CREWS Report 99-05

Document Maps: Semantic Structuring of Technical Document Collections

Andreas Becks, Stefan Sklorz, Matthias Jarke

Lehrstuhl für Informatik V, RWTH Aachen, Ahornstraße 55, 52056 Aachen, Germany
phone: +49 (0)241 21 516
{becks, sklorz, jarke}@informatik.rwth-aachen.de

Document Maps: Semantic Structuring of Technical Document Collections

Andreas Becks, Stefan Sklorz, Matthias Jarke

Lehrstuhl für Informatik V, RWTH Aachen, Ahornstraße 55, 52056 Aachen, Germany
{becks, sklorz, jarke}@informatik.rwth-aachen.de

Abstract. Corporate knowledge management in science and engineering-intensive organizations involves tasks such as standard generation and evaluation, comparison of related cases and experience reuse in their treatment, and the ability to rapidly retrieve all relevant documents around a certain topic or project even if this project has extended over years or decades. No single information retrieval technique is likely to adequately deal with such tasks independent of the specific situation. In this paper, we therefore present a modular approach that allows a variety of techniques from clustering, exploitation of semantic structure knowledge, and visualization to be used in the handling of technical document collections for knowledge management purposes. Usable implementations exist for large parts of the approach. Two real-world usage experiences with projects in the chemical engineering and medical domains provide initial evidence for the value of the approach.

1 Introduction

Information retrieval is a central research topic with still increasing relevance in the context of a global knowledge and information society. In times where specific information is of highest value for enterprises or organizations and knowledge becomes the most important production factor, techniques enabling an adequate access to information play the role of a key technology for an appropriate management of knowledge.

The identification and analysis of knowledge available in an enterprise is a key element of knowledge management. Often not the lack of knowledge sources in a company is a problem, but the flood of unstructured information [28]. Information technology (IT) in combination with human resources offers a hybrid solution towards an effective management of knowledge: IT can support experts in detecting "hidden" knowledge as well as structuring and condensing existing knowledge sources. Even more, IT can be seen as an "enabling technology" to build an organizational memory in a collaborative working environment [4].

Corporate knowledge can be contained in many of a company's documents. For example, consider engineering-intensive organizations like the chemical industries. Here, management documents, requirement definitions, technical guidelines or manuals of chemical plants contain important information about the company's goals or issues regarding configuration and maintenance of machines. An important element in the mosaic of knowledge identification is to obtain a structured overview of such documents available within an enterprise: Which and how many documents are concerned with which topic? What relationships between documents exist? Information retrieval systems limited to query interfaces are not sufficient to offer an adequate support. In general, it is hard or even not possible to state appropriate queries and to post-process the results in a way that provides the user with an overview about the existing material. Thus, developing automated semantic structuring methods for document collections becomes an important topic.

The different types of documents which are relevant in this context require an appropriate handling for gaining a reliable overview. The spectrum reaches from semi-structured documents like requirement definitions to knowledge intense texts like scientific document abstracts. Therefore, it is necessary to define a modular approach towards a semantic structuring which is able to meet the special needs of every specialized document collection interesting in a particular knowledge management task.

Such a framework has to address two basic demands: First, it has to allow the integration of adequate semantic criteria for assessing inter-document similarity. Second, the inherent structure of the collection induced by the similarity information has to be detected and visualized for providing an expressive user interface. We will adapt the concept of 'semantic document maps' for presenting the collection's structure.

1.1 Related Work

Relating to the application of semantics there is a wide spectrum of models developed in information retrieval research. For some classes of documents, e.g. requirement definitions, methods like the well-known vector space model (VSM) [29, 32] provide good results. The VSM compares term vectors according to their similarity, e.g. by applying a cosine measure. The term vectors consist of real-valued components describing the weight of the corresponding terms of the indexing vocabulary. An attempt to improve the performance of this model was made in [9], where 'latent semantic indexing' is introduced. This approach considers the correlation structure of terms and documents in order to obtain a more reliable indexing of documents.

For assessing the degree of similarity of other document types, such as management documents or scientific document abstracts, it is not sufficient to consider only syntactic aspects of the texts. In the context of more specialized document collections elaborated knowledge-based approaches become important. Improvements here include the use of thesaurus based approaches [7] or even retrieval models based on terminological (or description) logics [25, 33]. The latter approaches allow the definition of concepts and relationships between concepts in order to express background knowledge and thus to incorporate explicit semantics into information retrieval. Two examples of domain specific methods of semantic retrieval are [5], where a system for searching in usenet newsgroup files is introduced and discussed, and [14], dealing with semantic software retrieval.

The second research area deals with structuring a document collection and visualizing the result. Here the main idea is to detect document clusters where the objects (documents) within the clusters are very similar regarding a certain measure of similarity. The basis for this approach is the cluster hypothesis by van Rijsbergen [30] which states that 'closely associated documents tend to be relevant to the same requests'. A method of accessing documents based on clustering is proposed in [8]. The main idea of this so-called scatter/gather browsing is to provide an iteration of two steps for browsing through a document collection. In a first step documents are grouped according to dominant key words (scatter phase). The user then can chose interesting groups which are combined afterwards (gather phase). These phases are repeated until the user switches to a focussed search. Scatter/gather allows a dynamic structuring but the criteria for grouping the documents are very superficial. Furthermore, there is no intuitive representation of the documents' relationships. In [1] a combination of document clustering and visualization is proposed which aims at a better identification of relevant documents in query result sets. The basis for this approach is the vector space model. Using the cosine measure the similarity of all pairs of documents from the top ranking list of the result set is calculated. This relationship is then visualized in a 2D or 3D-space and leads to a so-called 'document map' [36]. The concept of 'semantic maps of documents' has also been addressed in several papers which use self-organizing feature maps (SOFM) [19] as a basis for clustering and visualization. A very promising approach is [22] which uses a SOFM in order to generate a map of documents where documents dealing with similar topics are located near each other. The similarity measure uses statistical information about short word contexts. In [24] a SOFM is used to order documents according to key terms extracted from document titles.

The approaches discussed are concerned with information retrieval issues in a general setting. They do not consider the requirements of special scenarios: Whereas documents containing semi-structured requirement definitions can be handled using simple statistical models, more sophisticated methods like knowledge based retrieval have to be applied when dealing with management documents or scientific document abstracts. In contrast, we propose a modular model for gaining a semantic structuring of specialized document collections which enables the incorporation of different levels of background knowledge in order to overcome the shortcomings of the 'isolated' approaches. As an interactive user interface we adapt the idea of 'document maps'. Furthermore, a concrete realization of the model will be presented. We discuss the use of suitable retrieval models in two examples and show how they fit into the framework for producing a semantic arrangement. The first example is concerned with so-called 'use cases' (documents from the field of requirements engineering), the second deals with medical document abstracts (cf. [2, 3]) . A case study presenting the results of a semantic structuring of use cases will show very promising results regarding its quality and usability.

1.2 Paper Organization

The next section discusses the notion of 'semantic structuring' in more detail and presents an appropriate interaction paradigm which allows a fruitful exploration of document collections. Section 3 introduces the proposed modular approach, section 4 presents two examples of specialized document collections along with their special requirements and adequate models for measuring the documents' similarity. The

following sections are concerned with detecting and visualizing the inherent structure of the collection. A case study completes the technical considerations before future research will be discussed.

