Classification of Web Documents Using Concept Extraction from Ontologies

Marina Litvak, Mark Last, and Slava Kisilevich

Department of Information Systems Engineering, Ben-Gurion University of the Negev,Beer-Sheva 84105, Israel
{litvakm,mlast,slaks}@bgu.ac.il

Abstract. In this paper, we deal with the problem of analyzing and classifying web documents in a given domain by information filtering agents. We present the ontology-based web content mining methodology that contains such main stages as creation of ontology for the specified domain, collecting a training set of labeled documents, building a classification model in this domain using the constructed ontology and a classification algorithm, and classification of new documents by information agents via the induced model. We evaluated the proposed methodology in two specific domains: the chemical domain (web pages containing information about production of certain chemicals), and Yahoo! collection of web news documents divided into several categories. Our system receives as input the domain-specific ontology, and a set of categorized web documents, and then perfroms concept generalization on these documents. We use a key-phrase extractor with integrated ontology parser for creating a database from input documents and use it as a training set for the classification algorithm. The system classification accuracy is estimated using various levels of ontology.

1 Introduction

To meet our information needs today, we need more intelligent agent systems to gather the useful information from the huge amount of data sources available on the Web. Web Content Mining uses the ideas and principles of data mining to screen the web data. One of the problems in the web content mining area is to represent web documents as a meaningful, informative input for data mining algorithms, and then to interpret the mining results in a meaningful and useful way.

Over last couple of years, a new, promising area of research has emerged in web content mining - the usage of *domain ontology*[1]. In this paper, we introduce an ontology-based web content mining application for analyzing and classifying web documents in a given domain. The proposed classification methodology can be used by intelligent information agents for retrieving the relevant documents

[1] According to the most cited definition in the literature ([4]), ontology is an explicit specification of a domain conceptualization. It denotes and organizes entities that do exist in a domain of interest using a formal declarative language. It accumulates and organizes knowledge in a machine-processable and human-readable way.

V. Gorodetsky et al. (Eds.): AIS-ADM 2007, LNAI 4476, pp. 287–292, 2007.

from the Web. We use *domain ontology*, which organizes concepts, relations and instances into a domain [5], for the purpose of enriching the term vectors representing documents with concepts. This approach has two benefits: first, it resolves synonyms; and second, it introduces more general concepts.

One of the first prototypes of an ontology-based system for information retrieval on the web was introduced by authors of [2]. The domain ontology was characterized by a set of relevant concepts and relationships between them. The global relevance grade of a given page was computed as a combination of a syntactic grade (based on page ranking by a search engine), a semantic-syntactic grade (based on the presence of domain-related words), and a semantic grade (based on the domain-specific semantic network). In another publication ([6]) Hotho et al. use ontologies to improve text document clustering.

Contrary to these papers, we deal here with a classification task. Our system builds the decision model after training on a set of documents introduced by a domain expert. Instead of building a model based on some pre-specified words, our model uses an ontology built for a specific domain. We use some of the strategies proposed in [6] to improve web document classification.

The rest of this paper is organized in the following way. Section 2 describes the methodology and Ontology-based Phrase Extractor. In Section 3, we present the results of initial experiments. Finally, in the last section we outline the conclusions and the future work.

2 Methodology

The goal of our system is to build a compact model (profile) of the pages collected from the web so that new unlabeled pages can be reliably categorized later on. To generate a collection of training documents for a given domain, a user can initiate the system operation by submitting a set of domain-related keyword queries to a search engine (such as Google^{TM}). A domain expert reads the retrieved documents and labels them as belonging to a specific category based on their content. We induce a classification model from a training collection that includes a mix of labeled pages from multiple categories. Each page is represented as a vector of $< term_i,\ weight_i >$ pairs, received from Ontology-based Phrase Extractor module, described in the sub-section below. The phrases are extracted using a list of domain-specific terms and ontology information. The term-frequency (*tf*) $weight_i$ indicates the frequency of a $term_i$ in the observed document.

2.1 Ontology Specification

In this paper, an ontology represents the conceptual information describing the domain of interest (see Section 3) by hierarchy of domain concepts with multiple inheritance. We use such ontology for the purpose of conceptual document representation, extraction of more meaningful and relevant (even abstract) information from text of documents, and, as a result, building more accurate classification models. Of course, in case of some ontology updates, the system should be retrained.

