Contextual Concept Discovery Algorithm

Lobna Karoui, Marie-Aude Aufaure, Nacera Bennacer
Ecole Supérieure d'Electricité
91192 Gif-sur-Yvette, France
{Lobna.Karoui, Marie-Aude.Aufaure, Nacera.Bennacer}@Supelec.fr

Abstract

In this paper, we focus on the ontological concept extraction and evaluation process from HTML documents. In order to improve this process, we propose an unsupervised hierarchical clustering algorithm namely "Contextual Concept Discovery" (CCD) which is an incremental use of the partitioning algorithm Kmeans and is guided by a structural context. Our context exploits the html structure and the location of words to select the semantically closer cooccurrents for each word and to improve word weighting. Guided by this context definition, we perform an incremental clustering that refines the context of each word clusters to obtain semantically extracted concepts. The CCD algorithm offers the choice between either an automatic execution or a user's interaction. The last function of the CCD algorithm is to provide a complementary support for an easy evaluation task. This functionality is based on a large collection of web documents and several context definitions deduced from it by applying a linguistic and a documentary analysis. We experiment our algorithm on HTML documents related to the tourism domain. Our results show how the execution of our context-based improves the conceptual quality and the relevance of the extracted ontological concepts and how our credibility degree criterion assists the domain experts and facilitates the evaluation task.

Introduction

Generally, the idea of using a context in each real situation is supported by the assumption that a context or a set of contexts may either contain the solution to a problem or may provide sufficient supporting data for a target object. In our work, we focus on the use of context in the ontology learning process. Most works have investigated various issues of ontology building such as methodology and automation aspects (Faure and Nedellec, 1998), (Maedche and Staab, 2001). Recently, some other researches are interested to the ontology evaluation (Maedche and Staab, 2002), (Navigli et al, 2004), (Holsapple and Joshi, 2005). In this paper, we focus on the ontological concept extraction and evaluation process from the HTML documents. In order to improve it, we propose an unsupervised hierarchical clustering algorithm namely "Contextual Concept Discovery" (CCD) based on an incremental use of the partitioning Kmeans algorithm, guided by a structural context and producing a support for an easy evaluation task. Our context definition is based on the html structure and the location of words in the documents. Each context is deduced from the various analyses included in the pre-processing step (Karoui et al., 2006). This explicitly contextual representation, titled "contextual hierarchy", guides the clustering algorithm to delimit the context of each word by improving the word weighting, the words pair's similarity and the semantically closer cooccurrents selection for each word. By performing an incremental process and by recursively dividing each cluster, the CCD algorithm refines the context of each word cluster. It improves the conceptual quality of the extracted concepts. The CCD algorithm offers the choice between either an automatic execution or an interactive one with the user. In order to help the domain expert during the evaluation, the last part of the CCD algorithm exploits a web collection and extracts the existing contexts. This information is used to compute the credibility degree associated to each word cluster in order to inform about it and facilitate the experts' semantic interpretation. So, the CCD algorithm extracts the domain concepts and proposes a quantitative and qualitative evaluation for the experts. Our evaluation proposition permits the ontology reuse and evolution since the informing elements that support the experts' interpretation are driven by the web changes and are stored with the experts' comments for a later use. We experiment the contextual clustering algorithm on French html document corpus related to the tourism domain. The results show that the appropriate context definition and the successive refinements of clusters improve the relevance of the extracted concepts in comparison with a simple Kmeans algorithm. Also, our observations and discussions with experts confirm that our evaluation process helps and assists the user.

The remainder of the paper is organized as follows: section 2 defines the context and the Contextual Concept Discovery" (CCD) algorithm, section 3 experiments our algorithm and section 4 concludes and gives some future directions.

The Contextual Concept Discovery Algorithm

Our algorithm proposition is constituted of three parts which are: a structural context definition, a clustering process to group the semantic words together and a web driven evaluation task based on various context types and a quantitative criterion (credibility degree).

Our context definition

In this section, we focus on the selection of the semantically closer cooccurrents and the word weighting process.

