Automatic Text Representation, Classification and Labeling in European Law

Erich Schweighofer
Institute of Public International Law
University of Vienna Research Center for Computers and Law
Universitätsstr. 2, A-1090 Vienna, Austria
Erich.Schweighofer@univie.ac.at

Andreas Rauber, Michael Dittenbach
Institute for Software Technology
Vienna University of Technology
Favoritenstr. 9-11 / 188, A-1040 Vienna, Austria
andi,mbach@ifs.tuwien.ac.at

ABSTRACT

The huge text archives and retrieval systems of legal information have not achieved yet the representation in the well-known subject-oriented structure of legal commentaries. Content-based classification and text analysis remains a high priority research topic. In the joint KONTERM, SOM and LabelSOM projects, learning techniques of neural networks are used to achieve similar high compression rates of classification and analysis like in manual legal indexing. The produced maps of legal text corpora cluster related documents in units that are described with automatically selected descriptors. Extensive tests with text corpora in European case law have shown the feasibility of this approach. Classification and labeling proved very helpful for legal research. The *Growing Hierarchical Self-Organizing Map* represents very interesting generalities and specialties of legal text corpora. The segmentation into document parts improved very much the quality of labeling. The next challenge would be a change from *tf* × *idf* vector representation to a modified vector representation taking into account thesauri or ontologies considering learned properties of legal text corpora.

1. INTRODUCTION

As ever larger amounts of legal documents become available electronically, ways for organizing these documents to provide comfortable means of access gain importance. Conventional information retrieval systems do not fulfill the requirements of lawyers and citizens regarding access to, location, and representation of legal information, leaving the information crisis in law still unsolved [25]. The classical way of information searching requires users to specify their information needs in terms of complex queries resulting in usually large lists of documents being retrieved and presented as a list sorted by some relevance criterion. The well-known subject-oriented structure that users are accustomed to is lost, leaving it to them to locate the actually relevant documents. These deficiencies are addressed in numerous research projects trying to introduce content-based information into document organization and representation.

A very promising approach to automatically detect topical similarity and to structure a document collection accordingly uses the *Self-Organizing Map* (*SOM*) [8], a popular unsupervised neural network to cluster documents. This method has shown to be very helpful also in legal applications. Yet, some peculiarities of the legal domain have to be addressed.

Legal language has some specialties that must be considered for classification and summarization. For example, statistical characteristics of important words are different from other text corpora. News articles repeat quite often the most important message. In law, relevant terms may appear only with one occurrence besides lengthy argumentation for other points of view. Therefore, the selection of an appropriate vector representation remains tricky. Length of documents may require a segmentation into structural parts. The weighting of terms is challenging because very special words or phrases have to be treated with particular attention that is not statistically evident. Furthermore, the content-based classification tends to become somewhat distorted for documents covering different topics or, especially for longer documents, when having specific sections with a rather strict, document-type based structure. In such a case, documents may not only be organized according to their content, but also to a large degree by their structure or type of document. We thus need to identify ways to provide a better content representation for legal documents.

Secondly, some deficiencies relating to the application of the *SOM* for automatically organizing, representing, and accessing legal information systems have to be noted. These relate, on the one hand, to the size of the resulting map, which tends to be unhandily large for big document collections containing hundreds or thousands of articles. We rather would prefer smaller maps representing various dedicated sections of the archive. Apart from the problem of the mere size of the resulting maps, no hierarchical structure of the various topics located on the map can be derived from the given representation. Again, this turns out to be disorienting for larger collections. Users definitely would benefit from a hierarchical structuring of the document archive, where each successive layer provides a more detailed view of the relevant subset of documents. As the precise topical hierarchy existing in such a collection is usually not known

ICAIL01 2001, St. Louis, MI
Copyright 2001 ACM 0-89791-88-6/97/05 ...$5.00.

in advance, the program should be capable of automatically detecting the topic hierarchy and adapting the architecture of the system accordingly.

Last, but not least, users require additional information about the various topical clusters in order to help them find orientation on the resulting map. We thus want to automatically extract labels describing the various topical clusters or branches in the hierarchy of legal documents. The next step would be automatic indexing, extracting, or assigning automatically a set of content identifiers to documents. Automatic summarization - the art of abstracting key content - would be the final step in this ambitious research agenda [6].

All these task are the aim of the our joint research within the KONTERM, SOM and LabelSOM projects. The focus of the KONTERM projects has moved from word sense disambiguation and semiautomatic text analysis [22] to automatic indexing and summarization. In former research, we have proven the feasibility of classification and labeling for legal text corpora using *SOM* and *LabelSOM* [23, 24]. In this paper we present an extended architecture for organizing legal documents to aid the user in finding information in legal document repositories. Instead of using a fix-sized flat *SOM* we developed the *Growing Hierarchical Self-Organizing Map* (*GHSOM*), a flexible and dynamic architecture capable of adapting both its size as well as the hierarchical layout according to the given data collection. As a result, a document collection is represented by a tree of interconnected individual *Self-Organizing Maps*, each of which describes parts of the archive in more detail. Furthermore, we no longer represent documents as single entities, but rather split them into various segments to allow multiple placements of the same document in various topical branches if different topics are touched upon. The labels of the various topical sections extracted by the *LabelSOM* technique [17] furthermore provide a description of the topical clusters guiding the users on their way to the desired documents.

