2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology

Location-Based Semantic Matchmaking In Ubiquitous Computing

Michele Ruta, Floriano Scioscia, Eugenio Di Sciascio, Giacomo Piscitelli
Politecnico di Bari
Via Re David 200
I-70125, Bari, ITALY
{m.ruta, f.scioscia, disciascio, piscitel}@poliba.it

Abstract

Ever increasing efforts are spent in developing techniques and tools for a full exploitation of semantics in mobile environments, able to overcome volatility and resource limitations of mobile contexts. This paper presents a platform-independent mobile semantic discovery framework as well as a working prototypical implementation which enables advanced knowledge-based services taking into account user's location. The proposed approach is clarified and motivated in a ubiquitous tourism case study, where some evaluations are presented to prove its feasibility and usefulness.

1. Introduction

Several issues concerning traditional service discovery are exasperated in "evanescent" scenarios such as ubiquitous environments, because of both host mobility and limited capabilities of mobile devices. Hence, many people equipped with handheld devices tend to prefer traditional fixed discovery channels so renouncing to an instant fruition of resources or services. Nevertheless the rising potentialities of wireless-enabled PDAs provide the needed basic requirements for implementing flexible discovery frameworks. Semantic Web technologies applied to resource retrieval, open new possibilities, including: (i) formalization of annotated descriptions that become machine understandable so enabling interoperability; (ii) reasoning on descriptions and inference of new knowledge; (iii) validity of the Open World Assumption (OWA) (what is not specified has not to be interpreted as a constraint of absence) [1], overcoming limits of structured data models.

This paper presents a general framework which enables a semantic-based discovery in ubiquitous environments. It has been implemented in an application for decision support purposes, presented here with reference to a u-tourism (ubiquitous tourism) [2] case study. Main features of the proposed approach are: (i) full exploitation of non-standard inferences presented in [3] to enable query refinement; (ii) semantic-based ranking of retrieved resources; (iii) fully graphical and usable interface with no prior knowledge of any logic principles; (iv) no physical space-temporal bounds in system exploitation.

The remaining of the paper is structured as follows: in the next section, essential background is revised, Section 3 describes framework and approach with the aid of the case study in Section 4. Some evaluation about the system are reported in Section 5 before concluding the paper.

2. Semantic matchmaking background

In this paper we exploit Description Logics (DLs), a family of logic formalisms for Knowledge Representation [1] (we suppose the reader be familiar with DL basics). In particular, we refer to the $\mathcal{ALN}$ (Attributive Language with Unqualified Number Restrictions) Description Logic, a subset of OWL DL having polynomial computational complexity for standard and non-standard inferences.

DL reasoners provide at least two basic reasoning services: concept *subsumption* (a.k.a. classification) and concept *satisfiability* (a.k.a. consistency) [1]. Given R (for Request) and O (for Offer) both consistent w.r.t. a common ontology $\mathcal{T}$ (containing axioms that model knowledge for the reference problem domain), logic-based approaches to matchmaking proposed in the literature [4] use classification and consistency to identify *full matches* (all features requested in R are provided by O) and *mismatches* (a.k.a. *partial* or *disjoint matches*: some features requested in R are conflicting with some other ones offered in O).

Full matches are the best ones from a requester's standpoint, but they are infrequent in practical scenarios. *Concept Abduction Problem (CAP)* and *Concept Contraction Problem (CCP)* [3] non-standard inference services are adopted in this work to enable a semantic matchmaking framework, granting three significant advantages: (1) it is possible to support non-full matches and to compute a logic-based ranking of resources best approximating the request; (2) an explanation of matchmaking outcome can be provided, which is highly desirable in order to increase user's trust in the system; (3) exploiting the open world semantics, the characteristics B (for *bonus*) [5] specified in O but not requested in R represent knowledge that can be elicited and proposed to the requester in a query refinement process.

3. System Outline

The overall system architecture is depicted in Fig. 1. A classical client/server paradigm is adopted. The resource server is a fixed host over the Internet, exposing an enhanced DIG [6] interface. The mobile client is connected through

978-0-7695-4191-4/10 $26.00 © 2010 IEEE
DOI 10.1109/WI-IAT.2010.300

wireless technologies, such as IEEE 802.11 or UMTS/CDMA.

