

# A novel ontology framework supporting model-based tourism recommender

Ho Quoc Dung<sup>1</sup>, Lien Thi Quynh Le<sup>2</sup>, Nguyen Huu Hoang Tho<sup>3</sup>, Tri Quoc Truong<sup>4</sup>,  
Cuong H. Nguyen-Dinh<sup>5</sup>

<sup>1,5</sup>School of Engineering and Technology, Hue University, Huế, Viet Nam

<sup>2</sup>University of Economics, Hue University, Huế, Viet Nam

<sup>3</sup>Tomas Bata University, Zlín, Czech Republic

<sup>4</sup>Faculty of Engineering, Van Lang University, Ho Chi Minh, Viet Nam

## Article Info

### Article history:

Received Dec 18, 2020

Revised Aug 12, 2021

Accepted Aug 20, 2021

### Keywords:

Ontology

Semantic similarity

Semantic vector

Supervised learning models

Tourism recommender

## ABSTRACT

In this paper, we present a tourism recommender framework based on the cooperation of ontological knowledge base and supervised learning models. Specifically, a new tourism ontology, which not only captures domain knowledge but also specifies knowledge entities in numerical vector space, is presented. The recommendation making process enables machine learning models to work directly with the ontological knowledge base from training step to deployment step. This knowledge base can work well with classification models (e.g., k-nearest neighbours, support vector machines, or naive bayes). A prototype of the framework is developed and experimental results confirm the feasibility of the proposed framework.

*This is an open access article under the [CC BY-SA](#) license.*



### Corresponding Author:

Cuong H. Nguyen-Dinh

School of Engineering and Technology

Hue University

1 Dien Bien Phu St., Hue City, Viet Nam

Email: ndhcuong@hueuni.edu.vn

## 1. INTRODUCTION

Recommender systems make use of machine learning models in their decision making process. These model-based recommender systems often use the vector-based recommender datasets (e.g., MovieLens [1], book-crossing [2]) for measuring performances in experiments. While these datasets are limited in several domains (e.g., movies, books), the graph-based open linked data (e.g., DBpedia [3]) provide data in many fields and have been used as a supplementary data source in recent recommender research [4], [5]. However, the graph nature of open linked data makes it difficult to be consumed by machine learning models and a few domains of recommender datasets are not enough to build real-life specific recommenders (e.g., tourism recommenders). In order to fill this gap, our study focuses on constructing vector-based data for ontological knowledge base and generating tourism recommendation items based on the use of these vectors.

In this paper, we introduce a novel ontological framework that supports model-based tourism recommender in generating top-K personalized recommendations. To be more specific, we design a tourism ontology for machine learning so-called tourism ontology for machine learning (TOML) which captures knowledge of tourism domain and also integrates with outsource knowledge bases (e.g. DBpedia or local databases). Furthermore, we construct the semanticvector class to encode every entity's properties in numerical vector space. Algorithms are proposed to quantify dimensional values for each instance of semanticvector. The recommendation engine is designed to generate top-K recommendations based on the

calculation of semantic similarity or the use of supervised learning models. Two experiments are conducted and the experimental results confirm the feasibility of our proposed framework.

The rest of this paper is organized as: Section 2 describes the related work. In section 3, the TOML, the architecture of TOML-based tourism recommender and its decision-making process are presented. Section 4 draws the experiments and discusses the results. Finally, section 5 gives the conclusion and states the future work.

## 2. RELATED WORK

In this section, we analyze the recent methods of tourism recommenders including machine learning and semantic web based approaches. The reviews of recommender systems and tourism recommenders are out of the scope of this study and can be found in the following surveys [6], [7], respectively. Traditionally, collaborative filtering, contentbased filtering and hybrid methods are dominant approaches to recommender systems. The strength and weakness of these methods are analyzed in [6]. Besides, machine learning is also applied in recommenders for giving personalized recommendations. Specifically, classification methods that are widely used in making recommendations are support vector machine (SVM) [8], k-nearest neighbors (kNN) [9], artificial neural network (ANN) [10], decision tree [11] or ensemble method [12] to name a few. In the domain of tourism recommenders, traditional methods [13] and machine learning methods [14] are also introduced to the literature. Both traditional recommender methods and machine learning-based methods are data dependent. This means that the quantity and the quality of data decide the performance of recommender systems.

