Processing natural language queries to disambiguate named entities and extract users' goals: application to e-Tourism

Sanjay Kamath, Lorraine Goeuriot, Marie-Christine Fauvet

LIG (MRIM), University of Grenoble Alpes, 38000, Grenoble, France

RÉSUMÉ. *Cet article présente une étude qui s'inscrit dans le cadre d'un projet plus large qui porte sur la conception et la réalisation d'un système visant à fournir à des utilisateurs mobiles des services personnalisés, dépendant de leur contexte, et adaptés à leurs besoins. Par exemple, un utilisateur peut vouloir des informations sur la météo du lendemain, ou bien réserver des billets d'entrée à un musée voisin, ou encore réserver une table dans un restaurant italien et obtenir les indications pour s'y rendre en voiture. Dans cet article, nous étudions plus particulièrement les problèmes liés à l'extraction de requête fournie par l'utilisateur des paramètres nécessaires à son traitement, ainsi que du contexte de l'utilisateur. Le système que nous proposons est basé sur le traitement de requêtes exprimées en langage naturel. Afin d'extraire de la requête les entités nommées ce système s'appuie sur des bases de connaissances et des outils de désambigüisation.*

ABSTRACT. *This paper presents a study which is part of a broader project. This latter aims at providing mobile users with context-aware personalised services. The E-tourism Project deals with a variety of queries submitted by a tourist, such as booking a hotel room, getting the weather conditions for the next day, or booking tickets in a museum in the neighbourhood and worth to visit. This paper focuses on the query management and processing. The module described analyses and structures the query by splitting it, identifying the named entities, solving ambiguities... To process the query, the system uses various external knowledge bases and Natural Language Processing tools to understand the named entities and proper context of the query using disambiguation techniques.*

MOTS-CLÉS : *e-Tourism, Traitement de Requêtes, Services Personnalisés Dépendants du Contexte*

KEYWORDS: *e-Tourism, Query Processing, Context Aware Personalised Services*

1. Introduction and Motivating Example

With the development of pervasive computing and mobile technologies, mobile applications are getting more and more attention. Distributed computing systems based on context awareness have been proposed in several domains such as health-care, logistics and tourism. Most of the existing service providers in the tourism domain focus only on a few specific goals, such as booking a restaurant, or searching a hotel. These applications, running on mobile devices, are challenged to locate and deliver the right service to the right person, with the appropriate rendering.

The work reported in this paper is part of a broader project which aims at designing and implementing a framework which provides context-aware personalized services for mobile users according to their needs, profile and context. The issues to be addressed in this project are related to system design, software architecture, distributed and heterogeneous resource access and integration, information retrieval and recommender systems. This paper focuses on the user's query management and processing, mainly to highlight the importance of NLP techniques used. Our system considers the user's input in the form of a text query which contains the details and requirements of the user. To illustrate the challenges tackled in this paper, let us consider the following scenario. Alice is an American tourist visiting Paris in France. She picks up her smartphone and issues the query, once connected to our e-Tourism system : *I Want to book a table tonight at the closest restaurant to the Eiffel Tower and know the directions to get there*. The study reported in this paper has the following contributions :

– Recognition and disambiguation of **named entities** using external knowledge sources such as WolframAlpha etc., according to Alice's context the value returned by the GPS embedded in her smartphone says she is located in Paris, Eiffel Tower is then recognized as a named entity in Paris (not in Las Vegas, Nevada, neither in Brisbane, Australia).

– Extraction of user's **goals** : The goals are inferred from the **Named Entities** extracted from the query, In the Example : Alice has two main goals, the first one is *"I want to book a table at the closest restaurant to Eiffel Tower for tonight"*, and the second one *"get directions to reach the restaurant"*.

This paper is organized as follows : we describe the architecture of the whole system in Section 2 ; while we present in Section 3 some related work. Section 4 details our proposed approach. Section 5 details the implementation of this approach and future evaluation of our system. Eventually, in Section 6 we conclude and sketch some further work.

2. Architecture of the eTourism System

The study described in this paper is conducted as part of a broader project whose main goal is to provide mobile users with services according to their needs (Na-Lumpoon *et al.*, 2013).

