A Semantic Based Question Answering System for Thailand Tourism Information

Alisa Kongthon, Sarawoot Kongyoung,
Choochart Haruechaiyasak and **Pornpimon Palingoon**
Human Language Technology Laboratory (HLT)
National Electronics and Computer Technology Center (NECTEC)
Thailand Science Park, Klong Luang, Pathumthani 12120, Thailand
alisa.kon@nectec.or.th,sarawoot.kon@nectec.or.th
choochart.har@nectec.or.th,pornpimon.pal@nectec.or.th

Abstract

This paper reports our ongoing research work to create a semantic based question answering system for Thailand tourism information. Our proposed system focuses on mapping expressions in Thai natural language into ontology query language (SPARQL).

Topic: Language processing, reasoning aspects

1 Introduction

The Semantic Web can provide significant impact on an information intensive industry such as tourism where information plays an important role for decision and action making. Tourism is one of the economic factors in Thailand. From the statistics provided by the Office of Tourism Development[1], the number of tourists visiting Thailand in 2010 is approximately 16 millions. Providing an automatic question-answering system on tourism information would be very useful for tourists to plan their trips.

In this paper, we propose a semantic based question answering system for Thailand tourism information. Our proposed system focuses on converting expressions in Thai natural language into SPARQL[2], an ontology query language.

Currently there already exist publicly available formal tourism ontologies. Notable ones include Harmonise Ontology (Fodor and Werthner, 2005), Mondeca Tourism Ontology[3], OnTour Ontology[4] and TAGA Travel Ontology[5]. These ontologies are designed to integrate and manage heterogeneous tourism data (Prantner et al., 2007). In our proposed system, we apply publicly available OnTour Ontology to represent the tourism concepts and relations such as place, accommodation, restaurant and attraction.

There are also number of studies that provide natural language interfaces to ontologies. Notable works include ORAKEL (Cimiano et al., 2007), NLP-Reduce (Kaufmann et al., 2007), PANTO (Wang et al., 2007), AquaLog (Lopez et al., 2007), QuestIO (Damljanovic et al., 2008) and FREyA (Damljanovic et al., 2010). However, these approaches only focus on English language query. Since Thai language characteristics are different than English, we propose an approach to map expressions in Thai natural language to ontology based on pattern analysis.

2 The Proposed System

The proposed system is illustrated in Figure 1. The information is collected, by using a crawler, from various websites related to tourism in Thailand. We collect two different types of information. The first type is general information such as places to visit, accommodations, attractions, and restaurants which are used to design our tourism (named *Tour*) ontology. The second type is requests or questions for tourism information posted on public discussion forums. These natural language requests are used to construct a tourism related lexicon and an annotated corpus for request pattern analysis.

[1] The office of Tourism Development, *http://www.tourism.go.th*

[2] SPARQL Query Language for RDF, *http://www.w3.org/TR/rdf-sparql-query/*

[3] Mondeca Tourism Ontology, *http://www.mondeca.com*

[4] DERI, OnTour Ontology, *http://e-tourism.deri.at/ont/index.html*

[5] TAGA Ontology, *http://taga.sourceforge.net/owl/index.html*

Proceedings of the KRAQ11 Workshop, pages 38–42,
Chiang Mai, Thailand, November 12, 2011.

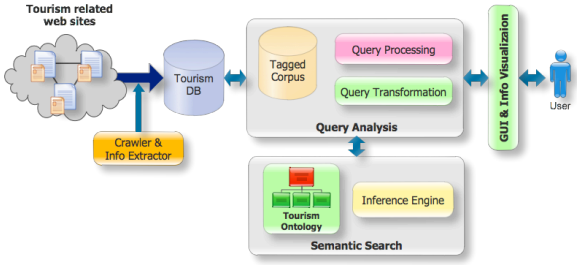

Figure 1. The proposed system

The performance of our proposed system depends on the design and completeness of the lexicon. We distinguish lexicon into two types: domain-dependent and domain-independent. For the domain of tourism, the domain-dependent lexicons could be, for instance, "place," "accommodation," "restaurant," "attraction," and "price". The domain-independent lexicon consists of regular words, which provide different functions in the sentence. For our proposed system, we design two different domain-independent lexicons as follows:

- **Particles (Par)**: In Thai language refer to the sentence endings which are normally used to add politeness of the speakers (Cooke, 1992). Examples are ครับ (Krub) and ค่ะ (Ka).