However, the lack of data often occurs in recommender studies. This is the root of the cold-start problem of recommender [15]. In order to support recommenders in building its prediction model, researchers have used supplementary datasets to overcome this difficulty. In which, open linked data has been adopted as the modern approach [15]. The use of open linked data and reasoning techniques of semantic web technology are also found in tourism recommenders [4] and in other kinds of recommenders [5]. As a result, combining machine learning with open linked data and semantic web technology has become a rising trend in recommender studies. In this paper, our target not only provides a new hybrid framework but also presents a new ontology to tourism recommender. The rest of this section reviews the recent studies with a focus on: i) ontology engineering methodologies; ii) ontologies for the tourism industry; and iii) discussion about the distinct characteristics of our proposed approach.

### 2.1. Ontology engineering methodologies

In 2001, Berner-Lee *et al.* [16] proposed the Semantic Web initiative which highlights the key role of ontology as an efficient way to capture domain knowledge in machine understandable format. Since then, the research trend named ontology engineering, which focuses on methods of developing domain ontology, has been raised. In this research trend, the tutorial of Noy and McGuinness [17] can be seen as one of the most popular methods of ontology building. The authors proposed 7 step method including: i) determine the scope of the domain ontology; ii) reuse existing ontologies; iii) enumerate domain concepts; iv) construct the class and the class hierarchy; v) define the properties of the class-slots; vi) define the facets of the slots; and vii) create instances. Although this method is efficient, it faces the difficulties in ontology evolution and collaborative building of ontology. Therefore, different ontology engineering methods have been presented. For example, Fernandez-Lopez *et al.* [18] focused on the major subtasks to develop new ontologies and the evolution of ontology throughout its lifetime. In another approach, Sure *et al.* [19] presented on-to knowledge methodology (OTKM) which takes account of the knowledge processes and the knowledge meta processes. The former process relates to the usage of ontologies, while the latter process makes initial setup. OTKM introduces the ways of integrating ontology in knowledge management applications. The NeON methodology [20] is different from previous methodologies. While previous studies build standalone ontologies, the NeON methodology constructs an ontology network by connecting different existing ontologies through their relationships.

### 2.2. Ontologies for tourism industry

Recently, Semantic Web technology and ontology have been applied to tourism recommenders in many aspects. To be more specific, Antonio Moreno *et al.* [21] used ontology to capture knowledge of tourism objects and populated the ontological instances with scores. These scores were the inputs of the recommendation algorithm. Lin Shi *et al.* [22] provided tourism recommendations based on the user's context. In which, ontology was used to describe and integrate tourism resources. Based on this knowledge foundation, the reasoning process was implemented to make personalized recommendations. Grun *et al.* [4] introduced an ontology-based method to support tourists' decision-making during their pre-trip phase. The authors matched tourists' profiles with characteristics of tourism objects through vector space where each

dimension is a tourist factor. In another approach, P. Ferraro and L. R. Giuseppe [23] proposed an architecture of a semantically adaptive recommender system assisting users in the travel planning phase and in on-site phase. Hybrid method of tourism recommender was also introduced to the literature in the research of Yan Chu *et al.* [24]. Firstly, the authors used association rules to find out related users and unrelated users. Secondly, for each group of users, they applied different collaborative filtering algorithms to make recommendations. Finally, the recommendations were expanded by using a tourism ontology.

### 2.3. Discussion

Both recommenders and machine learning models require data which is often in numerical vector format. However, this kind of data is not always available, especially in the research line of tourism recommender. On the other hand, there are many valuable open linked data sources (e.g. DBpedia), which reside under graph-based formats, can efficiently support the recommendation making process. The problem is to transfer directly the graph-based data to numerical vectors in order to serve different machine learning models in predicting user's preferences or generating top-K personalized recommendation lists. To solve this problem, our proposed framework is different from the aforementioned research in the following three aspects. Firstly, we introduce a new tourism ontology based on domain expert collaboration and outsource knowledge integration. Secondly, a Semanticvector concept is used to describe every entity of the ontology in a vector space model. This component provides semantically numerical data for all machine learning tasks including classification and clustering. Thirdly, we present algorithms for the recommendation engine which use directly the semantic numerical data in the recommendation making process. This approach is different from the previous use of other ontologies in the tourism domain.

## 3. TOML-BASED RECOMMENDER FRAMEWORK

In this section, we describe our ontological approach to the tourism recommender named TOML. TOML-based recommender framework has three major parts including TOML ontology, methods of populating TOML knowledge base and TOML-based recommendation engine. Figure 1 shows the overall architecture of this framework.