We call a goal, the task requested by the user (e.g. movie booking, events nearby...). A query might contain one or more goals. The *Context* includes spatial, temporal, physical, and environmental properties such as date, time, location etc. that could be collected by sensors embedded on the devices used to submit the queries. The *Profile* captures users' personal details, preferences and centers of interest.

Figure 1 sketches the overall architecture of this project dedicated to e-tourism (Fauvet *et al.*, 2015). The role of each module and the flow of information are rapidly introduced below.

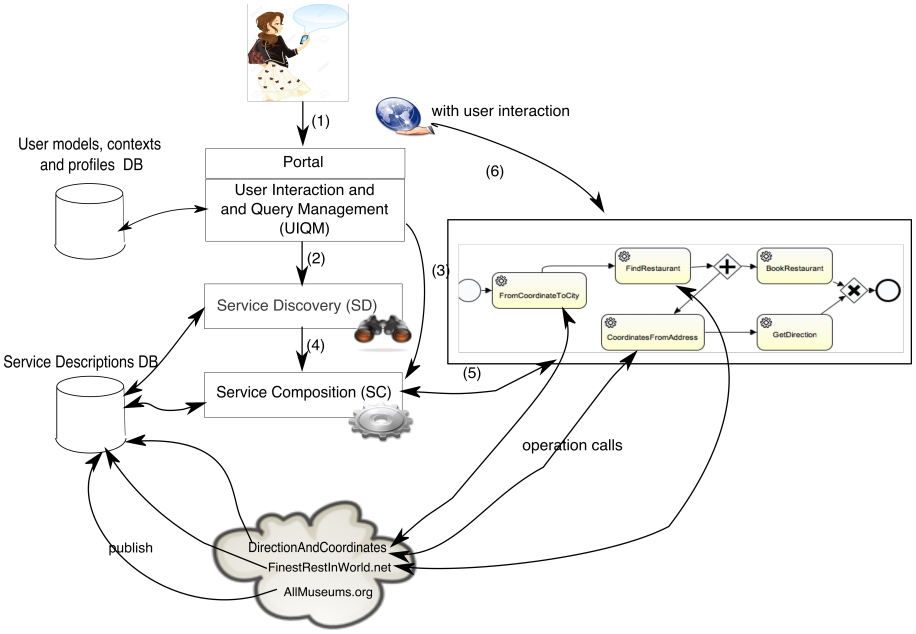

Figure 1. *A context-aware service discovery and composition system for mobile users.*

1) **User interaction and query management module** : aims at managing user interactions by handling the queries submitted by users on their mobile device. A user's query and her identification are received in this module in the data flow (1). This module extracts from the query, the information necessary for the choice and composition of the service components. With data flow (2) it sends single goal queries to the Discovery module and with data flow (3) the user's goals are sent to the Composition and Execution system. This paper focuses only on this module.

2) **Discovery module** : given the user's query received in the data flow (2), her profile and context, this module is responsible for retrieving services among a repository of service descriptions, that once composed can potentially meet the user's goals

expressed in her query. The retrieved services are then sent, in the data flow (4), to the next module. For details see (Caicedo-Castro, 2015 ; Caicedo-Castro *et al.*, 2015).

3) **Composition and execution system** : eventually this module is in charge of automatically composing and executing services returned by the discovery phase. The automated composition is meant to satisfy users' goals. This module results in a BPMN model whose tasks refers to a service operation by the data flow (5) (for details see (Na-Lumpoon, 2015)). During the execution of the resulting process, some interactions with the user might be necessary (see data flow (6)).

3. Related Work

(Tomai *et al.*, 2005) discuss a case study on trip planning using two ontologies to represent the user profile and an ontology for tourism data such as shopping malls, cinemas, activity spots nearby etc. Assumption of the user's location is always the city center because of which the user's device is not constantly looking for GPS signals and User profile related information is collected through a questionnaire to the user. The context matching algorithm uses this information to determine the feasibility of the plan in accordance to the time the traveler has.

Ontology population and the evaluation for tourism domain corpus is discussed in (Ruiz-Martınez *et al.*, 2011) who propose a method for extracting semantic content from textual web documents(such as data about hotels and restaurants) which were obtained from an official tourism web page, to automatically instantiate a tourism domain ontology. Ontology Population deals with the extraction and classification of instances of the concepts and relations that have to been defined in the ontology. Semantic contents are obtained after the NL processing of the web documents using GATE [1] framework and are considered as instance candidates which undergo disambiguation for semantic ambiguities with the use of NLP tools. These contents are then related with their ontology entities by ontology population process.

