Clustering Concept Hierarchies from Text

Philipp Cimiano, Andreas Hotho, Steffen Staab

Institute AIFB
University of Karlsruhe
{cimiano, hotho, staab}@aifb.uni-karlsruhe.de

Abstract

We present a novel approach to learning taxonomies or concept hierarchies from text. The approach is based on Formal Concept Analysis, a method mainly used for the analysis of data, i.e. for investigating and processing explicitly given information. Our approach is based on the distributional hypothesis, i.e. that nouns or terms are similar to the extent to which they share contexts. Further, we assume that verbs pose more or less strong selectional restrictions on their arguments. The concept hierarchy is built via Formal Concept Analysis using syntactic dependencies as attributes. The approach is evaluated by comparing the produced concept hierarchies against two handcrafted taxonomies from two different domains: tourism and finance. We compare the results of our approach against a hierarchical bottom-up clustering algorithm as well as against Bi-Section-Kmeans as an instance of a top-down clustering algorithm.

1. Introduction

Taxonomies or conceptual hierarchies are crucial for any knowledge-based system, i.e. any system making use of declarative knowledge about the domain it deals with. However, it is also well known that every knowledge-based system suffers from the so called *knowledge acquisition bottleneck*, i.e. the difficulty to actually model the knowledge relevant for the domain in question. In order to partially overcome this bottleneck, different methods have been proposed in the literature to address the problem of (semi-) automatically deriving a concept hierarchy from text. Basically, these methods can be grouped in two classes: the *similarity*-based methods on the one hand and the *set-theoretical* approaches on the other hand. Both methods adopt a vector-space model and represent a word or term as a vector containing features or attributes derived from a certain corpus. There is certainly a great divergence in which attributes are used for this purpose, but typically some sort of syntactic dependencies ares used such as conjunctions or appositives (Caraballo, 1999) or verb-argument dependencies (Faure and Nedellec, 1998; Hindle, 1990; Pereira et al., 1993). The first type of methods is characterized by the use of a similarity/distance measure in order to compute the pairwise similarity/distance between vectors corresponding to two words or terms in order to decide if they can be clustered together or not. Some prominent examples for this type of method are (Caraballo, 1999; Hindle, 1990; Faure and Nedellec, 1998; Pereira et al., 1993; Bisson et al., 2000). Set-theoretical approaches partially order the objects according to the inclusion relations between their attribute sets (Petersen, 2002; Sporleder, 2002). In this paper, we present a novel set-theoretical approach based on Formal Concept Analysis, a method mainly used for the analysis of data (Ganter and Wille, 1999). In order to derive attributes from a certain corpus, we parse it and extract verb/PP-complement, verb/object and verb/subject dependencies. For each noun appearing as head of these argument positions we then use the corresponding verbs as attributes. It is important to mention that in contrast to (Hindle, 1990) we count each syntactic position as a different attribute.

Furthermore, we directly compare the obtained results against a hierarchical bottom-up clustering algorithm and Bi-Section-Kmeans as an instance of a top-down clustering algorithm. In particular we show results about which syntactic-dependencies and thus which of the dependencies we consider seem to work best for the task at hand. The structure of the paper is as follows: Section 2. introduces Formal Concept Analysis as well as the other clustering approaches and describes the approach to learning concept hierarchies. Section 3. presents the text processing methods used to automatically derive features from the corpus. The approach is evaluated along the lines described in Section 4. and section 5. presents the concrete results. Finally, Section 6. discusses some related work and Section 7. concludes the paper.

2. Clustering Approaches

Formal Concept Analysis (FCA) is a method mainly used for the analysis of data, i.e. for investigating and processing explicitly given information. Such data are structured into units which are formal abstractions of concepts of human thought allowing meaningful comprehensible interpretation. The reader is referred to (Ganter and Wille, 1999) for the definitions of a *formal context*, *formal concept* and the subconcept-superconcept relation between formal concepts. In the approach presented in this paper we use Formal Concept Analysis as a conceptual clustering technique to automatically derive a partial order or concept hierarchy between terms on the basis of syntactic dependencies as features. For this we automatically derive the features for a certain term from the corpus, build the formal context as well as the corresponding lattice with the *Concepts*[1] tool and then transform the latter into a partial order. For a detailed description and some illustrations of our FCA-based approached the reader is referred to (Cimiano et al., 2003) and (Cimiano et al., 2004). Furthermore, in order to evaluate our FCA-based approach we compare it against hierarchical agglomerative clustering (Day and Edelsbrunner, 1984) as well as Bi-Section-Kmeans as as an instance of a divisive algorithm (Steinbach et al., 2000).

