Discovery and Evaluation of Non-Taxonomic Relations in Domain Ontologies

Albert Weichselbraun*, Gerhard Wohlgenannt
Institute for Information Business
Vienna University of Economics and Business, Austria
E-mail: {aweichse,wohlg}@ai.wu.ac.at
*Corresponding author

Arno Scharl
Department of New Media Technology
MODUL University Vienna, Austria
E-mail: scharl@modul.ac.at

Michael Granitzer
Know-Center Graz, Austria
E-mail: mgrani@know-center.at

Thomas Neidhart, Andreas Juffinger
Knowledge Management Institute
Graz University of Technology, Austria
E-mail: {tneidhart,ajuffinger}@tugraz.at

Abstract: The identification and labelling of non-hierarchical relations are among the most challenging tasks in ontology learning. This paper describes a bottom-up approach for automatically suggesting ontology link types. The presented method extracts verb vectors from semantic relations identified in the domain corpus, aggregates them by computing centroids for known relation types and stores the centroids in a central Knowledge Base (KB). Comparing verb vectors extracted from unknown relations with the stored centroids yields link-type suggestions. Domain experts evaluate these suggestions, refining the KB and constantly improving the components accuracy. Using four sample ontologies on 'energy sources', this paper demonstrates how link-type suggestion aids the ontology design process. It also provides a statistical analysis on the accuracy and average ranking performance of Batch Learning (BL) vs. Online Learning (OL).

Keywords: ontology learning, ontology extension, link type detection, non-hierarchical relations, non-taxonomic relations, vector space model

Reference to this paper should be made as follows: Weichselbraun, A., Wohlgenannt, G., Scharl, A., Granitzer, M., Neidhart, T. and Juffinger, A. (2009) 'Discovery and Evaluation of Non-Hierarchical Relations in Domain Ontologies', International Journal of Metadata Semantics and Ontologies, 4(3):xyz-xyz.

1 INTRODUCTION

Ontologies are a cornerstone technology of the Semantic Web. By describing vocabularies and business processes, they provide the means for a common understanding among different stakeholder groups. In dynamic organizations, domain-specific knowledge and the structure of workflows evolve continually. This requires a dynamic ontology engineering process to update ontologies, describing the environment and its various elements. Automatic and semi-automatic ontology extension frameworks, such as the one presented by Liu et al. (2005), facilitate this process by identifying relevant concepts and taxonomic links but do not support the discovery of non-taxonomic ontology link types. These relations have to be labeled by human ontology engineers – a non-trivial task, since various relations among instances of the same general concept are possible (Kavalec and Spyns, 2005). Manual labeling of non-taxonomic relations poses a serious constraint on the ontology engineering process and restricts the applicability of ontologies in dynamic environ-

Copyright © 200x Inderscience Enterprises Ltd.

ments. To overcome this problem, this paper suggests an automated method for aiding ontology engineers in the discovery of non-taxonomic link types.

1.1 Ontology Link Type Discovery

According to Maedche et al. (2002), ontology learning comprises (i) *ontology extraction* concerned with the identification of concepts C, taxonomic relations H^C, non-taxonomic relations R, and Axioms A^O, and (ii) *ontology maintenance* covering ontology pruning and refinement. In regards to ontology extraction, the identification and labeling of non-taxonomic relations as well as the learning of axioms are considered most challenging (Kavalec and Spyns, 2005).

Maedche et al. (2002) discover non-taxonomic relations by the use of association rules without labeling them further. They also cover the handling of relations between instances of the same concept (e.g. two instances of the concept "person" cooperate with each other). Liu et al. (2005) combine Hearst patterns, head nouns, subsumption, co-occurrence analysis and WordNet Fellbaum (1998) in their approach towards ontology extension. Their method is capable of identifying hierarchical and unlabeled non-hierarchical relations.

Kavalec and Spyns (2005) present a method for the automated labeling of relations by extracting relevant lexical items (verbs, verb phrases) frequently co-occurring with concept associations. The authors evaluate their labels in the tourism domain (Lonely Planet)[1] and on semantically tagged corpora (SemCor)[2] against a pre-defined "gold standard"-ontology. They also do so with the help of domain experts who evaluate the correctness of divergent link types. A good overview of learning hierarchical relations from heterogeneous sources is provided by Cimiano et al. (2005).

Some of the techniques and ideas applied in hierarchical relation discovery have been extended to non-hierarchical relations. Berland and Charniak (1999), for example, have been able to adapt Hearst patterns (Hearst, 1992) for the identification of meronyms.

Sánchez and Moreno (2008) list other approaches for learning specific link types, such as *Qualia* (Cimiano and Wenderoth (2005)), *Telic* and *Agentive* (Yamada and Baldwin (2004)), and *Causation* (Girju and Moldovan (2002)). Poesio and Almuhareb (2005) present a method for determining combinations of these link types. All these techniques have a common link that they are based on linguistic patterns. Linguistic patterns are highly successful in specific applications, but lack the generic ability of adding new domain-specific relation types, which is a fundamental aspect of the research presented in this paper.

Sánchez and Moreno (2008) start the process of learning non-taxonomic relationships with the extraction of verbs from sentences that contain domain concepts and hyponyms of domain concepts. Those verbs are used to retrieve and select related concepts. The approach heavily depends on querying web search engines, which provide suggestions for new concepts as well as the verbs for relationship labeling. The search engines also help assess domain relevance by contrasting the number of hits for the individual verb with the number of hits for a combined query consisting of the verb and a domain keyword. In contrast to this approach, the method presented in this paper relies exclusively on a body of text to label unknown relations between concepts.

