International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:2, No:9, 2008

A New Approach to Annotate the Text's of the Websites and Documents with a Quite Comprehensive Knowledge Base

Mohammad Yasrebi, Mehran Mohsenzadeh, and Mashalla Abbasi-Dezfuli

***Abstract*—**Machine-understandable data when strongly interlinked constitutes the basis for the SemanticWeb. Annotating web documents is one of the major techniques for creating metadata on the Web. Annotating websites defines the containing data in a form which is suitable for interpretation by machines. In this paper, we present a new approach to annotate websites and documents by promoting the abstraction level of the annotation process to a conceptual level. By this means, we hope to solve some of the problems of the current annotation solutions.

***Keywords*—**Knowledge base, ontology, semantic annotation, semantic web.

I. Introduction

SEMANTIC annotation is the process of inserting tags in a document to assign semantics to text fragments allowing creating the documents processable not only by humans but also automated agents [8]. The acquisition of masses of metadata for the web content would allow various Semantic Web applications to emerge and gain wide acceptance. At present there are various Information Extraction (IE) technologies available that allow recognition of named entities within the text, and even the relations, events, and scenarios in which they take part. Thus, metadata could be assigned to the document, presenting part of its information content, suitable for further processing. Such metadata can range from formal reference to the author of the document, to annotations of all the companies and amounts of money referred in the text [13].

The approach for automatic (versus manual) extraction of metadata is promising scalable, cheap, author-independent and (potentially) user-specific enrichment of the web content. However, at present there is no technology available to provide automatic semantic annotation in conceptually clear, intuitive, scalable, and accurate enough fashion. All existing semantic annotation systems rely on human intervention at hole or some point in the annotation process, therefore, the annotation process is manual or semi-automatic. In this paper, we present a new approach to semantic enrichment (annotate) websites and documents by taking the annotation process to a conceptual level and by integrating it into an existing knowledge base "WordNet". This approach is semi-automatic system.

By researching about methods and existing semantic annotation platforms we observe that all of these methods are using the source of information which is named knowledge base to define the concepts and semantics of words in texts. The knowledge bases which are used in these tools are defective and unable to define the concepts of some words. So, the idea of using extended knowledge base with more knowledge and information in most domains came to exist and is able to be complete more and more. In our developed approach there is no need for manual information extraction. It is not based on learning human-created samples either. The idea of information extraction lies in the concept of knowledge base, including a complete set of words, the collections of grammars, data frames and various lists of entities.

So, first of all, we discuss about the considered knowledge bases and then the function of our approach.

This paper is structured as follows. Section II discuss about the existing knowledge bases in our approach. Sections III and IV define the architecture of knowledge bases and present the model of our semantic annotation system and define the different stages in the annotation process. Section V including evaluation and conclusions are drawn in section VI.

II. The Role of Knowledge bases in our Approach

In this approach, two different knowledge bases used as follow:

- Primary knowledge base
- Secondary knowledge base

A. Primary Knowledge Base

The Primary knowledge base is the most important and essential part of our knowledge base. In fact, this knowledge base contains information about the concept/instance which is supplied by well-informed users. The primary knowledge base contains the set of data bases which are related to specific domain such as medicine, chemistry, physics, geographic, etc. Each data base includes words which are extracted from previous web pages and documents together with their concepts. As a word can convey different concepts in different domains, it may exist in two or more data bases. For example, the word "water" in chemistry means binary compound

M. Yasrebi is with the Islamic Azad University, Shiraz, Iran (phone: +98917-714-0793; e-mail: mohammadyasrebi@gmail.com).

M. Mohsenzadeh, is with the Islamic Azad University – Science and research branch, Tehran, Iran (e-mail: m_mohsenzadeh77@yahoo.com).

M. Abbasi-Dezfuli is with the Islamic Azad University – Science and research branch, Ahwaz, Iran (e-mail: abbasi_masha@yahoo.com).

(H2O), but in physics is in the category of liquids. Therefore, we have to have a data base in each domain for these words.