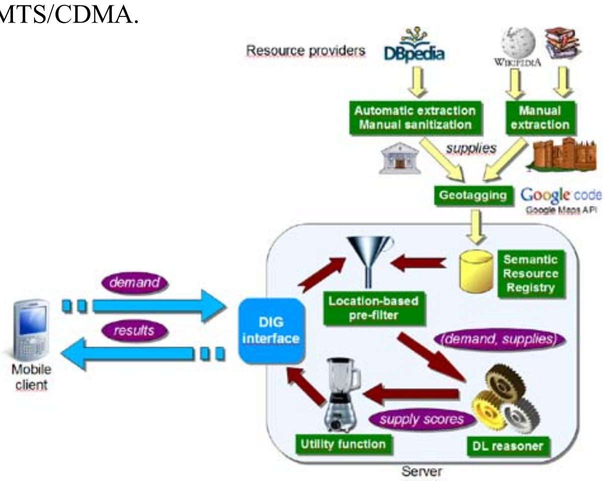

Figure 1. Architecture of the system prototype.

Available resources were collected from several sources. The DBpedia [1] RDF (Resource Description Framework) knowledge base, which is extracted from structured information within Wikipedia, was used to automatically obtain relevant information for many entries. Extracted resource profiles were then manually sanitized (*e.g.* by removing textual abstracts, redundant and unnecessary information) and matched with our custom ontology for the cultural heritage domain through a semi-automatic procedure. Other sources, such as books and Web sites were used for resources not included in DBPedia. Subsequently, each semantically annotated resource was geographically tagged exploiting the Google Maps API. Finally, all resources were stored into a *semantic-based resource registry*. Each resource record contains: (i) a semantic annotation (in DIG language); (ii) a numeric ontology identifier, marking the domain ontology the annotation refers to; (iii) a set of data-oriented attributes, including a (latitude, longitude) pair of geographical coordinates; (iv) a set of user-oriented attributes (in the current prototype, a picture and a textual description).

Resource matchmaking can be carried out only among requests and supplied resources sharing the semantics of descriptions, *i.e.* referring to the same ontology. Hence a preliminary agreement between client and server is required. Unique ontology identifier codes are used for this purpose [7]. Then the client can submit her *request*, which consists in: (i) a DIG expression of the required resource features; (ii) geographical coordinates of the current device's location; (iii) maximum acceptable resource distance. When a request is received, server components perform the following processing steps: (1) resources referring to the same ontology are extracted from the registry; (2) request and resources are fed to the location-based pre-filter, which excludes resources outside the maximum range (see the subsection below); (3) semantic distance between request and each in-range supplied resource is computed by the DL reasoning engine; (4) results of semantic matchmaking are transferred to the utility function calculation module, which computes the final ranking. Finally, the ranked list of best resource records is sent back to the client in a DIG reply.

[1] The DBpedia Data Set, available at http://wiki.dbpedia.org

3.1. Location-based resource filtering

In mobile scenarios, semantic-based matchmaking of annotated resource/service descriptions should be extended to take location into account, in order to provide an overall match degree that best suits the user needs in her current situation. The multi-attribute resource ranking problem can be solved by means of *utility functions*, a.k.a. *Score Combination Functions (SCF)*.

The framework devised in this paper integrates DL-based matchmaking with GPS-based resource and user positioning. *Semantic score* f_{ss} and *geographic score* f_{gs} are combined by the SCF f_{sc}. The operating principle is simple: a circular area is identified, with center in the user's position and maximum accepted range R (both supplied in the user request). Distance d is computed between the user and each resource, and the service provider will pass to the next processing stage only resources located within this area.

The semantic score is computed as:

$$f_{ss}(r,s) = \frac{s\_match(r,s)}{\max(s\_match)}$$

where *s_match(r,s)* is the semantic match distance from request r to resource s, computed by means of the above inference services; $\max(s\_match) \doteq s\_match(r,\top)$ is the maximum semantic distance, which depends on axioms in the reference domain ontology. Hence, $f_{ss} \in [0,1]$ and lower values are better.

The second local score takes into account the physical distance. It is defined as:

$$f_{gs}(d) = \frac{d}{R}$$

Therefore, also $f_{gs} \in [0,1]$ and lower values are preferable. It is useful to point out that in both local scoring functions denominators are maximum values depending on a given user request. They may change across different retrieval sessions, but correctly rank resources w.r.t. the request within the same session.

