Multidimensional Context-Aware Social Network Architecture for Mobile Crowdsensing

Xiping Hu, Xitong Li, Edith C.-H. Ngai, Victor C.M. Leung, and Philippe Kruchten

ABSTRACT

This article proposes a multidimensional context-aware social network architecture, which aims to provide a mobile ecosystem to enable context awareness in the development and utilization of mobile crowdsensing applications. This mobile ecosystem is constructed to provide context awareness capabilities for different roles (i.e., users or developers) in the system and facilitate interactions between them. This system can ease the development of context-aware mobile applications, and enable context-aware mobile crowdsensing considering environmental, personal, and social information. We present a flow of context-aware solutions designed on this system, and highlight the orchestrations and advantages of different context-aware schemes in the system for different types of users (requesters and participants) in mobile crowdsensing. We demonstrate the feasibility of the proposed mobile ecosystem by presenting a novel context-aware mobile crowdsensing application called *Smart City*, and evaluate the system performance based on this application.

INTRODUCTION

The widespread deployment of sensors in our daily living environments and the popularity of mobile devices, such as smartphones and tablets, have provided us with numerous sources of sensing data. Contemporary mobile devices are commonly equipped with sensors such as cameras, Global Positioning System (GPS) receivers, and accelerometers. The sensing data from wireless sensors and mobile devices enable the development of sophisticated context-aware mobile applications and services, which has motivated a wide range of innovative research [1].

As a special form of crowdsourcing [2], mobile crowdsensing aims to provide a mechanism to involve participants from the general public to efficiently and effectively contribute and utilize context-related sensing data from their mobile devices in solving specific problems in collaborations [3]. Also, different from conventional sensing solutions using specialized networks of sensors, mobile crowdsensing leverages human intelligence to collect, process, and aggregate sensing data using individuals' mobile devices (e.g., using a camera to capture a specific target) to realize a higher-quality and more efficient sensing solution. Thus, a connection exists naturally between mobile crowdsensing and context awareness, where context awareness could maximize the data quality for the requesters of crowdsensing by delivering personalized services relevant to their application context. Context awareness also allows better understanding of the participants' situations, which helps to delegate sensing tasks to appropriate participants and coordinate them to finish the crowdsensing tasks in an energy-efficient manner (e.g., minimize the energy consumption of their mobile devices). Simultaneously, mobile crowdsensing could be an effective approach for the aggregation of context-related sensing data to achieve context awareness of mobile applications.

A remarkable trend in mobile computing is the increasing use of mobile devices to access social networking services. The wide availability of sensing modules in mobile devices enables social networking services to be extended to incorporate location-based services, media tag services, and so on. Therefore, there is a growing interest in fusing social networking services with real-world sensing, such as crowdsensing [4]. Social networking not only can provide an ideal platform to encourage mobile users to participate in crowdsensing, but also help improve the context awareness of mobile applications and better assist users in mobile crowdsensing by analyzing and utilizing their social contexts.

Currently, a number of research solutions on context awareness with mobile crowdsensing have been proposed, including location-based data-dissemination-based solutions, the co-design of them [3], and a generic solution for developing mobile applications with capabilities of context-aware data communications [5]. However, from the human and system aspects, the research work that can outline the positions and

Xiping Hu, Victor C.M. Leung and Philippe Kruchten are with the University of British Columbia.

Xitong Li is with the Massachusetts Institute of Technology.

Edith C.-H. Ngai, is with Uppsala University.

 0163-6804/14/$25.00 © 2014 IEEE

orchestrations of different roles in context-aware mobile crowdsensing has not yet been investigated. In this article, we explore and illustrate how context-aware mobile crowdsensing can facilitate the contribution and utilization of different roles (users, developers, and service providers) through a network of interactions like an ecosystem [6] to create more value for each individual. We propose a mobile ecosystem that aims to facilitate the integration of ubiquitous context-related data with multiple mobile applications in a social network setting. This system can facilitate the context awareness of such mobile applications in crowdsensing with the following features:

- It considers multidimensional sources of contextual data for comprehensive and personalized understanding of user requirements, which can greatly improve data delivery rate and reduce energy consumption in mobile devices in crowdsensing.
- It integrates seamless and multidimensional flows of context-aware solutions to collect and elaborate on contextual data, which can improve and personalize mobile services for crowdsensing, and allow creation of new crowdsensing applications.
- It provides a generic model to deploy, examine, and evaluate different context-aware solutions for mobile crowdsensing applications.

In the rest of this article, we first outline the key components of the proposed mobile ecosystem, and then introduce how the system facilitates incorporation of context awareness in the development and runtime of applications. Next, we present the flow of context awareness in the mobile ecosystem, and highlight the orchestrations and advantages of different context-aware services in the system for different types of users (requesters and participants) in crowdsensing. Finally, based on the mobile ecosystem, we present a novel context-aware mobile application for crowdsensing called *Smart City*, and evaluate the system performance based on this application.

Mobile Ecosystem Overview

As shown in Fig. 1a, the overall architecture of the mobile ecosystem consists of four parts: a) social networks, b) ubiquitous sensors, c) a mobile context-aware platform (MCP), and d) the Vita cloud platform (introduced in our previous work [7]). In this architecture, a and b already exist, and are widely used in our daily lives. The main concerns are how to effectively communicate with them in the mobile ecosystem to obtain context-related data and make use of such data in context-aware applications. These concerns are addressed in the next section.

