Discovering Conceptual Relations from Text

Alexander Maedche and **Steffen Staab** [1]

Abstract. Non-taxonomic relations between concepts appear as a major building block in common ontology definitions. In fact, their definition consumes much of the time needed for engineering an ontology. We here describe a new approach to discover non-taxonomic conceptual relations from text building on shallow text processing techniques. We use a generalized association rule algorithm that does not only detect relations between concepts, but also determines the *appropriate level of abstraction* at which to define relations. This is crucial for an appropriate ontology definition in order that it be succinct and conceptually adequate and, hence, easy to understand, maintain, and extend. We also perform an empirical evaluation of our approach with regard to a manually engineered ontology. For this purpose, we present a new paradigm suited to evaluate the degree to which relations that are learned match relations in a manually engineered ontology.

1 Introduction

Ontologies[2] have shown their usefulness in application areas such as intelligent information integration, information brokering and natural-language processing, to name but a few. However, their widespread usage is still hindered by ontology engineering being rather time-consuming and, hence, expensive.

Therefore a number of proposals have been made to facilitate ontological engineering through automatic discovery from domain data, domain-specific natural language texts in particular (cf. [2, 3, 4, 8, 10, 14]). Nevertheless, most of these approaches have only looked at how to learn the taxonomic part of ontologies. Common approaches collect relevant domain concepts and cluster them into a hierarchy using combinations of statistic and linguistic data. Though this already helps a lot, major efforts in ontology engineering must be dedicated to the definition of non-taxonomic *conceptual relationships*, e.g. hasPart relations between concepts.

The determination of non-taxonomic conceptual relationships is not this well-researched. In fact, it appears to be the more intricate task as, in general, it is less well known how many and what type of conceptual relationships should be modeled in a particular ontology. This paper presents an approach for discovering non-taxonomic conceptual relations from text and, hence, for facilitating this second part of ontology engineering. Building on the taxonomic part of the ontology, our approach analyzes domain-specific texts. It uses shallow text processing methods to identify linguistically related pairs of words (cf. Section 2). An algorithm for discovering generalized association rules analyzes statistical information about the linguistic output (cf. Section 3). Thereby, it uses the background knowledge from the taxonomy in order to propose relations at the appropriate level of abstraction. For instance, the linguistic processing may find that the word "costs" frequently co-occurs with each of the words "hotel", "guest house", and "youth hostel" in sentences such as (1).[3]

(1) Costs at the youth hostel amount to $ 20 per night.

From this statistical linguistic data our approach derives correlations at the conceptual level, viz. between the concept Costs and the concepts, Hotel, Guest House, and Youth Hostel. The discovery algorithm determines support and confidence measures for the relationships between these three pairs, as well as for relationships at higher levels of abstraction, such as between Accommodation and Costs. In a final step, the algorithm determines the level of abstraction most suited to describe the conceptual relationships by pruning appearingly less adequate ones. Here, the relation between Accommodation and Costs may be proposed for inclusion in the ontology. A more comprehensive example will be presented in Section 4.

We have evaluated our approach against an ontology about the tourism domain that we had modeled before using standard knowledge acquisition techniques. Linguistic processing was done on a text corpus extracted from a web site about tourist information. We have performed evaluation with regard to standard measures, however, we have also found that evaluation needs to take account of the sliding scale of adequacy prevalent in a hierarchical target structure. Thus, we have also conceived of a new evaluation measure to evaluate our experiments (cf. Section 5). Then, we conclude with a brief survey of related work and an outline of further work.

2 Shallow Text Processing

Our approach has been implemented on top of SMES (Saarbrücken Message Extraction System), a shallow text processor for German (cf. [9]) that has been adapted to the tourism domain (cf. [13] for a project description). This is a generic component that adheres to several principles that are crucial for our objectives. *(i)*, it is fast fast and robust, *(ii)*, it yields dependency relations between terms, and, *(iii)*, it returns pairs of concepts the coupling of which is motivated through *linguistic* constraints on the corresponding textual terms. In addition, we made some minor changes such that principle *(iv)*, linguistic processing delivering a high recall on the number of dependency relations occuring in a text, is also guaranteed. We here give a short survey on SMES in order provide the reader with a comprehensive picture of what underlies our evaluation.