[1] www.lonelyplanet.com/

[2] www.cs.unt.edu/~rada/downloads.html#semcor

1.2 Paper Outline

The research presented in this paper focuses on adding link type discovery to a semi-automatic ontology extension architecture that builds domain specific ontologies based on a small seed ontology and a domain-specific corpus containing a large number of unstructured Web documents. Our approach distinguishes between taxonomic and non-taxonomic relations. It detects taxonomic relations by facilitating customized natural language processing techniques and databases. The non-taxonomic category is based on previously learned relations, assuming that similar relations between concepts are expressed via similar verbs. Comparing the vector space representation of verbs co-occurring with the target concepts to known verb-vectors using the cosine similarity metric yields the relation type of the unknown relation. Suggestions and evaluations from domain experts are fed back into the architecture adjusting its KB, leading to a constant improvement of the algorithm's accuracy.

This paper is organized as follows: Section 2 outlines the ontology extension system that identifies concepts and taxonomic relations. Section 3 extends the relation detection to non-taxonomic link types. Section 4 evaluates the link type suggestion architecture using different experimental setups. Section 5 covers ideas for future research. The paper concludes with a summary and outlook in Section 6.

2 ONTOLOGY EXTENSION AND TAXONOMIC LINKS

This section summarizes the set of methods used to semi-automatically build and extend ontologies. Initially, a small set of terms from domain experts or known ontology repositories is selected as a seed ontology. The seed ontology terms are then fed into the lexical analyzer, which distributes the input to different plugins for providing evidence sources.

The generated terms are then connected with the seed ontology terms via directed weighted links. Once a network of semantic associations is established, spreading activation identifies the most relevant terms and suggests their incorporation into the seed ontology. WordNet, head nouns and additional rounds of spreading activation help determine the new concepts' position within the ontology. Subsumption analysis (Sanderson and Croft, 1999), together with WordNet and head nouns, identify the type of semantic relations. For terms not confirmed automatically, domain experts are consulted, or another iteration of spreading activation over newly acquired terms is triggered to gather additional evidence. A detailed description of the architecture can be found in Liu et al. (2005).

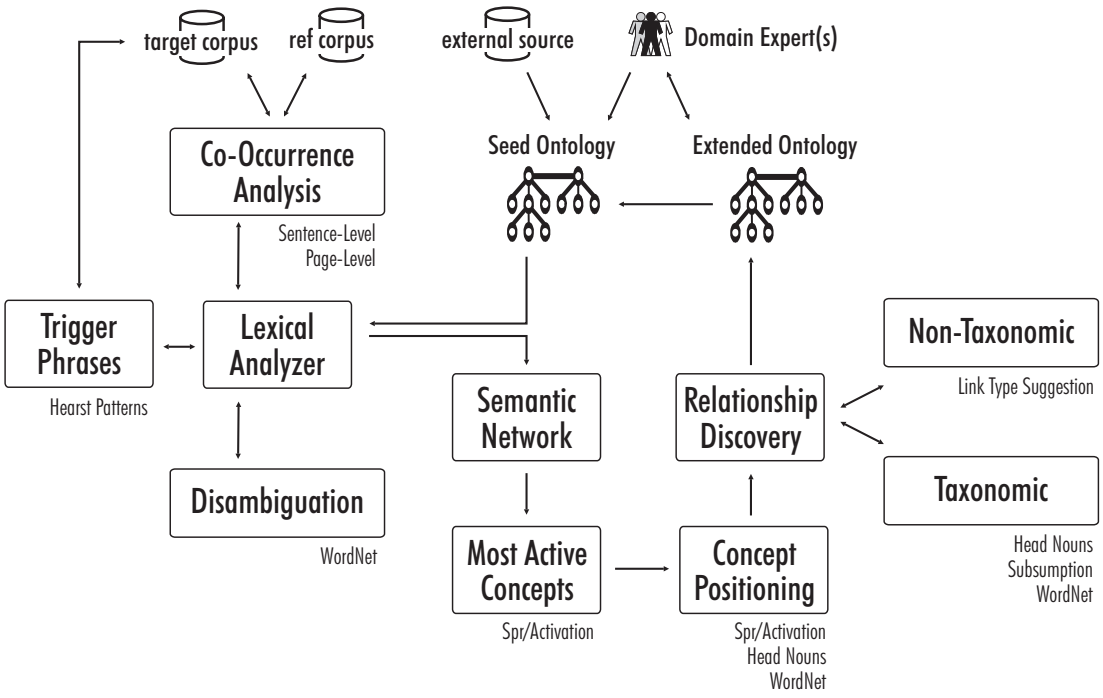

Figure 1: Ontology extension system architecture

2.1 Evidence Sources for Relevant Terms

Domain terminologies describe the "aboutness" of documents, i.e., the surface appearance of embedded concepts (Navigli and Velardi, 2004). Such terminologies may consist of unigrams such as *ice* or *water*, or n-grams such as *energy source* (noun compound) and *fossil fuel* (adjective-noun phrase). In the current architecture, three plugins garner candidate concepts from the domain corpus:

1. *Co-occurrence analysis* at both the sentence and the document level, limiting the influence of popular terms not related to the domain (Roussinov and Zhao, 2003). Specified via a threshold value on the co-occurrence significance, the plugin suggests 20 terms on the sentence level and 20 terms on the page level.

2. *Trigger phrases* matching a fragment of text that indicates a particular relation - e.g., parent-child (Joho et al., 2004).

3. *WordNet queries* (Fellbaum, 1998) after disambiguating the seed ontology concepts using a vector space model.

2.2 Selecting the Most Relevant Concepts and Weights

Spreading activation is a search technique inspired by the human brain's cognitive model where neurons fire activations to adjacent neurons. Connectionistic (as opposed to symbolic) artificial intelligence often uses spreading activation for retrieving hidden network information. Spreading activation is also widely used in associative information retrieval (Crestani, 1997). The spreading activation design involves the creation of a network data structure and the selection of the processing technique. The network structure typically consists of nodes connected by weighted links. The methods outlined in Section 2.1 generate candidate concepts for inclusion in the ontology. *Spreading activation* acts as the *glue* that combines the results of the various methods. Our approach builds the spreading activation network in two consecutive steps:

1. A semantic network is constructed using multiple evidence sources as input. Each term of the seed ontology is annotated via labeled, directed links that point to the candidate concepts and link metadata - e.g., the method's weight, the significance of result, etc.