The task we are now focusing on is: given a certain num-

[1] see http://www.fcahome.org.uk/

ber of terms (concepts) relevant for the domain in question, can we derive a concept hierarchy between them? In terms of FCA, the objects are thus given and we need to find the corresponding attributes in order to build a formal context, a lattice and finally a partial order representing a concept hierarchy. In the following section we describe how the features are automatically acquired from the corpus.

3. Feature Extraction

A straightforward possibility is to consider as features or attributes certain syntactic dependencies such as verb/object, verb/subject and verb/PP-complement dependencies. In order to extract these dependencies, we make use of LoPar, a trainable and statistical left-corner parser (Schmid, 2000). In our approach, LoPar is thus first trained on the corpora before actually parsing them. LoPar's output is then post-processed with tgrep[2] to actually yield the desired dependencies, i.e. the verbs and the nominal heads of the object/subject/PP-complement they subcategorize. Regarding the output of the parser, it has to be taken into account that on the one hand it can be erroneous and on the other hand not all the verb/argument dependencies produced are significant from a statistical point of view. Thus an important issue is actually to reduce the 'noise' produced by the parser before feeding the output into the clustering algorithm. Now in order to weigh the significance of a certain verb-argument/term pair (v_{arg}, t), we used three different measures: a measure based on the conditional probability, the mutual information measure used in (Hindle, 1990), as well as a measure based on Resnik's *selectional preference strength* of a predicate (Resnik, 1997). Here are the formulas:

Conditional: $P(t|v_{arg})$
Hindle: $log\ \frac{P(v_{arg},t)}{P(v_{arg})\ P(t)}$
Resnik: $P(t|v_{arg}) * S_R(v_{arg})$

where the *selectional preference strength of a verb* is defined according to (Resnik, 1997):

$$S_R(v) = \sum_{t \in T} P(t|v_{arg})\ log\ \frac{P(t|v_{arg})}{P(t)}$$

Thus, the selectional preference of a verb position is stronger the less frequent the terms are that appear at this position. In our approach, we then only consider those verb-argument/term pairs (v_{arg}, t) as attribute/object pairs for which the values of the above measures are above some threshold t.

4. Evaluation

In order to evaluate our approach, we compare the automatically generated concept hierarchies with handcrafted ontologies for two different domains: tourism and finance. The ontology for the tourism domain is the reference ontology of the comparison study in (Maedche and Staab, 2002), which was modeled by an experienced ontology engineer. The finance ontology is basically the one developed within the GETESS project (Staab et al., 1999); it was designed for the purpose of analyzing German texts on the Web, but also english labels are available for many of the concepts. Moreover, we manually added the english labels for those concepts whose german label has an english counterpart with the result that most of the concepts (>95%) finally yielded also an english label.[3] The tourism domain ontology consists of 289 concepts, while the finance domain ontology is bigger with a total of 1178 concepts.
We compare two ontologies with each other as described in (Maedche and Staab, 2002) by comparing their lexical as well as taxonomic overlap. The core ontological model on which we base our evaluation is defined as follows:

Definition 1 (Core Ontology)
A core ontology is a structure $O := (C, \leq_C)$ *consisting of (i) a set* C *called concept identifiers, (ii) a partial order* $\leq_C$ *on* C *called concept hierarchy or taxonomy.*