The Architecture of SMES comprises a *tokenizer* based on regular expressions, a *lexical analysis* component including a word and a domain *lexicon*, and a *chunk parser*.

Tokenizer. Its main task is to scan the text in order to identify boundaries of words and complex expressions like "$20.00" or "Mecklenburg-Vorpommern"[4], and to expand abbreviations.

[1] Institute AIFB, University of Karlsruhe, 76128 Karlsruhe, Germany, email: {maedche,staab}@aifb.uni-karlsruhe.de, http://www.aifb.uni-karlsruhe.de/WBS

[2] We restrict our attention in this paper to *domain ontologies* that describe a particular small model of of the world as relevant to applications, in contrast to *top-level ontologies* and *representational ontologies* that aim at the description of generally applicable conceptual structures and meta-structures, respectively, and that are mostly based on philosophical and logical point of views rather than focused on applications.

[3] For ease of presentation we mostly give English examples, however, our evaluation is based on our implementation that processes German texts.

[4] Mecklenburg-Vorpommern is a region in the north east of Germany.

Lexicon. The lexicon contains more than 120.000 stem entries and more than 12,000 subcategorization frames describing information used for lexical analysis and chunk parsing. Furthermore, the domain-specific part of the lexicon associates word stems with concepts that are available in the concept taxonomy. In our case, we have defined a set of concepts, $C := \{a_i\}$, and the domain-specific part of the lexicon contains 486 entries referring to one of these concepts.

Lexical Analysis uses the lexicon to perform, *(1)*, morphological analysis, *i.e.*, the identification of the canonical common stem of a set of related word forms and the analysis of compounds, *(2)*, recognition of name entities, *(3)*, retrieval of domain-specific information, and, *(4)*, part-of-speech tagging:

1. In German compounds are extremely frequent and, hence, their analysis into their parts, e.g. "database" becoming "data" and "base", is crucial and may yield interesting relationships between concepts. Furthermore, morphological analysis returns possible readings for the words concerned, e.g. the noun and the verb reading for a word like "man" in "The old man the boats."
2. Processing of named entities includes the recognition of proper and company names like "Hotel Schwarzer Adler" as single, complex entities, as well as the recognition and transformation of complex time and date expressions into a canonical format, e.g. "January 1st, 2000" becomes "1/1/2000".
3. The next step associates single words or complex expressions with a concept from the ontology if a corresponding entry in the domain-specific part of the lexicon exists. E.g., the expression "Hotel Schwarzer Adler" is associated with the concept Hotel.
4. Finally, part-of-speech tagging disambiguates the reading returned from morphological analysis of words or complex expressions using the local context.

Chunk Parser. SMES uses weighted finite state transducers to efficiently process phrasal and sentential patterns. The parser works on the phrasal level, before it analyzes the overall sentence. Grammatical functions (such as subject, direct-object) are determined for each dependency-based sentential structure on the basis of subcategorizations frames in the lexicon.

Dependency Relations. Our primary output derived from SMES consists of *dependency relations* [6] found through lexical analysis (compound processing) and through parsing at the phrase and sentential level. It is important for our approach that on these levels syntactic dependency relations coincide rather closely with semantic relations that are often found to hold between the very same entities (cf. [5]). Thus, we derived our motivation to output those conceptual pairs to the learning algorithm the corresponding terms of which are dependentially related. Thereby, the grammatical dependency relation need not even hold directly between two conceptually meaningful entities. For instance, in (2) "Hotel Schwarzer Adler" and "Rostock", the concepts of which appear in the ontology as Hotel and City, respectively, are not directly connected by a dependency relation. However, the preposition "in" acts as a mediator that incurs the conceptual pairing of Hotel with City (cf. [11] for a complete survey of mediated conceptual relationships).

(2) The *Hotel Schwarzer Adler* in *Rostock* celebrates Christmas.