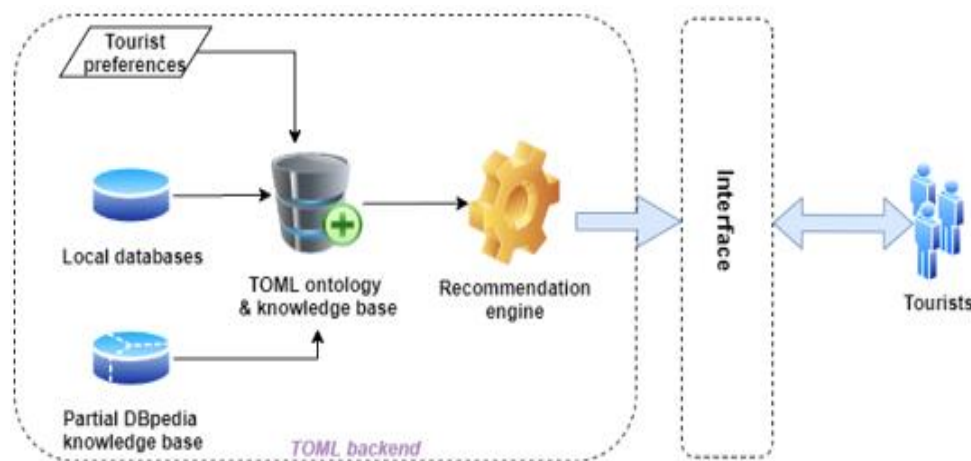


Figure 1. The overall architecture of TOML-based recommender framework

In this framework, the TOML ontology was designed through the proposed six-step process which is presented in the subsection 3.1. The TOML knowledge base was enriched by different ways like importing from DBpedia, local databases and tourists' preferences data. The enriching methods are discussed in the subsection 3.2. Subsection 3.3 introduces the recommendation engine of this framework.

### 3.1. TOML ontology

In general, a domain ontology can be defined as in Definition 1.

- Definition 1 The ontology of the domain  $D$  denoted as  $O_D$  is a triple  $O_D = \langle C_D; R_D; I_D \rangle$  where  $C_D$  is the set of domain concepts,  $R_D$  and  $I_D$  are the sets of domain properties (relations) and domain instances, respectively.

In order to build TOML ontology, we invited tourism expertises and knowledge engineers to work together. The working process of this group includes six steps: At first, we adapted the method of [17] for creating the first draft of the knowledge base. Specifically, expertises enumerated the concepts and relations of the tourism domain. Then, knowledge engineers transferred these information to ontology structure using Protégé [25] software. Secondly, the first step was repeated until all of the expertises and engineers reached their consensus. Thirdly, further specific descriptions were added to the ontology (e.g. the SemanticVector class). Fourthly, we enriched the ontological instances by using our local database and importing data from open knowledge-base (e.g. DBpedia) through mapping operations. Fifthly, we iterated over each entity of TOML knowledge base and computed its correspondent semantic vector by using our proposed algorithms. Finally, the ontology was carefully checked by both the expertises and the engineers in order to reach its first version. An excerpt of TOML is shown in Figure 2.

