Estimating User Preferences by Managing Contextual History in Context Aware Systems

Nazir A Malik, Muhammad Younus Javed
Department of Computer Engineering, College of Electrical and Mechanical Engineering, National University of Sciences and Technology, Pakistan
Email: {nazirmalik, myjaved}@ceme.edu.pk

Umar Mahmud
Department of Computer Science, College of Signals, National University of Sciences and Technology, Pakistan
Email: umar_mahmud@mcs.edu.pk

***Abstract*— Context awareness enables smart service discovery and adaptation for mobile and wireless hosts. The contextual data is acquired from sensors present in the smart space, which may be absent. The inherent noisy nature of wireless environments does not guarantee the gathering of correct data. A history module is thus required in conjunction with existing context-aware systems that overcomes these limitations by predicting the data. We present a modular approach that when coupled with existing context managers will be able to provide user preference on the basis of usage history.**

***Index Terms*— Context History, Context Awareness, Ubiquitous Computing**

I. Introduction

With the huge increase in development of mobile and wireless devices and their popularity in everyday use, Weiser's dream of pervasive computing is rapidly becoming true [1]. Latest research includes development of smart environments that are capable of performing intelligent tasks on behalf of the user. This skill is termed as context-awareness [2]. Context-Awareness is the knowledge of the surrounding conditions including the environment, the activity, the people, the devices and the applications [3]. Context-awareness is the prime issue of pervasive environments which strives to provide smart and efficient discovery and subsequent delivery of the services to the mobile and often naïve user [4]. Users are no longer restrained by tedious and manual service discovery and integration issues. The mobility and the nature of the user being volatile demands a mechanism that not only predicts the preferences of the user's but also adapts itself as anomalies occur. In addition to preference forecast, the history information can also identify missing contextual data and provides a means to purge the history while maintain its effect for future calculations. Furthermore, this history information is private data and must be kept secure, thus highlighting the need for security. The following paper proposes a history management system that provides both preference estimation as well as data prediction while ensuring privacy of data and control of access.

The remainder of this paper is structured as follows. Section II gives a comparison of different research efforts carried out over the years in context awareness with emphasis on history management. Section III discusses the various components of a context aware system while Section IV gives a detail of a context history manager. Section V describes results of a test case. Section VI concludes with future work.

II. Related Research

The pioneering context-aware systems like Active Badge [5] and Xerox PARC [2] were basically location-aware systems as they were only aware of the locality. Tour guide systems like GUIDE [6] and Cyberguide [7] evolved the concept of context by adding temporal information in addition to spatial information. These systems are primarily context-aware application designed to provide better and customized services to their users. The Context Toolkit [8] provides an Application Programming Interface (API) to develop context-aware applications but is limited to tightly coupled 'Widgets' that directly access the hardware contextual data sensors. The recent context-aware systems are frameworks that provide context-awareness through rich ontology based context representation. This rich context ontology considers all parameters relevant to an interaction as the context. Gaia [9] (a CORBA based distributed operating system) and CAMUS [10] (a JINI based service oriented framework) provides context-aware service delivery limited only to context-aware applications. CoBrA [11] is a mobile agent based framework that dispatches mobile agents to gather context information from the sensors in the environment. CAPEUS [12] uses a document based approach that exchanges context-aware packets that describe service requests. CAPP [4] is a service oriented architecture that provides context-aware service discovery for mobile users.

The history of information is maintained as a contextual database in SOCAM [13], Gaia, CASS [14] and CoBrA. CAML has been proposed to work in mobile pedagogical environments and highlight that context adaptation should be the research issue rather than

© 2009 ACADEMY PUBLISHER

context awareness [15]. SODA is a decentralized system designed as a Service Oriented Architecture (SOA) [16]. Context Management Framework (CMF) and Hydrogen both lack history of information as well as an Ontology based information representation technique [17 and 18]. CAPP is also a centralized architecture but lacks both history of information and conflict resolution [4]. System lacking a history of interaction like CAPP, CAMUS and CMF are limited to the availability of sensors and may result in unwanted inferences on the basis of vague data. A feature comparison of different context aware systems based on Rich context, use of Ontology, decentralized architecture and History support is shown in Table I.