These data bases (the parts of the primary knowledge base) are going to become complete as the time passes, and in an ideal situation all words of a specific domain are identified and implemented in the database. Another solution is having one general data base for all domains instead of a data base for each domain, but in this data base we consider different fields for different domains.

B. Secondary Knowledge Base

As its name implies, the secondary knowledge base is used to help the primary knowledge base. The latter includes three components as follow:

- basic knowledge source
- data frame library
- lexicons

1. Basic Knowledge Source

Basic knowledge source (BKS) is the first part of the secondary knowledge base. Like the virtual world, BKS contains the identified words of all concepts and extracted words in web documents and source information, subset of the words of this knowledge base. Thus, BKS is a general knowledge base and it is not designed for specific areas.

BKS contains semantic relations plus concepts and a set of instances data. These relations demonstrate the relations between concepts and existing words with in BKS.

In general, BKS has some attributes as follow:

- accessibility
- generality
- richness of relations between concepts

WordNet Ontology [12] completely covers three above attributes. However, we can not use only this ontology in order to perform the extraction and induction of data in its data bases and extracted semantic schemas. Because it is defective for some words, and we reduce these defects with other parts such as data frame library and lexicons. For example, the WordNet Ontology can not identify the word "alen" as a person's name, or "222-2222" as a telephone number, or "qwerty@yahoo.com" as an e-mail address, etc. Since WordNet basically consists of information about concepts and their relations (e.g. hyperonyms etc.) YAGO[1] could be considered as additional BKS, since this ontology incorporates a lot of instanceOf(instance, concept) relations with broad coverage.

2. Data Frame Library

Basically in computer-based sciences, data has poor structure and for describing these data we have to use simple classifications such as "integer", "real", "string", etc. On the other hand, we can not identify concepts with these classifications. Therefore, we have to use a classification with better structure. This classification is presented as data frame library and contains the second part of our secondary knowledge base. One of the ways to extract the concepts such as date, e-mail address, phone number, etc. is to use the regular expressions [11]. It is important to pay attention that these regular expressions are used to limit the concepts in ontology in addition to identify the concepts. In this paper, we name these regular expressions as data frame library such as the regexes in C# language for recognizing an e-mail address, telephone number, IP address, etc.

Also, the data frame might have other application. For example if we have a string as follows in a text: "Address: Shiraz – Eram St. – No. 120"

We have to consider a regular expression which can recognize "Shiraz-Eram St.-No.120" as an address. Thus, in this case we consider the "key/value" regular expression to recognize these concepts, as shown in Fig. 1[2].

```
Instance = " Address : Shiraz – Eram St. – No. 120";
Pattern  = @"(\w+)\s*:\s*(.*)\s*$";
```

Fig. 1 The example of data frame for recognizing strings that contain key/value

3. Lexicons

The other part of our secondary knowledge base is lexicons. Lexicons used to enrich WordNet ontology as BKS. Different sources exists for integrating these lexicons such as World Wide Web (www) and the "Hyponyms" relation in WordNet ontology. According to this discussion we can have the complete list of the name of persons, animals, capitals, etc. However, the lexicon plays an important role for recognizing the instances of the specific concepts and limiting the domain. For example, the WordNet can not identify the concept of the word "alen", but this word exists in the list of the person's name in lexicons and then lexicons can detect this word as the name of person.

III. Architecture of Knowledge Bases

Fig. 2 shows our knowledge bases architecture briefly. As it is shown, this architecture contains all the knowledge bases which are described in previous sections and their relations.

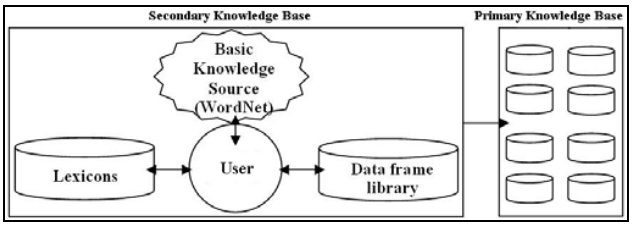

Fig. 2 The architecture of knowledge bases

In this architecture, the primary knowledge base recognizes the concept of extracted word in the inspected domain. If that word does not exit in the related data base (inspected domain), WordNet ontology as BKS recognizes the concepts. In cases where the WordNet Ontology is not able to identify some

[1] http://www.mpi-inf.mpg.de/~suchanek/downloads/yago/

[2] http://www.c-sharpcorner.com

concepts, data frame library and lexicons will help the WordNet ontology to recognize the unknown concepts.

