Learning Ontologies to Improve Text Clustering and Classification

Stephan Bloehdorn[1], Philipp Cimiano[1], and Andreas Hotho[2]

[1] Institute AIFB, University of Karlsruhe, D–76128 Karlsruhe, Germany
[2] KDE Group, University of Kassel, D–34321 Kassel, Germany

Abstract. Recent work has shown improvements in text clustering and classification tasks by integrating conceptual features extracted from ontologies. In this paper we present text mining experiments in the medical domain in which the ontological structures used are acquired automatically in an unsupervised learning process from the text corpus in question. We compare results obtained using the automatically learned ontologies with those obtained using manually engineered ones. Our results show that both types of ontologies improve results on text clustering and classification tasks, whereby the automatically acquired ontologies yield a improvement competitive with the manually engineered ones.

1 Introduction

Text clustering and classification are two promising approaches to help users organize and contextualize textual information. Existing text mining systems typically use the bag–of–words model known from information retrieval (Salton and McGill (1983)), where single terms or term stems are used as features for representing the documents. Recent work has shown improvements in text mining tasks by means of conceptual features extracted from ontologies (Bloehdorn and Hotho (2004), Hotho et al. (2003)). So far, however, the ontological structures employed for this task are created manually by knowledge engineers and domain experts which requires a high initial modelling effort. Research on *Ontology Learning* (Maedche and Staab (2001)) has started to address this problem by developing methods for the automatic construction of conceptual structures out of large text corpora in an unsupervised process. Recent work in this area has led to improvements concerning the quality of automatically created taxonomies by using natural language processing, formal concept analysis and clustering (Cimiano et al. (2004), Cimiano et al. (2005)).

In this paper we report on text mining experiments in which we use automatically constructed ontologies to augment the bag–of–words feature representations of medical texts. We compare results both (1) to the baseline given by the bag–of–words representation alone and (2) to results based on the MeSH Tree Structures as a manually engineered medical ontology. We show that both types of conceptual feature representations outperform

the Bag–of-Words model and that results based on the automatically constructed ontologies are highly competitive with those of the manually engineered MeSH Tree Structures.

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 describes our approach for automatically constructing ontology structures. Section 4 reviews the concept extraction strategies used to augment bag–of–words vectors. Section 5 finally reports on the results of the text classification and clustering experiments. We conclude in section 6.

2 Related Work

To date, the work on integrating background knowledge into text classification, text clustering or related tasks is quite heterogenous. Green (1999) uses WordNet to construct chains of related synsets from the occurrence of terms for document representation and subsequent clustering. We have reported promising results when using additional conceptual features extracted from manually engineered ontologies recently in Bloehdorn and Hotho (2004) and in Hotho et al. (2003). Other results from similar settings are reported in Scott and Matwin (1999) and Wang et. al (2003).

One of the earlier works on automatic taxonomy construction is reported in Hindle (1990) in which nouns are grouped into classes. Hearst's seminal work on using linguistic patterns also aimed at discovering taxonomic relations (Hearst (1992)). More recently, Reinberger and Spyns (2005) present an application of term clustering techniques in the biomedical domain. An overview over term clustering approaches for learning ontological structures as used in this paper is given in Cimiano et al. (2005).

Alternative approaches for conceptual representations of text documents that do not require explicit manually engineered background knowledge are for example Latent Semantic Analysis (Deerwester et al. (1990)) or Probabilistic Latent Semantic Analysis (Cai and Hofmann (2003)). These approaches mainly draw from dimension reduction techniques, i.e. they compute a kind of concepts statistically from term co-occurrence information. In contrast to our approach, the concept-like structures are, however, not easily human interpretable.

3 Ontology Learning as Term Clustering

In this paper we adopt the approach described in Cimiano et al. (2004) and Cimiano et al. (2005) to derive concept hierarchies from text using clustering techniques. In particular we adopt a vector-space model of the texts, but using syntactic dependencies as features of the terms[1] instead of relying only on word co-occurrence. The approach is based on the *distributional hypothesis*

[1] Here we also refer to multi-word expressions if detected from the syntax alone.