WordNet^{TM} is one of the famous examples of ontology widely used for experimental evaluations. Although not explicitly designed as an ontology, WordNet [8] largely fits into the ontology definitions given above. The WordNet database organizes simple words and multi-word expressions of different syntactic categories into the so called synonym sets (synsets), each of which represents an underlying concept and links these through semantic relations. The current version of WordNet comprises a total of 115,424 synsets and 144,309 lexical index terms. Its hierarchical structure is not necessarily a tree structure. It may also be a directed acyclic graph possibly linking concepts to multiple superconcepts at the same time.

2.2 Ontology-Based Phrase Extractor

The Ontology-based Phrase Extractor receives as input web documents, a domain ontology and a user-specified abstraction level (k) and creates concept vectors. At the first stage, the Extractor prepares phrase collection – XML file including all general thing names (instances in ontology) as phrases with their associated classes at the k^{th} level of hierarchy (bottom-up) as related concepts. In case of abstraction level equal to 0, the collection does not include any related concepts.

Note, that fusing of instances from ontology (as phrases) during the phrase collection can be replaced or complemented by extraction of significant phrases from text of documents, that is beneficial in case of general, not domain-specific ontologies like WordNet. Therefore, when we are working with WordNet, the phrase collection contains significant phrases with their associated super-concepts of ontology. For the purpose of extracting the associated super-concepts from WordNet, we utilize the disambigation function $dis(t)$ (see [6]) that returns the semantically closest concept for the term t based on the context of document. Since this function presents some type of *semantic distance measure* (see [1]), the depth of hierarchy can be ignored.

The Phrase Extractor scans the phrases included in the collection, and every time it finds the name of a thing (in a domain-specific ontology) or a significant noun (in case of WordNet) it refers to the related concepts. **Add Concepts** ("**add**") strategy ([6]) extends each term vector t_d by new entries for Wordnet concepts c appearing in the document set, while the **Replace Terms by Concepts** strategy expels all terms from the vector representations for which at least one corresponding concept exists.

For the purpose of disambiguation we also borrow from [6] two strategies: **All Concepts** ("**all**") (used in the chemical domain) and **Disambiguation by Context** ("**context**") (used in Yahoo! collections). **All Concepts** strategy does not do anything about disambigation and considers all concepts for augmenting the text document representation, while the **Disambiguation by Context** chooses the most appropriate concept by the document context using the disambiguation function (see *dis* function in [6]).

Note that different methods should be used for different datasets and ontologies while trying to adapt the methodology to their specific nature. For example, in

the chemical ontology, the synsets are disjoint, while the assignment of terms to concepts in WordNet is ambiguous. Therefore, when we are using WordNet, we should perform the word sense disambiguation for the purpose of prevention of extra noise in the dataset resulting from extending or replacing terms by concepts. For the same reason, extending the terms by the concepts instead of replacing them in the concept vectors is much more appropriate for an ambiguous ontology.

3 Experiments

The main goal of this research is increasing the classification accuracy of an information agent through maintaining an ontology. We tested our system on two domains: chemical domain, and Yahoo!^{TM} benchmark collection of web documents, called F-series. The experiments in the chemical domain have shown that most classification algorithms, except the Naive Bayes, are significantly improved by the synonyms handling. Conceptualization had an opposite effect on the same algorithms. The full description of these experiments as well as depiction of a domain-specific ontology, created for performing the experments are contained in [7]. For the F-series we used WordNet ontology [8].

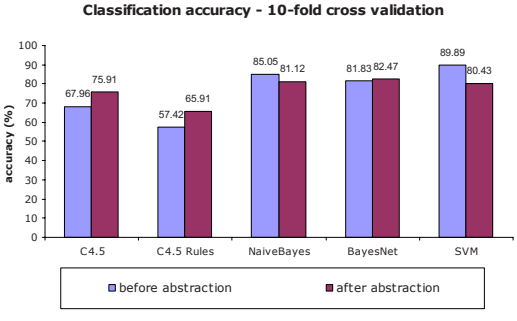

Fig. 1. Classification Accuracy with/without Abstraction – F-Series

The F-series contains 98 HTML documents belonging to one of four major category areas: manufacturing, labor, business & finance and electronic communication & networking. The results of experiments are presented in Fig. 1. The results of the *t-test* are presented in Table 1. As we can see, accuracy of decision trees models was improved after conceptualization, while NaiveBayes and SVM, conversely, got worse results than on usual term vectors.