Finally, the SCF is defined as:

$$f_{sc}(d,S) = 100 \cdot [1 - (f_{gs} + \varepsilon)^{\alpha \frac{R}{\beta}} \cdot (f_{ss} + \gamma)^{1-\alpha}]$$

It is a monotonic function allowing a consistent resource ranking, and it converts results to a more user-friendly scale with a higher outcome representing a better result. A *tuning* phase can be performed to determine parameter values following requirements of a specific discovery application. In detail, $\alpha \in [0,1]$ weighs the relevance of both semantic and geographic factors, respectively. With $\alpha \to 0$ we privilege the semantic score, whereas with $\alpha \to 1$ we make the geographic one more significant. The exponent of the geographic factor is multiplied by R/β. This is because, when the maximum search range R grows, distance must reasonably become a more selective attribute, giving more relevance to resources in the user's immediate proximity. The coefficient β regulates the curve decay.

Parameters $\varepsilon \in [0,1]$ and $\gamma \in [0,1]$ control the outcome in case of either semantic or geographic *full match*. Semantic full match is when all required features in the request are satisfied by the resource. Geographic full match is when the user is located exactly in the same place of resource she is looking for. Both cases are likeable but very unlikely in practical scenarios. Hence, in the model adopted for system evaluation we could pose $\varepsilon = \gamma = 0$:

$$f_{sc}(d,S) = 100 \cdot [1 - (f_{gs})^{\alpha \frac{R}{\beta}} \cdot (f_{ss})^{1-\alpha}]$$

This means that full matches will always be shown at the top of the result list, since either $f_{gs} = 0$ or $f_{ss} = 0$ implies $f_{sc} = 100$ regardless of the other factors.

4. Case Study

For a greater compatibility with various mobile computing platforms, our prototypical client tool was developed using Java Micro Edition (ME) technology. The GUI (Graphical User Interface) was entirely based on SVG (Scalable Vector Graphics), using the Scalable 2D Vector Graphics API JSR-226[2]. In order to allow location-based service/resource provisioning, the application exploits the Java Location API JSR-179[3]. The proposed tool supports a subset of the DIG 1.1 interface extended for MaMaS-tng[4] reasoner.

Functional and non-functional features of the proposed system are motivated within a concrete case study in the cultural heritage tourism sector. Semantic annotations of touristic points of interest in Apulia region were modeled in the service provider knowledge base, along with their position coordinates. The designed mobile client is task-oriented. It assists the user in the discovery process through the following three main tasks (as depicted in Fig. 2).

Ontology management. Different domain ontologies are used to describe general resource classes (*e.g.* accommodation, cultural heritage, movie/theatre shows). At application startup, a selection screen is shown (Fig. 3), with a list of managed ontologies. Each Ontology is labeled by a Universally Unique IDentifier (OUUID) which allows an early agreement between user and provider [7].

Semantic request composition. Fig. 4 shows the ontology browsing screen. A scrollable list shows the current *focus* in the classification induced by terminological definitions and subsumptions. Directional keys of mobile device or swipe gestures on the touchscreen are used to browse the taxonomy by expanding an item or going back one level. Above the list, a *breadcrumb* control is displayed, so that the user can orient himself even in deeper ontologies.

[2] Scalable 2D Vector Graphics API for Java Micro Edition (JSR 226): http://jcp.org/en/jsr/detail?id=226

[3] Location API for Java Micro Edition (JSR 179), http://jcp.org/en/jsr/detail?id=179