As shown in the Fig. 2, all components of the secondary knowledge base are available to the competent user. The user familiar with the domain removes the probable inconsistency among concept titles in basic knowledge base, lexicon, and data frame library. The user is also there to identify the word concept if there is no help from any knowledge base component.(If the different parts of the secondary knowledge base have the different outputs for one word, the user can eliminate the inconsistency of these concepts and select the main concept of the current word.) Finally, once the concept is identified by one of the system components or user, it is inserted in the domain database of the main database, and the main knowledge base would be updated then.

For example, if the expression "127.0.0.1" is extracted as one word and the primary knowledge base could not identify a concept, WordNet as BKS will help the primary knowledge base and will search this word in its data base, but WordNet is not able to identify the concept. Thus, through detecting the word, the data frame library will recognize that this word is an IP address. Besides, it identifies the concept, and will be inserted into primary knowledge base. From now on, if there is an expression such as "127.0.0.1", the primary knowledge base will identify it as an IP address. It shows that the primary knowledge base is getting more complete.

The worst case occurs when no knowledge bases can recognize the word's concept. For example, if "aajbc" is the abbreviation of a company's name, in this special case the user who knows the domain will help the knowledge base and inserts its concept.

Let us review some advantages of our suggested approach:

1. Employing a highly appropriate knowledge base of concepts related to instances and entities existing in the text.
2. Allowing user to remove the possible inconsistencies among knowledge base components.
3. Allowing user to decide on the appropriateness of the concept selected for the relevant instance.
4. Working on automatic procedure as much as possible.

IV. The Annotation Method in our Approach

After preparing the needed knowledge base, based on the methods outlined in previous sections, we can discuss on extracting the word and the concepts and also semantic annotation.

First, it is necessary to describe a general view on the architecture of our approach and then inspect the details of this project. Fig. 3 shows a general view of the architecture of our approach.

As Fig. 3 shows, this process contains 3 separate phases:

1. Determining the text's domain
2. Extracting the words and their concepts
3. Semantic annotation and inserting tag process

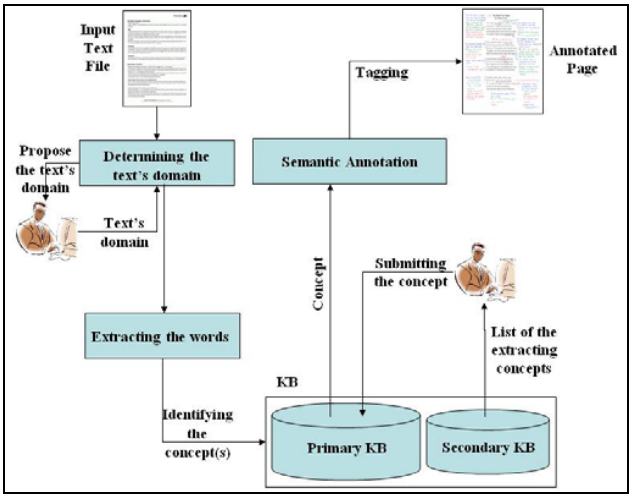

Fig. 3 Architecture of our approach

1. *Determining the Text's Domain*

As we have seen in Fig. 3, since the input file of the system is a text, we have to request the subject and the text's domain from the user who knows the domain.

This process can be done as an offer. In other words, the various domains are suggested to the user and then he will select one of them or may insert the domain manually.

2. *Extracting the Words and their Concepts*

In this phase, we need to extract words which are concepts or instances of a concept, and also explain a special meaning such as: email address, or name of person, etc.

Thus, by using a pattern which determines the words and a loop, we extract the words of the text one by one to the end of the text. So, after analyzing the text to words, we have to send the word one by one to knowledge base for determining their concepts.

