APPLICATION OF SEMANTIC WEB TECHNOLOGY IN BUILDING TOURISM INFORMATION SUPPORT SYSTEM

Pham Thi Thu Thuy¹, Tran Vinh Phuc²

¹Faculty of Information Technology, Nha Trang University
²Faculty of Informatics and Foreign Languages, Lam Dong College of Economics and Technology

thuthuy@ntu.edu.vn, tvp2510@gmail.com

ABSTRACT: Currently, there are many different systems helping visitors look up tourist information. However, to visit famous tourist attractions, visitors need to know all the information such as accommodation, nearby restaurants, and other attractions. Most tourists search for information using search engines, but most of the results do not meet their requirements. This article proposes a system for recommending tourist attractions in Da Lat City by using the Semantic Web, the Semantic Web Rule Language, and the KNN algorithm (K-Nearest Neighbor) for finding and displaying most relevant tourist information. This system is built in Python language and built as a web application. The system allows users to choose recommendations according to their personal preferences (accommodation, places to eat, type of attractions, etc.), thereby suggesting data to find appropriate and suitable destinations according to their needs.

Keywords: Ontology, Semantic Web, SWRL, KNN.

I. INTRODUCTION

Tourism is one of the most successful, dynamic industries in the world and it is constantly evolving because of technological advancement. Information technology is being used to enhance tourism services such as travel booking, itinerary planning, marketing, and information sharing. These services often can be deployed with the help from dynamic Web applications.

Existing travel apps rely on information sources such as websites to create travel products and services. There is a lack of important features can be found in these apps. For example, booking websites allow travelers to book, but often lack the functionality to suggest an itinerary based on traveler's interests. A Semantic Web application using ontology for general record-keeping and semi-structured query tools could help overcome the limitations of current web engineering and help build smart travel applications.

The current tourism support applications in Da Lat City also use dynamic web techniques and they have the same disadvantages as above. For example, the search for information for tourism in Da Lat today is the results we get directly from the web, which can provide information to answer questions that people ask. However, it requires analysis of the meaning of the data and its relation to the requirements. In addition, they cannot be solved automatically by computer and, more especially, complex search queries which often produce undesirable results [1, 2].

Due to the need for exact search results and the main purpose of the Semantic Web which makes searching easier if everything was put in semantics, we need to build an ontology to store information about the tourist information in Da Lat and build an information search engine on this ontology afterwards. With Semantic Web, more accurate results will be achieved and computers will also able to compare different information to give the most accurate results.

The purpose of the paper is to build a system to support the accurate search of tourist information from related data stored in ontology form. Specifically, there is a use of KNN algorithm to search and suggest places of tourist information for tourists when visiting in Da Lat city.

The article consists of the following parts: (I) Introduction to the article and the purpose of the research as well as the obtained results. (II) Related work. (III) Method of implementation: Method of construction ontology for tourism in Da Lat, using semantic Web rule set language, KNN algorithm to search for nearby places and suggest destinations for visitors. (IV) Testing and results. (V) Conclusion.

II. RELATED WORK

Semantic search based on ontology is a popular semantic search method and there have been some research in this field such as building semantic search engine architecture based on ontology [12], building ontology-based architecture for semantic search engine for Arabic language [13], building a semantic search engine based on ontology for the Koran [14] and semantic search based on ontology [15]. In general, all studies mentioned above have in common that they use ontology to store data and use SPARQL query language to retrieve information. Each study proposes to look up information on a specific ontology. Depending on the structure format of the ontology, different query statements were proposed.
Semantic Web applications in management and data access to tourism has been mentioned in several studies [3, 8, 9, 11]. Data on tourism based on ontology is applied in many countries such as Australia [11], Indonesia [8], or Thailand [9].

After having tourism data built on ontology, data search and subsequent recommendations are demanded by the application. A number of related studies on building an ontology-based data management and search system have been found. In the study of the group of authors Ranestari Sastriani [8], the authors proposed to use the Vector Space Model to compare the search results on the ontology to return the most correct results for the user. The most typical is the research of the author group Snae Namahoot [9]. Snae Namahoot and colleagues proposed to use the Naïve Bayes Model to classify the data of Thai tourism websites, and then suggest relevant search recommendations to users.

In general, above studies all emphasize the importance of ontology in data storage and related semantic search. Depending on the level of application, researchers suggest using SPARQL [16] or combining SPARQL with some similarity calculation tools to filter search results more accurately. The aforementioned applications mainly propose the exact search based on the comparison of semantic similarity of the data on the ontology. In addition to building an ontology to store travel information and using SPARQL to search for data, a Semantic Web inference language to supplement the semantics of the data can be used and the construction can be built up. Location coordinates of tourist destinations, thereby can propose the application of KNN algorithm to calculate nearby tourist places and recommend related tourism services to users.

