Supporting the Discovery and Labeling of Non-Taxonomic Relationships in Ontology Learning

J. Villaverde, A. Persson, D. Godoy and A. Amandi

ISISTAN Research Institute, UNICEN University
Campus Universitario, Paraje Arroyo Seco
CP 7000, Tandil, Bs. As., Argentina, Tel/Fax: +54 (2293) 440362
Also at CONICET, Consejo Nacional de Investigaciones Científicas y Técnicas
Bs. As. (C1033AAJ), Argentina

Abstract
Ontology learning (OL) from texts has been suggested as a technology that helps to reduce the bottleneck of knowledge acquisition in the construction of domain ontologies. In this learning process, the discovery, and possibly also labeling, of non-taxonomic relationships has been identified as one of the most difficult and often neglected problems. In this paper, we propose a technique that addresses this issue by analyzing a domain-text corpus to extract verbs frequently applied for linking certain pairs of concepts. Integrated in an ontology building process, this technique aims to reduce the work-load of knowledge engineers and domain experts by suggesting candidate relationships that might become part of the ontology as well as prospective labels for them.

1. Introduction
Ontology building and its representation in a formal language is usually carried out by Knowledge Engineers (KE), sometimes with the assistance of domain experts. This process involves the acquisition, conceptualization, evaluation and formalization of domain-dependent knowledge. The manual construction of ontologies has been largely identified as a costly, time-consuming, tedious and error-prone task [27]. Furthermore, additional technical difficulties such as the lack of standards to integrate or re-use existing ontologies and the absence of fully automatic knowledge acquisition methods have been reported as issues that hinder ontology building [28].

In recent years, the acquisition of ontologies from domain texts using machine learning and text mining methods has been proposed as a means of facilitating the ontology engineering process. In this context, ontology learning [17] has been identified as an emerging field which aims at assisting knowledge engineers as well as end-users in ontology construction. It can be seen as a multi-disciplinary field, which integrates disciplines such as ontology engineering, machine learning, and natural language processing, among others. The use of these technologies is distributed in three main phases, lexical entry extraction, taxonomy
extraction, and non-taxonomic relation extraction [17]; which all together allow either building an ontology from scratch or enriching an existing ontology using multiple sources of information.

Ontology learning from texts constitutes a promising means for ontology engineers to significantly speed up the ontology building process so that several approaches have been proposed for covering the different phases it involves. In this process, the phase of extraction of non-taxonomic relationships has been recognized as one of the most difficult [17] and least tackled problems [25]. This phase can be divided in two different problems: discovering the existence of a relationship between a pair of concepts and then labeling this relationship according to its semantic meaning. The assignment of labels to relationships is also difficult since various relationships among instances of the same general concepts are possible [17]. Moreover, even if the semantic is clear, it might still be hard to guess which among several synonymous labels are preferred by a certain community [10].

In this paper we propose a technique for the discovery of non-taxonomic relationships and the extraction of lexical items acting as connectors between the related concepts. This technique is based on the analysis of syntactic structures and dependencies among concepts existing in a domain-specific text corpus. Our technique tries to find information on the semantic relationships between concepts denoted by the verbs usually employed to connect them. Based on this information, the technique aims at supporting the work of the ontology engineers by suggesting possible relationships among concepts and helping them to select the most appropriate labels according to the analysis and processing of a domain text corpus.

This article is organized as follows. Section 2 presents the proposed technique for discovering and labeling non-taxonomic relationships starting from domain documents. Section 3 illustrates this process with an example. Section 4 explains the experiments conducted to validate our method and tune the parameters of the proposed technique. Section 5 discusses related works. Finally, concluding remarks are stated in Section 6.

2. Extraction of Non-taxonomic Relationships

In the technique introduced in this section both the existence and possible labels of a non-taxonomic relationship are identified through the exhaustive exploration of lexical items that recurrently describe an action between a pair of concepts. In this context, we combine statistical and linguistic analysis in order to exploit the syntactic structure and dependencies existing in texts for extracting relationships. Given a domain-specific text corpus, our method seeks for verbs that occur frequently in the context of pairs of lexical items describing concepts.

