Online ISSN 1848-3380, Print ISSN 0005-1144
ATKAFF 54(4), 388–400(2013)

Wei Wang, Suparna De, Gilbert Cassar, Klaus Moessner

Knowledge Representation in the Internet of Things: Semantic Modelling and its Applications

DOI 10.7305/automatika.54-4.414
UDK 004.738.5:81'372
IFAC 2.8.4; 21.0

Original scientific paper

Semantic modelling provides a potential basis for interoperating among different systems and applications in the Internet of Things (IoT). However, current work has mostly focused on IoT resource management while not on the access and utilisation of information generated by the "Things". We present the design of a comprehensive and lightweight semantic description model for knowledge representation in the IoT domain. The design follows the widely recognised best practices in knowledge engineering and ontology modelling. Users are allowed to extend the model by linking to external ontologies, knowledge bases or existing linked data. Scalable access to IoT services and resources is achieved through a distributed, semantic storage design. The usefulness of the model is also illustrated through an IoT service discovery method.

Key words: Internet of things, Knowledge representation, Ontology engineering, Distributed service storage, Service discovery, Service ranking

Prikaz znanja u internetu stvari: semantičko modeliranje i njegove primjene. Semantičko modeliranje pruža potencijalnu osnovu za međudjelovanje različitih sustava i aplikacija unutar interneta stvari (IoT). Međutim, postojeći radovi uglavnom su fokusirani na upravljanje IoT resursima, ali ne i pristupu i korištenju informacija koje generira "stvar". Predstavljamo projektiranje sveobuhvatnog i laganog semantičkog opisnog modela za prikaz znanja u IoT domeni. Projektiranje slijedi široko-priznate najbolje običaje u inženjerstvu znanja i ontološkom modeliranju. Korisnicima se dopušta proširenje modela povezivanjem na eksterne ontologije, baze znanja ili postojeće povezane podatke. Skalabilni pristup IoT uslugama i resursima postiže se kroz distribuirano, semantičko projektiranje pohrane. Upotrebljivost modela također je ilustrirana kroz metodu pronalaska IoT usluga.

Ključne riječi: Internet stvari, prikaz znanja, ontološko inženjerstvo, distribuirana pohrana usluga, pronalazak usluga, poredak usluga

1 INTRODUCTION

Advancement in wireless sensor networks has led to potential interests in integrating data and capabilities provided by the physical world objects into the current Internet. Alongside this, matured technologies for manufacturing Radio Frequency Identification (RFID) tags, sensors and actuators at low cost have led to millions of them being connected to the Internet. The IoT will have the interconnection of the objects or things from the physical world and their virtual representations on the Internet. It is envisioned by many that the information produced on the IoT, in combination with existing resources and services on the current Web, will enable revolutionary applications and business models.

As one of the fundamental constituents of the future Internet, IoT has attracted tremendous interests from various research communities and industry. During the past few years, the scope of research and development has been extended substantially, from the original focus on things traceability and accessibility using RFID tags to IoT infrastructure and architecture, communication protocols for constrained devices, (mobile) sensors and sensors networks, middleware, security and privacy, and many others. Among these developments, semantic oriented computing manifests its potential to cope with the challenging problems of heterogeneity and interoperability implied by the large number of things with different characteristics.

Issues related to interoperability, automation, and data analytics naturally lead to a semantic-oriented perspective towards IoT [1]. Applying semantic technologies to IoT promotes interoperability among IoT resources, information models, data providers and consumers and facilitates effective data access and integration, resource discovery, semantic reasoning, and knowledge extraction [2].

Already, we have seen many applications using semantic technologies in the IoT research, in particular the SSN ontology [3] for annotating sensors and sensor networks; "linked data" [4] for sensor data publishing [5] and discovery [6], and semantic sensor observation services (SemSoS) [7]. The semantic based methods, when combined with the principle of "service oriented computing" [8], provide a homogeneous and scalable means to access IoT information. This also allows existing methods for service discovery and composition to be easily integrated with the IoT based services to create context-aware and personalised services and applications.

Semantic modelling in many current works has mostly focused on IoT resource management while not on access of the information generated in IoT. Recent works in [9, 10] propose a modelling approach in which resources in the IoT are able to expose standard service interfaces (we term the services exposed by the connected Things in the physical world as *"IoT Services"*), embodying the "Entity-Device-Resource" modelling approach.

Our proposed description ontology integrates the existing models for sensor networks and related resource management platforms and extends them with IoT services and other important concepts. The ontology helps exploit the synergy of the existing efforts and provides support for crucial tasks in IoT such as resource and service discovery, IoT service testing, composition, adaptation, and so on. It is compatible with several widely used semantic models in IoT and is designed to be lightweight to promote reuse and support more efficient inference. More interestingly, it harnesses the "Linked Data" principles to leverage existing thematic and spatial data from various sources. This interlinking with other data sources which provide descriptions of particular aspects in more detail (e.g. geographical locations) supports the view noted by [11] which points out that one of the pillars of the IoT paradigm is for objects to interact with other entities.

The main contributions of this paper lie in three aspects: first, the design of a comprehensive while lightweight ontology (see Section 3) for modelling important concepts in the IoT domain is presented. Given the importance of the proximity knowledge in IoT, an indoor location ontology (with relative position) is also developed. We show how the ontologies can be used together to provide more fine-grained semantic annotations for services and resources, and to create meaningful linked IoT data for service/resource discovery. While the semantic realisation of the concepts provides a common platform for automated machine-interpretability, the scale of the IoT and the number of things engender new challenges for metadata storage and service/resource discovery. The second contribution is the design of a distributed storage solution which represents a scalable mechanism for accessing the semantically annotated IoT services and resources. Our third contribution is the illustration of the usefulness of the description ontology through the development of an IoT service discovery method. In particular, we show how the service matchmaking and ranking strategies can be formulated based on the description ontology, linked IoT data and semantic reasoning.

The rest of the paper is organised as follows. In Section 2, we review some of the representative semantic models pertinent to the IoT domain modelling and the methods for service discovery. Section 3 elaborates the design of the description ontology for IoT, specifically, the design principle, information model and its constituent modules are highlighted. In Section 4 we present an indoor location ontology and show how it can be used together with the description ontology to create the linked IoT data. Section 5 outlines our distributed storage solution for storing the linked semantic data created according to our ontologies. In Section 6, we show how the description ontology and the linked IoT data are used to design an effective discovery method (in particular, service matchmaking and ranking). Section 7 concludes the paper and briefly points out some future research issues.

