Research on Web Cross Language Information Retrieval Based on Domain Ontology

Birla Institute of Technology and Science, Pilani

Rohit Kumar Sharma 2012A7PS050H

November 2015

1 Abstract

Translation has been one of the salient concepts of Information Retrieval and information is now no longer limited to the native language of the user. With the enormous increase in information, research in Cross-Language Information Retrieval has gained prominence. In this paper, web cross language information retrieval model based on domain *ontology* is brought forward. Domain ontology is used to bring the conventional information retrieval to semantic level. An approach to incorporate ontology is specified and then the results are validated. The experiment involves querying in English to retrieve travel news from a Chinese website. The results clearly indicate improvement in average precision and recall of retrieval.

2 Problem Statement

Searching for information is part of our daily life in this information era. We desire information in our native language but required information is not always accessible in out native language. This led to the problem of cross-language information retrieval, in which the user queries in one language (*source language*) and the results are retrieved in an other language (*target language*).

CLIR research can be predominantly seen in European languages due to their ease in converting to English. Nevertheless, translation between Chinese and other foreign languages is a strenuous task in itself due to its own limitations. The challenge arise because Chinese language has no spaces between words and the traditional approach involves token extraction from the text.

The quality of translation can be ameliorated by consolidating ontology and semantic web technology in retrieval. Ontology is a nomenclature of the properties and relationships of the terms such that it remains in accordance with the real world. In essence, ontology captures the semantics of the norms. Semantic web which is an extension of the world wide web can be divided into 3 layers: *basic assertion layer*, *schema layer* and *logic layer*. The main idea is to enable information processing automatic by creating models of semantics which can be processed by machines. This paper will focus on semantic-based information retrieval technology for Web English-Chinese translation.

3 Solution Approach

3.1 Bilingual domain ontology

Currently, CLIR is done using query translation model in which queries are translated to the language of the documents. In contrast language translation uses machine readable dictionary and performs poor when many words have more than one meaning. Lack of consideration of semantic information in the above methods directly impacts the results. So *Bilingual Domain Ontology* is adopted to solve the above mentioned problems.

An English-Chinese bilingual domain ontology was built by mutually corresponding synonyms from English and Chinese. The representation of thus formed concepts was such that it did not use words from either of the languages. This was achieved by using special characters or numbers to represent. For example, the concept "one who travels for pleasure" is represented by "@12", its corresponding vocabulary is "tourist" in English and " 旅行者" in Chinese.

The main idea in Domain Ontology based CLIR is to use domain ontology as an intermediate language to maintain the concepts of bilingual domain ontology so that it remains consistent and the gap between language expression and representation of concepts is brought down. The below figure depicts the relationship between ontology and bilingual ontology.

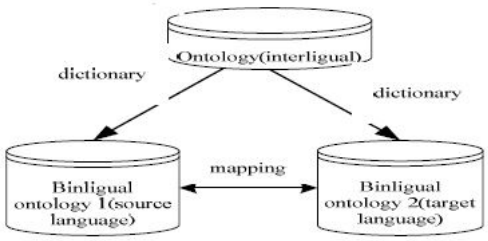

Figure 1: Relation between ontology and bilingual ontology

Every term in the documents and query is mapped to its corresponding concept of bilingual domain ontology. Let $q_E = \{t_1, t_2, ..., t_n\}$ be the terms in the query. Then for the mapping, a function S_E is defined for every term t_i in the query. So every query is expressed in the form of a concept set as $CS_E(t) = \{S_E(t_1), S_E(t_2), ..., S_E(t_n)\}$

There might be certain terms in the query or documents which might not be a part of the domain ontology. For such terms, dictionary is used for direct translation. Still some terms might be missing in the dictionary. Such terms might be translated manually and their mappings added to the domain ontology or skipped.

We can use a professional dictionary to translate and select first meaning as translation. By doing so, the translation results will not deviate too much for the words which are out of ontology because the system is based on specific domain for retrieval.

3.2 Concept similarity calculation

To compute similarity, *vector space model* is applied on semantic vector of user query and the document set. Similarity is computed among the vectors by considering two aspects: the *concept correlation value* in the document set and the *hierarchy structure similarity* in the ontology.