TABLE I. COMPARISON OF CONTEXT AWARE SYSTEMS

	Rich Context	Decentralized	Ontology	Inference Support	History of Information
Active Badge and Xerox PARC (1992)	×	×	×	×	×
Cyberguide and GUIDE (2000)	×	×	×	×	×
Context Toolkit(1999)	✓	×	×	✓	×
CAPEUS	✓	✓	×	✓	×
GAIA OS (2002)	✓	✓	✓	✓	✓
CMF (2003)	✓	✓	×	✓	×
CoBrA (2003)	✓	✓	✓	✓	✓
Hydrogen (2002)	✓	×	×	✓	×
SOCAM (2004)	✓	×	✓	✓	✓
CASS (2004)	✓	×	×	✓	✓
CAML (2004)	✓	×	×	✓	×
CAMUS (2005)	✓	✓	×	✓	×
SODA (2006)	✓	✓	×	×	×
CAPP (2006)	✓	×	✓	✓	×

III. CONTEXT AWARE SYSTEMS

The development of smart devices and their subsequent popularity in everyday life has opened new horizons in the pervasive world. Users can now access legacy applications through these devices as well share information with its peers. The need of a distributed, intelligent and adaptable context aware system is felt that facilitates resource discovery and adaptation, as indicated by the user demand. This system is able to identify and examine the user's context as well as the service's context and derive implications. These implications then help to decide how best to adapt the services for the user. A history associated with each user predicts the user's preferences for different interactions. A distributed model of a context aware system that adapts services on the basis of the inferred contextual information coupled with the history of the user is thus envisaged. Furthermore, this module resolves conflicts in interactions by providing a scheduler in the system. Fig. 1 shows a schematic view of the proposed system. This model has the features as described in the following sub-sections.

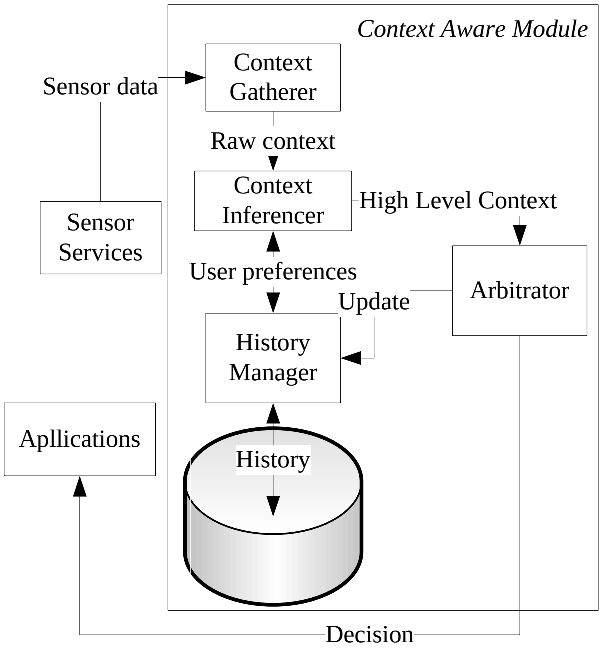

Figure 1. Context aware system

A. Context Gathering

The raw contextual data from the sensors present in the environment is gathered. The gathered context is stored as well as exchanged using a common platform independent standard. This context is then stored in the form of Ontology. The most common ontology is the Web Ontology Language (OWL) [19].

B. Context Inference and Adaptation

The raw data is then used to infer high-level meaningful context of the user as well as the services. The high-level context is then used to select the appropriate service and adapt the service to the user's needs. Among the different techniques available most of the systems are designed as rule based systems [4, 11, 13, 14 and 20]. The same task has been designed as a classification problem solved using various supervised learning techniques that follow the k-Means approach [17, 21, 22, 23 and 24].

© 2009 ACADEMY PUBLISHER

C. History of Information

A history of information is maintained in the system that predicts the user's preferences while effect of old interactions tied with the degree of satisfaction of the user is also preserved. The history is flexible to the occurrence of irregularities in the interactions.

D. Arbitrator

Finally, the arbitrator invokes the applications based on the decisions taken by the context manager. The updates of invocations are also recorded in the history.