Let us go back to our corpus and the various analyses performed on the HTML documents. We note that there exist relations between the existing HTML elements. For instance: <h1> → <p> (heading → paragraph). We also note that key tags (defined tags like <keywords>, <glossary>, etc.) and <title> are related to other existing HTML elements. For example: <TITLE_URL> (header of a hyperlink) → <H1> (headings of a part of document). The first group of links is physical noted P.L because it depends on the structure of the HTML document. But the second group shows a logical link (L.L) that is not always visible (elements are not necessarily consecutives). In order to represent links between tags, we define two new concepts: "contextual hierarchy" (C.H) based on HTML elements and "link co-occurrence" (L.C). A contextual hierarchy is a tag hierarchy. It illustrates possible relations existing within HTML documents and between them.

When we find two terms in the same context unit (paragraph, text), we speak about co-occurrence between these two words in a unit related to the context. In our study, the concept of context is variable, so we do not fix the context during the process but we define a new concept which is a contextual hierarchy represented explicitly. By taking this structure into account, we can define a link between terms, if they appear in the same block-level tag (<p>, <td>, etc.). In this case, the link is a neighbourhood co-occurrence (N.C) and the context is limited to the tag. However, if they appear in different tags that are related by a physical or a logical link defined in the contextual hierarchy, we define the concept of "link co-occurrence" (see Figure.1). In this second case, the context is the association of the two related tags. The neighbourhood co-occurrence permits finding the word's cooccurrents in a unique context while a link cooccurrence depends on the word position in a context and the relation between this context and the other existent ones. So, it is a generic context that will be instantiated according to the location of the term in HTML tags (for example the context can take various values since it could be in one case the tag <B> and in another case the association of <H1> and <TITLE>).

The structural context definition represents the term's adaptability in its corpus. The associated contextual model respects the word location in order to take a multitude of term situations into account when computing its weight. The term weighting is computed by applying the "Equivalence Index" defined by Michelet (1988) (Ci : term occurrence; Cij: terms cooccurrences) :

$$Eij = C2ij / (Ci \times Cj) \qquad (1)$$

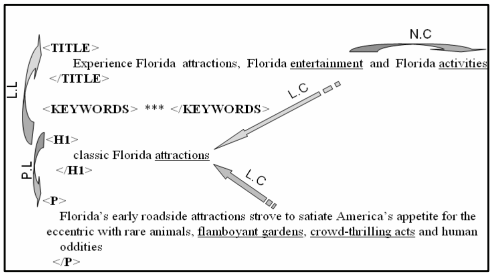
N.C
L.L
<TITLE>
Experience Florida attractions, Florida entertainment and Florida activities
</TITLE>
L.C
<KEYWORDS> *** </KEYWORDS>
<H1>
classic Florida attractions
</H1>
P.L
L.C
<P>
Florida's early roadside attractions strove to satiate America's appetite for the eccentric with rare animals, flamboyant gardens, crowd-thrilling acts and human oddities
</P>

Figure.1 Examples of contexts use and their deduced cooccurrents

The Fundamental directives of the evaluation task

Our idea is as follows: "looking in the Web in order to understand the meaning of each word or two words together and so on" could be a solution but why?

This task is a contextualization operation. Brézillon (1999) says: "Context is what gives meaning to data and contextualization is the process of interpreting data, transforming data to information".

In order to extract concepts, terms are selected from their context in order to group them, but they are presented to the knowledge engineer or the expert domain without any context after a decontextualization process that's why the evaluation step is always difficult. McCarthy (1993) says that decontextualization is "to abstract a piece of knowledge from contexts into a more general context that cover the initial context". In our case, for each cluster the general context is the domain (tourism). But this information is not sufficient to evaluate a cluster and to give it a semantic tag.

A solution for ensuring an easy analysis requires a big collection of web documents related to the same studied domain and extracting several contexts from them.

The context granularities degrees. Our hypothesis is "Having words in the same context imply that they share common information which permit to attribute an appropriate concept for these terms". In this case a context is an appropriate support for a semantic interpretation i.e it limits the associated knowledge of each word and gives a background for the evaluation and labeling task. In order to explain this idea, we take the sentence: "the possible accommodations in the east region of USA are hotels and residences. Within this example, when we limit the context to the association of 'hotels' and 'residences' by the conjunction 'and', we deduce that 'hotels' and 'residences' belongs to the same concept. However, when we limit the context to the entire sentence, we can say that the associated concept to these two words is 'accommodation'. So, thanks to the contextualization task, we can deduce either the meaning of each word, or the semantic association between some words or the concept associated to some words. Taking into account a static context i.e only one such as a sentence for all the word clusters is not sufficient since in some case the sentence does not contain all the words of a cluster. That's why, our evaluation is not restricted to a unique context on the contrary it depends on various granularity levels which are applied and considered consecutively. The several contexts defined from the domain web documents are provided by two sources. The first one is a linguistic analysis that permits to give us the various nominal groups and verbal groups. It also procures the various word associations by a preposition (of, on, etc.) or a co-ordinating conjunction (and, or, etc.). The second source is a documentary analysis that permits to give us the various sections of phrases (part of a phrase finished by a punctuation like ';' or ','), the sentences, the paragraphs and the documents. So, we have two types of contexts which are a linguistic context and a documentary context. By using the first one, we obtain the close words of our target terms. By using the second one, the context is more generalized than the linguistic one and the information deduced will be either complementary information or completely new information for the words of a cluster.