2. The semantic network created in the first step is then converted into a spreading activation network, replacing the annotations between the concepts with weighted, directed links. Weights are calculated based on the link types, the weighting and significance data embedded into the link.

2.3 Concept Positioning

Statistical lexical analysis is often criticized as "knowledge poor" (Grefenstette and Hearst, 1992). Moving towards a detailed semantic analysis - e.g., determining the hierarchical relation of two terms - is far from trivial. The following sections review reported heuristics for identifying hypernyms and building concept hierarchies (Caraballo, 1999; Joho and Sanderson, 2000; Joho et al., 2004; Barriere, 2005) before describing the spreading activation approach. Figure 2 shows a seed ontology for the energy domain, which represents the basis for all subsequent computations.

Figure 1 presents a conceptual view on the system architecture of the ontology extension prototype. Positioning the most important terms - i.e., those highly relevant to the domain and seed ontology - is the most challenging task. Our approach uses

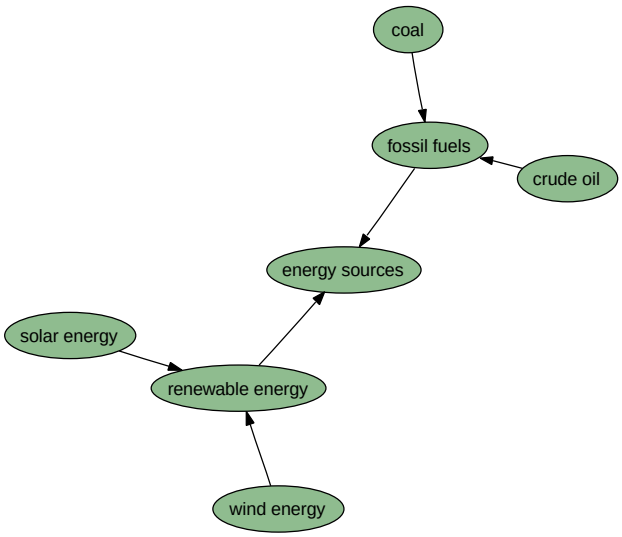

Figure 2: *Energy* seed ontology.

the following sequence: (i) accept semantic relations confirmed by WordNet and the head noun plugins; (ii) identify modifiers of a noun phrase that also appear in the activated list; (iii) trigger another round of spreading activation using the non-confirmed terms as seed terms to identify appropriate nodes for attaching these terms.

2.4 Discovering Taxonomic Link Types

The ontology extension architecture distinguishes between the discovery of taxonomic and non-taxonomic relations (see Section 3). The following steps identify taxonomic relations to be included in the domain ontology:

1. *Head noun analysis* adds terms that often subsume noun compounds to the network as potential hypernyms;

2. *WordNet hyponyms, hypernyms and synonyms* if both concepts are included in WordNet;

3. *Subsumption analysis*.

Subsumption analysis assumes that documents containing specific terms are a subset of the documents using general terms. According to Sanderson and Croft (1999), when considering two terms x and y, x is said to subsume y if the following condition holds:

$$\begin{aligned} P(x|y) &\geq 0.8 \qquad and \\ P(y|x) &< 1 \end{aligned}$$

Sanderson and Croft chose a value of 0.8 through informal analysis of hypo-/hypernym pairs identified through subsumption analysis in order to relax the initially strong condition $P(x|y) = 1$ (term x occurs whenever term y occurs).

Figure 3 shows the extended ontology after two iterations of spreading activation. The complexity of natural languages and the lack of contextual meaning in co-occurrence analysis inevitably lead to the inclusion of relevant but not hierarchically related terms. Unidentified relations are labeled (r), taking into account that hierarchical relations only represent a small subset of an ontology's possible relation types.

Unidentified relations (r) are candidates for the link type suggestion component capable of assigning labels (as for instance *effectOn*) to these relations. By verifying the proposed relation types, domain experts provide feedback for improving the method's accuracy in future iterations. The next section presents a more detailed description of the link type suggestion component.

3 NON-TAXONOMIC RELATIONS

Discovering ontology link types is closely related to methods that identify semantic relations in text corpora. The individual concepts and relation types occurring in semantic relations can be interpreted as instances of ontological classes and properties. Therefore, an aggregation of these individual appearances could provide valuable information regarding the relations in the domain ontology.

In contrast to the top-down approach presented by Dahab et al. (2008), which applies ontological relations to defining semantic patterns, this research uses on a bottom-up approach to identify ontological relation types by analyzing large repositories of domain-specific documents.

The Fourth International Workshop on Semantic Evaluations (SemEval 2007, previously known as SensEval)[3] competition reflects the growing importance of identifying semantic relations. The workshop included a task to classify semantic relations between nominals (Girju et al., 2007). The SemEval dataset contains 140 training and about 70 testing sentences for each of the seven given relation types with about 50% positive and 50% negative sentence classes. The sentences are tagged with nominals and the relation between those nominals. Additionally, WordNet sense keys for the nominals are provided, as well as the Google query used to collect training and target data. The SensEval competition for this task was subdivided into four categories, depending on whether or not the participants used the WordNet sense keys and Google query.

Among the participants with the best scores are Nakov and Hearst (2007), who use tailored Google queries to get a large set of verbs, prepositions and conjunctions appearing in sentences together with the target word pair. Together with the words from the sentence context, these features are then compared by similarity to features of the training word pairs using a variant of the Dice coefficient.

Giuliano et al. (2007) provide a kernel-based approach where the sources of information are represented by five basic kernel functions, which are linearly combined and weighted under different conditions.