Similar to DBpedia[2], some efforts were made in building TourPedia (Cresci *et al.*, 2014), which includes point of interests, accommodations, restaurants and attractions using various commercial data sources. However, TourPedia covers only a part of Europe.

Our system aims at extracting named entities from the query, disambiguate the similar results, link the goal(s) of the query with Point of Interest(s) and substitute the required parameters for the goals. To do so, we use widely studied and utilized approaches in Natural Language processing such as Named Entity recognition, Part of Speech tagging and Disambiguation methods.

1. `https://gate.ac.uk/`
2. `http://wiki.dbpedia.org`

4. Proposed Approach

As explained in Section 2, we focus here on the Query Management Module of the eTourism system which is depicted in Figure 2.

Firstly the queries are pre-processed using NLP tools to obtain an intermediate representation of the free text query, which facilitates the Named Entity recognition tool to determine the Named Entities from the query. After the Named Entities are obtained, the Disambiguation of them is carried out using external knowledge sources. The parameters required for the Service composition and Service execution modules are inferred from the User profile, Named entities and Goals from the query.

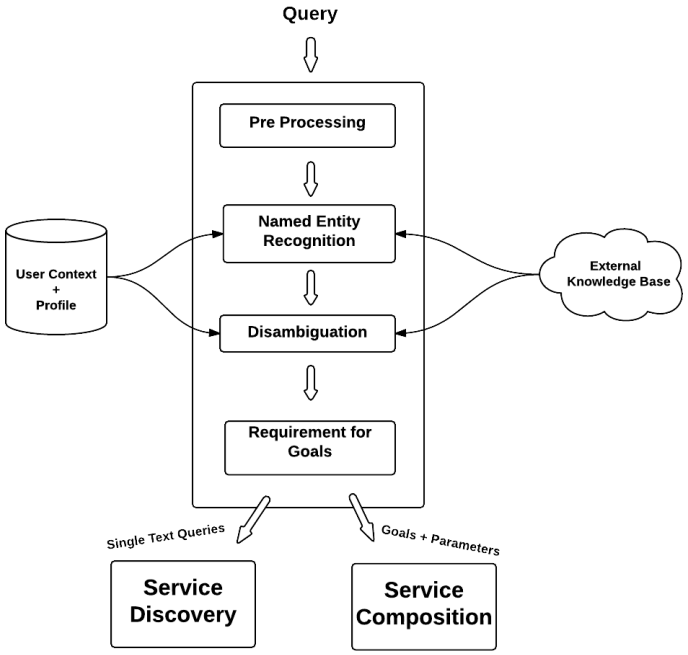

Figure 2. *Architecture of Query Management Module*

4.1. *Pre-Processing of the Queries*

The queries are short text composed of keywords. They can contain operators such as boolean operators (and, or, not) and conditional operators (if, then, else, otherwise, if not). Queries can contain one or multiple goals, separated by punctuation or operators.

Our approach performs the shallow parsing on the queries for keywords mentioned above, and splits the queries into separate single queries. We consider single queries

as queries containing an unique need (single goal). Single queries are separated using Boolean operator keywords as mentioned above. We perform Part of Speech tagging on the queries to extract the *Verbs* and our assumption is that the *Verbs* represent the Goals.

4.2. *Named Entity Recognition*

Named Entity Recognition (NER) is a subtask of information extraction which is performed on the input queries to determine the entities which represent locations, places, POIs etc. NER is a wider domain of Natural Language Processing, we discuss the use of NER for our system which uses knowledge sources as input in Section 5.

Example scenario : **Q1** : *How far is the Colosseum from here ?*
The NER tool takes the tokenized form of the query **Q1** as input and detects the Named Entity as **Colosseum**. If there are multiple Entities with the same name, there is ambiguity in the results. Hence we disambiguate such results, as discussed in the next subsection.

In the cases where the **Goals** of the query which are supposedly Verbs, are ambiguous with the Named Entities, we pose a question to the user and get a response to obtain the right choice. One of the example for such scenario is the use of the movie name **The Grand Budapest Hotel** which can either be a goal for Hotel booking query.