The MCP provides the initial environment and context-aware services to support users to participate in and perform crowdsensing tasks through their mobile devices. It mainly consists of three components: a mobile service-oriented architecture (SOA) framework, a context-aware semantic service (CSS), and a multidimensional contextual data aggregation (MCDA) service.

The mobile SOA framework (along with the mobile SOA server shown in Fig. 1a) was proposed and implemented in our former work [8]. It is an extensible and configurable framework built on the specifications and methodologies of RESTful web services [9]. It integrates popular social networking services (e.g., Facebook) and software agent services as shown in our earlier work [10], and adopts an SOA to support the development of multiple mobile web-service-based applications and services in an efficient and flexible way, with standard service interactions during mobile devices' runtime. Furthermore, in the mobile ecosystem, this mobile SOA framework works as a bridge between the MCP and Vita over the Internet. Working with the mobile SOA framework, the CSS and MCDA are designed to provide a seamless context-aware solution for mobile crowdsensing applications.

In the mobile ecosystem, Vita works as a central coordinating platform to store and process diverse contextual data from the participants in crowdsensing and the social networking service providers, as well as a development environment to support the development of context-aware mobile crowdsensing applications. It mainly consists of four components: a management interface, a storage service, a deployment environment, and a process runtime environment.

In the mobile ecosystem, Vita works as a central coordinating platform to store and process diverse contextual data from the participants in crowdsensing and the social networking service providers, as well as a development environment to support the development of context-aware mobile crowdsensing applications.

Management interface: It provides the development environment and application programming interfaces (APIs) to develop different context-aware applications and services for mobile crowdsensing. Also, it incorporates various cloud-based contextual data collection (CCDC) schemes.

Storage service: It supports automatic backup of data related to application services, task lists, sensing data uploaded by mobile devices, and the aggregated results from crowdsensing.

Deployment environment: It enables dynamic deployments of the MCP and various application services for participants of mobile crowdsensing. Also, it can automatically deploy new context-aware services uploaded by developers in the process runtime environment for executing diverse mobile crowdsensing applications.

Process runtime environment: It is built on open source techniques, and integrates the context-aware crowdsensing engine (CCE), which works with CCDC to process and utilize diverse contextual data from multiple sources to provide context-aware services supporting mobile crowdsensing in real time.

An overview and relations between different roles in the mobile ecosystem are shown in Fig. 1b. Note that the regions with the same color in Figs. 1a and 1b are mapped to each other. The mobile ecosystem provides foundation support to developers in two aspects:

- It eases the development of extensive context-aware mobile applications by service reuse, and enables such mobile applications to involve collaborations between mobile devices in localized social networking environments.
- Developers do not need to be concerned with the issues of transforming raw sensing