Nicolae et al. (2007) only use the data that was provided in the task. They generate syntactic, semantic and lexical features, from which a number of models are built with the Weka data mining software (Witten and Frank (2005)). Among those models are decision trees, decision rules, logistic regression and

[3] nlp.cs.swarthmore.edu/semeval

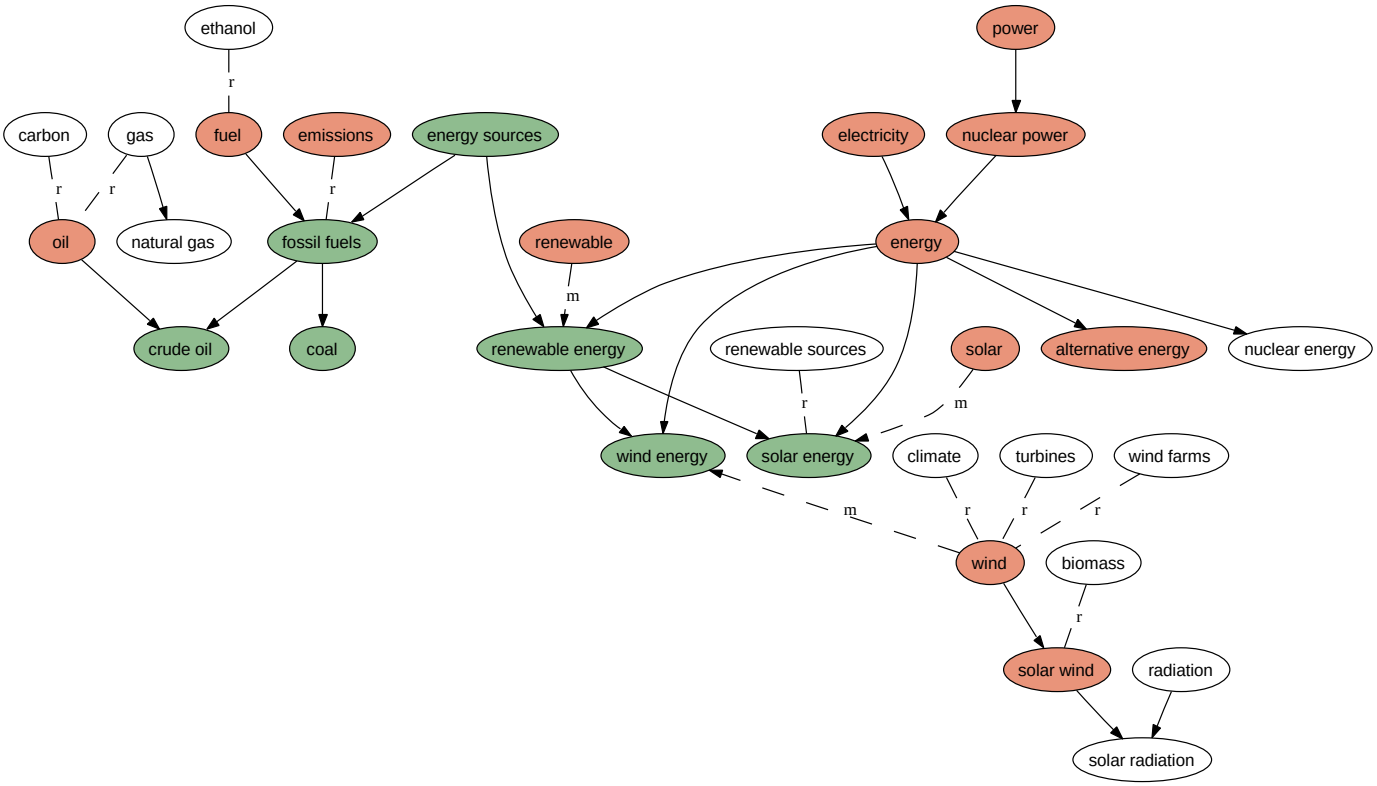

Figure 3: The extended *Energy* ontology after two extension rounds. The letter *"m"* denotes noun phrase modifier relations, whereas *"r"* marks unidentified relations.

"lazy" classifiers like k-nearest-neighbor. Weka performs a feature selection prior to the creation of the model. A voting mechanism decides upon the best fitting model for each subtask.

Section 2 presented an ontology extension architecture that is capable of identifying domain concepts and taxonomic relations in domain corpora based on a domain corpus and a seed ontology. Extending the relation discovery component outlined in Section 2 aims at providing a link type suggestion module for identifying arbitrary link types. The module is independent from the ontology extension architecture, but requires a domain corpus as well as an input ontology with labeled (optional) and unlabeled links.

Figure 4 illustrates the process. The link type suggestion component initially determines regular expressions for all domain concepts in the input ontology. Concepts retrieved from the seed ontology might have already been annotated with regular expressions confirmed by domain experts.

New concepts are automatically annotated with regular expressions, covering singular and plural forms as well as different notations for multi-term words (e.g. "solar energy", "solar-energy", etc.). The algorithm used for computing the plural/singular forms minimizes errors by combining grammatical rules with dictionaries.

Identifying sentences containing two concepts (C_m, C_n) from the input ontology and participating in a particular (unlabeled) relation $l_{mn}(C_m, C_n)$ yields sentences (s_i) containing semantic relations considering those two particular concepts. A Part-of-Speech (POS) tagger annotates these sentences to identify and extract embedded verbs.

A corpus-based normalization process converts all verb forms into the infinitive and transfers the derived terms into the vector space representation $v_i := verbs(s_i)$, describing the relation between the concepts involved. The similarity between the unknown relation's verb vector and the vectors stored for confirmed relations is computed and the relation type ($linktype_j$) of the most similar known relation (including link direction) is suggested to the domain expert.

3.1 Method

The link type suggestion component uses machine learning techniques to compile a KB of verb vectors from known relations. Consulting this KB yields suggestions for the link types of unknown relations. Below, we provide a formal description of this matching process.