Our context definition is dynamic since it depends on the presence of the target words in one context. For example, with a cluster with four words and by using two nominal groups, we find that these words are associated and we can give them a concept so we are not obliged to look for their documentary context. But, when the 8 words on a cluster do not belong to one of the four results of the linguistic context, we are obliged to look deeply into the documentary context. So, the expert evaluation thanks to our contextualization process is easier than the existing ones and it is done by respecting this order for each word clusters: (1) Linguistic context (Nominal groups based context, Verbal groups based context, Prepositional groups based context, Conjunctional groups based context); (2) Documentary context (Sections of phrase based context, sentences based context, paragraphs based context, documents based context).

In order to present a context analysis example, in the following sentence "the possible accommodations are hotels and residences", we find that "are hotels and residences" is the verbal group and "the possible accommodations" is the nominal group. So this sentence which is a context contains two contexts.

Algorithmic Principles

In this section, we present an unsupervised hierarchical clustering algorithm namely "Contextual Concept Discovery" (CCD) to extract ontological concepts from HTML documents. It is based on an incremental use of the partitioning algorithm Kmeans and is driven by the structural contexts in which words occur. We chose Kmeans because it is an unsupervised algorithm able to classify a huge volume of objects within a short execution time.

The clustering algorithm proceeds in an incremental manner. It computes the occurrences of each word and selects their semantically closer cooccurrents according to the context definition. Then, it divides the clusters obtained at each step in order to refine the context of each group of words. So, the algorithm refines at the same time the context of a word and the context of each cluster.

Also, it offers to the user the possibility to choose either a complete automatic execution or an interactive one. If he/she decides the first execution manner, he/she should either define some parameters or choose the default ones resulting from our empirical experiments. These parameters are: the highest number of words per cluster P, the accepted margin M representing an additional number of words in a resulting cluster accepted by the user and the similarity measure S. If he prefers to evaluate the intermediate word clusters, he should choose the interactive execution. In this case, the algorithm allows him to analyze the word cluster at the end of each intermediate clustering in order to define the value of k' and to decide whether he prefers to continue within an interactive execution or to run an automatic one for the rest of the clustering process. In the interactive execution, the process takes longer than the automatic one but it offers an opportunity to the user to intervene in order to obtain better hierarchical word clusters.

As input to this algorithm, firstly, the user should choose the input dataset. Secondly, he defines the number of clusters K, and chooses whether he prefers an automatic execution or an interactive one. An automatic execution of our algorithm is defined in four steps:

Step 0: Applying the context definition. In this step, the algorithm takes into account the data file in which we find the word and their outbuilding to the html tags. Based on this input, it applies the context definition and returns a file containing the candidates (words), their cooccurrents (attributes) and their associated weightings.

Step 1: Executing the first Kmeans. The goal of this step is to execute the kmeans algorithm in order to obtain the first word distribution Di. We obtain k word clusters.

➔ These steps 0 and 1 concern the application of the structural context which is deduced from the various analyses, computed from the dataset F and stored in a matrix and the execution of the partitioning algorithm to obtain the first k clusters.

Step 2: Performing the hierarchical clustering. This step finds clusters respecting the criterion defined by the user: the P value. For each intermediate execution of kmeans, we should define the value K' which represents the number of clusters related to the words of a cluster that could be divided. In an automatic execution of the algorithm, this value is not defined by the user but computed by the system. We implement a proportional function that automatically defines the value of K'. Based on several empirical experiments and the domain experts' knowledge, we define a proportional link needed for our function. By using this information and solving the following equation, we can compute the value of K'. The useful equation, which is applied in any domain, is:

K' = a *ln (Word-Number (Ci)* b). (2)

If the number of words in a cluster (Word-Number (Ci)) is less than (1/b), the value of K' will not be computed but defined as 2. When the number of words per cluster Ci is less or equal to P, we include the cluster Ci to the set WC.

