack, and C. Diot, "Exploring second life," *Networking, IEEE/ACM Transactions on*, vol. 19, no. 1, pp. 80–91, 2011.

[20] M. Bylund and F. Espinoza, "Testing and demonstrating context-aware services with Quake III Arena," *Commun. ACM*, vol. 45, pp. 46–48, Jan. 2002.

[21] D. Salber, A. K. Dey, and G. D. Abowd, "The context toolkit: aiding the development of context-enabled applications," in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, CHI '99, (New York, NY, USA), pp. 434–441, ACM, 1999.

[22] S. Lange, "First-Person Multiplayer Spiele als ubiquitäre Umgebung für die Extraktion und Vorhersage von Bewegungskontexten," 2013. Universität Kassel, Bachelorarbeit.

Chapter 4

Legal assessment of context prediction approaches

Context data, e.g. location-based context data, have been used in different application fields such as home automation or even pedestrian safety. Often, location-based context data represents sensitive personal user data. In most cases the context data are processed on external servers, e.g. to receive a better performance or due to needed storage space. This may offer the possibility to unauthorised third parties to gain access to these data. This chapter presents legal issues that arise with the usage of context prediction. Hence, the right to informational self-determination is discussed and applied to the context prediction process. From this discussion, criteria are derived, which are used to legally assess various context prediction approaches. Finally, KORA, a method to consider legal requirements in the design process of informational technology, is outlined and used to combine the conflicting objectives of context prediction and the right to informational self-determination.

4.1 Motivation

Nowadays, various sensors pervade our daily life and affect us in different situations and areas. In the field of health care, e.g. in the matrix project, possibilities were elaborated to give patients the opportunity to be monitored even if they are outside of a hospital using ubiquitous sensors [1]. So called smart homes and smart rooms adapt their services to the lifestyle habits of occupants and the working routines of clerks by observing and learning their behaviour patterns [2, 3]. The automotive application domain represents another area, which is strongly influenced by ubiquitous sensors. A more detailed discussion of this area is given in Chapter 5. These sensors are, e.g. used for collision detection [4] and [5]. With the aid of smart badges, conference attendees can be grouped by their interests. They can be automatically informed about similar activities of other members [6]. Further, RFID sensors can be used to detect whether conference attendees are talking to each other, how long their conversation took and which talks participants have visited to provide them with additional information. This information can, e.g. be other interesting talks at the conference or other attendees with similar interests based on a user's profile [7, 8]. Another scope for the application of sensors in ubiquitous environments is the gathering and the usage of location-based context information of the users to adapt or to proactively adapt services. An overview in context prediction using location-based context information is presented in Section 2.2.2.

An important constituent of context based applications in ubiquitous environments is represented by context prediction. Context prediction is used, e.g. to predict future actions or future whereabouts of users. As a result, adaptive systems can customise themselves to future situations without requiring the users' actions. To infer future contexts, an enormous number of already gathered history of sensor data are needed. The more data, the more reliable the context prediction can be. Due to the fact that most of the sensors utilised in ubiquitous environments are not visible respectively unobtrusive, the user does not know, which of her data are actually implicitly sensed and utilised by the ubiquitous environment. In addition, these data are mostly personal and can easily be used to clearly identify a user by a third party, e.g. by using a user's different locations over time. These concerns are well known problems of context prediction and have also been mentioned but not further investigated in literature [9, 10, 11].

Among other problems, it is this gathering of history data that especially contradicts the German right to informational self-determination, whose principles have also found their way into European legislation through the Data Protection Directive[1] in recent years. Therefore, it is essential to recognise conflicts between context prediction and data protection early enough to develop proposals for solutions.

4.2 The right to informational self-determination

At first it is important to clarify that the German right to informational self-determination differs in respect of its scope to the right to privacy ([12] at page 86). Privacy is interpreted as the "right to be let alone" ([13] at page 193, as well: [12] at page 86). In contrast, the German right to informational self-determination is to protect every process where personal data are used. However, the basic idea is always the same: The data subject is to maintain control of his or her own data ([12] at page 86). This right has been developed by the German Federal Constitutional Court within a widely observed final judgement, the population

[1] Directive 95/46/EC, Official Journal L 281 , 23/11/1995, 0031 - 0050.

census decision (*Volkszählungsurteil*), in 1983.[2] In this judgement the court developed several principles. These principles have to be fulfilled to make sure that the aggrieved party persons are able to carry out their right to informational self-determination. First, users need to be aware of data processing tasks running in a ubiquitous environment. Further, the users have to have the possibility to explicitly agree or to disagree with the data processing task, which collects or processes their data. This consent, however, can only be related to a specific purpose. This also applies to cases where the data processing is approved by the legislator. Relating to the collection of data for other purposes than the specified, additional consent is needed.

To fulfill this purpose only data, which are really essential, e.g. for a certain context prediction task, are allowed to be collected and processed. Furthermore, the user must have the possibility to influence the processing of data by correction, blocking or deletion. Additionally, principles of transparency and participation rights have been adopted. Finally, the data processing has to be controlled by an independent authority. Altogether, these principles can be summarised by the avoidance of building a *profile* of a user, by providing *transparency* to the user, by providing the possibility of giving *consent* by the user, considering the *necessity* of data and by giving information about the parties that are *responsible* for the data collection process.

These principles can be considered as the key principles of data protection in Europe, since they were all implemented in the Data Protection Directive 95/46/EC ([14] at pages 63 - 108 and [12] at page 87). In the following, the problems that may arise when techniques of context prediction compete with principles and rules regarding the right to informational self-determination are discussed.

[2] BverfGE 65, 1. http://www.servat.unibe.ch/dfr/bv065001.html. (last accessed: 2013-04-06)

Profiles To the best of our knowledge, all state of the art outlined in Section 2.2.2 uses personal context data of the users, to provide reliable context predictions. Location information and current activities of the users can, e.g. be used to predict daily routines. The context information stored in a user's context history may be sufficient enough to result in a complete profile of a person, which can be easily used to identify a certain person or even to identify key elements of one's personality (e.g. political and religious or sexual orientation etc.). For this reason, predicted context may be sensitive. Informational self-determination gives the user the right to decide whether the user wants her information disclosed or not. In a situation where the concerns of individuals have only insufficient means of controlling either the veracity or the usage of gathered and inferred context data [15], it leads into the restriction of the users' decision making autonomy[3]. Thus, this autonomy will be undermined by the use of context information for context prediction, if no transparency can be guaranteed.

Transparency The principle of transparency requires the data to be collected directly from the affected person. Further, the person whose data are collected, has to be aware of this process. The goal of ubiquitous systems or ubiquitous environments, however, is to support the user by being unobtrusive, respectively invisible, to the user [16]. Therefore, the data collection and data processing task of such a system or environment is not to be considered as a deficiency for the user from the technical point of view. From a legal perspective the form of transparency used in ubiquitous computing systems is precisely the opposite. The context prediction process should be transparent in a way that the user is able to understand the collection and processing of her data at any time. Nevertheless, it would contradict the principle of transparency if the user will receive a notification during every instance of data that has been collected or processed. Nevertheless, a minimum of transparency can be ensured, if the user has access to her context data.

[3]BverfGE 65, 1

Consent Even though every use of personal data are seen as interfering with the right to informational self-determination, the violation of the right may be justified, if the user has given her consent to the use of her personal data.[4] This consent must be based on the user's own opinion and shall be given voluntarily.[5] Consequently, the question arises whether the principle of giving consent is compatible with the idea of unobtrusiveness and therefore adaptable to the process of context prediction. In this connection, it might be difficult to find solutions to the questions when consent should be given for which specified purpose of context prediction and whether it is possible to identify each purpose of context prediction at any time.

Necessity While a more extensive context history of a user may lead to a higher flexibility, operability and reliability of context prediction systems [18, 17, 19], it might interfere with the principles of data reduction. During the process of context data collection, the purpose why the context data have been collected must not be changed. Moreover, the variety of context data must be adequate to the prediction purpose the context data have been collected for. For example if a user's indoor locations have to be predicted, the additional collection of outdoor location data would disregard the principle of necessity.

Responsibility Context prediction can be a complex task. It can consist of different aggregated steps, which are responsible for the collection of context data, the preprocessing of the data, the transformation of the data from low-level contexts to high-level contexts and finally the performance of the actual prediction of a user's context (cf. Section 2.3). Moreover, this task can be provided by several external service providers. Accordingly, it is difficult to determine which component or which provider has transmitted the data and which communication path the data have been taken.

[4] Article 7 (a) Directive 95/46/EC.

[5] Article 2 (h) Directive 95/46/EC, see also: [17], at page 115.

Thus, the use of some described systems ([20] and [21]) can lead to a disregard of responsibility ([22] at page 74). Altogether, as soon as context data have been collected, transmitted or processed externally by a system that is unknown to the user, the compliance with the principle of responsibility may be difficult.

As discussed above, the process of context prediction, which mostly utilises personal context information without the knowledge of the respective user, contradicts the right to informational self-determination and its principles. Based on the outlined principles, legal evaluation criteria are derived and discussed next. These criteria are applied to assess the algorithms presented in Section 2.2 and in Section 3.3. A solution how context prediction approaches can be implemented and utilised while taking the right to informational self-determination and its principles into account, is outlined in Chapter 6.

4.3 Legal evaluation criteria

The technique, which is unable to cause privacy problems, is the most effective technique to ensure data protection. The need to enforce law would be reduced.[6] To enhance privacy by design, privacy protection rules should be used to create legal requirements. Context prediction algorithms must ensure compliance with data protection law, respectively with the user's right to informational self-determination. Normally, for this purpose, the separate laws of each country have to be considered. To enlarge the scope to a European range, mainly the data-protection rules of the European Union will be contemplated in the following. These data-protection rules are expressed by several data-protection principles. They were first developed by the German Constitutional Court in the final census decision.[7]. They can be considered as the key principles

[6]Roßnagel, in: Roßnagel Handbuch Datenschutzrecht, München 2003, Kap. 3.4, Rn. 47.

[7]Federal Constitutional Court of Germany (Bundesverfassungsgericht)

of data protection in Europe, since they were all implemented in the Data Protection Directive 95/46/EC [12]. In general, most algorithms used for context prediction tasks process personal data. The predictions of contexts are related to an identified or identifiable natural person. Consequently, context can be described as personal data as defined in Art. 2 (a) of the Data Protection Directive 95/46/EC. In the following, the evaluation criteria are outlined that can be used to assess in how far existing context prediction algorithms fulfil the data-protection rules of the European Union respectively the user's right to informational self-determination.

To avoid the creation of user profiles as one principle of the right to informational self-determination, an anonymous or pseudonymous processing of data has to be considered. The anonymisation of personal data requires that it is impossible to establish a relation between the context data and the affected person. Information, respectively context data that cannot be linked to a person by legal definition cannot violate personal privacy. Despite this desirable goal it is unlikely that anonymisation could be implemented in practice. The purpose of context prediction is to support the user on the one hand or to automatically adapt services with regard to given context information on the other hand. Anonymous processing of the context would hinder the support or prevent adaption by the application. This does not mean that the prediction output could not be used in a pseudonymous way. Pseudonymisation refers to replacing the identifiers with pseudonyms known only by the processor. The collected context-information itself can be pseudonymised. This pseudonymisation should prevent third parties from reconstructing the behaviour of an identified or identifiable natural person.

Unfortunately, there is no anonymisation or pseudonymisation that would satisfy legal requirements. Nevertheless, this method would hinder conclusions regarding the user's behavior and therefore enhances privacy. In the following, this type of processing will be called context-data pseudonymisation (*cdp*). Further, it is in the user's interest that as little personal data as possible are collected to fulfil the obligation in Art. 6 (e) Data Protection Directive 95/46/EC. The obligation signifies that personal data must be kept in a form that ensures the identification of data subjects for no longer than for the purposes the data have been collected for. Moreover, this guarantees the principle of necessity and the earmarking of the collected data to a specific purpose, published in Art. 6 (b) Data Protection Directive 95/46/EC. In addition, collected data that have no effects on the context prediction have to be erased. The principle of necessity is evaluated by examining whether the applied context prediction approaches are able to support indexing to delete unnecessary context data (*indexing*), the context prediction approaches work in a collaborative manner (*collaborative*) and whether a context prediction approach needs a high number of context data (*necessity*) to make reliable predictions.

The reduction of the data volume enables *data processing* on a user's client, e.g. on a user's own smartphone. This would comply to the principle of transparency, because the history data of a user are stored on her own device. Further, it would comply to the principles of consent and responsibility. This is because, context data that are directly processed and predicted on a user's own device do not evoke any concerns according to these principles. In contrast, the process of data on the server side would raise several data protection issues because the external processing of data may offer the possibility to unauthorised third parties to gain access to these data.

The identified evaluation criteria is used in the next section to indicate if existing context prediction approaches meet the proposed principles to the right to informational self-determination.

4.4 Evaluation

In the following, the derived evaluation criteria (*cdp*, *necessity*, *collaborative*, *data processing*, *indexing*) as outlined in the previous section, will be applied to the algorithms presented in the Sections 2.2 and 3.3. Further, the results are evaluated and discussed with regard to the possible consequences for the used prediction approaches.

Table 4.1 shows the different prediction approaches and the criteria used to assess these approaches from a legal perspective. If a prediction approach satisfies a legal criterion it gains one point. If an approach partially satisfies a criterion it gains half a point. The maximum score that can be obtained by the examined algorithms is five points. The more points a prediction approach receives the more it satisfies the criteria outlined in Section 4.3. A prediction approach receives a point if it is able to handle pseudonymised context data, if its necessity of data is low and if it utilises only the context data of the person whose next context has to be predicted. Further, an approach receives a point if it can be directly used on a person's smartphone and if it supports indexing to be able to automatically delete context data that is not frequently used. The points received in total by the prediction approaches are shown by Σ.

Table 4.1: Legal assessment of different context prediction approaches.

	cdp	*necessity*	*collaborativeness*	*data processing*	*indexing*	Σ
AL	yes	mid	non	runnable	yes	4.5
ALZ	yes	high	non	not runnable	yes	3.0
SP	yes	high	non	runnable	yes	4.0
CCP	yes	high	colab.	runnable	yes	3.0
Tree	yes	low	non	runnable	yes	5.0
BN	yes	low	non	runnable	yes	5.0

Context-data pseudonymisation Context-data pseudonymisation means that arbitrary placeholders are used to replace the contexts stored in a user's context history. The name of the user the history belongs to will not be replaced. How this criterion effects the accuracy of the prediction approaches is exemplarily elaborated using the freely available Augsburger data set[8]. The prediction approaches have been applied to the original data and the context

[8] http://www.pervasive.jku.at/Research/Context_Database/index.php

histories whose contexts have been replaced by pseudonyms. The number and order of context information remained unchanged. The results have shown that the prediction results for all approaches have remained the same of the given data set. For this reason, each algorithm satisfies this criterion. The Augsburger data set, however, consists of nominal data, which represents a person's current location. The replacement of context data through pseudonyms is not possible if the data are stored ordinally or numerically. In case that context data are, e.g. represented in GPS coordinates, the data could be pseudonymised by adding a global threshold to disguise the context data. Still, transformed contexts may have side affects, which can, e.g. affect the runtime behaviour of the prediction approach but this has not been further investigated in this thesis.

Necessity of data It can be assumed that the higher the number of context data that can be utilised to train a context prediction approach, the more accurate the approach can potentially be. Nevertheless, there are approaches that need a larger amount of training data to be able to make reliable predictions and there are approaches that can work with a smaller amount of training data. With regard to the legal assessment of the prediction approaches those, which can achieve higher prediction accuracy on a smaller size of training data, perform better. In order to find the approach, which performs best using only a small amount of data to train its prediction model, the Augsburger data set has been used again. The different sub data sets of the Augsburger data set have been merged into one data set. The resulting data set has been split into a test data set (10%) and a train data set (90%). Following, three small training data sets with a size of 10%, 20%, and 30% were randomly drawn from the 90% training set. This selection was performed five times for all data set sizes in order to obtain a mean and variance of the results. Figure 4.1 presents the averaged results, obtained by the different algorithms using the different training data sets.