Each concept (C) in the domain ontology is represented by a list of regular expressions (C^r) and connected to other concepts by labeled or unlabeled links $l_{mn}(C_m, C_n)$. Equation 1 gives the definition of the list of verb vectors L^v_{mn} that characterize the semantic relation between the concepts C_m and C_n.

$$L^v_{mn} = \{verbs(s_i) \quad | \quad match(C^r_m, s_i) \wedge match(C^r_n, s_i) \quad (1)$$
$$\wedge \quad idx(C^r_m, s_i) < idx(C^r_n, s_i) \quad \}$$

L^v_{mn} is composed of the vector space representation $\vec{v}_i := verbs(s_i)$ of verbs (Salton et al., 1975) occurring in a sentence s_i together with the domain concepts C_m and C_n. The *match* operators return true if sentence s_i matches at least one of the regular expressions in the list C^r.

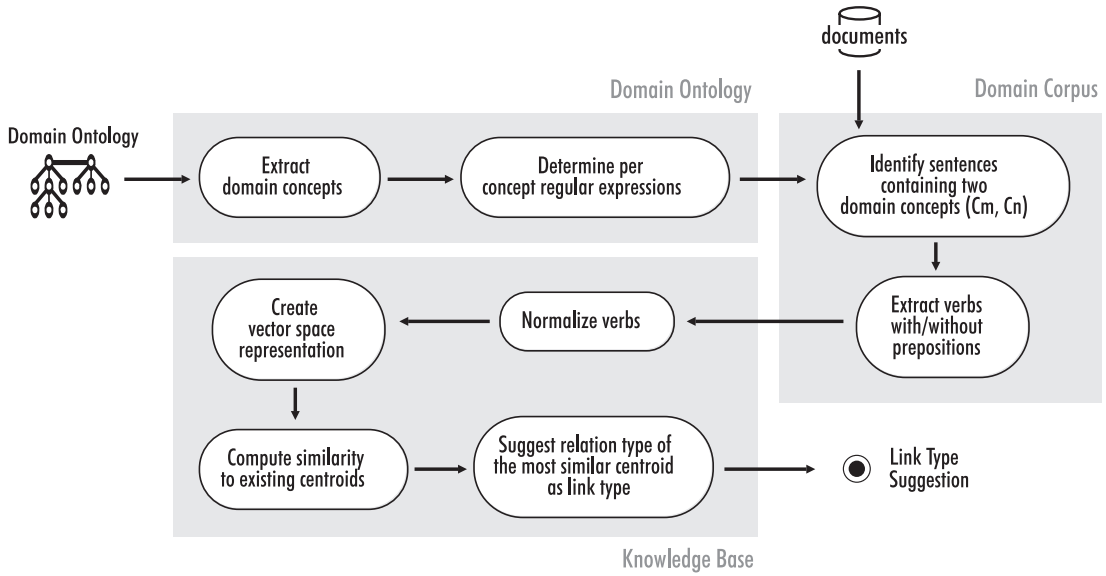

Figure 4: Architecture of the link type suggestion component for non-taxonomic relations.

The *verbs(s_i)* operator returns a vector space representation of the infinitive form of all verbs present in sentence s_i. In some cases, the use of prepositions changes the direction or even the meaning of relations (e.g. *deal in* versus *deal with*). For assessing the effect of prepositions on the method's accuracy we compiled two Knowledge Bases (KB, KB') that support two different *verbs(s_i)* functions. The first KB solely considers verbs. KB' stores verbs and prepositions (if available) for the suggestion process. The evaluation in Section 4 provides a comparison of the average ranking performance of link types computed with these approaches.

The order of the concepts is important for the evaluation process. We define that $l_{mn}(C_m, C_n) := \neg l_{nm}(C_n, C_m)$, which effectively reverses the direction of a relation. The *idx* operator in the second term of the definition ensures that the first concept (C_m) occurs before the second concept (C_n). Table 1 illustrates the relevance of concept order for the relation type.

Equation 2 computes the list of verb vectors L_j^v from relations with a known link type j. The $linktype_j$ operator returns $true$ if the concepts C_m and C_n are connected by a link of type j. These vectors L_j^v are merged to the centroid $\vec{V}_j$ (Equation 3).

$$L_j^v = \{\cup L_{mn}^v | linktype_j(C_m, C_n) = true\} \quad (2)$$

$$\vec{V}_j = \sum_{i=1}^{|L_j^v|} \vec{v}_i / \Big| \sum_{i=1}^{|L_j^v|} \vec{v}_i \Big| \quad \text{for all } \vec{v}_i \in L_j^v. \quad (3)$$

The set S_j of all known link types j and the associated centroids $\vec{V}_j$ form the KB of the link suggestion component.

$$KB = (S_j, \{\vec{V}_j | j \in S_j\}) \quad (4)$$

The types of unknown links l_{mn} are determined by computing the centroid $\vec{V}_{mn}$ (Equation 5) of their verb vector list L_{mn}^v and comparing the vector against the centroids $\vec{V}_j$ from the KB using the cosine similarity measure. The link type of the most similar centroid is then suggested for the unknown relation.

$$\vec{V}_{mn} = \sum_{i=1}^{|L_{mn}^v|} \vec{v}_i / \Big| \sum_{i=1}^{|L_{mn}^v|} \vec{v}_i \Big| \quad \text{for all } \vec{v}_i \in L_{mn}^v. \quad (5)$$

The link type suggestion component therefore derives the link type by comparing the centroid from (i) L_{mn}^v - a list of vector space representations of verbs appearing in semantic relations together with the domain concepts, with the centroid computed from (ii) L_j^v, which contains a list of vector space representations of verbs appearing in links of a particular type. The architecture assigns the relation type of the instances (and therefore verb vectors) which best fit with instances of unknown relation types.

3.2 User Feedback and Learning Process

The KB stores known link types from the domain ontology (i.e., all relations contained in the seed ontology), including those confirmed by domain experts. The system presents suggestions for unknown link types to the domain experts who either confirm or discard the suggested relations.