2 Semantic Structuring and 'Document Landscapes'

An abstract and merely logical view of a semantic structuring of specialized document collections is presented in figure 1: A previously unarranged set of documents from a pre-selected collection, e.g. a thematically focussed branch of an enterprise digital library or documents categorized by a common set of metadata, is structured considering the document semantics adequately. This results in grouping similar documents and in pointing out relationships between the groups. Thus, the inherent structure of the collection will be worked out.

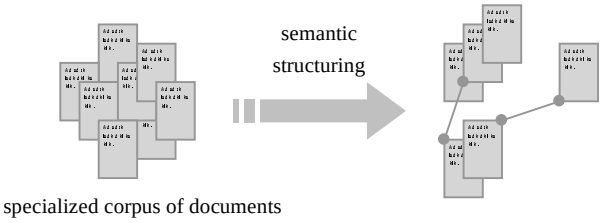

Figure 1: Logical view of a semantic structuring

Supplemental to a semantic structuring an interactive and intuitive interface is necessary which presents the structure and allows some interaction for exploring it. In this work we propose the approach of producing a 'landscape of documents' visualized in a semantic document map which expressively presents the structure of the document collection. Figure 2 shows our concept of a semantic document map and its interactive features.

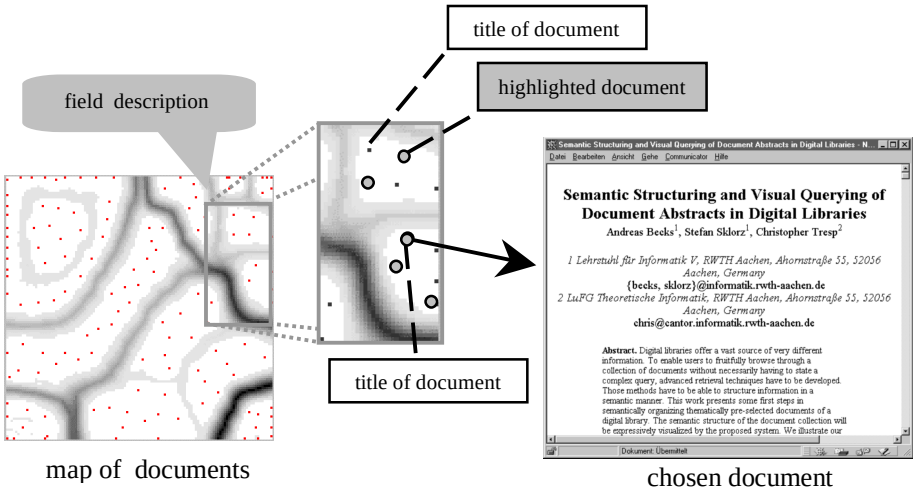

Figure 2: Interaction with a map of documents

The semantics of this map can be described by a metaphor of "mountains and valleys": The documents of the collection are represented as points in the map. Similar documents – according to the similarity measure used during the analysis of the document collection – are grouped as neighbored points located in common bright shaded areas. These areas or "valleys of similar documents" are separated by dark borders or "mountains" representing the distance between document groups. The darker the color, or the higher the "mountain", the more dissimilar are the separated groups of documents.

To characterize each group of documents field descriptions can be generated. This can be done, for example, by merging the most relevant key terms of the document set under consideration using the SMART indexing model [32].

For interacting with the map the user can mark an area and zoom into the specified filed. The documents within the fields are described by their titles. By clicking on a point in the map the user receives the corresponding document. In addition to browsing through the map a query interface can highlight relevant documents with respect to an explicitly formulated information need. The retrieval value for each document regarding the query can be represented by different color shades.

The interaction paradigm described here allows the user to browse through the documents interactively. Thus, he or she can explore the collection and learn about the inherent structure of the corpus, i.e. identify relationships between documents and between groups of documents. The benefit coming along is twofold: First, the user's information need can stepwise be refined with respect to the information available. Furthermore, the user can retrieve relevant information regarding information needs which are hard to express by an explicit query. The second advantage is that besides using the structuring as an information retrieval interface the semantic document arrangement supports the process of analysing and condensing the enterprise knowledge contained in the documents.

3 A Modular Approach for Gaining a Semantic Structuring

In this section the modular approach for gaining a semantic structuring of specialized document collections will be presented. The idea of developing such a modular scheme is that a concrete realization of a semantic arrangement has to meet the special requirements of each collection and application under consideration. This includes both capturing the semantics of the collection and presenting the result of the structuring process. The similarity of requirement definitions or technical manuals may be relatively easy to assess due to the use of specialized key terms and phrases whereas management documents are more sophisticated to compare against each other. At the knowledge intense end of the spectrum medical document abstracts can be found, for example. Depending on the application context different needs for interaction and presentation may arise. Thus, it should be possible to combine different techniques of information retrieval and varying visualization approaches. We now describe on a logical level the elements leading to a semantic structuring and its visualization. Figure 3 presents two logical units and their interface. The first component is concerned with analysing the collection, i.e. with capturing the documents' semantics and defining their relationships. Based on an appropriate indexing scheme a measure of similarity has to be defined. Depending on certain stylistic features of the documents and matters of content it has to be decided whether simple and common retrieval approaches are sufficient or whether more elaborated and specialized methods have to be applied.

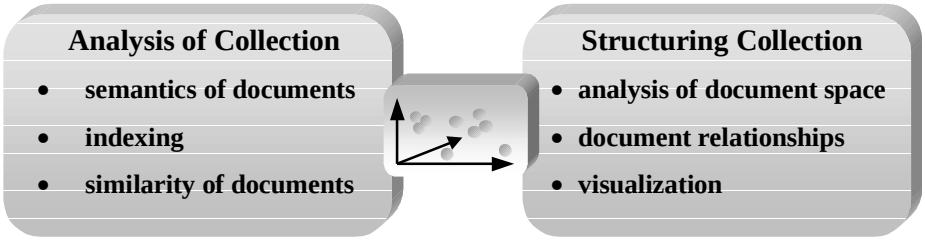

Figure 3: Conceptual scheme

As pointed out in the introduction, there are many different methods suitable for comparing specialized documents, ranging from syntactical and statistical to logic-based approaches. The indexing models cover numerical as well as symbolic approaches. Thus, for detecting the inherent structure of a collection in a general setting the indexing schemes of the documents cannot be used directly due to their different nature. Instead, the spatial information which can be extracted from the similarity values of each pair of documents can be used as an interface between the 'analysis' and the 'structuring' component (cf. section 5). This 'semantic space of documents' enables the use of different methods from data mining (cf. [12]) for detecting structures of high-dimensional spaces. Visualization techniques can be adopted from the field of data mining, too (see for example [17]). Which of them are suitable in an information retrieval environment will be a topic of future research.

The next sections introduce a concrete 'instance' of the framework proposed. We present methods for assessing the similarity of two exemplary specialized document collections, discuss the problem of generating the 'semantic space of documents' and use a neural network to analyse and visualize the structure of the document collection.

4 Assessing the Similarity of Documents

In this section retrieval models suitable for two exemplary document collections will be presented. The first collection contains requirement definition documents from software engineering. It turns out that these documents can be handled using the vector space model. The second corpus contains medical document abstracts. This collection needs a more elaborated retrieval model which is based on a fuzzy terminological description language.

4.1 Use Cases

In object-oriented software engineering documents called 'use cases' are produced during the analysis of requirements. These documents are an essential part of the Unified Modeling Language (UML) and describe processes and situations of using object-oriented systems. Formally, use cases are semi-structured narrative texts which introduce the way an external actor uses a software system to complete a certain process. They are no formal specifications but imply requirements by presenting usage scenarios [23]. A semantic structuring of the use cases created in a project can help in gaining more insight about component relationships and redundancies. Thus, semantic structuring turns out to be an important factor for the knowledge management process in software engineering.

