

# Uncertainty in Context-Aware Systems: a Case Study for Intelligent Environments

Leandro O. Freitas<sup>1</sup>, Pedro Rangel Henriques<sup>1</sup> and Paulo Novais<sup>1</sup>

<sup>1</sup>ALGORITMI Centre, Department of Informatics, University of Minho, 4710-057, Campus de Gualtar, Braga, Portugal  
leanfrts@gmail.com, {prh, pjon}@di.uminho.pt

Keywords: intelligent environments, context awareness, uncertainty handling

Abstract: Data used by context-aware systems is naturally incomplete and not always reflects real situations. The dynamic nature of intelligent environments leads to the need of analysing and handling uncertain information. Users can change their acting patterns within a short space of time. This paper presents a case study for a better understanding of concepts related to context awareness and the problem of dealing with inaccurate data. Through the analysis of identification of elements that results in the construction of unreliable contexts, it is aimed to identify patterns to minimize incompleteness. Thus, it will be possible to deal with flaws caused by undesired execution of applications.

## Acknowledgements

This work has been supported by COMPETE: POCI-01-0145-FEDER-0070/43 and FCT – Fundação para a Ciência e Tecnologia within the Project Scope UID/CEC/00319/2013.

## 1 Introduction

Intelligent Environments aim the development of systems capable of observe, acquire and use data from the environment and from users, seeking to improve the experience among them. These environments learn by observing the user and their apprenticeship allows the deployment of dynamic applications (?). They should have context-aware systems with the ability of learning and adapting to users needs and from the natural evolution of their preferences or desires. Context-aware systems use events to evidence changes in the environment. When new states of entities are identified, new events are created. Actions are defined as response to events and used to characterize the behaviour of applications (?).

Information in intelligent environment is naturally incomplete and, many times, ambiguous. Incompleteness leads to uncertainty, which can be defined as any partial, dubious, vague, outdated or nebulous data that is used by the system in its knowledge base (?). The use of such data results in service orchestrations that instead of assist users, may affect them negatively.

This work presents a case study to be applied in the elucidation of concepts as context awareness and uncertainty in intelligent environments. Section ?? presents the case study; section ?? discusses context-aware computing, including acquisition of data, modelling and processing. Section ?? addresses the uncertainty. Section ?? presents the future directions of research. The case study is applied along the paper to illustrate the concepts covered.

## 2 Case study

John suffers from a Chronic Obstructive Pulmonary Disease (COPD) and needs to take medicine every eight hours, to control the advance of it. Occasionally, John takes an additional medication to relieve the side effects. The treatment is compromised if he forgets to take the medicine according to the schedule. John can receive the treatment at home, once his house is equipped with context-aware system, with sensors sending data for the central control station to analyse. Also, the computational structure monitors the health state of John. This includes blood pressure, heart rate, temperature, level of oxygen, among others. This data is periodically processed and used to update John's medical record. Thus, his physician can analyse John's progression from his office. If the physician finds a change in John's health, he can take a proper action, like, modify the treatment, request

medical exams or get in contact with John or his family.

The system can communicate with devices like mobile phones and TVs, through which, it can trigger alarms and interact with John. When he takes a medicine, he must inform the system. If he forgets, a reminder is presented in an intelligent device next to him. For this to happen, the system must be able to acquire data from the environment, for instance, time of the day and compare with information stored in the knowledge base. If the environment's time is the same of the hour that he should take the medicine, it shows the reminder. The system knows the available devices and the resources they have. With this, the system decides which one supports the reminder, considering John's location.

When John leaves the house in a period of time that is very close to the hour of his medication, the system can deduce that he will not be able to return to take it and so he will compromise his treatment. In this case, the virtual assistant installed in his smart phone presents a notification suggesting him to keep the medication with him. Thus, he will be able to take it in the correct time. The virtual assistant can also remind him at the correct moment.

The system must also be aware of John's current activity. According to this information, it will be possible to decide what is the best way to advise him. For instance, if John is watching a movie, the system can put the video on pause and show the reminder on the TV. In another situation, if John is on the phone, the system can send a low sound alert to the phone in a discreet way in order to not disturb the conversation, but yet, warning him about the hour.

Besides monitoring the administration of medication, the system can be able to adjust the environment according to John's preferences, previously defined. For instance, adjust the temperature of the room, start ambient sound or music when a visitor arrives or even present information that interests the user (news on the TV or smart phone) and send reminders about events of his agenda.