At first, we send the word to the primary knowledge base and the primary knowledge base by identifying the determined text's domain will search the word in the data base which contains the words related to the domain. If the word exists in the inspected data base, the concept will be returned; otherwise, the secondary knowledge base will help the primary knowledge base and determine its concept. The first choice for determining current word is the WordNet as BKS. In this part, we have to inspect the word as a noun, verb, adjective or adverb. If the word is a noun the concepts will be extracted. So, we can get count of senses which are related to current word in WordNet. Just three modes may occur:

1. No sense exists for being noun.
2. Existing sense(s) for being noun and also other types(verb, adjective, adverb)
3. Existing sense(s) just for noun and no sense for other types.

For the first mode, we do not have to inspect the current word and then extract the concepts for this word, because the current word is not a noun at all. For the second mode, we have to compare the count of sense(s) related to the noun with

the other sense(s) which are related to the each type such as verb, adjective or adverb. If the counts of the sense(s) which are related to the noun are more than the other types, it is obvious that this word can be a noun. Otherwise, we do not have to inspect the current word and then extract the concepts for this word. For the third mode, it is obvious that the current word is certainly a noun and we have to extract its concepts. After we recognized that the word is a noun, we search the concepts in WordNet. A list of the extracted concepts is shown to the user and the user will choose the related concept of the word from the list, or if the user's concept is not in the list, he has to insert it manually.

After the user submits this process that word will be inserted with its concept into the data base which is related to the text's domain, and as a result the primary knowledge base is updated and completed more and more.

The above cases happen when WordNet can identify the concept of the word, otherwise, data frame library or lexicons will help the WordNet.

If the word is the same as the one of the existing patterns (regular expressions) in data frame library, the concept is determined. For example, it specifies that this word is an email address, or a phone number, or IP address, etc. Otherwise we have to search in different lists of lexicons and if the same case is found the concept will be determined. For example, it specifies that this word shows a person's name. If all of these knowledge bases could not find the concept(s) of this word, the user who knows the text's domain has to insert the concept manually. After determining the concept of the current word, we have to go to the next word and we continue this process to the end of the text. To prevent doing this process twice for the words which are repeated more than once, we recognize these repeated words, and the process of extracting the concept for these words just operates once.

3. Semantic Annotation and Inserting Tag Process

In this last phase, the extracted words in the text with their concept are accessible. Thus, by identifying the location of the words in the text, we insert and add tags which contain the concept of the words into the text. For example if the word "water" is appeared in the text and its domain is chemical, this tag "Binary_Compound" will be added to the text as follows:

<Binary_Compound> water </Binary_Compound>

At the end of this phase, the first text that is considered as an input file is annotated with semantic tags. The performed tagging is only for presentation, and RDF format would be considered at the moment.

V. Evaluations

In this section we deal with the performance and achievement of our system. To do so, the evaluation process is carried out in two phases. First, the system output was compared with manual output of a human annotator. It was thought that manual annotation is done under an ideal, highly accurate condition. Such evaluation, however, would be time-consuming and awkward especially when it involves a great number of documents and web-pages. As such, relying on software even with a margin of error would be reasonable. In the second phase of evaluation, the system output was compared with one of the existing annotation tools, called Ontea. We selected this tool since it was noticeably compatible with our system. Ontea employs regular expressions and patterns as well as knowledge base to perform annotation process. In this evaluation, 50 html web-pages on business job offer were delivered to both systems and both systems' outputs were compared. To cope with the task, following standard parameters were taken into account [15]-[9]:

$$\mathrm{Re}\,call = \frac{TP}{TP + FN} \quad (1)$$

$$\mathrm{Pr}\,ecision = \frac{TP}{TP + FP} \quad (2)$$

TP is the number of items correctly assigned to a category (True Positives).
FP is the number of items incorrectly assigned to a category (False Positives).
FN is the number of items incorrectly rejected from a category (False Negatives).