Heuristics. Chunk parsing such as performed by SMES still returns many phrasal entities that are not related within or across sentence boundaries. This however means that our approach would be doomed to miss many relations that often occur in the corpus, but that may not be detected due to the limited capabilities of SMES. For instance, it does not attach prepositional phrases in any way and it does not handle anaphora, to name but two desiderata. We have decided that we needed a high recall of the linguistic dependency relations involved, even if that would incur a loss of linguistic precision. The motivation is that with a low recall of dependency relations the subsequent algorithm may learn only very little, while with less precision the learning algorithm may still sort out part of the noise. Therefore, the SMES output has been extended to include heuristic correlations beside linguistics-based dependency relations:

- The *NP-PP-heuristic* attaches all prepositional phrases to adjacent noun phrases.
- The *sentence-heuristic* relates all concepts contained in one sentence if other criteria fail. This is a crude heuristic that needs further refinement. However, we found that it yielded many interesting relations, e.g. for enumerations, which could not be parsed successfully.
- The *title-heuristic* is very specific for our domain. It links the concepts such as referred to in the HTML title tags with all the concepts contained in the the overall document. This strategy might utterly fail in other domains, but it was successful for our hotel and sight descriptions.

To sum up, linguistic processing outputs a set of concept pairs, $CP := \{(a_{i,1}, a_{i,2}) | a_{i,j} \in C\}$. Their coupling is motivated through various direct and mediated linguistic constraints or by several general or domain-specific heuristic strategies.

3 Learning Algorithm

Our learning mechanism is based on the algorithm for discovering generalized association rules proposed by Srikant and Agrawal [12]. Their algorithm finds associations that occur between items, e.g. supermarket products, in a set of transactions, e.g. customers' purchases, and describes them at the appropriate level of abstraction, e.g. "snacks are purchased together with drinks" rather than "chips are purchased with beer" and "peanuts are purchased with soda".

The basic association rule algorithm is provided with a set of transactions $T := \{t_i | i = 1 \dots n\}$, where each transaction t_i consists of a set of items $t_i := \{a_{i,j} | j = 1 \dots m_i, a_{i,j} \in C\}$ and each item $a_{i,j}$ is from a set of concepts C. The algorithm computes *association rules* $X_k \Rightarrow Y_k$ ($X_k, Y_k \subset C, X_k \cap Y_k = \{\}$) such that measures for *support* and *confidence* exceed user-defined thresholds. Thereby, support of a rule $X_k \Rightarrow Y_k$ is the percentage of transactions that contain $X_k \cup Y_k$ as a subset, and confidence for $X_k \Rightarrow Y_k$ is defined as the percentage of transactions that Y_k is seen when X_k appears in a transaction, *viz.*

$$(3)\quad \text{support}(X_k \Rightarrow Y_k) = \frac{|\{t_i | X_k \cup Y_k \subseteq t_i\}|}{n}$$

$$(4)\quad \text{confidence}(X_k \Rightarrow Y_k) = \frac{|\{t_i | X_k \cup Y_k \subseteq t_i\}|}{|\{t_i | X_k \subseteq t_i\}|}$$

Srikant and Agrawal have extended this basic mechanism to determine associations at the right level of a *taxonomy*, formally given by a taxonomic relation $H \subset C \times C$. For this purpose, they first extend each transaction t_i to also include each ancestor of a particular item $a_{i,j}$, i.e. $t'_i := t_i \cup \{a_{i,l} | (a_{i,j}, a_{i,l}) \in H\}$. Then, they compute confidence and support for all possible association rules $X_k \Rightarrow Y_k$ where Y_k does not contain an ancestor of X_k as this would be a trivially valid association. Finally, they prune all those association rules $X_k \Rightarrow Y_k$ that are subsumed by an "ancestral" rule $\hat{X}_k \Rightarrow \hat{Y}_k$, the itemsets $\hat{X}_k, \hat{Y}_k$ of which only contain ancestors or identical items of their corresponding itemset in $X_k \Rightarrow Y_k$.

For the discovery of conceptual relations we may directly build on their scheme, as described in the following four steps that summarize our learning module:

1. Determine $T := \{\{a_{i,1}, a_{i,2}, \dots, a_{i,m'_i}\} | (a_{i,1}, a_{i,2}) \in CP \wedge l \geq 3 \rightarrow ((a_{i,1}, a_{i,l}) \in H \vee (a_{i,2}, a_{i,l}) \in H)\}$.
2. Determine support for all association rules $X_k \Rightarrow Y_k$, where $|X_k| = |Y_k| = 1$.

3. Determine confidence for all association rules $X_k \Rightarrow Y_k$ that exceed user-defined support in step 2.
4. Output association rules that exceed user-defined confidence in step 3 and that are not pruned by ancestral rules with higher or equal confidence and support.