Concept correlation value Let f_{ij} be the frequency of occurrence of concept C_i in the document corpus. Now corresponding to each document the frequencies of occurrences constitute the vector of concept C_i as $\boldsymbol{w_i} = (f_{i1}, f_{i2}, ..., f_{ij}, ..., f_{in})$. The similarity between two concepts C_i and C_j can now be calculated as:

$$Sim_d(C_i, C_j) = \frac{\boldsymbol{w_i} \bullet \boldsymbol{w_j}}{|\boldsymbol{w_i}|.|\boldsymbol{w_j}|} = \frac{\sum_{k=1}^{n} f_{ik}.f_{jk}}{\sqrt{\sum_{k=1}^{n}(f_{ik})^2.\sum_{k=1}^{n}(f_{jk})^2}} \quad (1)$$

Hierarchy structure similarity

$$Sim_s(C_i, C_j) = \begin{cases} (1 - \frac{\alpha}{dep(R(C_i,C_j))+1}) \times \frac{\beta}{d(C_i,C_j} \times \frac{son(C_j)}{son(C_i)}, & d(C_i, C_j) \neq 0 \\ 1, & d(C_i, C_j) = 0 \end{cases} \quad (2)$$

Where $dep(R(C_i, C_j))$ is the nearest root concept depth between C_i and C_j, $d(C_i, C_j)$ is the distance between C_i and C_j, $son(C)$ is the number of sub-tree nodes for C in the ontology. α and β are tuned factors, we take $\alpha = \beta = 0.5$.

Now the combined similarity of concepts C_i and C_j is computed as:

$$Sim(C_i, C_j) = \delta.Sim_d(C_i, C_j) + (1 - \delta).Sim_s(C_i, C_j) \quad (3)$$

where $\delta \in [0, 1]$ is the *tuning factor*. We set $\delta = 0.7$.

3.3 Semantic query expansion

It is very difficult for users to convey the information that is relevant for them by specifying certain key words in query. To tackle this, domain ontology is used to understand the retrieval request and a concept hierarchy is used for logical inference.

User query: The initial set of concepts from user input. Let it be denoted as $Q = \{C_1, C_2, ..., C_i, ..., C_j, ..., C_m\} \forall i, j : 1 \leq i, j \leq m$, where C_i, C_j is user query keywords.

Super class set: $\{F_i\} = \{C_i | \forall C_i, C_j \in subclassof(C_i)\}$

Sub class set: $\{S_i\} = \{C_j | \forall C_i, C_j \in subclassof(C_i)\}$

Equivalence set: $\{E_i\} = \{C_j | \forall C_i \in F_i, C_j \in subclassof(C_i), C_i \neq C_j\}$

3.4 Weight calculation

$tf - idf$ weighting scheme is followed to compute the weight of each concept relative to each document.

Let D be web document set, $D = \{d_1, d_2, ..., d_n\}$
Q_c be set of query concepts, $Q_c = \{C_1, C_2, ..., C_m\}$
$\{C_i\}$ be the set of C_i and its synonyms
f_{ij} be the frequency in d_j of set $\{C_i\}$. Then,

$$w_{i,j} = log(f_{ij} + 1) \times log(n_i/n + 1) \quad (4)$$

where n_i is the number of documents related to $\{C_i\}$.

We similarly compute weights of $\{F_i\}$, $\{S_i\}$, $\{E_i\}$ and the synonyms $\forall C_i in Q_c$. The final weight, w_{id} which is the average weight of C_i and its expansions in D is:

$$w_{id} = w_{i,j} \times k_1 + w([F_1], D) \times k_2 + w([S_1], D) \times k_3 + w([E_1], D) \times k_4; k_1 + k_2 + k_3 + k_4 = 1 \quad (5)$$

where k_i's are tuning factors which represent the relative importance of $\{C_i\}, [F_i], [S_i], [E_i]$.

Now use these weights to form vectors for documents and queries as $d_j = (w_{1j}, w_{2j}, ..., w_{nj})$ where w_{ij} is the weight of ith concept in jth document and $Q_c = (w_{1c}, w_{2c}, ..., w_{nc})$. Using these, we compute similarity between a document and a query as:

$$f = \sum_{i=1}^{m} \frac{w_{i,j}.w_{i,c}}{\sqrt{\sum_{i=1}^{m}(w_{i,j})^2 . \sum_{i=1}^{m}(w_{i,c})^2}} \quad (6)$$

The query Q is classified into two parts; one with keywords included in ontology, Q_o and the other with keywords not in ontology, Q_k. The concepts and relations in Q_o are then mapped to ontology and all their synonyms are combined. Keywords in Q_k are mapped with document set. Finally the retrieval results are returned combining both the above mappings.

4 Architecture

To solve semantic loss and distortion problem while translating between query language and retrieval language, bilingual domain ontology created is used. Recall/Precision for keyword retrieval is low as they are not part of ontology and dictionary translation is used for them. Fig 2 indicates the design of the retrieval model.

