A Knowledge Representation Based User-Driven Ontology Summarization Method

Yuehang DING†(a), Member, Hongtao YU†, Jianpeng ZHANG†, Huanruo LI†, and Yunjie GU†(b), Nonmembers

SUMMARY As the superstructure of knowledge graph, ontology has been widely applied in knowledge engineering. However, it becomes increasingly difficult to be practiced and comprehended due to the growing data size and complexity of schemas. Hence, ontology summarization surfaced to enhance the comprehension and application of ontology. Existing summarization methods mainly focus on ontology’s topology without taking semantic information into consideration, while human understand information based on semantics. Thus, we proposed a novel algorithm to integrate semantic information and topological information, which enables ontology to be more understandable. In our work, semantic and topological information are represented by concept vectors, a set of high-dimensional vectors. Distances between concept vectors represent concepts’ similarity and we selected important concepts following these two criteria: 1) the distances from important concepts to normal concepts should be as long as possible which ensures that important concepts are not similar to each other. K-means++ is adopted to select important concepts. Lastly, we performed extensive evaluations to compare our algorithm with existing ones. The evaluations prove that our approach performs better than the others in most of the cases.

key words: ontology, ontology summarization, semantic distance, K-means++

1. Introduction

As a structural knowledge base, knowledge graph has been increasingly applied to the intelligent semantic searches, mobile personal assistants and question answering systems. Ontology is the superstructure of knowledge graph. It plays an important role in both knowledge reasoning and knowledge graph construction. With the explosive growth of data size and ontology complexity, ontology’s understanding and application are becoming more and more difficult. Ontology summarization is the process to distill knowledge from ontologies. It can effectively reduce the difficulty of ontology understanding and speed up ontology application, thus it effectively alleviates the above problems.

Ontology summarization is the process of distilling knowledge from ontology to produce an abridged version for particular users and tasks[1]. Existing ontology summarization methods can be divided into user-driven and task-driven according to their application requirements[2]. The user-driven summarization aims to help users quickly understand an ontology. The task-driven summarization focuses on accelerating a specific task. Most existing user-driven summarization methods extract important concepts according to the ontologies’ topology, rarely taking semantic information of concepts into consideration. However, we find that the more semantic information is used, the higher the accuracy of the summarization: Xiang Zhang et al.[1] proposed a summarization method based on the ontology structure, considering the triples as nodes, projecting ontologies to graphs. Peroni et al.[3] transferred ontologies into graphs by considering the concepts as nodes and the relations as edges. Apart from ontologies’ topological information, they also chose important concepts by name simplicity, which preferred concepts labeled with simple names. Paulo et al.[4] extracted important concepts from ontologies according to their topology and their relations’ labels. The accuracy of the above three algorithms is increasing in order. At the same time, their extent of utilizing semantic information is incremental. In fact, summarizations without semantic information may neglect representative concepts connecting with fewer relations, which causes a low accuracy. To further improve the accuracy of ontology summarization, we propose a knowledge representation based summarization method which utilizes concepts’ semantic information.

In this paper, we transfer concepts into embeddings based on ontologies’ structural and semantic information. The distance from concept A to concept B indicates the possibility that A can summarize B: the shorter the distance from A to B is, the better A can summarize B. We hope that the distances from important concepts to normal concepts can be as short as possible, which means important concepts can summarize normal concepts well. At the same time, we hope that the distances from an important concept to the others can be as long as possible, which means each important concept is not similar to the others. To satisfy the limitations above, we put the idea of K-means++ on concept embeddings to select important concepts. Specifically, our contributions are listed as follows:

A novel ontology summarization method is proposed, which takes ontologies’ semantic information and topological information into consideration.

To measure the concepts’ summarizing ability, concept embeddings are generated based on the word embedding algorithm, i.e., word2vec. Distance from one concept to another represents their similarity.

To select important concepts from ontologies, we pro-
pose a maximum-minimum iteration method using the idea of K-means++ method.

To avoid extracting similar concepts as important concepts, we set the similarity among important concepts as one of our algorithm’s decision criteria. To our knowledge, existing methods only consider the similarity between important concepts and normal concepts, and do not consider that important concepts should be as dissimilar as possible.