The reader may note that we here have chosen a baseline approach considering the determination of the set of transactions T. Actually, one may conceive of many strategies that cluster multiple concept pairs into one transaction. For instance, given a set of 100 texts each describing a particular hotel in detail. Each hotel might come with an address, but it might also have an elaborate description of the different types of public and private rooms and their furnishing resulting in 10.000 concept pairs returned from linguistic processing. Our baseline choice considers each concept pair as a transaction. Then support for the rule {Hotel}⇒{Address} is equal or, much more probably, (far) less than 1%, while rules about rooms and their furnishing or their style, like {Room}⇒{Bed}, might achieve ratings of several percentage points. This means that an important relationship between {Hotel} and {Address} might get lost among other conceptual relationships. In contrast, if one considers complete texts to constitute transactions, an ideal linguistic processor might lead to more balanced support measures for {Hotel}⇒{Address} and {Room}⇒{Bed} of up to 100% each.

Thus, discovery might benefit when background knowledge about the domain texts is exploited for compiling transactions. In the future, we will have to further investigate the effects of different strategies.

4 Example

For the purpose of illustration, this chapter gives a comprehensive example, which is based on our actual experiments. We have processed a text corpus by a WWW provider for tourist information (URL: http://www.all-in-all.de). The corpus describes actual objects, like locations, accomodations, furnishings of accomodations, administrative information, or cultural events, such as given in the following example sentences.

(5) a. *Mecklenburg's* schönstes *Hotel* liegt in Rostock. (*Mecklenburg's* most beautiful *hotel* is located in Rostock.)

b. Ein besonderer Service für unsere Gäste ist der Frisörsalon in unserem Hotel. (A *hairdresser* in our *hotel* is a special service for our guests.)

c. Das Hotel Mercure hat *Balkone* mit direktem *Strandzugang*. (The hotel Mercure offers *balconies* with direct *access* to the beach.)

d. Alle *Zimmer* sind mit *TV*, Telefon, Modem und Minibar ausgestattet. (All *rooms* have *TV*, telephone, modem and minibar.)

Processing the example sentences (5a) and (5b), SMES (Section 2) outputs dependency relations between the terms, which are indicated in *slanted fonts* (and some more). In sentences (5c) and (5d) the heuristic for prepositional phrase-attachment and the sentence heuristic relate pairs of terms (marked by *slanted fonts*), respectively. Thus, four concept pairs – among many others – are derived with knowledge from the domain lexicon (cf. Table 1).

Table 1. Examples for linguistically related pairs of concepts

Term_1	$a_{i,1}$	Term_2	$a_{i,2}$
Mecklenburgs	area	*hotel*	hotel
hairdresser	hairdresser	*hotel*	hotel
balconies	balcony	*access*	access
room	room	*TV*	television

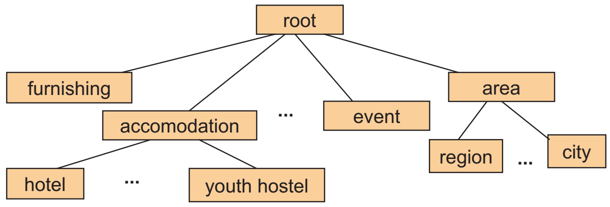

Figure 1. An example scenario

The algorithm for learning generalized association rules (cf. Section 3) uses the domain taxonomy, an excerpt of which is depicted in Figure 1, and the concept pairs from above (among many other concept pairs). In our actual experiments, it discovered a large number of interesting and important non-taxonomic conceptual relations. A few of them are listed in Table 2. Note that in this table we also list two conceptual pairs, viz. (area, hotel) and (room, television), that are not presented to the user, but that are pruned. The reason is that there are ancestral association rules, viz. (area, accomodation) and (room, furnishing), respectively with higher confidence and support measures.