### 3 Context-aware computing

According to (?), context is any sort of data that can be used to characterize the current state of entities (emotional state of users, focus on tasks, location or orientation, time). However, for the development of this research, it is believed that inner state of users are not considered context, but the result of one. For instance, the emotional state of a user is result of what happens to him. Context of entities is anything exter-

nal to it that can be used to influence its behaviour.

Context awareness refers to the identification of states of users and their surrounding entities to influence the behaviour of applications, providing a better adaptation to each scenario. Inference rules process context data considering the semantics of relations between entities. They allow the deduction of new knowledge. In the case study, a inference rule is applied when John leaves the house right before the time of his medication. The system cannot know for how long John will be out. However, based on information about the time he usually takes when goes out, it assumes that he will be (or not) at home on time. Thus, the system suggests him to keep the medicine before leave.

Applications of context-aware computing take advantage of user's environmental data, such as his location and current activity, time of the day, other users and intelligent devices. In the case study, this is evidenced when the system interacts with John through a device. For instance, when it pauses the movie to show a notification about the time of the medicine, or when the virtual assistant suggests him to grab the medication before leave the house, or even when the system starts a music when John receives visit. User experience can be enhanced by enabling devices and applications to automatically sense and adapt themselves to the changes in surrounding environments. Environmental data creates context for the interaction between users and devices (?).

Context-aware applications must use models able to adapt themselves to the dynamicity of the domain without loose the power of processing. In (?), Djoudi, Bouanaka and Zeghib propose a framework to facilitate the conception of context-aware systems. They merged two modelling approaches, Model-Driven Engineering and Formal Methods. A transformation technique with bidirectional mapping was used to merge them. According to the authors, the specifications generated are executable, which allows the validation of systems developed with this framework. The framework is validated through a case study.

#### 3.1 Context acquisition and modelling

Daily tasks may have similar steps even if they have completely different goals. It is difficult to determine what the user is trying to accomplish. Applications should use explicit and implicit input of data allowing them to have levels of awareness (?). Explicit inputs refer to any manual insertion of data. Implicit inputs refers to automatic acquisition, through sensors and result of processing. In reactive actuation, the system

makes requests to the sensor. It characterizes isolated events. The acquisition must be done at the time it occurs. In proactive actuation, sensors send data to the system every time they detect something. It characterizes events that tend to repeat within a period of time (?).

Context-aware applications are distributed once they use data from different sources (?). Modelling context involves aspects like heterogeneity of entities, mobility of users, relations among entities, periodic events, uncertainty, reasoning, usability and efficiency (?). One of the main issues on context modelling refers to the subjectivity. The same set of data may be valid in one situation, but not in another. Thus, the more complete is the data about a context, the more accurate will be the model built based on that. This subjectivity leads to the necessity of handling uncertainty. A context-aware system should be able of obtaining enough data to identify clearly all information present in the environment.

### 3.2 Context processing

Machine learning algorithms allow the creation of association rules. Based on the analysis of behaviour of individuals, it is possible to infer action patterns. This happens, in the case study, when John takes an additional medicine to assuage the side effects of the treatment. He prefers to take it at night because it helps him to sleep. By analysing the dataset with information about patterns behaviour of the patient, the system processes the rule  $P(C, T | S)$ , where C represents the controlled medicine, T is the time of the day and S is the drug for side effects. If, in one given situation, the values applied for the rule are  $P(C=Yes, T=Night | S=Yes [0.7, 0.85])$ , it means that, there is a support of 70% for the relation between controlled medicine with the side effects in the period of night. For this rule, the reliability level is 85%. This means that there is a high probability of the patient taking the controlled medicine and also one for side effects, in the period of night. So, if the user takes only the controlled medicine, the system suggests him to take the side effects medication as well.

Machine learning deals with Binary Classification where one of two values is associated to a pattern (e.g. 0 or 1) and Multiclass with a bigger range (?). This is applied to the case study when John decides to leave the house close to the hour his medication. The probability of not returning on time can be categorized in Low, Medium or High. These levels can have more values, or only two, if using Binary Classification. The system identified the following pattern, after analyse previous situations where the patient left

the house: 2 or more hours before the medication time – *low* probability of not returning on time (e.g. up to 35%); from 1h59m to 1 hour before the medication time – *medium* probability of not returning on time (e.g. 35.01% to 70%); less than 1 hour before the medication time – *high* probability of not returning on time (more than 70%).