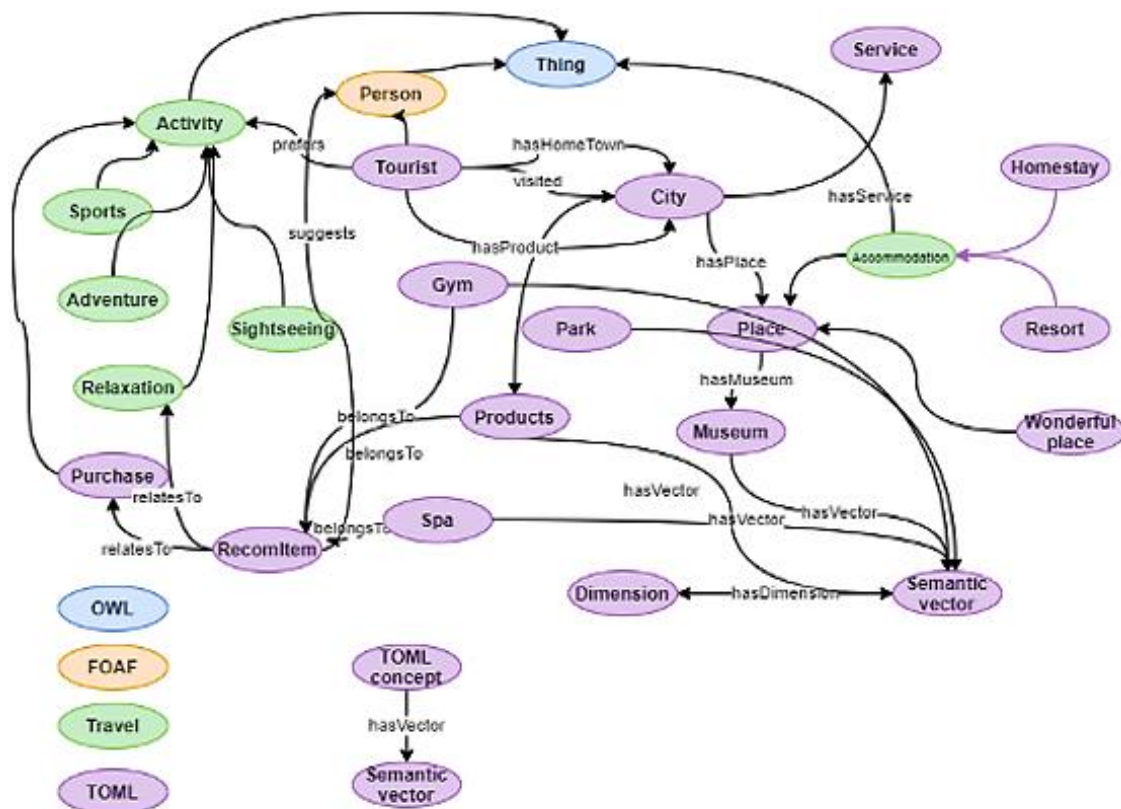


Figure 2. An excerpt of TOML ontology

In general, TOML has 157 concepts, 65 object properties and 24 data properties. Due to these large numbers of concepts and properties, we describe TOML by summarizing its characteristics and highlight our own contribution in specifying the tourism domain knowledge. Firstly, we develop concepts that relate to tourist, place, service, facility and activity. For example, the concept `toml:Tourist` is inherited from `foaf:Person` concept and has three different object properties with `toml:City` concept including `toml:hasHomeTown`, `toml:visited` and `toml:visits`. The `toml:Tourist` concept plays the key role of our ontology in capturing the knowledge about tourist's personal information (e.g., gender, and name), tourists' preferences through the relation with `travel:Activity` and its subconcepts.

Secondly, we elaborate and specify more concepts about tourist's activity like `toml:Purchase`, `toml:Listen` or `toml:Festival` to name a few. These activity concepts are efficient in capturing tourist's preferences. And they are used in the first phase of the recommendation process by linking with other concepts through `toml:suggest` object property.

Thirdly, every sub-concept of `toml:Place`, `toml:Products` or `toml:Service` has relation with `toml:SemanticVector` concept. This concept provides the quantitative vector for every entity of the related concept. This vector is the base for any further use of machine learning models or decision making process. We propose specific algorithms to build semantic vectors for every related entity of TOML knowledge base.

Finally, we propose the `toml:RecomItem` concept to capture one or more recommended things. For example, in case that tourists prefer to buy products, and the products are found in a local market where it is required to use the public transport service to go to, the recommended items for tourists should take account of not only the product itself but also the available public transport service and route guide. This is the different characteristic of tourism recommendation in comparison with other kinds of recommenders like books or movies.

### 3.2. Enriching TOML knowledge base

In order to populate the TOML knowledge base, we imported relevant data from open linked data sources (e.g., DBpedia) and local databases to the TOML knowledge base. The importing process depends on the mapping methods of class and property. In which, correspondent concepts between Dbpedia and TOML were figured out. Similarly, the mapping rules between database tables and TOML ontology were defined. Then, relevant DBpedia entities and their properties were selected by SPARQL queries and were exported to RDF/JSON format. In case of integrating local databases into TOML, the relevant table records were selected by and exported to RDF/JSON files. Finally, these batch files were imported directly to the TOML knowledge base. The pseudo codes of importing data from DBpedia and local database are shown in Figures 3 and 4, respectively.

---

```

corr_dbpedia ← map_bt看(toml;dbpedia);
files ← [];
foreach c in corr_dbpedia do
  entities ← select_all_entities(c);
  foreach e in entities do
    props ← select_all_props(e);
    files ← export(c;e; props; 'RDF=JSON');
import_TOML(files);

```

---

Figure 3. Pseudo-algorithm of retrieving relevant knowledge from DBpedia

---

```

(concepts;tables) ← map_bt看(toml;tables);
files ← [];

foreach (c, t) in (concepts, tables) do
  records ← select_all(t);
  files ← export(c;t;records; 'RDF=JSON');

import_TOML(files);

```

---

Figure 4. Pseudo-algorithm of populating TOML ontology by local database

The primary purpose of TOML knowledge base is to provide data for machine learning models. While machine learning models require inputs as numerical vectors, open linked data (e.g., DBpedia, TOML knowledge base) provide data under graph-based formats (e.g., RDF, OWL). We transferred the property value of an entity by using (1). Then, our solution to building numerical vectors based on available linked data for every TOML's entity applied (1) in pseudo-algorithm of Figure 5. Each property of the entity now plays the role of a dimension in the semantic vector.

$$value = \log \left( \frac{1}{N_{triples}^{<c,p,v>} + 1} \right) \quad (1)$$

where  $N_{triples}^{<c,p,v>}$  triples is the total number of triples which have the same subject concept (class) -  $c$ , the same property -  $p$  and the same property value -  $e$ .