In order to illustrate the proposed approach, suppose a concept hierarchy of a medical domain ontology providing concepts like *Surface Receptor* and *T cell Activation*, but no relationship between the two of them. By examining some medical texts we may find sentences where both concepts appear together,
Figure 1: Overview of the process of discovering and labeling relationships

such as “Activation of the CD28 surface receptor provides a major costimulatory signal for T cell activation resulting in enhanced production of interleukin-2 (IL-2) and cell proliferation”. In this sentence the concepts seem to be related by the verb “provides”. If we gather enough statistical evidence to support the fact that “Surface Receptor” and “T cell Activation” might be related by “provides” we could suggest the inclusion of this relationship in the ontology.

For discovering and labeling non-taxonomic relationships, our technique takes as input a corpus of domain-specific texts and either a list of candidate concepts or a concept hierarchy describing taxonomic relationships among concepts and looks for further possible relationships among these concepts in the text corpus. The discovered relationships are suggested to the knowledge engineer in order to be added to the domain ontology or validated with domain experts. The process of discovering and labeling non-taxonomic relationships is composed of a number of stages which are described in the following subsections. Figure 1 shows a schematic view of the complete process.

2.1. Text Pre-processing and Indexing

In the first stage, unstructured documents are transformed into suitable representations for been used as input for machine learning algorithms by applying a number of pre-processing steps. This pre-processing stage first takes the documents and applies a stop-word elimination process. A list of stop-words or negative dictionary is composed of a set of words that due to their frequency or their semantic do not have sufficient discriminative power [13]. These words add few or no meaning to the document representation to be used later in keyword searching, thus a substantial performance improvement is achieved by simply eliminating stop-words.

After the elimination of these words, text documents are translated to a bag-of-words representation according to the Vector Space Model (VSM) described in [24]. In this model, each document is identified by a feature vector in a space in which each dimension corresponds to a distinct term associated with a numerical value or weight indicating its importance. Each document \( d \) from
the document collection $\mathcal{D}$ is then identified by a vector in the $t$-dimensional space, where each vector component $w_{ij}$ represents the weight of term $t_i$ in the document $d_j$:

$$d_j = (w_{1j}, w_{2j}, \ldots, w_{|\mathcal{T}|j})$$

(1)

Each term $t \in \mathcal{T}$, where $|\mathcal{T}|$ represents the number of terms in the document, is weighted using the Term Frequency-Inverse Document Frequency (TF-IDF) [23] term weighting function shown in Equation 2.

$$tfidf(t_k, d_j) = \#(t_k, d_j) \times \log \left( \frac{|T_r|}{\#_{T_r}(t_k)} \right)$$

(2)

where $t_k$ denotes a term, $d_j$ denotes a document, $T_r$ is a set of documents used for training, $\#(t_k, d_j)$ denotes the term frequency, that is, the times the term $t_k$ occurs in $d_j$, $|T_r|$ denotes the number of documents in $T_r$, and $\#_{T_r}(t_k)$ denotes the document frequency, that is, the number of documents in $T_r$ in which $t_k$ occurs.

The next step consists in document indexing, the task of assigning terms to documents for retrieval purposes [4]. In this work, sentences are considered as text units for analysis since they are more informative than documents for detecting occurrences of concept pairs (i.e., concepts occurring in different parts of a same document can not be linked by a single verb). For this purpose, we used Lucene\(^1\) [6], an open-source tool that provides full-text indexing and searching capabilities. This pre-processing step enables an efficient retrieval of sentences based on keyword-based queries for the search of recurrent patterns in later stages.

2.2. Identifying Relationship Occurrences

The goal of this stage is to discover patterns in the domain documents indicating the existence of some relation between a pair of concepts. Non-taxonomic relationships are usually expressed by verbs relating pairs of concepts [9, 25]. Then, co-occurrence of concepts together with the verb describing the nature of the relationship appearance are extracted in this stage. Each of these concept co-occurrences with the associated verb can be considered as a candidate relationship that must undergo a validation process in further stages before being suggested to knowledge engineers or domain experts as a possible relationship existing in the domain between concepts.