We can explain such conflicting behaviour of these algorithms by their characteristics. During the abstraction (concept vectors building) we extend the term vectors by concepts. The relevance of these concepts is dependent on the disambiguation method. "Bad" methods insert a lot of noise to the data. Decision trees handle this problem by using the feature selection procedure that filters out the irrelevant features while building the model. NaiveBayes and SVM, contrarily,

Table 1. Results of t-test – F-Series

C4.5	C4.5 Rules	Naive Bayes	Bayes Network	SVM
↑ 0.0009*	↑ 0.0012*	↓ 0.0232*	↑ 0.6012	↓ 0.0001*

consider all features, that in case of noise, distort the results. Based on some current publications (see [9]) and our experience we can conclude that using the WordNet is not sufficient to reliably disambiguate word senses in text.

In our opinion, one of the promising approaches to solution of this problem is utilization of *domain-specific* ontology. Today, there are hundreds of ontologies available from the Internet, that usually cover very specific domain areas. A researcher can find something suitable for the processed domain, update it, merge several ontologies or build a new one based on domain knowledge as we did for the chemical domain. We need the knowledge of a qualified domain expert to express the domain very accurately via relations among instances and classes. A dictionary of key phrases for different categories of a given domain has proved to be useful too.

4 Conclusions and Future Work

In this paper we presented a new ontology-based methodology for automated classification of web documents by an information agent. The main contribution of this work is using domain-based ontology in the conceptual representation of documents. In contrast to the results on the chemicals domain reported in [7], where the synonyms and the taxonomic relationships were handled, the results received on the Yahoo! collections did not demonstrate such significant improvements. We explain such results by an insufficiency of the WordNet ontology to disambiguate word senses and generalize text representation properly as well as the experimental data specifications. According to different experimental results we can expect that the textual data with extensive number of synonyms (like documents describing the chemicals or food, etc.) will produce the best accuracy.

We intend to enhance our methodology via utilizing some tools (like GATE [3]) for automatic construction of ontologies. Such update will enable us to make our system completely domain-independent. We can also extend an existing ontology by several ontologies for the same set of documents. Also, we would like to use hyperlinks as well as the text from the linked pages to improve the classification accuracy.

References

1. Budanitsky, A., & Hirst, G.: Semantic Distance in WordNet: An Experimental, Application-oriented Evaluation of Five Measures. Workshop on WordNet and Other Lexical Resources, in the North American Chapter of the Association for Computational Linguistics (NAACL-2000), Pittsburgh, PA

2. Cesarano, C., d'Acerno, A., & Picariello, A.: An Intelligent Search Agent System for Semantic Information Retrieval on the Internet. Proc. of the Fifth ACM International Workshop on the Web Information and Data Management, New Orleans, Louisiana, USA (2003) 111–117
3. GATE - General Architecture for Text Engineering, The Natural Language Processing Research Group, Department of Computer Science, University of Sheffield [http://gate.ac.uk/]
4. Gruber, T. R.: A translation approach to portable ontologies. Knowledge Acquisition, 5 (2) (1993) 199–220
5. van Heijst, G., Schreiber, A.Th., & Wielinga, B.J.: Using explicit ontologies in KBs development. IJHCS (1997) 183–291
6. Hotho, A., Staab, S., & Stumme, G.: Ontologies Improve Text Document Clustering. In Proc. of ICDM-03 (2003)
7. Litvak, M., Last, M., & Kisilevich, S.: Improving Classification of Multi-Lingual Web Documents using Domain Ontologies. ECML/PKDD-2005, October, 2005, Porto, Portugal
8. Miller, G.A., Beckwith, R.T., Fellbaum, C.D., Gross, D., & Miller, K.: Wordnet: An Online Lexical Database. International Journal of Lexicography, 3 (4) (1990) 235–244
9. Voorhees, E.: Using WordNet^{TM} to disambiguate word senses for text retrieval. Proc. of the 16th annual international ACM SIGIR conference, Pittsburgh, PA, (1993)
10. Witten, I.H., Frank, E., Trigg, L., Hall, M., Holmes, G., Cunningham, S.J.: Weka: Practical machine learning tools and techniques with java implementations. In Proc. of ICONIP/ANZIIS/ANNES'99 Int. Workshop on Emerging Knowledge Engineering and Connectionist-Based Info. Systems (1999) 192–196
11. Yao, Y.Y., Zhong, N., Liu, J., & Ohsuga, S.: Web Intelligence (WI): research challenges and trends in the new information age. In Zhong et al., eds., Web Intelligence: research and development, LNAI 2198, Springer-Verlag (2001) 1–17
12. Zhong, N., Liu, J., Yao, Y. Y.: In search of the wisdom Web. IEEE Computer, 35 (11) (2002) 27–31