By implementing the algorithm shown in Figure 5, every entity has its own semantic vector, however, some properties may appear or not in different entities. In other words, different entities may have different vector spaces. Therefore, building the common vector space for all selected semantic vectors is necessary. Firstly, all semantic vectors related to the recommendation task are selected by SPARQL SELECT query. Then, all of the distinct properties are figured out and are sorted in ascending order of property names.

These are the dimensions of the vector space. Finally, for each semantic vector, its original values are filled properly into corresponding dimensions. The rest of dimensions, which are not filled, receive zero values. This procedure is expressed in Figure 6.

---

```

Input:  $c$  - a given TOML concept
foreach entity  $e \in c$  do
  foreach property  $p \in e$  do
    calculate  $N_{\langle c,p,v \rangle}^{\text{triples}}$ ;
     $\text{dim\_value} = \log\left(\frac{1}{N_{\langle c,p,v \rangle}^{\text{triples}} + 1}\right)$ ;
     $e.\text{SemanticVector.add}(\text{dimension} = p,$ 
       $\text{value} = \text{dim value});$ 

```

---

Figure 3. Pseudo-algorithm of calculating semantic vector

### 3.3. TOML-based recommendation engine

TOML-based recommendation strategies were designed to cope with the two popular recommendation cases: (i) with the availability of tourist preference data; and (ii) without the availability of tourist preference data. In case that the tourist preference data is not available, the recommendation strategy is as: Assuming that tourists want to get a top-K recommendation list about a given concept (e.g. place, food or product). First, an entity relating to the recommended concept is randomly selected via a SPARQL SELECT query. This entity should be specified as “famous” in the knowledge base. We use this entity as the starting point and find other (k-1) nearest entities by calculating semantic similarity between this entity and the other entities within the same concept. The Euclidean distance is accepted to compute semantic similarity. The pseudocode of this strategy is shown in Figure 7.

---

```

Input:  $V_D$  - list of semantic vectors in different vector spaces
Output: result - list of semantic vectors in the common vector space  $V_C$ 
 $V_C \leftarrow \{ \};$ 
foreach  $v_i \in V_D$  do
  foreach  $j$  in range length( $v_i$ ) do
     $\text{dim}_i^j \leftarrow \text{arg}(v_i[j]);$ 
    if  $\text{dim}_i^j \in V_C$  then
       $V_C \leftarrow V_C \cup \{ \text{dim}_i^j \};$ 
 $\text{result} \leftarrow \{ \};$ 
foreach  $v_i \in V_D$  do
   $v_i^c \leftarrow \{ \};$ 
  foreach  $j$  in range length( $v_i$ ) do
     $\text{dim}_i^j \leftarrow \text{arg}(v_i[j]);$ 
    foreach  $k$  in range length( $v_i^c$ ) do
      if  $\text{dim}_i^j = \text{arg}(v_i^c[k])$  then
         $v_i^c[k] \leftarrow v_i[j];$ 
        break;
   $\text{result.add}(v_i^c);$ 
return result;

```

---

Figure 6. Pseudo-algorithm of constructing common vector space

---

```

Input:  $s$  - SPARQL SELECT query
 $c$  - the concept that needs to get recommendations
 $n$  - the size of common vector space
 $k$  - number items in top-K recommendation list
Output: result - the top-K recommendation list
 $\text{famous\_ent} \leftarrow \text{implement\_query}(s);$ 
 $\text{entities} \leftarrow \text{get\_entities}(c);$ 
 $\text{vectors} \leftarrow \text{get\_semantic\_vector}(\text{entities});$ 
 $\text{result} \leftarrow \{ \};$ 
foreach  $(e, v) \in (\text{entities}, \text{vectors})$  do
   $\text{sim} = \sqrt{\sum_{i=1}^n (\text{famous\_ent}_{e_{nt_i}} - v_i)^2};$ 
   $\text{results.add}((e, \text{sim}));$ 
 $\text{result.sort\_asc}(\text{sort\_by} = \text{sim});$ 
 $\text{result} \leftarrow [\text{famous\_ent}, \text{result}];$ 
 $\text{result} \leftarrow \text{result}[0 : k];$ 
return result;

```

---

Figure 7. Pseudo-algorithm of top-K recommendations based on semantic similarity

In case that tourists provide preference data for creating labeled data, the supervised learning models are applied to generate top-K personalized recommendation items. Different classification models can be plugged into the recommendation engine via parameter input. And the prediction scores were used to rank the top-K recommendation list. Figure 8 shows the pseudocode of this strategy.