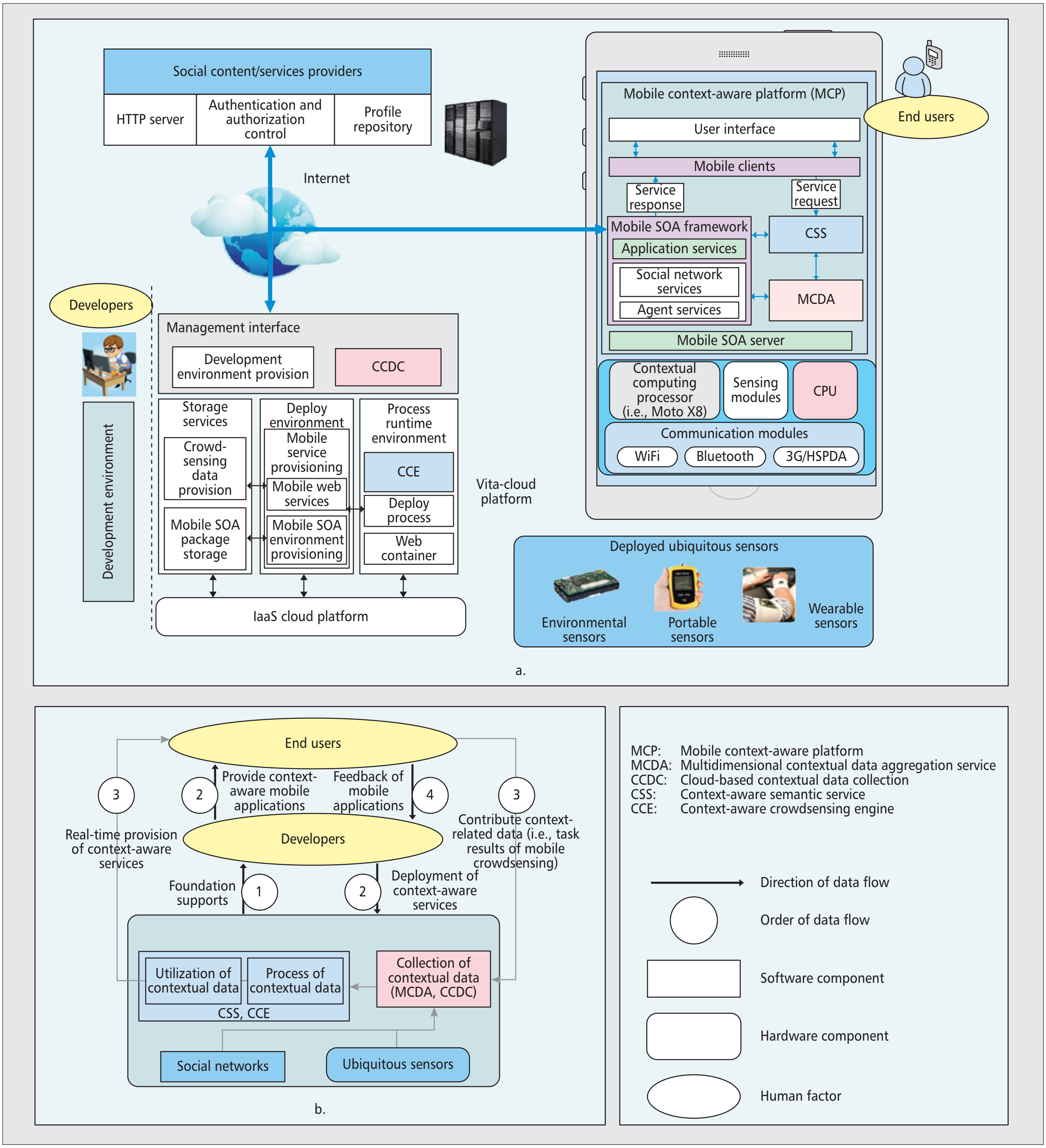

Figure 1. Overview of the mobile ecosystem.

data to contextual information, and mapping between specific service requests of mobile applications and the corresponding context information at runtime. Instead, they can focus on developing the functions of the mobile application. From the end users' perspective, they could not only use various context-aware mobile crowdsensing applications provided by developers, but are also supported by the real-time context-aware services of the system.

On the other hand, with more developers working on the ecosystem, more context-aware services could be deployed on it. Since the system provides real-time context-aware services to users in mobile crowdsensing, users can contribute more context-related data to the system during their interactions, which could enrich the context information of each user in the system. It also allows the system to provide personalized context-aware services to individual users intelligently.

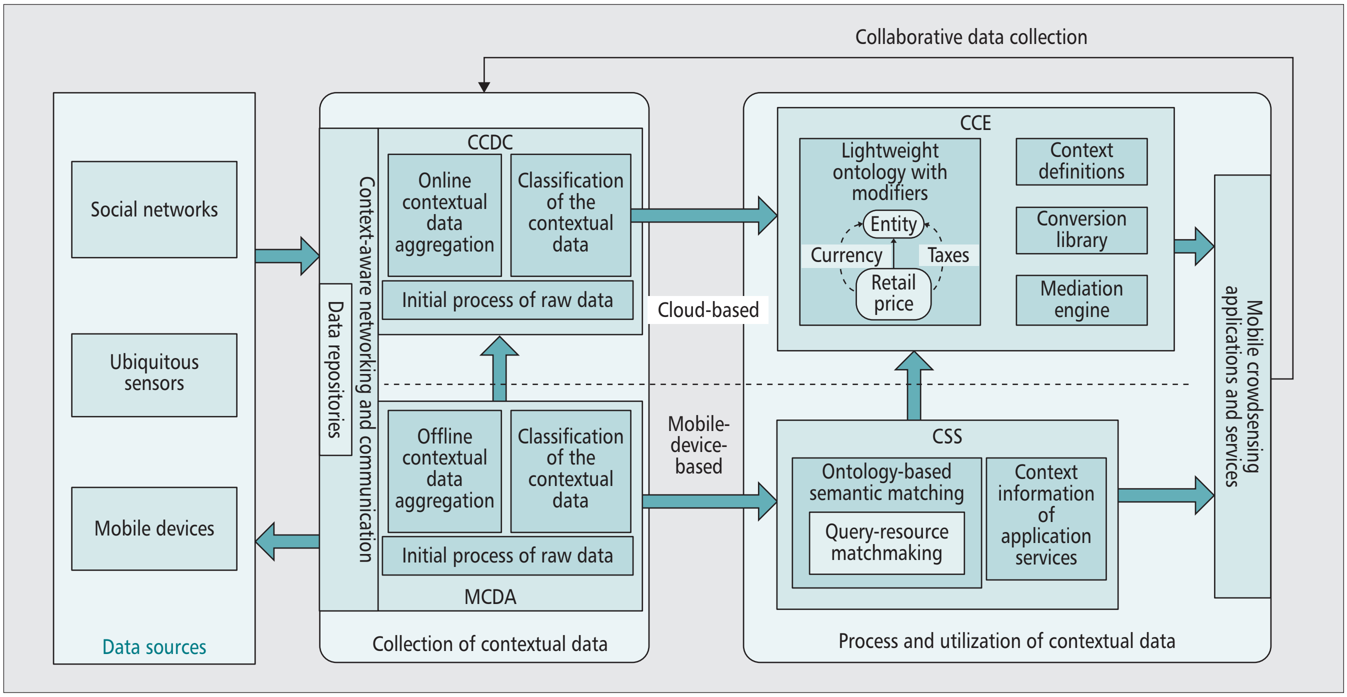

Figure 2. Overall flow of context-aware solutions in the mobile ecosystem.

Flow of Context Awareness in the Mobile Ecosystem

As shown in Fig. 2, our architecture is multidimensional in two aspects: it considers multidimensional data sources in contextual data collection, and it adopts multidimensional flows of context-aware solutions (mobile-device-based and cloud-based). The flow of context-aware solutions in our mobile ecosystem consists of three steps: collection, processing, and utilization of contextual data. We present the three steps as follows.