Figure 4 shows two simple examples of use cases from the CAPE-OPEN[1] project (cf. section 7). The 'description' part contains text describing the actor's behavior. It can be observed that the linguistic style is simple: The texts present a process by describing the actions chronologically. Actions and objects are clearly pointed out by using unequivocal terms and key words. These characteristics allow the application of simple statistical information retrieval models for assessing the similarity of the documents.

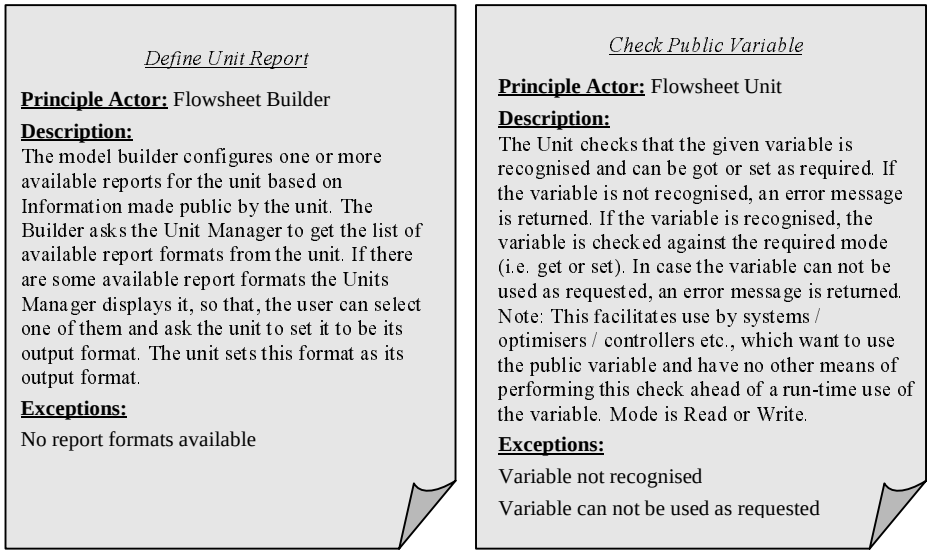

Figure 4: Use cases as used in the CAPE-OPEN project.

The model used here is the well-known vector space model. The documents are described by means of numeric term vectors which consist of indexing term weights. To sketch the calculation of description vectors for the documents briefly: After eliminating common words like conjunctions, articles and other insignificant terms from the documents the remaining terms are reduced to their stems by using the Porter stemming algorithm [27], for example. The resulting set of terms $T =_{def} \{t_1,...,t_n\}$ serves as the indexing vocabulary. In term vectors $\boldsymbol{v} =_{def} \{v_1,...,v_n\}$, $v_i \in \mathbb{R}$, each component v_i corresponds to a term $t_i \in T$ and is called the term weight of term t_i. These weights can be simply the term frequency tf_{ik} of t_i in document k or, more elaborated, the SMART weight of t_i. The latter is calculated as

$$v_i =_{def} tf_{ik} \cdot idf_i,$$

[1] CAPE-OPEN is a world-wide effort of the chemical industries, vendors and research institutes to standardize interfaces for an open simulator environment, funded by the European Comission.

where idf_i is the inverse document frequency of t_i in the collection, i.e.

$$idf_i =_{def} \log (N / n_i)$$

with N denoting the number of documents in the collection and n_i the number of documents containing t_i. To compensate the document length the description vectors should be normalized after calculating the weights. Thus, every component v_i is transformed to v_i' with

$$v_i' =_{def} v_i / |\mathbf{v}|,$$

where $|\boldsymbol{v}|$ denotes the length of the description vector $\boldsymbol{v}$. The similarity of documents k and m can now be defined as the cosine of the angle between their corresponding description vectors $\boldsymbol{v}_k$' and $\boldsymbol{v}_m$', according to the heuristic that similar documents have description vectors pointing to similar directions. Thus, the similarity $\sigma(\boldsymbol{v}_k, \boldsymbol{v}_m)$ of the documents k and m is defined as the inner product of the normalized vectors $\boldsymbol{v}_k$' and $\boldsymbol{v}_m$'.

4.2 Medical Document Abstracts

To assess the similarity of medical document abstracts more elaborated techniques have to be used. There are some fine-granular differences in the meaning of certain key terms (e.g. tumour, cyst) that can hardly be captured by applying thesauri. Furthermore, disciplines like medicine are characterised by a certain 'vague' notion, e.g. 'high aged person', 'very high relative risk' (see example below). We use a multi-valued terminological logic [38] in order to take linguistic vagueness, inherent in the medical terminology, into account and to support the concept of similarity in an adequate way.

The next section presents our approach for describing the documents. This description has to be powerful enough to serve as a basis for a knowledge based comparison of documents. After that we sketch the process of indexing the abstracts which requires some natural language processing. Finally, we introduce a knowledge based measure of similarity for comparing the indexed documents.

4.2.1 Representing Medical Abstracts

Terminological knowledge representation systems [26] are very expressive and semantically sound. They offer an excellent basis for comparing texts in a knowledge based manner. This section shows how these formalisms can serve to measure the similarity of texts.

Due to the vague notion inherent in the medical terminology, medical facts cannot be adequately expressed using a crisp logical representation. This implies that classical description logic based systems are not suitable for this kind of knowledge. There are only a few approaches dealing with representation of vague knowledge. In [35, 43] a term subsumption language is extended using concepts of many-valued logic. In the context of vagueness inherent to medical terminology both approaches are not sufficiently expressive. [43] uses multi-valued predicates only on the level of atomic concepts, [35] lacks the possibility to express linguistic hedges [45].

In [38] basic concepts of a more expressive representation language and fundamental patterns of reasoning for vague concepts are introduced. The proposed system is a terminological one, which means that it consists of a formalism called TBox for defining concept knowledge, and a language for filling the concepts and their relations with real world objects. The latter formalism is an assertional component called ABox. Table 1 shows an extract of the formal syntax and semantics of the TBox formalism. The notion of classical hybrid systems is extended by the introduction of the so-called VBox which serves to define vague attribute values. Those attributes are represented as fuzzy sets [44] on a concrete domain.

