Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 29 (2012) 3914 – 3923

Procedia Engineering

www.elsevier.com/locate/procedia

2012 International Workshop on Information and Electronics Engineering (IWIEE)

Using ART2 Neural Network and Bayesian Network for Automating the Ontology Constructing Process

Maryam Hourali[a], Gholam Ali Montazer[b,*]

[a,b]*IT Eng. Dept.,School of Engineering,Tarbiat Modares University, Tehran, Iran, P.O.Box:14115-179*

Abstract

Ontology is one of the fundamental cornerstones of the semantic Web. The pervasive use of ontologies in information sharing and knowledge management calls for efficient and effective approaches to ontology development. Ontology learning, which seeks to discover ontological knowledge from various forms of data automatically or semi-automatically, can overcome the bottleneck of ontology acquisition in ontology development.. In this article a novel automated method for ontology learning is proposed. First, domain-related documents were collected. Secondly, the C-value method was implemented for extracting meaningful terms from documents. Then, an ART neural network was used to cluster documents, and terms' weight was calculated by TF–IDF method in order to find candidate keyword for each cluster. Next, the Bayesian network and lexico-syntactic patterns were applied to construct the initial ontology. Finally, the proposed ontology was evaluated by expert's views and using the ontology for query expansion purpose. The primary results show that the proposed ontology learning method has higher precision than similar studies.

Keywords: Ontology; ART Neural Network; Term Frequency–Inverse Document Frequency (TF-IDF); C-value Method; Bayesian network; Lexico-Syntactic Patterns.

1. Introduction

Ontologies are defined as formal, explicit specifications of a shared conceptualization [1]. They are an essential component in many knowledge-intensive areas like the Semantic Web [2], knowledge management, and electronic commerce. The construction of domain ontologies relies on domain

* Corresponding author. Tel:+98 (21) 82883990,
E-mail address: montazer@modares.ac.ir.

doi:10.1016/j.proeng.2012.01.594

modellers and knowledge engineers, which are typically overwhelmed by the potential size, complexity and dynamicity of a specific domain [3]. In consequence, the definition of exhaustive domain ontologies is a barrier that very few projects can overcome. Due to these reasons, nowadays, there is a need of methods that can tackle, or at least ease, the construction of domain ontologies. Automated Ontology Learning methods allow a reduction in the time. Zhou, presented a classification of ontology learning techniques which is composed of three categories: statistics-based, rule-based, and hybrid techniques. [4]. Shamsfard, classified the ontology learning approach to statistical and symbolic. The symbolic approaches are the logical, linguistic-based and template-driven approaches. Heuristic methods may be used to facilitate each approach [2].

In this paper we combined different ontology learning methods such as: statistical, linguistic, and pattern-based for proposing a novel automated ontology learning system in query expansion domain. For this purpose, natural language processing methods such as: depletion of stop words, linguistic processing, statistical processing, etc was implemented on domain related documents set. Next, by using C-value method, we extracted main terms, and construct term-document matrix respectively. Then, we clustered this matrix with ART neural network. In order to choose a candidate for each cluster, we implemented the TF-IDF method and selected the term with the highest weight as the candidate. Furthermore, the Bayesian network was implemented for constructing ontology hierarchy. Then we used lexico syntactic patterns to extract non taxonomic relations and ontology instances. Hence, the initial ontology was constructed and evaluated by domain related experts. The remainder of this paper is organized as follow: section 2, describes the architecture of ontology learning system step by step accompanied by associated empirical results. In section 3, performance evaluation of the system is described. The paper makes conclusion and suggests directions for future research in section 4.