(Harris (1968)) claiming that terms are semantically similar to the extent to which they share similar syntactic contexts. For this purpose, for each term in question we extract syntactic surface dependencies from the corpus. These surface dependencies are extracted by matching text snippets tagged with part–of–speech information against a library of patterns encoded as regular expressions. In the following we list syntactic expressions we use and give examples of the features extracted from these expressions, whereby `a:b ++` means that the count for attribute `b` of instance `a` is incremented by 1:

adjective modifiers: *alveolar macrophages*
`macrophages: alveolear++`
prepositional phrase modifiers: *a defect in cell function*
`defect: in_cell_function ++, cell function: defect_in ++`
possessive modifiers: *the dorsal artery's distal stump*
`dorsal artery: has_distal_stump ++`
noun phrases in subject or object position:
the bacterium suppresses various lymphocyte functions
`bacterium: suppress_subj ++, lymphocyte function: suppress_obj ++`
prepositional phrases following a verb:
the revascularization occurs through the common penile artery
`penile artery: occurs_through ++`
copula constructs: *the alveolar macrphage is a bacterium*
`alveolar macrophage: is_bacterium ++`
verb phrases with the verb *to have***:**
the channel has a molecular mass of 105 kDa
`channel: has_molecular_mass ++`

On the basis of these vectors we calculate the similarity between two terms t_1 and t_2 as the cosine between their corresponding vectors: $\cos(\sphericalangle(\boldsymbol{t}_1, \boldsymbol{t}_2)) = \frac{\boldsymbol{t}_1 \cdot \boldsymbol{t}_2}{\|\boldsymbol{t}_1\| \cdot \|\boldsymbol{t}_2\|}$. The concept hierarchy is built using hierarchical clustering techniques, in particular hierarchical agglomerative clustering (Jain et al. (1999)) and divisive Bi-Section KMeans (Steinbach et al. (2000)). While agglomerative clustering starts with merging single terms each considered as one initial cluster up to one single cluster Bi-Section KMeans repeatedly splits the initial cluster of all terms into two until every term corresponds to a leaf cluster. The result is a concept hierarchy which we consider as a raw ontology. Due to the repeated *binary* merges and splits the hierarchy typically has a higher overall depth as manually constructed ones. For this reason we consider in our experiments a reasonable higher number of superconcepts than with manually engineered ontologies. More details of the ontology learning process can be found in Cimiano et al. (2004) and Cimiano et al. (2005).

4 Conceptual Document Representations

In our approach, we exploit the background knowledge given by the ontologies to extend the bag–of–words feature vector with conceptual features on a

higher semantic level. In contrast to the simple term features, these conceptual features overcome a number of shortcomings of the bag–of–words feature representation by explicitly capturing multi–word expressions and conceptually generalizing expressions through the concept hierarchy. In our approach we only consider concepts which are labelled by noun phrases. As a lot of additional information is still hidden in the standard bag–of–words model we use a *hybrid* representation using concepts and the conventional term stems.

Concept Annotation. We describe here the main aspects of the concept annotation steps, the interested reader is referred to the more detailed description in Bloehdorn and Hotho (2004). (1) *Candidate Term Detection*: due to the existence of multi-word expressions, the mapping of terms to the initial set of concepts can not be accomplished directly by compiling concept vectors out of term vectors. We use a candidate term detection strategy that moves a window over the input text, analyzes the window content and either decreases the window size if unsuccessful or moves the window further if a valid expression is detected. (2) To avoid unnecessary queries to the ontology we analyze the part–of–speech patterns in the window and only consider noun phrases for further processing. (3) *Morphological Transformations*: typically the ontology will not contain all inflected forms of its entries. Therefore we use a fallback strategy that utilizes stem forms maintained in a separate index for the ontology, if the search for a specific inflected form is unsuccessful[2].

Generalization. The generalization step consists in adding more general concepts to the specific concepts found in the text, thus leading to some kind of 'semantic smoothing'. The intuition behind this is that if a term like *arrythmia* appears, the document should not only be represented by the concept [arrythmia], but also by the concepts [heart disease] and [cardiovascular disease] etc. up to a certain level of generality. This thus increases the similarity with documents talking about some other specialization of [cardiovascular disease]. We realize this by compiling, for every concept, all superconcepts up to a maximal distance h into the concept representation.

The result of this process is a "concept vector" that can be appended to the classical term vector representation. The resulting hybrid feature vectors can be fed into any standard clustering or classification algorithm.

5 Experiments

We have conducted extensive experiments using the OHSUMED text collection (Hersh et al. (1994)) which was also used for the TREC-9 filtering track[3].

[2] Typically, the problem of disambiguating polysemous window content has to be addressed properly (Hotho et al. (2003)). The ontologies we report on in this paper, contained only concepts that were unambiguously referred to by a single lexical entry thus eliminating the need for word sense disambiguation strategies.