Initially, the set of all pairs of concepts $\mathcal{P}$ is generated using as input a concept hierarchy modeling a number of domain concepts $c \in \mathcal{C}$ and, optionally, their taxonomic relationships:

$$\mathcal{P} = \mathcal{C} \times \mathcal{C}$$

(3)

\(^1\)http://lucene.apache.org/
Each concept belonging to $C$ is expanded to include its synonymous using WordNet\textsuperscript{TM}\cite{31} lexical database before creating $P$. In this database, terms are organized into synonym sets, named synsets, each representing one underlying lexical concept. For instance, the items *organism* and *being* constitute a synset representing the concept corresponding to the dictionary definition “*a living thing that has (or can develop) the ability to act or function independently*”. The use of synsets instead of single keywords enables to increase the matching rate when looking for concepts in the text corpus.

Each co-occurrence of a given pair of concepts in a sentence of some document in the text corpus has to be completed with the addition of the verb that links these two concepts. Hence, each element $o$ in the set $O$ of relationship occurrences resulting of this stage will be a three-tuple as follows:

$$o = \langle \text{concept}_1, \text{concept}_2, \text{verb} \rangle$$

where $\text{concept}_1$ and $\text{concept}_2$ are synsets representing two different concepts and $\text{verb}$ is the verb that was found connecting words belonging to each of these synsets.

In order to generate $O$, all sentences from the document collection having both candidate concepts of a potential relationship are first retrieved using the index created in the previous stage, which allows an efficient retrieval of sentences from the document collection, and the verbs acting as nexus between the two concepts have to be discovered.

The Stanford Part-Of-Speech (POS) tagger\textsuperscript{2} [30, 29] is used to assign tags to content and function words, and to identify noun and verb phrases in each sentence. Part-of-speech tagging is the process of assigning a part-of-speech, like noun, verb, pronoun, preposition, adverb, adjective or other lexical class marker to each word in a given sentence [8]. The input to a tagging algorithm is a natural language sentence and a specified tag set (a finite list of part-of-speech tags), whereas the output is a single best POS tag for each word.

The POS tagger is applied to all the sentences recovered from the document collection containing both candidate concepts looking for relations fulfilling the pattern: $\langle \text{term} \rangle \langle \text{verb} \rangle \langle \text{term} \rangle$, where $\langle \text{term} \rangle$ is a member the synsets corresponding to either the first or the second concept involved.

To ensure that the occurrence of both concepts takes place in a same semantic context within a document, the search is restricted to concept appearances separated by no more than $N$ terms within a sentence. *Lucene*, the library used for indexing the document corpus, provides a rich group of queries that can be applied during the creation of the index to facilitate the search of nearby concepts.

\textsuperscript{2}http://wordnet.princeton.edu/
\textsuperscript{3}http://nlp.stanford.edu/software/tagger.shtml
2.3. Mining Associations

Once a set candidate relationships between concepts is available, evidence have to be collected to validate them before making suggestions to end users. To this matter we use an association rule mining algorithm, which discovers elements that co-occur frequently within a data set [18]. This provides statistical information of concepts and verbs that frequently co-occur within sentences of the domain documents.

Formally, the problem of mining association rules can be stated as follows [1]. Let \( I = I_1, I_2, \ldots, I_m \) be a set of binary attributes, called items. Let \( T \) be a database of transactions. Each transaction \( t \) is represented as a binary vector, with \( t[k] = 1 \) if item \( I_k \) appears in \( t \), and \( t[k] = 0 \) otherwise. Let \( X \) be a set of some items in \( I \). The transaction \( t \) satisfies \( X \) if for all items \( I_k \) in \( X \), \( t[k] = 1 \).

An association rule is an implication of the form \( X \Rightarrow I_j \), where \( X \) is a set of some items in \( I \), and \( I_j \) is a single item in \( I \) that is not present in \( X \). The rule \( X \Rightarrow I_j \) is satisfied in the set of transactions \( T \) with the confidence factor \( 0 \leq c \leq 1 \) if, and only if, at least \( c \% \) of transactions in \( T \) that satisfy \( X \) also satisfy \( I_j \). The notation \( X \Rightarrow I_j \mid c \) is used to specify that the rule \( X \Rightarrow I_j \) has a confidence factor of \( c \).