The results obtained by the Tree-based classifier (C4.5) and by

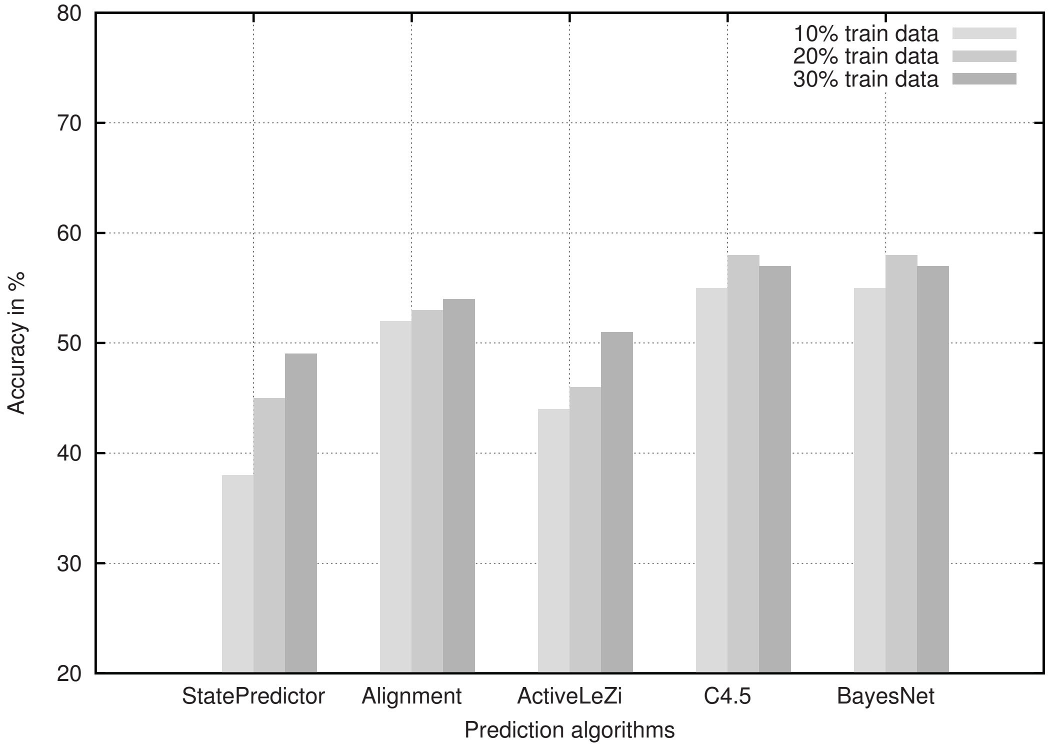

Figure 4.1: Testing the necessity of data of the different prediction approaches using small training set to classify a given test data set [23].

the classifier based on a Bayesian Network are quite similar. Both achieved prediction accuracy up to 58%. Alignment achieved a prediction accuracy up to 53% and ActiveLeZi up to 51%. The most inaccurate classifier is the StatePredictor whose accuracy rate was lower than 50%. CCP has not been evaluated. Due to its collaborative character it is only suitable for using multiple context histories, which automatically requires a high necessity of personal context data. But, if the needed "personal" context data are appropriately pseudonymised the usage of CCP may be lawful.

Collaborativeness Basically, all presented prediction algorithms can be used in a collaborative manner by simply concatenating the context histories of different users or by using the histories of the users separately and by combining the results by a voting approach. The Collaborative Context Prediction (CCP) approach, however, is the only approach that specifically requires context histories of additional users. Consequently, the approach explicitly relies on personal context information of other persons. Accordingly, the CCP approach failed this criterion.

Data processing From the perspective of law it would be best to process context data directly on the user's smartphone in order to ensure that the user's data keep private. This would also meet the principles of transparency, consent and responsibility as mentioned above. To elaborate whether the presented prediction approaches are suited for the direct execution on a smartphone, a benchmark with respect to the following aspects has to be performed: measuring the time needed to train the prediction model; measuring the time needed to perform a single prediction and measuring the time needed for all prediction for a given data set.

It has to be considered that the performance of algorithms depends on their implementation and on the size of the data set that is used for the evaluation. The higher the number of attributes (contexts) and the higher the number of characteristics an attribute can have, the more complex will be the creation of a reliable prediction model. The applied Augsburger data set consists of 2120 training instances and 200 test instances. Each instance consists of four contexts whereas each context can have 16 different characteristics.

Table 4.2 shows the times of the several approaches needed to be trained on a smartphone. It shows the overall prediction time needed to make a prediction for all 200 test instances and it outlines the average prediction time per instance. For a better comparison, the values received on the smartphone are opposed with the values received using a PC. As smartphone, a Samsung Galaxy S III and as PC, an Intel Core i7 with 2 GHz and 8 GB RAM has been used.

It is obvious that all prediction times received on the Samsung Glaxy S III are higher than the prediction times on the PC. With regard to the training time needed to construct the respective prediction models, all approaches received suitable results except

Table 4.2: Training and prediction times of the approaches using the Augsburger data set.

Algorithms	**Smartphone**			**PC**		
	training	prediction	prediction p.i.	training	prediction	prediction p.i.
Alignment	92 ms.	47.12 sec.	0.24 sec.	69 ms.	424 ms.	2.12 ms.
ActiveLeZi	320.4 sec.	241.7 sec.	1.21 sec.	20.51 sec.	5.28 sec.	26.4 ms.
StatePredictor	0.24 sec.	6 ms.	0.03 ms.	97 ms.	6 ms.	0.03 ms.
CCP	3.9 sec.	696.75 sec.	3.48 sec.	332 ms.	19.1 sec.	95.17 ms.
Tree-based (J48)	364 ms.	22 ms.	0.11 ms.	74 ms.	15ms.	0.02 ms.
Bayesian Net	194 ms.	40 ms.	0.2 ms.	53 ms.	7 ms.	0.035 ms.

for the ActiveLeZi approach. It needs 320.4 sec. to be trained on the smartphone. For this reason, the used implementation of the approach is not applicable on the test smartphone. With regard to the overall prediction time and to the needed prediction time per instance, the CCP approach took longest. If no real time context prediction is needed, CCP is still usable on a smartphone with its average prediction per instance of 3.48 seconds. The best results were achieved by the Weka implementations of the J48 and the Bayesian Net classifier and by the StatePredictor approach.

An approach, which outlines the applicability of different context prediction approaches for distributed and collaborative context prediction using P2P communication that is directly executed on a person's smartphone and therefore considers legal perspectives has been presented in [24].

Indexing In order to store as little personal context data of a person as possible, the indexing of context data is considered. The idea is to mark context data that is frequently used by prediction approaches to predict a next context as important. In contrast, context data that is not often used is marked as less important. The easiest way to fulfil the principle is to implement a counter that remembers how frequently a certain context is used during the prediction process. Using the example of a decision tree, the frequency of traversing a certain node or a sub-tree in order to achieve a prediction can be counted. With regard to the examined algorithms, it is possible to implement an additional code in all algorithms, using a technique called "hooking", to enhance the predictors by providing counter functionality to additional indexing context information. Subsequently, the indexed context information can be used to weaken certain contexts during the prediction process or to pre-filter less important context information before the prediction process. Table 4.1 indicates that all algorithms can support a functionality to index context information.

The evaluation of the different legal aspects shows that

Alignment, the Tree-based approach and the approach based on Bayesian Networks receive the highest scores (cf. Table 4.1). That implies that these prediction approaches mostly satisfy the legal aspects demanded in Section 4.3. CCP and the ActiveLeZi approach disregard the identified legal criteria the most. Reasons are the collaborative character and the high necessity of data of the CCP approach and the slow learning time of ActiveLeZi on modern smartphones.

4.5 Inferring a legally compatible context prediction process using KORA

In this chapter legal problems of context prediction techniques with respect to the right to informational self-determination were outlined and discussed. Further, an evaluation was presented, which determines how well existing context prediction approaches fulfil certain legal evaluation criteria that were derived from the principles of a user's right to informational self-determination. In this section, KORA (Konkretisierung rechtlicher Anforderungen) [25] an approach to integrate and consider legal requirements in the design process of informational technology is outlined. It will be exemplarily demonstrated how KORA can be used to help to design a context prediction process to be more compatible with the right to informational self-determination by outlining technical design proposals.

If informational technology has to be developed in a way that it is legally acceptable it becomes obvious that legal norms mostly do not include concrete guidelines how a piece of informational technology, e.g. the process of context prediction, can be developed to be consistent with a certain law or directive. The reason for this is that legal requirements for example the right to informational self-determination, which is presented in Section 4.2 are mostly formulated in a very generic way. KORA tries to close this lack in description by gradually inferring technical design requirements from the underlying abstract legal requirements.

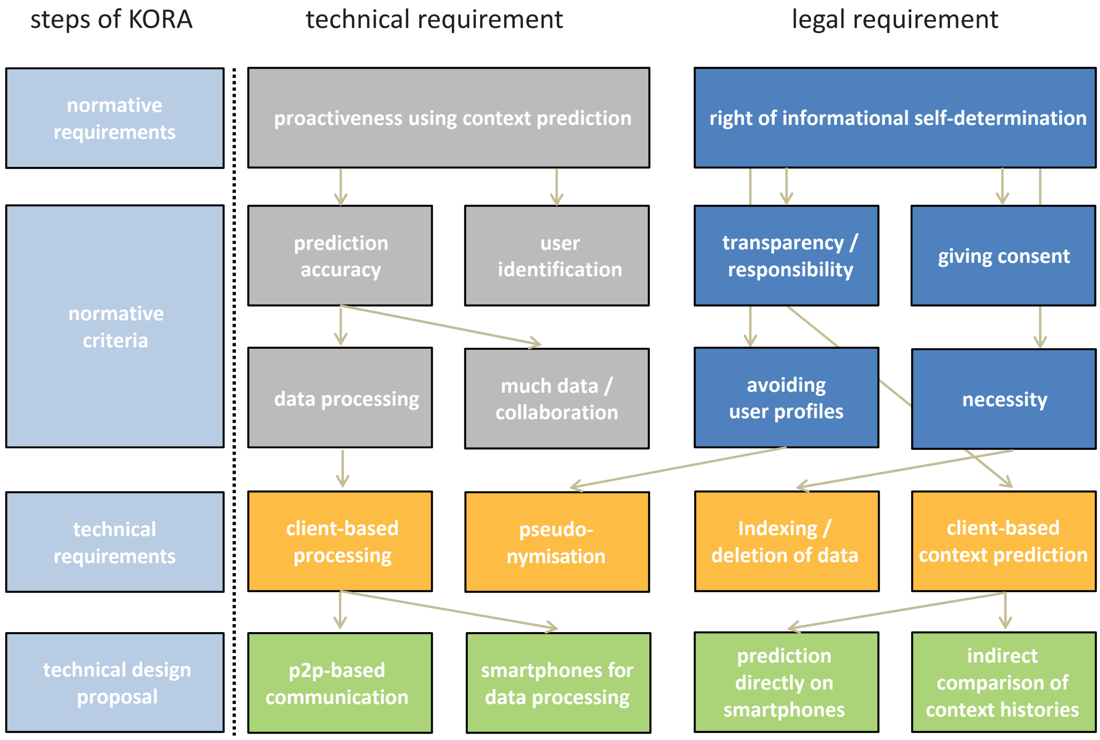

Figure 4.2: Supposing a technical solution for a socially acceptable context prediction process using KORA.

KORA can be used to infer technical design proposals from legal requirements in four different steps as outlined in [26]. Also KORA can be used to derive technical design proposals from technical non-functional requirements. In this example, a technical design proposal is derived from one technical non-functional requirement and from one legal requirement at the same time using the KORA approach. In doing so, the underlying legal requirement and the underlying non-functional technical requirement are becoming step by step more and more technically realisable. Hence, it can be illustrated how a technical and a legal requirement lead to quite contrary criteria, which can be agreed by inferring suitable technical requirements afterwards. In the following, KORA is applied to the technical requirement "provide proactiveness by utilising context prediction" with respect to the consideration of the legal requirement "informational self-determination". As a result, a technical design proposal is received that enables a context prediction process to be more legally acceptable by design. Figure 4.2 shows the four different steps of KORA regarding the technical requirement and the legal requirement. Next, the four different steps of KORA are outlined.

Normative requirements The normative requirement from a technical point of view is the provision of proactiveness, e.g. by proactively enabling services or applications in a ubiquitous environment, required to adapt to a user's needs using context prediction. From the legal perspective the normative requirement that is considered is the user's right to informational self-determination. If there are no legal requirements that can be explicitly considered with regard to the respective technical requirement, the requirements of the respective national constitution will automatically be taken into consideration.

Normative criteria Normative criteria that can be derived from the process of predicting users' contexts are a high prediction accuracy to provide reliable predicts and the possibility to clearly identify the user whose context data are used to make a prediction. Otherwise, the predicted contexts cannot be assigned to the user. Additional criteria are collaboration to provide suitable predictions, even if the user's own context history does not provide sufficient context information, and the normative criteria that as many data as possible can be utilised. It is obvious that the more context data can be provided to train a prediction model the more reliable the prediction results may be. Finally, to obtain prediction results, already gathered sensor data have to be transmitted and preprocessed on a suitable computer system. The prediction process itself has to run on a suitable computer system too.

Normative criteria that can be derived from the right to informational self-determination are the principles providing transparency, providing the possibility of giving consent, providing information about the responsibility of components of a certain step in the prediction process. Further criteria are to avoid the building of user profiles and to focus only on data that are needed for a certain prediction process. All principles have been outlined in more detail in Section 4.2.

By comparing the normative criteria derived from the technical and the legal requirement it becomes obvious that the different criteria pursue contrary goals. On the technical side a high prediction accuracy while using collaborative techniques and as much context data as possible are important and on the legal side, the avoidance of user profiles and the prevention of uncontrolled data collection are dominant aspects. In addition, to the best of our knowledge, the clear identification of the user the context data belongs to without her knowledge is required by most applications and services that apply context prediction techniques. This contradicts the criteria of enabling the user of giving consent, if her personal data are processed. Further, to work efficiently, data processing tasks as the pre-processing of sensor and low-level context data as well as the prediction process are performed on a server structure in most cases. Therefore, the user has no further control of her personal context data. In contrast, there are the legal criteria of transparency and responsibility.

Technical requirements Technical requirements represent abstract design goals, which are used to fulfil the derived normative criteria. It has to be considered that technical requirements do not represent concrete technical implementation requirements.

To solve the derived contrary goals of the technical and legal requirement, client-based processing of the context data, the pseudonymisation of context data, the deletion of not relevant context data and the avoidance of transmitting personal context data to third parties is proposed. Using client-based processing and client-based context prediction, it can be ensured that the user does not loose control of her data and for this reason no service provider has to ensure transparency or to provide the possibility of giving consent which may additionally weaken the idea of ubiquitous computing. With the pseudonymisation of the context data, it can be ensured that even if the data are transmitted to third parties, they are not able to build a profile of the user. Adding of the support of indexing enables the possibility to delete context data that is not required for the context prediction process.

With the consideration of these technical requirements it is possible to fulfil both the normative criteria of the technical and the legal requirement. The proof of concept that existing context prediction approaches can be developed according to these requirements has been given in Section 4.4.

Technical design proposal To ensure a technical solution that is socially acceptable, the derived technical requirements have to be specified into a technical design proposal in the last step of the KORA approach. To ensure that the personal context data, derived from various sensors that surround the user in a ubiquitous environment, can be client-based processed, the user's own smartphone is used to process the data. The same refers to ensure client-based context prediction.

In case the collaborative-based context prediction is used as outlined in Section 3.3, a P2P-based communication between the smartphones of the users is proposed to prevent a server-based structure that centralised stores the context data. If no collaborative context prediction is used, no data of other users has to be transferred. If context data have to be transferred to other users, they have to be pseudonymised first. In the case of collaborative-based context prediction, similar users have to be identified. Therefore, context histories of users are not compared directly to each other but similar context histories are identified by only comparing anonymised characteristic values, whereby a characteristic value represents a user's context history.

A concrete implementation of a context prediction process based on this technical design proposal to be more legally reliable is presented in Chapter 6 in detail.

4.6 Conclusions

In this chapter the collaborative-based context predictor and other well-known context prediction techniques have been evaluated with regard to their compatibility to the user's right to informational self-determination, respectively to the Data Protection Directive. Therefore, problems have been presented that arise when context prediction competes with principles and rules regarding the right to informational self-determination. Subsequently, legal evaluation criteria have been derived, considering the principles to informational self-determination. Based on these legal criteria, the context prediction approaches were evaluated. The evaluation shows that the Alignment predictor and predictors based on a Decision Tree and on a Bayesian Network most likely fulfil the required criteria. On account of its collaborative character, the CCP often disregards the identified legal criteria. Furthermore, KORA, a method to consider legal requirements in the design process of informational technology, was outlined. Finally, KORA has been applied to the technical requirement "provide proactiveness by utilising context prediction" with respect to the consideration of the legal requirement "informational self-determination" to infer a concrete technical design proposal that enables a context prediction process to be more legally compatible and therefore acceptable by design.

References

[1] S. L. Lau, I. König, K. David, B. Parandian, C. Carius-Düssel, and M. Schultz, "Supporting patient monitoring using activity recognition with a smartphone," in *The Seventh International Symposium on Wireless Communication Systems (ISWCS'10)*, (York, UK), pp. 810–814, 2010.