2. Ontology Construction

The constructing ontology process comprises four steps: (1) analyzing documents, (2) clustering documents, (3) learning ontology relations, and (4) Learning ontology instances. The proposed architecture of ontology learning system is shown in Fig.1, and the process is described in detail bellow:

2.1. *Analyzing Documents*

The first step to construct ontology in a certain domain such as information technology domain, is collecting the domain related documents. We deleted the duplicated articles and reached to 3345 articles as a document set for analysis. Secondly, C-value method in linguistic and statistical parts was used to extract main terms for ontology construction. The linguistic part consists of the part of speech (POS) tagging of the corpus, the linguistic filter, and the stop list. So, (Adj|Noun)+Noun filter was used to identify those terms that contain a noun or contain an adjective that accompanied with a noun, such as: 'applied computer science', 'Internet technology'. Our stop list included some high frequency words such as: great, numerous, several, year, just, good, etc.

To some up, 3700 terms with a considerable amount of C-value in range [1,3] were extracted (about 67 percent of total terms). For increasing the accuracy and depletion the noisy inputs, we considered those terms which were existed in 3 different and separate document [13]. Finally, 1220 main terms were selected in constructing the ontology. Samples of these terms are shown in Table 1. Extracted terms were located in a term-document matrix. In this matrix, rows indicate the document sets and columns indicate the extracted terms. Matrix's cells are normalized value of TF-IDF.

Table 1. Some Extracted Terms Based on C-Value Method

##Term	*C-value*
information technology	1746.92
information system	536.27
communication technology	315
software application	138.33
telecommunication	119.82
computer science	112.82

The proposed architecture of ontology learning system is shown in Fig.1.