From a probabilistic point of view, support can be seen as the join probability of items in the consequent and the antecedent of a rule, i.e.

\[
\text{support}(R) = p(X \cup I_j)
\]

and confidence can be interpreted as the conditional probability of \( I_j \) having occurred \( X \) in the transaction, i.e.

\[
\text{confidence}(R) = p(I_j \subseteq T \mid X \subseteq T)
\]

In association rules, the support threshold describes the minimum percentage of transactions containing all items that appear in the rule, whereas the confidence threshold specifies the minimum probability for the rule consequent to be true if the antecedent is true. In the context of our technique the support and confidence of a candidate rule \( C \Rightarrow v \), where \( C \) denotes both concepts and \( v \) the associating verb, are calculated as follows:

\[
\text{support}(C \Rightarrow v) = \frac{|\{o \in \mathcal{O} | C \cup v \subseteq t\}|}{|\mathcal{O}|}
\]

\[
\text{confidence}(C \Rightarrow v) = \frac{\text{support}(C \cup v)}{\text{support}(C)}
\]

In this context, a transaction in \( T \) represents the occurrence of a pair of concepts with some linking verb in the text corpus (i.e. an element of \( R \)). For example, given the concepts Lipoygenase (Li) and Reactive Oxygen Intermediates (ROI) and the verb Activate (Ac), the transaction \( t = \langle Li, ROI, Ac \rangle \) represents the fact that the three of them appear together in one of the documents of the collection. If the rule \( \langle Li, ROI \rangle \Rightarrow \langle Ac \rangle \) is supported by the data, the strength
of the association of both concepts with the label will be given by the rule confidence.

The problem of discovering association rules can be decomposed into two sub-problems [1]:

1. Find all sets of items (itemsets) that have transaction support above minimum support. The support for an itemset is the number of transactions that contain the itemset. Itemsets with minimum support are called large itemsets, and all others small itemsets.

2. Use the large itemsets to generate the desired rules. A straightforward algorithm for this task can be defined as follows. For every large itemset \( l \), find all non-empty subsets of \( l \). For every such subset \( a \), output a rule of the form \( a \Rightarrow (l - a) \) if the ratio of support \( (l) \) to support \( (a) \) is at least \( \minconf \). All subsets of \( l \) have to be considered to generate rules with multiple consequents.

Apriori algorithm [2] is used for mining association rules based on the statistical information of concepts and verbs that co-occur frequently within text documents. This algorithm makes multiple passes over transactions to determine all large itemsets and generates the candidate itemsets to be counted in each pass by using only the itemsets found large in the previous pass. This allows the algorithm to build large itemsets of increasing size by adding items to itemsets that are already found to be large. The result of this mining process is a set of association rules involving statistically significant non-taxonomic relationships between concepts from the concept hierarchy given as input (both part of the antecedent of the rule) and a verb that can be use for labeling this relationship (the consequent of the rule). In an ontology learning process, association rules are used to suggest the existence of a relationship between two given concepts if both concepts match the antecedent of at least one extracted rule. If this is the case, the knowledge engineer will receive one or more recommendations about suitable labels, being these recommended labels the consequents of the matching rules order by confidence.

3. Step-by-step Example

In this section we present an example of how a labeled relationship between two concepts is obtained to be suggested to a knowledge engineer constructing an ontology. First, a pair of concepts has to be selected from an existing concept hierarchy. For this example, we suppose the existence of two concepts named \( \text{lipoygenase} \) and \( \text{reactive oxygen intermediates} \). After that, we proceed to expand these concepts and, for example, we find that the concept \( \text{reactive oxygen intermediates} \) has an abbreviation named \( \text{ROI} \). Clearly, the use of both the original terms and the abbreviation increases the matching rate when the search for these concepts in the text corpus takes place.

In a second stage, a search across the indexed corpus looking for sentences including both concepts is performed. In this way, we are able to extract and
analyze those verbs that most frequently occur in the context of the target pair of concepts. Let suppose that in this search we find the following sentence:

*Our data suggest that lipoygenase metabolites activate ROI formation which then induce IL-2 expression via NF-kappa B activation.*

The previously considered concepts lipoygenase and ROI appear in this sentence. Then, we have to determine which linking verb connects them. To do this, we parse the sentence using the mentioned POS tagger algorithm. The result of parsing the previous sentence is shown in Figure 2. The parsing tree shows that between these concepts there is a verb which in this case is “activate”. Consequently, we include this occurrence of a relationship between the two concepts in the set of occurrences or candidate relationships, being this occurrence composed of the following lexical items: *lipoygenase, activate, reactive oxygen intermediates*. The process continues by generating the set of all possible relationships linking both concepts until all verbs that appeared relating both concepts in the corpus have been identified.