[2] R. Andrich, V. Gower, A. Caracciolo, G. D. Zanna, and M. D. Rienzo, "The dat project: A smart home environment for people with disabilities.," in *ICCHP* (K. Miesenberger,

J. Klaus, W. L. Zagler, and A. I. Karshmer, eds.), vol. 4061 of *Lecture Notes in Computer Science*, Springer, 2006.

[3] M. Danninger and R. Stiefelhagen, "A context-aware virtual secretary in a smart office environment," in *MM '08: Proceeding of the 16th ACM international conference on Multimedia*, (New York, NY, USA), pp. 529–538, ACM, 2008.

[4] T. Gandhi and M. M. Trivedi, "Pedestrian protection systems: Issues, survey, and challenges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 3, pp. 413–430, 2007.

[5] K. David and A. Flach, "Car-2-x and pedestrian safety," *Vehicular Technology Magazine, IEEE*, vol. 5, no. 1, pp. 70–76, 2010.

[6] J. A. Paradiso, J. Gips, M. Laibowitz, S. Sadi, D. Merrill, R. Aylward, P. Maes, and A. Pentland, "Identifying and facilitating social interaction with a wearable wireless sensor network," *Personal and Ubiquitous Computing*, vol. 14, pp. 137–152, February 2010.

[7] M. Atzmüller, D. Benz, S. Doerfel, A. Hotho, R. Jäschke, B.-E. Macek, C. S. Mitzlaff, Folke, and G. Stumme, "Enhancing social interactions at conferences," *it - Information Technology*, pp. 101–107, 2011.

[8] M. Atzmueller, S. Doerfel, A. Hotho, F. Mitzlaff, and G. Stumme, "Face-to-face contacts at a conference: Dynamics of communities and roles," in *Modeling and Mining Ubiquitous Social Media*, LNAI, pp. 21–39, Heidelberg, Germany: Springer Verlag, 2012.

[9] M. Satyanarayanan, "Pervasive computing: Vision and challenges," *IEEE Personal Communications*, vol. 8, pp. 10–17, 2001.

[10] R. Mayrhofer, "Context prediction based on context histories: Expected benefits, issues and current state-of-the-art," *Cogni-*

tive Science Research Paper-University of Sussex csrp, vol. 577, p. 31, 2005.

[11] P. Nurmi, M. Martin, and J. A. Flanagan, "Enabling proactiveness through context prediction," in *Proceedings of the Workshop on Context Awareness for Proactive Systems, Helsinki*, vol. 53, 2005.

[12] G. Hornung and C. Schnabel, "Data protection in germany: The population census decision and the right to informational self-determination," *Computer Law & Security Report*, vol. 25, no. 1, pp. 84–88, 2009.

[13] B. o. Warren, "The right to privacy," *Harvard Law Review*, vol. 4, no. 5, pp. 193–220, 1890.

[14] C. Kuner, *European Data Protection Law, Corporate Compliance and Regulation.* Oxford University Press, 2007.

[15] G. M. Rehm, "Just judicial activism? privacy and informational self-determination in U.S. and German constitutional law," 2000.

[16] M. Weiser, "The computer for the 21st century," *Scientific American*, vol. 265, pp. 66–75, Sept. 1991.

[17] A. Roßnagel, *Datenschutz in einem informatisierten Alltag.* Friedrich-Ebert-Stiftung, 2007.

[18] M. Langheinrich, V. Coroama, J. Bohn, and F. Mattern, "Living in a smart environment – implications for the coming ubiquitous information society," *Telecommunications Review*, vol. 15, p. 132–143, feb 2005.

[19] H. Skistims, C. Voigtmann, K. David, and A. Roßnagel, "Datenschutzgerechte Gestaltung von kontextvorhersagenden Algorithmen," *Datenschutz und Datensicherheit - DuD*, vol. 36, no. 1, pp. 31–36, 2012.

[20] K. L. Mika, M. Raento, and H. Toivonen, "Adaptive on-device location recognition," in *In Proceedings of the Second International Conference on Pervasive Computing*, pp. 287–304, Springer Verlag, 2004.

[21] K. Laasonen, "Clustering and prediction of mobile user routes from cellular data," in *in PKDD. 2005*, pp. 569–576, Springer Verlag, 2005.

[22] A. Roßnagel, "Modernisierung des Datenschutzes für eine Welt allgegenwärtiger Datenverarbeitung," *Multimedia und Recht*, no. 2, pp. 71 – 75, 2005.

[23] C. Voigtmann, K. David, H. Skistims, and A. Roßnagel, "Legal assessment of context prediction techniques," in *Vehicular Technology Conference (VTC Fall)*, (Quebec City, QC), pp. 1–5, IEEE, 2012.

[24] C. Voigtmann, C. Schütte, A. Wacker, and K. David, "A new approach for distributed and collaborative context prediction," in *10th IEEE Workshop on Context Modeling and Reasoning 2013 (CoMoRea 2013)*, (San Diego, USA), pp. 20–24, IEEE, Mar. 2013.

[25] A. Roßnagel and V. Hammer, "Kora. eine Methode zur Konkretisierung rechtlicher Anforderungen zu technischen Gestaltungsvorschlägen für Informations- und Kommunikationssysteme," *Infotech*, vol. 1, pp. 21–24, 1993.

[26] A. Roßnagel, *Rechtswissenschaftliche Gestaltung der Informationstechnik*, ch. Rechtswissenschaftliche Gestaltung der Informationstechnik, pp. 381–390. Wissen, Vernetzung, Virtualisierung, Festschrift zum 65. Geburtstag von Udo Winand, Köln 2008, 2008.

Chapter 5

Pedestrian safety as a use case for CCP

Unfortunately, traffic accidents involving pedestrians or cyclists cause thousands of road casualties and serious injuries worldwide every year. Therefore, improving the safety of vulnerable road users is an international demand. Currently, passive as well as active collision avoidance systems have already been installed in cars to protect pedestrians. To design an "ideal" protection system, several challenges have to be tackled. In this chapter, the CCP approach is used to proactively filter pedestrians whose next step brings them close to the street to provide an extra time advantage for collision avoidance systems. In this use case, CCP takes advantage of the movement patterns extracted from contexts such as acceleration or orientation in 3D of the pedestrians received by sensors installed in their smartphones. To evaluate the CCP approach, comparisons with the state of the art context prediction approaches, already outlined in the previous chapters, are performed.

5.1 Motivation

Road traffic crashes and injuries are a serious public health problem. Every year, more than one million people are killed as a result of traffic accidents [1] worldwide. 400,000 of them are pedestrians [2]. The reasons for accidents between pedestrians and cars are various. In some cases, car drivers or pedestrians are simply inattentive. In other cases complex spots such as curves or parking spots prevent direct visual contact between the road users and might lead to dangerous situations.

In order to reduce accidents between cars and pedestrians, several research groups use different technologies to develop passive and active pedestrian protection systems. The target of passive pedestrian protection is the reduction of the impact on a pedestrian when the accident is no longer avoidable. This is achieved by mechanisms like rising hoods or pedestrian airbags that are under investigation to prevent the pedestrian from hitting the engine block respectively the windshield [3]. First passive systems are also provided in products of car manufactures like BMW, Audi or Honda.

Passive pedestrian protection is a first step to improve pedestrian safety, however, the better solution is to actively avoid a collision. Current approaches use vehicle-based sensors such as infrared, radar or laser to detect pedestrians that might collide with a vehicle. In [4], an automotive night vision system for pedestrian detection based on infrared sensors is illustrated. The authors present a pre-processing technique based on a Support Vector Machine classifier that filters pedestrians out of a given picture. A comprehensive overview of different methods using laser, video and infrared sensors to avoid pedestrian-vehicle collisions is given in [5]. First active systems, enabling a car "to see what is on the road", have already been introduced in products of e.g. Mercedes and Toyota to detect pedestrians. All these approaches are promising, but they need a direct line of sight and do not use any contextual information of the pedestrians they are trying to protect.

The main challenges to provide an optimum solution to prevent car pedestrian accidents are the following [6]: Providing an overview about the scenario (car, the pedestrian, street, other pedestrians, etc.); filtering pedestrians at risk (out of potentially many); and finally a way to communicate this information and providing a mechanism to warn the relevant road users.

First approaches that try to prevent possible pedestrian-vehicle accidents using car systems and information from pedestrians are outlined in [7] and [8]. The proposed systems use a pedestrian's mobile phone and a car navigation system. GPS coordinates of the pedestrian and the car are sent to a server. Then, the collision risk is calculated and the driver will be alerted of the likelihood of an accident. Another method uses radio frequency tags to avert collisions between pedestrians and cars. [9] and [10] describe strategies based on RF-communication between a long power transponder that is attached to a pedestrian and a receiver placed in a vehicle. Thus, pedestrians can be detected for a distance up to 60m by the car without a direct line of sight and without transferring additional information of the pedestrian to the car. In [11] the pedestrian's spatial location, velocity and heading angle are used to predict her long term movement behaviour. This information is used by the vehicle to find the optimal path to prevent a possible vehicle-pedestrian accident.

The above mentioned approaches show possibilities of detecting pedestrians that are walking on the pavement or even pedestrians obscured by objects, but they do not filter the pedestrians that may be at risk in advance. For this reason, these approaches may be inefficient in terms of needed calculation time and battery consumption.

A technique to filter pedestrians that may be at risk from those who are not is proposed in [13] and [6]. The authors present various filters differing by input information like movement speed, movement directions and possible intersection points between cars and pedestrians. Figure 5.1 presents possible architectures to filter pedestrians. The first architecture uses ad hoc communication (3) between the pedestrian's mobile phone (2) and the navigation system of a car (1), the second architecture uses cellular communication

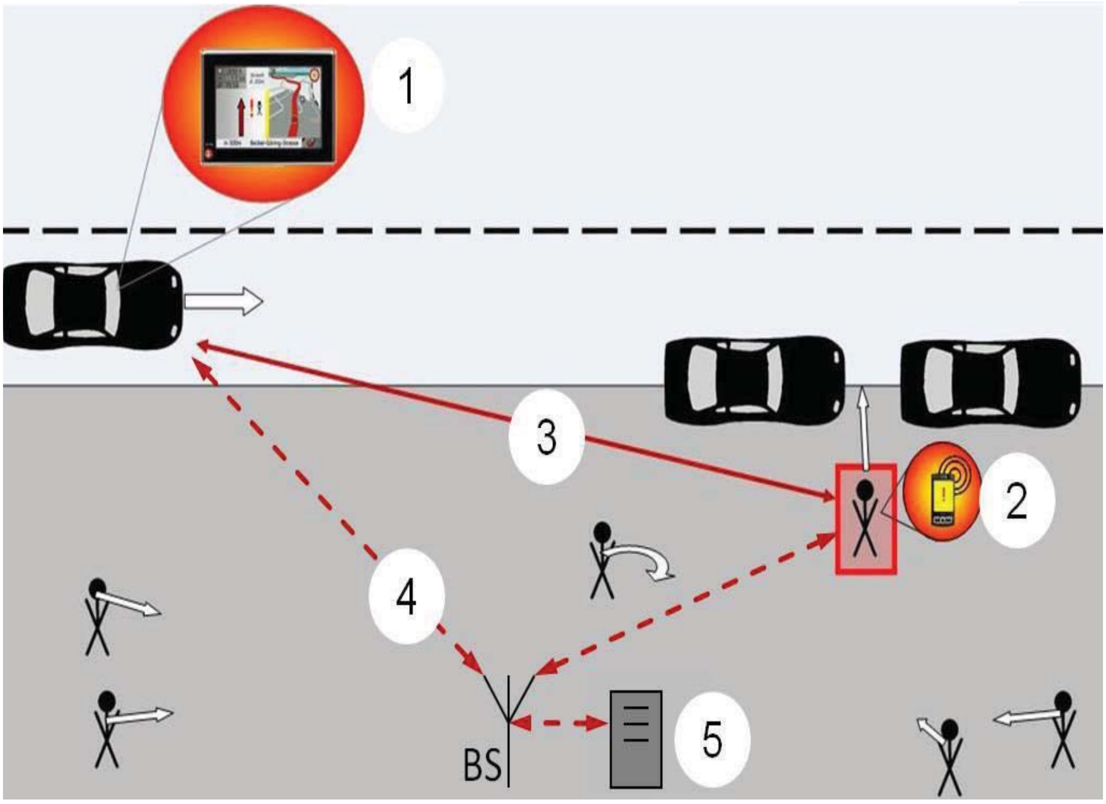

Figure 5.1: Filtering pedestrians at risk [12].

(4) between the pedestrian's mobile phone, a central server unit (5) and a car navigation system.

In this chapter CCP is applied to a realistic and collaborative use case, the protection of pedestrians, to demonstrate its usefulness. Therefore, the following contributions, which have been published in [12], are presented:

(i) Extending the idea of filtering pedestrians by predicting a pedestrian's next step using her context information (movement and orientation). Hence, it is possible to proactively filter pedestrians and provide a collision avoidance system with an additional time advantage.

(ii) Using simulated and real movement data. The real movement data were measured by a Samsung Galaxy S II smartphone the pedestrians carried in their left trouser pocket to gather realistic input data.

(iii) Finally, the results of the CCP approach, predicting the pedestrian's next step are compared to the algorithms introduced in Section 2.2.

5.2 Use case description

The scenario, which describes the presented use case consists of a pavement beside a street that has been segmented into several parts (cf. Figure 5.2). The different parts are used to locate a pedestrian's current position on the pavement. The size of one part of the pavement is 0.75cm × 0.75cm and results from the average length of a step of pedestrians. Hence, with each step, a pedestrian reaches a new part on the pavement. This segmentation has been used because current GPS technologies do not offer sufficient accuracies that are needed to precisely locate the current position of a pedestrian on a pavement. Current standard implementations of GPS devices only achieve accuracies of 3 to 5 meters [14]. In our experiment the average speed of a pedestrian is determined as $1.34\frac{m}{s}$ [15]. Hence, a pedestrian needs approximately 0.56 seconds to move from one part to another.

In order to describe the pedestrian's current position on the pavement the different parts have been labelled horizontally with numbers and vertically with letters. The pavement is divided into two areas. One area close to the street $A0$ till $A14$ is marked in red and indicates that a pedestrian might be at risk. The other area, marked in black, has more distance from the street and indicates that pedestrians inside this area are currently not at risk.

In Figure 5.2 paths in different colours can be seen. The position where the pedestrian enters the pavement is known a priori and is not in the focus of the examination outlined in this thesis. A technology for relative positioning of a pedestrian on a pavement that can be used to detect her entrance point on the pavement is, e.g. presented in the AMULETT project [9].

In this scenario according to [16], it is assumed that the way taken by the pedestrians is always the path that requires the smallest

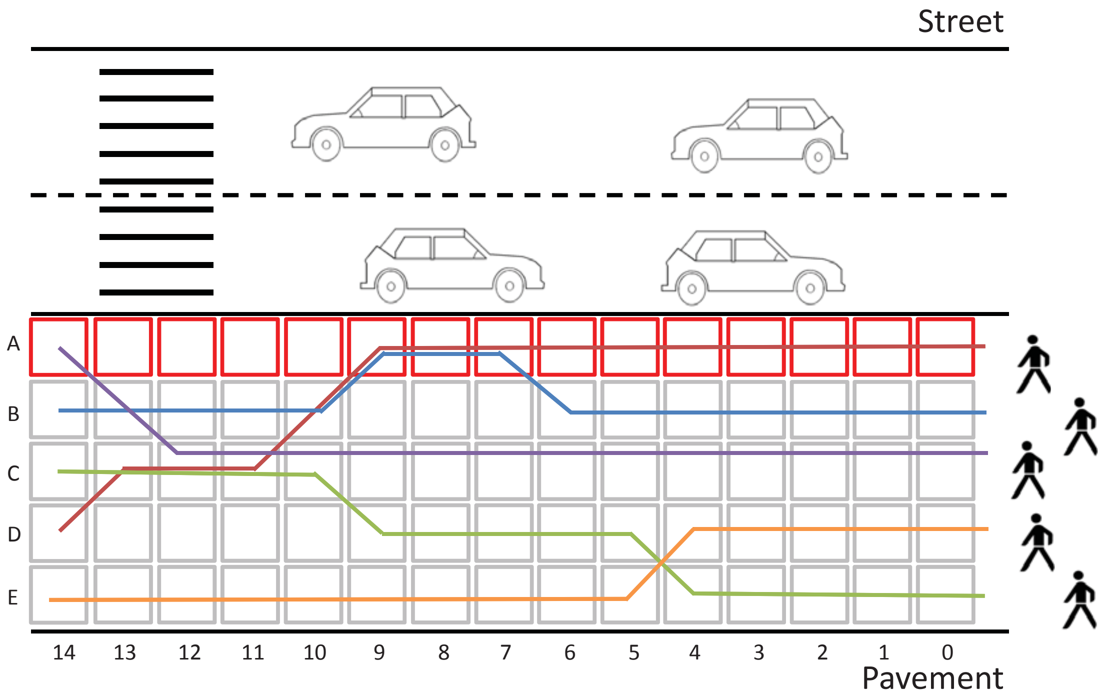

Figure 5.2: The underlying scenario consists of a pavement aside a street divided in several labelled parts. Dangerous spots are marked in red [12].

number of steps to reach a destination. Hence, the assumption is made that pedestrians only change their movement direction after three steps. The parts that can be reached by a pedestrian with the next step is either the closest part in front of the pedestrian or the closest part diagonally to the left or to the right of the pedestrian's current position. Each path as, e.g. outlined in Figure 5.2 belongs to a pedestrian and indicates her recorded movement sequence on the pavement using a smartphone. A set of different runs of a pedestrian represents her movement history, respectively her context history. Hence, the context history consists of the different parts that form the paths walked by the pedestrian. The context histories are used by the prediction algorithms to predict the pedestrian's next step. As can be seen in Figure 5.2 the movement behaviours of the pedestrians show similarities. These similarities between the movement patterns in the context histories of the pedestrians are utilised by the CCP to make a reliable next step prediction even if the pedestrian's current movement behaviour is untypical and is not represented in her own history.