2. Concept Distances

In this section, we transfer concepts into embeddings to calculate their similarities. Firstly, we use Google’s pre-trained word vectors to represent the words in concepts. Then we define a concept distance measurement based on these vectors. By this measurement, we can calculate the semantic distances between all concept pairs. Next, we define a measurement to calculate the probability that a concept can summarize the others according to the ontology’s topology. Finally, we integrate the topological probability into semantic distances to get the final distances between concepts.

2.1 Semantic Distance

Word2vec is a tool for computing continuous distributed representations of words. These representations contain words’ semantic information. To transfer concepts into vectors, we use Google’s pre-trained word vectors to represent words in concepts. We define the semantic distance between concepts as follows.

Definition 1 (Semantic Distance) Assume word \( w_i \in c_1, w_j \in c_2 \), \( c_1 \) and \( c_2 \) are two concepts in the target ontology. Then the semantic distance from \( c_1 \) to \( c_2 \), denoted by \( D_{c_1, c_2}(c_1, c_2) \), is calculated by Eq. (2):

\[
D_{c_1, c_2}(c_1, c_2) = \sum_{w_i \in c_1} \sum_{w_j \in c_2} \min(dist(w_i, w_j))
\]

where \( dist(w_i, w_j) \) is the semantic distance between \( w_1 \) and \( w_2 \). \( w_2(w_i) \) is word \( w_i \)’s embedding, and \( L2(x, y) \) is the L2 norm of \( x, y \). Note that \( D_{c_1, c_2}(c_1, c_2) \neq D_{c_2, c_1}(c_2, c_1) \). The semantic distance from \( c_1 \) to \( c_2 \) measures the semantic similarity from \( c_1 \) to \( c_2 \). It means the more \( c_1 \) can summarize \( c_2 \) semantically, the shorter the distance is. Besides, this measurement tends to give shorter distance to concepts containing fewer words, which matches the criterion, i.e., name simplicity, proposed in [3].

2.2 Topological Probability

To integrate the topological information into semantic distance, we define the notion of the topological probability according to relations between concepts.

Definition 2 (Topological Probability) Assume a triple \((c_1, r, c_2)\) in an ontology, where \( c_1, c_2 \) are concepts, \( r \) is the relation from \( c_1 \) to \( c_2 \). Define the weight of relation \( r \) as \( \text{weight}(r) \), the topological probability that \( c_1 \) can summarize \( c_2 \) as \( p(c_1, c_2) \).

\[
\text{weight}(r) = \begin{cases} 
0, & r = \text{subClassOf} \\
1, & r = \text{inv_subClassOf} \\
0.5, & r = \text{other_relations} 
\end{cases} \tag{3}
\]

\[
p(c_1, c_2) = \max_i \left( \prod_{k \in \text{path}_i} \text{weight}(r^k) \right) \tag{4}
\]

where \( r^k \) is the \( k \)-th relation in relation path \( i \). A relation path from \( c_1 \) to \( c_2 \) consists of a path of relations through which \( c_1 \) can reach \( c_2 \). If there exists \((c_1, \text{subClassOf}, c_2)\), which means \( c_2 \) is the superclass of \( c_1 \), then we can affirm that \( c_2 \) can summarize \( c_1 \). \( c_1 \) cannot summarize \( c_2 \). Thus we set \( p(c_1, c_2) = 0 \), \( p(c_2, c_1) = 1 \). For other kinds of relations, we set the probability be 0.5. The topological probability quantifies relations between concepts. Thus we can integrate the topological information into semantic distances. Finally, we recalculate the distance from \( c_1 \) to \( c_2 \) by Eq. (5):

\[
D(c_1, c_2) = D(c_1, c_2)(1 - p(c_1, c_2)) \tag{5}
\]

where \( D(c_1, c_2) \) is the final distance from \( c_1 \) to \( c_2 \).

3. Selection of Important Concepts

Having calculated the distances between concepts, we focus on the selection of important concepts. We select important concepts based on the following two criteria: firstly, we hope that the distances from selected concepts to unselected concepts can be as short as possible. It means that the selected concepts can summarize the unselected concepts well. At the same time, we hope that the distances between selected concepts can be as long as possible. It means the selected concepts are not similar to each other.

We use the idea of K-means++ to select the important concepts. The exact algorithm is as follows:

1. Consider concepts as nodes, relations as edges, choosing initial cluster centers according to K-means++.

2. Assign each node to the cluster whose mean has the least concept distance.

3. For each cluster, select the node with the shortest sum of distances to other nodes in the cluster to be the centroid.