IV. CONTEXT HISTORY MANAGER

The context history manager is responsible for storing history of interactions and inferring the predicted context as well as identifying the users' preferences. This component is designed to work in collaboration with existing context managers and hence is envisaged as a module. Fig. 2 gives a layout of the context history manager. The major modules of this context history manager are discussed in the following sub-sections.

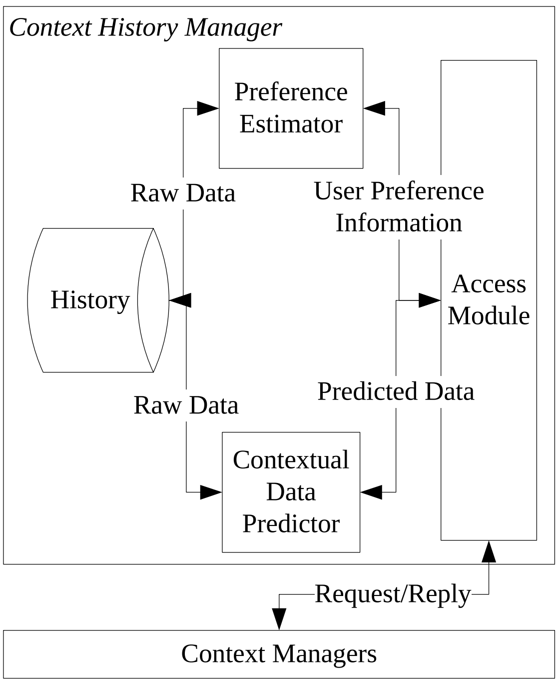

Figure 2. Context history manager

A. Preference Estimator

This module is responsible for estimating the user preferences for each type of interaction. The module analyzes the history of previous interactions, between the user and the services, to estimate the user inclinations for new interactions. This module returns the most likely service that maybe selected by the user for the given task. Our approach is to give high weights to the preference that has been selected the most number of times in previous interactions of the same type. The recommended service is the one that has been used the most and has the highest satisfaction in recent history. The suggested service maybe rejected by the user and would require recalculation of the weights identifying the level of recommendation. The user once selects service is required to display his satisfaction which is used to calculate the preference value. The degree of satisfaction is represented using different normalized values as shown in Table II. The satisfaction range is hard limited at regular intervals. Equation 1 defines the satisfaction range.

$$0 \le Satisfaction(St) \le 1 \tag{1}$$

TABLE II. SATISFACTION LEVELS

Satisfaction Level (St)	Value Range
Ultimate	>0.95
Excellent	>0.85
Above Average	>0.75
Average	>0.65
Good Enough	>0.55
Satisfactory	>0.45
Reasonable	>0.35
Fair	>0.25
Dissatisfied	>0.15
Highly dissatisfied	<0.16

Initially all the services are equally probable and given the same weights. The new weights are calculated by first depreciating the old weight according to the time elapsed and then multiplying with the satisfaction level for each service as shown in Equation 2. On use of service A, the weights of all the services are then recalculated. The services that are not used have their weights depreciated as shown in Equation 3. The depreciation occurs by dividing the old weights by number of days elapsed since last use. The new weights are normalized to be maintained between 0 and 1.

$$Weight_{New} = \left(\frac{Weight_{Old}}{TimeElapsed} \times St \right)_{Normal} \tag{2}$$

$$Weight_{New} = \left(\frac{Weight_{Old}}{TimeElapsed} \right)_{Normal} \tag{3}$$

B. Contextual Data Predictor

This module predicts contextual data on the basis of history information in the event of missing sensor values. This typically returns the most reported value by that sensor. This module is still in its infancy in our system.

C. Access Module

It is also responsible for communicating with the context managers while hiding the underlying private data of user preferences from external access. Also, this module denies access to unauthorized and untrustworthy context managers.

© 2009 ACADEMY PUBLISHER

D. History Storage

This stores actual history information preferably in encrypted form to maintain privacy. This module stores the data in the form of OWL based ontology. This allows a rich context based history to be maintained through a tagged based language.