III. METHODS OF IMPLEMENTATION

This section will describe the method of ontology building and the search program for tourism information in Da Lat. The system is divided into four parts: (1) Building ontologies for tourism in Da Lat, (2) Proposing some sets of rules to add semantics to the ontology, (3) Finding the nearest location using the KNN algorithm, and (4) Recommend tourist places based on query. The details of the sections are presented as follows.

A. Building Ontology for tourism in Da Lat

Through the actual survey and system design analysis, the ontology for the tourism support system in Da Lat was built, including four main classes as follows. Activity: Including activities such as entertainment, events and activities. events taking place in Da Lat city or as tourist attractions; Destination: Contains subclasses of information about geography, tourist routes or areas containing tourist attractions; Population: Contains information related to people such as interests when visiting Da Lat city: sightseeing, taking pictures, watching sunset,...; Service: Contains services for tourism in Da Lat such as dining places, hotel locations, restaurants or shopping places.

Figure 1. Four main classes in the ontology

Details for the main classes are as follows:

Activity:
Entertainment class: This class contains information about entertainment such as Cinema, Circus, and Theater.

Event class: Contains information about event classes such as Exhibition, Expo, and Festival.

Sightseeing class: Contains information about popular types of tourism in Da Lat city such as Culture Tourism, Community Tourism, Eco Tourism, Festival Tourism, Historical Tourism, Island resort Tourism, and Trade Tourism.

Destination:

Direction class: The class contains tourist attractions located on the same road such as: Ho Xuan Huong, Tuyen Lam Lake, etc.

Location class: The area layer contains tourist attractions such as hotels, restaurants, and shopping places in a geographical area so that visitors can easily find information.
Tour class: Contains tourist attractions that are included in the sightseeing tours to discover Da Lat City which are popular and prevalent today.

Population: This is a class used for tourists to choose tourist attractions based on their preferences in addition to the above criteria. Inside it, there is a sub-class named which contains the tourist interest entities:

In addition, in the Population class, there is also a Criteria class, which plays the role of containing criteria when visitors need to filter conditions when searching for places to visit such as hotels, dining,...
The Service class includes the following subclasses:

![Children classes of Services](image)

Accommodation: Contains information about types of hotels available in Da Lat City so that visitors can search by area or by type of service. FoodAndDrink: The class contains information about places to eat and is classified into subclasses so that visitors can choose according to the purpose. Shopping: Contains information about locations so that visitors can search for business locations to buy and sell agricultural products and food according to their consumption needs when traveling in Da Lat city.

All of the above classes have a common attribute that are identifiers and are distinguished according to each type of use such as FoodAndDrink class has a distinguishing code of AU or hotels have a code of KS, etc. These identifiers make querying and extraction simpler in the application.

### B. Adding semantic rules to the ontology

To supplement and enrich the semantics of the ontology, the Semantic Web Rule Language (SWRL) were used to build some rule sets. These rule sets support the process of accessing and linking data conveniently and accurately. Here are some key rules in the program:

The law of finding places that are geographically close to each other to provide recommendations for nearby destinations for tourists to refer to:

- `isNear(?x, ?y) -> isNear(?y, ?x)`
- `isNear(?x, ?y) ^ isNear(?y, ?z) -> isNear(?x, ?z)`
- `isLocation(?x, ?y) ^ isLocation(?x, ?z) -> isNear(?y, ?z)`

The law of grouping places close to each other to form a Tour or have the same closed travel schedule is illustrated:

- `isNear(?x, ?y) ^ isLocation(?z, ?x) -> isDirection(?z, ?y)`
- `isTour(?x, ?y) ^ isNear(?y, ?z) -> isTour(?x, ?z)`

In addition, the program also has a set of rules that aim to create places that have the same function as the type of tourism or the same type of Tour to suggest places that are similar to tourists' travel preferences:

- `isType(?x, ?y) ^ isType(?x, ?z) ^ isHobbit(?h, ?y) -> isHobbit(?h, ?z)`
- `isTour(?x, ?y) ^ isTour(?x, ?z) ^ isHobbit(?h, ?y) -> isHobbit(?h, ?z)`

The set of rules included in the program is represented in the Protégé program as follows:

![The set of rules in the Protégé](image)
C. Finding the nearest location

In this study, the KNN algorithm is used to calculate the similarities of tourist information, in order to support the system to recommend suitable tourist destinations for users. The primitive KNN algorithm is used to group data that are similar or have close information values as well as compares the data sets with the input data and finds the minimum distance between them [7].