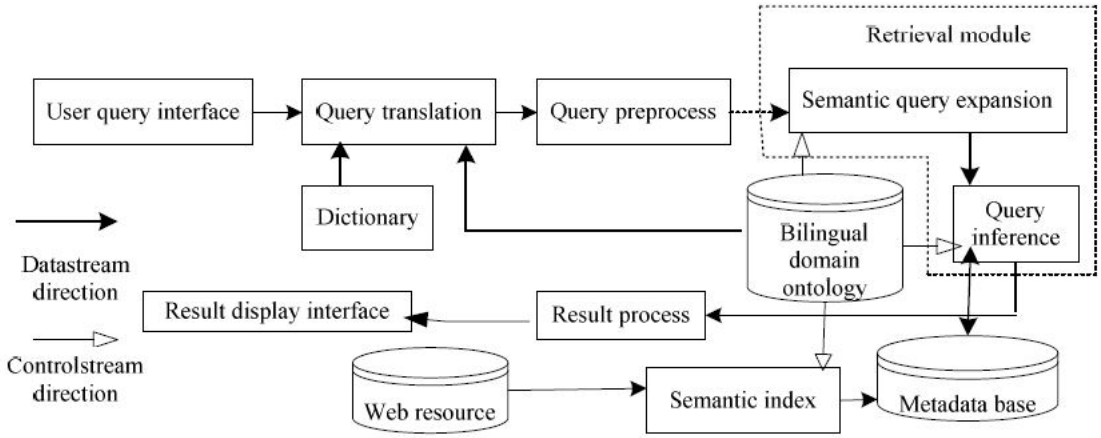

Figure 2: English-Chinese information retrieval model based on Domain Ontology

The model is mainly divided into three sections: bilingual ontology, metadata base and retrieval module. Each of these is described briefly:

Bilingual domain ontology replaces dictionary to translate query terms in this model. The concepts in the query are mapped to domain ontology.

Metadata base Sentences are segmented be removing any stop words. The documents are analyzed by referring to bilingual domain ontology and all the indexes are stored in index-base.

Retrieval module The index base is used to compute the weights of each concept in query and documents and vector space model is used to find the similarity of the query with document and returned to the users.

The steps followed in the model creation are described as follows:

1. Bilingual Domain Ontology is established by considering all the information resources.
2. Semantic metadata-base is created by crawling the web resource base.
3. The user query is translated through query interface and translation module. The translated concepts are sent to query preprocess module and then to information retrieval module
4. Information retrieval module uses domain ontology to execute the query and the results are returned.
5. The result process module then delivers results to the display interface which are then displayed to the user.

5 Experiment and result analysis

In the experiment a bilingual travel domain ontology is used. Document set comprises 500 Chinese documents from Sina website all of them belonging to travel field. The value of Precision, P and Recall, R are used as a measure of performance indication.

Three IR models were used for the analysis: IR_1 is the semantic retrieval using Chinese language as query, IR_2 is the retrieval model developed in this paper and IR_3 uses CLIR based dictionary.

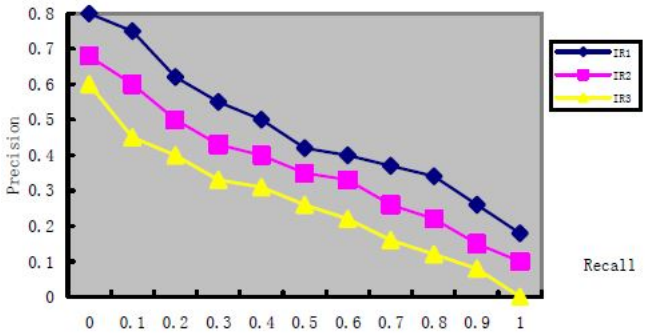

Figure 3: Precision-recall

IR_1's precision was not 100% because the domain ontology is not accurate enough. The performance of IR_2 is not as good as IR_1 because the corpus of translation is not big enough and some of the terms rely on dictionary translation only. When recall approached 1, the precision of semantic retrieval became greater than keywords based retrieval.

6 Conclusion

The ontology proved to be an important addition to the CLIR in web information retrieval. By incorporating domain semantics, the problem of low precision and recall is solved. This model can be significant for not only Chinese-English but any language translations be developing bilingual domain ontologies for the corresponding source-target language pair.

Even though the concept of ontology is noteworthy, the model is still not perfect because of its own limitations. The process of constructing bilingual ontology is difficult in practice because of the computational complexity and its need for the domain knowledge. Hence, there is a scope of further development of Semantic Web technology and reducing complexity of constructing the ontology. If this is achieved, one of the major drawbacks of the ontology based information retrieval will be untangled.

The amount of calculations needed to maintain the ontology for every new addition of concepts can also be a problem. The users can't query while these type of computations occur. So there has to be a way such that these computations can be done in real time and made the system ready for further queries. The managers of this system may periodically keep adding new concepts or modifying older ones and once in every month, these changes can be incorporated into the model.