V. TEST EVALUATION

For calculation of the preference value we have monitored the behaviour of a user in the department. The context is kept constant in which the users come into the department and use print service as desired. There are five print services in the department. These are, MSPR01, MSPR02, BEPR, FACPR and CSFAX. Table III gives a description of these services.

TABLE III. SERVICE REGISTER

ID	Type	Location
MSPR01	Laser Printer	MSLAB
MSPR02	Color Printer	MSLAB
BEPR	Laser Printer	BELAB
FACPR	Laser Printer	Office
CSFAX	Print and Fax	Office

All services have the same initial probability i.e., 0.20. When a user selects a service an entry is made for that day in the user space and after the interaction is finished the user is asked to rate the service with a satisfaction value as shown in Table I. The new weights are calculated as shown in Eq. 2 and 3. The time elapsed is simulated by calculating the number of days that has passed since last use as days elapsed. This can be modified to hours elapsed or even minutes elapsed depending on the frequency of use of a given service type in a smart space.

User was monitored for his access to different printing service and satisfaction values were recorded after each interaction. Table IV shows the collected data of user for a fortnight. For the days where the user has not used any service, weights of all the services are depreciated. It can be seen from Table IV that MSPR01 has the highest frequency of use.

TABLE IV. USAGE RECORD

Day	Time	Service Selected	Satisfaction Level (St)
1	1000	MSPR01	0.63
	1215	MSPR01	0.96
2	0900	BEPR	0.32
	0930	BEPR	0.22
	1230	MSPR02	0.88
3	-	-	-
4	1015	MSPR01	0.78
	1345	MSPR02	0.65
5	1535	FACPR	0.99
6	1020	MSPR01	0.64
	1155	MSPR01	0.53
	1300	BEPR	0.68
7	1155	FACPR	0.36
	1500	MSPR02	0.69
8	-	-	-
9	-	-	-
10	0945	FACPR	0.52
	1035	MSPR01	0.78
11	1215	FACPR	0.87
	1330	CSFAX	0.15
	1505	FACPR	0.89
12	0900	MSPR01	0.87
	0955	MSPR01	0.91
	1300	MSPR01	0.78
13	1150	BEPR	0.66
14	1250	BEPR	0.44
	1330	MSPR02	0.78
15	1530	MSPR01	0.87

The normalized weights are calculated every day for each print service as shown in Table V. Fig. 3 shows the result obtained in the form of a line chart. It can be observed from Fig. 3 that MSPR01 is predicted as the most likely service.

TABLE V. USAGE RECORD CALCULATION

MSPR01	MSPR02	BEPR	FACPR	CSFAX
0.45	0.45	0.45	0.45	0.45
0.3	0.48	0.48	0.48	0.48
0.82	0.28	0.28	0.28	0.28
0.99	0.08	0.03	0.08	0.08
0.98	0.12	0.01	0.12	0.12
0.97	0.12	0.05	0.14	0.14
0.84	0.51	0.03	0.14	0.14
0.86	0.42	0.03	0.19	0.19
1	0.03	0	0.03	0.03
0.99	0.07	0.01	0.05	0.05
0.96	0.08	0.01	0.26	0.1
1	0.02	0	0.05	0.04
1	0.01	0	0.02	0.02
1	0.01	0	0.01	0.03
1	0.01	0	0.03	0.04
1	0.04	0	0.02	0.05

© 2009 ACADEMY PUBLISHER

1	0.03	0	0.02	0.06
1	0.02	0	0.01	0.07
0.99	0.02	0	0.06	0.1
1	0	0	0	0.01
1	0	0	0.01	0
1	0	0	0	0.01
1	0.01	0	0.02	0.01
1	0	0	0	0
1	0	0	0	0
1	0	0	0	0
1	0	0	0	0
1	0	0	0	0
1	0.01	0	0	0

VI. CONCLUSION AND FUTURE WORK

History management in context aware systems provides a means to predict user future choice in service arbitration. A history management module is proposed that estimate the user's choice on the basis of usage history as well as the satisfaction level of the user. In effect it can be concluded that the most recently used service as well as the most satisfying service is the most likely service. It can also be seen that a service that is regarded as more satisfying than the rest of the services, if commits a fault that reduces the services current satisfaction level will reflect a less reduction in weight if it has shown high satisfaction in its old interactions.