- **Request (Req)**: Phrases that are used for request information. Example is ช่วยแนะนำ (could you suggest).

The collected natural language requests or questions are first manually annotated according to the domain-dependent and domain-independent tag sets. From this tagged corpus, we can construct request patterns by collecting text segments, which contain both domain-dependent and domain-independent words.

Given a user query of natural language request, the query analysis module starts by performing the word segmentation. Some of the tokenized words with their variations or similar meanings will be normalized into a standard term. The next step is to construct patterns from these processed natural language requests. These patterns will be converted into SPARQL, which is then used to query our *Tour* ontology located in the semantic search module. Figure 2 shows an example of our *Tour* ontology and a partial instantiation of an accommodation. The ontology consists of seven different classes: *Accommodation, Restaurant, Attraction, Type, Cost, Address,* and *GPSCoordinates*. Each class contains different properties. For instance, the Accommodation class contains "hasName," "hasType," "hasMaxPeople," "hasCost," and "hasGPSCoordinates" properties. Inference engine is also used to derive new knowledge. For instance, when a user requests for an accommodation, some nearby restaurants can also be recommended to him/her.

3 An Illustrative Case: Accommodation Information Request

To evaluate the proposed system, we perform an experiment with an illustrative case on accommodation information request.

3.1 Corpus Preparation

We collected 300 natural language requests on accommodation from *Pantip.com*, one of the famous Thai language discussion forums. Table 1 shows the lexicon related to accommodation request.

Type	Examples
Accommodation Clue <Acc_Clue>	ที่พัก (accommodation), ที่นอน (sleeping place)
Accommodation Type <Acc_Type>	โรงแรม (hotel), รีสอร์ท (resort), โฮมสเตย์ (homestay)
Accommodation Condition <Acc_Cond>	หมาเข้าได้ (dogs allowed), มีสระว่ายน้ำ (have swimming pool)
Place <Place>	กรุงเทพ (Bangkok), เชียงใหม่ (Chiangmai), หัวหิน (Hua Hin)
Location Clue <Loc_Clue>	ที่ (at), บน (on), ใกล้ (near), แถวๆ (not far from)
Price Clue <Price_Clue>	ราคา (price), ค่าที่พัก (accommodation price)
Price Condition <Price_Cond>	ถูก (cheap), ไม่แพง (not expensive), กำลังดี (moderate)
People Clue <Ppl_Clue>	ไปกัน (going with), มากัน (coming with), มีทั้งหมด (total number of)
Unit <Unit>	คน (person), บาท (Baht)

Table 1. Lexicon related to accommodation

3.2 Experiments

To verify if our lexicon is sufficient in identifying accommodation request, we first perform request identification task. This task aims to distinguish between requests for accommodation