Step 3: Affecting the single word to the formed clusters. When applying the division process, we can obtain clusters with only one word. Our idea is to automatically associate each word alone in a cluster resulting from step 2. Another problem appears when the algorithm affects too many words to the same cluster (by respecting the similarity). In this case and for little clusters, we can obtain clusters with a great number of words. If a word is assigned to a cluster already containing P+M words, the CCD algorithm will choose the cluster which is the closest centroid to the target word.

Choosing an interactive process implies applying the same steps but with the user intervention firstly after step 1 and during step 2 and 3. The clustering method adopts a vector-space model and represents a term as a vector containing attributes that belong to the corpus and are stored in a matrix. In step 2, if a cluster is divided, the algorithm allows to refine the context of each cluster by taking into account only the associated attributes of its belonging words. By applying this method, the similarity computed better represents the association degree between each two words. By applying the step 3, it avoids the cases of having only one word in a cluster and those containing the majority of the single words.

Step 4: Evaluating the word clusters. To obtain a domain web collection of French documents, we use a cleaner (for example HTTrack Website Copier). Then, we treat them thanks to the pre-processing step of our system. This last cleans and structures the collected web pages. Then, we perform various analyses (Karoui et al., 2006). Now, the problem is that the expert is incapable, even when he is given all the results of the analysis, to find the possible association of the targets words, especially when working on a big corpus. In order to facilitate this process, we define a semantic index which represents the credibility of the target words' association in relation with the different contexts. This index is named "credibility degree". It is computed for each word cluster and for each context definition in an automated way. More details about this criterion and its importance are provided in sub section titled "the impact of the credibility degree".

In the following section, we experiment the Contextual Concept Discovery algorithm.

Experimental Evaluation of the Contextual Concept Discovery Algorithm (CCD)

In our experiments and in order to obtain ontological domain concepts incrementally, we begin the process with key and title tags to give an outline of the domain's semantic information. These terms are those chosen by the site designer as keyword, document title, glossary, etc. So, our general context is the first two levels of the contextual model (key tags + title tags+ headings tags). Our objective is to define an incrementally concept extraction process. In our next research, this method will progressively integrate all the html elements and the terms of the corpus in order to obtain more domain concepts. In this section, we evaluate the CCD algorithm results by comparing them to those obtained by the Kmeans one. We chose the Euclidian distance as a similarity measure. Our dataset is composed of 872 words. The Kmeans algorithm distributes them in 156 clusters. The CCD algorithm is experimented with various values for each parameter. We present an automatic execution with respectively to k, P and M the values 20, 10 and 22 (more significant results). We obtain 162 clusters.

Before executing the step 4 of the CCD algorithm, we propose a manually domain experts' evaluation. We present the results to two domain experts. As a first step, individually, each of them evaluates and labels manually the word clusters. Then, they work together in order to discuss about the results, their labels proposition and to give us only one evaluation and labeling for which they agree. In our experiments, we obtain two results. The first one is a comparison between two contextual definitions in order to show how the definition of a context improves the words' pairs weighting and the words' pair's similarities. These results are presented in (Karoui et al., 2006).The second one is a comparison between two algorithms with the same context definition by defining four criteria which are:

Word Distribution. With the Kmeans algorithm, we have 13% of our initial words that are grouped together. While with the CCD algorithm, we obtain only 3.66% of our initial set of words in the same cluster.

Semantic Interpretation. The domain expert notes that there are three types of word clusters which are advisable clusters, improper clusters and unknown clusters. Advisable clusters are those for which the expert is able to associate a label and in which words belonging to the same group are close semantically. Improper clusters are either those with an amount of words having no relation with the principle extracted concept, or those containing more than one concept. Unknown clusters are clusters where words do not have any semantically relation and the expert could not find any semantic interpretation. Thanks to the P and the M values, in each word cluster, the percentage of noisy elements decreases a lot. As a consequence, the percentage of unknown and improper clusters is reduced (respectively 20. 51% and 26. 28% for the kmeans algorithm versus only 14. 81% and 16. 66% for the CCD algorithm). Moreover, we obtain 68.52% advisable clusters with the CCD algorithm which is more important than only 53.2% with the Kmeans one.