As we injectively map terms onto concepts in all three clustering approaches, we neglect the fact that terms can be polysemous.[4] We calculate the lexical recall between two ontologies as follows: $LR'(O_1, O_2) = \frac{|C_1 \bigcap C_2'|}{|C_2'|}$

where C_2' is the set of terms/concepts in C_2 which also appear in the results of our syntactic dependency extraction process. The motivation here is not to penalize the system for not ordering terms which do not appear in the dataset.
In order to compare the taxonomy of the ontologies, we use the *semantic cotopy* (SC) presented in (Maedche and Staab, 2002). In particular we use a modified version SC' of the semantic cotopy in which we only consider the common concepts in the semantic cotopy SC', i.e.

$$SC'(c_i, O_1, O_2) := \{c_j | c_j \in C_1 \cap C_2 \wedge (c_j \leq_{C_1} c_i \vee c_i \leq_{C_1} c_j)\}$$

The taxonomic overlap $\overline{TO}$ between two ontologies will then be calculated as in (Maedche and Staab, 2002), but using the modified semantic cotopy SC'. Finally, to balance the lexical recall and the taxonomic overlap against each other, we compute the F-Measure of them as follows: $F(LR', \overline{TO}) = \frac{2 * LR' * \overline{TO}}{LR' + \overline{TO}}$
In particular we evaluate our automatically created concept hierarchies by calculating the lexical recall of the learned ontology O_{AUTO} against the reference ontology, O_{REF}, i.e. $LR'(O_{AUTO}, O_{REF})$, as well as how much of the concept hierarchy of O_{REF} is covered by O_{AUTO}, i.e. $\overline{TO}(O_{REF}, O_{AUTO})$. Then, we balance these two values by the above F-Measure.

5. Results

The evaluation of our approach has been conducted on two different domains: tourism and finance. For the tourism domain we used two domain-specific corpora: a collection of texts from *http://www.lonelyplanet.com* as

[2] see http://mccawley.cogsci.uiuc.edu/corpora/treebank3.html

[3] Certainly, there were some concepts which did not have a direct counterpart in the other language.

[4] In principle, FCA is able to account for polysemy of terms; however, we will gloss over this aspect in the present paper.

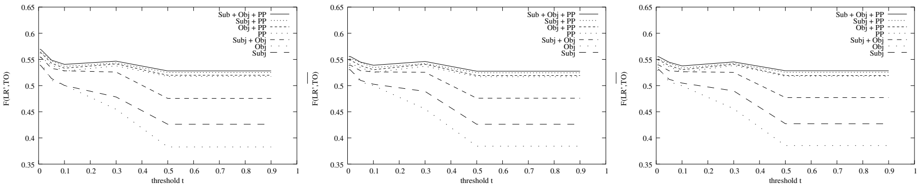

Figure 1: Attributes Tourism (FCA/Hierarchical Clustering with Complete Linkage/Bi-Section-KMeans)

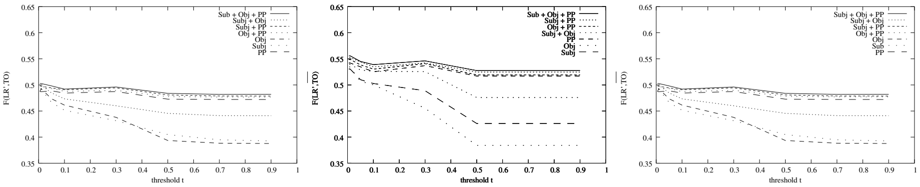

Figure 2: Attributes Reuters (FCA/Hierarchical Clustering with Complete Linkage/Bi-Section-KMeans)

well as from *http://www.all-in-all.de*, a site containing information about accommodation, activities etc. of *Mecklenburg Vorpommen*, a region in northeast Germany. Furthermore, we also used a general corpus, the British National Corpus. Altogether the corpus size was over 118 Million tokens. For the finance domain we considered Reuters news from 1987 with over 185 Million tokens.

As already mentioned in the introduction, we compare the FCA-based approach described in section 2. against a hierarchical bottom-up clustering algorithm (Day and Edelsbrunner, 1984) as well as against Bi-Section-Kmeans (Steinbach et al., 2000). In order to determine the similarity of two object vectors, we make use of the cosine measure (Manning and Schuetze, 1999). As linkage metric for the agglomerative clustering algorithm we use *complete linkage*, i.e. the most dissimilar elements of two clusters are considered to calculate the similarity of the whole clusters. Further, if the bottom-up clustering algorithm does not find any non-zero similarities, it will put the remaining elements directly under the root node of the cluster tree in line with the way FCA orders objects with disjoint features. In contrast, Bi-Section-Kmeans will produce a hierarchy by random splits.