4. Repeat step 2 and 3 until the assignments no longer change.

With precisely selected important concepts, users can understand the target ontology clearly and quickly. Our algorithm is divided into the following three steps:

1. Calculate the semantic distances between concepts based on Google’s word vectors.

2. Combine ontology’s topological information with concepts’ semantic distances.

3. Select important concepts with the idea of K-means++.
We perform an extensive evaluation to assess the effectiveness of our algorithm. The above three ontologies are used to compare our algorithm with the algorithms proposed by Peroni et al. [3], Queiroz-Sousa et al. [4] and Troullinou et al. [6]. To compare these algorithms, we use the reference summaries and the results published in [3] and [4]. Peroni et al. [3] combined cognitive science, network topology, and lexical statistics to automatically select important concepts. Queiroz-Sousa et al. [4] proposed an algorithm that selected important concepts through centrality measures or user indication. Troullinou et al. [6] used the information from the data layer to calculate the relative cardinality of relations. Then they combined relative cardinality with node centrality to select important concepts. The reference summaries used in this paper were generated by Peroni et al. and were also used by Queiroz-Sousa and Troullinou in their evaluations [6]. The reference summaries were generated by eight human experts, who were requested to select up to 20 concepts [3].

We evaluate the degree of agreement between a generated summary and a reference summary by the measurement proposed in [6]. Assuming Sim(G_S, G_R) is the similarity between two summaries, G_S = (V_S, E_S) is the generated summary, G_R = (V_R, E_R) is the reference summary. Assuming {c_k,...,c_p} are the classes in V_R that are subclasses of the classes {c'_1,...,c'_n} of V_S. Assuming {c_m,...,c_n} are the classes in V_R that are superclasses of the classes {c'_1,...,c'_n} of V_S. Sim(G_S, G_R) is defined as follows:

\[
Sim(G_S, G_R) = \frac{|V_S \cap V_R| + 0.6 \star \sum_{i \neq k} \frac{1}{|V_{c_i} - V_{c'_i}|} + 0.3 \star \sum_{i \neq m} \frac{1}{|V_{c'_i} - V_{c_i}|}}{|V_R|} \tag{6}
\]

Consequently, the summarization accuracy of an algorithm is calculated by the average of the similarities between the generated summary and expert selected summaries [6].

4.2 Experiments

To evaluate the effectiveness of our algorithm, we put our algorithm to the three ontologies introduced in Table 2. We calculate the similarities between summaries produced by our algorithm and those expert-selected ones. The summaries produced by our algorithm are shown in Table 3.

We compared the four algorithms by their similarities with reference summaries. The result is shown in Fig. 2.

We can observe that our algorithm performs better in
Table 3 Synthetic network parameter.

<table>
<thead>
<tr>
<th>Ontology</th>
<th>Algorithm choices</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bank</td>
<td></td>
</tr>
<tr>
<td>Biosphere</td>
<td></td>
</tr>
</tbody>
</table>

Biosphere and Aktors Portal ontology. Although our algorithm performs slightly worse than Troullinou’s algorithm in Financial ontology, it performs better than others in other cases.

The reason that our algorithm performs worse than Troullinou’s in Financial is that the meaning of a concept label may differ from the words in it. In Financial ontology, many concept labels consist of several words. The meaning of these words may be irrelevant with the meaning of the entire label. However, our algorithm calculates semantic similarity between two labels according to the words in them. Thus the accuracy in Financial is at a lower level.

5. Conclusion

In this paper, we proposed an improved-novel ontology summarization algorithm which made use of concept label’s semantic information. At first we transfer concepts into high-dimensional vectors according to their labels. Then we adjust distances between concept vectors based on the relationship between concepts. Experiments reveal that our algorithm outperforms previous studies merely concerning topological information as we take semantic and topological information into consideration.

From the experiments, we have proved that integrating semantic information into ontology summarization can improve summarization accuracy. Combined with other algorithms, we draw the following conclusion: more semantic information integration leads to more accurate summarization.

Thus, in the future, we will find methods to integrate more semantic information into summarization process. For instance, we will crawl corpora according to concepts, and have concept vectors trained corresponding to those corpora, or jointly training concept vectors on the basis of both corpora and ontology’s topology. Moreover, since experts’ cognition vary, their summaries should be weighted depending on reliability, where weights could be decided by the similarity of their summaries.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61521003).

References