The outlined scenario forms the basis that is used to evaluate the CCP approach with regard to its ability to proactively filter pedestrians at risk.

5.3 Evaluation Method

Figure 5.3 gives a description of the method used to evaluate the accuracy of the context prediction algorithms using simulated data as outlined in Section 5.4 and using realistic data as outlined in Section 5.5.

In step one, the movement data histories are generated by using a simulator or they are gathered by using a smartphone the pedestrians carry in their trouser pockets. Afterwards, the data are stored in a data pool. In step two, the histories in the data pool are pre-processed using a sliding window approach as introduced in Section 2.1. In Step three the training data and the test data used

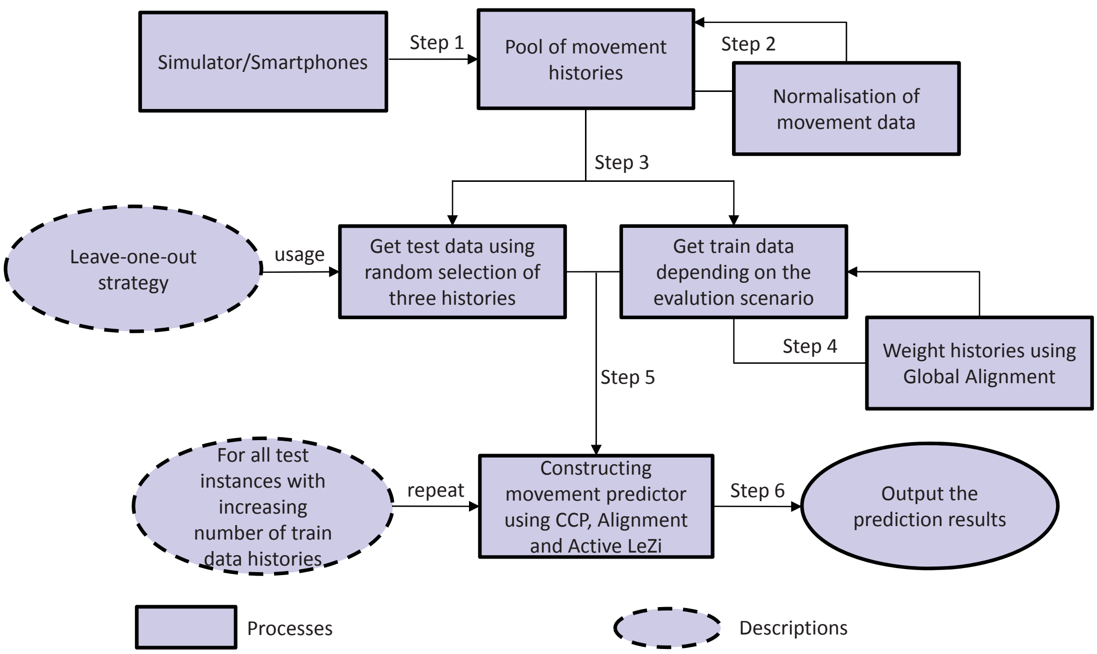

Figure 5.3: Used evaluation method.

for the evaluation are prepared. For the simulation-based evaluation ten movement histories are selected. For the evaluation based on realistic data, movement data from eight pedestrians are collected. Out of these ten respectively eight histories, three histories are selected by chance and are utilised as test histories. The remaining histories represent the training data. Training and test histories are split into instances. Each instance represents a movement part of a pedestrian on the pavement and consist of four parts of the pavement (cf. Section 5.2).

In Step four, the training histories are weighted regarding their similarity to the selected test histories. Therefore, the Needleman-Wunsch algorithm is used. During the evaluation process the leave-one-out method is applied. Hence, the instance used for prediction is temporarily erased form the movement history (test data set) of the respective pedestrian.

To further elaborate the prediction accuracy of the algorithms, different numbers of training data sets are used. The first training data set is represented by the histories of the three pedestrians the test data set is generated from. Afterwards, the training data set is increased by adding the histories of the remaining pedestrians regarding their obtained similarity score. Each time the number of used training data sets is changed, the prediction model is built anew in step five. In step six the accuracy modification depending on the chosen number of training data sets is outlined.

5.4 Evaluation using simulated data

To get an impression of how different context predictors perform under ideal circumstances with respect to the use case outlined in Section 5.2, they were applied to simulated data. Ideal circumstances mean that the movement data of the pedestrians do not have to be recognised and preprocessed by a smartphone first, as it is performed in Section 5.5. Altogether, two different environments were used to simulate movement data. The environments differ with respect to the segmentation of the pavement and the number

Table 5.1: Used Simulation settings.

Name	Size of Pavement	Nr. of paths	Instances
Figure 5.4	5×15	3 - 5	138
Figure 5.5	5×15	5 - 10	315
Figure 5.6	10×30	3 - 5	247
Figure 5.7	10×30	5 - 10	830

of movement paths a history of a pedestrian contains. The higher the number of paths stored in a pedestrian's history, the higher the resulting number of instances. The different simulation settings are presented in Table 5.1.

Figure 5.4 and 5.5 present the results gained by CCP, ActiveLeZi and Alignment for a pavement dimension size of 5×15 parts. Utilising the histories of the three pedestrians as training data the test data set has been generated from, CCP and ActiveLeZi receive a prediction accuracy of less than 5%. This is due to the fact that none of the three histories contains useful information since the movement pattern whose next step should be predicted is completely deleted from all of the three histories because the the leave-one-out strategy was used. Only the Alignment predictor receives a reliable prediction result. This might indicate that Alignment is more suitable to make reliable predictions for training data sets that contain unambiguous information. The similar affect has already been observed in the experiment outlined in Section 3.4.2. In this experiment, Alignment performed better than other prediction approaches on a less unambiguous data set, which has not been preprocessed using the sliding window approach.

While context histories that contain additional simulated movement paths were added to the training data set one after another, the prediction accuracy of CCP increases constantly and outperforms the accuracy of the two other prediction approaches. If the accuracy tends of Alignment and ActiveLeZi are considered it can be seen that the enlargement of the information space does not automatically improve the prediction accuracy of the two approaches. Rather, Figure 5.4 and 5.5 illustrate that the prediction accuracy can even drop, if the number of movement paths in a history is increased (cf. the results in Figure 5.4 using nine and ten simulated histories). In contrast, the prediction results of CCP remain nearly constant.

Figures 5.6 and 5.7 outline the results of the context prediction algorithms using a pavement size of 10×30 parts. First, it can be recognised that a higher search space, i.e., increasing the number of parts on a pavement from 5×15 to 10×30, does not decrease the prediction accuracy of CCP and ActiveLeZi. Only the overall accuracy of the Alignment approach decreases. Similar to the first two results, Alignment reaches the best accuracy considering only the histories as training data the test data set has been generated from. After adding additional simulated movement data the accuracy of the Alignment predictor only slightly increases. In contrast, CCP and ActiveLeZi are nearly able to constantly increase their prediction accuracy. In the case of five or more simulated histories of pedestrians, CCP always receives the highest prediction accuracy for all settings presented in Table 5.1.

The obtained prediction results up to 95% of CCP on the simulated data look quite promising. Next, realistic movement data obtained from pedestrians are used to evaluate the proposed prediction approaches.

5.5 Evaluation using realistic data

In this section, the context predictors have been used to predict next positions of a pedestrian based on realistic movement data. To collect the pedestrians' movements and direction changes on a pavement a Samsung Galaxy S II smartphone with Android 2.3.3 operating system, the pedestrians were wearing in their trouser pocket, has been used. Three ground truth annotations (W = walk straight ahead, L = turns left and then continues walking, R = turns right and then continues walking) were made with a Nokia

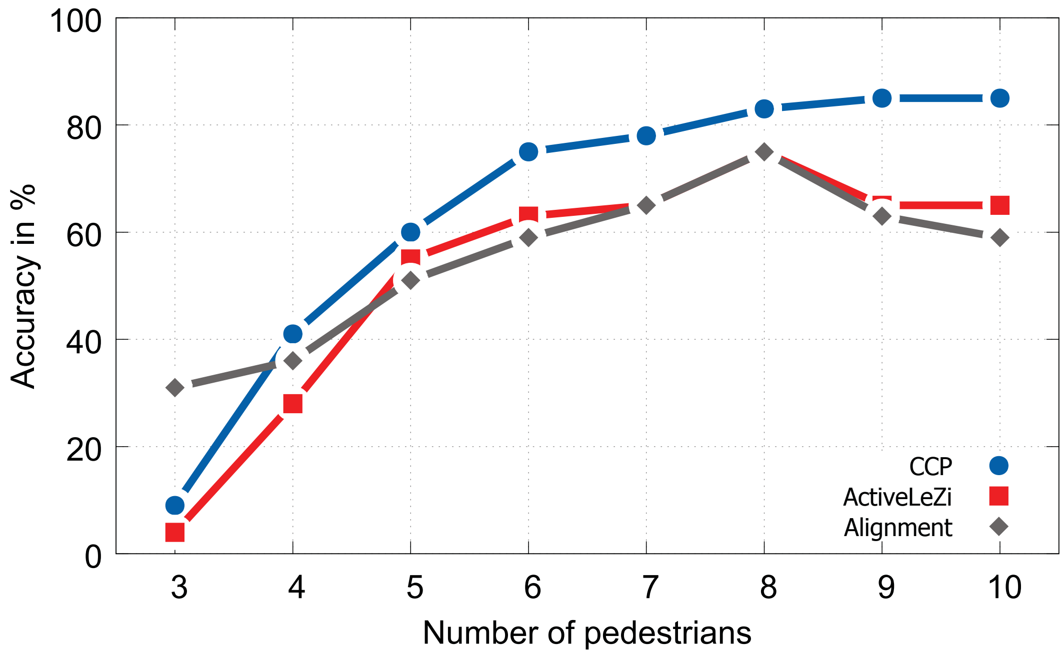

Figure 5.4: Prediction results using simulated data (cf. Table 5.1).

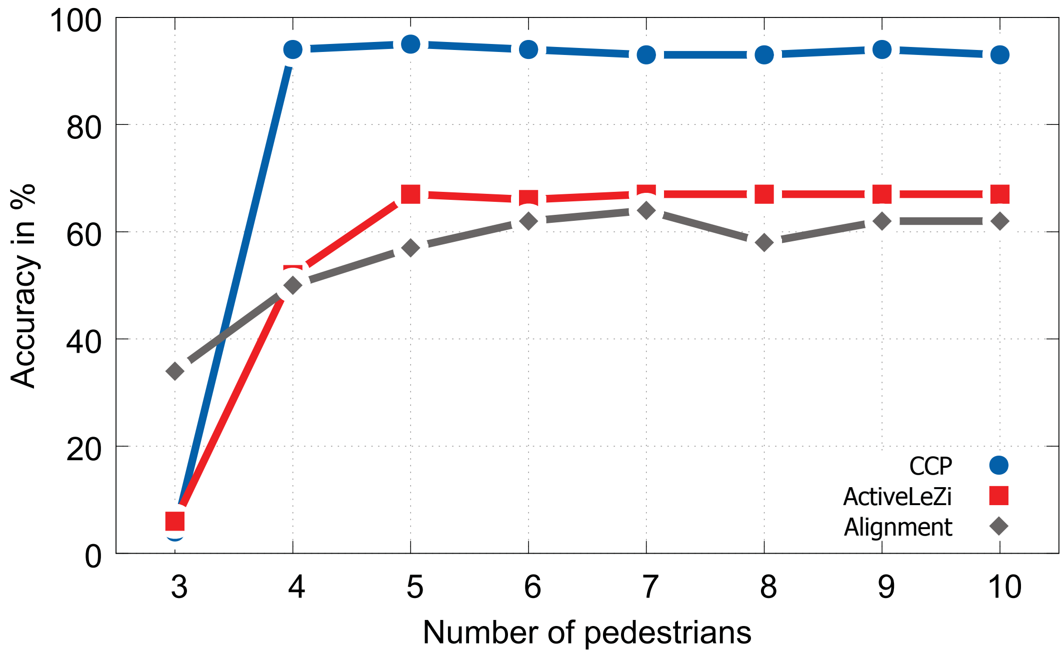

Figure 5.5: Prediction results using simulated data (cf. Table 5.1).

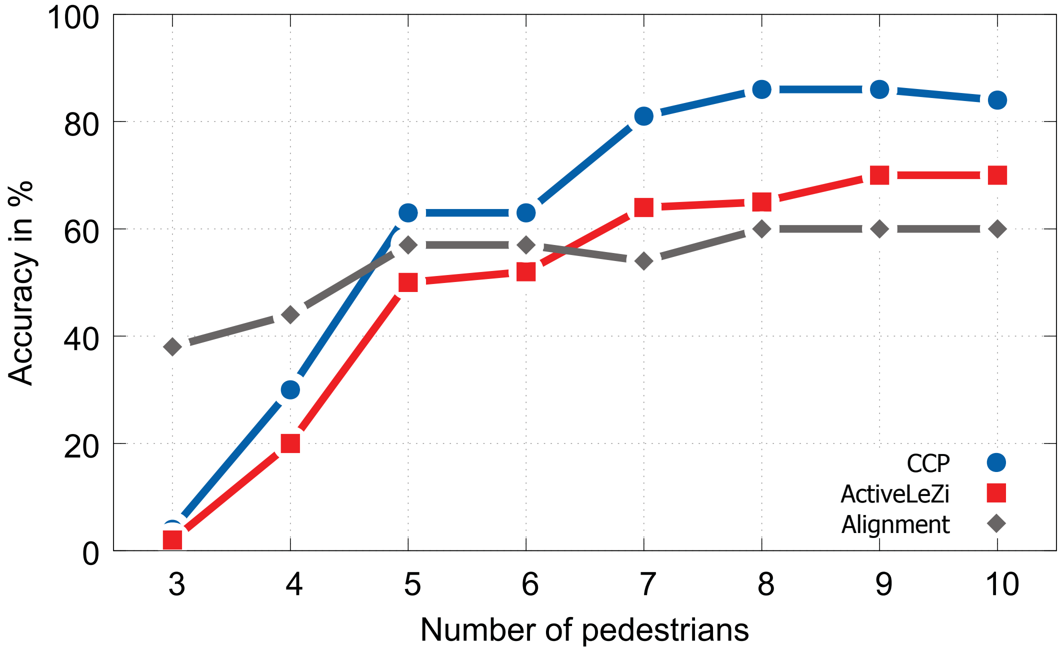

Figure 5.6: Prediction results using simulated data (cf. Table 5.1).