Table 1: An extract from the formal syntax and semantics of the TBox formalism

Syntax	Semantics
$C_1 \leq$ anything	Introduction of the primitive concept C_1 as a subset of the *universe of discourse*, denoted by Δ_C: $Inc[C_1, \Delta_C] =_{def} 1$, where *Inc*: $F(\Delta) \times F(\Delta) \rightarrow [0,1]$ denotes a fuzzy inclusion, e.g. *Lukasiewicz's* inclusion, and $F(\Delta) =_{def} \{\Phi \mid \Phi: \Delta \rightarrow [0,1]\}$.
$C_1 \leq T$ or $C_1 := T$	Introduction of the defined concept C_1 (necessary or sufficient condition, respectively) where C_T denotes the concept defined by the Term T: $Inc[C_1, C_T] =_{def} 1$.
R: $C_1 \times C_2$	Introduction of role R as a fuzzy binary relation between the two fuzzy concepts C_1 and C_2, R: $C_1 \times C_2 \rightarrow [0,1]$, with $\mu_R(x,y) =_{def} \begin{cases} \geq 0 & , \text{ if } \mu_{C1}(x) > 0 \wedge \mu_{C2}(y) > 0 \\ = 0 & , \text{ otherwise.} \end{cases}$
CON: $C_1 * G[C_1]$	Introduction of the feature connection CON between the concept C_1 and the concrete fuzzy set domain $G[C_1]$ of this concept, CON: $X \rightarrow F(G[C_1])$, where $X =_{def} \{ x \mid x \in \Delta_{C1} \wedge \mu_C(x) > 0 \}$ denotes the instances of C_1.
$\neg C_1$	The concept operator $\neg$ calculates the negation of a concept. We use the Lukasciewicz negation : $\forall d \in \Delta_{C1}: \mu_{\neg}(d) =_{def} 1 - \mu_{C1}(d)$
$C_1 \sqcap ... \sqcap C_n$ $C_1 \sqcup ... \sqcup C_n$	The concept operator $\sqcap$ ($\sqcup$) combines a finite number of concepts C_1, ..., C_n using a T-norm τ (co-T-norm σ): $\forall d \in \Delta_C: \mu_{and}(d) =_{def} \tau(\mu_{C1}(d),..., \mu_{Cn}(d))$ or $\forall d \in \Delta_C: \mu_{or}(d) =_{def} \sigma(\mu_{C1}(d),..., \mu_{Cn}(d))$, respectively.
restrict (CON,F_1)	Restriction of possible attribute values of a feature connection CON to the attribute value F_1. CON(d) denotes the feature value of d assigned by the feature connection CON: $\forall d \in \Delta_C: \mu_{restrict}(d) =_{def} Inc[CON(d),F_1]$

The language proposed in [38] can be used for indexing medical abstracts. To illustrate this, consider the following example taken from an abstract of the *Journal of Clinical Oncology*[2]:

(1) Patients who survive Hodgkin's disease at a young age are at very high relative risk of subsequent malignant neoplasms throughout their lives.

To keep the example simple, assume the following alternative sentence:

(2) High aged patients surviving cancer are not very likely to bear subsequent diseases.

[2] The journal of clinical oncology is © by the American Society of Clinical Oncology. A number of abstracts from this journal is available at http://www.jcojournal.org.

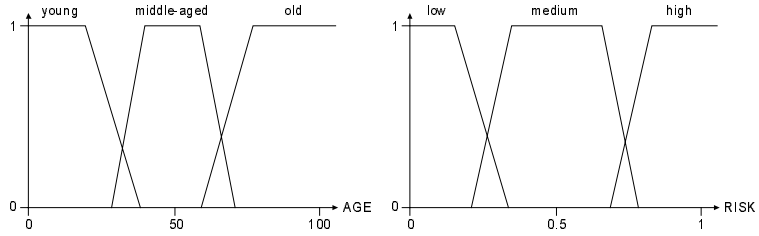

Figure 5: Visualization of vague features

Table 2: Indexing texts using ABoxes. The asterisks indicate that the corresponding strings relate to objects. The double colon (::) marks object introductions

Abox A_1 for Sentence (1)	ABox A_2 for Sentence (2)
HODGKIN_DISEASE* :: LYMPHOGRANULOMATOSIS	DISEASE* :: DISEASE
NEOPLASM* :: DISEASE	CANCER* :: CANCER
PATIENT* :: HUMAN ; RISK* :: RISK	PATIENT* :: HUMAN; RISK* :: RISK
AGED (PATIENT*, YOUNG)	AGED (PATIENT*, OLD)
SURVIVES (PATIENT*, HODGKIN_DISEASE*)	SURVIVES (PATIENT*, CANCER*)
RELATED_WITH (PATIENT*, RISK*)	HAS_FORM (DISEASE*, SUBSEQUENT)
HAS_INTENSITY (RISK*, HIGH)	HAS_INTENSITY (RISK*, LOW)
RELATED_WITH (RISK*, NEOPLASM*)	RELATED_WITH (RISK*, DISEASE*)

The two sentences can now be represented as follows. The fundamental conceptual knowledge has to be defined in the TBox, which contains at least the following concept and role definitions and feature connections:

- Primitive concepts "HUMAN" and "RISK" have to be introduced. Furthermore, the concept "LYMPHOGRANULOMATOSIS" as a sub-concept of "CANCER" and "DISEASE" must be defined.
- Necessary roles to be defined include SURVIVES: HUMAN × DISEASE as well as RELATED_WITH: ANYTHING × ANYTHING.
- To associate concepts with vague attributes, feature connections like AGED: HUMAN * AGE[HUMAN] and HAS_INTENSITY: PHENOMENON * INTENSITY[PHENOMENON] are necessary. The feature connection HAS_FORM: DISEASE * FORM[DISEASE] associates diseases with (not necessarily) vague attribute values.

Some vague features defined in the VBox are shown graphically in figure 5. Medical abstracts can be seen as concrete knowledge about objects of the real world and are thus represented in terms of an ABox as demonstrated in table 2. To summarize, we use a terminological knowledge representation system which enables vague concept interpretations, allows relationships with uncertain structure between concepts and introduces vague features. The background knowledge is implemented using the language provided by the TBox for concepts – which may be defined as vague ones – and the VBox for vague features, respectively. The ABox formalism is used for indexing the abstracts. An alternative to the language used in this section may be [39] where a sound and complete subsumption algorithm for a similar expressive knowledge representation language is introduced.

4.2.2 Indexing Medical Abstracts

One key problem is how to extract relevant information (concepts and instances) from a textual source, in this case medical abstracts that are written down in medical terminology. Overall, this problem belongs to the complex domain of natural language processing (NLP). An interesting idea of how to connect knowledge representation with language processing is originated by [42] and was adapted in [40]. The latter work presents a medical terminus parser, i.e. a parser which copes with specific characteristics of the medical terminology (medical findings in the example used in the cited paper). This special NLP method is characterized by

a) a more weight carrying semantic component instead of a very elaborated syntactic component. The input is very often wrapped into *telegram style* sentences and therefore is not subject to conventional syntactic rules of well formed expressions,
b) the need for elaborated special domain knowledge within a supporting knowledge base.

During the process of syntax analysis, a rule of the underlying grammar is applied if both the syntax is accurate and there is a semantic correlation to corresponding knowledge in a knowledge base. In the case of medical findings the telegram style does not allow a complete verification of the syntax. Nevertheless, the idea to couple syntax and semantic analysis remains fruitful. In the processing of medical document abstracts analyzing the syntax serves as a stronger filtering mechanism.

A further problem addresses the intensive use of vague linguistic notions in medical terminology. It is plausible to use our special knowledge representation formalism to handle this kind of vague knowledge. The analysis of medical texts is done by using the formalism as follows: Medical statements of an abstract are translated into ABox objects within a component for medical terminology processing. For example, descriptions like 'slight elongation of the liver' are transformed into an ABox instance of the general concept liver connected with the corresponding modified feature SLIGHT(ELONGATION) where SLIGHT is defined in the VBox.

4.2.2 A Knowledge-Based Measure of Similarity

The degree of similarity of two ABoxes can be computed by comparing the features connected with a concept *c* and the matching roles in each box. First of all, the major steps to compare single constructs from each Abox are described. Then a measure for computing the degree of similarity of the two ABoxes, representing the abstracts, will be presented briefly. This measure delivers the values for the necessary similarity matrix.