Extracted Concepts. We take into account only the advisable clusters in the two cases and we compute the precision. In our study, "Precision is the ratio of relevant words having between them a great semantic similarity with total words for a given cluster". By applying this criterion, we obtain respectively 86.18% and 86.61% with the kmeans algorithm and the CCD one.

Generality Degree of the Extracted Concepts. Another element which affects the concept's quality is the level of generality for a concept. In order to evaluate the generality degree of the concepts, we focus only on concepts extracted from the advisable clusters. We based our manual evaluation on the OMT thesaurus (1999). It contains generic and open terms presenting general key domain concepts. In our experiments, we obtain respectively with the kmeans algorithm and the CCD one 78.31% and 85.58% general concepts. Also, we remark that the clusters obtained with the CCD algorithm are more enhanced than with the Kmeans algorithm. For example, we find with the Kmeans algorithm the cluster C1:{Event, festival, music} while with the CCD one we obtain C2:{Event, festival, music, party}.

The impact of the P parameter and the step 2 of the CCD Algorithm. A first execution of the kmeans algorithm with 20 clusters (step1) gives a cluster having 68.69% of the initial set of words. By defining the P parameter, we decide to divide the word clusters and consequently to perform an intermediate clustering based only on the common attributes of the P words. So, we refine the context of this cluster and we obtain better results thanks to new similarities between the P words. For example, with the kmeans algorithm, we found the word "civilization" with a big set of words without any relation between them, but with the CCD algorithm, we found it in a cluster where there are words semantically similar like "archaeology", "ethnology", etc.

The impact of the step 3 of the CCD Algorithm. The CCD algorithm allows assigning these single words to the existing clusters. We remarked that some words are assigned to the appropriate cluster. For example before and after the step 3, we obtain respectively the clusters {Academy, club, golfer} and {Academy, club, golfer, golf}.

We now present the importance of the credibility degree criterion in the evaluation task, especially to facilitate the semantic interpretations of the results by the domain experts.

The impact of the Credibility Degree. Let us take some resulting word cluster in order to explain concretely the credibility degree's importance:

Table 1. Examples of term clusters

Examples	Word Clusters
Example 1	academy, golf, golfer, club
Example 2	Civilization, archeology, ethnology, people
Example 3	Park, national, cliff, rock
Example 4	Cult, church, evangelization, memory, religious, sanctuary
Example 5	excursion, foot, person
Example 6	Hiker, gorges

Our « Credibility Degree Computation » criterion is executed on a set of word clusters in order to compute their credibility degrees. For instance, with the example 1 (Table 1) and according to one context definition (nominal groups or sentence), the algorithm finds all the possible combination in the context i.e tries to find the four words (academy, golf, golfer, club), then the association of three words and so on. For each found association, it presents the associated words and gives a degree representing how many times this type of association is found. For example, with the same example, it finds two possible associations with three words which are {academy, golf, golfer} and {golf, golfer, club} so the credibility degree is 32 i.e two associations of three words.

Our criterion has several functionalities which are:

- Finding the associations between some words in order to facilitate the labelling step. With the Example 5, our criterion finds only the association that permits to the user to give a label by him self like 'excursion on foot'.
- Finding in the same time the available association in the context and the concept (Example 2, the concept is 'civilization').
- Detecting the noisy elements in a cluster and either delete them or move them to another cluster. For instance, in the Example 5, the word 'hiker' is found inside the several association returned by our algorithm and corresponding to 'excursion' and 'foot' but the problem is that this word belongs to the Example 6. Our evaluation process decides to remove it from the example 6 and includes it into the example 5 because no association is found related to the other words of the cluster 6. If the words exist in two associations related to two clusters, the CDD algorithm presents the word using the red colour to announce a possible ambiguous situation.
- Enhancing a cluster by other words from the associations. For the Example 3, we can enrich the group by the word "nature" and "inheritance" and find the concept which is 'natural inheritance'.

Since our evaluation task is based on various context definitions, if the user does not find a connection between some words by using the linguistic contexts, he can analyse the provided association by the documentary contexts in which the probability to find more relation is bigger than with the first context type (Example 4).