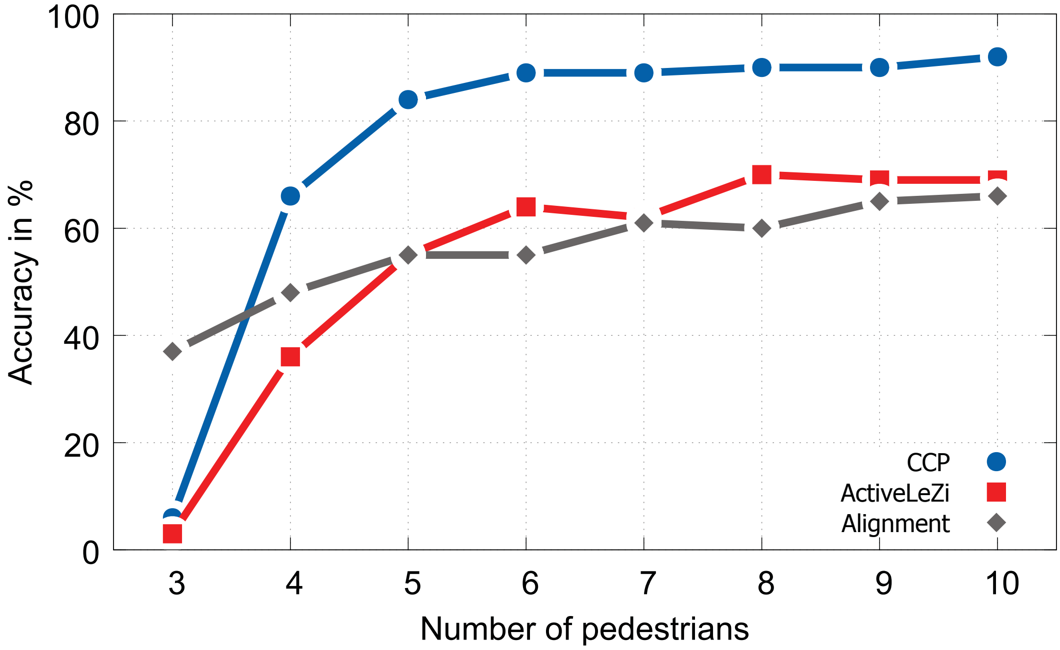

Figure 5.7: Prediction results using simulated data (cf. Table 5.1).

N800 Tablet during the measurement process of the sensor data. The annotated movement data has been saved to the respective context history of the pedestrian called $CH_{annotated}$.

The following software sensors available on the Samsung Galaxy S II smartphone have been selected: gravity, accelerometer, magnetic field, gyroscope, rotation and orientation. These sensors are derived from available hardware sensors installed in the smartphone, such as accelerometer, magnetometer and gyroscope sensors. The sensors deliver values in the x-, y-, and z-axis, which are relative to the screen of the phone in its default orientation. The gravity sensor provides a three dimensional vector that indicates the direction and magnitude of gravity. The accelerometer sensor measures the acceleration of the pedestrian. If the pedestrian is not moving, the accelerometer delivers only the value of $9.81 \frac{m}{s^2}$, which is the influence of gravity. Therefore, if the smartphone is stationary, the output of the accelerometer sensor should be identical to the output of the gravity sensor.

The magnetic field sensor measures the ambient magnetic field of each axis, while the gyroscope provides the angular speed around each axis. The rotation sensor provides a vector $x * sin(\frac{\theta}{2})$, $y * sin(\frac{\theta}{2})$, $z * sin(\frac{\theta}{2})$, where θ is the rotation angle and x, y, z are the axes relative to the device. The orientation sensor delivers three types of values, which are Yaw, Pitch and Roll. Yaw represents the compass heading in degrees. Pitch represents the tilt of the top of the smartphone while Roll represents the side-way tilt of the smartphone. The sensor values for each sensor were measured at a sampling rate of 32 Hz.

The gravity and accelerometer sensors were expected to capture the motion of the pedestrian. The other sensors were used to detect direction changes of the pedestrian. The measured sensor data and the corresponding movement patterns are presented in Figure 5.8. To reduce the dimensionality of the sensor data and to allow classification of the movement patterns, features were extracted from the obtained sensor data. Mean and standard deviation values were computed for each axis of every sensor. Further, the magnitude of all three axes was calculated. The sliding window technique was used to compute features. Each window consisted of one second of measurement data, which is equivalent to 32 instances of sensor values. No overlapping of windows was used for the computation of features. As a result, a total of 48 features were received.

To extract the movement paths of the pedestrians from the recorded sensor information, a Java implementation of the C4.5 decision tree learning algorithm was used. A partial depiction of the generated decision tree is outlined in Figure 5.9. The decision tree classifier automatically recognises the movement patterns of the pedestrians based on the computed features. As input for the learning algorithm, the computed features were combined with the ground truth annotations. The generated decision tree produced a recognition accuracy of 96.64%, while the training data was also used as test data. The outputs (W, L, R) were saved in the respective context history of the pedestrian called $CH_{recognised}$. If the classifier performs perfectly, the history $CH_{recognised}$ of a pedestrian corresponds to the history $CH_{annotated}$ of the respective pedestrian.

In order to use the recognised movement patterns to proactively filter pedestrians at risk, the recognised patterns have been mapped to the coordinate system, outlined in Section 5.2. The mapping is needed to assign the recorded and annotated movement patterns to the locations on the pavement where the movement of the pedestrians actually happened. A converter has been used to map the data stored in the context histories $CH_{annotated}$ and $CH_{recognised}$ to the actual coordinate representation of the pavement. In relation to the movement path depicted in Figure 5.10 the mapping results in "A0, A1, A2, A3, B4, B5, B6, B7, A8, A9, A10". Later, the resulted movement path is segmented into movement patterns (instances), using the sliding window approach.

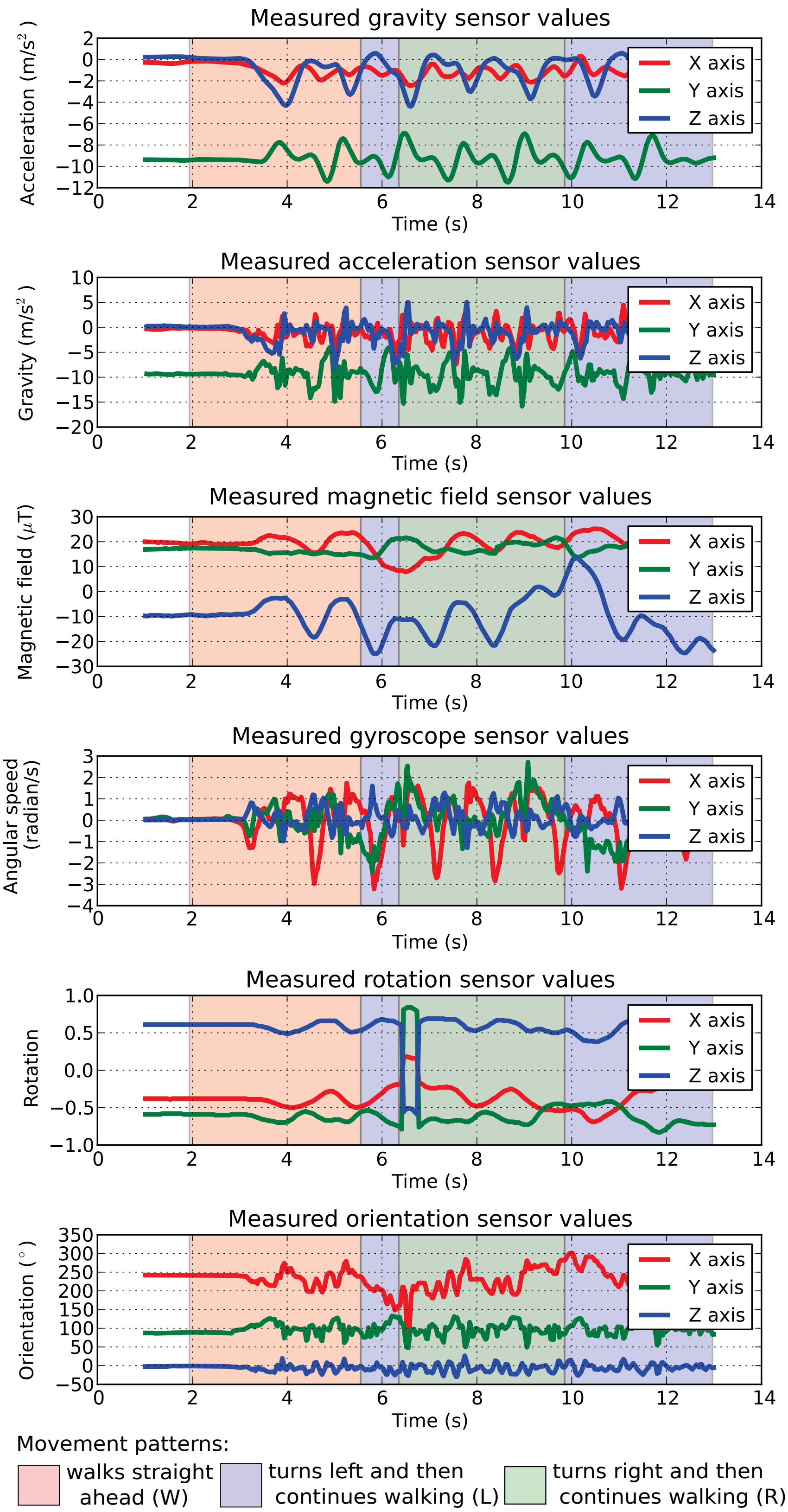

Figure 5.8: Different measured sensor values provided by a smartphone [12].

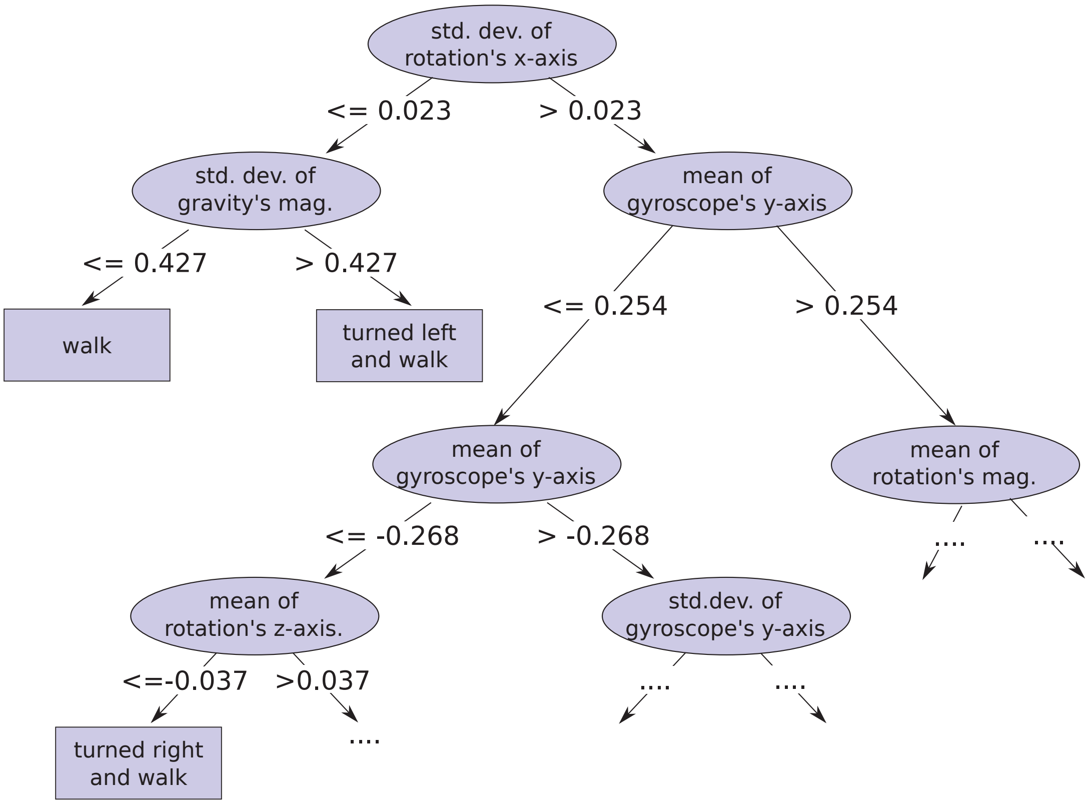

Figure 5.9: A partial depiction of the generated decision tree.

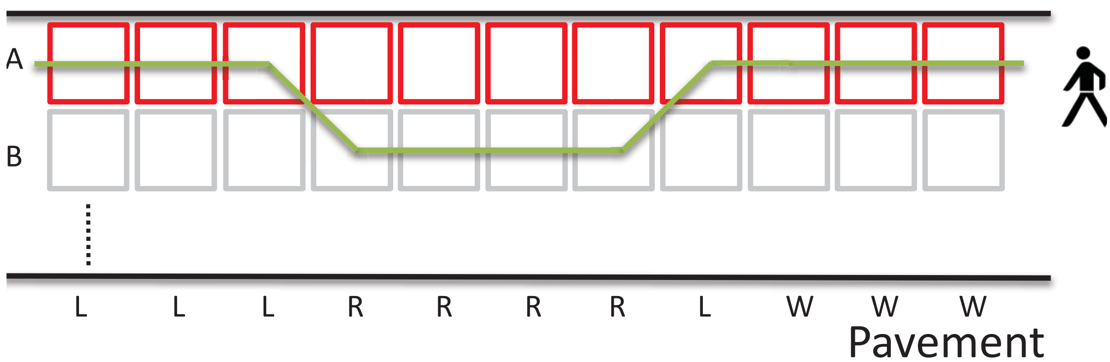

Figure 5.10: Example of the movement pattern recognition output [12].

5.6 Evaluation and Discussion

In this section the evaluation results of CCP, which has been used to proactively filter pedestrians at risk, are discussed. Next, the results of CCP are compared with the results gained by ActiveLeZi,

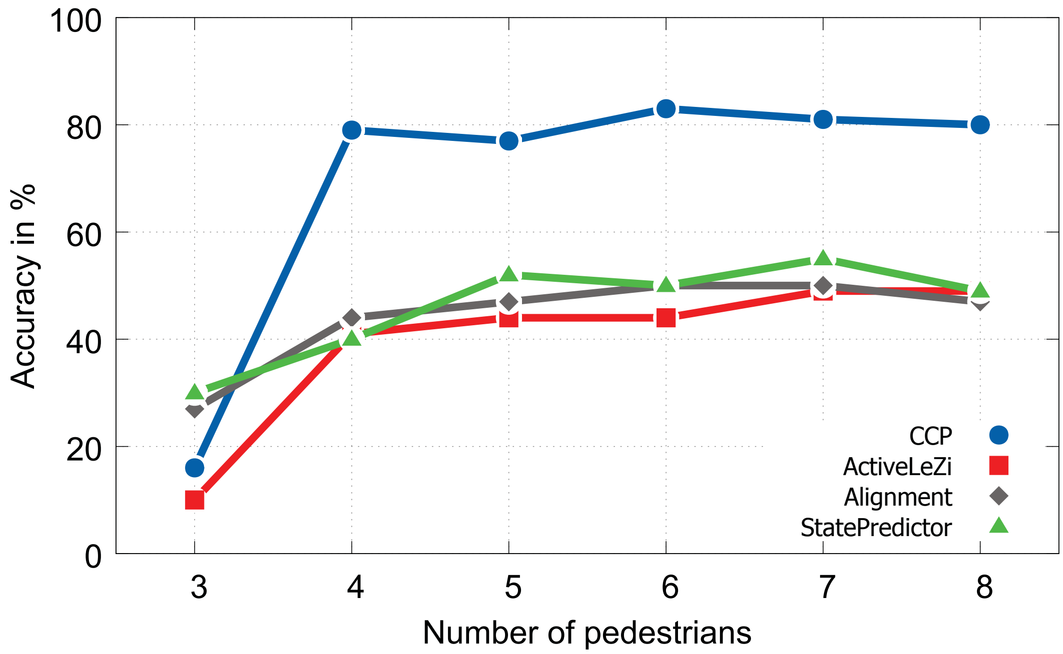

Figure 5.11: Maximum prediction baseline using $CH_{annotated}$ [12].

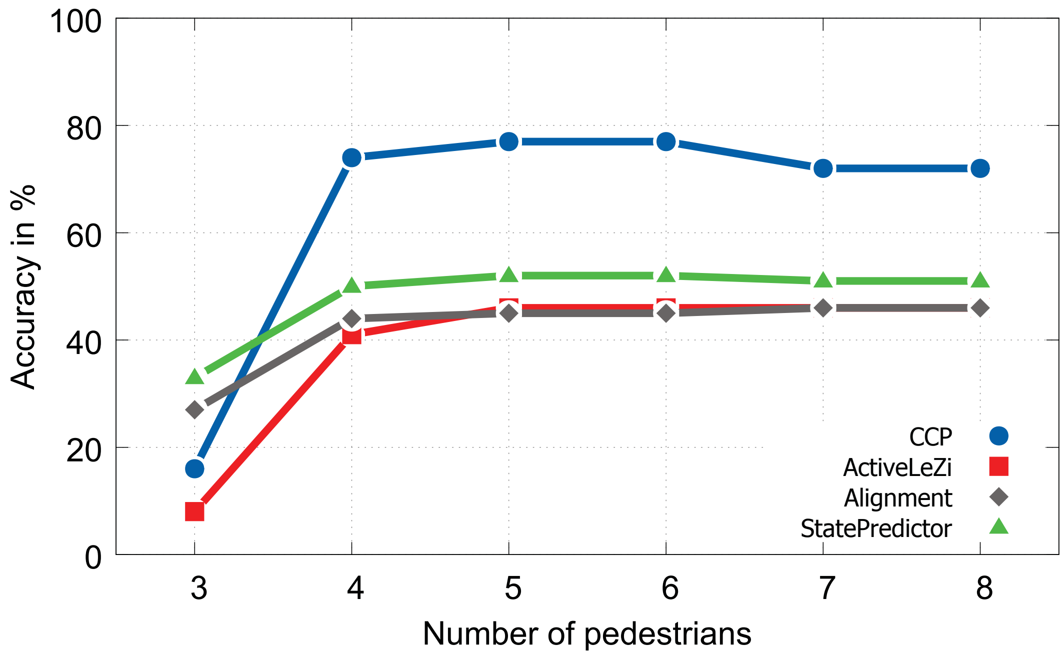

Figure 5.12: Prediction results using $CH_{recognised}$ [12].