Based on the top-K recommendation list, which is generated by algorithms in either Figure 7 or Figure 8, the route planning algorithm is applied to find the shortest path from the tourist’s current location to all locations of  $k$  suggested items. The location data was stored in TOML knowledge base and Google



map API was used to find the location-to-location route. Figure 9 shows the pseudocode of route planning recommendation.

---

**Input:**  $D$  - training data,  $M$  - the classification model,  
 $t$  - tourist profile,  $C$  - the concerning concept,  
 $k$  - number of recommended items

**Output:** result - the top- $K$  recommendation list

```

trained_M ← train_model(T, D);
entities ← select_unseen_entities_of_concept(t, C);
scores ← []; ents ← [];
foreach e in entities do
    label, score ← trained_M.predict(
select_semantic_vector( e ));
    if label = 'like' then
        ents.add(e);
        scores.add(score);
sort(ents, by=scores, order='desc');
result ← ents[0 : k];
return result;

```

---

Figure 8. Pseudo-algorithm of top- $K$  recommendations generated by classifiers

---

**Input:** top\_  $K$  - the top- $K$  recommendation list  
 $curr\_loc$  - tourist's current location  
 $k$  - number of recommended items

**Output:** route - the suggested route

```

locations ← [curr_loc];
foreach entity in top_ K do
    location ← select_location(entity);
    locations.add(location);
adjacency_matrix ← build_matrix(locations);
route ← visit_all_vertices(start =curr_loc,
graph = adjacency_matrix);
return route;

```

---

Figure 9. Pseudo-algorithm of route planning recommendation

#### 4. EXPERIMENTS

The experiments were conducted to evaluate the efficiency of TOML knowledge base and its recommendation engine. We developed a prototype in Python programming language which implements all of the algorithms proposed in section 3.1. The tests of user satisfaction and the feasibility of implementing machine learning models with TOML knowledge base were presented in subsections 4.1 and 4.2, respectively.

##### 4.1. Experiment 1: building top- $K$ recommendation list without user preference

In this experiment, tourists' preference data is not available. This situation causes the cold-start problem of the recommendation research field. In real word, the tour guides often provide suggestions without having tourists' preferences. Therefore, we decide to compare the top- $K$  recommendation lists yielded by TOML-based prototype to those of tour guides.

The experiment was designed as: Questionnaires were sent to tour guides of 5 different local tourist companies. The survey closed after 2 months and there were 32 tour guides who completed the survey. The tour guides were asked to give top-10 recommendations for foods, places and products of a given city. Due to the complexity of collected data, we summarized the experimental results in Table 1. To be more specific, top-10 place recommendations generated by both TOML and tour guides are visualized in Figure 10 for better understanding of the recommended results.

As shown in Table 1, all of the p-values of three different groups of top-10 recommendations are greater than 0.05. These statistical results imply that there is no difference in personalized recommendation lists between tour guides and TOML-based prototype. In other words, the TOML-based prototype can provide suggestions as good as those of tour guides. Furthermore, the statistical results also indicate that TOML knowledge base has captured experts' domain knowledge efficiently.

Table 1. p-values results of top-10 recommendations

	Total recommended items		p-values
	TOML	Tour Guides	
Place	10	15	0.57
Food	10	17	0.46
Product	10	14	0.63

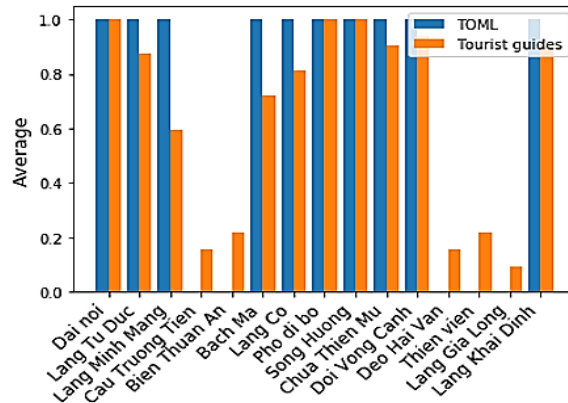


Figure 10. Top-10 places recommended by TOML and by tour guides

**4.2. Experiment 2: Implementing various classifiers with TOML knowledge base**

The purpose of this experiment is to demonstrate the ability of TOML knowledge base in terms of providing data for machine learning models. Specifically, labeled data is required to train supervised learning models for predicting which entity should be presented to tourists. However, it was hard to ask tourists to join this experiment. Hence, we invited tour guides joining this experiment under the role of tourists. Each participant figured out which things (places, foods, and products,) she or he likes or dislikes. These preferences were updated to the correspondent entities in TOML knowledge base via the data property `toml:hasPreference`. This property was also added to the semantic vector as the label dimension. The preference data were associated with the concept `toml:Tourist` which captures tourist profiles.

