Semantic Structuring of Document Collections and Building of Term Hierarchies

An Unsupervised and Knowledge-free Approach

FLORIAN HOLZ

In this paper, I address the problem of structuring document collections in a semantic way when no information about the language or domain of the documents is available.

As underlying thematic structure a term hierarchy which is a tree with sets of words a nodes is chosen, which also can give descriptions of the resulting subsets of documents by providing key words – e.g. for hierarchical browsing.

For the extraction of the term hierarchy the used features are the statistical distributions of terms over documents which also the hierarchical structuring of the document collection is based on. Therefore as a proof-of-concept a clustering and a latent concept approach are evaluated.

The evaluation of the algorithms is done on document collections which a structuring exists for by comparing the structuring results with the given structure.

The result is a proof-of-concept that semantic structuring can be done in an unsupervised and knowledge-free way.

Introduction

For the exploration of document sets like browsing and searching a hierarchical, semantic structuring can be helpful. To get an overview over a certain topic, for instance, one might use a hierarchical structured document set with a direct access to thematically general documents located in the upper nodes. In the case of a evolving interest in a certain subtopic then there also is a direct access to the more specialised documents covering the specific subtopic. To allow this structuring when no information of the documents specific domain or even language is given it is necessary to research in the field of unsupervised and knowledge-free algorithms.

In this paper I want to show a proof-of-concept, that document collection structuring and getting descriptions of found documents and document subsets can be done automatically and without the need for expensive human made resources.

Basic Considerations

Before concrete algorithms can be chosen two questions have to be answered:

- Which properties of the documents are to be used?
- Which (kind of) is the thematic structure?

The answers for these two questions are merely related and cannot be answered separately. So we got to have a look at the available features and structures.

Thematic and semantic structures are closely related to the field of ontologies. Besides the philosophical meanings there are different views on the term "ontology" in computer science. Main common points of these views are, that ontologies are conceptualisations and instances of certain reality models (Biemann, 2005). They are mainly hierarchically organisedand mostly contain relations like PART-OF (prototype-based ontologies), HYPERNYM-OF (taxonomies) and MORE-ABSTRACT-THAN (term hierarchy).

Documents have several properties: domain, content structure, linguistic and statistical properties. For the purpose of a structuring process which should be applicable as wide as possible only statistic properties come into focus. All the other properties need too much specific previous knowledge. For a survey of the unconsidered linguistic effects and the expected relevance for the process see (Holz, 2007).

Basically interesting for the statistical analysis of text is Zipf's Law (Heyer et al., 2003). The very most frequent words are stop words and not only in one part of text very frequent but in all texts at all and with nearly the same fraction. Using this they can be ignored at the further processing of the documents. In contrast to the stop words there are some words in a document which are surprisingly frequent concerning for instance a common language reference corpus containing the totalised frequencies of words out of thousands or millions of documents. These words are often autosemantica, mostly nouns or adjectives, and constitute good candidates for describing the document's terminology and topic. Based on the reference frequency and the current frequency a significance can be calculated for the word which can serve as a measure of the relevance. For a more detailed analysis thereof see (Witschel, 2004). There a terminology extraction package has been developed which is applied here.

Zipf's Law holds not only for a concrete document but for text in general and so also for text which is distributed over several documents or composed out of those. Hence also more general terminological terms can be identified regarding similar documents as one at different levels of abstraction.
These characteristics may vary from domain to domain and one might expect that they are more expressed in e.g. scientific publications containing technical terminology than in e.g. boulevard articles of an every day newspaper.

The Approach

According to the considerations explained in the previous section, documents are treated as bags of words. Using the terminology extraction package of (Witschel, 2004) the documents are represented in the vector space model. The thematic structure is the term hierarchy which will be explained more in detail later in this section.

The terminology extraction package can make usage of certain previous knowledge like POS tags, stemming and a stop word list. For the purpose of a knowledge-free process these aspects have to investigated for substitution and elimination respectively (see Sect. Experiments and Varied Parameters). The results on leaving out the stemming may be not generalisable, but there are promising unsupervised and knowledge-free approaches of learning morphology, see (Bordag, 2005) and (Kazakov, 1997).