Table 2. Examples of discovered relations

Discovered relation	Confidence	Support
(area, accomodation)	0.38	0.04
~~(area, hotel)~~	~~0.1~~	~~0.03~~
(room, furnishing)	0.39	0.03
~~(room, television)~~	~~0.29~~	~~0.02~~
(accomodation, address)	0.34	0.05
(restaurant, accomodation)	0.33	0.02

5 Evaluation

For our evaluation we analyzed 2234 HTML documents, 16 million words and HTML tags, from our text corpus (cf. Section 4) with SMES (Section 2). The linguistic and heuristic preprocessing came up with approx. 51.000 linguistically related pairs, such as the ones in Table 1. For our overall project we had modeled an ontology, which contained 284 concepts and 88 non-taxonomic conceptual relations. The ontology, $O := (C, H, R)$, served for two purposes. On the one hand, the taxonomic structure of concepts, C, of our domain ontology was given as an input, viz. as the taxonomic relation $H \subset C \times C$, to the learning algorithm described in Section 3. On the other hand, we evaluated the success of our learning approach against the set of non-taxonomic relations, $R \subset C \times C$, that had been hand-coded into the very same ontology before. Thus, we could compare the learning approach against human performance. Though human decisions in this matter should not be taken for pure gold[5], we think it is necessary to have measures that allow the comparison of different approaches and parameter settings — even when the bases of these measures depend to some extent on the quality of and on rather arbitrary, but equally plausible, choices between modeling decisions.

Precision and Recall. The first measures that we considered were *precision* and *recall* such as often used in information retrieval. When we denote the set of discovered relations by $D \subset C \times C$, they are defined by precision $:= |D \cap R|/|D|$ and recall $:= |D \cap R|/|R|$.

Running our experiments we found that precision and recall gave us some hints about how to gauge our thresholds for support and confidence (cf. Table 3). Nevertheless, these measures lacked a sense

[5] In fact, we are currently preparing an experiment. We want to determine the extent to which conceptual relations coincide when several ontology engineers introduce them independently from each other into a given taxonomy.

for the sliding scale of adequacy prevalent in our hierarchical target structures. To evaluate the quality of relations proposed to the ontology engineer, we also wanted to add some bonus to relations that *almost* fitted a hand-coded relation and, then, to compare different learning schemes on this basis. For this reason, we conceived of a new evaluation measure that reflected the distance between the automatically discovered relations D and the set of non-taxonomic, hand-coded relations R.

The **Generic Relation Learning Accuracy ($\overline{\textbf{RLA}}$)** is defined to capture intuitive notions for relation matches like "utterly wrong", "rather bad", "near miss" and "direct hit". $\overline{\text{RLA}}$ is the averaged accuracy that the instances d of discovered relations D match against their best counterparts from R — disregarding arbitrary relational directions.

(6) $\overline{\text{RLA}}(D, R) = \frac{1}{|D|} \sum_{d \in D} \text{RLA}(d, R)$.

(7) $\text{RLA}(d, R) = \max_{r \in R}\{\text{MA}(d, r), \text{MA}(d, r^{-1})\}$.

We determine the accuracy that two relations match, MA, based on the geometric mean value of how close their domain and range concepts match such as given by the conceptual learning accuracy CLA (Note that $\text{MA}(d, r) = \text{MA}((a_1, a_2), (b_1, b_2))$).[6]

(8) $\text{MA}((a_1, a_2), (b_1, b_2)) := \sqrt{\text{CLA}(a_1, b_1) \cdot \text{CLA}(a_2, b_2)}$.

CLA is very similar in style to learning accuracy as introduced by Hahn & Schnattinger [4] who evaluate the categorization of unknown objects in a taxonomy. However, they assume that the target concept that is to be learned is always a leaf concept and, hence, a categorization learned for an object may not be more specific than the correct categorization. In our approach this assumption does not hold, hence our CLA is a slight, symmetric variant of their measure. Basically, this accuracy measure reaches 100% when both concepts coincide (i.e., their distance $\delta(a, b)$ in the taxonomy H is 0); it degrades to the extent to which their distance increases; however, this degradation is seen as relative to the extent of their agreement such as given by the distance between their least common superconcept, lcs, and the top concept root.[7]

(9) $\text{CLA}(a, b) := \dfrac{\delta(\text{lcs}(a, b), \text{root})}{\delta(\text{lcs}(a, b), \text{root}) + \delta(a, b)} \in [0, 1]$.

The length of the shortest path $\delta(a_s, a_e)$ between a_s and a_e in the taxonomy H is defined via an auxiliary predicate Path that denotes all the valid paths in H.