2 RELATED WORK ON SEMANTIC MODELLING IN IOT

In this section, we first present the state-of-the-art on recent works on the IoT domain concept modelling followed by a review of relevant works in IoT discovery. Standardisation efforts in the allied areas of sensor description and observation data modelling have been driven by the W3C's Incubator Group on Semantic Sensor Networks [1] and the OGC Sensor Web Enablement [12] suite of XML-based standards. The SSN ontology [3] represents a high-level schema model to describe sensor devices, their capabilities, platform and other related attributes in the semantic sensor networks and the sensor Web applications. The SSN ontology, however, does not include modelling aspects for features of interest, units of measurement and domain knowledge that need to be associated with the sensor data to support autonomous data communications, efficient reasoning and decision making.

The OGC standards suite is aimed at Web accessible sensor networks and archived sensor data that can be discovered and accessed using standard protocols and APIs. The standards consist of modelling schemas (Observation and Measurement and SensorML [13]) and Web Service interfaces (Sensor Alert Service, Sensor Planning Service and Sensor Observation Service) that facilitate the exchange of information through APIs. The research work

[1] http://www.w3.org/2005/Incubator/ssn/

in [7] provides a semantically enabled Sensor Observation Service, called SemSOS, which provides the ability to query high-level knowledge of the environment as well as low-level raw sensor data. 52North's [2] SOS implementation is designed to provide interfaces to sensor observation data stored in a database, with the sensor descriptions stored in XML files. The work presented in [14] proposes an ontology-based model for service oriented sensor data and networks. The ontology consists of three main components: ServiceProperty (functionality), LocationProperty, and PhysicalProperty for contextual and physical characteristics of the sensor nodes in WSN architectures. The system, however, does not specify how sensor data will be described and interpreted in a sensor network application.

The IoT-A project has identified entities, resources and IoT services as key concepts within the IoT domain [15]. The entity is the main focus of interactions by humans and/or software agents. An IoT service exposes resource functionality hosted on devices that provide physical access to the entity. The associated semantic models [9] consist of entities which are modelled to have attributes related to the domain (i.e. observable or actionable features), location attributes as well as other type and identifier specifications. The resource model captures different resource types (e.g. sensor, actuator, RFID tag), hosting device location as well as a link to the service model that exposes the resource capabilities. The service model exposes resource functionalities in terms of the input, output, precondition, and effect. The type of the service specifies the actual technology used to invoke the service (e.g. OWL-S[3], RESTful, etc.).

Semantic modelling is a fundamental process to support interoperability and has important applications in service oriented computing for the IoT. For example, with the emerging practice of exposing IoT sensors and actuators as services [16–18], service discovery has become a prominent topic in IoT research. Service discovery in the IoT is more challenging (e.g., IoT service are mostly less reliable, exposed by devices with limited processing power and operated in highly dynamic environments) than discovery in the enterprise environments where reliable service resources can be abundant. Existing work on service discovery and matchmaking are mostly developed for general Web services and can be grouped into three main categories: logic-based approaches, non-logic-based approaches, and hybrid approaches [19]. Logic-based semantic service discovery approaches [20] use a reasoner to infer new knowledge from the concepts and relationships defined in semantic service descriptions and tend to be accurate. Non-logic-based approaches [21,22] aim to reduce the complexity of matchmaking by analysing service descriptions based on information retrieval techniques. Hybrid Matchmakers [23,24] combine the advantages of Non-Logic-based techniques with the fine grained reasoning capabilities of Logic-based techniques.

3 A COMPREHENSIVE SEMANTIC DESCRIPTION ONTOLOGY

Semantic technologies and service oriented computing principles have been fundamental in recent IoT research to promote interoperability among heterogeneous parties (e.g., IoT data providers and consumers) and to facilitate effective access, integration, discovery and utilisation of the IoT resources and data at large scale. The description ontology developed in our work builds upon and extends the existing efforts in modelling and standardising the IoT domain concepts, and aims to capture most of the important relationships among those concepts. It is designed using a knowledge-driven methodology: concepts that were isolated in previous works are integrated and linked to each other; the ontology also provides constructs that allow linking to concepts in external domain ontologies and creating linked IoT data. In the following sections, we first present the design principles used in our work and then elaborate the design of different ontology modules.

3.1 Characterising IoT Domain Concepts

The IoT semantic description ontology is centred on the concepts of "Resources" and "Services". According to De *et al* [9], "the software component that provides information on an entity or enables controlling of an IoT device is called a *"Resource"*; a *"Service"* provides a well-defined and standardised interface, offering all necessary functionalities for interacting with entities and related processes". Based on the service oriented computing principles as well as the need to access the IoT resources and data, an IoT service is modelled as a virtual concept that is exposed by an IoT resource.

IoT services mostly have limited computation capabilities and their exposing resources often operate in highly dynamic environments. Compared to general Web services, they are less reliable; their logic is much simpler and their output usually represents observation and measurement data of feature of interests associated to physical world objects. For these reasons, a semantic IoT service representation model preferably needs to be lightweight to facilitate computation and to represent the phenomenon related to real world objects. The service model also should be associated with concepts in the existing ontologies and domain knowledge base (e.g., Geonames ontology[4]) or the Linked Open Data[5].

[2] http://52north.org/
[3] http://www.ai.sri.com/daml/services/owl-s/1.2/
[4] http://www.geonames.org/ontology/
[5] http://linkeddata.org/

3.2 Design Principles

The major consideration in our design is to balance the tradeoff between being lightweight and complete. The ontology is designed based on the following four principles:

- *Lightweight*: experiences on ontology development in the past years show that a lightweight ontology model that balances well expressiveness and inference complexity is more likely to be widely adopted and reused.
- *Completeness*: we aim to develop a more complete description ontology for the IoT domain by integrating and extending existing works on IoT modelling. Users of the ontology can exploit the synergy of integration to support common tasks in IoT.
- *Compatibility*: the ontology needs to be consistent with those well designed, existing ontologies to ensure compatibility. Wherever possible, we reuse the existing concepts in well defined semantic IoT ontologies.
- *Modularity*: the designed ontology is developed with a highly modular approach to facilitate its evolution, extension and integration with external ontologies.

The ontology design also reuses some of the existing ontologies and domain models in IoT, in particular:

- *The SSN ontology* [3] is reused to represent the sensor resources. In the proposed semantic description model, the sensor class is defined as a subclass of both IoTResource and ssn:Sensor classes. The sensor resource inherits all the specific properties defined in the SSN ontology.
- *The ontology for Quantity Kinds and Units*[6] is used to describe the observation and measurement data generated in IoT systems.
- *The GeoNames ontology* is used to add geospatial information to the IoT resources. The information can be utilised to design location-aware discovery methods for IoT resources and services.

We also consider some of the existing design patterns during the ontology development as they represent the current best practices and have been widely recognised by the community.

- The design pattern for modelling the IoT domain proposed in [9] is adopted and the information model in [9] is used as the basis for our ontology design.