User feedback, which confirms *correctly* suggested links, is incorporated by adding the verbs of the verb vector list L_{mn}^v to the matching list L_j^v. A refined verb vector $\tilde{L}_j^v$ and the associated centroid $\vec{V}_j'$ are then added to the Knowledge Base KB'.

$$\tilde{L}_j^v = L_j^v \cup L_{mn}^v \quad (6)$$

The feedback algorithm accounts for *incorrect* suggestions by adding L_{mn}^v to the correct link type $L_{j'}^v$ (which might be a current or a new one).

$$\tilde{L}_j^v = \begin{cases} L_{j'}^v \cup L_{mn}^v & \text{if } j' \in S_j; \\ L_{mn}^v & \text{otherwise.} \end{cases} \quad (7)$$

Therefore, feedback of domain experts refines the KB and constantly improves the component's accuracy. Storing the correct

Text	Verbs	Relation
energy resources *deployed* like coal	deploy	¬subClassOf(energy resources, coal)
coal *is* an important energy resource	be	subClassOf(coal, energy resource)
climate *is influenced* by emissions	be, influence	¬effectOn(climate, emissions)
emissions *change* the climate	change	effectOn(emissions, climate)

Table 1: Discovery of relations in free text

relations and the appropriate verb vectors in the refined KB allows identifying link types more accurately in succeeding runs.

4 EVALUATION

This section summarizes a series of experiments conducted to evaluate the performance of the outlined method, comparing two training strategies that are based on four input ontologies and corpora:

(i) *Batch learning (BL)* - the KB is pre-trained with domain specific relations (see Section 4.1); all links are evaluated at once.

(ii) *Online learning (OL)* - in addition to pre-training, suggested links are immediately verified by a domain expert. The information from the verification process is fed back to the learning algorithm, yielding an improved KB for the suggestion of the next link types.

All tests have been performed evaluating vector space representations ($\vec{v}_i$) from verbs appearing together with the concepts C_m and C_n in (i) the same sentence, and (ii) within a sliding window size of five, six, and seven words.

To assess a preposition's influence on the relationship suggestion, we performed experiments considering prepositions and compared them with computations neglecting prepositions.

4.1 Experimental Setup

For our evaluation, we drew upon a list of 156 news media sites based on the Newslink.org, Kidon.com and ABYZNewsLinks.com directories. The webLyzard suite of Web mining tools[4] crawled these sites to generate four corpora between November 2005 and August 2006, each comprising about 200,000 documents.

Table 2 lists the link types used for labeling unknown relations and the number of sentences in the corpora satisfying Equation 1 (Section 3) from which verb vectors for that particular link type could be extracted.

The link suggestion uses a total of 25,207 sentences from the corpus for its evaluation, 10,215 of which are unique. The mirroring process does not only capture the latest publications but also news archives, which results in a high number of redundant sentences. Common page elements like disclaimers, copyright notes, etc. also contribute to this redudancy. Therefore, the link suggestion component only considers unique sentences.

Applying the ontology extension architecture described in Section 2 to an *energy* seed ontology (comprising seven hierarchically linked concepts) yields four extended versions of the ontology, each representing the knowledge contained in one of the corpora gathered between November 2005 and August 2006. Individually, these extended ontologies are too small for evaluating the link suggestion component (each one comprises only between 17 and 30 concepts). We therefore combine them to create an integrated ontology with a total of 102 links. Some of the links in the input ontology have already been classified by the previous taxonomic link discovery component as hierarchical (*isA*) or modifiers (*modifies*). There are 27 links which are unlabeled. Removing unrelated concepts reduces this number to 17 unknown links with no further overlap. These links are used to evaluate the link type suggestion component.

linkType	¬ linkType	$\text{sentences}_{unique}$
subClassOf	superClassOf	4411
use	usedBy	1688
hasEffectOn	isAffectedBy	3483
oppositeOf	oppositeOf	633

Table 2: Link types used in the evaluation

4.2 Training Sample

The KB is trained with 15 pre-defined domain specific learning patterns per link type, which are applied to the corpus extracting verb vectors appearing together with the concepts in the learning patterns. Table 3 shows three examples of learning patterns used for the training of the link type suggestion architecture. The training yields a KB with 451 verb vectors.

C_m^r	linkType	C_n^r
{coal}	subClassOf	{energy sources?}
{motors?}	use	{petrol, gasoline}
{oils?}	oppositeOf	{renewables?}

Table 3: Example training patterns

4.3 Results

In the experiments, the link type suggestion component assigns one of the link types in Table 2 to unknown relations. The evaluation distinguishes between results derived from batch learning (BL) versus online learning (OL). Due to the included feedback mechanism, online learning tends to deliver better results than batch learning.

For rows marked with "dir", the link type *and* direction have been computed. For rows identified by the term "nodir", only the correct link type has been suggested. The average ranking precision (ARP) for randomly chosen link types is 2.5 for

[4] www.weblyzard.com/

guessing the correct link type and 4.0 for picking the right link type and direction. Using verbs from whole sentences outperformed approaches based on sliding windows for links where link direction was not taken into account, as directed links sliding windows yielded better results.

Table 4 summarizes the different approaches' ARP, specifying the average number of tries required to pick the correct link type from an ordered list of suggestions. This measure is highly relevant, as the ontology link type suggestion has been designed to aid the domain expert in assigning links types. The ARP indicates how many choices the domain expert has to check on average in order to identify the correct label.

	verbs only		verbs and prepositions	
	sliding[5]	sentence	sliding[3]	sentence
dir BL	3.4	3.9	3.3	3.5
dir OL	3.3	3.6	3.3	3.3
nodir BL	2.0	1.8	2.0	1.9
nodir OL	2.0	1.8	1.9	1.8

Table 4: Average Ranking Precision (ARP)

Retrieving the verb vectors on a whole sentence level yields the best results with an average precision of 1.8 for ranking relations considering verbs only. The inclusion of prepositions into the link suggestion in most cases improves the results for the suggestion of link type and direction, reaching an average ranking performance of 3.3 when a sliding window is used.

Table 5 summarizes the results as a percentage of correctly identified link types. The "1st guess correct" column shows the percentage of relations correctly identified by the first suggestion. The "2nd guess" column gives the percentage of relations correctly labeled by the first or second suggestion.