[3] `http://trec.nist.gov/data/t9_filtering.html`

It consists of titles and abstracts from medical journals indexed with multiple MeSH descriptors and a set of queries with associated relevance judgements.

Ontologies and Preprocessing Steps: In our experiments we used domain ontologies that were extracted automatically from the text corpus on the one hand and the Medical Subject Headings (MeSH) Tree Structures Ontology as a competing manually engineered ontology on the other. The *automatically extracted ontologies* were built according to the process described in section 3 using the 1987 portion of the collection, i.e. a total of 54,708 documents. The actual concept hierarchy was built using hierarchical agglomerative clustering or divisive Bi-Section KMeans. In overview, we performed experiments with the following configurations:

agglo-7000: automatically constructed ontology, linguistic contexts for the 7,000 most frequent terms[4], taxonomy creation via agglomarative clustering;

bisec-7000: automatically constructed ontology, linguistic contexts for the 7,000 most frequent terms[4], taxonomy creation via Bi-Section KMeans divisive clustering;

bisec-14000: automatically constructed ontology, linguistic contexts for the 14,000 most frequent terms, taxonomy creation via Bi-Section KMeans divisive clustering;

mesh: manually constructed ontology compiled out of the Medical Subject Headings (MeSH)[5] containing more than 22,000 concepts enriched with synonymous and quasi-synonymous language expressions.

In all experiments, term stems[6] were extracted as a first set of features from the documents. Conceptual features were extracted as a second set of features using the ontologies above and a window length of 3.

Text Classification Setting: For the experiments in the text classification setting, we also used the 1987 portion of the OHSUMED collection. Two thirds of the entries were randomly selected as training documents while the remainder was used as test set, resulting in a training corpus containing 36,369 documents and a test corpus containing 18,341 documents. The assigned MeSH terms were regarded as categories for the documents and binary classification was performed on the top 50 categories that contained the highest number of positive training documents. In all cases we used AdaBoost (Freund and Schapire (1995)) with 1000 iterations as classification algorithm and binary weighting for the feature vectors. As evaluation measures for text

[4] More accurately, we used the intersection of the 10,000 most frequent terms with the terms present in the MeSH Thesaurus, resulting in approx. 7,000 distinct terms here.

[5] The controlled vocabulary thesaurus of the United States National Library of Medicine (NLM), `http://www.nlm.nih.gov/mesh/`

[6] In these experiments, term stem extraction comprises the removal of the standard stopwords for English defined in the SMART stopword list and stemming using the porter stemming algorithm.

Table 1. Performance Results in the Classification Setting.

			macro-averaged (in %)			
Ontology	**Configuration**	**Error**	**Prec**	**Rec**	**F_1**	**BEP**
[none]	term	00.53	52.60	35.74	42.56	45.68
agglo-7000	term & concept.sc10	00.53	52.48	**36.52**	**43.07**	46.30
agglo-7000	term & concept.sc15	00.53	**52.57**	36.31	42.95	**46.46**
agglo-7000	term & concept.sc20	00.53	52.49	36.44	43.02	46.41
bisec-7000	term & concept.sc10	00.52	53.39	36.79	43.56	**46.92**
bisec-7000	term & concept.sc15	00.52	54.36	**37.32**	**44.26**	47.31
bisec-7000	term & concept.sc20	00.52	**55.12**	36.87	43.86	47.25
bisec-14000	term & concept.sc10	00.53	51.92	36.12	42.60	45.35
bisec-14000	term & concept.sc15	00.53	52.17	**36.86**	43.20	45.74
bisec-14000	term & concept.sc20	**00.52**	**53.37**	36.85	**43.60**	**45.96**
mesh	term & concept	00.52	**53.65**	37.56	**44.19**	**47.31**
mesh	term & concept.sc5	00.52	52.72	**37.57**	43.87	47.16
			micro-averaged (in %)			
Ontology	**Configuration**	**Error**	**Prec**	**Rec**	**F_1**	**BEP**
[none]	term	00.53	55.77	36.25	43.94	46.17
agglo-7000	term & concept.sc10	00.53	55.83	**36.86**	**44.41**	46.84
agglo-7000	term & concept.sc15	00.53	**55.95**	36.67	44.30	**46.99**
agglo-7000	term & concept.sc20	00.53	55.76	36.79	44.33	46.97
bisec-7000	term & concept.sc10	00.52	56.59	37.25	44.92	47.49
bisec-7000	term & concept.sc15	00.52	**57.24**	**37.71**	**45.46**	**47.76**
bisec-7000	term & concept.sc20	00.52	57.18	37.21	45.08	47.68
bisec-14000	term & concept.sc10	00.53	54.88	36.52	43.85	45.86
bisec-14000	term & concept.sc15	00.53	55.27	**37.27**	44.52	46.27
bisec-14000	term & concept.sc20	**00.52**	**56.39**	**37.27**	**44.87**	**46.44**
mesh	term & concept	00.52	**56.81**	37.84	**45.43**	**47.78**
mesh	term & concept.sc5	00.52	55.94	**37.94**	45.21	47.63

classification we report classification error, precision, recall, F_1-measure and breakeven point[7].