[6]http://www.w3.org/2005/Incubator/ssn/ssnx/qu/qu

- The service model in our ontology is based on a design pattern called "Profile-Model-Grounding", which is essentially a variant of the OWL-S semantic service design pattern, "Profile-Process-Grounding". Our intention is to design a lightweight and service technology independent semantic service model which has the potential to be widely reused, therefore, the process modelling which contains significant complexity, has been removed from our model.
- The hREST design pattern [25] is a micro format for semantically describing RESTful services and is intended to be used with HTML. hREST is a lightweight service description mode and is suitable for IoT based services (although it is not service technology independent).

3.3 Ontology Overview and Modules

The description ontology contains seven main modules capturing different aspects of the domain, namely, IoT Services, Service Test, QoS and QoI, Deployment, System and Platform, Observation and Measurement, IoT Resources, and Entity of Interest and Physical Locations. Figure 1 shows an overview of the ontology. For some of the modules in the description ontology, only properties are defined; this allows users to link to the concepts in external ontologies or existing linked data.

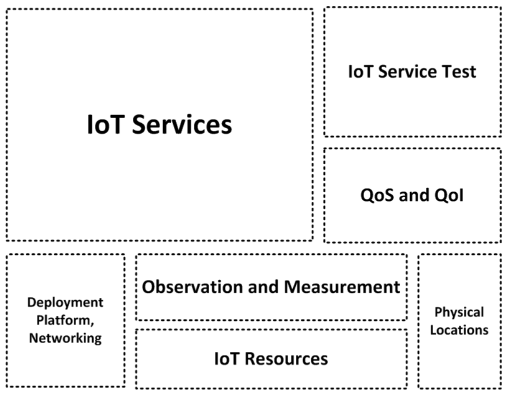

Fig. 1. Modules in the ontology

- *IoT Resources*: existing works for modelling the IoT resources primarily focus on sensors and sensor network [5, 6]. This module extends the SSN ontology [3] by including other important resources in the IoT domain such as Actuator, IoT Gateway and Server.
- *IoT Services*: the scale and distributed nature of the IoT requires scalable and interoperable means for managing and accessing information pertaining to the

entities in our physical world. With the service interfaces exposed by the IoT resources, existing business applications and services need intelligence and context awareness could be easily integrated with the low level IoT services. An important consideration is to model IoT services in a way that adheres to conventions in existing service standards.

- *Quality of Services (QoS) and Quality of Information (QoI)*: QoS and QoI are important concepts in many areas such as networking, communication and Web services. IoT features a vast number of capability constrained and mobile resources that usually operate in harsh and dynamic environments. This makes QoS and QoI particularly important in service composition and adaptation for IoT service providers and consumers.
- *Service Test*: the test components are proposed for testing and verifying functional and non-functional capabilities of IoT services during the design and deployment stages.
- *Deployment, Systems and Platforms*: this module provides descriptions on how the IoT resources are organised and deployed as well as the system they form. Modelling and linking together these concepts enable a high-level view on the relationships among the IoT resources and the systems and platforms that support them.
- *Observation and Measurement*: concepts in this module are used to describe the actual data generated on the IoT. As the data is almost always related to properties of entity of interest (e.g., temperature or speed) and associated with a unit of measurement (e.g., Km/second), we reuse the concepts in quantity kinds and unit from the SSN QU ontology.
- *Entity of Interest and Physical Locations*: entity of interest represents an object in the physical world that is of interest to a user or application. Physical locations are associated with entity of interest and essential for IoT resource and service resolution, lookup and discovery.

Modelling methods for the IoT Resources, Entity of Interest and Physical Locations, Deployment, Systems and Platforms, and Observation and Measurement have been extensively discussed in existing works such as [3, 6, 9]. We extend these works with a particular emphasis on those modules that facilitate us to discover, access, utilise and verify the information generated by the IoT resources. In the following sections, we explain the design for the modules on IoT services (including Service Test), QoS and QoI.

3.4 IoT Services

The IoT service modelling is designed to be independent of particular service technologies. The model is created based on the two most widely used service technologies, i.e., SOAP/WSDL and RESTful. The SOAP/WSDL based services have strong associations with business process modelling and have been widely adopted in the business world, while RESTful style services are data-centric and have been prevalent in Web 2.0 applications recently due to their flexibility and simplicity. RESTful services are usually described using the Web Application Description Language[7] (WADL).

We design the concepts for IoT services based on the analysis of commonalities and distinctiveness in both SOAP/WSDL and RESTful services: a concept for service is defined for each common term in both service technologies (e.g., an operation or input/output parameter); for the term appearing in one while not in the other, a concept is defined if its presence adds extra semantics to the service model (e.g., the InputMessage concept which consists of zero or more input parameters can be referenced to a concept defined in a domain ontology). Some of the concepts are designed as optional, such as the precondition and effect concepts for RESTful services. The OWL-S ontology is a semantic model for SOAP/WSDL services and is designed based on the so-called "Profile-Process-Grounding" pattern. It is noted that much of the complexity originates from the process modelling (as defined as the Process ontology in OWL-S). On the contrary, the hREST model [25] for RESTful services is a simple service ontology: it excludes the profile and grounding modelling which is important for service discovery and access. The service model developed in our work can be seen as a trade-off between these two modelling approaches: it is lightweight and service technology independent, while at the same time providing sufficient modelling constructs to facilitate service computation (e.g., service discovery, adaptation and composition). We refer to our proposed design pattern as "Profile-Model-Grounding". Profile and Grounding are adapted from the OWL-S and refined (so it can also be used for representing RESTful services); the Model excludes the process modelling and is based on the atomic service modelling in OWL-S and RESTful service modelling in hREST [25]. The overall structure of the service model is shown in Figure 2.

- The profile of a service defines the non-functional aspects of a service. It contains properties for linking to semantic concepts in existing knowledge base or taxonomies which are essential for service search and discovery. Besides reusing some of the properties

[7] http://www.w3.org/Submission/wadl/

and attributes from the OWL-S profile model (e.g., service name, service category, contact information, etc), it defines properties linking to concepts on platform, network and deployment which are important and specific for IoT services.

- The model of a service defines the functional aspects of a service. It is developed by identifying and analysing commonalities between different service technologies. It represents a trade-off between the SOAP/WSDL based and RESTful services.
- The concept of grounding explains how to interact with a service by specifying the service and endpoint addresses, communications protocols, etc. It also provides a mapping between the concepts defined in the semantic description ontology and those defined in the service documents such as WSDL (for SOAP based services) or WADL (for RESTful services). These mapping concepts are optional to IoT services which usually do not present service documents.