Thanks to the credibility degrees computed for each cluster and for each context, the user obtains an amount of information useful and in some cases sufficient to manipulate (delete word, remove word, etc.), evaluate and label the cluster. For example, for a same cluster, if he finds the three credibility degrees (5_1, 4_3, 3_8, 2_{15}) which are a quantitative indications, he starts by analysing the association with 5 words. If it is not sufficient, he analyses the three associations of four words (4_3) and so on. If the information returned by our algorithm to this cluster and for one context is not enough, he can look to the other credibility degrees provided by the other contexts by respecting the previous order (linguistic context type then documentary context one).

So, our web driven concept evaluation method provides two revealing aspects: the qualitative ones based on the word associations deduced from the various contexts and the quantitative ones resulting from the computed credibility degree index. The qualitative evaluation provides a semantic support for an easy user interpretation. Moreover, our proposition, based on a large collection of domain web documents, several context definitions and different granularity degrees, permits to an ordinary user to help the expert by manipulating the word clusters and giving him semantic tags as suggestions. Consequently, the expert should decide on the appropriateness of these labels as well as clusters homogeneities which are not labeled. Moreover, our algorithm assures ontology reuse and evolution since the elements on which the expert's interpretations are based (the provided word associations) depend on the web changes. For example, when the web documents change, the various extracted contexts change too and the results of the mapping operation between the words belonging to the clusters and the contexts are updated. This resulting information about the word clusters is presented to the experts in order to help them during the evaluation task. Then, they are stored with the experts' comments in order to be reused by another expert either during the same period or later (after some months or years depending on the frequency of updates). Also, thanks to our context based hierarchical clustering, the CDD algorithm follows the document changes whether it is at the level of the structure or the content.

Conclusion

It is necessary to discover and organize the available knowledge about an application domain in a coherent manner. In this paper, we defined a hierarchical clustering algorithm namely "Contextual Concept Discovery" (CCD), based on an incremental use of the partitioning algorithm Kmeans, guided by a structural contextual definition and providing a web driven evaluation task to support the domain experts. The CCD algorithm proceeds in an incremental manner to refine the context and to improve the conceptual quality of each word cluster. The last functionality of our algorithm is a new evaluation method based on a large web collection and different contexts extracted from it. We experiment the CCD algorithm to incrementally extract ontological concepts. We show that the context-based hierarchical algorithm gives better results than a usual context definition and an existing clustering method. Moreover, our evaluation process permits to help either an ordinary user (knowledge engineer) or the expert to take the write decision about the semantic homogeneity of a cluster. As a perspective, we will define and experiment (with the CCD algorithm) a linguistic context and a documentary context combined with the structural one and apply them to words belonging to the HTML tags in order to enhance the set of extracted concepts, to obtain more specific concepts and to take into account other types of documents such as textual ones. Also, we will develop ideas related to the instance and relation discovery.

References

Brézillon, P. 1999. Context in problem solving: A survey. The Knowledge Engineering Review, Volume: 14, Issue: 1,Pages: 1-34.

Faure, D., Nedellec, C. and Rouveirol, C. 1998. Acquisition of semantic knowledge using machine learning methods: the system ASIUM. Technical report number ICS-TR-88-16, inference and learning group.

Holsapple, C. and Joshi, K.D. 2005. A collaborative approach to ontology design. Communications of ACM, 45(2): 42-47.

Karoui, L, N. Bennacer, M-A., Aufaure. 2006. Extraction de concepts guidée par le contexte », XIIIème Rencontres de la Société Francophone de Classification.

McCarthy, J. 1993. Notes on formalizing context. Proceedings of the 13th IJCAI, Vol. 1, pp. 555-560.

Maedche, A. and Staab S. 2001. Ontology learning for the semantic Web, IEEE journal on Intelligent Systems, Vol. 16, No. 2, 72-79.

Maedche, A and Staab, S. 2002. Measuring similarity between ontologies. Proc. CIKM 2002. LNAI vol.2473.

Michelet, B. 1988. L'analyse des associations. Thèse de doctorat, Université de Paris VII, UFR de Chimie, Paris.

Navigli, R., Velardi, P., Cucchiarelli, A. and Neri, F. 2004. Quantitative and qualitative evaluation of the ontolearn ontology learning system. In Workshop on ontology learning and population, Valencia, Spain.

Han, H. and Elmasri, R. 2000. Architecture of WebOntEx: A system for automatic extraction of ontologies from the Web. Submitted to WCM 2000.