Multidimensional Context-Aware Data Collection

Context-aware data collection is concerned with the acquisition of context information of users' situations, including the characteristics of users: their locations, activities, and environments. It starts with collection of raw data from different external sources, such as environmental sensors, mobile devices, or data, which can be input actively by the mobile users in mobile crowdsensing tasks. The raw data from external sources refers to measurements collected by hardware sensors (e.g., location, light, sound, movement, touch, and temperature). It can also be specified by a user or captured by monitoring user interactions, such as the user's goals, tasks, work context, and emotional state, from their online social activities.

We introduce personalized context awareness in mobile crowdsensing, which involves multidimensional context-aware data collection from different sources and the aggregation of contextual data of individual users. Context-aware data collection obtains raw sensing data and provides initial processing to transform the raw data to contextual data by contextual computing, and it can be implemented in the CCDC and MCDA components in our system design, as shown in Figs. 1 and 2. We divide the contextual data into three main categories:

- Environmental data
- Personal activity data
- Social data

The raw sensing data and contextual data are summarized in Table 1.

Contextual data collection from multiple sources enables a comprehensive and personalized understanding of user interests, characteristics, and behaviors. This contextual data of individual users is updated automatically and adaptively according to environment and behavioral changes. The data help understand the needs of users in different applications and provide them with personalized services that satisfy their application requirements. In addition, the contextual data can be used to improve the performance in networking and communication for data collection. For example, the network infrastructure and communication can be optimized (i.e., deployment of data repositories) to increase the data delivery rate and reduce the communication overhead by considering the interests and mobility patterns of mobile users [11, 12]. Since many mobile phones are equipped for short-range communications (e.g., using Bluetooth) nowadays, they can be used to read sensing data from the wireless sensors in their surroundings. Similarly, contextual data can help to improve data availability and improve energy efficiency in opportunistic data sharing [13].

Processing and Utilization of Contextual Data

The collected contextual data described above can further be processed in the mobile ecosystem and transformed into the appropriate context

Raw sensing data		Context-aware data collection →	Contextual data	
Mobile data	–Location data (GPS, WiFi, cell tower) –Human interaction with phone (texting, calling) –Activity data (accelerometer) –Noise (microphone)		Environment data	–Temperature, humidity, pollution –Indoor, outdoor –Location (home, work, leisure, in transit)
Sensor data	–Environment data (temperature, humidity, pollution) –Health data from body sensors (heart rates, blood pressure, stress level)		Personal activity data	–Walking, running, sleeping –Emailing, texting, calling –Heart rate, blood pressure, stress level –Mobility pattern
Online data	–Facebook, Twitter (social relationships)		Social data	–Social relationships (friendship, connection) –Online/offline social behaviors and activities

Table 1. Multi-dimensional context-aware data collection.