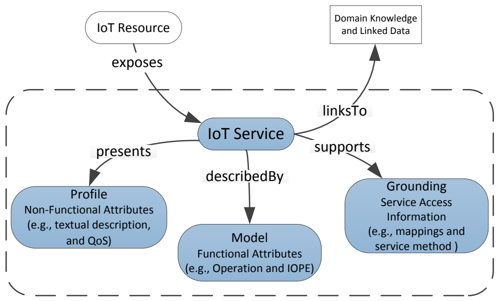

Fig. 2. Overview of the IoT service description model

As an IoT service is a type of a real-world service, so its association to the IoT resource needs to be modelled. In the ontology, a relationship "exposes" (and its reverse property "isExposedBy") is defined between the Resource and Service classes. It should be noted that an IoT resource is usually linked to a geographical concept which can also be used during the service discovery process.

IoT services are exposed by IoT resources that usually operate in harsh and dynamic environments; the resources are mostly capability constrained (e.g. in terms of battery, computing and communication capabilities) and may appear or vanish in many situations; therefore, it is expected that a large number of IoT devices will demand for self-testing capabilities. The dynamic environment also brings significant needs for service adaptation and even recomposition. For this purpose, an IoT service can link to one or more instances of the "Test" class through the property "hasTest" (see [26] for more information on service test).

3.5 Quality of Service and Quality of Information

QoS and QoI have been extensively studied in many areas such as networking and communication [27], Web services [28], and can be used as important criteria for designing complex service composition and adaptation algorithms [29]. They are particularly important for the IoT domain which exhibits a much higher level of dynamicity. In our work we do not try to enumerate and model all the parameters for QoS and QoI since they are often application dependent. Instead we define the parameters that are common to many application domains. In the current version of the ontology, both QoS and QoI are modelled as classes (with a number of subclasses for each) and linked to both IoT Service class and IoT Resource class. QualityOfService is defined as the top-level QoS class that has networking related subclasses (e.g., Throughput and Delay), Availability, Reliability, Security, etc; QualityOfInformation has subclasses such as Correctness, Precision, Provenance, etc, as shown in Figure 3.

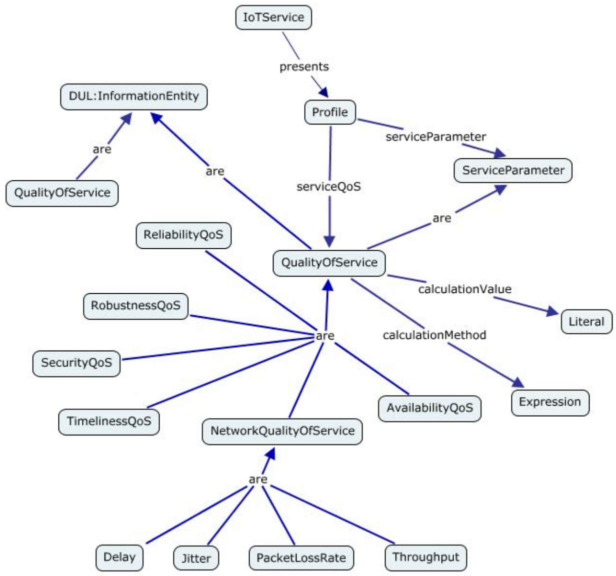

Fig. 3. The Quality of Service and Quality of Information model

All these classes have the properties of "calculationValue" (value of the QoS or QoI paramter) and "calculationMethod" (method for calculating the QoS or QoI value). The range of the "calculationMethod" property is a computation method that can be represented using appropriate expressions or URIs to facilitate the reuse of QoS or QoI information.

3.6 Remarks

We have shown our recent work on development of a comprehensive while lightweight description ontology to capture and represent knowledge for the IoT domain. We pay special attention to the design of the service model which provides a foundation for service oriented computing for IoT. Besides describing the functional properties of the services (e.g., operations, input and output), the description ontology also provides descriptions in terms of many non-functional properties (e.g., IoT resources, platforms, networking parameters, QoS and QoI). More importantly, concepts in the ontology need to link to those in existing domain knowledge.

As pointed out in [2], "providing ontologies and semantic descriptions alone does not provide semantic interoperability and will not solve all the issues regarding discovery, management of data, and supporting autonomous interactions". The important issue is how to use the ontology and semantic models to support common tasks in a semantic and service oriented IoT. In the following sections, we demonstrate the usefulness of the ontology and power of semantic computing with the domain ontologies and the linked IoT data. We also show how the linked IoT data created using the ontologies (i.e., the description ontology and the indoor location ontology) can be used to design service discovery and ranking algorithms in IoT.

4 CREATING LINKED IOT DATA

Following the definition of the concepts describing the IoT domain in the previous section, this section focuses on the issue of annotation and publication of the semantic description data to support various automated scenarios. The usefulness of the semantic data would be limited without linking IoT services and resources to existing domain knowledge and linked data available on the Web. Such linking exposes more information related to a particular data item by exploring the links across different concepts and domains. In our approach, we harness the Linked Data principles in order to create linked IoT data, especially with the use of spatial data. In the IoT service ontology, the range of the hasServiceArea property could either be bounding box coordinates (e.g. a rectangle, represented by the coordinates of the two corners, e.g., NE and SW: {[49.408321, 8.67774], [49.404216, 8.686495]}) or an instance of an location ontology that offers specification of locations (e.g., a city, street, or in terms of indoor environments). As mechanisms for IoT service discovery should include facilities for expressive query against services observing (for sensor services) or affecting (for actuator services) a particular area, detailed proximity information needs to be captured.

While existing linked data sources such as the GeoNames ontology can be used to annotate IoT services with a geographical scope such as cities and organisations, describing coverage information of an indoor environment necessitates a more fine-grained description of the location concept. For instance, consider a university campus consisting of several buildings, with each building containing labs, offices, teaching rooms etc. The rooms could have several deployed IoT resources such as light and presence sensors, exposing the relevant IoT service. While the GeoNames ontology provides user-friendly location names or geographical coordinates and captures the associated contextual information on region containment and distance among locations, the indoor location ontologies could offer similar contextual information for indoor environments and provide such contextual information with much finer granularity. For this purpose, we also developed the indoor location model as depicted in Figure 4.