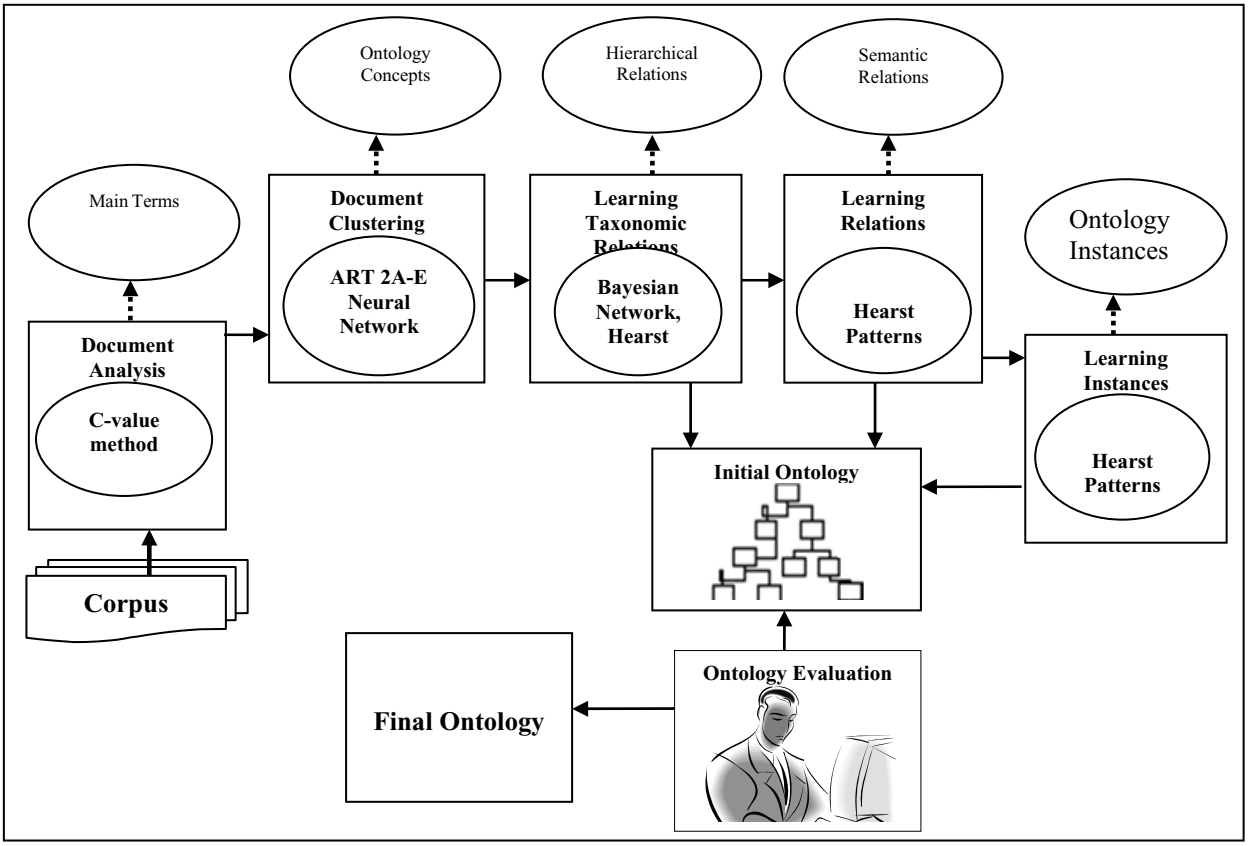

Fig.1. The Architecture of Ontology Learning System

The matrix dimension is too large (3345×1220), so it is not possible to show it completely. Table 2 represents a part of this matrix as an illustration.

Table 2. Some Part of TF-IDF Term-Document Matrix

Term / *Document*	*information technology*	*information system*	*communication technology*	*software application*	*telecommunication*
1	0.77	0.55	0.45	0.13	0.14
2	0	0	0.13	0.53	0.15
3	0	0	0	0	0
4	0.5	0.12	0	0	0
5	0	0	0.22	0.2	0.21
7	0.55	0.52	0	0	0
7	0	0	0.7	0.22	0.75
5	0	0	0	0	0
9	0	0	0	0	0
10	0	0.57	0	0	0

2.2. *Clustering Documents*

This phase seeks to cluster the documents and to find similar groups of documents. Hence, ART 2A-E neural network was implemented, and term-document matrix values were considered as the network input. In order to chose the best vigilance value, different thresholds from (0,1) were examined empirically. The vigilance 0.37 delivered the best quality (homogenous quantity of documents in clusters) of clustering. Using this vigilance value, 150 clusters have been identified. TF-IDF weight was calculated for each term in every cluster. Finally, a candidate for each cluster was selected according to the term with the highest TF-IDF value. The process of TF-IDF calculation for some terms of the first cluster is represented in Table 3. As it is shown in Table 3, the term "information system" has the highest value of TF-IDF, so this term is selected as a candidate to represent cluster one. Similarly, each group generated a representative term. By deleting the identical representative terms, and consolidating their related groups, 177 terms were obtained. Indeed, these terms are ontology concepts.

Table 3. TF-IDF Calculation for Some Terms of the First Cluster

Term	F_{ij}	**TF**	N_i	***IDF***	***TF−IDF***
Information System	479	1	350	3.27	3.27

Communication Technology	305	0.74	312	3.42	2.20
Electronic Government	55	0.12	50	7.07	0.73
Information Ecology	34	0.07	32	7.71	0.45

2.3. *Learning Ontology's Relations*

Two types of relation between ontology concepts are taxonomic and non-taxonomic relation. In the taxonomic relations, each class is introduced as a subclass of another class ("is-a" relations). Other relations except "is-a" relation is considered as non-taxonomic relation (Shamsfard, 2004). For example "part-of" relation belongs to non-taxonomic relations. In this article, we introduced learning of "is-a" and "part-of" relations. The details are described below:

2.3.1. Learning of "is-a" relation

We used the Bayesian network to construct the ontology. The system determines the order of inference based on weight of concepts automatically. Furthermore, we set a threshold to avoid the error of insertion. The following procedure was used for "is-a" relation extraction: Since, knowing the prior probability is essential for conditional probability calculation. So the prior probability 0.5 was presumed for all related calculations empirically. The concept "information technology" which was considered to construct its ontology was located in the top, and other concepts came under it. Then, concept's weights were calculated based on TF-IDF value. The concepts with the weight greater than threshold value were selected. Next, the related conditional probability of those concepts inferred to "information technology" concept was calculated. Those concepts with conditional probability greater than threshold value were inserted under "information technology" concept.