4.3. *Disambiguation*

When two Named Entities such as name of hotels (Park Hyatt, Paris and Park Hyatt, Nice) are the outputs from a query of the user, we decide to consider the closest one to the user (in terms of location) as the correct one. In the case of an ambiguity, we need methods to rank the potential entities. We use an algorithm for the ranking function `R` to do so using the user and the entity locations. The main steps of this algorithm are :

```
Inputs: SRC = User's location obtained using GPS
       S = list containing the named entities from external sources
       Threshold = the maximum distance between the user and POIs
                  (Threshold value must be agreed by the user.)

R(NE, SRC, Threshold):
        for(i=0; i< S.length; i++)
        {if (distance (SRC, location(S)) > Threshold)  then (remove item(i));}
        S.sort(ascending);
Output:
       Top result from List S.
```

We need the top most result after sorting the list in ascending order of the location.

Example scenario : **Q1** : *How far is the Eiffel Tower from here ?*
If the User is in France when he inputs this query, then the **Eiffel Tower** is the one in Paris, France. Whereas if the User is in Las Vegas, Nevada then the **Eiffel Tower** is the one in Las Vegas, Nevada. The context of the location changes the meaning of certain Named Entities. In the example, the result of the algorithm will be Eiffel Tower, Paris because of the user's location which is in France.

4.4. *Requirements for goals*

Each of the user's goal has specific parameters required for composition and execution of services, as shown in Table 1. For instance, a movie booking has as parameters the movie name, cinema name, number of people, the city where the cinema is located and day and time of the booking.

Goals	Parameters
Movie booking	movie name, cinema name, city name, people number, day and time of booking
Hotel booking	hotel name, city name, people number, checking day, checkout day
Museum booking	museum name, city name, people number, visit day
Direction	mode, source, destination

Tableau 1. *Expected parameters depending on goals.*

In cases where the system has conflicts in filling in the parameters, the interaction module asks the user to input that particular parameter again rather than making wrong assumptions. We encountered the *synonymy problem* in this phase. Words like 'Book' and 'Reserve' both refer to the same verb. To tackle this problem, we use *Wordnet*. The results from the query processing module are the parameters required for the composition and execution of services.

An example query and its intermediate representation :

If it rains, then order a pizza to my room at 8pm.
Q1 : *If it rains* **Q2 :** *order a pizza to my room at 8pm.*
Rain = True ; Goal=PizzaOrder
User Profile : Favorite_Pizza : Orientale ;
Hotel_Address : Room No.23, Park Hotel,Grenoble.

Parameters :
Goal : Weather_Information,Pizza_Order
SrcLoc : 45.1877997,5.7646932

Favorite_Pizza : Orientale ;
Hotel_Address : Room No.23, Park Hotel,Grenoble.
Time : 20 :00

5. Implementation and Evaluation

The Figure 2 sketches the overall architecture of the *User Interactions and Query Management module*. The role of each sub module is discussed in details below. The query is obtained from the user through a *Graphical User Interface*[3]. Our system is still in implementation phase.

5.1. *Dataset*

We built a set of 45 simple queries in Natural Language representing various use case scenarios such as Hotel Booking, Restaurant Reservation, Movie Ticket Booking, Driving Directions etc. We plan to build larger corpus of queries for performing evaluations. The queries are divided into different categories such as Single goal queries, Multiple goal queries(separated by keywords such as and,or,along with,also etc.), Conditional queries(separated by keywords such as if,not,if not,without etc.).

5.2. *System Setup*

Apache OpenNLP[4] is a Java machine learning toolkit for Natural Language Processing (NLP). It supports the most common NLP tasks, including rule based and statistical Named Entity Recognition. For processing the queries we use the provided tools such as Tokenizer and POS Tagger[5]. Tokenizer splits the queries into tokens and the POS Tagger appends the tokens with the part of the speech corresponding to that word in the query. This part of the speech value can be used to determine the goals, as goals are always verbs in our approach.

5.3. *Named Entity Recognition and Disambiguation*

Apache OpenNLP toolkit also provides a tool for Named Entity Recognition. The external knowledge sources we use are WolframAlpha[6], Google places[7], Foursquare[8].