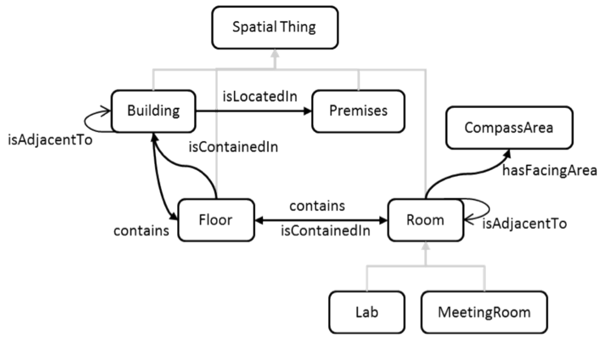

Fig. 4. Indoor location ontology

The model captures indoor location concepts representing objects such as buildings, rooms or other premises, with formal definitions which allow reasoning tasks to be performed. The 'Building', 'Premises', Floor, and 'Room' concepts are subclasses of the upper level concept 'SpatialThing'; 'Lab', 'TeachingRoom' and 'MeetingRoom' are subclasses of 'Room'. The model also specifies the relationships among different concepts, for example, buildings are located in premises (e.g., an instance of a premise could be a university campus) and contain floors. The 'contains' (and its inverse 'isContainedIn') property is asserted to be transitive, which allows inference on region containment of individual rooms within buildings. A room has a 'CompassArea' property whose range can be one of orientation concepts such as, 'North', 'NorthEast', 'East' or 'SouthEast'. To describe the fact that location concepts are situated next to each other, the 'adjacentTo' property is defined to specify the adjacency of objects.

To annotate IoT service instances with location specific information, we employ our Sense2Web platform [9]. The platform has a Web application that allows human users to publish service instance data in terms of linked data. The application can also provide suggestions to the users to interlink to various location instances in existing knowledge bases. When the user keys in, for instance, a room or building identifier, the platform formulates and issues a SPARQL query to the indoor location ontology to retrieve the relevant instances, which are then shown as suggestions in the interface. The platform also offers a machine-to-machine interface for publishing linked data automatically. Figure 5 shows a SPARQL query to retrieve the indoor location instances corresponding to the label 'BA' - in this case, the instance 'BA_building' is retrieved.

```
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX loc:<http://www.surrey.ac.uk/ccsr/ontologies/LocationModel.owl#>
SELECT ?id
WHERE{
          {?id rdf:type loc:Floor}
          UNION {?id rdf:type loc:Building}
          UNION {?id rdf:type loc:Room}
          FILTER regex (str(?id), "BA")
    }
```

Fig. 5. SPARQL query for indoor ontology

The IoT service data publisher may opt to use a different indoor location ontology and it will not affect the connectivity to the IoT domain description ontology and the query mechanism to access the location instances (as long as the ontology schema is available). In situations where detailed indoor location models are not available, the service scope can be defined in terms of geographical bounding boxes or instances in the GeoNames ontology.

5 DISTRIBUTED SEMANTIC DATA STORAGE

Adding semantic descriptions to the large number of "Things" and publishing them as linked IoT data facilitate interoperability among heterogeneous sources and provide a foundation for automated processing and reasoning of the semantic information. However, the resulting tremendous amount of semantic information together with the continuously generated real-time IoT data necessitates highly distributed storage solutions and more efficient query mechanisms [2]. One of the challenges is to promptly locate the storage servers or directories that are able to provide relevant information to queries.

Designing and implementing storage solutions that enable publishing and accessing semantic data in large distributed and dynamic environments (i.e., the IoT) is different from those designed for general Web services [2]. Most of the current research works on semantic Web service computing assume that the semantic description data for services (as well as resources) is stored in a centralised fashion [18, 30] or using the linked data principle [31]. Centralised storage solutions are not appropriate for the IoT domain because of the scalability issues. Publishing semantic descriptions as linked data on a large scale is a promising approach; however, the overhead related to updating and maintaining of the linked IoT data is inconceivable. More importantly, discovery of services and resources using the linked data needs to specify the exact addresses of the semantic repositories that contain potentially relevant information, which is extremely inefficient given the large-scale and dynamic nature of the IoT.

The purpose of the distributed storage is to enable efficient query forwarding and processing and to facilitate discovery. The designs need to identify which storage is likely to provide answers to the queries. The architecture for the distributed semantic storage consists of two types of important entities: directory server (may be organised in a hierarchy) and gateway. Our approach exploits the fact that the queries for IoT resources and services are almost always related to geographical locations. Using the data structure for indexing spatial and geographical data (e.g., R-Tree [32]) as proposed in [33], the directories or gateways (see below for more information about the gateway design) can be indexed with the geographical region that they manage. They are organised in a distributed and hierarchical way and can be matched with the geographical information in the queries efficiently[8]. The design is also based on our previous works on service connectivity (using gateway) [17] and publishing linked IoT data [5]. Our approach assumes that IoT services will be offered by gateways (not IoT resources themselves) as IoT resources are mostly capability-constrained and less reliable. Besides managing a wireless sensor network, a gateway maintains a semantic repository that stores the semantic metadata for the resources and services in that network. During service or resource search and discovery, once a particular gateway (or gateways) is located, the resources and services can be retrieved based on reasoning of the geographical information. Moreover, a SPARQL[9] endpoint is implemented on a gateway to handle requests and retrieve more fine-grained semantic information.

6 SEMANTIC SERVICE DISCOVERY AND RANKING

Service discovery is the most important and challenging task in service oriented computing. The technologies currently used in service discovery are developed for the carefully designed and maintained enterprise Web services and are not suitable for the discovery of IoT services for two main reasons: first, finding services in the right geographical location (and with the right functionality) is of great importance; second, many of the IoT services are exposed by real-world devices which are limited in processing capabilities and energy. In this section, we discuss our

[8]The information can be represented either as a spatial point or region. The matching process checks if the spatial point or region in the query falls into or overlaps with indexed spatial region on the directory server or gateway.

[9]http://www.w3.org/TR/rdf-sparql-query/

method for discovery of IoT services and show how the ontologies (presented in Section 3) and semantic service descriptions can be utilised to facilitate this task.

Service discovery solutions generally consist of three components [19], namely, service representation, discovery architecture and service matchmaking. We have elaborated the description ontology for knowledge representation in the domain of IoT in Section 3 and 4 and the discovery architecture in Section 5. In the following sections we focus on the problems of matchmaking and ranking in automated service discovery.

6.1 Expanding Discovery Results using Linked IoT data

An interesting application of the linked IoT data in service discovery is the query expansion which is able to retrieve more meaningful results (i.e., IoT services). Before discussing the service matchmaking and ranking algorithms, we show how the discovery results can be expanded based on semantic reasoning on the linked IoT data and the indoor location ontology presented in Section 4. Once an initial set of services are found, our method makes use of the semantic inference mechanisms to derive region containment of the indoor location instances and compute the transitive closure of the 'contains' property in the indoor location ontology (Note that this procedure is performed prior to service matchmaking). For example, since all instances of 'buildings' contain 'floors' and floors in turn contain 'rooms', the transitive closure infers that all building instances also contain the room instances therein. As the service areas are linked to instances of the indoor location ontology, queries for services in a particular location automatically benefit from the inferred knowledge. In addition to region containment, the logic matchmaking process also makes use of the relative positioning properties encapsulated in the indoor ontology, i.e. the 'isAdjacentTo' property. This provides another way of finding services which might be 'near' a given location. The structure of the encoded relations is shown in Figure 6, where the 'loc' prefix refers to the indoor ontology namespace.