We also calculated F-measure, the harmonic mean of recall and precision:

$$F - measure = \frac{2 \times \mathrm{Re}\,call \times \mathrm{Pr}\,ecision}{\mathrm{Re}\,call + precision} \quad (3)$$

After achieving the outputs, the relevant parameters were calculated. The results are shown in Table I:

TABLE I
Comparison of our System with Ontea

	Recall	Precision	F-measure
Our approach	90%	75%	81%
Ontea	83%	64%	71%
Human	High	High	High

As shown in the Table I, the measure of recall indicates that only %10 of the required correct annotation is not performed by this system. In other words, in %90 of cases our system has managed to map the instances existing in the text to the appropriate concepts of the ontology, and the result is statistically satisfying. Needless to say, the amount of recall is likely to reach %100 if the structure of pages are improved.

The measure of precision parameter indicates that %25 of annotation performed by the system is incorrect, or an instance is mapped to a wrong concept. The high rate of this figure, i.e. %25 is due to the polysemy of words in different pages. Even sometimes one word may have two totally different concepts in two different documents with one similar domain. In such special case, our system inputs the concept in the second document as it was done in the former one. It

would be wrong, however, a user familiar with the domain is able to resolve the trouble. The F-measure also shows the general status of the system. In sum, the results of performance of our system imply its efficiency.

The main reason of our system's better performance is our more comprehensive knowledge base. As Ontea works only with patterns, it is more useful in pages which follow explicit, pre-defined structures. For example, if the name of a company that offers a job is as follows, Ontea would be able to identify it:

Company: Logitech

Therefore, it would be an appropriate tool to identify such pages. But, on pages which lack a clear-cut structure, Ontea fails to identify the existing entities of the text. The knowledge base of our system is a database including a quite complete lexicon as well as a comprehensive grammar and regular expressions, and also lists of various entities. It is not only a much better knowledge base that can identify the entities on explicit structures, but also it is able to identify the entities on unstructured pages. Table II, extracted from [3] indicates the superiority of our system to other mentioned ones.

TABLE II
EXPERIMENTAL RESULTS FROM [3]

	Method	relevance %	precision %	recall %	disadvantages	Adventages
Ontea	regular expresions, search in knowledge base (KB)	71	64	83	high recall, lower precesion	high succes rate, generic solution, solved duplicity problem, fast algorithm
SemTag	disambiguatiy check, searching in KB	high	high		works only for TAP KB and English	fast and generic solution
Ontea creation	regular expresions (RE), creation of individuals in KB	83	90	76	aplication specific patterns are needed	support Slovak language
Ontea creation RTFS, TS	RE, creation of individuals in KB + RFTS	73	94	69	low recall	disambiguities are found and not annotated
Wrapper	document structure	high	high		zero success with unknown structure	high success with known structure
PANKOW	pattern matching	59			low success rate	generic solution
C-PANKOW	POS taging and pattern matching Qtag library	74		74	suitable only for English, slow algorithm	generic solution
Hahn et al.	semantic and syntactic analysis	76			works only for English not Slovak	
Evans	clustering	41			low success rate	
Human	manual annotation	high	high	high	problem with creation of individuals duplicities, inacuracy	high recall and precesion

In general, our system performs successfully on pages which make use of numerous words and concepts. When the pages include a great number of figures, however, our system loses its efficiency. This problem arises because of our basic knowledge base, i.e. WordNet. The drawback could be overcome by structuring such pages using regular expressions.

VI. CONCLUSION

The Semantic Web requires the widespread availability of document annotations in order to be realized. Benefits of adding meaning to the Web include: query processing using concept-searching rather than keyword-searching [1]; custom web page generation for the visually-impaired [16]; using information in different contexts, depending on the needs and viewpoint of 48 the user [5]; and question-answering [10].

