A MULTILAYERED UNCERTAINTY MODEL FOR CONTEXT AWARE SYSTEMS[1]

Susan McKeever, Juan Ye, Lorcan Coyle and Simon Dobson [2]

Abstract

Context-aware systems typically use data sensed from the environment to drive adaptive behaviour. Sensed data is inherently imprecise and uncertain; in addition, new uncertainties are introduced when sensed data is fused with other data to infer context at the more abstract level of situations. We present an uncertainty model which aggregates context uncertainty and provides mechanisms to capture uncertainty at situation level. We demonstrate the application of the model in a sample scenario using an experimental data set.

1. Introduction

Pervasive systems frequently rely on context information sensed from the environment to determine adaptive behaviour. Sensors suffer from technical limitations so their sensed data will only be an approximation to real values. Other context information quality problems can also arise, such as the failure of users to carry their locator tags in a location-aware system. Failure to make appropriate assumptions about the quality of context information can lead to incorrect behaviour by adaptive applications [3]. In order to address this problem, uncertainty models have been proposed that typically capture some measures of confidence about context, and a decay parameter to indicate the useful lifetime of the data [2, 5]. Our model seeks to improve the capture of uncertainty of context information in three ways: (1) by producing a model of uncertainty with specific metrics for each *layer* of context information; (2) by providing a model that is not limited to location based context (3) by aggregating uncertainty from sensor through to *situation* level. This is becoming increasingly important as context aware applications transition to using the more abstracted situation instead of discrete context facts to determine context aware behaviour.

2. Related Work

We use Henricksen and Indulska's classification of context information (sensed, static, profiled, derived [3]) to assist in identifying uncertainty sources. Ranganathan *et al.* incorporate calculations of uncertainty into their sensor fusion algorithm [8]. We use a similar approach when deriving uncertainty of location data but we use a layered model of uncertainty that aggregates uncertainty from source data up to situation level. In addition, the model is not confined to location-aware systems.

---

[1]This work is partially supported by Enterprise Ireland under grant number CFTD 2005 INF 217a, "Platform for User-Centred Design and Evaluation of Context-Aware Services" and by Science Foundation Ireland under grant numbers 05/RFP/CMS0062 "Towards a semantics of pervasive computing" and 04/RPI/1544 "Secure and predictable pervasive computing".

[2]Susan.mckeever@ucd.ie, Systems Research Group, School of Computer Science and Informatics, Complex and Adaptive Systems Laboratory, University College Dublin, Belfield Dublin 4. Ireland

3. Types of uncertainty

If the source data used to identify context is uncertain, then the inferred context at higher level should preserve this uncertainty [4]. Our model for uncertainty aggregation is shown in Figure 1. At the bottom of the context ladder, *source data* consists of the sensed and other environmental data that are used to derive context facts. These are grouped in our model into two types: physical sensor readings such as location sensors and temperature sensors and profiled data such as user calendar data. The uncertainty metrics for source data will depend on the nature of the errors that can occur. Typical measures for location systems, for example, are precision and accuracy. Electronic calendar data, on the other hand, may be erroneous if not kept up to date by the user and we measure this using association confidence (see Section 4). Static data such as room layouts of a building are assumed to be correct and thus have no uncertainty measures.

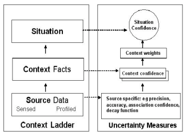

Figure 1. Context ladder with uncertainty measures for each layer

Moving up the context ladder, *context facts* (e.g., Bob located in meeting room) are derived from source data. Each fact has a measurable context confidence that is derived from the underlying source data uncertainties. *Situations* are created by combining context facts e.g., Context fact 'Bob located in meeting room' is one of the context facts that contribute to the Meeting situation. Situation confidence is calculated from the underlying context facts by combining the confidence of context facts with appropriate *context weights*, resulting in a single measure of situation confidence. A context weight is set for each context fact. Its purpose is to quantify a context fact's contribution to the occurrence of the situation, relative to the other context facts of the situation. The greater the importance of a context fact to a situation, the higher its associated weighting. An example of how context weights are used is shown in Section 4. The situation confidence quantifies the extent to which the situation is believed to be occurring at a particular point in time. The measure of confidence at situation level is important because situation abstractions will be used by applications directly.

4. Uncertainty Model Demonstration

To demonstrate our uncertainty model, we tracked the movements of a research student over five working days using three sources as shown in Table 1: A Ubisense tag based sensor location system situated in our offices, a keyboard logger agent which tracks user activity on their keyboard and electronic calendar entries. Our sample scenario follows the Meeting situation, which is deduced from three context facts as shown in Table 1. We use the source data and their associated uncertainty metrics to demonstrate how context and situation confidences can be calculated.

Source Data: Table 1 shows the uncertainty metrics identified for each of our three data sources. *Association* confidence indicates the probability that a tracked object (e.g., person) is physically located with the tracked source [6] e.g., calendar entry location, RFID tag. We use association confidence in our context confidence calculations if conflicting facts exist when fusing source data. Decay function for a data source identifies the nature and speed of decay of data over time for that source. E.g., Location readings for a moving user will be subject to rapid decay.