As a first experiment, we determined which combination of the syntactic dependencies we consider i.e. verb/object, verb/subject and verb/PP-complement dependencies work best for the task at hand. Figure 1 shows the results of this first experiment using the tourism corpus for all possible combinations of syntactic dependencies. The combinations are listed top-down in order of performance. Figure 2 gives the corresponding results for the Reuters corpus[5]. All figures show the value of the F-Measure balancing $\overline{TO}$ and LR' against each other over the different thresholds used. In particular we used the following values for the threshold t: 0.005, 0.01, 0.05, 0.1, 0.3, 0.5, 0.7 and 0.9. The figures show that for all clustering approaches the use of all features outperforms every other subset of them.

Figure 3 (left) depicts the results of all the three different methods using all the syntactic dependencies for the tourism corpus. In general it seems that our FCA-based approach performs reasonably well when compared to other

[5]The reason why some values are missing is that the corresponding contexts were to big for processing them with FCA.

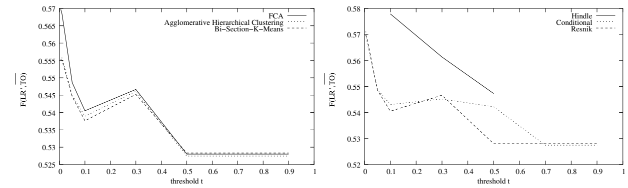

Figure 3: Comparison of clustering algorithms (left) and information measures (right)

clustering techniques. Interestingly, we also found out that the FCA-based approach produces bigger ontologies than the agglomerative clustering method. This is due to the fact that FCA introduces also abstract concepts (Ganter and Wille, 1999). In the future we will thus aim at removing these abstract concept by pruning the lattice. Finally, Figure 3 (right) shows the results of the different information measures for the FCA-based approach on the tourism domain. It becomes clear that the measure based on the conditional probability and the *Resnik* measure have a similar performance. Furthermore, the *Conditional* measure seems even to be slightly better (at $t = 000.5$). In contrast, the *Hindle* measure shows quite a different behaviour. At lower thresholds it does not cut off any information such that the contexts are so big that they can not be processed by FCA, while at higher threshold (above 0.5) it cuts off all the information. For the few data points depicted in Figure 3 (right) it seems that it achieves higher values than the other measures. In practice, higher thresholds should thus be used for this measure.

6. Related Work

In this section, we discuss some work related to apply clustering algorithms to learn taxonomies from texts as well as to the use of FCA for NLP. There exist several approaches which are based on the distributional hypothesis and which make use of clustering techniques to derive term hierarchies from text by using certain syntactic dependencies. (Hindle, 1990) for example takes into account nouns appearing as subjects and objects of verbs, but does not distinguish between these argument positions in his similarity measure. (Faure and Nedellec, 1998) present an iterative bottom-up clustering approach of nouns appearing in similar contexts. At each step, they cluster together the two most similar extents of some argument position of two verbs. (Pereira et al., 1993) present a top-down clustering approach to build an unlabeled hierarchy of nouns. As in our approach, they also make use of verb-object relations to represent the context of a certain noun. (Caraballo, 1999) also uses clustering methods to derive an unlabeled hierarchy of nouns by using data on conjunctions of nouns and appositive constructs, but goes further in that at a second step she also labels the abstract concepts of the hierarchy by considering the Hearst patterns (compare (Hearst, 1992)) in which the children of the concept in question appear as hyponyms. The most frequent hypernym is then chosen in order to label the concept. Furthermore, at a further step she also compresses the produced ontological tree by eliminating internal nodes without a label. Finally, the idea of using FCA in NLP in general is certainly not new. In (Priss, 2004) for example, several possible applications of FCA in analyzing linguistic structures, lexical semantics and lexical tuning are mentioned. (Sporleder, 2002) and (Petersen, 2002) apply FCA to yield more concise lexical inheritance hierarchies with regard to morphological features such as numerus, gender etc. In (Basili et al., 1997) FCA has also been applied to the task of learning subcategorization frames from corpora. However, to our knowledge it has not been applied before to the acquisition of domain concept hierarchies such as in the approach presented in this paper.