In this system, concepts are extracted based on a quite comprehensive knowledge base. This knowledge base includes a Basic Knowledge Base including a quite complete set of words, the sets of grammars and data frames, and various lists of different entities' names. The performed procedure in our system has been done under the control of a user familiar with the text domain, and therefore annotation process is performed semi-automatically. The superiority of our system to other similar ones is illustrated through a comparative study. Our future endeavor is enhancing the used algorithm, enriching the primary and secondary knowledge base, and also increasing the system's capability in identifying numerical concepts in unstructured web-pages. Other future work would be further evaluation on our suggested method considering other aspects. We hope to evaluate the system on higher number of pages, numerous domains, and pages with various contents including words, numbers, and figures.

REFERENCES

[1] T. Berners-Lee, J. Hendler., O. Lassila, "The Semantic Web," Scientific American, 2001, pp. 34-43.
[2] E. Charniak, M. Berland, "Finding parts in very large corpora," in *Proc. 37th Annual Meeting of the ACL Conf.*, 1999, pp. 57-64.
[3] M. Ciglan, M. Laclavik, M. Seleng, L. Hluchy, "Document indexing for automatic semantic annotation support," 2007.
[4] P. Cimino, S. Handschuh, S. Staab, "Towards the Self-Annotating Web," in *13th International Conf. on World Wide Web*, 2004, pp. 462-471.
[5] S. Dill, N. Eiron, D. Gibson, D. Gruhl, R. Guha, A. Jhingran, "SemTag and Seeker: Bootstrapping the Semantic Web via Automated Semantic Annotation," in *12th International World Wide Web Conf.*, Budapest, Hungary, 2003, pp. 178-186.
[6] S. Handschuh, S. Staab, F. Ciravogna, "S-CREAM -- Semi-automatic CREAtion of Metadata," in *SAAKM 2002 -Semantic Authoring, Annotation & Knowledge Markup - Preliminary Workshop Programme*, 2002.
[7] A. Kiryakov, B. Popov, I. Terziev, D. Manov, D. Ognyanoff, "Semantic Annotation, Indexing, and Retrieval," *Elsevier's Journal of Web Sematics*, vol. 2, 2005.
[8] N. Kiyavitskaya, N. Zeni1, J.R. Cordy, L. Mich, J. Mylopoulos, "Semi-Automatic Semantic Annotations for Web Documents," 2005.
[9] N. Kiyavitskaya, N. Zeni1, J.R. Cordy, L. Mich, J. Mylopoulos, "Tool-Supported Process for Semantic Annotation: An Experimental Evaluation," 2005.
[10] P. Kogut, W. Holmes, "AeroDAML: Applying Information Extraction to Generate DAML Annotations from Web Pages," in *Proc. Workshop on Knowledge Markup and Semantic Annotation at the First International Conference on Knowledge Capture* (K-CAP 2001), Victoria, BC, 2001.
[11] M. Laclavik, M. Seleng, E. Gatial, Z. Balogh, L. Hluchy, "Ontology based Text Annotation – OnTeA," *Information Modelling and Knowledge Bases XVIII. IOS Press, Amsterdam, Marie Duzi, Hannu Jaakkola, Yasushi Kiyoki, Hannu Kangassalo (Eds.), Frontiers in Artificial Intelligence and Applications*, vol. 154, February 2007, pp.311-315.
[12] G. Miller, "WordNet: An On-line Lexical Database," *Special Issue, International Journal of Lexicography*, vol. 3, 1990. WordNet: http://wordnet.princeton.edu/
[13] B. Popov, A. Kiryakov, A. Kirilov, D. Manov, D. Ognyanoff, M. Goranov, "KIM – Semantic Annotation Platform," in *2nd International Semantic Web Conf. (ISWC2003)*, Florida, USA, 2003, pp. 834-849.
[14] L. Reeve, H. Han, "Survey of semantic annotation platforms," in *SAC '05*, ACM Press, NY, USA, 2005, pp. 1634-1638.
[15] Y. Yang, "An evaluation of statistical approaches to text categorization," *Journal of Information Retrieval*, vol. 1, 1999, pp. 67–88.
[16] Y. Yesilada, S. Harper, C. Goble, R. Stevens, "Ontology Based Semantic Annotation for Visually Impaired Web Travellers," in *Proc. 4th International Conference on Web Engineering (ICWE 2004)*, Munich, Germany,2004, pp. 445-458.