Alignment and the StatePredictor. According to the evaluation method, outlined in Section 5.3, three different $CH_{annotated}$ histories have been picked by chance from the eight available histories. These data have been used to represent the test instances.

Afterwards, the $CH_{annotated}$ histories as well as the $CH_{recognised}$ histories were used to classify the test instances. The results outlined in Figure 5.11 present the maximum prediction baseline of the four context predictors using the $CH_{annotated}$ histories to train the prediction models. The results presented in Figure 5.12 show the prediction accuracies of the algorithms using the recognised movement data in $CH_{recognised}$ to train the prediction models.

The results presented in the Figures 5.11 and 5.12 show that CCP clearly outperforms ActiveLeZi, Alignment and the StatePredictor. In both cases, the gained accuracy of CCP is nearly 30% higher than the accuracy gained by the state of the art context predictors. The accuracy achieved by ActiveLeZi and Alignment is almost the same. The StatePredictor slightly obtains better results than Alignment and ActiveLeZi. Compared to state of the art context predictors, which try to find the best possible match for a given movement sequence, the utilisation of existing relations in the movement behaviours of the different pedestrians result in better prediction accuracies.

Comparing the results obtained by CCP using $CH_{annotated}$ and $CH_{recognised}$, it can be recognised that the prediction accuracy only slightly decreases using the recognised movement patterns of the pedestrians. The highest accuracy gained by CCP using $CH_{annotated}$ is 82% and 77% using $CH_{recognised}$. Due to the fact that the prediction accuracy of CCP using $CH_{recognised}$ only decreases by approximately 5% compared to the baseline given by $CH_{annotated}$, the obtained prediction results of CCP on $CH_{recognised}$ are quite promising. Consequently, the results indicate that the approach used to recognise a pedestrian's movement works reliable. The reduction of the accuracy can result from errors during the recognition of the realistic movement data of the pedestrians, e.g. due to incorrect movement annotations.

Altogether, the feasibility to automatically infer pedestrians' movements, using sensor information provided by a smartphone, has been proven. A recognition rate of 96.64% applying a C4.5 classifier has been achieved. Further, CCP was applied to the automatically recognised movement patterns to proactively filter pedestrians at risk. An accuracy rate of approximately 80% was achieved. With regard to the experiments outlined in this section, CCP needed 0.01 seconds to predict a pedestrian's next step on average. If the time needed to predict a pedestrian's next step is subtracted from the time a pedestrian needs to make a step (cf. Section 5.2) a significant time advantage of 0.55 seconds can be obtained. The time advantage can be used by a collision avoidance system to detect a possible collision between a pedestrian and an approaching vehicle in advance. Supposing the nearby car has a speed of 50 $\frac{km}{h}$ and the driver could have been alerted 0.55 seconds earlier by a collision avoidance system, under ideal circumstances, the driver may react 6.21 meters earlier to prevent a possible collision with a pedestrian.

5.7 Conclusions

In this chapter CCP is used to proactively filter pedestrians at risk by predicting their next step on the pavement in advance. As contexts, acceleration and orientation data, that describe the movements of the pedestrians, were used. The contexts were extracted from various sensors provided by a smartphone the pedestrians carried in their trouser pockets.

CCP was evaluated using two experiments. In the first experiment simulated context data were used and in the second experiment realistic movement data of the pedestrians were used. The prediction accuracy of CCP was compared to the state of the art context predictors introduced in Section 2.2. The results gained on the simulated data, showed that CCP is able to constantly increase its accuracy adding additional simulated context histories. CCP obtained a next step prediction accuracy up to 95%. In contrast, the accuracy of Alignment and ActiveLeZi is only up to 75% on the simulated movement data.

In the second experiment realistic movement data of pedestrians were used. CCP obtained a prediction accuracy of 82% using $CH_{annotated}$ histories and 77% using $CH_{recognised}$ histories. In contrast, the prediction accuracy of Alignment, ActiveLeZi and the StatePredictor only obtained results around 50%. The prediction accuracy of CCP only decreases by approximately 5% using the recognised movement data. Therefore, the obtained prediction result on the recognised movement data is quite promising.

With regard to the performed evaluation, the correct predict of a pedestrian's next step can offer a collision avoidance system a time advantage of approximately 0.55 seconds under ideal circumstances.

References

[1] "World report on road traffic injury prevention," 2004. `http://whqlibdoc.who.int/publications/2004/9241562609.pdf`, last accessed: 2013-05-12.

[2] H. Naci, D. Chisholm, and T. D. Baker, "Distribution of road traffic deaths by road user group: a global comparison.," *Inj Prev*, vol. 15, no. 1, pp. 55 –9, 2009.

[3] K. Fuerstenberg, "Pedestrian protection using laserscanners," in *Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE*, pp. 437 – 442, sept. 2005.

[4] L. Andreone, F. Bellotti, A. De Gloria, and R. Lauletta, "Svm-based pedestrian recognition on near-infrared images," in *Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on*, pp. 274 – 278, sept. 2005.

[5] T. Gandhi and M. Trivedi, "Pedestrian protection systems: Issues, survey, and challenges," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 8, pp. 413 –430, sept. 2007.

[6] K. David and A. Flach, "Car-2-x and pedestrian safety: Innovative collision avoidance system," *IEEE Vehicular Technology Magazine*, vol. 5, pp. 70 –76, march 2010.

[7] C. Sugimoto, Y. Nakamura, and T. Hashimoto, "Prototype of pedestrian-to-vehicle communication system for the prevention of pedestrian accidents using both 3g wireless and wlan communication," in *Wireless Pervasive Computing, 2008. ISWPC 2008. 3rd International Symposium on*, pp. 764 –767, may 2008.

[8] C. Sugimoto, Y. Nakamura, and T. Hashimoto, "Development of pedestrian-to-vehicle communication system prototype for pedestrian safety using both wide-area and direct communication," in *Advanced Information Networking and Applications, 2008. AINA 2008. 22nd International Conference on*, pp. 64 –69, march 2008.

[9] R. Raßhofer, D. Schwarz, E. Biebl, C. Morhart, O. Scherf, S. Zecha, R. Grünert, and H. Frühauf, "Pedestrian protection systems using cooperative sensor technology," in *Advanced Microsystems for Automotive Applications 2007*, VDI-Buch, pp. 135–145, Springer Berlin Heidelberg, 2007.

[10] A. Fackelmeier, C. Morhart, and E. Biebl, "Dual frequency methods for identifying hidden targets in road traffic," in *Advanced Microsystems for Automotive Applications 2008*, VDI-Buch, pp. 11–20, Springer Berlin Heidelberg, 2008.

[11] Z. Chen, D. C. K. Ngai, and N. H. C. Yung, "Pedestrian behavior prediction based on motion patterns for vehicle-to-pedestrian collision avoidance," in *Conference on Intelligent Transportation Systems, Beijing, China, October 12-15, 2008*, march 2008.

[12] C. Voigtmann, S. L. Lau, and K. David, "Evaluation of a collaborative-based filter technique to proactively detect pedestrians at risk," in *Vehicular Technology Conference (VTC Fall), 2012 IEEE*, pp. 1–5, 2012.

[13] A. Flach and K. David, "Combining radio transmission with filters for pedestrian safety: Experiments and simulations," in *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, pp. 1 –5, sept. 2010.

[14] *Global Positioning System Standard Positioning Service Performance Standard*, 2008. `http://www.gps.gov/technical/ps/2008-SPS-performance-standard.pdf`, last accessed: 2013-07-25.

[15] U. Weidmann, "Transporttechnik der Fußgänger," in *Schriftreihe des IVT Nr. 90*, (ETH Zürich), 1993.

[16] S. Bitgood and S. Dukes, "Not another step! economy of movement and pedestrian choice point behavior in shopping malls," *Environment and Behavior*, vol. 38, pp. 394–405, May 2006.

Chapter 6

Distributed and Collaborative Context Prediction

In this chapter the collaborative-based context prediction approach as well as the state of the art context predictors are directly applied on users' smartphones. The approaches utilise context information from various users collaboratively. The communication between the smartphones of the users is realised using peer-2-peer. Consequently, no centralised server unit is needed to process the context information of the users externally. Hence, most of the legal problems associated with context prediction, identified in this thesis, can be addressed. Finally, an evaluation of the proposed P2P-based context prediction architecture is provided and its possibilities to ensure a user's right to informational self-determination are discussed.

6.1 Motivation

With the evolution of today's smartphones into powerful and ubiquitous computing devices, it is possible to predict future contexts in a distributed and collaborative way. Up-to-date smartphones offer additional sensors like an accelerometer, a gyroscope or even a barometer or a sensor for near field communication that can be used to collect additional contexts of a user. Due to improved battery and processing power, collected context data can directly be processed on smartphones. Moreover, the increased available mobile bandwidth enables the user to send and receive data almost continuously.

The extension of the context prediction process by the additional usage of distributed and collaborative mechanisms increases the benefit for the user and for services that proactively adapt to the user's needs equally. Thus, for example, shopping places, a user is going to visit next, can be automatically predicted using the user's context history and the context histories of other users whose shopping interests show sufficient similarities. If the user makes her predicted shopping interest visible to her environment, personalised advertising can be displayed on her smartphone.

In addition, the proposed distributed and collaborative context prediction approach combines all tasks required to make existing applications in context prediction, like Car-2-Pedestrian scenarios [1] or users' next place prediction [2, 3] more suitable for daily live usage. From a technical perspective, gathered contexts must not be transferred and pre-processed on a server anymore, which is in most cases complex, time consuming and typically prevents just-in-time prediction. From the user's perspective, the prevention of external data processing hinders unauthorised third parties to gain access to personal data to create profiles. Hence, the principle in a user's right to informational self-determination, to avoid building profiles, can be satisfied.

Figure 6.1 shows the proposed distributed and collaborative context prediction approach. Distribution and collaboration in the context prediction process can be achieved by using peer-to-

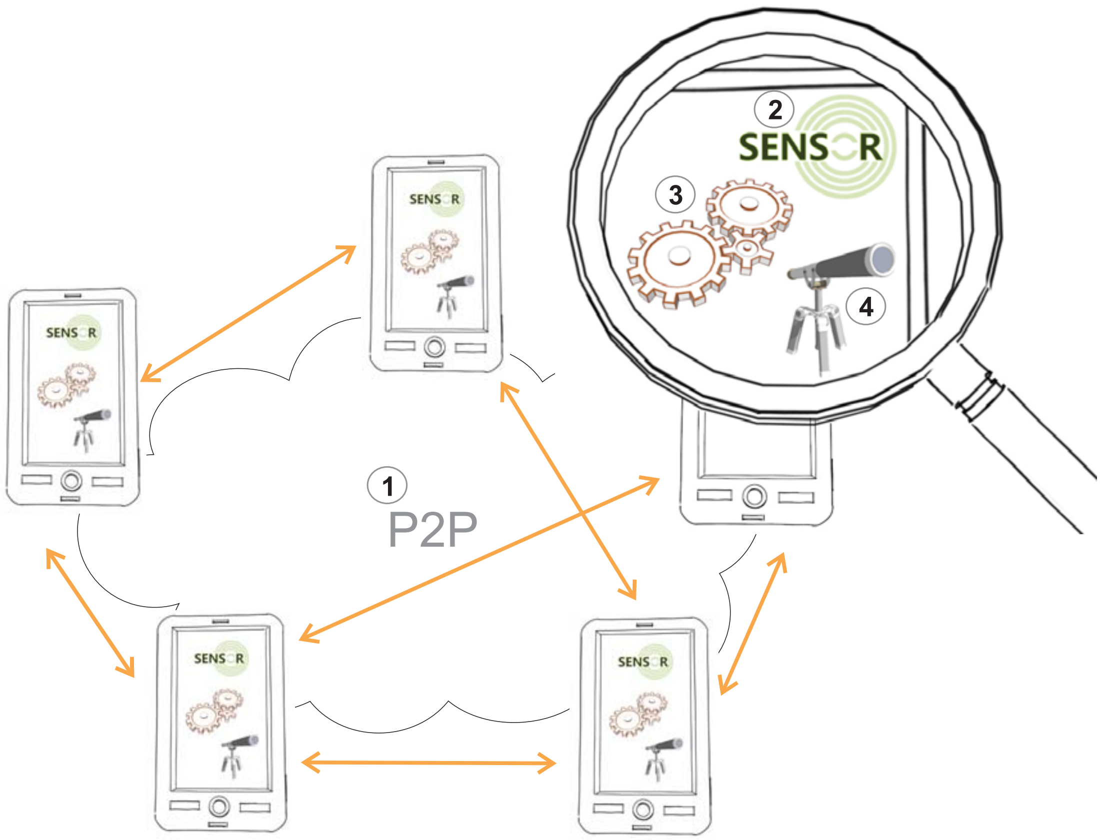

Figure 6.1: Collaborated and distributed context prediction process [4].

peer (P2P) communication (1) for the exchange of context data between different users, collected (2) and pre-processed (3) by their smartphones. Moreover, these contexts are utilised by prediction algorithms (4) that are directly executed on the smartphone to forecast a user's next context.

A first solution that uses a hybrid server and P2P approach for context monitoring, reasoning and prediction is proposed in [5]. The limitations of the used mobile devices prevented a standalone P2P solution. Another approach that built up a P2P-based context-aware information system using data gathered by mobiles is introduced in [6]. Mobile data is directly collected and shared by mobile phones of users. Due to limited battery and processing power of the mobile phones the devices cannot be used for the processing part of the context data.

An approach that proposes a P2P infrastructure to derive high-level context data from low-level context data is outlined in [7]. The main focus presented in this research work is the evaluation of the proposed P2P infrastructure with regard to memory consumption and query processing. In contrast to the above-mentioned research, current smartphones have been used directly to perform the context prediction tasks. Furthermore, no centralised server is used to handle communication between devices but P2P is used to enable direct communication. Next, following to the technical design proposals outlined in Section 4.5, to derive a legally acceptable context prediction implementation, the system model to describe the distributed and collaborative context prediction approach is characterised. Subsequently, the developed architecture is presented in more detail. Finally, the prediction times of the different approaches using the described architecture are evaluated and the impact of the proposed architecture to the right to informational self-determination is discussed.

6.2 System Model

To determine the requirements for the distributed and collaborative context prediction approach, a system model is defined that describes the technical design proposals outlined in Section 4.5 in more detail. The resulting system model describes the underlying environment of the proposed approach, characterises the approach and outlines its objectives. In total, the system model comprises three different dimensions, as shown in Figure 6.2 and detailed below.

The first dimension of the system model specifies the components the environment consists of (*structure model*). It is assumed that the algorithms to predict a user's next context are used in a distributed and collaborative manner. Supported context prediction algorithms are the CCP approach and the state of the art context predictors introduced in Section 2.2. Furthermore, it is assumed that the knowledge base to train a prediction model is not restricted to the user's own context history but also uses additional knowledge in context histories of other users. Additionally, it is assumed that no centralised server unit is used to perform the prediction of a user's next context. Hence, the users do not have to trust one central processing unit.