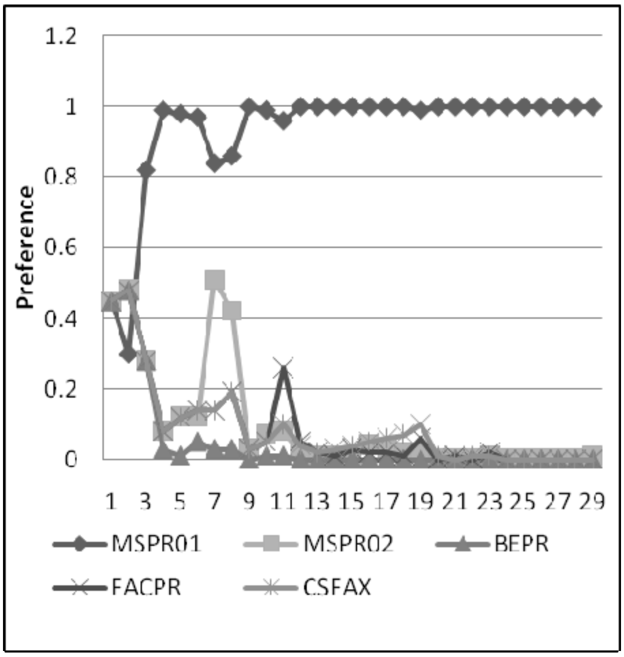

Figure 3. Context history manager

Our test case has shown that the system recommends the service that has a high satisfaction level in its recent history of use. The test results though promising have to be tested on real systems. The efficiency of the complete context aware system coupled with the history manager is yet to be established the project is still underway

REFERENCES

[1] M. Weiser, 'The Computer for the 21st Century'. *Scientific American*. 1991.

[2] B. N. Schilit, D. M. Hilbert, and J. Trevor, 'Context-Aware Communication'. *IEEE Wireless Communication*, 2002, pp. 46-54.

[3] S. B. Mokhtar, D. Fournier, N. Georgantes, and V. Issarny, 'Context-Aware Service Composition in Pervasive Computing Environments', in *Lecture Notes in Computer* Science, vol. 3943, 2006, pp. 129-140.

[4] U. Mahmud, N. Iltaf, A. Rehman, and F. Kamran, 'Context-Aware Paradigm for a Pervasive Computing Environment (CAPP),' *in proceedings of WWW/Internet 2007*, Villa Real, Portugal, vol. 1, pp. 337-346, 5th - 8th October, 2007.

[5] R. Want, A. Hopper, V. Falcao and J. Gibbons, 'The Active Badge Location System', *ACM Transactions on Information Systems*, 10(1), 1992, pp. 91-102.

[6] K. Chaverst, K. Mitchell, and N. Davies, 'Design of an Object Model for a Context Sensitive Tourist GUIDE', *in proceedings of the International Workshop on Interactive Applications of Mobile Computing*, 1998.

[7] G. D. Abowd, et al., 'Cyberguide: a mobile Context-Aware Tour Guide', *ACM Wireless Networks*, 1997.

[8] A. K. Dey and G.D. Abowd, 'The Context Toolkit: Aiding the Development of Context-Aware Applications', *in Workshop on Software Engineering for Wearable and Pervasive Computing*, 2000.

[9] M. Roman, et al., 'Gaia: A Middleware to Enable Active Spaces', *IEEE Pervasive Computing*, 2002, *pp*. 74-83.

[10] M. Riaz, S. L. Kiani, S. Lee, S. Han and Y. Lee, 'Service Delivery in Context Aware Environments: Lookup and Access Control Issues', *in proceedings of RTSCA'05*, 2005.

[11] H. Chen, T. Finin and A. Joshi, 'An Intelligent Broker for Context-Aware Systems', *in Proceedings of UbiComp 2003*, 2003, pp. 183-184.

[12] M. Samulowitz, F. Michahelles and C. Linnhoff-Popien, 'CAPEUS: An Architecture for Context-Aware Selection and Execution of Services', *in New developments in distributed applications and interoperable systems*, 2001, pp. 23 – 39.