Finally, we need to determine if the previously detected relationship has enough statistical significance to be considered as a feasible relationship that can
be suggested to a knowledge engineer with a certain degree of confidence. To accomplish this goal, we calculate the support and confidence of the corresponding association rule. If the rule exceeds the predefined support and confidence thresholds, it is suggested for further validation with domain experts.

4. Experimental Results

Evaluation of ontology learning techniques is recognized to be an open problem [27], which can be done by either evaluating the learning methods or the accuracy, efficiency and completeness of the built ontology. In addition, the unsupervised nature of ontology learning techniques makes evaluation even more difficult [16]. Even though the number of contributions in the area of ontology learning has significantly increased in recent years, especially because of its relation with the Semantic Web, the ontology community is still in the process of assessing an evaluation framework [20].

In this context, we propose the assessment of a quantitative performance measure for the proposed technique through the comparison of the extracted non-taxonomic relationships with the potential relationships existing in a manually-built reference ontology (i.e. an ontology manually built by domain experts). Thus, we inspect the results of our technique to determine whether each given triple is appropriate for describing a possible relationship existing in the domain. Clearly, this evaluation could be considered as centered on the precision of the technique because it measures the percentage of all relationships selected by the model that express a real association between a given pair of concepts.

Both a text corpus and a predefined ontology are needed to perform the evaluation. For conducting the experimental evaluation we selected the Genia corpus and its corresponding ontology[4]. On the one hand, Genia corpus is a collection of articles extracted from Medline database[5]. This corpus currently contains 1000 abstracts taken from the cited database, enclosing more than 400000 different words. On the other hand, Genia ontology presents a taxonomy of 47 biologically relevant nominal categories and comprises concepts related to gene expression and its regulation, including cell signaling reactions, proteins, DNA, and RNA [12].

The output of our method for extracting non-taxonomic relationships using both Genia corpus and ontology was set of 304 patterns, each consisting in a pair of concepts and its respective semantic relationship given by a verb. Out of these patterns, 77% (232) expressed valid biological relations, e.g. “Virus encode_individual protein”, whereas 23% (72) of the extracted relationships were incorrect, e.g. “Virus host_cells.corticosteroid tissue”. Figure 3 shows a screenshot of the output given by our technique once inserted in a ontology building environment in which the Genia corpus and ontology were used to extract relationships. A list of concept-concept-verb triples is shown ordered according

---

Figure 3: Example of assistance provided to knowledge engineers
to their confidence calculated using the Apriori algorithm for different concept pairs belonging to the Genius ontology. For the pair of concepts depicted in red in the concept map, for example, two different relationships are highlighted in the figure and can be selected by the knowledge engineer for enriching the ontology.

As the result of a meticulous examination of the sentences that lead to incorrect patterns, we identified two important factors causing recurring types of errors. First, some sentences contain POS-tagging mistakes. However, as a large corpus of documents is taken as input by our technique, candidate relationships produced because of part-of-speech tagger errors will not have enough frequency in the corpus to produce a confident association rule. Second, an important fraction of the error rate depends on \( n \), the chosen number of words between two concepts that is considered. If a small number for \( N \) is used, some significant relationships can be missed. Conversely, if a large scope of lexical items between two concepts is established, unconnected words will be possible added in the analysis and introduce noise in the association mining process. Therefore, we tested different values of \( N \) to determine the optimal number of words or window size in which two concepts have to appear in order to be considered as an occurrence of a certain relationship.

Experiments were performed with different values of \( N \) in order to determine a suitable range for this parameter. To this end, we used both the Genia corpus as its respective ontology through successive runs varying the number of words to analyze different window sizes for the appearance of concept pairs. Figures 4 and 5 depict the results obtained in these experiments, summarized in terms of true positives (TP), the proportion of relationships that were correctly identified, and false positives (FP), the proportion of relationships that were incorrectly classified as positive ones, according to the total number of discovered rules.