Thus, it is possible to create an association rule to classify the risk of John not return on time:  $P(R | H,M)$ , where R is the risk value, H is the hour John left home and M is the time of his next medication. Thus, if he has to take the medication at 10 o'clock and leaves the house at 8 o'clock, based on the rule values, he has a low probability of missing medication time:  $P(R=0 | H=8,M=10)=0.35$  (35%). If he leaves at 8:30, he has a medium probability of miss it:  $P(R=1 | H=8:30,M=10)=0.65$  (65%). If he leaves at 9:15, he has a high probability of miss the time:  $P(R=2 | H=9:15,M=10)=0.85$  (85%). With data from past situations, it is possible to predict the probability of patient's returning time, assuming that if it was true in the past, in the future it will be true (?).

## 4 Uncertainty

For the scope of this research it was adapted the concept proposed by (?), assuming that **uncertainty** is the result of incomplete, contradictory or outdated information. In other words, incomplete data may lead to uncertainty.

Aspects related to the state of mind of users can influence them causing, changes in their acting patterns within a short time interval. In (?), Bobek and Nalepa state that mobile context-aware systems deal with a significant quantity of data, while this can change fast in a small period of time, becoming outdated and, consequently, useless for building context-aware models. They propose four requirements that should be taken into account. The first is Intelligibility, where the user should be aware of what the system does and change it, if necessary. The second is Robustness. The dynamic nature of a context-aware system, it should be able to adapt itself to different environments. The third is Privacy, where all data from users should be secure and accessible only by authorized people. The last is the Efficiency, in which mobile context-aware systems should use the available resources with good responsiveness. A project named KnowMe<sup>1</sup> is presented to validate the proposal, describing all the phases of the development.

Considering the scope of this research, there are

<sup>1</sup><https://geist.re/pub/projects:knowme:start>

two main types of uncertainty in context-aware systems, from the probabilistic perspective. **Aleatory uncertainty** refers to hardware, energy and communication failures. These problems may result in a random behaviour of the system. It is related to problems of interpretation of sensor signs and statistics variability. **Epistemic uncertainty** is related to the level of knowledge about the domain, including problems of reasoning. The more information the system has access to, the lower is the level of epistemic uncertainty (?).

Aleatory uncertainty may lead to epistemic uncertainty. In the case study, this can be seen if one of the sensors that monitors the vital signs of John starts to behave unexpectedly. If this is not identified soon, this sensor will send outdated data or within a higher range of time than it should. Thus, when John's physician analyses his medical record he would be using inappropriate data to decide the treatment. In this scenario, problems of epistemic uncertainty can be evidenced when the system starts to deal with outdated data. Besides having a well-defined model to represent the context, the system do not have access to updated data due to communication problems with the sensors. The level of knowledge it is not enough for the system to work properly. Epistemic uncertainty tends to be reduced along the time through system's learning. The reduction of epistemic uncertainty helps dealing with aleatory type. When the user interacts with applications, he is helping the system to learn and consequently, he is helping the reduction of epistemic uncertainty (?).

#### 4.1 Quality of context

**Quality of Context(QoC)** is any information about the quality or reliability of context data (?). This includes the level of accuracy and completeness of how the data describes an entity. In the case study, this is evidenced when the user is walking inside his house and passes by the front door. Depending on the position of the sensor that monitors John's movements, the system may infer that he is leaving the house. If this happens close to the hour where John should take his medication, the system will send a notification to his mobile phone suggesting him to grab the medicine before leave. Instead of helping, the system would be disturbing the user with unnecessary notifications. The QoC lies in the fact that despite of being near the front door, the user was not intending to leave. To increase the QoC, other variables should be monitored and an analysis of the semantics of the situation should be performed before trigger any application.