In this paper, the KNN algorithm is applied to rank and recommend tourist information to meet the needs of users. The user must select the type of tourist accommodation, cuisine, attractions, tourist activities, etc. Based on those selections, the system will suggest appropriate recommendations. The execution of the algorithm is as follows:

Step 1: Calculate Euclidean distance

The first step is to calculate the distance between two rows in a data set. The rows of data are mainly made up of numbers. Therefore, in order to calculate the distance between two rows of those numbers (or vector), a line must be drawn. To calculate the straight line distance between two vectors, the Euclidean distance measure should be used. The Euclidean distance is calculated as the square root of the sum of the squares of the difference between the two vectors.

\[
\text{Euclidean distance} = \sqrt{\sum_{i=1}^{N} (x_{1i} - x_{2i})^2}
\]

Where x1 is the first row of data, x2 is the second row of data, and i is the index of a particular column. From this, we sum across all columns. With Euclidean distance, the smaller the value, the more similar two records will be. A value of 0 means there is no difference between the two records. The steps to calculate the Euclidean distance are represented in the programming language as follows:

```python
def euclidean_distance(row1, row2):
    distance = 0.0
    for i in range(len(row1)-1):
        distance += (row1[i] - row2[i])**2
    return sqrt(distance)
```

Step 2: Find the nearest neighbor

After calculating the Euclidean distance, the next step is to find the nearest neighbors or neighborhoods. Neighborhoods of a new piece of data in the dataset are k nearest cases and the neighborhood is defined by the Euclidean distance measure. To determine the neighbor position of a new piece of data in a dataset, the distance between each record in the dataset must be calculated with the new data part.

After the distances are calculated, there will be an act of sorting all the records in the training dataset by their distance from the new data. Then the first k to return as the most similar neighbors can be chosen. The distance of each record in the dataset is stored as a tuple and sorted in descending order. The process of finding near neighbors can be expressed in programming language as follows:

```python
def get_neighbors(train, test_row, num_neighbors):
    distances = list()
    for train_row in train:
        dist = euclidean_distance(test_row, train_row)
        distances.append((train_row, dist))
    distances.sort(key=lambda tup: tup[1])
    neighbors = list()
    for i in range(num_neighbors):
        neighbors.append(distances[i][0])
    return neighbors
```

Step 3: Make a prediction

The nearest neighbor values collected from the training data set can be used to make predictions. In the case of classification, we can return the closest representative class among the neighboring classes. The closest representative class can be obtained by executing the max() function on a list of output values from the neighborhoods. From the list of neighboring classes, the max() function will return a set of unique and closest class values. This prediction process can be expressed as a programming language as follows:
```
def predict_classification(train, test_row, num_neighbors):
    neighbors = get_neighbors(train, test_row, num_neighbors)
    output_values = [row[-1] for row in neighbors]
    prediction = max(set(output_values), key=output_values.count)
    return prediction
```

**D. The system of recommending tourist places in Da Lat**

Firstly, users enter the type of travel, type of accommodation, cuisine, etc. Then, the system will follow the following steps:

- **Step 1.** Take user input information and match it with inference rules stored in SWRL format.
- **Step 2.** Calculate the cruise traffic to meet the user's needs by applying the KNN algorithm.
- **Step 3.** Find and rate items such as hotels, restaurants, services, events (or festivals) that tourists should visit during their stay.
- **Step 4.** Suggested places will be suggested close to the tourist's selected places are showed as results.

The system is built as a web application using the Python programming language. Thus, it can extract travel information from the ontology and perform inference rules to give the best recommendation to the user.

**IV. TESTING AND RESULTS**

The tourist destination recommendation system is built in the form of a website, including functions so that visitors can easily find information that suits their needs and preferences.

The main interface of the program is as follows:

![Main interface of the application](image1.png)

**Figure 11. Main interface of the application**

The main functions of the application include information search tools when tourists visit in Da Lat such as hotels, dining, shopping, tourist attractions, etc.

![Interface of the main functions of the application](image2.png)

**Figure 12. Interface of the main functions of the application**

Users can search for information about places to eat according to each selection criteria as shown in Figure 13.
APPLICATION OF SEMANTIC WEB TECHNOLOGY IN BUILDING TOURISM INFORMATION SUPPORT SYSTEM

If users have a demand for search by type of hotel, dining, shopping, tourist attractions, the system will suggest location information close to the user's search results like Figure 14.

If customers demand to search for shopping location, the system will return the results of the nearest shopping locations which go along with entertainment, dining and sightseeing places, etc. as shown in Figure 15.

V. CONCLUSION

In this paper, a database of tourism has been stored and organized in the form of ontology and a tourism recommendation system for visitors is also built. The main contribution of the paper consists of three parts. Firstly, the paper has proposed a model to store tourism data in Da Lat City. Besides, this model can be applied to similar tourism information management systems in Vietnam as well as abroad. Secondly, the paper has proposed some rule sets to add semantics to the data. This is the outstanding advantage of using ontology to store data compared to storing it using a relational database. Based on these rule sets, users can add new rule sets to enrich the data semantics. Thirdly,
paper proposes to use the KNN algorithm to calculate the nearby places, thereby making appropriate recommendations based on the search criteria of users (tourists). In the future research, there might be an integration of the GIS geographic information system into the application to increase the accuracy of the search for location and return real-time search results.

REFERENCES


ỨNG DỤNG CÔNG NGHỆ WEB NGỮ NGHĨA TRONG XÂY DỰNG HỆ THỐNG HỖ TRỢ THÔNG TIN DU LỊCH
Phạm Thị Thu Thùy, Trần Vĩnh Phúc