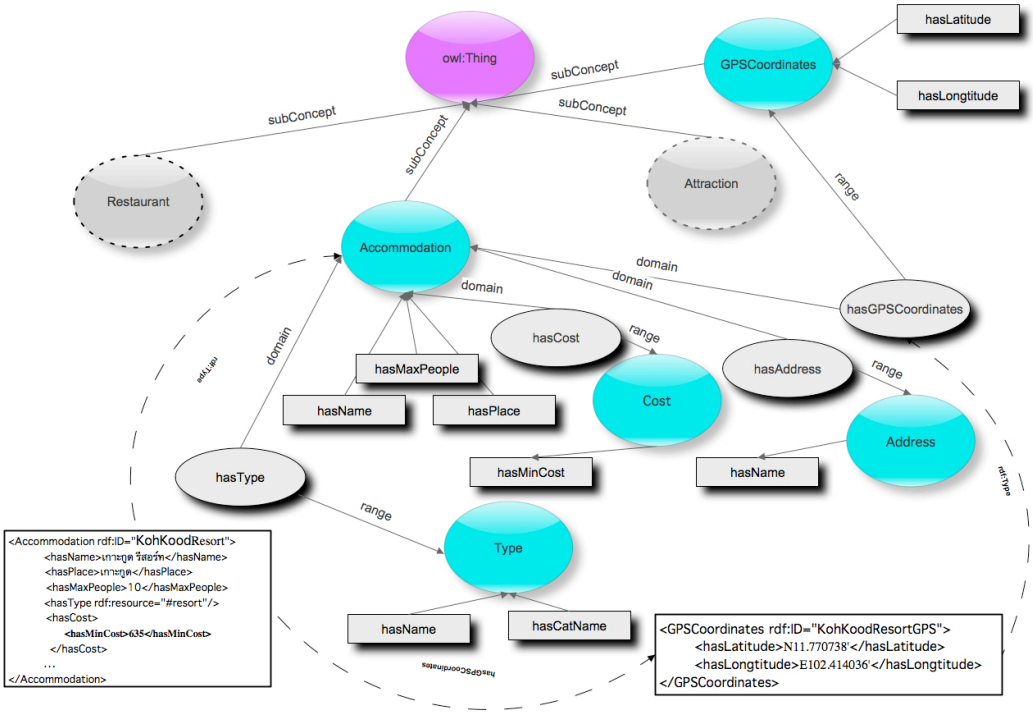

Figure 2. An example of *Tour* ontology

and any other types of requests or questions. We initially performed evaluation on 600 questions where 300 are accommodation requests and other 300 are not. The identification result yields 89% recall and 95% precision.

We then use the tagged corpus and the extracted lexicon to construct the most frequently occurred patterns. For our 300 accommodation requests, the total number of extracted patterns is 168. Table 2 shows some examples of the top-5 accommodation request patterns. These patterns are then converted into SPARQL queries. The following example (Example 1) illustrates the conversion from natural language requests to SPARQL queries. The given request asks for recommended accommodations with a constraint on the number of people staying.

Example 1:

Natural language request input:
รบกวนแนะนำที่พักที่เกาะกูดหน่อยครับ ไปกัน 10 คน
Could you suggest accommodation at Koh Kood “noi krub?”, going with 10 people

Word segmentation process:
รบกวนแนะนำ|ที่พัก|ที่|เกาะกูด|หน่อย|ครับ| |ไปกัน| |10| |คน|

Pattern construction process:
รบกวนแนะนำ<Req>**|ที่พัก**<Acc_Clue>**|**
|ที่<Loc_Clue>**|เกาะกูด**<Place>**| |หน่อย**<Par>
|ครับ<Par>**| |ไปกัน**<Ppl_Clue>**| |**10<Number>**|**
|คน<Unit>**|**

Extracted pattern:
<Req> <Acc_Clue> <Loc_Clue> <Place>
<Ppl_Clue> <Number><Unit>

SPARQL conversion:

```
SELECT ?a WHERE
  {
  tour:Tourist   tour:hasAccomodation ?a.
  {{?a      tour:hasPlace "เกาะกูด".}
  UNION { ?a    tour:hasAddress ?a.
    ?b   tour:hasName "เกาะกูด". }}
  ?a      tour:hasMaxPeople "10"
  }
```

No.	Top-5 accommodation request patterns
1	**<Req> <Acc_Clue> <Loc_Clue> <Place>** <ช่วยเลือก> <ที่พัก> <ที่> <พัทยา> <Please help select> <accommodation> <in> <Pattaya>
2	**<Req> <Acc_Clue> <Place>** <แนะนำ> <ที่พัก> <หาดป่าตอง> <Suggest> <accommodation> <Patong Beach>
3	**<Req> <Acc_Type> <Loc_Clue> <Place>** <รบกวนแนะนำ> <โฮมสเตย์> <ใกล้> <ตลาดน้ำอัมพวา> <Please suggest> <homestay> <near> <Amphawa floating market>
4	**<Req> <Acc_Clue> <Loc_Clue> <Place> <Acc_Cond>** <ช่วยแนะนำ> <ที่นอน> <แถวๆ> <หัวหิน> <มีสระว่ายน้ำใหญ่ๆ> <Please suggest> <sleeping place><near> <Hua Hin> <having a large swimming pool>
5	**<Req> <Acc_Clue> <Place> <Acc_Cond>** <แนะนำ> <ที่พัก> <เขาใหญ่> <เอาน้องหมาไปได้> <Suggest> <accommodation> <Khao Yai> <dogs allowed>

Table 2. Top-5 accommodation request patterns with examples

The derived SPARQL will be used to query our *Tour* ontology. For instance, our system will select "Koh Kood Resort" (as shown in the instantiation in Figure 2) as one of the recommended accommodations for the request in Example 1.