There were 6 tour guides who participated in this experiment and constructed 327 records of their preferences. We used three popular classification models including k-NN, Naive Bayes and SVM to predict personalized tourism recommendations. There were three suggestion lists about place, food and product. Each participant evaluated on every suggested item that she or he satisfied or not. Table 2 shows this experiment results.

It is important to emphasize that this experiment does not target at introducing new classifiers with highly predicted capabilities but demonstrating that machine learning models can work well directly with TOML knowledge base. As indicated in Table 2, the averages of satisfied ratios range from 40% to 63.3%, while those of unsatisfied ratios range from 36.7% to 56.7%. The three traditional classifiers reach and overcome the 50% threshold 6 times in total. These results confirm the efficiency usage of TOML-based semantic vectors in machine learning models.

Table 1. Users react to recommendation lists generated by machine learning models

		User 1		User 2		User 3		User 4		User 5		User 6		Average	
		satisfy	not	satisfy	not	satisfy	not	satisfy	not	satisfy	not	satisfy	not	satisfy	not
Place	SVM	5	5	4	6	6	4	5	5	6	4	7	3	5.5	4.5
	Naïve Bayes	3	7	4	5	4	5	5	5	2	8	6	4	4	5.67
	k-NN	4	6	4	6	7	3	6	4	3	7	3	7	4.5	5.5
Food	SVM	7	3	5	5	4	6	7	3	8	2	7	3	6.33	3.67
	Naïve Bayes	6	4	6	4	2	8	4	6	6	4	3	7	4.5	5.5
	k-NN	6	4	9	1	3	7	7	3	6	4	6	4	6.17	3.83
Product	SVM	1	9	9	1	7	3	1	9	5	5	7	3	5	5
	Naïve Bayes	2	8	4	6	8	2	6	4	5	5	8	2	5.5	4.5
	k-NN	5	5	4	6	1	9	8	2	5	5	8	2	5.17	4.83

**5. CONCLUSION**

In this paper, an ontological framework supporting model-based tourism recommender so-called TOML has been presented. The recommendation making process makes use of the strength of both Semantic Web technology and supervised learning models. Especially the design of TOML ontology composes of a specific semantic vector concept which enables machine learning models to consume data directly in graph-based structure of ontological knowledge base. Two experiments were conducted to validate the efficiency



and the promising usage of TOML-based framework. While the results obtained from experiment 1 indicate that TOML knowledge base has captured experts' domain knowledge efficiently, those gained from experiment 2 confirm the efficiency usage of TOML-based semantic vectors in machine learning models. The future work of this study will focus on building TOML-based web service and integrating TOML ontology with other tourism ontologies in order to enlarge the knowledge base and building TOML-based framework as a backbone of tourism recommendation service.