Based on the vector space model representation the document collection can be structured in a usual way using clustering or latent concept approaches. Taking now the built subsets of the whole collection and treating them as "documents" analogously it is possible to extract the relevant terminology for these and to structure them again to build up next level of the hierarchy. Thus the hierarchical structure is revealed by iterated classical structuring of sets of entities. The next step is to find out which level and which node a concrete term has to be assigned to. The result is a so called term hierarchy, a tree with sets of words as nodes. Based on this term hierarchy the document hierarchy is obtained by classifying the documents into the set of nodes of the term hierarchy. Hence the document and the term hierarchy does not only have the same structure but rather are one hierarchy with two views: one concentrating on the document sets assigned to the nodes and one with term sets assigned to the nodes. These term sets can serve as descriptions of the according document sets.

The term hierarchy is a hierarchical structurisation of terms and term sets, but it is not like a taxonomy or prototype-based ontology. Unlike in a taxonomy there is no strict IS-A-relation and unlike in a prototype-based ontology the terms do not occur at every level of the hierarchy but on only one so there is no plain PART-OF-relation.

In fact the terms are expected to be grouped semantically based on the statistics of their distributions over the documents and over several levels of document sets. So they are sorted by a kind of similarity in respect of belonging to the same topic and having the same level of abstractness. A illustration is given in Fig. 1.

The extracted term hierarchy is the information which is needed to structure the document collection semantically. Not taking the result of the iterated classical structuring but relying on the extracted term hierarchy serves as a kind of restriction on the features selection from a machine learning perspective It also provides the possibility to choose structuring algorithms which not only don't provide a hierarchical result but even don't give exact categorisations – like latent concept approaches – without getting the usual artefacts by hard categorising these results by taking the highest ranked class/concept. Finally the term hierarchy serves as description of the extracted subsets and also shows their contexts.

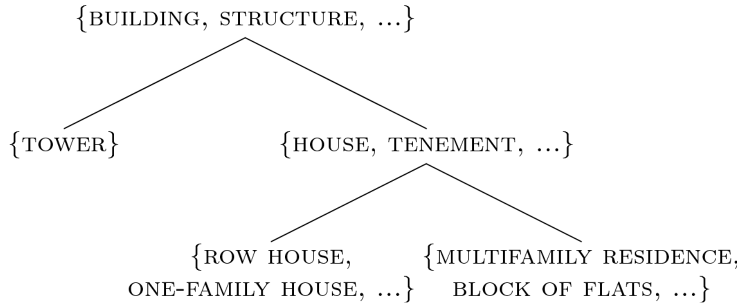
{BUILDING, STRUCTURE, ...}
{TOWER}
{HOUSE, TENEMENT, ...}
{ROW HOUSE, ONE-FAMILY HOUSE, ...}
{MULTIFAMILY RESIDENCE, BLOCK OF FLATS, ...}

Figure 1 Example of a term hierarchy

Related Work

Both the structurisation of document collections and the extraction of thematic hierarchies out of them has already been researched on with different main foci. In contrast to the existing approaches I neither just want to structure the collection by clustering or classification nor to extract a ontology as the aim itself.

The expected advantages of the chosen workflow – extract first a term hierarchy and then use it to structure the document collection – are to get

- a semantically motivated structuring ensured by the statistic properties of text and
- a description of every extracted subset of documents.

The structurisation of sets of entities can be done in several ways, the main common are clustering and latent concept analysis.

The main clustering algorithms can be grouped into the following families: hierarchical like hierarchical-agglomerative clustering, graphbased like Chinese whispers (Biemann, 2006) and graph-factorisation clustering (Yu et al., 2005), multiway clustering (Tishby et al., 1999) and (Slonim and Tishby, 2000), and the *k*-means-family.

All of them are useable with documents given as vectors of terms. None make use of semantic considerations and all of them besides the hierarchical family give flat structuring results. For more detailed surveys see (Holz, 2007), (Heyer et al., 2003), (Manning and Schütze, 1999) and (Pullwit, 2003)

Like clustering the latent concept approach is applyable on data given by document-term-matrices. The latent semantic analysis as the first exponent relied on singular value decomposition with no semantic foundation (Deerwester et al, 1999). The subsequent algorithms – e. g. probabilistic latent semantic analysis and latent dirichlet allocation – are based on document generation models, see (Hofmann, 2001) and (Blei et al., 2003).

Besides the general possibilities to extract ontologies from different sources like relational databases, dictionaries, thesauri and structured document sets there are several approaches working on unstructured text – e. g. the usage of Hearst-Pattern to extract relations. Here, I concentrate on statistic-based approaches. Besides these there are many other approaches – e. g. based on formal conept analysis (Cimiano et al., 2005) A wider overview of approaches on unstructured text provides (Biemann, 2005).