3.3 Discussion

Our experiment shows that most words from the lexicon related to accommodations can be derived and mapped into relevant structure in our *Tour* ontology. However, some content, especially those belong to accommodation condition (i.e., <Acc_Cond>), are difficult to extract since the ways to explain conditions can be much varied and very descriptive. Table 3 shows some examples of challenging cases for accommodation condition.

No.	Some difficult cases for "accommodation condition"
1	นั่งรถประจำทางไปเที่ยวสะดวก (convenient to take the bus)
2	ใกล้กับงานแห่เทียน (closer to the Candle Festival)
3	เหมาะสำหรับจัดกิจกรรมรับน้อง (suitable for holding a college orientation activity)
4	เน้นกินเหล้าสังสรรค์ (focus on drinking and partying)
5	มีรถมอเตอร์ไซด์เช่าแถวสนามบิน (have motorcycle rental service near the airport)

Table 3. Examples of difficult cases for "accommodation condition"

Some of these conditions would require an inference engine to help identify the answer. For example, in Case No.2, the distance between each retrieved accommodation and attraction (i.e., the Candle Festival in this case) will be calculated based on their GPS coordinates. Only the accommodations located near the Candle Festival will be recommended to the user. In addition, some rule base can be applied in order to transform descriptive language into more structured format.

4 Conclusion and future work

In this paper, we proposed a framework for a semantic question-answering system for Thailand tourism information. Our proposed system focuses on mapping expressions in Thai natural language into ontology query language (SPARQL). The proposed method first constructs a set of patterns from a tagged corpus containing both domain-dependent and domain-independent lexicons. The derived patterns are then converted into relevant SPARQL queries. We performed an experiment on a case study regarding accommodation information requests. Our experiment results showed that some features such as place, accommodation type, price, and number of people staying can be extracted from the natural language query and easily converted into corresponding SPARQL queries. However, in some cases such as the description of accommodation condition are more difficult to extract. For future work, we plan to apply some rule base and inference engine to help derive answers to the user.

References

Cimiano P., Haase P. and Heizmann P. 2007. Porting natural language interfaces between domains: an experimental user study with the orakel system. *Proc. of the 12th International Conference on Intelligent User Interface*. 180-189.

Cooke, J.R. 1992. Thai Sentence Particles: Putting the Puzzle Together. *Proc. of the Third International Symposium on Language and Linguistics*, 1105-1119.

Damljanovic D., Tablan V. and Bontcheva K. 2008. A text-based query interface to owl ontologies. *In 6th Language Resources and Evaluation Conference (LREC)*.

Damljanovic D., Agatonovic M. and Cunningham H. 2010. Natural language interfaces to ontologies: combining syntactic analysis and ontology-based lookup through the user interaction. *The Semantic Web: Research and Application.* 106-120.

Fodor O. and Werthner H. 2005. Harmonise: A Step Toward and Interoperable E-Tourism Marketplace. *International Journal of Electronic Commerce*, 11-39.

Kaufmann E., Bernstein A. and Fischer L. 2007. NLP-Reduce: A naïve but domain-independent natural language interface for querying ontologies. *Proc. of the European Semantic Web Conference*.

Lopez V., Uren V., Motta E. and Pasin M. 2007. Aqualog: An ontology-driven question answering system for organizational semantic intranets. *Web Semantics: Science, Services and Agents on the World Wide Web*. 5(2), 72-105.

Prantner K., Ding Y., Luger M., Yan Z. and Herzog C. 2007. Tourism Ontology and Semantic Management System: State-of-the-arts Analysis. *IADIS International Conference WWW/Internet 2007*. 111-115.

Wang C., Xiong M., Zhou Q. and Yu Y. 2007. Panto: A portable natural language interface to ontologies. *The Semantic Web: Research and Application*.473-487.