As a result, some concepts were located under "information technology" concept. Through repetition of this procedure, all concepts were located in the hierarchy. The threshold 0.73 delivered the best hierarchy structure, so this threshold value was selected. The results are depicted in Fig.2.

2.3.2. Learning of "Part-Of" Relation

Lexico-syntactic patterns as originally defined by Hearst [12] which have been used for learning "part-of" relations. Thus, this Hearst pattern was applied to the document set :
NP including {NP,}* {and | or} NP, NP contain {NP,}* {and | or} NP, NP encompass {NP,}* {and | or} NP, NP engulf {NP,}* {and | or} NP.

Exerting the above patterns in the text, have resulted some patterns . one of them is as bellow:
computer security including intrusion detection, encryption technology, and cryptography, The produced relations were added to the ontology and the related results are: intrusion detection part-of computer security, encryption technology part -of computer security and cryptography part -of computer security.
Other "part-of" relations have been extracted and located in the ontology respectively. These relations are depicted in Fig.2.

2.4. *Learning Ontology's Instances*

Instances are data records which are allocated to the concepts or relations [2] The Hearst patterns were used for learning the ontology instances. Hence, the following patterns were added to the document set in the ontology construction process.

$$Hearst1: \pi\left(c_{\sup}\right) such\ as \left(i|c_{sub}\right)$$

$$Hearst2: such\ \pi\left(c_{\sup}\right) as \left(i|c_{sub}\right)$$

$$Hearst3: \pi\left(c_{\sup}\right), (especially \mid including)\left(i|c_{sub}\right)$$

$$Hearst4: \left(i|c_{sub}\right)(and \mid or) other \pi\left(c_{\sup}\right)$$

In the patterns, the variable *i* standing for the name of an Instance and the variables *Csup* and *Csub* standing for the name of a concept from the given ontology. The plural of c is denoted by $\pi(c)$ that generated by adding 's' at the end of each word.

By implementing the Hearst patterns in the text, the bellow patterns were extracted:Object oriented languages such as Java, Python, and C++ (Hearst1), Such content provider language as XML (Hearst2), Radio frequency, specially VHF and UHF (HEARST3), Mouse, Keyboard and other computer input devices (HEARST4). As a result, the following instances are added to the ontology:
Java instance of object oriented languages, Python instance of object oriented languages, C++ instance of Object oriented languages, XML instance of content provider language, VHF instance of Radio frequency, UHF instance of Radio frequency, Mouse instance of computer input devices, Keyboard instance of computer input devices. Consequently other instances were extracted and added to the ontology which is shown in Fig.2.
In order of readability the figure, only the extracted ontology up to three levels is depicted in Fig.2.

3. Performance Evaluation of the System

For performance appraisal of system two approaches is considered. One of them is based on the expert's views and the other is based on its performance in query expansion application. In follow, details of this approaches is presented.

3.1. *Evaluation based on expert's views*

After producing the ontology, its precision must be evaluated. However, there was no another ontology to compare it with, so we invited domain experts to evaluate its precision. In order to estimate the precision of the system, we defined two kinds of precision evaluation methods, as follows:

Fig. 2 The Information Technology Ontology

Concept_precision demonstrates the precision of the keywords the system selects and concept_location_precision also demonstrates the precision of the location in the hierarchy relations. The formula of concept precision and concept location precision are listed below [14,15]:
A: Keywords (concepts) that the system generates and the expert has defined, B: Keywords (concepts) that the system generates but the expert has not defined.
C: Keywords (concepts) that the system generates and the expert defines whose locations are right, D: Keywords (concepts) that the system generates and the expert defines whose location is in error. The average precision(c_p) of domain experts is 0.886 (almost 89%), and the average precision(c_l_p) of domain experts is 0.834 (almost 85%).