Based on a given hierarchically categorised set of documents (Makagonov et al., 2005) propose the extraction of a domain onotology by using identifying several levels of abtractness of text chunks. Their approach is in certain aspects similar to the upper baseline in my experiments (see Sect. Upper Baseline HTE).

A latent concept approach which extenses the LDA of (Blei et al., 2003) to a hierarchical model propose (Blei et al., 2004).

Applied Algorithms

As founded in Sect. Basic Considerations, documents are treated as bags of words and for every document a vector is calculated which contains the significant term and their frequencies as term weights. For the purpose of giving a proof-of-concept, that semantically motivated, hierarchical structurisations of document collections can be done without almost any language specific previous knowledge besides the statistical assumptions like Zipf's law rather simple algorithms have been chosen to be implemented.

For the structuring step these are the hierarchical-agglomerative clustering (HAC) and the probabilistic latent semantic analysis (PLSA). As implementation of the PLSA the PennAspect package of (Schein et al.) was used. As the classification in the step when the document hierarchy is built a classification using the cosine similarity measure between the document's vector and the term hierarchy nodes treated as vectors with weight 1 for all their terms is applied. Figure 2 shows the experimental setup.

The hierarchical term extraction (HTE) serves as the upper baseline which has knowledge about the given hierarchical structure of the test document collections (see Sect. Evaluation). Both the PLSA and the HAC need to know how much concepts and classes respectively are to be extracted. To know this is an unnatural assumption but if it does not work with this knowledge it is very likely not to work without, too.

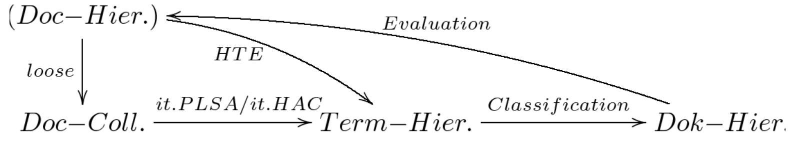

Figure 2 Experimental setup

Upper Baseline HTE

This algorithms takes the given document collection structure and constructs a term hierarchy with the same structure. For every term it is decided iteratively top-down which node it will be assigned to. Starting at the root it is decided for every node whether the term appears sufficiently uniform distributed in all the subnodes of the current node. A node is a node of the document hierarchy and consists of all the node's documents plus all the documents of its subnodes. If the term appears sufficiently evenly in all the subnodes, the term is assigned to this node. If the term does not appear sufficiently evenly, the subnode where the term occurs with the highest frequency is chosen as the new current node. As measure whether the term appears sufficiently evenly serves the coefficient of variation:

$$\text{appears\_evenly}(\ f_1, \ldots, f_k) = \begin{cases} true & , \text{VarK}(f_1, ..., f_k) < s \\ false & , else \end{cases} \qquad (1)$$

$$\begin{aligned} \text{with VarK}(X) &= \frac{\text{D}(X)}{\text{E}(X)} = \frac{\sqrt{\text{Var}(X)}}{\text{E}(X)} \quad \text{coefficient of variation} , \\ X &= (f_1, \ldots, f_k) , \\ f_i &= \text{frequency of the current term in node } i , \\ s &= \text{threshold} . \end{aligned}$$

Iterated PLSA

This structuring algorithm is based on the PLSA by (Hofmann, 2001) which uses a document-term-matrix as input. The original PLSA is applied iteratively to build up a hierarchy bottom-up:

- iterate PLSA, result concept$_{\text{level } i}$-term-matrices and concept$_{\text{level } i}$-concept$_{\text{level } i+1}$-matrices
- extract the hierarchical structure out of the concept$_{\text{level } i}$-concept$_{\text{level } i+1}$-matrices
- assign the terms to the nodes in the structure using the concept$_{\text{level } i}$-term-matrices

The concept-term-matrices serve as input for the next iteration replacing the document-term-matrix. The concept-document-matrix from the first iteration is according to the concept$_{\text{level } i}$-concept$_{\text{level } i+1}$-matrices at higher levels.

The structure is extracted top-down by assigning every subconcept *sc* as new node to that concept *c* where $p(c|sc)$ is maximal. $p(c|sc)$ is equivalent to $p(c|d)$ at the lowest level.

Filling in the terms into the structure happens analogously to the HTE but with looking up the frequencies in the according concept-concept-matrices.