Context Fact	Weight	Source Data	Source uncertainty metrics
1. User in meeting room	3	Ubisense	Precision, accuracy, association confidence, decay function
2. Keyboard not active	1	Keyboard agent	Association confidence, decay function
3. Meeting scheduled	1	User calendar	Association confidence, decay function

Table 1. Context facts and source data for Meeting situation

Context facts: Moving up the context ladder, the confidence level of context information is derived using the underlying uncertainties of the source data. Each context fact has an attached confidence measure ranging from 0 to 1. For the Meeting situation, the derivation of confidence for each of the underlying contexts is as follows:

`1. User in meeting room`: The Ubisense tag reading is abstracted to room-level location by using the precision measure of Ubisense to identify a *precision circle* around the tag reading in which the reading occurs, similar in approach to Ranganathan *et al.* [8]. Accuracy, α, of the Ubisense system represents the probability that the a tag reading is correct, for the given precision. The calculation of context confidence γ is based upon the percentage of the precision circle, ω, that falls within the meeting room, modified for Ubisense accuracy: $\gamma = \omega \times \alpha$
`2. Keyboard not active`: The output of the decay functions determines the extent to which the keyboard is considered inactive. Sampling frequency can be used to determine the time threshold below which the keyboard is considered fully active (e.g., 30 seconds).
`3. Meeting scheduled`: To calculate confidence level, γ, a decay function for this diary source is applied as a function that is highest in value at the middle of the meeting and lower at meeting start/end times.

Meeting Situation confidence: Various methods may be used to infer the confidence of a situation from its constituent contexts, including Bayesian networks [9, 7] and probabilistic logic [7]. We combine probabilistic logic with context weights, where these weights reflect the relative importance of each context to the overall situation occurrence. The context weights for the Meeting situation are set at 3:1:1, reflecting the fact that the location of the user in a meeting room has a weighting of three times the impact of the other two constituent contexts when determining if a meeting is occurring. The calculation of the Meeting situation confidence, σ using context weights β_1, β_2, β_3 against each of our three context confidences, γ_1, γ_2, γ_3 is:

$$\sigma = \frac{\sum\limits_{1 \leq i \leq 3} \gamma_i \times \beta_i}{\sum\limits_{1 \leq i \leq 3} \beta_i} \tag{1}$$

Using context weights introduces an additional dimension to situation uncertainty measurement. Determining the value of their contribution to situation uncertainty calculation versus the contribution of individual source data uncertainties will form part of our future work.

5. Conclusions and Future Work

We have presented a model of context uncertainty that allows for aggregation of uncertainty from source data through to situation level, using a sample scenario from our experimental data set. We also used context weights to reflect the relative contribution of individual context facts to situation confidence. Next, we will use our experimental data set to populate values for source data uncertainty metrics, produce actual situation confidence measures and examine how individual source data uncertainties and context weights can be tuned to assist in situation identification. As part of this, we will use the output of our experimental measures of Ubisense uncertainties [1] to improve our data set. We will then compare this uncertainty model with our work on situation identification using Bayesian networks with situation lattices [9] to determine whether a combination of the two improves the identification of situations and their confidence levels.

References

[1] Lorcan Coyle, Juan Ye, Emerson Loureiro, Stephen Knox, Simon Dobson, and Paddy Nixon. A proposed approach to evaluate the accuracy of tag-based location systems. In *Ubiquitous Systems Evaluation (USE 2007). Workshop at UbiComp2007*, pages 292–296, 2007.

[2] Simon Dobson, Lorcan Coyle, and Paddy Nixon. Hybridising events and knowledge as a basis for building autonomic systems. *IEEE TCAAS Letters*, 2007. To appear.

[3] Karen Henricksen and Jadwiga Indulska. Modelling and using imperfect context information. In *PERCOMW '04: Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, page 33. IEEE Computer Society, 2004.

[4] Jeffrey Hightower, Barry Brumitt, and Gaetano Borriello. The location stack: A layered model for location in ubiquitous computing. In *WMCSA '02: Proceedings of the Fourth IEEE Workshop on Mobile Computing Systems and Applications*, page 22. IEEE Computer Society, 2002.

[5] Glenn Judd and Peter Steenkiste. Providing contextual information to pervasive computing applications. In *PERCOM '03: Proceedings of the First IEEE International Conference on Pervasive Computing and Communications*, page 133. IEEE Computer Society, 2003.

[6] Jussi Myllymaki and Stefan Edlund. Location aggregation from multiple sources. In *MDM '02: Proceedings of the Third International Conference on Mobile Data Management*, pages 131–138. IEEE Computer Society, 2002.

[7] Anand Ranganathan, Jalal Al-Muhtadi, and Roy H. Campbell. Reasoning about uncertain contexts in pervasive computing environments. *IEEE Pervasive Computing*, 3(2):62–70, 2004.

[8] Anand Ranganathan, Jalal Al-Muhtadi, Shiva Chetan, Roy Campbell, and M. Dennis Mickunas. Middlewhere: a middleware for location awareness in ubiquitous computing applications. In *Middleware '04: Proceedings of the 5th ACM/IFIP/USENIX international conference on Middleware*, pages 397–416. Springer-Verlag New York, Inc., 2004.

[9] Juan Ye, Lorcan Coyle, Simon Dobson, and Paddy Nixon. Representing and manipulating situation hierarchies using situation lattices. *Submitted to Revue d'Intelligence Artificielle*, 2007.