[4] The MatchMaking Service: http://sisinfab.poliba.it/MAMAS-tng/

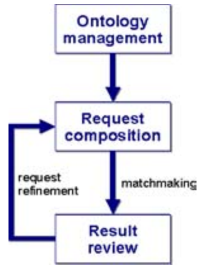

Figure 2. Workflow of user interaction with the mobile client

Figure 3. Ontology selection screen

Figure 4. Ontology browsing screen

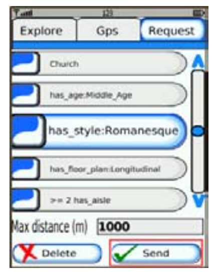

Figure 5. Request confirmation screen

Figure 6. Displayed results

Figure 7. Result details screen

TABLE I. AXIOMS IN THE CASE STUDY TOY ONTOLOGY

AD ⊑ Age	BC ⊑ Age	Middle Age ⊑ AD
Centralized ⊑ Floor Plan	Longitudinal ⊑ Floor Plan	Quadrangular ⊑ Floor Plan
Square ⊑ Quadrangular	Byzantine ⊑ Style	Romanesque ⊑ Style
Gothic ⊑ Style	Baroque ⊑ Style	Portal ⊑ Architectural Element
Cathedra ⊑ Architectural_Element	Aisle ⊑ Architectural_Element	Altar ⊑ Architectural_Element
Pulpit ⊑ Architectural Element	Crypt ⊑ Architectural Element	Apse ⊑ Architectural Element
Window ⊑ ArchitecturalElement	Single_Light ⊑ Window	Double_Light ⊑ Window
Triple _Light ⊑ Window	Religious ⊑ Destination	Private ⊑ Destination
Public ⊑ Destination	Private ⊑ ¬Public	Private ⊑ ¬Religious
Building ⊑ ∃ has age ⊓ ∃ has floor plan ⊓ ∃ has style		
Residence ⊑ Building ⊓ ∃ Destination ⊓ ∀ Destination.Private		
Church ⊑ Building ⊓ ∃ Destination ⊓ ∀ Destination.Religious ⊓ ∃ has_altar ⊓ ∀ has altar.Altar		
Castle ⊑ Residence		

Let us suppose a user located in the city centre of Bari would like to visit a Romanesque Middle Age church, with longitudinal floor plan and two aisles. W.r.t. the cultural heritage ontology (partially reported in Table I for the sake of brevity), the request can be formally expressed as:

D: Church ⊓ ∀ has_age.Middle_Age ⊓ ∀ has_floor_plan.Longitudinal ⊓ ≥2 has_aisle ⊓ ∀ has_style.Romanesque

in DL classical notation. The user can select desired features (corresponding to ontology concepts and role constructors) to build her request. Finally, the tabs on top of the screen allow user to switch from the Explore screen to the Request confirmation one (Fig. 5). There the user can remove previously selected features. Eventually, she specifies a retrieval distance threshold *R* and submits her request to the resource provider.

Results review and query refinement. The server processes the request as explained in Section 3. For example, let us consider the following resources in the knowledge base of the provider:

S1: Basilica of St. Nicholas, Bari (distance from user: d = 0.9 km). A Romanesque Middle Age church, with longitudinal floor plant, an apse, two aisles, three portals and two towers. Other notable elements are its crypt, altar, cathedra and Baroque ceiling. W.r.t. domain ontology, this is expressed as:

Church ⊓ =2 has_aisle ⊓ ∀ has_age.Middle_Age ⊓ ∀ has_style.Romanesque ⊓ =1 has_apse ⊓ =3 has_portal ⊓ =1 has_crypt ⊓ =1 has_altar ⊓ =2 has_tower ⊓ =1 has_cathedra ⊓ ∃ ceiling_style ⊓ ∀ ceiling_style.Baroque ⊓ ∀ has_floor_plan.Longitudinal

S2: Norman-Hohenstaufen Castle, Bari (d = 0.57 km). It is described as a Middle Age castle, with Byzantine architectural style and a quadrangular plan with four towers.

Castle ⊓ ∀ has_floor_plan.Quadrangular ⊓ =4 has_tower ⊓ ∀ has_style.Byzantine ⊓ ∀ has_age.Middle_Age

S3: Church of St. Scholastica (d = 1.3 km). It is described as a Romanesque Middle Age church, with longitudinal floor plan, three aisles, an apse and a tower.

Church ⊓ ∀ has_style.Romanesque ⊓ ∀ has_age.Middle_Age ⊓ ∀ has_floor_plan.Longitudinal ⊓ =3 has_aisle ⊓ =1 has_tower ⊓ =1 has_apse

S4: Church of St. Mark of the Venetians, Bari (d = 0.65 km). It is described as a Romanesque Middle Age church with two single-light windows and a tower.

Church ⊓ ∀ has_style.Romanesque ⊓ ∀ has_window.Single_Light ⊓ =2 has_window ⊓ ∀ has_age.Middle_Age ⊓ =1 has_tower

Results are ranked according to their semantic and physical distance from the request and best matching ones are returned to the user.