Vague features, represented as fuzzy sets, can be compared using a reciprocal fuzzy measure of inclusion, such as

$$incl(A,B) =_{def} inf\{min(1,1-A(x)+B(x)) \mid x \in U\}$$

for arbitrary fuzzy sets *A* and *B*, defined on universe *U*. The measure of equality of two vague attributes F_1 and F_2 may then be defined as

$$F_1 \approx F_2 =_{def} incl(F_1,F_2)\ et_f\ incl\ (F_2,F_1),$$

where et_f denotes a fuzzy conjunction. The choice of this measure is not compelling, other measures of equality based on fuzzy connectives can be used. Which of these are suitable in a concrete modelling task cannot be decided *a priori*. Figure 6 illustrates two fuzzy sets representing vague features which share some objects to a certain degree and therefore can be regarded as similar to a certain degree.

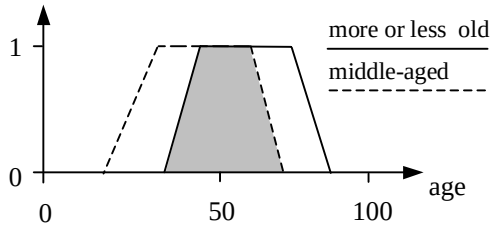

Figure 6: Fuzzy sets representing similar vague features

Note that besides comparing attributes, i.e. vague features, and roles it is possible to measure the similarity of concepts itself. Using the definition of concepts from the knowledge base the subsumption relations of the conceptual knowledge can be computed which will produce a "vague" subsumption network with the edges between concepts being weighted to their degree of similarity (cf. [38]). Concepts, e.g. lymphoma (tumour attacking the lymph gland), can be described by introducing their specific (possibly vague) characteristics. Thus, lymphoma could be characterised by their degree of malignity and histological and cytological criteria. A simple definition of the concept "tumour" could be given as follows:

tumour := lump ⊓ restrict (degree_of_malignity, high).

This leads to a point where certain concepts are recognized as more similar than others. Figure 7 visualizes the numerical similarity value between different concepts. Note, that the semantics underlying this hierarchy are in some way different from networks displaying an 'is-a' hierarchy.

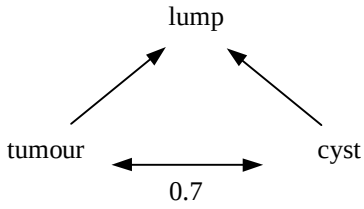

Figure 7: Vague network of concepts

Now pairs of objects have to be compared. Obviously, in our simple example we have pairs like HAS_INTENSITY (RISK*, HIGH) – HAS_INTENSITY (RISK*, LOW) or AGED (PATIENT*, YOUNG) – AGED (PATIENT*, OLD), which can be compared using the ideas shown above. In order to get a single degree of similarity, denoted by α with $\alpha \in [0,1]$, the following procedure has to be applied: Let $B_{max} =_{def} \min(|A_1|, |A_2|)$ denote the maximal number of comparable constructs, i.e. concepts which have a semantic relation, where $|A_i|$ denotes the cardinality of ABox A_i. Choose the ABox with minimal cardinality, say A_1. For $1 \leq i \leq B_{max}$ choose construct κ_i^1 from ABox A_1 and choose a comparable construct for κ_i^1, say κ_i^2. Then, compute the degree of similarity for κ_i^1 and κ_i, denoted by α_i, and if there is no matching construct, set $\alpha_i =_{def} 0$, respectively. The degree of similarity of A_i and A_2, i.e. the similarity of the related abstracts, is defined as

$$\alpha =_{def} \frac{\sum \alpha_i}{\max(|A_1|, |A_2|)}.$$

By applying this procedure to every pair of abstracts, we obtain a similarity matrix for the abstracts in the database. The advantage of this knowledge based comparison is that the resulting degree of similarity is more comprehensible for the user than a pure heuristic measure. There are several approaches, cf. [15], which use semantic networks as a basis for comparison. Although the loss of the ability to compare concepts in a semantic way is a drawback, the basic concepts proposed here can be combined with those heuristic methods working on semantic networks. Thus, an expressive measure of similarity can be achieved which can now directly be used to calculate a matrix of similarity as a basis for clustering the documents regarding to their semantic resemblance. This matrix represents the overall similarity of each pair of documents abstracts.

5 Calculating a Semantic Space of Documents

Up to now we have analysed the document collection regarding its semantics. This step's result was a matrix of similarity values for each pair of documents. The next goal is to analyse and visualize the inherent structure of the collection provided by the similarity information. The interface between the two logical components of the framework is a semantic space of documents, i.e. a multi-dimensional space where the similarity information of the document collection is encoded (cf. section 3).

This space can be generated by discovering the spatial structure of the similarity information, i.e. each object (document) of the collection is mapped to a point in an m-dimensional space preserving the similarity information as good as possible. Indeed, this is exactly the task of metric Multi-Dimensional Scaling (MDS) [37]. In MDS, the objects are mapped into m-space minimizing the relative error of the distances in m-space regarding the 'true' distances of the objects (figure 8). Thus, the similarity values s have to be transformed into distance values d. In cases where the similarity values fall into a predefined range with a fixed maximal similarity value s_{max} a linear conversion can be chosen, e.g. $d =_{def} s_{max} - s$. Some retrieval functions calculate similarity values where no maximal value is defined. In these cases a nonlinear inversion transformation like $d =_{def} b^{-s}$ for some fixed $b > 1$ has to be applied (cf. [6]).

There are several algorithms for MDS. The basic method is described in [21]. It is important to note that MDS does not require the distances between object to respect the triangular inequality since it

minimizes the error between given distances and distances of mapped objects. In [11] a method different from MDS is introduced which performs the mapping in linear time with still satisfactory results if the distance measure used satisfies the properties of a metric. This algorithm was used in our case study presented in section 7 of this paper.

Calculating this semantic space of documents may be useful even when the documents themselves are indexed using real-valued weight vectors. In this case MDS serves as a dimensionality reduction of the document space.

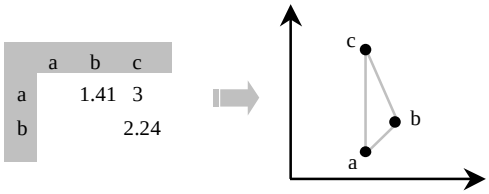

Figure 8: Mapping objects into m-space based on their distances

6 Analysing the Structure of the Document Space

The goal of this work is to detect the inherent semantic structure of a specialized document collection. At this point we have generated a semantic space of documents which reflects the document similarities. This space serves as the input to the next logical component of the structuring model, concerned with analysing the structure of the collection, which is performed by a cluster analysis method.

The task of a cluster analysis is to divide a set of multidimensional objects – here representing the documents – into groups of similar objects, where the objects in each group are as similar as possible and the overall dissimilarity of groups is as distinct as possible [10]. In general, it has to be assumed that there is no further information about a pre-classification of documents. In particular, the number of document groups, their arrangement or naming is not known. This leads to the application of unsupervised learning methods. In this work we use the neural network approach of Kohonen's self-organizing feature maps (SOFM) [18, 19], which will be explained in the next section.

There are three reasons why we use this approach for analysing and visualizing the semantic structure of the document collection (cf. [3]): First, SOFMs map high dimensional feature vectors into two dimensions without losing too much topological information. Second, we can apply a very expressive visualization technique which is based on SOFMs and leads to the semantic map of documents proposed in section 2. The third reason concerns an extended retrieval feature: The generalization property of the neural model allows the association of *a priori* unknown input vectors in a useful manner. This means that documents which are mapped to the semantic document space but were not used during the set up of the document map can be associated with their appropriate group. Thus, new documents can be fed into the structure without re-training the map or can serve as prototype objects to identify relevant groups in the map.