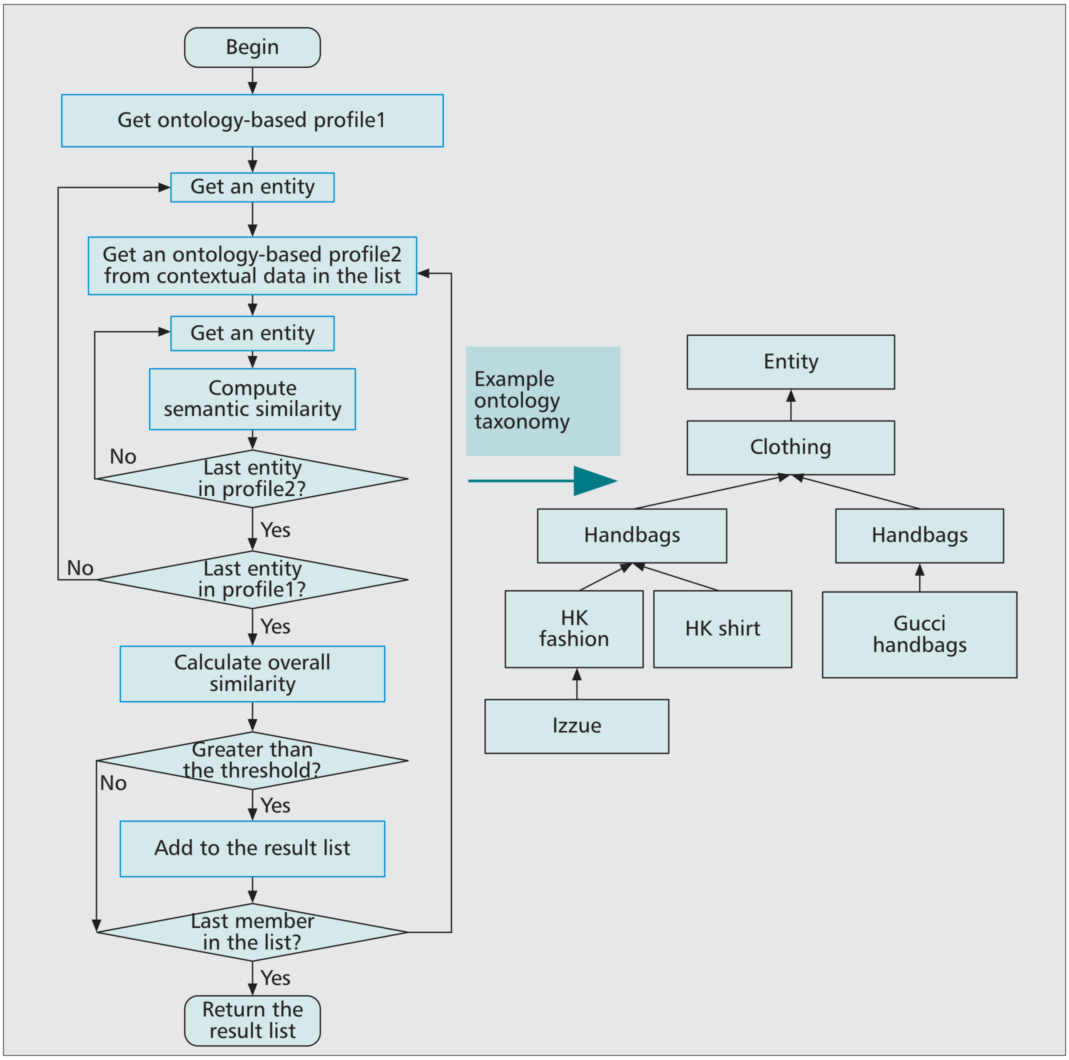

Figure 3. Flow of ontology-based semantic matching..

information relevant to the situation, activities, and social context of different users in crowdsensing. For requesters of crowdsensing, the context information can be used to enable them to deliver the relevant crowdsensing tasks to the right participants at the right time and place, and improve information retrieval when gathering diverse crowdsensing results from different participants. Also, the context information can be used to coordinate participants of crowdsensing to collect sensing data that are relevant and valuable to their activities, interests, and social and environmental contexts. Collaborative data collection supported in crowdsensing can optimize data

Figure 4. Smart City: context-aware mobile crowdsensing application.

accuracy and energy consumption for the community considering the locations and availability of participants, and simultaneously provide an effective dimension for aggregation of contextual data.

Mobile-Device-Based — As shown in Fig. 1b, CSS processes and utilizes the contextual data collected by MCDA to improve the context awareness of the mobile crowdsensing applications. CSS is deployed in the MCP, which can be invoked by a mobile client to provide context-aware matchmaking between service requesters and providers in runtime. The design of the CSS mainly consists of two parts:

- Definition of application services and contextual data
- Ontology-based semantic matching

In CSS, we define three categories of context information on the application services deployed on MCP:

Description information: It includes the description of an application service (e.g., name, category).

Interface information: It consists of the information transformation, for example, the application specified contextual data needed for computation and the result of computation, and the state changes caused by service execution (e.g., required logic condition and result of execution).

Implementation information: It contains technical specifications including business logic and data, coding, and database configuration.

Moreover, various kinds of context information could be extracted from the contextual data (from MCDA). It is hard to enumerate all the types of context information completely, but it is feasible to predefine and classify the core context into several main categories in the shape of an information tree according to specific applications. Developers can follow such a method to define all types of context information available in the application services and their related contextual data when they deploy such services to MCP so as to standardize its data interactions and pave the way for ontology-based semantic matching.

Ontology-Based Semantic Matching

There are various ways of entity matching. For example, the traditional information search approach compares the descriptive information about two entities and measures their similarity using simple string-matching methods. Such an approach cannot work well as it requires crowdsensing requesters and participants who join the crowdsensing tasks to use exactly the same words in open and dynamically changing environments, which is unrealistic. In this article, we propose

We propose an ontology-based semantic matching method. Compared to traditional ID or keyword-based methods, using ontology-based matching makes it more flexible and adaptable to the dynamic mobile environment.

an ontology-based semantic matching method. Compared to traditional ID or keyword-based methods, using ontology-based matching makes it more flexible and adaptable to the dynamic mobile environment. An ontology is a specification of a conceptualization that describes the concepts and relationships existing in a domain. A number of ontology-based approaches for context-aware mobile service have been proposed [14]. Instead, in CSS, we propose a novel similarity computing method as a lightweight and generic solution for mobile devices, which can make use of different ontologies to calculate the semantic similarity of two items. Application developers can flexibly adopt any existing ontologies for simplicity, or design application-specific ontologies with the help of domain experts to achieve greater efficiency.

To obtain the context information sought by a mobile client, the CSS executes the query-resource matchmaking component, whose core task is to compute the similarity between concepts. The semantic similarity of two concepts (words) is mainly decided by impacts from the following three aspects: the distance (the length of the path) between two words; the depth of two words and the depth of their most specific common parent in the common ontology; and whether the direction of the path between the two words is changed.

Figure 3 shows an example of the taxonomy in a common ontology. First, the distance between two words refers to how many steps it takes from one word to reach the other in the ontology; for example, there are five steps from the word "Izzue" to the words "Gucci handbags." The more steps between two words, the larger the distance between them, and the lower their similarity. Second, considering the depths of two words in the ontology satisfies our intuitive understanding of an ontology's classification tree: the classification tree is a progressive refinement work; when two classes are deeper in the classification tree, the relationship between them becomes closer, so the distance is smaller and the similarity is higher. For example, the words "HK fashion" and "HK shirt" are closer than the relationship between the words "Fashion" and "Handbags." Furthermore, two words have a higher similarity when they are closer to their most specific common parent in the taxonomy tree. Finally, if the direction of the path between two words changes, the two words must not belong to the same tree branch, and their similarity should be degraded by assigning a lower score. For example, the path between the words "HK fashion" and "HK shirt" changes direction, while the path between "HK fashion" and "clothing" does not.