3. http://lig-membres.imag.fr/etourism/en/
4. https://opennlp.apache.org/documentation/1.5.3/manual/opennlp.html
5. http://opennlp.apache.org
6. http://www.wolframalpha.com
7. http://developers.google.com/places
8. https://foursquare.com/

For queries containing the use of keywords representing POIs in it, such as *Eiffel Tower*, we use the Foursquare data to obtain results of places similar to the Named entity. For other keywords such as "Movie", "Monument" etc. we use Wolfram Alpha to extract the Named entities. We use Google Places API to obtain the coordinates for these locations which we use in the Disambiguation process. The System parses the output from the sources and ranks them using the ranking function as described in the Disambiguation Algorithm.

5.4. *Requirements for goals*

Goal(s) from the query are extracted using the tools discussed above. We use Wordnet data to compare the results with goals. We classify the goals mentioned above using a key for representing each goal. This key is stored along with the set of goals. For example, "Book" is a key for all booking-based queries. Whenever a *Verb* is encountered in the query, Wordnet is accessed to get the verbs which are synonyms to the input verb. Data obtained from Wordnet are matched with keys to determine the key for the set of goals which it matched. We substitute the data present in the query for the parameters we assumed in Table 1. Some of the parameters present in the query match the parameters assumed, and some parameters might be missing.

The system labels it as missing parameter and passes it to the service composition module. This module does not invoke an interaction with the user for the missing parameters, because the service operations might need few parameters and some of the parameters obtained from the user might not be used. Hence invoking the user for appropriate missing parameter is solely dependent on service execution module.

5.5. *Results*

The results generated from this module are : 1) Single Text queries or Single Goal queries which are the inputs for the Service Discovery module. 2) Goals + Parameters which are the inputs for the Service Composition module.

5.6. *Evaluation*

The Implementation of eTourism system is in progress yet along with the corpus of queries. We plan to evaluate this module in two ways :

– Comparing the expected goals from a query with the predefined goals for each query in the corpus.

– Determining the number of missing goals from the query which contains complete set of information.

6. Conclusion and Future Work

We present an approach to disambiguate Named entities related to Tourism based on location information. Other context information can also be used to process the queries. One of the main concerns about the above approach is privacy. Users might not disclose their location. The temporal information which are hidden in named entities can be used to determine the time related information and compute the intervals between the events to make the system more dynamic to hidden timed entities and hidden time intervals in them.

Acknowledgement

This work has been carried out in the context of the Guimuteic project funded by Fonds Européen de Développement Régional (FEDER) of région Auvergne Rhône-Alpes.

7. Bibliographie

Caicedo-Castro I.-B., « Sniffer : A text description-based service search system », , PhD Dissertation, University of Grenoble, 2015.

Caicedo-Castro I.-B., Fauvet M.-C., Lbath A., Duarte-Amaya H., « Toward the highest effectiveness in text description-based service retrieval », *Document Numérique*, vol. 2-3, p. 155-177, 2015.

Cresci S., D'Errico A., Gazzé D., Lo Duca A., Marchetti A., Tesconi M., « Towards a DBpedia of Tourism : the case of Tourpedia », *Proceedings of the 2014 International Conference on Semantic Web-Poster and Demo Track, ISWC2014*, 2014.

Fauvet M.-C., Kamath S., Caicedo-Castro I., Lbath A., Goeuriot L., « Offering Context-Aware Personalised Services for Mobile Users », *Proc. of the Inter. Conf. on Service Computing Conf. (ICSOC) demo session*, Goa, India, 2015.

Na-Lumpoon P., « Toward a Framework for Automated Service Composition and Execution : E_Tourism Applications », , PhD Dissertation, University of Grenoble, 2015.

Na-Lumpoon P., Lei M., Caicedo-Castro I., Fauvet M., Lbath A., « Context-Aware Service Discovering System for Nomad Users », *Proc. of the 7th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, 2013.

Ruiz-Martınez J., Minarro-Giménez J., Castellanos-Nieves D., Garcıa-Sánchez F., Valencia-Garcia R., « Ontology population : an application for the E-tourism domain », *International Journal of Innovative Computing, Information and Control (IJICIC)*, vol. 7, n° 11, p. 6115-6134, 2011.

Tomai E., Spanaki M., Prastacos P., Kavouras M., « Ontology assisted decision making–a case study in trip planning for tourism », *On the Move to Meaningful Internet Systems 2005 : OTM 2005 Workshops*, Springer, p. 1137-1146, 2005.