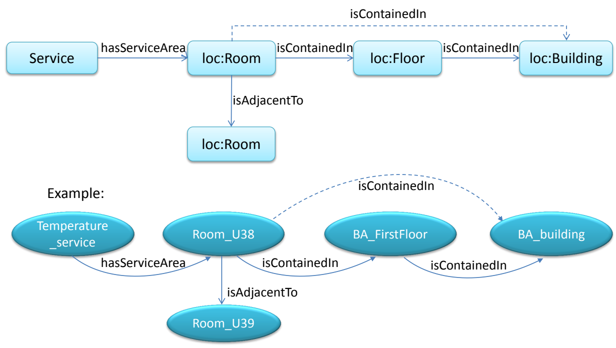

Fig. 6. Region containment and relative positioning

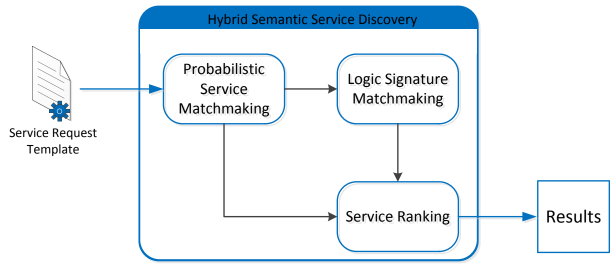

Fig. 7. Components of the hybrid semantic service discovery.

6.2 Matchmaking

The matchmaking method described here is a hybrid matchmaker that builds upon our previous work on service matchmaking [34]. In Section 5 we have discussed how the distributed storage architecture helps limit the scope of search by matching the location information in the query and the indexed geographical location information. Our hybrid matchmaking process works on the set of returned services and aims to find the services most relevant to the query and to rank them in order of relevance (see Section 6.3). Structure of the matchmaker is shown in Figure 7. It uses a non-logic-based probabilistic service matchmaking component [35, 36] to find a short list of more relevant services and a logic-based component that uses individual *Links* between a source parameter and a destination parameter [34] to verify that the services in the short list are compatible with the *IO* signature of the request.

6.3 Service Ranking

Finding services that are highly relevant to a service request is the core function of service discovery; however the way the results are presented to the client is also a matter of great importance. Presenting search results in a ranked order makes service selection easier for the client. There have been a number of works on Web service ranking, for example, Segev and Toch propose an algorithm for ranking possible candidates for service composition based on clustering techniques for context matching [21]; The work in [37] uses QoS parameters as important ranking criteria; however, obtaining the QoS parameters of services is challenging because computing the QoS values for a large list of candidate services is extremely difficult and expensive in automated service discovery; there are also methods

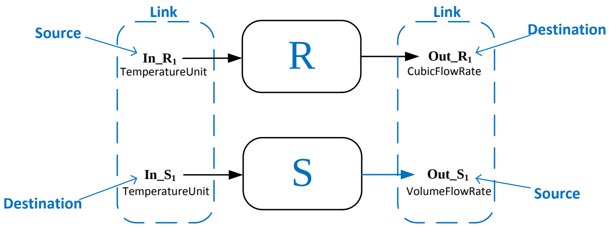

Fig. 8. Source and destination parameters in links

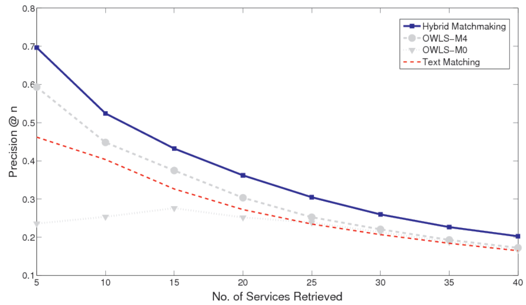

Fig. 9. Averaged $P@n$ values

for ranking Web services based on information retrieval techniques using the service descriptions (mostly based on OWL-S) [23, 38].

Our ranking algorithm is built upon the probabilistic machine learning technique (the probabilistic latent factor analysis in the non-logic-based component) and semantic reasoning (the logic-based component). The non-logic-based component works by computing the degree of match between a service request and a service description in the latent factor space. We map the request templates into the latent factor space using the folding-in techniques as described in [36]. The degree of matching between the probability distribution of latent factors for the request and a service description can be calculated using a vector similarity measure (e.g., the Cosine similarity). The services that score the highest degree of matching to the query are stored in a short candidate list. The length of this list can be specified by the client.

The shortened list from the probabilistic component is then passed to the logic-based component, which subsequently computes the degree of matching between a service and a service request by analysing the *Links* [34] between source and destination parameters (shown in Figure 8).

Individual link analysis makes it possible to dissect the degree of match between a service and request in a finer grained way than *IO* matchmaking filters [39]. The rationale of the approach is that the most important part in a service request are the outputs and as long as all the required outputs can be provided by a service, it doesn't matter if the service can produce extra outputs that will not be used. Similarly, if a request specifies that the client is capable of supplying certain parameters as inputs, it doesn't matter if the service found only requires a subset of these available inputs to work. The matchmaking process works by assigning weights to individual links and the degree of match between a service and a request is then given by summing together the weights of the individual links [34]. Our ranking mechanism ranks the results primarily based on their weighted-link score, *i.e.,* the service with the highest weighted link score is the most relevant to the service request. If two or more services have the same ranking, the score derived from the probabilistic component is used for further ranking.

6.4 Performance Evaluation

We perform a comparative study on the service discovery methods using the OWL-S service retrieval test collection OWLS-TC v3.0[10] (which consists of 1007 services). The services are divided into seven categories and a total of 29 queries are provided together with a relevant answer set for each query. The hybrid service discovery method was compared to a text-matching approach powered by Apache Lucene[11] and also methods from the OLWS-MX 2.0[12] hybrid semantic Web service matchmaker (e.g., the OWLS-M0 and OWLS-M4, respectively) [40]. OWLS-M0 is a logic-based approach and OWLS-M4 is a hybrid approach which uses both logic and non-logic based methods. We evaluated our matchmaking approach by calculating the *Precision at n* ($P@n$) [41], which is a standard evaluation techniques used in *Information Retrieval* to measure the accuracy of a search mechanism with respect to completeness of the returned results.

The averaged precision at N results are shown in Figure 9. These results show that our discovery method outperforms all the other state-of-the-art service discovery methods in terms of precision at N. Currently, the experiments are performed using a single dataset in a centralised fashion; we aim to extend the experiments and evaluation using distributed semantic datasets.