Iterated HAC

The HAC is not taken as a hierarchical clustering here, but it is a representative for every clustering algorithm From the result simply the view as a set of clusters is taken. Here as similarity measure the cosine was taken. As input for the next iteration the result clusters are taken. They are described by vectors formed by the sum of the vectors of the clustered documents or clustered clusters of the next lower level respectively.

The structure forms directly out of the tree of clusters. Filling in the terms into the structure happens analogously to the HTE but with looking up the frequencies in the according vectors of the clusters.

Evaluation

Evaluation is done in a relative way by comparing the structured document collection to a formerly given structure of the same collection. This way is chosen because on the one hand there is no gold standard neither as a kind of algorithm which structures document sets nor as a semantically (respectively ontologically) structured document set. On the other hand there is no kind of established absolute measure for ontological and semantically hierarchical structures using intrinsic qualities. For instance no one can measure how good WordNet is in an general and objective way just by using a “ontologieness meter”.

Also there is no commonly accepted measure to evaluate a hierarchy of item sets against another. There are some approaches which rely on the knowledge of the right node like learning accuracy by (Hahn and Schnattinger, 1998), but this information is not accessible here. Remaining possibilities are information theoretic measures like variation of information and the *F*-value which combines precision and recall. Experiments have shown that these two measures are in accord with each other in the field of our evaluation task, c.f. (Holz, 2007). That’s why here only results showing the *F*-value are considered. The definition of precision and recall therefore rely on the document sets in the original and the result hierarchy. For every document it can be calculated how many of its neighbours in the node in the original hierarchy are assigned to its node in the result hierarchy, too. This gives the recall value *R*. Analogously it can be calculated how many of its result neighbours have already been neighbours in its node in the original hierarchy. This give the precision value *P*. *F* is calculated by $F=2*P*R/(P+R)$.

Experiments and Varied Parameters

In the experiments several parameters have been varied to estimate their influence on the result. While the HTE as an informed algorithm defines an upper baseline there has been taking two lower baselines into account: the random hierarchy (*Random*) where every document is assigned to a node randomly based on a uniform distribution of the nodes and the ”root hierarchy“ (*Root*) where every document is assigned to the root nodes which means that no structuring of the document collection has been done at all.

This threshold has a great influence on the structuring result. In general a smaller value means that the appearance of terms in subnodes/concepts/clusters must be more even and a bigger value means that the appearance of terms in subnodes/concepts/clusters can be less even. Hence with a very small threshold all terms and so also all documents will be located in the leaf nodes of the hierarchy and with a very big value all terms and all documents will be assigned to the root. This can be seen by looking at precision and recall (Fig. 3,4): High threshold and more documents together in higher nodes means high recall. Low threshold and more documents in lower nodes means more precision.

A good estimation for this threshold is 1 (cf. Fig. 3-9).