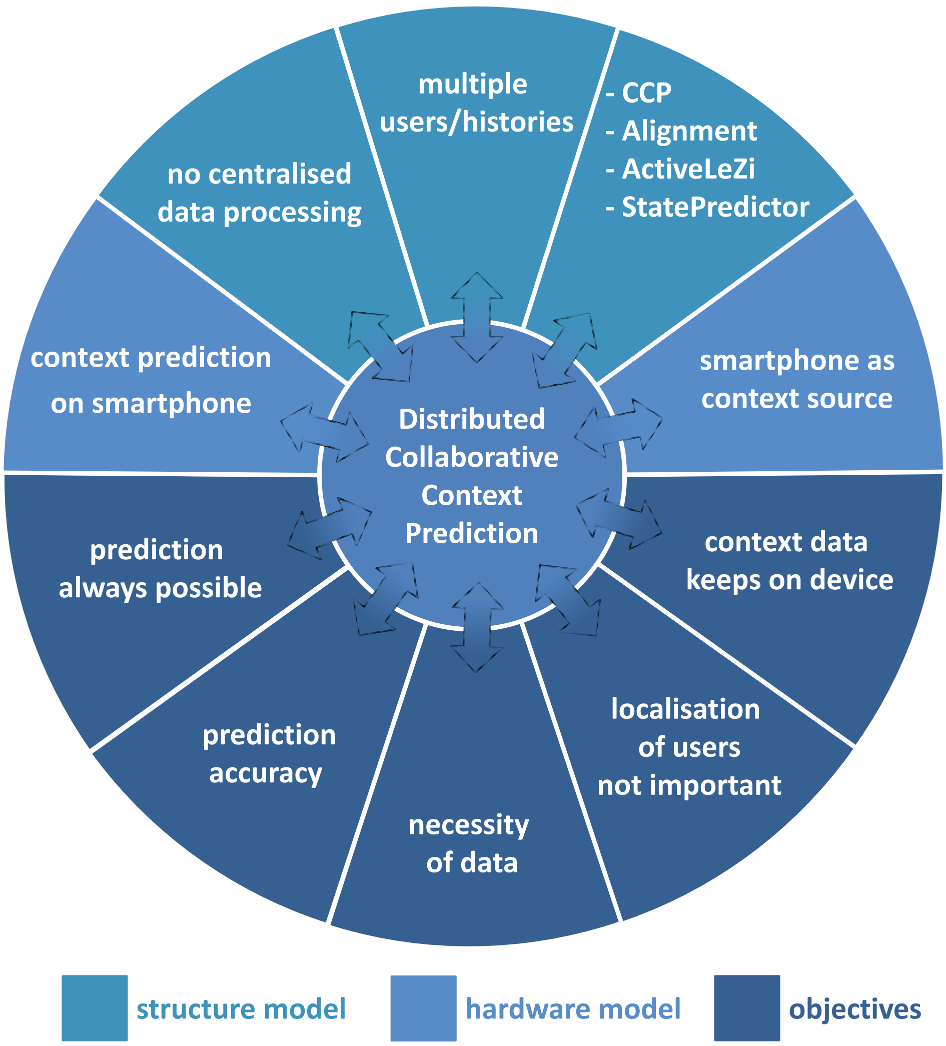

Figure 6.2: System model for distributed and collaborative context prediction [4].

The *hardware model* describes the assumptions about the hardware components used in the distributed and collaborative context prediction approach. It is assumed that in the environment only smartphones are utilised. Hence, smartphones are utilised to gather the contexts of a user. Further, it is used for the prediction process.

Objectives of the distributed and collaborative context prediction approach are that the prediction of a user's next context always has to be possible, even if a user's own context history does not provide sufficient context information. Therefore, it should also use context information of other users whose context histories show sufficient similarities. Further objectives are that current whereabouts of the users, whose context histories are used for the prediction process, are not important for the distributed and collaborative context prediction approach. For this reason, a geographical proximity of users is not necessary. Context data of a user is only stored on the user's smartphone. If context information has to be transmitted, it has to be pseudonymised. Additionally, only context information that is necessary for the prediction process is stored. Any contexts that are not relevant have to be deleted. Finally, the achieved prediction accuracy of a used algorithm has to be sufficiently accurate.

6.3 Requirements

Based on the system model outlined in the previous section, technical and general requirements are derived in this section. These requirements provide the basis for a realistic implementation of the distributed and collaborative context prediction approach.

In the system model a user's smartphone is proposed as a computational device for the distributed and collaborative context prediction scenario. In the proposed approach, smartphones utilise built-in soft- and hardware sensors, e.g., accelerometer, magnetometer, gyroscope, etc. to automatically collect context information of a user. Collected context information is only stored on the user's smartphone. Further, smartphones serve as processing units. Hence, context prediction algorithms run directly on the smartphone to predict a user's next context.

In order to use smartphones for these tasks, the devices have to be up-to-date with respect to their processor unit and

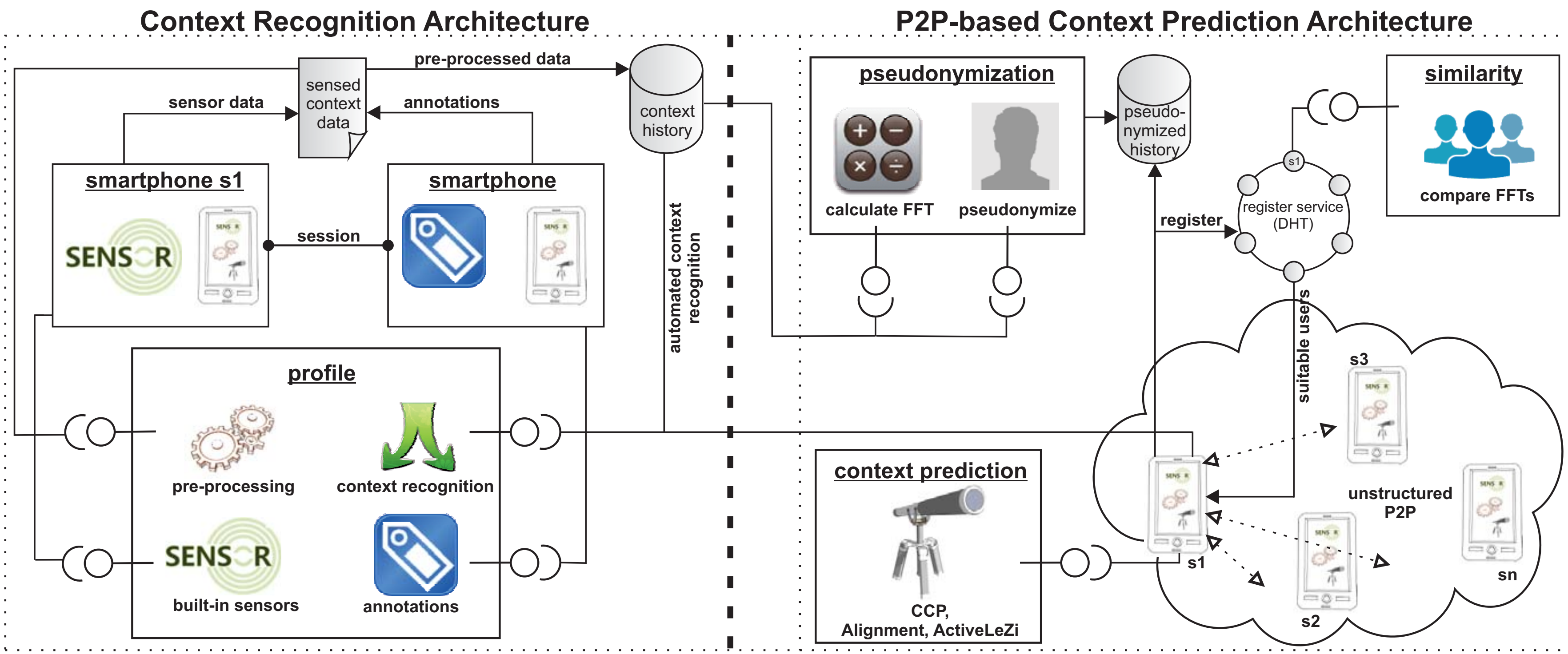

Figure 6.3: Proposed architecture for the distributed and collaborative context prediction approach [4].

internal memory size. Otherwise, the time needed to predict a user's next context directly on the device might take too long to provide just-in-time context prediction. Collaboration, respectively the combined usage of context information of different users, is used to achieve high prediction accuracy and to provide context prediction even if the context history of the user does not contain suitable information. This implies that contexts located in histories belonging to other users that are stored on their own smartphones must also be utilised by the prediction process if necessary. Therefore, smartphones require a stable connection to the internet. If a context predictor needs context information from other users to make a reliable prediction, required context data must be transmitted pseudonymised. A centralised server unit must not handle the communication between the smartphones of the users during a prediction process. A prerequisite is the usage of P2P communication between the smartphones of the users. Thus, context information is not concentrated on a processing unit of a single service provider.

As context prediction approaches, Alignment, ActiveLeZi, StatePredictor and the Collaborative Context Predictor (CCP) have to be supported. All approaches are state of the art context prediction algorithms. Regarding their different working methods, the following requirements have to be considered: To use Alignment, ActiveLeZi or the StatePredictor in a collaborative manner, the context sequence which is used to predict a user's next context is sent to the smartphones of appropriate users using P2P communication. Subsequently, the user who sends the context sequence and the users who also receives her context sequence use Alignment, ActiveLeZi or the StatePredictor to make a prediction on their own devices using their own context histories. Afterwards, each user returns the predicted context to the user the context sequence originally came from. Finally, voting is used to determine the context that follows up the given context sequence. In contrast to the other algorithms, CCP needs at least one additional context history of another user on the same smartphone of the user whose next context has to be predicted, to work properly. For this reason, the context histories of the users, which will also be used, have to be completely transmitted to the smartphone of the user. At that time, a distributed calculation of the prediction model of the CCP is not implemented. Consequently, CCP is executed on the user's smartphone whose next context has to be predicted.

To limit the number of additional context histories used to make reliable predictions, those who are most appropriate have to be identified first. Hence, it is necessary to compare the history of the user whose context has to be predicted with those histories available in the P2P network of the other users. Context histories do not need to be compared directly to each other to avoid additional communication traffic and to avoid that context histories are processed in plain text centrally.

6.4 Our Approach

In this section, the approach for the distributed and collaborative context prediction is presented. The underlying architecture is outlined in Figure 6.3. The proposed architecture is divided into two parts: The *Context Recognition Architecture* describes how high-level context information of a user can be automatically received and processed using the built-in sensors of a user's smartphone. It was developed in the course of the following master thesis [8]. The *P2P-based Context Prediction Architecture* describes the P2P-based context prediction process which is executed on users' smartphones and was developed in the course of the following bachelor thesis [9].

6.4.1 Context Recognition Architecture

To provide an easy-to-use-possibility for a user to collect context information, a web application is provided. By using this web application the user is able to create profiles, respectively templates, to automatically gather high-level context data. Defined profiles can be simultaneously accessed and used by arbitrary smartphones.

After a profile has been chosen on smartphone s1, it automatically starts the tasks specified in the profile.

A profile determines which built-in sensors of a smartphone are utilised to collect context information of a user. All available hard- and software sensors are supported. It is also possible to specify the pre-processing of collected sensor data. An example can be the deletion of redundant sensor information or clustering sensor information to meaningful high-level context information using, e.g. k-Means or other appropriate algorithms. In addition, low-level sensor data can be mapped to high-level sensor data using annotations predefined in the profile. Annotations can, e.g. be *walking, sitting, standing*, if built-in sensors are used to recognise the movement behaviours of a user. These annotations can be used by another smartphone, which accesses the same profile to label the sensor information currently collected by the smartphone s1.

Annotations and collected sensor information are automatically merged after s1 has stopped its data collection process. The merging result represents the context history of the user. The history can be used by context recognition approaches also defined in the profile to automatically derive high-level sensor data from low-level sensor data, gathered by built-in sensors using supervised approaches. Then, no manual annotation of the gathered sensor data is needed.

6.4.2 P2P-based Context Prediction Architecture

The second part of the architecture performs the prediction process to forecast a user's next context based on her most recently recognised sensor data. The user's most recent context data is automatically derived from the sensor data using her context history located on s1 and a supervised learning approach specified in the profile. Before the prediction process starts, the context history of the user is pseudonymised. In addition, the Fast Fourier Transformation (FFT) of the user's context history is calculated on s1. Subsequently, the FFT representation is transferred in a vector of quantifiers where each FFT floating-point value is transferred into a value of discrete range between 0 and 4. This vector of quantifiers represents the context history of a user and is used to identify similar context histories of other users without comparing the histories directly but by comparing the vectors.

The usage of a server-based register would be the simplest way to perform the similarity check of the vectors. However, this would violate the requirement of a pure decentralised architecture with no single-point-of-failure. Thus, a register service, which is distributed among all users by using a distributed hash table (DHT) as the underlying architecture for a P2P network, is proposed. Each device that is available in the P2P network can be used to predict a user's next context. Therefore, it registers, with the register service, its current IP-address, with its vector of quantifiers and with the profile ID the context history of the user has been generated with. Thus, the used key for storing this information in the DHT-based register service must be derived from its vector of quantifiers while preserving order to enable other devices to search for similar histories. A device can also be registered with several vectors that belong to different context histories that have been generated using different profiles. As soon as the user selects the preferred context prediction approach on device s1, the smartphone sends a prediction request to the register service, i.e., the P2P network. The prediction request includes the profile ID, the IP address of s1 and the quantifier as the key. The responsible device in the DHT for the requested vectors returns the IP addresses of the devices to s1 whose quantifiers are most similar to the quantifiers sent from s1, i.e., all values found in the keyspace around the requested key ($k \pm c$, with k being the requested key and c a constant). After that, s1 initialises connections to the devices of the users, whose context histories show the most sufficient similarities, using socket communication.

If the user chooses Alignment, ActiveLeZi or the StatePredictor to perform the prediction task, s1 sends the pseudonymised context pattern, whose next context has to be predicted, the chosen prediction approach and the profile ID of the current context history to the devices a connection has been established with. Subsequently, all devices connected to s1 and s1 itself perform the prediction task for the current context pattern using their own pseudonymised context history.

After the prediction task has been finished, all devices return their prediction to s1. The final prediction results from a majority vote of all incoming prediction results. If the user chooses the CCP approach, the pseudonymised context histories of the connected users have to be sent to s1 first. Afterwards, the prediction task is performed directly on s1. A distributed calculation of the HOSVD has not been implemented, so far. If the prediction task is finished, the received context histories are deleted. The proposed *P2P-based Context Prediction Architecture* complies to the requirements described in Section 6.3 for the distributed and collaborative context prediction approach: mobile devices instead of PCs are used for the calculation tasks; context prediction approaches are directly executed on the mobile devices; recognised context data are solely stored on a user's mobile device; context histories do not have to be transferred to other user's devices except if CCP is used for prediction; transferred context data, e.g. the current context pattern or context histories for CCP are pseudonymised; only devices of users are used whose context histories show sufficient similarities to the user whose next context has to be predicted; communication between devices is handled using P2P-based communication, no centralised server unit is needed.

6.5 Experimental Evaluation

In this section, the experimental evaluation of the *P2P-based Context Prediction Architecture* is discussed. Experiments to determine the prediction time needed by the distributed and collaborative context prediction approach and the three state of the art approaches using Wi-Fi and UMTS connectivity are outlined. In the scenario, four users were involved. Each user had its own smartphone. On each smartphone, training data belonging to three different context data sets have been stored. The first data set consists of movement behaviours (sitting, standing, walking, etc.) of four persons, as described in Section 3.4.1. Here, a data set is called *mov*. The second data set contains outdoor movement paths of four pedestrians derived from various sensors built-in a smartphone carried by pedestrians, as described in Chapter 5. Here, the data set is called *ped*. Further, the modified version of the Augsburger data set, as described in Section 3.4.2, was used. Here, it is called *augs*. Each data set consists of training- and test data. The training data is used to build the prediction model for a chosen context predictor. The test data is used to evaluate the results of the predictors using their trained model for a certain data set. The training data belonging to a certain context data set is unique on all smartphones. Hence, each user provides different context information for a certain context data set. During the experiments the prediction tasks in the P2P environment were performed on Motorola DROID RAZR MAXX smartphones. Each smartphone has a dual-core 1.2 GHz Cortex-A9 processor and 1 GB RAM. In the experiments, the required time to make various forecasts with a user's smartphone s1 was proposed.

First, a baseline is given by measuring the needed prediction times of the predictors for the three different data sets on a server unit (*pc*). The server unit has an Intel Core i7 with 2 GHz and 8 GB RAM. In addition, the needed prediction times are also measured using only the smartphone of one user (*s1 local*) that holds the test data of the three data sets. In both cases, the training data of all four users are previously merged to one big training data for each data set. This is because no P2P communication has been used for this experiment to derive additional context information of other users. Moreover, the prediction time, needed to predict the contexts for all instances of a given test data belonging to a certain data set using P2P communication, was measured.