Self-Organizing Feature Maps

The idea of using Kohonen's feature maps is that they order the document representatives from the space $\mathbb{R}^m$ according to their similarity in two dimensions. This arrangement is realized in a self-organizing manner and preserves the distance relationships between the input patterns as good as possible.

The architecture of the neural network is as follows: The model consists of one layer of active units which are disposed in a two dimensional grid without being connected with each other. Each unit i in the grid is linked with all m units of the input layer by means of m weighted edges (figure 9), formally realized by a weight vector w_i. Initially, random numbers are assigned to the components of this weight vector. The output function calculated by each unit i at the position (i_1,i_2) in the grid measures the distance $\eta_i(x, w_i)$ between the input pattern x and the unit's weight vector w_i, for example the Euclidian distance.

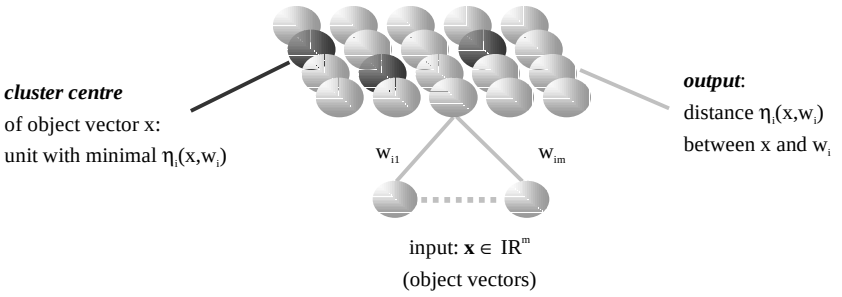

Figure 9: Architecture of SOFMs

The training of the network is unsupervised, i.e. there are no external signals which are used for "teaching" the SOFM. During the learning process, a single unit will be determined for each object vector x at each time, where the unit's vector w_i is most similar to x. Such a winning unit is called the *cluster center* of the considered object vector. The weight vector of the cluster center and the weight vectors of units in a certain surrounding, defined by a neighborhood function, now are shifted towards the input vector. The amount of shifting depends on a learning rate, the difference between w_i and x and the unit's position in the area surrounding the cluster center. Both, learning rate and the size of the area defined by the neighborhood function decrease in time.

The major advantage of the SOFM is gained by the observation that after the learning process the relative positions of different cluster centers towards each other in the grid show the similarity between corresponding object vectors. Furthermore, the positions of all weight vectors w_i are ordered in the grid according to their similarity. Therefore we can expect that the cluster centers are arranged in the same regions of the grid, if the corresponding object vectors form clusters in the input space. Thus, the SOFM has "learned" the structure of the semantic document space. For interpreting the feature map it has to be detected to which areas the input patterns used for training, i.e. the object vectors representing the documents, have been mapped. But without knowledge about cluster membership of the object vectors it is difficult to identify those regions in the grid. This problem will be solved in the following section.

Visualizing the Structure of the Semantic Document Space

In [34] a graphical method for visualizing the information encoded in the neural network is introduced which directly uses the topological properties of SOFMs. For each grid point $i = (i_1, i_2)$ a value is calculated which exhibits the greatest similarity of the weight vector w_i of unit i and all object vectors x_1, x_2, ..., x_n from the training set according to

$$P[i_1][i_2] =_{def} \min_{1 \le k \le n} \{\eta_i(x_k, w_i)\}.$$

The resulting matrix P reflects the density of the document space which is encoded in the weight vectors w_i of the trained SOFM: Units in the grid with weight vectors which are rather dissimilar to all vectors from the input space were trained rarely according to the learning algorithm. Those weight vectors represent the "empty space" between document clusters. In contrast, clusters of similar input patterns lead to a more intensive training of a particular region in the grid. The weight vectors in these regions can be expected to be close to each other and their minimal distance to the input patterns will be rather low. Thus, high values in P separate regions of similar objects. A deeper theoretical discussion regarding the density of weights and object vectors can be found in [19, 31].

The information provided by the matrix P can be visualized by assigning a shade of grey to the values in P: Using a linear function that maps the minimum value to white and the maximum value to black it is possible to visually detect regions of similar documents. These regions will be displayed as bright shaded areas, separated by dark shades corresponding to large numbers in P.

Other graphical approaches for visualizing the information provided by the SOM (cf. [16, 20, 41]) display the distances between the weight vectors w_i of neighboring units. The results received in this way are comparable to the one gained by the P-matrix, if most of the weight vectors w_i are very similar or

almost equal to the object vectors. In contrast, *P* still provides good results, if the vectors w_i are uniformly distributed in those subspaces of the input space which correspond to the clusters to be discovered.

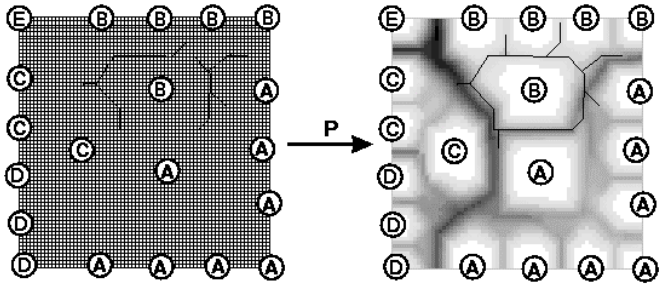

Figure 10: Assigning shades of grey to the values in *P* (illustration courtesy of S. Sklorz). High values in *P* separate regions of similar documents.

The final task is to mark the places in the map where the documents of the collection belong to. Therefore, the unit of the neural network representing a particular document has to be identified. This is done by determining the unit in the grid associated with the weight vector which is most similar to the vector from the semantic document space which represents the document. The corresponding position in the map can now be colored and inscribed by the document title or a short document summary.

Figure 11 illustrates the visualization of a simple two-dimensional document space. The SOFM maps the document representatives to the grid by optimizing their relative positions. It is important to note that the distances on the map do not represent the absolute value of the documents' dissimilarity. Rather, the 'document landscape' gives an impression of the global document relationships within the collection.

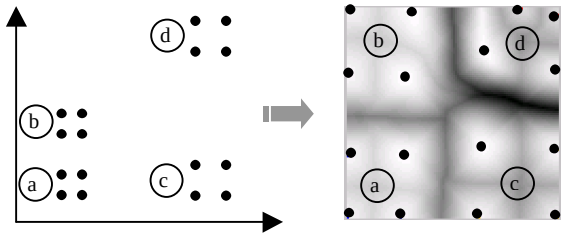

Figure 11: The document map presents the global relationship of the documents.

7 Case Study: A Semantic Map of Use Cases

As pointed out at the beginning of section 4.1 a semantic structuring of use cases can support the requirements engineers in a complex requirements analysis task: The 'use cases landscape' helps to learn more about relationships between modules even beyond the structures predefined by the engineers. To assess the benefits of a semantic structuring in a real project environment the methods presented here were applied to the CAPE-OPEN project. CAPE-OPEN defines standards for chemical process engineering simulators. Therefore, use cases are used to describe the functionality of simulator objects. The next sections describe the parameters of a semantic structuring of use cases, present the results and discuss the quality of the automatically derived structures.