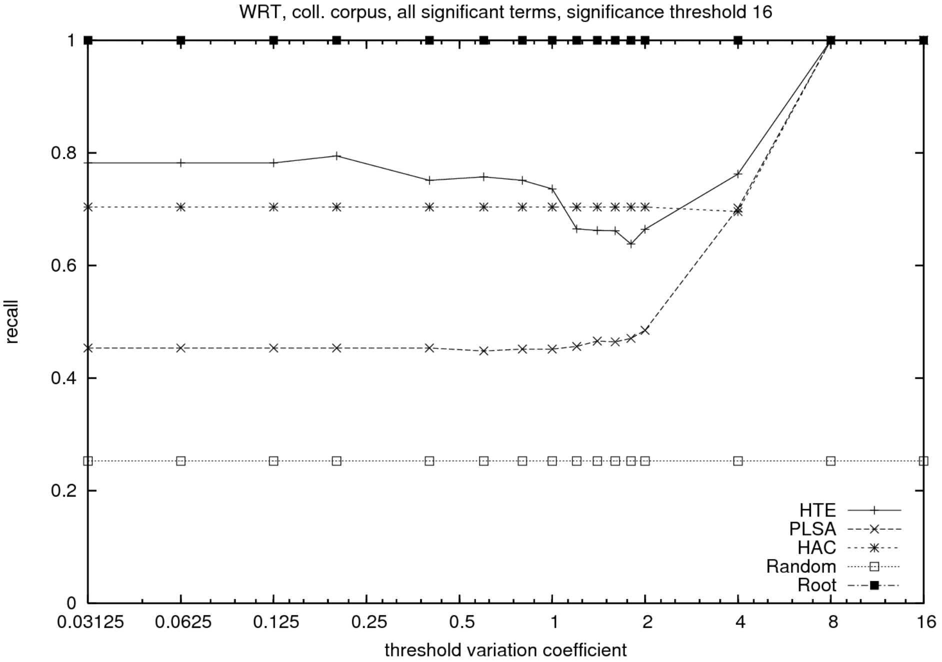

Figure 3 Recall according to threshold of coefficient of variance

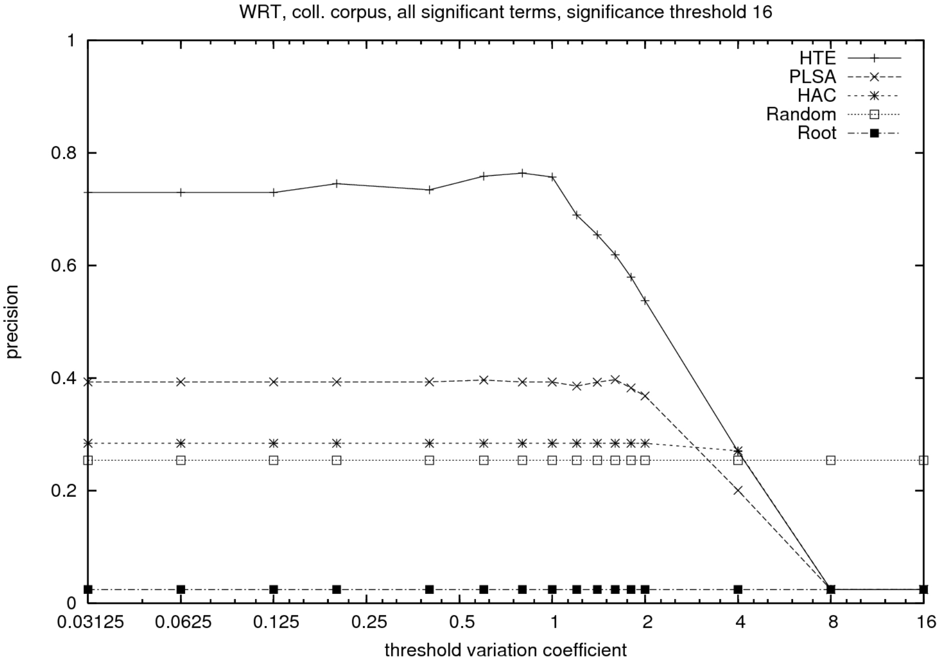

Figure 4 Precision according to threshold of coefficient of variance

As explained the distribution of terminology over the documents is crucial. The approach can only be successful if there exist terms which discriminate the documents well but also identify related documents. This is expected to be true for scientific text but can possibly denied for other texts like boulevard articles. Therefore experiments with the scientific book *Wissensrohstoff Text* (Heyer et al., 2003) of which the content structure had to be rebuilt and a collection of articles of the *Spiegel* boulevard magazine.