	1st guess correct (%)		2nd guess correct (%)	
	sliding	sentence	sliding	sentence
dir BL	30.0	31.0	41.4	51.7
dir OL	34.5	32.9	41.2	50.0
nodir BL	44.3	55.7	64.6	71.1
nodir OL	47.1	44.1	72.4	79.4

Table 5: Percentage of correctly identified link types in the evaluation (sliding window size of seven words)

Obviously, it is much harder to guess link type *and* link direction (seven possibilities and a probability of approximately 14% to randomly guess the correct type)[6] than to only guess the link label (four possibilities and a probability of 25% of randomly choosing the correct type). Conducting a Chi-squared test on the results presented in Table 5 shows that the method is particularly successful on first guesses of link types, where significance levels are between 90% and 95%. The second guess does not add much additional benefit, presumably caused by the small search space of only four (seven) link types. The accuracy of 55.7% (79.4%) for determining the link type's relation in Table 5 is equivalent to an F-measure[7] of 0.72 (0.89) when retrieving link types only.

Schutz and Buitelaar (2005) point out that evaluating and comparing the performance of ontology learning approaches still poses a serious challenge. This is particularly true for learning non-taxonomic relationships. The tasks tackled range from the mere detection of unnamed non-taxonomic relations between concepts to labeling with a specific set of relationship types, and to arbitrary labels – with several variantion. Some works concentrate on specific domains, such as the biomedical domain, while others have a more general focus. Automatic evaluation against gold standards does not seem feasible, as they rarely include non-taxonomic relations. Sánchez and Moreno (2008) propose a method to automatically evaluate relations via WordNet (Fellbaum (1998)) similarity measures, which suffers from the problem (amongst others) that lexical entries for both concepts have to be contained in WordNet.

The comparison of F-measures between different approaches is not straightforward, because the methods often differ fundamentally in their capabilities and the way the evaluation has been performed. Kavalec and Spyns (2005), for instance, reach an F-measure of approximately 0.63. Their method does not consider link direction but is not limited to a predefined set of relation types and can therefore be used to extract arbitrary relations. Other methods, such as the one presented by Finkelstein-Landau and Morin (1999), do not provide a formal evaluation of their accuracy.

5 FUTURE RESEARCH

Future versions of the link type suggestion component will integrate domain knowledge into the suggestion process, especially regarding the domain and range of the link type. Annotation modules based on named entity recognition annotate the identified concepts (C_m, C_n) with concept types such as organization, person and country. Querying third-party resources via SPARQL, REST or SOAP queries like DBpedia[8] and Freebase[9] yield structured information to describe the identified concepts. The refined link type suggestion component will provide domain experts with a user-friendly means for specifying ontological knowledge via link type-specific *link description ontologies* using OWL Lite. Matching these link-type specific description ontologies with concept annotations allows removing or penalizing invalid link types.

Figure 5 exemplifies this process. After identifying a link between two concepts, the refined architecture annotates these concepts with their respective types using named entity recognition. Based on the link type domain (*rdfs:domain*) and range (*rdfs:range*) specifications of the domain experts in the link description ontology, the link type suggestion component reduces the number of possible matches by eliminating incompatible link types like *hasExperience* and *locatedIn* .

The increased availability of structured information sources puts additional emphasis on integrating and resolving conflicts

[5] computed with a sliding window size of seven words.

[6] compare Table 2 (the *oppositeOf* link type is symmetric).

[7] The F-measure is the weighted harmonic mean of precision and recall.

[8] www.dbpedia.org

[9] www.freebase.com

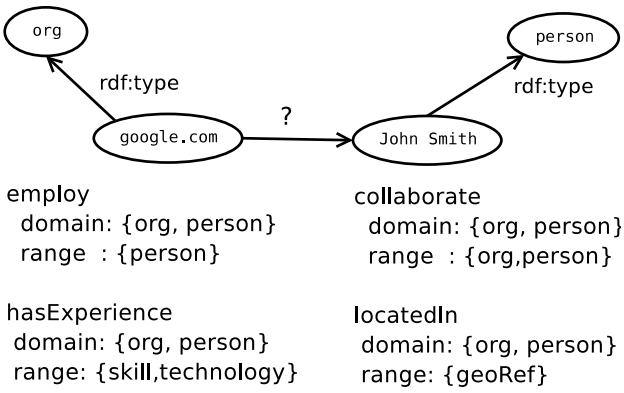

Figure 5: Considering lightweight domain ontologies for suggesting link types.

between annotations provided by such sources. Disambiguation and mediation techniques are a cornerstone for addressing this challenge and providing a more fine grained and accurate link type assessment.

6 CONCLUSIONS

This paper presents a novel approach for suggesting ontology link types by extracting verb vectors from sentences containing domain concepts and computing centroids assigned to known link types. Both the centroids and their link types are stored in a KB. Unknown relations are labeled by comparing their centroids with these KBs. Domain experts verify the labels, thereby providing important feedback for refining the KB.

The main contributions of this research are: (i) introducing a novel method for suggesting ontology link types based on domain knowledge extracted from a text corpus, (ii) integrating this approach into an existing semi-automatic ontology extension architecture, and (iii) evaluating the method's usefulness in labeling unknown link types.

One of the key success factors for suggesting correct link types is the choice of general training patterns used to compose the KB. Further research will focus on optimizing training patterns and strategies. Improving the accuracy of the suggested link direction is another promising research avenue. Possible approaches include the optimization of the training set, a more advanced parsing of the sentences' grammatical structure (e.g. detection of passive forms), and the consideration of additional discriminators (e.g. phrases). In addition to the extensions outlined in Section 5, future research will focus on performing large-scale empirical validations by applying this method to the suggestion of Wikipedia[10] link types. Evaluations with optimized, pre-learned and domain-specific KBs will further improve the architecture's predictive capabilities.