Cloud-Based — Considering the limited computing power of mobile devices, diverse contextual data need to be uploaded to the cloud platform for further processing. In our mobile ecosystem, the contextual data uploaded from mobile devices to the cloud platform of Vita could be in the format of web services [7, 8]. Such web-based contextual data are processed by CCE, which works on the cloud platform of Vita, to transform them to context information.

Data from different independent sources often have implicit assumptions of interpretation. For example, a retail price collected from a mobile device in the United States is assumed to be in U.S. currency, not including retail taxes. On the other hand, when a retail price is collected from Hong Kong, it is assumed to be in H.K. currency and includes retail taxes. Such implicit assumptions of data interpretation need to be explicitly specified in order to make the raw data context-aware. During application development, developers can pre-specify the context definition for the mobile devices. For instance, developers can indicate the context of the mobile devices according to their locations. When some data about retail prices come from the United States, CCE can automatically annotate the data as U.S.-related context, while CCE can annotate the retail price as H.K.-related context if the data come from Hong Kong.

CCE uses a lightweight ontology as the common vocabulary for the integration of the contextual data from different mobile devices. The unique feature of the lightweight ontology lies in the addition of modifiers. As introduced in our prior work [15], a modifier is used to capture additional information that affects the interpretation of generic concepts. A generic concept (e.g., retail price) in the ontology can have multiple modifiers, each of which indicates an orthogonal dimension of the variations in data interpretation. In a certain context each modifier is assigned by a specific modifier value. For instance, as shown in Figure 2, the concept of retail price can have two modifiers: one is currency and the other is taxes. When the data of retail price is annotated with U.S.-related context, the value assigned to the currency modifier is *USD*, and the value assigned to the taxes modifier can be *not-including*. Similarly, when the data of retail price is annotated with HK-related context, the value assigned to the currency modifier is *HKD*, and the value assigned to the taxes modifier can be *including*. Accordingly, when CCE knows where the data come from (e.g., the location of the mobile devices), CCE can understand the context of the data sources and therefore know how to interpret the data based on the values of the modifiers associated with the corresponding context.

Given the lightweight ontology with augmented contexts (including modifiers), CCE can understand the semantics of the data collected by different mobile devices. More important, CCE can automatically process the context information and convert the data by using the context annotation of the mobile devices. For example, CCE can understand the semantics and context of a retail price collected from the United States and convert it to the appropriate information for a receiver located in Hong Kong. To fulfill the data conversion, CCE can select the appropriate conversion rules from a prespecified conversion library based on the contexts of the source and receiver.

Compared to the ad hoc brute force approach, the proposed approach based on the lightweight ontology with augmented contexts has some desirable properties, including adaptability, extensibility, and scalability, which can significantly alleviate the reconciliation efforts needed for the interactions of diverse web services. A systematic comparison and evaluation of the proposed design approach can be found in [15].

Application Case

Based on the proposed mobile ecosystem with our context-aware solutions, we develop a prototype application called *Smart City* to demonstrate the functionalities of context-aware services provided by our system for mobile crowdsensing in our daily lives. *Smart City* consists of two generic functions: *services* and *crowdsensing*; and two application-specific functions: *eating* and *shopping tour*. The screen shots of some of these functions are shown in Fig. 4. The function of *services* is based on the mobile SOA framework and takes advantage of the RESTful web service, which allows developers to flexibly extend new functions to this application; and the function of *crowdsensing* is based on the social network services on MCP, which enables users to post and/or accept crowdsensing tasks by their mobile devices through social networks.

Eating — As shown in the screenshot in the left corner of Fig. 4, the eating function is designed to enable people who travel to a new city to conveniently find out and/or share preferred food information. In the following, we use an example to demonstrate this function. Assume that a visitor named Betty travels to Hong Kong from the United States. As shown in Fig. 4, she posts the related crowdsensing request through the eating function of the *Smart City* application on her phone: *What is the delicious food in Hong Kong?* Based on Betty's social activity records (i.e., the frequency and type of restaurants she has visited) stored in the cloud platform of Vita, Betty's dining habits can be identified, showing that she favors non-spicy food. Therefore, CCE collects this context information from Vita and annotates the request posted by Betty with her dining preference (i.e., context annotation). Subsequently, CCE can automatically push the request to people who have similar dining preferences as Betty and happen to be dining at some restaurants in Hong Kong. Note that people with different dining preferences ar not presented with Betty's request, making sure that the request is delivered to an appropriate target group, which leads to a relevant and accurate query result. Through the user interface of the *Smart City* application, diners with similar dining preferences as Betty take photographs of the food and attach their comments. The photographs and comments are automatically combined with the location and map services, and then the answers are uploaded to the cloud platform of Vita. After the cloud platform of Vita gets all of the answers, it can automatically combine them, and the overall answers are then returned to Betty's mobile phone (shown in the left corner of Fig. 4). Thus, with the help of context-aware mobile crowdsensing through *Smart City*, Betty may enjoy local flavors in Hong Kong that are preferable to her.