TABLE II. MATCHMAKING RESULTS

Supply	Match type	s_match [max =54]	Outcome	Score [α=0.5,β=1 γ=0.014, ε=0]
S1: Basilica of St. Nicholas	Full	0	Hypothesis H: ⊤ Bonus B: =1 has_apse ⊓ =3 has_portal ⊓ =1 has_crypt ⊓ =1 has_altar ⊓ =2 has_tower ⊓ =1 has_cathedra ⊓ ∃ ceiling_style ⊓ ∀ ceiling style.Baroque	88.8
S4: Church of St. Mark	Potential	3	Hypothesis H: ≥2 has_aisle ⊓ ∀ has_floor_plan. Longitudinal Bonus B: =1 has_tower ⊓ =2 has_window ⊓ ∀has_window.Single_Light	78.3
S2: Norman-Hohenstaufen Castle	Partial	11	Give up G: Church Keep K: Building ⊓ ∀has_age.Middle_age Hypothesis H: ∀has_floor_plan.Longitudinal ⊓ ≥2 has_aisle ⊓ ∀has_style.Romanesque Bonus B: =4 has_tower ⊓ ∀has_style.Byzantine	64.8
S3: Church of St. Scholastica	N.A.	N.A.	Discarded due to distance	N.A.

Tab. II reports matchmaking results for the above example. **S3** is discarded in the location-based pre-filter step, as its distance from the user exceeds the limit, even though it would result in a full match. **S1** is a full match with the request, because it explicitly satisfies all user requirements. On the other hand, **S4** is described just as Romanesque Middle Age church, therefore due to OWA it is not specified whether it has a longitudinal floor plan with aisles or not: these characteristics become part of the *Hypothesis* computed through CAP. Finally, **S2** produces a partial match with user request, since it refers to a castle: this concept is incompatible with user request, so it forms the *Give Up* feature computed through CCP. Overall scores of advertised resources are finally computed. The result screen is reported in Fig. 6. Retrieved resources are listed – best matching first – along with their overall scores. When the user selects a resource, its picture is shown as in Fig. 7 along with address, distance from the user and semantically relevant properties contributing to the outcome. If the user is not satisfied with results, she can go back to the ontology browsing screen to modify the request. Furthermore, she can select some elements of the *Bonus* (respectively *Give Up*) list in the result screen and they will be added to (resp. removed from) the request.

5. Conclusion and Future Work

We have presented a framework for a semantic-enabled resource discovery in ubiquitous computing. It has been implemented in a visual mobile decision support system able to retrieve resources/services through a fully dynamic wireless infrastructure. It recognizes via GPS the user location and grades matchmaking outcomes according to vicinity criteria. Future work aims at simplifying the complexity of matchmaker module claiming for optimization and rationalization of the reasoner structure. Furthermore, the application user interface has to be enhanced and redesigned. Finally, we are investigating a new approach browsing the DBpedia KB directly.

References

[1] Baader, F., Calvanese, D., Mc Guinness, D., Nardi, D., Patel-Schneider, P. *The Description Logic Handbook*. Cambridge University Press, New York, 2002.

[2] Watson, R., Akselsen, S., Monod, E., Pitt, L. "The Open Tourism Consortium: Laying The Foundations for the Future of Tourism." *European Management Journal*, Vol. 22, No. 3, 2004, pp. 315–326.

[3] Colucci, S., Di Noia, T., Pinto, A., Ragone, A., Ruta, M., Tinelli, E. "A Non-Monotonic Approach to Semantic Matchmaking and Request Refinement in E-Marketplaces". *International Journal of Electronic Commerce*, Vol. 12, No. 2, 2007, pp. 127-154.

[4] Li, L., Horrocks, I. "A Software Framework for Matchmaking Based on Semantic Web Technology. International Journal of Electronic Commerce, Vol. 8(4), 2004, pp. 39–60.

[5] Colucci, S., Di Noia, T., Di Sciascio, E., Donini, F.M., Ragone, A. "Knowledge Elicitation for Query Refinement in a Semantic-Enabled E-Marketplace". In *7th International Conference on Electronic Commerce ICEC 05*, ACM Press, 2005, pp. 685-691.

[6] Bechhofer, S., Möller, R., Crowther, P. "The DIG Description Logic Interface." In: *Proceedings of the 16th International Workshop on Description Logics (DL'03).* Vol. 81 of *CEUR Workshop Proceedings,* 2003.

[7] Ruta, M., Di Noia, T., Di Sciascio, E., Donini, F.M.. "Semantic based collaborative p2p in ubiquitous computing." *Web Intelligence and Agent Systems*, Vol. 5, No. 4, 2007, pp. 375–391.

[8] Ruta, M., Di Noia, T., Di Sciascio, E., Scioscia, F. "Abduction and Contraction for Semantic-based Mobile Dating in P2P Environments." In: *Proc. of 6th IEEE/WIC/ACM International Conference on Web Intelligence WI08*, IEEE, 2008, pp. 626–632.