[10]www.wikipedia.org

ACKNOWLEDGMENT

The AVALON and IDIOM projects underlying this research are funded by the Austrian Ministry of Transport, Innovation & Technology and the Austrian Research Promotion Agency within the strategic objective FIT-IT Semantic Systems (www.fit-it.at). The authors would like to thank Arinya Eller for proofreading the manuscript, and the anonymous reviewers for their valuable feedback and suggestions during the preparation of this article.

REFERENCES

Barriere, C. (2005). Building a concept hierarchy from corpus analysis. *Terminology*, 10(2):241–263.

Berland, M. and Charniak, E. (1999). Finding parts in very large corpora. In *ACL'99*.

Caraballo, S. A. (1999). Automatic acquisition of a hypernym-labeled noun hierarchy from text. In *37th Annual Meeting of the Association for Computational Linguistics*, pages 120–126, Maryland, USA.

Cimiano, P., Pivk, A., Schmidt-Thieme, L., and Staab, S. (2005). *Ontology Learning from Text*, chapter Learning Taxonomic Relations from Heterogeneous Sources of Evidence, pages 59–76. IOS Press, Amsterdam.

Cimiano, P. and Wenderoth, J. (2005). Automatically learning qualia structures from the web. In *Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition*, pages 28–37, Ann Arbor, Michigan. Association for Computational Linguistics.

Crestani, F. (1997). Application of spreading activation techniques in information retrieval. *Artificial Intelligence Review*, 11:453–482.

Dahab, M. Y., Hassan, H. A., and Rafea, A. (2008). Textontoex: Automatic ontology construction from natural english text. *Expert Syst. Appl.*, 34(2):1474–1480.

Fellbaum, C. (1998). Wordnet an electronic lexical database. *Computational Linguistics*, 25(2):292–296.

Finkelstein-Landau, M. and Morin, E. (1999). Extracting semantic relationships between terms: Supervised vs. unsupervised methods. In *International Workshop on Ontological Engineering on the Global Information Infrastructure*, pages 71–80, Dagstuhl.

Girju, R. and Moldovan, D. (2002). Text mining for causal relations. In *In Proceedings of the FLAIRS Conference*, pages 360–364, Richardson, Texas.

Girju, R., Nakov, P., Nastase, V., Szpakowicz, S., Turney, P., and Yuret, D. (2007). Semeval-2007 task 04: Classification of semantic relations between nominals. In *Proceedings of*

the Fourth International Workshop on Semantic Evaluations (SemEval-2007), pages 13–18, Prague, Czech Republic. Association for Computational Linguistics.

Giuliano, C., Lavelli, A., Pighin, D., and Romano, L. (2007). FBK-IRST: Kernel methods for semantic relation extraction. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 141–144, Prague, Czech Republic. Association for Computational Linguistics.

Grefenstette, G. and Hearst, M. A. (1992). A method for refining automatically-discovered lexical relations: Combining weak techniques for stronger results. In *AAAI Workshop on Statistically-based Natural Language Programming Techniques*, pages 64–72, Menlo Park, CA. AAAI Press.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *COLING'92*, pages 539–545.

Joho, H. and Sanderson, M. (2000). Retrieving descriptive phrases from large amounts of free text. In *9th International Conference on Information and Knowledge Management*, pages 180–186, McLean, VA.

Joho, H., Sanderson, M., and Beaulieu, M. (2004). A study of user interaction with a concept-based interactive query expansion support tool. In *Advances in Information Retrieval, 26th European Conference on Information Retrieval*, pages 42–56, University of Sunderland, U.K.

Kavalec, M. and Spyns, P. (2005). *Ontology Learning from Text*, chapter A Study on Automated Relation Labelling in Ontology Learning, pages 44–58. IOS Press, Amsterdam.

Liu, W., Weichselbraun, A., Scharl, A., and Chang, E. (2005). Semi-automatic ontology extension using spreading activation. *Journal of Universal Knowledge Management*, 0(1):50–58. http://www.jukm.org/jukm_ 0_ 1/semi_ automatic_ ontology_ extension.

Maedche, A., Pekar, V., and Staab, S. (2002). Ontology learning part one - on discoverying taxonomic relations from the web. In Zhong, N., Liu, J., and Yao, Y., editors, *Web Intelligence*, pages 301–322. Springer.

Nakov, P. and Hearst, M. (2007). UCB system description for semeval task 4. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 13–18, Prague, Czech Republic. Association for Computational Linguistics.

Navigli, R. and Velardi, P. (2004). Learning domain ontologies from document warehouses and dedicated web sites. *Computational Linguistics*, 30(2):151–179.

Nicolae, C., Nicolae, G., and Harabagiu, S. (2007). UTD-HLT-CG: Semantic architecture for metonymy resolution and classification of nominal relations. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 454–459, Prague, Czech Republic. Association for Computational Linguistics.

Poesio, M. and Almuhareb, A. (2005). Identifying concept attributes using a classifier. In *Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition*, pages 18–27, Ann Arbor, Michigan. Association for Computational Linguistics.

Roussinov, D. and Zhao, J. L. (2003). Automatic discovery of similarity relationships through web mining. *Decision Support Systems*, 35:149–166.

Salton, G., Wong, A., and Yang, C. S. (1975). A vector space model for information retrieval. *Communications of the ACM*, 18(11):613–620.

Sánchez, D. and Moreno, A. (2008). Learning non-taxonomic relationships from web documents for domain ontology construction. *Data Knowl. Eng.*, 64(3):600–623.

Sanderson, M. and Croft, W. B. (1999). Deriving concept hierarchies from text. In *22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 206–213, Berkeley, USA.

Schutz, A. and Buitelaar, P. (2005). Relext: A tool for relation extraction from text in ontology extension. In *Proceedings of the 4th International Semantic Web Conference (ISWC 2005)*, pages 593–606. Galway, Ireland.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Series in Data Management Systems. Elsevier, Amsterdam, second edition.

Yamada, I. and Baldwin, T. (2004). Automatic discovery of telic and agentive roles from corpus data. In *Proceedings of the 18th Pacific Asia Conference on Language*, pages 115–141, Tokyo, Japan.