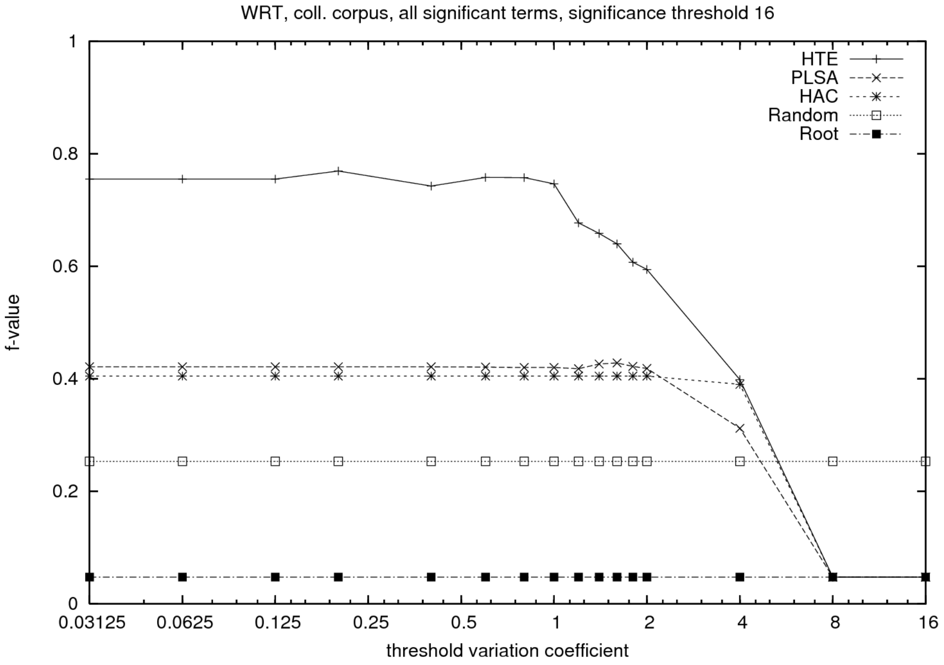

Figure 5 F-*value according to threshold of coefficient of variance*

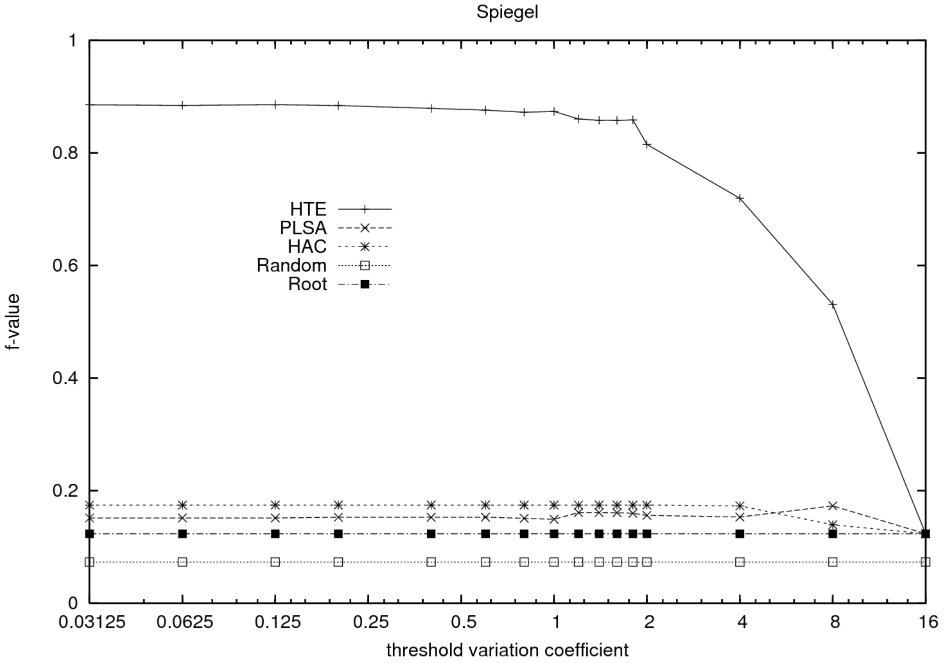

Figure 6 F-*value according to threshold of coefficient of variance*

On the scientific documents the approach applies well (Fig. 5). Both the PLSA and the HAC achieve values clearly over both lower baselines. The need for clear discriminating terminology can be seen at the result with the *Spiegel* collection. These articles consisting of mainly everyday language couldn't be restructured well (Fig. 6).