7 CONCLUSION AND FUTURE WORK

Modelling using semantic technologies has shown considerable effectiveness for supporting interoperability among distributed and heterogeneous sources on the IoT. Recently, the research trend has shifted from IoT devices and resources to IoT information, since the ultimate goal

[10] http://www.semwebcentral.org/projects/owls-tc/
[11] http://lucene.apache.org/
[12] http://semwebcentral.org/projects/owls-mx/

of the IoT research is to enable ubiquitous access and utilisation of the physical world information, especially for high level business services and applications that need context awareness and intelligent decision making. An interesting idea in this line is to provide IoT information through standard service interfaces, which coincides with the service oriented computing paradigm and potentially ensures scalability. To this end, a description ontology (that balances the tradeoff between being comprehensive and lightweight) is needed to capture and represent service and others important concepts in the IoT domain. The description ontology we present here integrates the existing efforts for modelling the IoT domain concepts and is extended with essential concepts such IoT service, test, and QoS/QoI (which is particularly important for IoT based service composition and adaptation).

We recognise the fact that creating a comprehensive ontology only does not provide significant contributions to the research for IoT and the most important issue is how to use the ontology to support important tasks in a semantic and service oriented IoT. We demonstrate the applications of our ontology through a number of scenarios. In the first scenario, we show how the ontology can be linked together with other ontologies (e.g., the indoor location and position ontology) to create linked IoT data. Based on the linked IoT data we then explain how more effective semantic reasoning can be performed in expanding the results for IoT services and resources discovery. In the second scenario, we present the design of distributed repositories based on the geographical information available in the semantic linked data. The distributed semantic data storage is the underlying platform for the tasks of service discovery and composition. Finally, we present the design of our IoT service discovery and ranking methods based on the description ontology and the linked IoT data. Our future research aims to improve the current service ranking algorithm using the contextual information available for the IoT resource and to design efficient service composition methods to integrate the physical world services and existing business services.

ACKNOWLEDGMENT

The first and second authors are funded by the EU FP7 Programme IoT.est project: Internet of Things Environment for Service Creation and Testing (http://ict-iotest.eu/iotest/), contract number: 288385. The third author is funded by the EU IoT-A project: Internet of Things - Architecture (http://www.iot-a.eu/) contract number: 257521.

REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey,” *Computer Networks*, vol. 54, pp. 2787–2805, 2010.

[2] P. Barnaghi, W. Wang, C. Henson, and K. Taylor, “Semantics for the Internet of Things: early progress and back to the future,” *International Journal on Semantic Web and Information Systems, special issue on sensor networks, Internet of Things and smart devices*, vol. 8, pp. 1–21, 2012.

[3] M. Compton, P. Barnaghi, L. Bermudez, R. G. Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog, V. Huang, and K. Janowicz, “The SSN Ontology of the Semantic Sensor Networks Incubator Group,” *Journal of Web Semantics*, 2011.

[4] C. Bizer, T. Heath, and T. Berners-Lee, “Linked Data - The Story So Far,” *International Journal on Semantic Web and Information Systems*, vol. 5, no. 3, pp. 1–22, 2009.

[5] P. Barnaghi, M. Presser, and K. Moessner, “Publishing Linked Sensor Data,” in *Proceedings of the 3rd International Workshop on Semantic Sensor Networks (SSN), Organised in conjunction with the International Semantic Web Conference (ISWC) 2010*, 2010.

[6] J. Pschorr, C. Henson, H. Patni, and A. Sheth, “Sensor discovery on linked data,” in *Proceedings of the 7th Extended Semantic Web Conference, ESWC2010*, 2010.

[7] C. A. Henson, J. K. Pschorr, A. P. Sheth, and K. Thirunarayan, “SemSOS: Semantic Sensor Observation Service,” in *Proceedings of the 2009 International Symposium on Collaborative Technologies and Systems (CTS 2009), Baltimore, MD*, May 2009.

[8] OASIS, “Reference model for service oriented architecture,” 2006.

[9] S. De, T. Elsaleh, P. Barnaghi, and S. Meissner, “An Internet of Things Platform for Real-World and Digital Objects,” *Journal of Scalable Computing: Practice and Experience*, vol. 13, no. 1, 2012.

[10] A. Serbanati, C. M. Medaglia, and U. B. Ceipidor, “Building blocks of the internet of things: State of the art and beyond,” in *Deploying RFID - Challenges, Solutions, and Open Issues, C. Turcu, Ed.*, InTech, 2011.

[11] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, “Internet of things: Vision, applications and research challenges,” *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497–1516, 2012.

[12] OGC White paper, “Open Geospatial Consortium (OGC) Sensor Web Enablement: Overview and High Level Architecture,” 2007.

[13] OGC Specification, “OpenGIS Sensor Model Language (SensorML) Implementation Specification,” 2007.

[14] J.-H. Kim, H. Kwon, D.-H. Kim, H.-Y. Kwak, and S.-J. Lee, “Building a Service-Oriented Ontology for Wireless Sensor Networks,” in *Proceedings of the Seventh IEEE/ACIS International Conference on Computer and Information Science (icis 2008)*, ICIS ’08, pp. 649–654, 2008.

[15] C. M. et al, “IoT-A project Deliverable D1.4 - Converged architectural reference model for the IoT v2.0,” 2012.

[16] F. Chen, C. Ren, J. Dong, Q. Wang, J. Li, and B. Shao, "A Comprehensive Device Collaboration Model for Integrating Devices with Web Services under Internet of Things," in *Web Services (ICWS), 2011 IEEE International Conference on*, pp. 742 –743, july 2011.

[17] W. Wang, P. Barnaghi, G. Cassar, F. Ganz, and P. Navaratnam, "Semantic Sensor Service Networks," in *Proceedings of the IEEE Sensors 2012 Conference*, 2012.

[18] D. Guinard, V. Trifa, S. Karnouskos, P. Spiess, and D. Savio, "Interacting with the SOA-Based Internet of Things: Discovery, Query, Selection, and On-Demand Provisioning of Web Services," *IEEE Trans. Serv. Comput.*, vol. 3, pp. 223–235, July 2010.

[19] M. Klusch, "Chapter 4: Semantic Web Service Coordination," in *CASCOM: Intelligent Service Coordination in the Semantic Web*, 2008.

[20] T. Pilioura and A. Tsalgatidou, "Unified publication and discovery of semantic web services," *ACM Trans. Web*, vol. 3, pp. 11:1–11:44, July 2009.

[21] A. Segev and E. Toch, "Context-Based Matching and Ranking of Web Services for Composition," *IEEE Transactions on Services Computing*, vol. 2, no. 3, pp. 210–222, 2009.