(10) $\text{Path}(a_0, \ldots, a_n) \;:\Leftrightarrow\; \forall i \in 1 \ldots n : (a_{i-1}, a_i) \in H \cup (a_i, a_{i-1}) \in H$.

(11) $\delta(a_s, a_e) := \min\{n | a_1, ..., a_{n-1} \in C \wedge \text{Path}(a_s, a_1, ..., a_{n-1}, a_e)\}$.

The only restriction for CLA applies to extremely general relations that use the root concept in their domain or in their range. In our scenario, no such relation appeared in the hand-coded ontology O. Indeed, we found it appropriate to consider such relations as derived from noise that may easily be pruned.

Thus, $\overline{\text{RLA}}$ captures the fact that relations can be introduced at different levels of the taxonomy and that the quality of relations that are learned may vary within a range of degrees.

Example Evaluation. Figure 2 illustrates our definition of the generic relation learning accuracy with two small examples. On the left hand side of Figure 2 the relation that best matches $d' := (a'_1, a'_2)$ is $r' := (b'_1, b'_2)$. The distances between domain and range concepts count 1 each. The distances $\delta(\text{lcs}(a'_1, b'_1), \text{root})$ and $\delta(\text{lcs}(a'_2, b'_2), \text{root})$ counts 1 and 2, respectively. Hence, we compute

(12) $\text{RLA}(d', R) = \text{MA}(d', r') = \sqrt{\frac{1}{1+1} \cdot \frac{2}{2+1}} = \sqrt{\frac{1}{3}} \approx 0.58$.

Similarly, for $d'' := (a''_1, a''_2)$ and $r'' := (b''_1, b''_2)$.

(13) $\text{RLA}(d'', R) = \text{MA}(d'', r'') = \sqrt{\frac{1}{1+1} \cdot \frac{1}{1+2}} = \sqrt{\frac{1}{6}} \approx 0.41$.

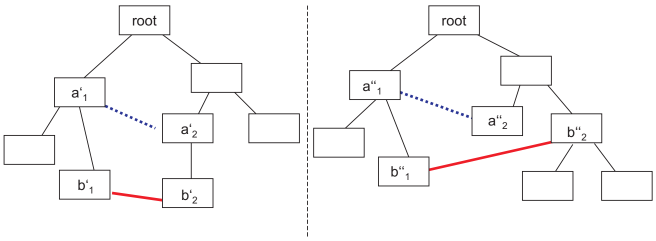

Figure 2. Relation Learning Accuracy

Results. An excerpt of our evaluation that surveys the most characteristic results is given in Table 3. We have computed the number of discovered relations D, $\overline{\text{RLA}}$, recall and precision for varying *support* and *confidence* thresholds. Calculating all relations using a support and confidence threshold of 0 yields 8058 relations, scoring a $\overline{\text{RLA}}$ of 0.51. As expected, both the number of discovered relations D and recall is decreasing with growing support and confidence thresholds. Precision is increasing monotonically at first, but it drops off when so few relations are discovered that almost no one is a direct hit. Higher support thresholds correspond to larger $\overline{\text{RLA}}$ values. Moving confidence thresholds from 0 to 1, $\overline{\text{RLA}}$ peaks between 0.1 and 0.2, but decreases thereafter. This behaviour may be due to our definition of transaction sets and will have to be further explored. The best $\overline{\text{RLA}}$ is reached using a support threshold of 0.04 and a confidence threshold of 0.01 and achieves 0.67 (better then example 12). This constellation also results in the best trade off between recall and precision (13% and 11%). The $\overline{\text{RLA}}$ value of 0.53 remains meaningful, even when recall and precision fall to 0%, due to a lack of exactly matching relations.