Experimental Setting

Within the CAPE-OPEN project 158 use cases were designed. For gaining a semantic structuring the complete documents were indexed using the vector space model. The indexing vocabulary was gained by eliminating stopwords from the texts and stemming the remaining terms (cf. section 4.1). Besides using the SMART indexing model simple term histograms were used and all description vectors were normalized. The similarity functions used were the cosine measure and a measure based on the Euclidian distance of the vectors. Both indexing types and similarity measures led to good results, but normalized term histograms in conjunction with the cosine measure of similarity produced the clearest structures.

For calculating the semantic space of documents the method presented in [11] was applied (cf. section 5). The document maps were generated using the MIDAS data mining system [13]. This tool uses a self-organizing feature map for detecting structures of feature patterns and realizes the visualization method based on the *P*-matrix (cf. section 6). The document landscapes presented next were obtained using normalized term histogram vectors and the cosine measure of similarity.

Some Results

Figure 12 shows a semantic map of use cases which was automatically generated. Only the document titles were assigned manually due to the limited features of the early system prototype. Looking at the 'use case landscape' the user can identify four major areas, each of which is subdivided into smaller areas containing sub-groups of documents. Zooming into the marked group leads to an area of use cases describing physical processing unit operations. Whereas the inscribed documents at the top are concerned with calculation matters the sub-group below deals with handling 'unit' software objects.

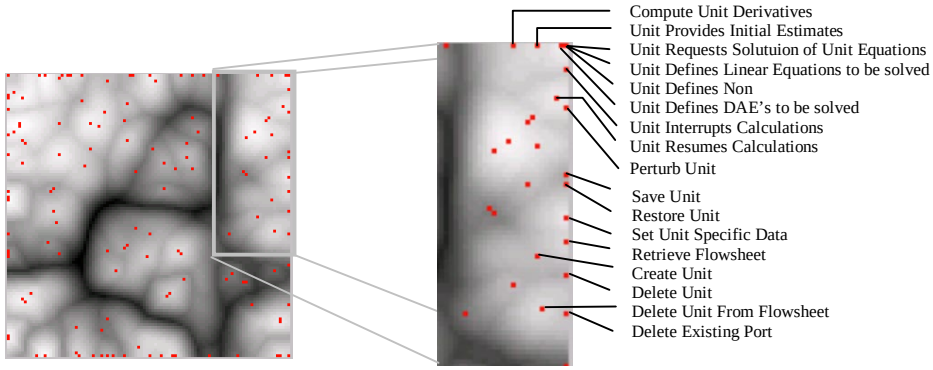

Figure 12: Use case landscape. The detail shows a part of the 'Unit' use case group.

Concerning the quality of the structure it turns out that the automatically derived use case landscape rather faithfully reconstructs the *a priori* structure shown in figure 14. Furthermore, it provides additional information about the relationship of sub-groups. Figure 13 presents the map of use cases where the document representatives are marked by an icon which identifies the sub-group the document belongs to. It can be recognized that the group of 'solver' use cases is more similar to the use cases describing physical processing 'units' than the 'solver' use cases are to the 'GAT' use cases describing a graph analysis tool. Moreover, single use cases from the sub-groups are related to each other in a more detailed manner than shown by the predefined hierarchy.

8 Conclusion and Future Work

A modular approach for gaining a semantic structuring of specialized document collections was proposed. On the one hand, this framework allows the incorporation of different retrieval models for generating high-quality similarity values for documents. On the other hand, it offers the possibility to apply varying methods of structure analysis and visualization techniques depending on the requirements of a certain application context.

The benefit of semantic structuring itself is twofold: It serves as an information retrieval paradigm as well as a support for analysing textual knowledge sources. A case study has shown the usability and quality of a semantic document map.

In our future research we will examine the requirements of more specialized document collections deeper. Among the considered collections are technical manuals and management documents. An important topic is to combine meta-rules for structuring, further background knowledge and statistic retrieval models, so that different types of technical documents with different levels of 'knowledge intensity' can be captured in a dynamic model.

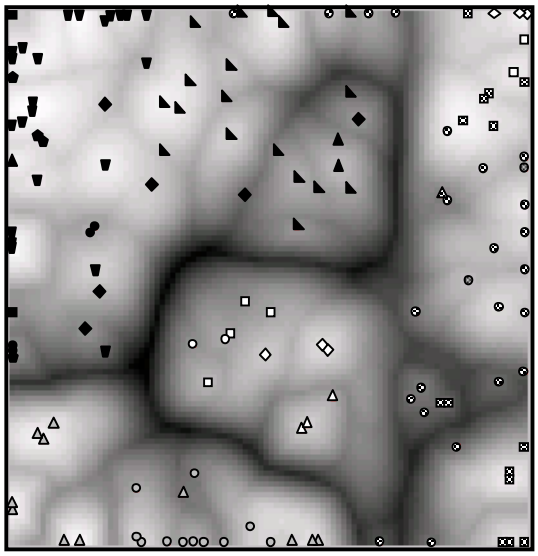

Figure 13: Landscape of use cases, marked by their group membership in the *a priori* structure.

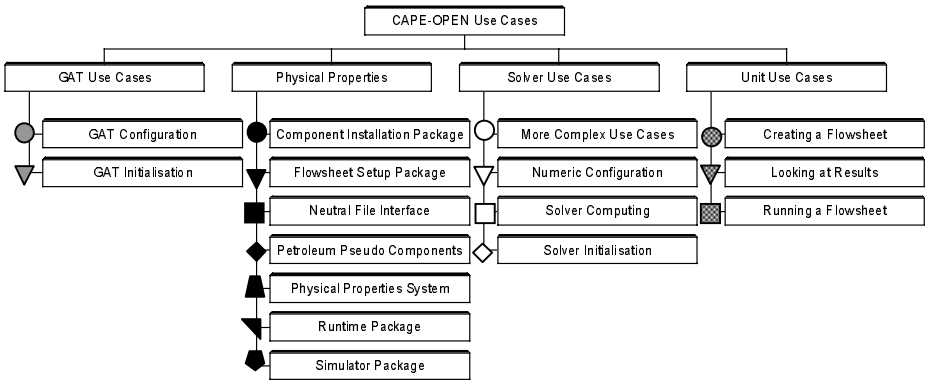

Figure 14: *A priori* structure of the use cases.

Acknowledgements. This work was supported by the Deutsche Forschungsgemeinschaft (DFG) in its focused doctoral programme on Informatics and Engineering at RWTH Aachen and in RWTH Aachen's Collaborative Research Centre IMPROVE. Furthermore, parts of this work were supported by the Commission of the European Union under BRITE-EURAM project CAPE-OPEN and under ESPRIT Long Term Research Project CREWS. The authors wish to thank C. Tresp for productive discussions about vague knowledge representation.