3.2. *Performance Appraisal of the System Based on Query Expansion Application*

The performance appraisal of information retrieval system has been considered for tow situations: lack of ontology and using crisp ontology expansion. For this purpose, the query expansion approach is implemented. The query expansion algorithm based on ontology is explained, and the information retrieval results are presented.

3.2.1.The Proposed Query Expansion Algorithm Based on the Ontology

In this algorithm it is assumed that entered user query terms are existed in the Information Technology ontology. If the query terms were not in the related ontology of a specific domain, the most common approach is adding synonym terms based on a general dictionary (commonly English comprehensive ontology WordNet). General dictionaries do not consider any specific domain, so a satisfied accuracy will not be obtained. Therefore, in the proposed query expansion algorithm, situations are considered that the entered query terms exist in the Information Technology ontology. Different approaches are used for query expansion based on the existence or the lack of relations between query terms in the ontology. These approaches are explained in the following:

i. Query terms exist in the Information Technology ontology and have semantic relation:

The associated relations of terms in an ontology are extracted (it is possible that one term is father or child of another term). For each father term, all its father terms (generalized terms) are extracted, and for each child term, all its child terms (specialized terms) are extracted as well and then these extracted terms are entered in the expanded query, consequently. As an illustration, consider query = (“storage system”, “network storage”). This terms have “storage system” > “network storage” relation in the ontology; thus, instead of term “storage system” generalized terms such as “computer hardware” will be entered in the expanded query. Similarly, instead of term “network storage”, specialized terms such as “storage area network” are added in the expanded query.

ii. Query terms exist in the Information Technology ontology, and have not semantic relations.

In this situation, all related father and child terms will be entered in the expanded query. As a case in point, consider query = (“readiness assessment”, “smart card”). Because there is no relation between these query terms in the ontology, all of their father and child terms are added in the expanded query; so,

instead of term “readiness assessment”, the term “information technology application” and instead of term “smart card” the terms “e-banking service” will be added in expanded query.

3.2.2.Performance Evaluation of Information Retrieval System

The two most frequent and basic measures for information retrieval effectiveness are precision and recall [13, 14 ,15]. So, we used these measures for the ontology performance evaluation. Precision and recall are defined in terms of a set of retrieved documents (e.g. the list of documents produced by a web search engine for a query) and a set of relevant documents (e.g. the list of all documents on the internet that are relevant for a certain topic).

The average precision plots at each standard recall level across all queries and evaluates overall system performance on a document/query corpus [16,17].

For performance evaluation of information retrieval system in local searches, the Google Desktop search engine has been implemented, and a lot of related documents to the Information Technology domain have been entered in. Then both unique and multiple query terms were entered in the search engine.

$$Precision = \frac{\#(relevant\ items\ retrieved)}{\#(retrieved\ items)} = P(relevent|retrieved) \tag{4}$$

$$Recall = \frac{\#(relevant\ items\ retrieved)}{\#(relevant\ items)} = P(retrieved|relevent) \tag{5}$$

In Fig.3, the average precision for different level of recall values has been depicted for the two situations: lack of ontology(no expansion) and ontology query. These situations are selected to evaluate the ontology performance rather than lack of ontology. As Fig.3.shows the ontology based query expansion has better average precision in retrieving relevant documents.

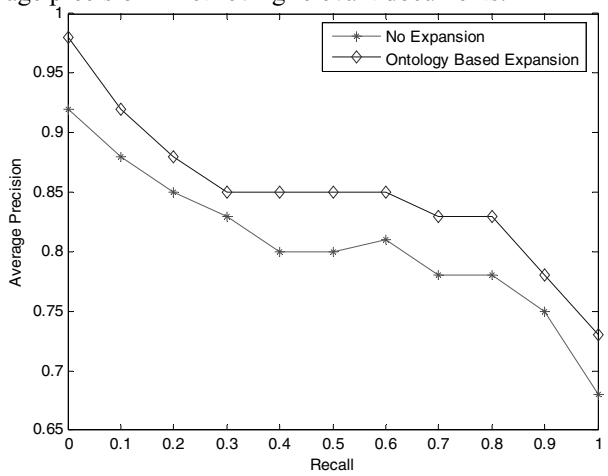

Fig.3. The average precision plot for two situations: lack of ontology and ontology query expansion