Table 1 summarizes some of the classification results. In all cases, the integration of conceptual features improved the results, in most cases at a significant level. The best results for the learned ontologies could be achieved with the bisec-7000 ontology and a superconcept integration depth of 15 resulting in 44.26% macro-avg. F_1 which is comparable to the results for the MeSH ontology.

Text Clustering Setting: For the clustering experiments we first compiled a corpus which contains only one label per document. We used the 106 queries provided with the OHSUMED collection and regarded every answer set of a query as a cluster. We extracted all documents for all queries which occur in only one query. This results in a dataset with 4389 documents and 106 labels (clusters). Evaluation measures for Text Clustering are entropy, purity, inverse purity, and F_1-measure[7].

Table 2 presents the results of the text clustering task, averaged over 20 repeated clusterings with random initialization. With respect to macro-averaging, the integration of conceptual features always improves results and also does so in most cases with respect to micro-averaging. Best macro-averaged results were achieved for the bisec-14000 ontology with 20 super-

[7] For a review of evaluation measures refer to Sebastiani (2002) in the text classification setting and to Hotho et al. (2003) in the text clustering setting.

Table 2. Performance Results in the Clustering Setting.

		macro-averaged (in %)			
Ontology	**Configuration**	**Entropy**	**F_1**	**Inv. Purity**	**Purity**
[none]	terms	2,6674	19,41%	17,22%	22,24%
agglo-7000	term & concept.sc1	2,6326	19,47%	17,68%	21,65%
agglo-7000	term & concept.sc10	**2,5808**	**19,93%**	17,55%	**23,04%**
agglo-7000	term & concept.sc20	2,5828	19,88%	**17,69%**	22,70%
bisec-7000	term & concept.sc1	2,5896	19,84%	**17,72%**	22,53%
bisec-7000	term & concept.sc10	2,5361	**20,17%**	17,38%	**24,02%**
bisec-7000	term & concept.sc20	**2,5321**	20,01%	17,38%	23,59%
bisec-14000	term & concept.sc1	2,5706	19,96%	**17,76%**	22,80%
bisec-14000	term & concept.sc10	**2,4382**	**21,11%**	17,68%	**26,18%**
bisec-14000	term & concept.sc20	2,4557	20,77%	17,46%	25,67%
mesh	term & concept.sc1	2,4135	21,63%	**17,70%**	27,78%
mesh	term & concept.sc10	**2,3880**	**21,93%**	17,64%	**28,98%**
		micro-averaged (in %)			
Ontology	**Configuration**	**Entropy**	**F_1**	**Inv. Purity**	**Purity**
[none]	terms	3,12108	14,89%	14,12%	15,74%
agglo-7000	term & concept.sc1	**3,1102**	**15,34%**	14,56%	**16,21%**
agglo-7000	term & concept.sc10	3,1374	15,21%	14,43%	16,08%
agglo-7000	term & concept.sc20	3,1325	15,27%	**14,62%**	15,97%
bisec-7000	term & concept.sc1	**3,1299**	**15,48%**	**14,84%**	**16,18%**
bisec-7000	term & concept.sc10	3,1533	15,18%	14,46%	15,98%
bisec-7000	term & concept.sc20	3,1734	14,83%	14,23%	15,48%
bisec-14000	term & concept.sc1	**3,1479**	**15,19%**	**14,63%**	**15,80%**
bisec-14000	term & concept.sc10	3,1972	14,83%	14,33%	15,37%
bisec-14000	term & concept.sc20	3,2019	14,67%	14,07%	15,36%
mesh	term & concept.sc1	**3,2123**	**14,92%**	**14,91%**	**14,93%**
mesh	term & concept.sc10	3,2361	14,61%	14,64%	14,59%

concepts. These result is competitive to the one we obtained with the mesh ontology. Surprisingly the best micro avg. results could be found for the strategy adding a single superconcept only.