[22] C. Platzer, F. Rosenberg, and S. Dustdar, "Web service clustering using multidimensional angles as proximity measures," *ACM Trans. Internet Technol.*, vol. 9, no. 3, pp. 1–26, 2009.

[23] S.-L. Pan and Y.-X. Zhang, "Ranked Web Service Matching for Service Description Using OWL-S," *Web Information Systems and Mining, 2009. WISM 2009. International Conference on*, pp. 427 –431, nov. 2009.

[24] G. Fenza, V. Loia, and S. Senatore, "A hybrid approach to semantic web services matchmaking," *Int. J. Approx. Reasoning*, vol. 48, pp. 808–828, August 2008.

[25] J. Kopecký, K. Gomadam, and T. Vitvar, "hRESTS: An HTML Microformat for Describing RESTful Web Services," in *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, vol. 1 of *WI-IAT '08*, pp. 619–625, IEEE Computer Society, 2008.

[26] W. Wang, S. De, R. Toenjes, E. Reetz, and K. Moessner, "A Comprehensive Ontology for Knowledge Representation in the Internet of Things," in *Proceedings of the 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, TRUSTCOM '12, (Washington, DC, USA), pp. 1793–1798, IEEE Computer Society, 2012.

[27] A. Bar-Noy, G. Cirincione, R. Govindan, S. Krishnamurthy, T. F. LaPorta, P. Mohapatra, M. J. Neely, and A. Yener, "Quality-of-information aware networking for tactical military networks.," in *PerCom Workshops*, pp. 2–7, IEEE, 2011.

[28] J. Cardoso, A. Sheth, J. Miller, J. Arnold, and K. Kochut, "Quality of Service for Workflows and Web Service Processes," *Journal of Web Semantics*, vol. 1, no. 3, 2004.

[29] T. Yu, Y. Zhang, and K.-J. Lin, "Efficient algorithms for Web services selection with end-to-end QoS constraints," *ACM Trans. Web*, vol. 1, no. 1, 2007.

[30] F. Colasuonno, A. Ragone, L. L. Scorcia, T. D. Noia, and E. D. Sciascio, "jUDDI+: A semantic web services registry enabling semantic discovery and composition," in *Proceedings of 2006 IEEE Joint Conference on E-Commerce Technology (CEC06) and Enterprise Computing, E-Commerce and E-Services (EEE06*, pp. 442–444, 2006.

[31] C. Pedrinaci, D. Liu, M. Maleshkova, D. Lambert, J. Kopecky, and J. Domingue, "iServe: a linked services publishing platform," in *Ontology Repositories and Editors for the Semantic Web Workshop at The 7th Extended Semantic Web*, vol. 596, June 2010.

[32] A. Guttman, "R-trees: a dynamic index structure for spatial searching," *SIGMOD Rec.*, vol. 14, pp. 47–57, June 1984.

[33] IoT-A project, "Deliverable 4.1: Concepts and Solutions for Identification and Lookup of IoT Resources," 2012.

[34] G. Cassar, P. Barnaghi, W. W., and K. Moessner, "A Hybrid Semantic Matchmaker for IoT Services," in *Internet of Things (iThings), 2012 IEEE International Conference on*, nov. 2012.

[35] G. Cassar, P. Barnaghi, and K. Moessner, "Probabilistic Methods for Service Clustering," in *International Semantic Web Conference*, Springerlink, 2010.

[36] G. Cassar, P. Barnaghi, and K. Moessner, "A Probabilistic Latent Factor approach to service ranking," in *Intelligent Computer Communication and Processing (ICCP), 2011 IEEE International Conference on*, pp. 103 –109, aug. 2011.

[37] L. Meng, J. Li, and H. Sun, "WSRank: A Collaborative Ranking Approach for Web Service Selection," in *Computer and Information Technology (CIT), 2011 IEEE 11th International Conference on*, pp. 103 –108, 31 2011-sept. 2 2011.

[38] O. Mola, P. Emamian, and M. Razzazi, "A Vector Based Algorithm for Semantic Web Services Ranking," *Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on*, pp. 1 –5, apr. 2008.

[39] M. Klusch and P. Kapahnke, "iSeM: Approximated Reasoning for Adaptive Hybrid Selection of Semantic Services," in *Semantic Computing (ICSC), 2010 IEEE Fourth International Conference on*, pp. 184 –191, sept. 2010.

[40] M. Klusch, B. Fries, and K. Sycara, "Automated semantic web service discovery with OWLS-MX," in *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, AAMAS '06, (New York, NY, USA), pp. 915–922, ACM, 2006.

[41] C. Buckley and E. M. Voorhees, "Evaluating evaluation measure stability," in *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '00, (New York, NY, USA), pp. 33–40, ACM, 2000.

Wei Wang is a research fellow at the Centre for Communication and Systems Research at the University of Surrey. He obtained his PhD in Computer Science from University of Nottingham in 2009 and BSc in Computer Science from the same university in 2006. Before joining the University of Surrey, he worked as an Assistant Professor at the University of Nottingham Malaysia Campus. His research interests include semantic search, Internet of things, service computing, ontology learning from unstructured text, knowledge acquisition, data mining, and machine learning.

Suparna De is a post-doctoral Research Fellow at the Centre for Communication Systems Research at the University of Surrey, UK. Since completing her PhD at the University of Surrey in 2009, she has been involved in several EU projects such as IoT-A, iCore and IoT.est which investigate various aspects of service provisioning in the Internet of Things domain. Suparna did her BSc. in Physics (Honours) at St. Stephen's College, Delhi University, India and her MSc in Communications Networks and Software at the University of Surrey. He research interests include discovery and retrieval methods, semantic association analysis and knowledge engineering methods.

Gilbert Cassar is a Research Fellow in the Centre for Communication Systems Research (CCSR) at the University of Surrey. His research interests include Internet of Things, Web services, service composition, machine learning, and probabilistic graphical models.

Klaus Moessner is a Professor in Mobile Communications in the Centre for Communication Systems Research at the University of Surrey, UK. Klaus earned his Dipl-Ing (FH) at The University of Applied Sciences in Offenburg, Germany, an MSc from Brunel University, UK and his PhD from the University of Surrey (UK). His research interests include reconfigurability of different system levels, as well as service provision in the IoT. He is involved in investigation and teaching of mobile service platforms, service oriented architectures, mobile service delivery and service enablers.

AUTHORS' ADDRESSES

Wei Wang, Ph.D.
Suparna De, Ph.D.
Gilbert Cassar, Ph.D.
Prof. Klaus Moessner, Ph.D.
Centre for Communication Systems Research (CCSR),
University of Surrey,
Surrey GU27XH, GB-GU2, Guildford, United Kingdom,
email: {wei.wang,S.De,G.Cassar,K.Moessner}@surrey.ac.uk

Received: 2012-11-10
Accepted: 2013-03-15