Table 3. Evaluation Results — number of discovered relations, $\overline{\text{RLA}}$, recall, precision

	Confidence			
Support	0.01	0.1	0.2	0.4
0.0001	2429 / 0.55	865 / 0.57	485 / 0.57	238 / 0.51
	66% / 2%	31% / 3%	18% / 3%	2% / 1%
0.0005	1544 / 0.57	651 / 0.59	380 / 0.58	198 / 0.5
	59% / 3%	30% / 4%	17% / 4%	1% / 1%
0.002	889 / 0.6	426 / 0.61	245 / 0.61	131 / 0.52
	47% / 5%	27% / 6%	16% / 6%	1% / 1%
0.01	342 / 0.64	225 / 0.64	143 / 0.64	74 / 0.53
	31% / 8%	19% / 8%	14% / 8%	1% / 1%
0.04	98 / **0.67**	96 / **0.67**	70 / 0.65	32 / 0.51
	13% / 11%	11% / 10%	6% / 7%	0% / 0%
0.06	56 / 0.63	56 / 0.63	48 / 0.62	30 / 0.53
	6% / 9%	6% / 9%	3% / 6%	0% / 0%

Standard deviation ranged between 0.22 and 0.32 in our experiments. Given that our average $\overline{\text{RLA}}$ scored well in the sixties, this means that we had a significant portion of bad guesses, but — what is more important — a large number of very good matches, too. Hence, we may infer that our approach is well-suited for integration into an interactive ontology editor. The reason is that an ontology engineer

[6] The geometric mean reflects the intuition that if either domain or range concepts utterly fail to match, the matching accuracy converges against 0, whereas the arithmetic mean value might still turn out a value of 0.5.

[7] Multiple inheritance may result in several least common superconcepts for a pair (a, b). Then we continue using the best value for CLA. All the other definitions remain applicable as they are stated here.

does not require near perfect discovery, but a restriction from a large number of relations, e.g. $283^2 = 80089$ (squared number of concepts leaving out root), to a selection, e.g. a few hundred, that contains a reasonable high percentage of good recommendations.

Random Choice. Finally, we have explored the significance of our $\overline{\text{RLA}}$ measure as compared against a uniform distribution of all possible, viz. 283^2, conceptual relations. The $\overline{\text{RLA}}$ computed from this set was 0.39 and, thus, significantly worse than learning results in our approach. Standard deviation achieved 0.17 and, thus, it was lower than for our discovery approach — the good match by random is indeed very rare. One may note that though the overall mean of 0.39 is still comparatively high (comparable to the one of example (13)), there are relations that score with the minimum, i.e. $\exists d \in D : \text{RLA}(d, R) = 0$, in our ontology.

6 Related Work

As mentioned before, most researchers in the area of discovering conceptual relations have "only" considered the learning of taxonomic relations. To mention but a few, we refer to some fairly recent work, e.g., by Hahn & Schnattinger [4] and Morin [8] who used lexico-syntactic patterns with and without background knowledge, respectively, in order to acquire taxonomic knowledge.

For purposes of natural language processing, several researchers [1, 10, 14] have looked into the acquisition of verb meaning, subcategorizations of verb frames in particular. Resnik [10] has done some of the earliest work in this category. His model is based on the distribution of predicates and their arguments in order to find selectional constraints and, hence, to reject semantically illegitimate propositions like "The number 2 is blue." His approach combines information-theoretic measures with background knowledge of a hierarchy given by the WordNet taxonomy. He is able to partially account for the appropriate level of relations within the taxonomy by trading off a marginal class probability against a conditional class probability, but he does not give any application-independent evaluation measures for his approach. He considers the question of finding appropriate levels of generalization within a taxonomy to be very intriguing and concedes that further research is required on this topic (cf. p. 123f in [10]) .

Faure and Nedellec [3] have presented an interactive machine learning system called ASIUM, which is able to acquire taxonomic relations and subcategorization frames of verbs based on syntactic input. The ASIUM system hierarchically clusters nouns based on the verbs that they co-occur with and *vice versa*.

Wiemer-Hastings *et al.* [14] aim beyond the learning of selectional constraints, as they report about inferring the meanings of unknown verbs from context. Using WordNet as background knowledge, their system, Camille, generates hypotheses for verb meanings from linguistic and conceptual evidence. A statistical analysis identifies relevant syntactic and semantic cues that characterize the semantic meaning of a verb, e.g. a terrorist actor and a human direct object are both diagnostic for the word "kidnap".

The proposal by Byrd and Ravin [2] comes closest to our own work. They extract named relations when they find particular syntactic patterns, such as an appositive phrase. They derive unnamed relations from concepts that co-occur by calculating the measure for mutual information between terms — rather similar as we do. Eventually, however, it is hard to assess their approach, as their description is rather high-level and lacks concise definitions.