Shopping Tour — Similar to the eating function, this function is designed to assist visitors in a city to conveniently find out and/or share shopping information in which they are interested. As shown in the screen shot in the right corner of Fig. 4, Betty posts the crowdsensing request: *Recommendations of branded clothing in Hong Kong*. As shown in the Unified Modeling Language (UML) diagrams at the bottom of Fig. 4, with the help of MCDA and CSS on her phone, it can automatically retrieve the implicit context information about her task in two categories: clothing information, such as Izzue fashion and Gucci handbags (the stores she usually visits); and user information, for example, female, H.K. branded preference (through basic information from her social account), and average consumption in fashion \$350–\$900. After that, the MCP posts her task with the context information to the cloud platform of Vita. On the other hand, the MCDA and CSS automatically retrieve the related context information of the participants who chose to join Betty's task. After the MCP confirms the shopping information they provide that match the context of Betty's task, it uploads such information to the CCE of the cloud platform. Finally, the system automatically combines all the shopping information that fits Betty's crowdsensing task and returns them to her phone; for example, COUR CARRE clothing (similar to Izzue) has a 10 percent discount today.

Based on the proposed mobile ecosystem with our context-aware solutions, we develop a prototype application called Smart City to demonstrate the functionalities of context-aware services provided by our system for mobile crowdsensing application in our daily lives.

System Evaluation

We evaluate the system performances in terms of three parameters: time efficiency, energy consumption, and networking overhead on mobile devices, as these parameters are of particular concern to mobile users when they are participating in mobile crowdsensing. The communications between the cloud platform and MCP of the system use the standard web service format based on the HTTP protocol and XML data format, and the experimental environment is: ***Hardware***: Amazon EC2 M1 Medium Instance; 3.75 Gbytes memory; 2 EC2 compute unit; 410 Gbytes instance storage; 32-bit or 64-bit platform; I/O performance: moderate; EBS-Optimized available: no. ***Software***: operating system: Ubuntu 12.04.1; servers: ApacheTomcat 7.0.33; BPEL engine: BPEL4; people environment: ODE1.3.5.

Three LG E970 (Android v. 4.0.4, 2100 mAh battery capacity) mobile devices were used in our experiments, from which we obtained three sets of data simultaneously. A total of 10 tests were run over two days, and the average results were calculated. Each E970 sent an intensive computing task (benchmark — ε£-GALEN ontology) to the cloud platform of Vita according to a Poisson distribution with an arrival rate of E = 3/min, and the screen was turned off during runtime. The time delay consists of:

- Response and communication time between the Vita cloud platform and the mobile devices
- Processing time of the task

The experimental results are summarized in Table 2. We find that the results of the three

Experimental results have demonstrated the effectiveness and feasibility of our system for real-world deployment. Furthermore, our system architecture provides a generic model to deploy, examine, and evaluate different context-aware solutions for mobile crowdsensing applications.

Parameters		Data set 1	Data set 2	Data set 3
Time delay (ms)		**Response time** Max.: 19,631, min.: 9475 Average: 10,673	**Response time** Max.: 20,377, min.: 6874 Average: 10,865	**Response time** Max.: 20,636, min.: 9665 Average: 10,239
		Process time Max.: 2776, min.: 2265 Average: 2573	**Process time** Max.: 2946, min.: 2132 Average: 2685	**Process time** Max.: 2834, min.: 2156 Average: 2343
Battery consumption		75 mAh/30 min	72 mAh/30 min	75 mAh/30 min
Network overhead		0.73 Mbyte/90 requests	0.70 Mbyte/86 requests	0.79 Mbyte/97 requests

Table 2. Overall system performance.

sets of data are very similar, with all averaging about 13.1 s. Also, we find that the same task processed on LG E970 needs to spend 136 s on average with a battery consumption level of 300 mAh/30 min. It shows that our cloud platform can perform communication and computation efficiently with much shorter delay than with local computation. Since the current crowdsensing server of Vita is designed to support different types of potential crowdsensing tasks simultaneously, the response time of the server is inevitable a bit longer comparing to the servers deployed for only specific applications.

In addition, we have evaluated the performance on energy saving with context-aware location-based services. We requested the users to enter the locations of interest (e.g., shops that they want to get a reminder for buying certain items). We have shown that GPS updates can be reduced from 22 times to only 10 times without missing any locations of interest. The result demonstrates that we can save more than 50 percent of energy by considering the sensing context (locations and interests) of users.

Conclusions

In this article, we have proposed a multidimensional context-aware social network architecture, which provides a mobile ecosystem that can improve the context awareness of multiple mobile crowdsensing applications and support both developers and users. We have described the key components of our mobile ecosystem and discussed how they can facilitate context awareness for multiple roles in the system, and the interactions between them. Our proposed system provides a flow of context-aware solutions that orchestrates the advantages of different context-aware schemes for different types of users (requesters and participants) in crowdsensing. We have presented a context-aware mobile crowdsensing application, *Smart City*, developed on our system to demonstrate its practicality and usefulness in people's daily lives. Experimental results have demonstrated the effectiveness and feasibility of our system for real-world deployment. Furthermore, our system architecture provides a generic model to deploy, examine, and evaluate different context-aware solutions for mobile crowdsensing applications.

Acknowledgment

This work is supported in part by the Canadian Natural Sciences and Engineering Research Council through the NSERC DIVA Strategic Research Network, by TELUS and other industry partners, and by STINT Initiation Grant for International Collaboration (2013) and SSF ProFuN project (2009–2014) in Sweden.