#### 4.2 Presentation of uncertainty

Real world environments are naturally uncertain due to their complexity. Consequently, context-aware system have to deal with incomplete scenarios. The more accurate is the abstraction of the world, the lower are the gaps of data. Thus, uncertainty may have different levels, which are related to how much it is known about a context or how much data sent by sensors the system is able to analyse and transform in relevant information. Depending on the level and type of uncertainty, a good approach is to ask the user to help the system solve gaps of data. In the scenario presented in subsection ??, where the system misinterpreted the fact that the user was not leaving the house, it would be less intrusive if the system had requested for more information to build a more complete context. Instead of suggesting the user to grab the medication before live, it could present a message asking if he was intending to leave. Another alternative is to use an sensor outside the front door and send the alarm only when it identifies John.

When the level of certainty in a context is high, the level of uncertainty can help to speed the performance of tasks if displayed to the user to analyse. In high or medium levels of certainty, the users still can improve the performance of tasks. If the certainty is low, the interaction with the user is not recommended, once he would have to analyse much more data to prepare the system (?).

### 5 Conclusion

This paper introduced a case study to help on the design and proposal of approaches to handle successfully concepts like context-aware systems with uncertainty. The analysis of the case study helped the description of aspects for the identification of important elements when designing context-aware systems, which is the main contribution of this work. The problem of uncertainty must be addressed because intelligent environments tend to present high level of dynamism, with rich semantics connecting entities. Considering that even when a context model has a high level of certainty and low level of uncertainty, the unknown aspects may involve entities with higher level of importance and they should be dealt with more priority. The next step of the project is to propose a model to classify levels of uncertainty based on the impact to the environment. It is intended to analyse aspects of incompleteness in Intelligent Environments, originated by problems of context data, such as, capability of interpretation, accuracy and dy-

dynamic aspects. It is not possible to have an Intelligent Environment without a degree of uncertainty. However, it is believed that, by analysing the sources that generate incomplete data, it is possible to handle with uncertainty.

## REFERENCES

- Perera, C., Zaslavsky, A., et al.: Context Aware Computing for The Internet of Things: A Survey. in *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 414-454, (2014). doi: 10.1109/SURV.2013.042313.00197
- Bobek, S., Nalepa, G.J.: Uncertain context data management in dynamic mobile environments. *Future Generation Computer Systems*, 66:110–124.(2017)
- Alpaydin E (2010) *Introduction to Machine Learning*. The MIT Press, 2nd edition. ISBN 026201243X, 9780262012430.
- Buchholz, T., Schiffers, M.: Quality of context: What it is and why we need it. In *Proceedings of the 10th Workshop of the OpenView University Association*. (2003)
- Dey, A.K.: Context-aware computing: The cyberdesk project. In *AAAI Spring Symposium on Intelligent Environments*, pages 51–54, Palo Alto. AAAI Press. (1998)
- Djoudi, B., Bouanaka, C., Zeghib, N.: A formal framework for context-aware systems specification and verification. *Journal of Systems and Software*, 122:445 – 462.(2016)
- Musumba, G., Nyongesa, H.: Context awareness in mobile computing: A review. *International Journal of Machine Learning and Applications*. Vol 2, No 1. (2013)
- Novais, P., Carneiro, D.: The role of non-intrusive approaches in the development of people-aware systems. *Progress in AI*, 5(3).doi: 10.1007/s13748-016-0085-1 (2016)
- Nugroho, L.E.: Context-awareness: Connecting computing with its environment. In *2015 2nd International Conference on Information Technology, Computer, and Electrical Engineering*, pages 3–7. doi: 10.1109/ICITACEE.2015.7437760 (2015)
- Rukzio, E., Hamard, J., et al.: Visualization of uncertainty in context aware mobile applications. In *8th Conference on Human-computer Interaction with Mobile Devices and Services*, 247–250. doi: 10.1145/1152215.1152267 (2006)
- Senge, R., Bösner, et al. Reliable classification: Learning classifiers that distinguish aleatoric and epistemic uncertainty. *Inf. Sci.*, 255:16–29. doi: 10.1016/j.ins.2013.07.030(2014)
- Sheikh, K., Wegdam, M., Sinderen, Mv.: Quality-of-context and its use for protecting privacy in context aware systems. *Journal of software* 3(3):83–93. (2008)
- Smola, A., Vishwanathan, S.V.N.: *Introduction to Machine Learning*. Cambridge University press, 1nd edition. ISBN 0521825830. (2008)
- Yaghlane, A.B., Denoeux, T., Mellouli, K.: Uncertainty and intelligent information systems. In *Elicitation of Expert Opinions for Constructing Belief Functions*,75–89. World Scientific. (2008)