The measurements have been performed while the four smartphones have been connected using Wi-Fi (*p2p wi-fi*) respectively using UMTS (*p2p umts*). If all devices are located in the same Wi-

Fi network, a direct connection between devices can be established. Otherwise, if the devices use the UMTS network, they have to share the same VPN connection. A direct connection between mobile devices using UMTS is not possible because the telecommunication provider blocks it. Furthermore, the accuracy (*acc*) gained by the prediction approaches and the number of test instances (*ins*) included in a test data set are outlined.

Table 6.1: Measured prediction times in seconds using the P2P-based context prediction architecture.

CCP						
data	*pc*	*s1 local*	*p2p wi-fi*	*p2p umts*	acc	ins
mov	0.74	13.9	13.5	49.1	90%	20
ped	0.30	1.39	3.9	41.2	87.5%	24
augs	5.3	166	162	471	74%	50
Alignment						
data	*pc*	*s1 local*	*p2p wi-fi*	*p2p umts*	acc	ins
mov	90 ms.	0.96	16.1	86.9	75%	20
ped	61 ms.	0.59	4.5	73.7	66.6%	24
augs	167*ms.*	12.3	75.4	196.2	8%	50
ActiveLeZi						
data	*pc*	*s1 local*	*p2p wi-fi*	*p2p umts*	acc	ins
mov	160 ms.	5.43	19.6	62.45	80%	20
ped	94 ms.	1.13	7.8	49.1	66.6%	24
augs	1.7	82	750	$---$	10%	50
StatePredictor						
data	*pc*	*s1 local*	*p2p wi-fi*	*p2p umts*	acc	ins
mov	1 ms.	4 ms.	4.5	14.3	84%	20
ped	1 ms.	5 ms.	4.9	14.1	78.3%	24
augs	2 ms.	9 ms.	18.1	40.3	65.3%	50

The results of the experiments are shown in Table 6.1. The baseline presented by (*pc*) shows that the server always needs the shortest execution times for all prediction approaches and all data sets. The same experiments needed longer execution times when performed directly on the smartphone s1 (*local s1*). In both cases CCP required the longest execution time because of its complex mathematical computations. *P2P-Wi-Fi* and *P2P-UMTS* show the execution times of the algorithms on the two data sets using P2P for direct communication between the four devices. The measured execution times are significantly higher than the execution times measured without P2P communication. The reason is the additional cost of communication needed to send the context pattern to the other smartphones respectively to receive the prediction results and the context histories from the other smartphones. Nevertheless, the average prediction times per instance for the algorithms are quite promising. They range between 0.16 and 0.98 seconds in *p2p wi-fi* and between 1.72 and 4.34 seconds in *p2p umts* for a single prediction, depending on the chosen algorithm and on the chosen data set. The faster prediction times of CCP on the *mov* and *ped* data sets compared to Alignment and ActiveLeZi result from the less demand of communication needed between the smartphones. CCP needs to establish a P2P communication to the other three devices only once to get the context histories of the users. ActiveLeZi and Alignment establish a P2P communication to the other devices for every test instance. The highest prediction accuracy for all data sets has been achieved by CCP.

6.6 Legal considerations

In this section it is discussed if the proposed distributed and collaborative context prediction approach ensures the user's right to informational self-determination and if it can therefore be considered to be legally acceptable.

To come straight to the point - all aspects which have to be fulfilled that the presented approach can be considered as fully legally acceptable can not be addressed in this thesis. Thus, for example, current decryption strategies that can be used to secure the communication channels between the smartphones as well as the complete decryption of the context data on a user's smartphone are not considered. However, the missing consideration to secure the communication channels has been addressed implicitly because VPN has been utilised to establish the communication while using UMTS connectivity (cf. Section 6.5).

Nevertheless, the presented approach for distributed and collaborative context prediction addresses four of the five principles of a user's right to informational self-determination. The principles have been outlined in Figure 4.2 in Section 4.5 as conflicting legal requirements with respect to the technical requirements of a context prediction process. In Table 6.2 the different technical implementations of the approach are mapped to the principles of a user's right to informational self-determination:

The mapping of the provided functions to the according principles outlines that the presented approach also considers the principles to informational self-determination in addition to collaborative-based and distributed context prediction. Therefore, the design proposals, suggested by KORA (cf. Figure 4.2), have been implemented successfully. For this reason, the proposed approach can be considered as a first step to provide context prediction in a way that it is legally acceptable for the users. Only the principle of giving consent has not been addressed by the approach to make sure that it remains unobtrusively. Otherwise, the user would be forced each time to confirm if her smartphone is allowed to be utilised during the prediction process of a user's next context.

6.7 Conclusions

In this chapter, an approach for distributed and collaborative context prediction has been presented. The idea of the approach is to provide a solution for the legal implications caused by existing context prediction approaches and architectures as outlined in Chapter 4. For this reason, the presented approach implements the design proposals, which have been figured out by the KORA method in this thesis.

First, a system model based on the design proposals has been

Table 6.2: Mapping the principles of a user's right to informational self-determination to the different technical implementations.

principles	**technical implementations**
transparency	Contexts are solely stored on a user's smartphone. For this reason, the user is able to look at her data at any time.
necessity	Solely context histories of other users are used in the collaborative-based prediction process that show sufficient similarity to the history of the user whose next context has to be predicted. Therefore, it is ensured that only data, which are needed are utilised.
profile	Histories are not compared in plain text but by their vector representations. As a result, context histories do not have to be transferred form a user's device to another.
profile	If context data have to be transmitted, e.g. for the CCP approach, the data are transmitted pseudonymised.
transparency	The processing of the context data and the prediction process is only performed on a user's smartphone.
responsibility	No centralised server unit is used to handle the communication between the smartphones but p2p. Therefore, no additional service provider is needed.
transparency	The user has access to the context collection profile used to collect certain context data. Further, the user also has the possibility to configure the profile as described in Section 6.4.1.

presented to determine the requirements for the approach in more detail. Subsequently, the architecture the approach consists of has been outlined. The architecture consists of two parts, the context recognition architecture and the p2p-based context prediction architecture. The evaluation given in this chapter showed that the algorithmic approaches presented in this thesis could provide context prediction results in a reasonable time, using the p2p-based context prediction architecture.

Finally, the different functions of the approach have been mapped to the principles of the right to informational self-determination. It could be shown that the approach can be considered as a first step to provide context prediction in a way that it is legally acceptable to the users.

References

[1] C. Voigtmann, S. L. Lau, and K. David, "Evaluation of a collaborative-based filter technique to proactively detect pedestrians at risk," in *Vehicular Technology Conference (VTC Fall), 2012 IEEE*, pp. 1–5, Sept. 2012.

[2] T. Yoon and J.-H. Lee, "Goal and path prediction based on user's moving path data," in *Proceedings of the 2Nd International Conference on Ubiquitous Information Management and Communication*, ICUIMC '08, (New York, NY, USA), pp. 475–480, ACM, 2008.

[3] T. H. N. Vu, K. H. Ryu, and N. Park, "A method for predicting future location of mobile user for location-based services system," *Comput. Ind. Eng.*, vol. 57, Aug 2009.

[4] C. Voigtmann, C. Schuette, A. Wacker, and K. David, "A new approach for distributed and collaborative context prediction," in *10th IEEE Workshop on Context Modeling and Reasoning 2013 (CoMoRea 2013)*, (San Diego, USA), pp. 20–24, IEEE, Mar. 2013.

[5] P. Nurmi, M. Martin, and J. A. Flanagan, “Enabling proactiveness through context prediction,” in *In: CAPS 2005, Workshop on Context Awareness for Proactive Systems.*, 2005.

[6] K. F. Yeung, Y. Yang, and D. Ndzi, “A context-aware system for mobile data sharing in hybrid p2p environment,” in *Autonomic and Autonomous Systems, 2009. ICAS ’09. Fifth International Conference on*, pp. 63–68, April 2009.

[7] T. Gu, H. K. Pung, and D. Zhang, “Peer-to-peer context reasoning in pervasive computing environments,” in *Pervasive Computing and Communications, 2008. PerCom 2008. Sixth Annual IEEE International Conference on*, pp. 406–411, March 2008.

[8] A. Wilhelm, “Framework für die automatisierte Erfassung und Verarbeitung personenbezogener Kontextinformation,” Master’s thesis, Kassel University, 2012.

[9] C. Schütte, “Kontextvorhersage in Smartphone-basierten peer-to-peer Netzwerken unter Betrachtung rechtlicher Implikationen,” 2012. Universität Kassel, Bachelorarbeit.

Chapter 7

Conclusions and Outlook

In this PhD thesis, a contribution to the understanding and development of a collaborative-based context prediction approach has been given. Further, the presented collaborative-based context prediction approach has been evaluated by comparing the approach with state of the art context predictors. Evaluation results have been presented regarding their prediction accuracies and considering their compatibility with the right to informational self-determination. Finally, an architectural approach, that provides collaborative-based context prediction in a way that it is legally acceptable to the users, has been introduced. This chapter summarises the thesis and provides an outlook to open research questions.

7.1 Conclusions

Context prediction is one important technique used in ubiquitous computing systems. Context prediction enables ubiquitous computing systems to proactively adapt its services, applications, algorithms, etc. proactively to the user's needs. To provide predictions, appropriate algorithms are used. As knowledge base, the algorithms utilise the context history of a user. If the history of a user does not provide sufficient context data, current prediction approaches will fail in predicting the next context.

As reviewed in Chapter 2, there do not exist approaches that provide context prediction in a collaborative manner. Considering the fact that most ubiquitous computing environments, e.g. smart homes, meeting rooms, cars or public places like shopping malls and airports are highly collaborative, it might be obvious that users who share the same environment may assist each other by also sharing their context information. This additional information can be used in the context prediction process to provide reliable prediction, even if the history of the user does not provide sufficient information.

Context data used to predict a user's next context are in most cases highly personal. A user's context data can, e.g. provide information to her locations in the past and the time the user has been at a certain place, her habits, her contacts, etc. All these contexts are mostly collected unobtrusively by a ubiquitous environment and also used unobtrusively, e.g. by context prediction approaches. This contradicts with the German right to informational self-determination as well as it contradicts with the European Data Protection Directive. The contradiction will be considerably greater, if context histories of other users are automatically integrated in a collaborative-based context prediction process.

Therefore, in this thesis, an approach to collaborative-based context prediction, its technical and legal evaluation and an architectural approach to bring collaborative-based context prediction approach in line with the right to informational self-determination are presented in the Chapters 3-6.

A possible solution, not solely to apply the user's history to predict her next context but to integrate the histories of the users that share the same ubiquitous environment, is given with the

Collaborative-based Context Predictor (CCP). In Chapter 3, the term Collaborative Ubiquitous Environment has been introduced. The Collaborative Ubiquitous Environment consists of three different entities, the users $\mathrm{U} \in \mathcal{U}$, the possible context patterns $\mathrm{Cp} \in \mathcal{CP}$ and the predictable future contexts $\mathrm{Fc} \in \mathcal{FC}$. The Collaborative Ubiquitous Environment represents the knowledge base used by CCP to make its predictions. CCP utilises the Higher-order Singular Value Decomposition (HOSVD) technique to enrich the context histories of the users with additional latent information. Consequently, the knowledge base is represented by a 3-order tensor structure. HOSVD is applied on the 3-order tensor to calculate the core tensor $\underline{\mathbf{\Sigma}}$ that spans the information space that only contains the most relevant information of the Collaborative Ubiquitous Environment. Afterwards, based on the reduced information space, the tensor $\underline{\mathbf{T}}'$ that includes additional latent relations between the entities of the collaborative ubiquitous environment, is calculated. This additional latent information are further used to provide a more reliable and collaborative-based context prediction to the user. A proof-of-concept of CCP is given in three evaluations. First, CCP has been used to forecast a user's next step based on the context data retrieved of her and other users' smartphones. Second, the CCP approach has been applied to the freely available Augsburger data set. Finally, CCP has been utilised on a synthetic data set, retrieved by extracting the location data of six different characters controlled by users, applying the game Quake III Arena. In all experiments, CCP has been evaluated against state-of-the-art context predictors. The promising results obtained by the CCP approach show, that it is able to provide predictions even if the user whose context has to be predicted does not provide sufficient contexts. Furthermore, CCP achieved quite accurate prediction results.

A major concern in context prediction is the fact that used context data is mostly highly personal. Hence, the assumption can be made that it might be difficult to utilise context prediction approaches in real world services or products, in everyday use. On the one hand personal context data are unobtrusively collected form sensors in the ubiquitous environment and mostly stored and processed externally by a service provider. On the other hand the data, e.g. collected by a user's smartphone have to be transferred to an external server that provides the context prediction. In both cases, the user looses control of his data. To identify legal problems of the context prediction process, it has been applied to the principles of the right to informational self-determination. Altogether, these principles can be summarised by the avoidance of building a profile of a user, by providing transparency to the user, by providing the possibility of giving consent by the user, considering the necessity of data and by giving information about the parties that are responsible for the data collection process. From the principles legal evaluation criteria have been derived. The criteria have been used to legally assess different prediction algorithms. The evaluation results showed that Bayesian networks and the Tree-based classifier satisfy the legal criteria the most. Due to its collaborative character CCP often disregards the derived legal criteria. To bring collaborative-based and non collaborative-based context predictors in line with the right to informational self-determination KORA has been used. Applying the KORA method, concrete technical design proposals to enable context prediction processes to be more legally compatible by design, have been inferred.

In Chapter 5 a realistic and collaborative use case, the protection of pedestrians, has been used to demonstrate the practical usefulness of CCP. In order to provide a possibility to reduce accidents between cars and pedestrians, CCP has been applied to proactively filter endangered pedestrians out of potentially many. Endangered pedestrians are those, whose next step brings them close to the street. Hence, it might be possible to provide a collision avoidance system with an additional time advantage. To evaluate the prediction accuracy of CCP, simulated and realistic movement data have been used to forecast a pedestrian's next step. To derive realistic context data of pedestrians, their movement patterns have been extracted from low-level contexts such as acceleration or orientation in 3D, which have been received by smartphones. Using the simulated and the realistic context data, CCP obtained the most accurate prediction results. Further, the evaluation results have shown that the accuracy increased almost continuously, while using additional context histories of pedestrians.

The final Chapter outlines an approach for distributed and collaborative context prediction. The aim of this approach is to enable collaborative-based context prediction to be used in real world applications. This is received by regarding the design proposals derived by the KORA method to bring collaborative-based context prediction in line with the right to informational self-determination.

Hence, a system model that describes the technical design proposals in more detail has been presented first. Afterwards, requirements such as the processing of context data directly on a user's smartphone, the fact that contexts have to be solely stored on a user's smartphone, the integration of context histories of user's that show sufficient similarities in the prediction process, the pseudonymised transfer of context data and the transmission of context data without using a central server unit, have been derived.

The requirements have been implemented by developing two architectures. The Context Recognition Architecture, which describes the automatic receiving and processing of low-level context data using the built-in sensors of a user's smartphone. The P2P-based Context Prediction Architecture, which describes the P2P-based context prediction process that is directly executed on users' smartphones. Experiments to determine the prediction time needed by different prediction approaches using Wi-Fi and UMTS connectivity have also been outlined.

The results look quite promising and serve as evidence that collaborative-based context prediction using the proposed architectures to collect and predict context data is feasible. Finally, it has been outlined that the proposed distributed and collaborative context prediction approach satisfies the principles of the right to informational self-determination and therefore can be considered as a first step to provide context prediction in a way that it is legally acceptable for the users.

7.2 Outlook

The thesis shows the applicability of collaborative-based context prediction in ubiquitous computing systems, utilising the proposed CCP approach. Furthermore, the thesis outlines a possibility to bring collaborative-based context prediction in line with the right to informational self-determination and therefore provides context prediction in a way that it is legally acceptable for the users. Nevertheless, there are additional interesting research questions that have been not addressed in this phd thesis so far. In the following, they are summarised.

First, the runtime of CCP needed to make its prediction on a PC but especially on smartphone-based devices, needs to be reduced. Although the algorithm introduced in Section 3.2.4 presents a first opportunity, the needed prediction times of CCP compared to, e.g. the SatePredictor are quite high. A possibility to reduce the runtime can be a distributed calculation of the CCP approach.

Second, with regard to the distributed and collaborative context prediction approach presented in Chapter 6, a real distributed calculation of the prediction model of the CCP approach would avoid the necessity to transfer the pseudonymised histories of the other users to the device of the user, which calculates the collaborative-based prediction model.

Third, context prediction as well as collaborative-based context prediction cannot be considered to be a native task on a user's smartphone. For this reason, their affect on the battery of the smartphone has to be investigated.

Fourth, investigations on the scalability of the CCP approach are still missing and have to be performed to provide a reliable statement of how many histories of different users can be integrated in the prediction process at most.

Finally, it would be interesting to see how the proposed distributed and collaborative prediction approach effects a user's trust and privacy concerns in reality. This can, e.g. be achieved by performing a representative user survey.