References

[1] M. J. Sinderen *et al.*, "Supporting Context-Aware Mobile Applications: an Infrastructure Approach," *IEEE Commun. Mag.*, vol. 44, no. 9, Sept. 2006, pp. 96–104.
[2] J. Howe, "The Rise of Crowdsourcing," *Wired*, June 2006.
[3] R. K. Ganti, F. Ye, and H. Lei, "Mobile Crowdsensing: Current State and Future Challenges," *IEEE Commun. Mag.*, vol.49, no.11, Nov. 2011, pp. 32–39.
[4] G. Cardone *et al.*, "Fostering Participaction in Smart Cities: A Geo-Social Crowdsensing Platform," *IEEE Commun. Mag.*, vol. 51, no. 6, June 2013, pp. 112–19.
[5] M. Raento *et al.*, "ContextPhone: A Prototyping Platform for Context-Aware Mobile Applications," *IEEE. Pervasive Computing*, vol. 4, no.2, 2005, pp. 51–59.
[6] J. Bosch and P. Sijtsema, "From Integration to Composition: On the Impact of Software Product Lines, Global Development and Ecosystems," *J. Systems and Software*, vol. 83, no. 1, 2010, pp. 67–76.
[7] X. Hu *et al.*, "Vita: A Crowdsensing-Oriented Mobile Cyber Physical System," *IEEE Trans. Emerging Topics in Computing*, vol. 1, no. 1, June 2013, pp. 148–65.
[8] X. Hu *et al.*, "A Service-Oriented Mobile Social Networking Platform for Disaster Situations," *Proc. HICSS*, 2013, pp. 136–45.
[9] C. Pautasso, O. Zimmermann, and F. Leymann, "RESTful Web Services vs. 'Big' Web Services: Making the Right Architectural Decision," *Proc. WWW*, 2008, pp. 805–14.
[10] X. Hu *et al.*, "A Semantics-Based Multi-Agent Framework for Vehicular Social Network Development," *Proc. ACM DIVANet*, 2011, pp. 87–96.
[11] E. C.-H. Ngai, J. Liu, and M. Srivastava, "OppSense: Information Sharing for Mobile Phones in Sensing Field with Data Repositories," *Proc. IEEE SECON*, 2011, pp. 107–15.
[12] E. C.-H. Ngai, M. B. Srivastava, and J. Liu, "Context-Aware Sensor Data Dissemination for Mobile Users in Remote Areas," *Proc. IEEE INFOCOM*, Mar., 2012, pp. 25–30.
[13] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic Networking: Data Forwarding in Disconnected Mobile Ad Hoc Networks," *IEEE Commun. Mag.*, vol. 44, no. 11, Nov. 2006, pp. 134–41.
[14] T. Gu, H. Pung, and D. Zhang, "A Middleware for Building Context-Aware Mobile Services," *Proc. IEEE VTC 2004-Spring*, 2004, pp. 2656–60.
[15] X. Li, S. Madnick, and H. Zhu, "A Context-Based Approach to Reconciling Data Interpretation Conflicts in Web Services Composition," *ACM Trans. Internet Technology*, vol. 13, no. 1, article 1, Nov. 2013.

Biographies

XIPING HU (xipingh@ece.ubc.ca) joined the University of British Colombia (UBC) in September 2011, and he is now working toward his Ph.D. in the Electrical and Computer Engineering Department of UBC. His current research areas at UBC are mobile social networks, mobile computing, and crowdsourced sensing.

XITONG LI (xitongli@mit.edu) received his B.S. and Ph.D. in engineering from Tsinghua University, China. Currently, he is a fifth-year Ph.D. student in the Information Technologies group at the Sloan School of Management, Massachusetts Institute of Technology. His research interests include how to use online data and services, and the causal impacts of using online data/information.

EDITH C.-H. NGAI (edith.ngai@it.uu.se) is currently an associate professor in the Department of Information Technology, Uppsala University, Sweden. She received her Ph.D. from the Chinese University of Hong Kong in 2007. She did her post-doc at Imperial College London, United Kingdom. Her research interests include wireless sensor and mobile networks, information-centric networking, the Internet of Things, and network security and privacy. She has been a visiting researcher at the University of California at Los Angeles, Simon Fraser University, and Tsinghua University. She is a VINNMER Fellow (2009) awarded by VINNOVA in Sweden. Her co-authored papers have received best paper runner-up awards at IWQoS '10 and IPSN '13.

VICTOR C. M. LEUNG [F] (vleung@ece.ubc.ca) [S'75, M'89, SM'97, F'03] is a professor of electrical and computer engineering and holder of the TELUS Mobility Research Chair at UBC. His research is in the areas of wireless networks and mobile systems, on which he has co-authored more than 700 technical papers in archival journals and refereed conference proceedings, several of which have won best paper awards. He is a Fellow of the Royal Society of Canada, the Canadian Academy of Engineering, and the Engineering Institute of Canada. He is serving/has served on the editorial boards of *IEEE JSAC*, *Transactions on Computers*, *Wireless Communications*, *Vehicular Technology*, *Wireless Communications Letters*, and several other journals. He has provided leadership to the technical and organizing committees of many international conferences. He was the recipient of a 2012 UBC Killam Research Prize and an IEEE Vancouver Section Centennial Award.

PHILIPPE KRUCHTEN [SM] (pbk@ece.ubc.ca) is a full professor of software engineering in the Department of Electrical and Computer Engineering of UBC. He holds an NSERC Chair in Design Engineering. He joined UBC in 2004 after a 30+ year career in industry, where he worked mostly on large software-intensive systems design, in the domains of telecommunication, defense, aerospace, and transportation. Some of his experience is embodied in the Rational Unified Process (RUP). He holds a doctorate degree from Ecole Nationale Supérieure des Télécommunications, Paris.