7. Conclusion

We have presented a new approach to automatically acquire term hierarchies from text by using Formal Concept Analysis. Our results show that this method performs relatively well compared to other state-of-the-art clustering algorithms. Further, we have also shown that for all clustering approaches using all syntactic dependencies works better than any other subset of them. Finally, we have also analyzed and discussed different information measures with regard to the task at hand.

As further work we will address the question if the set of attributes, i.e. all the verbs in the corpus, can be reduced to a smaller set of features such as Levin classes (Levin, 1993) so that the results of the clustering process are more intuitive to understand.

8. References

R. Basili, M.T. Pazienza, and M. Vindigni. 1997. Corpus-driven unsupervised learning of verb subcategorization frames. In *Proceedings of the 5th Conference of the Italian Association for Artificial Intelligence (IA*AI97)*.

G. Bisson, C. Nedellec, and L. Canamero. 2000. Designing clustering methods for ontology building - The Mo'K workbench. In *Proceedings of the ECAI Ontology Learning Workshop*.

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

P. Cimiano, S.Staab, and J. Tane. 2003. Automatic acquisition of taxonomies from text: FCA meets NLP. In *Proceedings of the PKDD/ECML'03 International Workshop on Adaptive Text Extraction and Mining*.

P. Cimiano, A. Hotho, G. Stumme, and J. Tane. 2004. Conceptual knowledge processing with formal concept analysis and ontologies. In *Proceedings of the 2nd International Conference on Formal Concept Analysis*.

W. Day and H. Edelsbrunner. 1984. Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of Classification*, 1(7).

D. Faure and C. Nedellec. 1998. A corpus-based conceptual clustering method for verb frames and ontology. In P. Velardi, editor, *Proceedings of the LREC Workshop on Adapting lexical and corpus resources to sublanguages and applications*.

B. Ganter and R. Wille. 1999. *Formal Concept Analysis – Mathematical Foundations*. Springer Verlag.

M.A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th International Conference on Computational Linguistics*.

D. Hindle. 1990. Noun classification from predicate-argument structures. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Beth Levin. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press.

A. Maedche and S. Staab. 2002. Measuring similarity between ontologies. In *Proceedings of the European Conference on Knowledge Acquisition and Management (EKAW)*. Springer.

C. Manning and H. Schuetze. 1999. *Foundations of Statistical Language Processing*. MIT Press.

F. Pereira, N. Tishby, and L. Lee. 1993. Distributional clustering of english words. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 183–190.

Wiebke Petersen. 2002. A set-theoretical approach for the induction of inheritance hierarchies. *Electronic Notes in Theoretical Computer Science*, 51.

Uta Priss. 2004. Linguistic applications of formal concept analysis. In G. Stumme and R. Wille, editors, *Formal Concept Analysis - State of the Art*. Springer.

Philip Resnik. 1997. Selectional preference and sense disambiguation. In *Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*

Helmut Schmid. 2000. Lopar: Design and implementation. In *Arbeitspapiere des Sonderforschungsbereiches 340*, number 149.

Caroline Sporleder. 2002. A galois lattice based approach to lexical inheritance hierarchy learning. In *Proceedings of the ECAI 2002 Workshop on Machine Learning and Natural Language Processing for Ontology Engineering*.

S. Staab, C. Braun, I. Bruder, A. Düsterhöft, A. Heuer, M. Klettke, G. Neumann, B. Prager, J. Pretzel, H.-P. Schnurr, R. Studer, H. Uszkoreit, and B. Wrenger. 1999. Getess - searching the web exploiting german texts. In *Proceedings of the 3rd Workshop on Cooperative Information Agents*. Springer Verlag.

M. Steinbach, G. Karypis, and V. Kumar. 2000. A comparison of document clustering techniques. In *KDD Workshop on Text Mining*.