Figure 4 shows the number of discovered relationships for different values of \( N \). It is possible to observe in the figure that with a separation of \( N = 4 \) words between the occurrence of both concepts, the algorithm is capable of discovering 628 possible relationships out of which 485 were identified as correct ones, i.e. relationships that would have been resulted in good recommendations for an knowledge engineer to be consider for inclusion in the domain ontology. Afterwards, the number of discovered rules starts to decrease as the window size is widen.

Figure 5 shows the proportion of true and false positives in the discovered relationships for different values of \( N \). It worth noticing that \( N = 4 \) is not only the number of separating words that generates more candidate relationships (as it can be seen in Figure 4), but also this value allows the technique to reach a good compromise between true and false positives. Therefore, we can conclude that the most appropriate value for the number of separating words between concepts is \( N = 4 \) since it produces a considerable amount of candidate relationships to be included in the domain ontology and, consequently, allows the technique to contribute for the ontology completeness.

The final publication is available at http://dx.doi.org/10.1016/j.eswa.2009.01.048

Figure 4: Number of discovered rules according to the window size

Figure 5: Percentage of true and false positives in the discovered relationships
5. Related Work

Most of the work on learning of non-taxonomic relationships from textual sources combine different levels of statistical and linguistic analysis. In this context, association rule mining has been used for acquiring semantic, non-predefined relationships from texts. For instance, Maedche et al. [14, 15] use generalized association rules to identify co-occurrence between pairs of words to propose relationships at the appropriate level of a taxonomy. Even though these algorithms contribute to the ontology learning process, they do not address the problem of labeling relationships. As consequence of the lack of automatic labeling, knowledge engineers have to face this task without any assistance.

In other line of research, a number of works attempt to acquire information about semantic relations using regular expressions [7, 19]. In these approaches, an exhaustive inspection of text is made to identify instances of patterns that indicate a certain relation. A relation is recognized when a sequence of words in the text matches a pre-defined pattern. In relation to this line, numerous algorithms have been applied to extending ontologies with hypernymy and holonymy relationships [11, 3]. In these algorithms, the rules to generate relationships have to be created in advance, which forces the end user to have an extensive knowledge of the domain (examples of this approach can be find in the biomedical field in which large text collections are available [22]) or to construct patterns for well-known relationships such as hypernymy, meronymy, etc. (an example is an approach to find causal relations presented in [5]). In the same way, related approaches try to exploit hypernym, synonym and other structures given by WordNet lexical database. For example, within OntoLearn project [21] WordNet was used to automatically allocate relations considering a small predefined set of structures. Even though these approaches allow to identify an important number of relationships, they fail to capture further semantic relationships that might exist among concepts.

A growing number of researchers is focused on the management of syntactic structures inside a domain-specific text corpus based on natural language processing (NLP) techniques. Kavalec and Svátek [10] select verbs frequently occurring in the context of pairs of concepts for labeling semantic relationships. In this work, the association between concepts is assessed using a measure of the conditional frequency of a pair of concepts given a verb. In contrast to this approach, we use the strength of the association between concepts and verb given by the confidence of an extracted rule to suggest multiple labels for a established relationship between concepts. RefExt [26] is a system that is capable of automatically identifying highly relevant triples (pairs of concepts connected by a relation) over concepts from an existing ontology. This system uses a statistical measure of relevance to filter terms based on their observed frequencies in a domain specific corpus and then extract highly ranked verbs and nouns according to their co-occurrence score. In [25] an approach is proposed for learning non-taxonomic relationships starting from the discovery of domain-related verb phrases and using Web scale statistics to validate them instead of domain texts.
6. Conclusions

In this paper we have presented a technique that contributes to the process of semi-automatic ontology learning through the discovery and labeling of non-taxonomic relationships. This technique can be used not only to automatically discover relationships between pairs of concepts, but also to assign an appropriate label to these relationships. Essentially, the proposed technique exploits the syntactic structures and dependencies located in a domain-specific text corpus in the process of discovering and labeling relationships. To accomplish this goal, verbs are considered as candidate lexical items to express relationships between concept pairs belonging to an ontology. Experimental results have shown that an important percentage out of the total number of discovered relationships can be considered as valid ones and can result in valuable recommendations for end-users. Although more experimentation is needed, we can conclude that this technique can help to reduce the burden of knowledge engineers during the construction of a domain ontology.

References