[13] T. Gu, H. K. Pung and D. Q. Zhang, 'A Middleware for Building Context-Aware Mobile Services', *in Proceedings of IEEE Vehicular Technology Conference*, LA, USA. 2004, pp. 2656-2660.

[14] P. Fahy, and S. Clarke, 'CASS A Middleware for Mobile Context-Aware Applications', *in Proceedings of MobiSys 2004*, Boston, USA. 2004.

[15] Y. K. Wang, 'Context Awareness and Adaptation in Mobile Learning', *in proceedings of 2^{nd} IEEE International Workshop on Wireless and Mobile technologies in Education (WMTE'04)*, 2004.

[16] S. Deugd, R. Carroll, K. E. Kelly, B. Millett and J. Ricker, 'SODA: Service-Oriented Device Architecture', *in IEEE journal of Pervasive Computing*, pp. 94-97, 2006.

[17] P. Korpipaa, J. Mantyjarvi, J. Kela, H. Keranen and E. Malm, 'Managing Context Information in Mobile Devices', *IEEE Pervasive Computing Journal*, pp.42 - 51, 2003.

[18] T. Hofer, W. Schwinger, M. Pichler, G. Leonhartsberger and J. Altmann, 'Context-Awareness on Mobile Devices –

© 2009 ACADEMY PUBLISHER

The Hydrogen Approach', *in proceedings of the 36th Annual Hawaii International Conference on System Sciences*, pp. 292-302, 2002.
[19] M. Horridge, H. Knublach, A. Rector, R. Stevens, and C. Wroe, 'A Practical Guide to buliding OWL Ontologies Using The Protege-OWL Plug-in and CO-ODE Tools Edition 1.0.' *Protege.* http://protege.stanford.edu.
[20] H. Guo, G. Gao, J. Ma, Y. Li and R. Huang, 'Research of an Adaptive System in Mobile Learning Environment', *R&D Center for Knowledge Engineering*, Beijing Normal University, China, 2008.
[21] R. Mayrhofer, 'An architecture for context prediction', *in Advances in Pervasive Computing*, vol. 176, pp. 65-72, Austrian Computer Society (OCG), April 2004.
[22] M. L. Blum 'Real-time Context Recognition', *MS Thesis*, 2005, Swiss Federal Institute of Technology (ETH).
[23] O. Bridiczka, J. L. Crowley and P. Reignier, 'Learning Situation Models for Providing Context-Aware Services', *in Lecture Notes in Computer Science*, vol. 4555/2007, pp. 23-32, 2007.
[24] J. Yuan, Y. Wu, 'Context Aware Clustering', *in proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-8. 2008.

Nazir Ahmed Malik did his B.Sc. (Hons.) in Naval Sciences from University of Karachi, Pakistan in 1996 and then M.Sc. in Computer Science from UAF, Faisalabad, Pakistan in 2000. After completing his Masters, he has been serving at various related positions. He then enrolled in MS in Information Security and is now doing his PhD degree in information security from College of Electrical and Mechanical Engineering, National University of Sciences and Technology, Pakistan. His areas of interest are computer networks, information security and context-awareness.

Dr Muhammad Younus Javed did his PhD in Adaptive Communication Systems from University of Dundee, Scotland, United Kingdom in 1991 and MS in Predictive Systems from the same university in 1988. He completed BE Electrical Engineering from UET Lahore, Pakistan, in 1982. He is serving in the College of Electrical and Mechanical Engineering (CE&ME) since 1991 and has taught a number of courses at undergraduate and postgraduate level. He is currently Head of the Computer Engineering Department at CE&ME, National University of Sciences & Technology (NUST), Rawalpindi, Pakistan. His areas of interest are biometrics, parallel systems, operating systems, computer networks, digital image processing, database systems and design & application of algorithms. He has 148 national/international publications to his credit.

Engr. Umar Mahmud did his MS in Software Engineering from National University of Sciences and Technologies, (NUST) Pakistan in 2006 and BE in Software Engineering from the same university in 2003. He is presently pursuing PhD degree in Software Engineering at College of Signals, NUST. His areas of interest are context-awareness, pervasive computing and machine learning.

© 2009 ACADEMY PUBLISHER