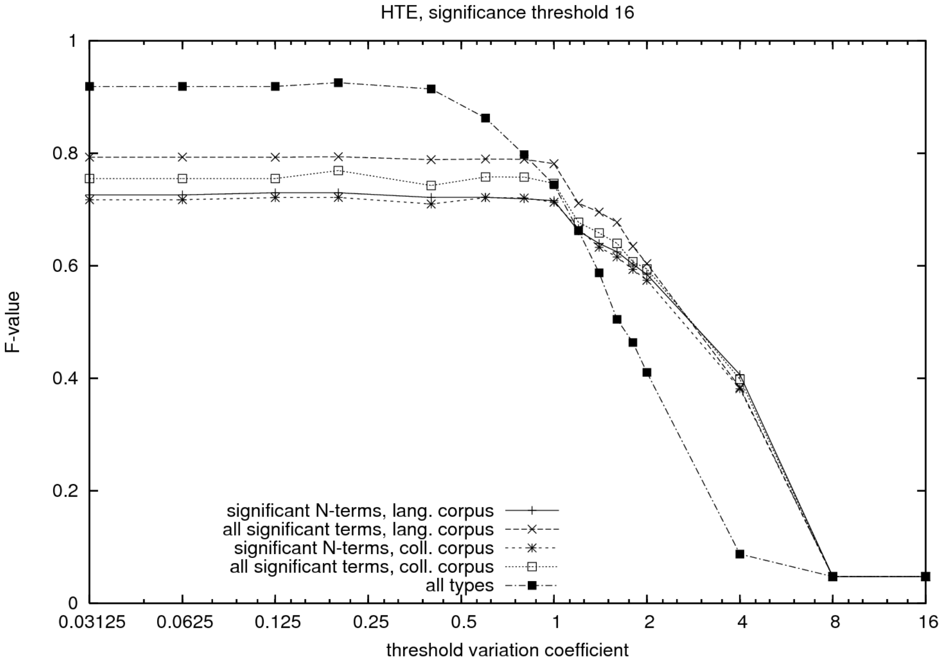

Figure 7 HTE: Comparison of term selection strategies

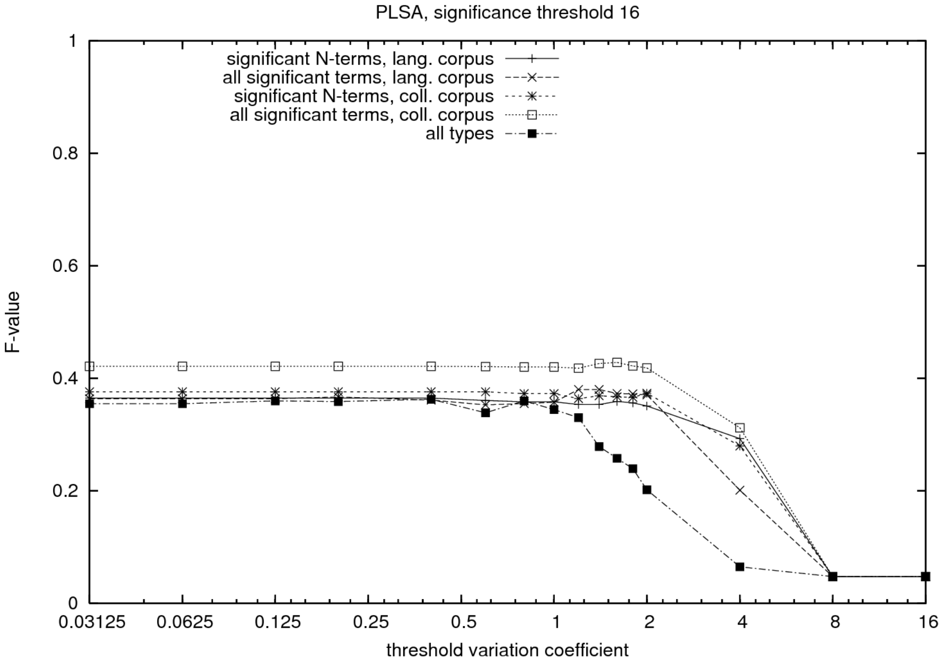

Figure 8 PLSA: Comparison of term selection strategies

The document collection (*coll. corpus*) is sufficient to estimate the significance values and a general reference corpus (*lang. corpus*) is not needed. A restriction on N-Terms and therefore a POS tagger is also not needed because simply taking all significant terms achieve the same results (Fig. 7-9).