To contrast our approach with the research just cited, we want to mention that all the verb-centered approaches may miss important conceptual relations not mediated by verbs. Regarding evaluation, they have only appealed to the intuition of the reader [2, 3] or used application-dependent evaluation measures. We have evaluated our approach in blind experiments using two standard and our original $\overline{\text{RLA}}$ measure. The latter has been thoroughly tested for plausibility and validated against the set of all possible relations.

7 Conclusion

We have presented an approach towards learning non-taxonomic conceptual relations from text. We have evaluated the approach on a set of real world texts against conceptual relations that had been modeled by hand. For this purpose, we used standard measures, viz. precision and recall, but we also developed an evaluation metrics that took into account the scales of adequacy prevalent in our target structures. The evaluation showed that though our approach is too weak for fully automatic discovery of non-taxonomic conceptual relations, it is highly adequate to help the ontology engineer with modeling the ontology through proposing conceptual relations. The presented approach has been integrated into the Ontology Learning Environment TextToOnto (cf. [7]) supporting the entire ontology engineering process from text.

Acknowledgments. The research presented in this paper has been partially funded by BMBF under grant number 01IN802 (project "GETESS"). We thank our student Raphael Volz who implemented large parts of the learning algorithm and our project partners, in particular Günter Neumann, from DFKI, language technology group, who supported us in using their SMES system.

References

[1] R. Basili, M. T. Pazienza, and P. Velardi, 'Acquisition of selectional patterns in a sublanguage', *Machine Translation*, **8**(1), 175–201, (1993).

[2] R. Byrd and Y. Ravin, 'Identifying and extracting relations from text', in *NLDB'99 — 4th International Conference on Applications of Natural Language to Information Systems*, (1999).

[3] D. Faure and C. Nedellec, 'A corpus-based conceptual clustering method for verb frames and ontology acquisition', in *LREC workshop on adapting lexical and corpus resources to sublanguages and applications*, Granada, Spain, (1998).

[4] U. Hahn and K. Schnattinger, 'Towards text knowledge engineering', in *Proc. of AAAI '98*, pp. 129–144, (1998).

[5] E. Hajicova, 'Linguistic meaning as related to syntax and to semantic interpretation', in *Language and Artificial Intelligence. Proceedings of an International Symposium on Language and Artificial Intelligence*, ed., M. Nagao, pp. 327–351, Amsterdam, (1987). North-Holland.

[6] R. Hudson, *English Word Grammar*, Basil Blackwell, Oxford, 1990.

[7] A. Maedche and S. Staab, 'Semi-automatic engineering of ontologies from text', in *Proceedings of the 12th Internal Conference on Software and Knowledge Engineering. Chicago, USA*. KSI, (2000).

[8] E. Morin, 'Automatic acquisition of semantic relations between terms from technical corpora', in *Proc. of the Fifth International Congress on Terminology and Knowledge Engineering - TKE'99*, (1999).

[9] G. Neumann, R. Backofen, J. Baur, M. Becker, and C. Braun, 'An information extraction core system for real world german text processing', in *ANLP'97 — Proceedings of the Conference on Applied Natural Language Processing*, pp. 208–215, Washington, USA, (1997).

[10] P. Resnik, *Selection and Information: A Class-based Approach to Lexical Relationships*, Ph.D. dissertation, University of Pennsylania, 1993.

[11] M. Romacker, K. Markert, and U. Hahn, 'Lean semantic interpretation', in *Proc. of IJCAI-99*, pp. 868–875, (1999).

[12] R. Srikant and R. Agrawal, 'Mining generalized association rules', in *Proc. of VLDB '95*, pp. 407–419, (1995).

[13] S. Staab, C. Braun, A. Düsterhöft, A. Heuer, M. Klettke, S. Melzig, G. Neumann, B. Prager, J. Pretzel, H.-P. Schnurr, R. Studer, H. Uszkoreit, and B. Wrenger, 'GETESS — searching the web exploiting german texts', in *CIA'99 — Proceedings of the 3rd Workshop on Cooperative Information Agents*, LNCS, Berlin, (1999). Springer.

[14] P. Wiemer-Hastings, A. Graesser, and K. Wiemer-Hastings, 'Inferring the meaning of verbs from context', in *Proceedings of the Twentieth Annual Conference of the Cognitive Science Society*, (1998).