9 References

1. Allan, James, Leouski, Anton V., Swan, Russell C.: Interactive Cluster Visualization for Information Retrieval. Tech. Rep. IR-116, Center for Intelligent Information Retrieval, University of Massachusetts (1997)
2. M. Baumeister, A. Becks, S. Sklorz, Ch. Tresp, U. Tüben: Indexing Medical Abstract Databases. Proceedings of the European Workshop on Multimedia Technology in Medical Training, ABI, Bd. 20, Aachen, Germany (1997)
3. A. Becks, S. Sklorz, C. Tresp: Semantic Structuring and Visual Querying of Document Abstracts in Digital Libraries. In: Lecture Notes in Computer Science 1513: Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries, Crete, Greece, 1998, pp. 443-458
4. R. Bentley, Th. Horstmann, J. Trevor: The World Wide Web as enabling technology for CSCW. The Journal of Collaborative Computing 2-3, Kluwer Academic Publishers, Amsterdam, (1997)

5. R. Burke, K. Hammond, V. Kulyukin, S. Lytinen, N. Tomuro, S. Schoenberg: Question Answering from Frequently-Asked Question Files: Experiences with the FAQ Finder System. The University of Chicago, Technical Report TR-97-05 (1997)
6. H. Chavarria Garza, R. Korfhage: Retrieval Improvement by interaction of queries and user profiles. In: Proceedings of COMPSAC '82, 6th International Conference on Computer Software and Applications, Chicago, 1982
7. Chen, Hsinchun. Collaborative Systems: Solving the Vocabulary Problem. IEEE Computer, May (1994)
8. Cutting, D.R., Karger, D.R., Pedersen, J.O., Tukey, J.W. Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections. Proc. of the 15th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, Copenhagen (1992)
9. Deerwester, Scott, Dumais, Susan T., Harshman, Richard. Indexing by Latent Semantic Analysis. Journal of the Society for Information Science, 41(6), 1990, pp. 391-407
10. G. Deichsel, H.-J. Trampisch: Clusteranalyse und Diskriminanzanalyse. Gustav Fischer Verlag, Stuttgart, 1985
11. Faloutsos, Lin: Fastmap: A Fast Algorithm for Indexing, Data-Mining and Vizualization of Traditional and Multimedia Datasets. Proceedings of the Int. Conf. on Management of Data (SIGMOD'95); 2(24), June 1995
12. U.M. Fayyad, G. Piatetsky-Shaprio, G. Smyth, P. Uthurusamy (eds.): Advances in Knowledge Discovery in Data Mining, Cambridge, USA, 1996
13. M. Gebhardt, M. Jarke, M.A. Jeusfeld, C. Quix, and S. Sklorz: Tools for Data Warehouse Quality. Proceedings of the 10th International Conference on Scientific and Statistical Database Management (SSDBM '98), Capri, 1998, pp.229-232
14. M.R. Girardi, B. Ibrahim: An approach to improve the effectiveness of software retrieval. Proceedings of the 3rd Irvine Software Symposium, Irvine, California (1993)
15. K. Hammond, R. Burke, C. Martin, S. Lytinen: FAQ Finder: A Case-Based Approach to Knowledge Navigation. AAAI Spring Symposium on Information Gathering from Distributed, Heterogeneous Environments, Stanford, CA. (1995)
16. J. Iivarinen, T. Kohonen, J. Kangas, S. Kaski: Visualizing the clusters on the self-organizing map. Proc. of the Conf. on AI Research in Finland, Helsinki (1994), pp. 122 –126
17. D. Keim, H.-P. Kriegel: Visualization Techniques for Mining Large Databases: A Comparison. In IEEE Transactions on Knowledge and Data Engineering, Vol. 8, No. 6, December 1996
18. T. Kohonen: Clustering , Taxonomy and Topological Maps of Patterns. Proceedings of the 6th Int. Conf. on Pattern Recognition, München (1982)
19. T. Kohonen: Self-Organizing Maps. Springer, Berlin, 2nd Edition (1995)
20. M. A. Kraaijveld, J. Mao, A. K. Jain: A nonlinear projection method based on Kohonen's topology preserving maps. IEEE Transactions on Neural Networks, Vol. 6, pp. 548 – 559
21. J.B. Kruskal, M. Wish: Multidimensional scaling. SAGE publications, Beverly Hills, 1978
22. K. Lagus, T. Honkela, S. Kaski, T. Kohonen: Self-Organizing Maps of Document Collections: A New Approach to Interactive Exploration. Proc. of the Second International Conference on Knowledge Discovery and Data Mining, AAAI Press, California (1996)
23. C. Larman: Applying UML and Patterns – An Introduction to Object-Oriented Analysis and Design. Prentice Hall, New Jersey, 1998
24. X. Lin, D. Soergel, G. Marchionini: A Self-Organizing Map for Information Retrieval. Proc. of SIGIR '91, Chicago, USA, pp. 262-269
25. Meghini, Carlo, Straccia, Umberto: A Relevance Terminological Logic for Information Retrieval. Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Zürich, Switzerland (1996)
26. B. Nebel: Reasoning and Revision in Hybrid Representation Systems. Springer (1990)
27. M.F. Porter: An Algorithm for suffix stripping. Program 14, pp. 130-137, 1980
28. G. Probst, S. Raub, K. Romhardt: Wissen managen. Gabler, Wiesbaden (1997)
29. V. Raghavan, S. Wong: A Critical Analysis of Vector Space Model for Information Retrieval. Journal of the American Society for Information Science, 37(5), 1986
30. C.J. van Rijsbergen: Information Retrieval. 2nd edition, Butterworths, London (1979)
31. H. Ritter: Asymptotic level density for a class of vector quantization process. IEEE Transactions on Neural Networks, Vol 2, 1991
32. G. Salton (Ed.): The SMART Retrieval System – Experiments in Automatic Document Processing. Prentice Hall, New Jersey (1971)
33. Schmiedel, Albrecht: Semantic Indexing Based on Description Logics. In: Reasoning about Structured Objects: Knowledge Representation Meets Databases, Proc. of 1st Workshop KRDB '94. Saarbrücken, Germany (1994)
34. S. Sklorz: A Method for Data Analysis based on Self Organizing Feature Maps., Proceedings of the World Automation Congress (WAC '96), Vol.5 TSI Press Series, 611-616, ISBN 1-889335-02-9, Albuquerque, USA (1996)
35. Straccia, Umberto: A Fuzzy Description Logic. In: Proceedings of AAAI-98 (1998)
36. R. Swan, J. Allan: Improving Interactive Information Retrieval Effectiveness with 3-D Graphics. Technical report IR-100, Dept. of CS, University of Massachusetts at Amherst, 1996
37. W.S. Torgerson: Mulidimensional scaling. Psychometrika, 17, pp. 401-419, 1952
38. C. Tresp, A. Becks, R. Klinkenberg, J. Hiltner: Knowledge Representation in a World with Vague Concepts, In: Intelligent Systems: A Semiotic Perspective, Gaithersburg (1996)

39. C. Tresp, R. Molitor: A description logic for vague knowledge. In : Proceedings of the 13th biennial European Conference on Artificial Intelligence (ECAI'98), pp. 361--365, Brighton, UK, 1998. J. Wiley and Sons.
40. C. Tresp, U. Tüben: Medical terminology processing for a tutoring system. International Conference on Computational Intelligence and Multimedia Applications (ICCIMA98), Monash Univ., Australia (1998)
41. A. Ultsch, H. Simon: Exploratory Data Analysis: Using Kohonen Networks on Transputers. Technical Report, No. 329, University of Dortmund (1989)
42. M. Schröder: Erwartungsgestützte Analyse medizinischer Befundungstexte. Ein wissensbasiertes Modell zur Sprachverarbeitung. Infix DISKI, Sankt Augustin (1995)
43. J. Yen: Generalizing Term Subsumption Languages to Fuzzy Logic. In: Proceedings of IJCAI-91, Sydney, Australia (1991)
44. L.A. Zadeh: Fuzzy Sets. In: Information and Control, 8, 1965
45. L.A. Zadeh: A fuzzy-set-theoretic interpretation of linguistic hedges. Journal of Cybernetics, 2:4-34, 1972