4. Conclusion and future work

This paper proposed a novel method that automates construction of the ontology by integrating C-value method, ART neural network, TF-IDF weighting method, Bayesian network, and pattern based methods. C-value method was used for key terms extraction, ART neural network was applied for clustering of documents and ontology concepts clustering. The TF-IDF method was implemented for selecting the associated concept of the clusters which had been extracted by neural network. The ontology evaluation process, was done based on domain experts views and applying the ontology in the query

expansion domain. The average concept precision and average concept location precision of domain experts is gathered 0.887 and 0.852 respectively. Furthermore, average precision of the ontology based query expansion was higher than keyword base search. In the future, we will propose a method for fuzzifying ontology and will implement it in various applications and domains.

Acknowledgment

This research has been partially supported by Iran Telecommunication Research Center (Contract No. 20127/500) and the authors have to appreciate its supportive role gratefully.

References

[1] Zhou L,Ontology learning: state of the art and open issues. *Inf Technol Manage*, 2007, 8, 241–252.

[2] Shamsfard M, and Abdollahzadeh Barforoush A, The state of the art in ontology learning: a framework for comparison. The Knowledge Engineering Review, 2003, 18, 293–316,.

[3] Gruber T.R, A translation approach to portable ontologies. Knowledge Acquisition, 1993, 5(2), 199–220.

[4] Manning C, and Schu¨tze H, Foundations of Statistical Natural Language Processing. MIT Press, Cambridge, MA. 1999.

[5] Kang S, Automatic classification of WWW documents using a neural network. International Conference on Production Research, Thailand, 2000.

[6] Pham D.T, and Sukkar M.F, A predictor based on adaptive resonance theory. Artificial Intelligence in Engineering, 1998, 12, 219-228.

[7] Frank T , Comparative Analysis of Fuzzy ART and ART-2A Network Clustering Performance", IEEE Transactions On Neural Networks, 1998, 9(3).

[8] Cimiano P, Ontology Learning and Population from Text Algorithms, Evaluation and Applications. Springer, Germany, 2006..

[9] Frantzi K, Automatic recognition of multi- word terms. International Journal of Digital Libraries, 2000, 3(2), 115-130.

[10] Shamsfard M, Abdollahzadeh Barforoush A, Learning ontologies from natural language texts. Human-Computer Studies, 2004, 60, 17-63.

[11] Mihalcea R, Introduction to Information Retrieval. CSCE 5200 Information Retrieval and Web Search. North Texas, 2009.

[12] Hearst M, Automatic acquisition of hyponyms from large text corpora. Proceedings of the 14th International Conference on Computational Linguistics (COLING), Nantes, France, 1992, 539-545.

[13] Neshati M, Ontology and Taxonomy Extraction from Text Documents. Master of Science Thesis in Software Engineering, Sharif University of Technology Computer Engineering Department, Tehran, Iran, 2007.

[14] Teimourpour B, Sepehri M.M, and Pezeahk L, A new method for intelligent categorization of scientific texts (case of Iran's nanotechnology papers), Journal of Science and Technology Policy, 2009, 2(2) 5–16.

[15] Farough A.M, and Loghman, L.S, Evaluation of procedures and methods of university researches commercialization; the case of Tabriz University, Journal of Science and Technology Policy, 2011, 3(4), 15–29.

[16] Tabatabaeian H, and Entezari, M, Institution mapping of power industry as an example, Journal of Science and Technology Policy, 2008, 1(1) 53–64.

[17] Bandarian, R, Measuring Commercial Potential of Technology with fuzzy Logic, Journal of Science and Technology Policy, 2008, 1(1), 15–32.