6 Conclusion

The contribution of this paper is twofold. We presented a novel approach for integrating higher-level semantics into the document representation for text mining tasks in a fully unsupervised manner that significantly improves results. In contrast to other approaches, the discovered conceptual structures are well understandable while not based on manually engineered resources. On the other hand, we see our approach as a new way of evaluating learned ontologies in the context of a given text clustering or classification application. Further work is directed towards improving the automatically learned ontologies on the one hand. On the other, it will aim at a tighter integration of the conceptual knowledge, including the exploration of more fine-grained and unparameterized generalization strategies.

Acknowledgements This research was partially supported by the European Commission under contract IST-2003-506826 SEKT (http://www.sekt-project.com) and the by the German Federal Ministry of Education, Science, Research and Technology (BMBF) in the project SmartWeb (http://smartweb.dfki.de).

References

BLOEHDORN, S. and HOTHO, A. (2004): Text Classification by Boosting Weak Learners based on Terms and Concepts. In: *Proceedings of ICDM, 2004* . IEEE Computer Society.

CAI, L. and HOFMANN, T. (2003): Text Categorization by Boosting Automatically Extracted Concepts. In: *Proceedings of ACM SIGIR, 2003*. ACM Press.

CIMIANO, P.; HOTHO, A. and STAAB, S. (2004): Comparing Conceptual, Partitional and Agglomerative Clustering for Learning Taxonomies from Text. In: *Proceedings of ECAI'04*. IOS Press.

CIMIANO, P. and HOTHO, A. and STAAB, S. (2005): Learning Concept Hiearchies from Text Corpora using Formal Concept Analysis. *Journal of Artificial Intelligence Research*. To appear.

DEERWESTER, S.; DUMAIS, S.T.; LANDAUER, T.K.; FURNAS, G. W. and HARSHMAN, R.A. (1990): Indexing by Latent Semantic Analysis. *Journal of the Society for Information Science, 41, 391–407*.

FREUND, Y. and SCHAPIRE, R.E. (1995): A Decision Theoretic Generalization of On-Line Learning and an Application to Boosting. In: *Second European Conference on Computational Learning Theory (EuroCOLT-95)*.

GREEN, S.J. (1999): Building Hypertext Links By Computing Semantic Similarity. *IEEE Transactions on Knowledge and Data Engineering, 11, 713–730*.

HARRIS, Z. (1968): *Mathematical Structures of Language*. Wiley, New York, US.

HEARST, M.A. (1992): Automatic Acquisition of Hyponyms from Large Text Corpora. In: *Proceedings of the 14th International Conference on Computational Linguistics (COLING)*.

HERSH, W. R.; BUCKLEY, C.; LEONE, T.J. and HICKAM, D.H. (1994): OHSUMED: An Interactive Retrieval Evaluation and new large Test Collection for Research. In: *Proceedings of ACM SIGIR, 1994*. ACM Press.

HINDLE, D. (1990): Noun Classification from Predicate-Argument Structures. In: *Proceedings of the Annual Meeting of the ACL*.

HOTHO, A.; STAAB, S. and STUMME, G. (2003): Ontologies Improve Text Document Clustering. In: *Proceedings of ICDM, 2003* . IEEE Computer Society.

JAIN, A. K., MURTY, M. N., and FLYNN, P. J. (1999): Data Clustering: A review. *ACM Computing Surveys,31, 264–323*.

MAEDCHE, A. and STAAB, S. (2001): Ontology Learning for the Semantic Web. *IEEE Intelligent Systems, 16, 72–79*.

REINBERGER, M.-L. and SPYNS, P. (2005): Unsupervised Text Mining for the Learning of DOGMA-inspired Ontologies. In: *Ontology Learning from Text: Methods, Evaluation and Applications*. IOS Press. To appear.

SALTON, G. and MCGILL, M.J. (1983): *Introduction to Modern Information Retrieval*. McGraw-Hill, New York, NY, US.

SCOTT, S. and MATWIN, S. (1999): Feature Engineering for Text Classification. In: *Proceedings of ICML, 1999*. Morgan Kaufmann. 379–388.

SEBASTIANI, F. (2002): Machine Learning in Automated Text Categorization. *ACM Computing Surveys, 34, 1–47*

STEINBACH, M., KARYPIS, G., and KUMAR, V. (2000): A Comparison of Document Clustering Techniques. In: *KDD Workshop on Text Mining 2000*.

WANG, B.; MCKAY, R.I.; ABBASS, H.A. and BARLOW, M. (2003): A Comparative Study for Domain Ontology Guided Feature Extraction. In: *Proceedings of ACSC-2003*. Australian Computer Society.