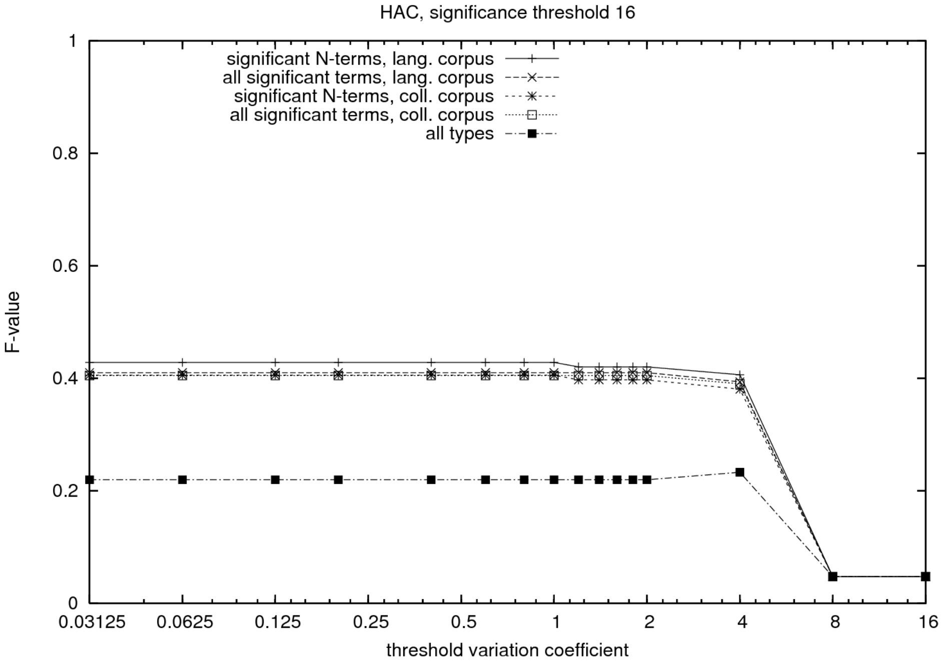

Figure 8 HAC: Comparison of term selection strategies

Conclusions and Future Work

It could be shown, that semantic structuring of document collections can be done based on statistic assumptions of term distribution and without other language specific knowledge. As a future work the statistical analysis could be refined and instead of the applied simple algorithms more sophisticated ones should be evaluated.

References

Biemann, C.; *Ontology Learning from Text – A Survey of Methods*. LDV-Forum **20**(2) 75–93, 2005.

Biemann, C.; *Chinese whispers – an efficient graph clustering algorithm and its application to natural language processing problems*. In: Proceedings of the HLT-NAACL-06 Workshop on Textgraphs, 2006.

Blei, D.M., Ng, A.Y., , Jordan, M.I.: *Latent dirichlet allocation*. Journal of Machine Learning Research **3**, 993–1022, 2003.

Blei, D.M., Griffith, T., Jordan, M.I., Tenenbaum, J.: *Hierarchical topic models and the nested chinese restaurant process*. In: Advances in Neural Information Processing Systems. Volume 16., MIT Press, 2004.

Bordag, S.; *Unsupervised knowledge-free morpheme boundary detection*. In: Proceedings of RANLP 05, 2005.

Cimiano, P., Hotho, A., Staab, S.: *Learning concept hierarchies from text corpora using formal concept analysis*. Journal of Artificial Intelligence Research **24**, 305–339, 2005.

Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.T., Harshman, R.: *Indexing by latent semantic analysis*. Journal of the American Society for Information Science **41(6)**, 391–407, 1990.

Heyer, G., Quasthoff, U., Wittig, T.; *Text Mining: Wissensrohstoff Text – Konzepte, Algorithmen, Ergebnisse*. W3L-Verlag, 2003.

Hahn, U., Schnattinger, K.: *Towards text knowledge engineering*. In: AAAI/IAAI. 524–531, 1998.

Hofmann, T.: *Unsupervised learning by probabilistic latent semantic analysis*. Machine Learning **42** (2001) 177–196

Holz, F.; *Automatische Extraktion von Termhierarchien aus Dokumentenkollektionen für die semantische Strukturierung (Automatic Extraction of Term Hierarchies for Semantic Structuring)*. Diplomarbeit (Master's Thesis), University of Leipzig, 2007.

Kazakov, D.; *Unsupervised learning of nave morphology with genetic algorithms*. In: Workshop Notes of the ECML/MLnet Workshop on Empirical Learning of Natural Language Processing Tasks, 105–112, 1997.

Makagonov, P., Figueroa, R., Sboychakov, K., Gelbukh, A.: *Learning a domain ontology from hierachically structured texts*. In: Proceedings of the ICML 2005, 2005

Manning, C.D., Schütze, H.: Foundations of Statistical Natural Language Processing. MIT Press, 1999.

Pullwitt, D.: *Explorative Analyse von Textkorpora mit Clusterverfahren*. Dissertation, Institut für Informatik, Universität Leipzig, 2003.

Schein, A.I., Popescul, A., Ungar, L.H.: Pennaspect: *Two-way aspect model implementation*, http://www.cis.upenn.edu/datamining/software_dist/pennaspect/index.html

Slonim, N., Tishby, N.: *Document clustering using word clusters via the information bottleneck method*. In: Research and Development in Information Retrieval. 208–215, 2000.

Tishby, N., Pereira, F., Bialek, W.; *The information bottleneck method*. In: Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing. 368–377, 1999.

Witschel, H.F.; *Text, Wörter, Morpheme – Möglichkeiten einer automatischen Terminologie-Extraktion*. Ergon Verlag, 2004.

WordNet2.1; *Database Statistics*: http://wordnet.princeton.edu/man/wnstats.7wn

Yu, K., Yu, S., Tresp, V.; *Soft clustering on graphs*. In: Advances in Neural Information Processing Systems **18**, 2005.