## REFERENCES

- [1] F. M. Harper and J. A. Konstan, "The movielens datasets: history and context," *ACM Transactions on Interactive Intelligent Systems*, vol. 5, no. 4, pp. 1-19, 2015, doi: 10.1145/2827872.
- [2] C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen, "Improving recommendation lists through topic diversification," *WWW '05: Proceedings of the 14th International Conference on World Wide Web*, 2005, pp. 22-32, doi: 10.1145/1060745.1060754
- [3] C. Bizer *et al.*, "DBpedia-a crystallization point for the web of data," *Journal of Web Semantics*, vol. 7, no. 3, pp. 154-165, Sep. 2009, doi: 10.1016/j.websem.2009.07.002.
- [4] C. Grün, J. Neidhardt, and H. Werthner, "Ontology-based matchmaking to provide personalized recommendations for tourists," in *Information and Communication Technologies in Tourism 2017*, 2017, pp. 3-16.
- [5] K. Bagherifard, M. Rahmani, M. Nilashi, and V. Rafe, "Performance improvement for recommender systems using ontology," *Telematics and Informatics*, vol. 34, no. 8, pp. 1772-1792, 2017, doi: 10.1016/j.tele.2017.08.008.
- [6] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowledge-Based Systems*, vol. 46, pp. 109-132, Jul. 2013, doi: 10.1016/j.knosys.2013.03.012.
- [7] J. Borràs, A. Moreno, and A. Valls, "Intelligent tourism recommender systems: a survey," *Expert Systems with Applications*, vol. 41, no. 16, pp. 7370-7389, Nov. 2014, doi: 10.1016/j.eswa.2014.06.007.
- [8] L. Ren and W. Wang, "An SVM-based collaborative filtering approach for top-n web services recommendation," *Future Generation Computer Systems*, vol. 78, pp. 531-543, Jan. 2018, doi: 10.1016/j.future.2017.07.027.
- [9] D. K. Chae, S. C. Lee, S. Y. Lee, and S. W. Kim, "On identifying k-nearest neighbors in neighborhood models for efficient and effective collaborative filtering," *Neurocomputing*, vol. 278, pp. 134-143, Feb. 2018, doi: 10.1016/j.neucom.2017.06.081.
- [10] T. K. Paradarami, N. D. Bastian, and J. L. Wightman, "A hybrid recommender system using artificial neural networks," *Expert Systems with Applications*, vol. 83, pp. 300-313, Oct. 2017, doi: 10.1016/j.eswa.2017.04.046.
- [11] S. Linda and K. K. Bharadwaj, "A decision tree-based context-aware recommender system," in *Intelligent Human Computer Interaction*, 2018, pp. 293-305, doi: 10.1007/978-3-030-04021-5\_27.
- [12] J. Liu *et al.*, "Representing conditional preference by boosted regression trees for recommendation," *Information Sciences*, vol. 327, pp. 1-20, Jan. 2016, doi: 10.1016/j.ins.2015.08.001.
- [13] M. Nilashi, O. bin Ibrahim, N. Ithnin, and N. H. Sarmin, "A multi-criteria collaborative filtering recommender system for the tourism domain using Expectation maximization (EM) and PCA-ANFIS," *Electronic Commerce Research and Applications*, vol. 14, no. 6, pp. 542-562, Oct. 2015, doi: 10.1016/j.elerap.2015.08.004.
- [14] M. Nilashi, K. Bagherifard, M. Rahmani, and V. Rafe, "A recommender system for tourism industry using cluster ensemble and prediction machine learning techniques," *Computers and Industrial Engineering*, vol. 109, pp. 357-368, Jul. 2017, doi: 10.1016/j.cie.2017.05.016.
- [15] S. Natarajan, S. Vairavasundaram, S. Natarajan, and A. H. Gandomi, "Resolving data sparsity and cold start problem in collaborative filtering recommender system using linked open data," *Expert Systems with Applications*, vol. 149, Jul. 2020, doi: 10.1016/j.eswa.2020.113248.
- [16] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific American*, vol. 284, no. 5, pp. 28-37, 2001.
- [17] N. F. Noy and D. L. McGuinness, "Ontology development 101: A guide to creating your first ontology," Stanford Knowledge Systems Laboratory, Tech. Rep. KSL-01-05, Mar. 2001.
- [18] M. Fernández-López, A. Gómez-Pérez, and N. Juristo, "Methontology: from ontological art towards ontological engineering," AAI, Tech. Rep., 1997.
- [19] Y. Sure, S. Staab, and R. Studer, "On-to-knowledge methodology (OTKM)," in *Handbook on ontologies*, Springer, 2004, pp. 117-132.
- [20] M. C. Suarez-Figueroa *et al.*, "NeOn methodology for building contextualized ontology networks," *NeOn Deliverable D*, vol. 5, pp. 4-1, 2008.
- [21] A. Moreno, A. Valls, D. Isern, L. Marin, and J. Borràs, "SigTur/E-destination: ontology-based personalized recommendation of tourism and leisure activities," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 1, pp. 633-651, Jan. 2013, doi: 10.1016/j.engappai.2012.02.014.
- [22] L. Shi, F. Lin, T. Yang, J. Qi, W. Ma, and S. Xu, "Context-based ontology-driven recommendation strategies for tourism in ubiquitous computing," *Wireless Personal Communications*, vol. 76, no. 4, pp. 731-745, Jun. 2014, doi: 10.1007/s11277-013-1550-9.
- [23] P. Ferraro and G. Lo Re, "Designing ontology-driven recommender systems for tourism," in *Advances onto the Internet of Things: How Ontologies Make the Internet of Things Meaningful*, 2014, pp. 339-352.
- [24] Y. Chu, H. Wang, L. Zheng, Z. Wang, and K. L. Tan, "TRSO: A tourism recommender system based on ontology," *Int. Conf. on Knowledge Science, Enginee. Management*, 2016, pp. 567-579, doi: 10.1007/978-3-319-47650-6\_45.
- [25] M. A. Musen, "The protégé project: a look back and a look forward," *AI Matters*, vol. 1, no. 4, pp. 4-12, 2015.