The remainder of this paper is structured as follows. We start with a review of some related work in Section 2. In Section 3 we provide a brief introduction to the feature extraction and content representation process, followed by a review of the architecture and training process of a *Self-Organizing Map* in Section 4. We then present the new *Growing Hierarchical Self-Organizing Map* architecture in detail in Section 5, followed by the *LabelSOM* technique used for extracting descriptive keywords for both types of maps in Section 6. This is followed by a detailed presentation of results obtained from a large-scale corpus of documents from the CELEX and EUR-Lex databases in Section 7. Section 8 summarizes the lessons learned and points at future steps necessary to address the challenges identified with the current system.

2. RELATED WORK

Knowledge representation was and still is one of the main tasks of legal informatics. Knowing sufficient bibliographic data, access to every legal document has become quite easy. Searching these huge digital libraries still heavily relies on mastering the terminology. The neglected field of classification, indexing and summarization may show some solution for this problem.

Lawyers are used to highly developed indexing instruments like hand books, commentaries or citations. Automating these tasks for a huge legal information system is the main challenge for research in legal knowledge representation. The high quality of commentaries pose a challenge for any automatic or semiautomatic approach offering - at least at the moment - a limited compression but also a more faceted picture.

As vectors remain the most efficient representation for statistical analysis, an efficient solution has to be found for the representation of diverse legal documents. In this setting the *tf* × *idf* weighting scheme [21], well-known in information retrieval, is still state of the art for content representation. As successfully shown in the FLEXICON project [26], a term extraction module recognizing concepts, case citations, statute citations and fact phrases lead to a very helpful structured document profile. This profile is transformed into a weighted vector [21] allowing searching for related documents or problems. Another approach has been developed within the KONTERM project [14]. The various documents are represented as feature vectors of the form $x = (t_1, ..., t_m, c_1, ..., c_n, m_1, ..., m_o)$. The t_i represent terms extracted from the full text of the document, the c_i are the context-sensitive rules, and the m_i represent the meta rules associated with the document. Like in the FLEXICON project, concepts are recognized by matching a list but also by applying some heuristic rules. Linguistic templates are found by context-sensitive rules. The wording of rules is facilitated allowing probabilistic expressions, too. Meta rules represent a concept that must be defined as a combination of rules occurring in the same document or section of a document. The result is a weighted vector giving more importance to linguistic templates and meta rules. In both approaches, evaluation results were very promising.

Besides the knowledge base of thesauri and rules like in FLEXICON or KONTERM projects, the more developed knowledge representation of case-based reasoning has become a starting point. In the SPIRE project, inference net representations of important cases were used as search for similar documents [3]. In the SMILE project, factors should be found using techniques of machine learning [2]. More developed linguistic representations can be found in the projects ILAM [10] and SALOMON [16] focusing on automatic abstracting. These results are not yet applicable to unsupervised automatic indexing due to problems of knowledge acquisition or necessary supervised learning but will provide important guidance if possible limits of automatic indexing require legal ontologies for improvement (for ontologies in law see [29]).

A very fascinating approach to the concept of document organization uses neural networks, specifically the *Self-Organizing Map* (*SOM*) [7, 8], a popular unsupervised neural network, to organize documents by their content. The resulting representation allows a content-based browsing and access to large document collections [9, 13, 18]. While legal documents pose several additional challenges to the automatic organization of document collection, as will be described in more detail below, we find this approach to work well for legal document collections [14, 19, 23, 24]. Using these maps we can provide a convenient way for accessing legal document collections allowing users to find similar documents located in neighboring regions of the map.

3. FEATURE EXTRACTION AND DOCUMENT REPRESENTATION

In order to allow content-based classification of documents we need to obtain a representation of their content. One of the most common representations uses word frequency counts based on full text indexing. A list of all words present in a document collection is created to span the feature space within which the documents are represented. While handcrafted stop word lists allow for specific exclusion of frequently used words, statistical measures may be used to serve the same purpose in a more automatic way. For our experiments we thus remove all words that appear either in too many documents within a collection (e.g. say in more than 50% of all documents) or in too few (say, less than 5 documents) as these words do not contribute to content representation. The words are further weighted according to the standard *tf* × *idf*, i.e. term frequency times inverse document frequency, weighting scheme [21]. This weighting scheme assigns high values to words that are considered important for content representation.

However, when analyzing the legal domain we find documents to be highly structured, i.e. to consist of several segments, within each of which certain peculiarities in terms of vocabulary used can be observed. As legal documents are highly structured and contain very diverse issues, a good segmentation allows a better classification of the various parts of the documents. To give just one example, a document of the European Court of Justice consists of the title (case no., parties, eventually submitting court, short content description), the summary, the heading, the reasons (submissions, presented arguments, reasons of the court) and the ruling of the court. The reasons are numbered per paragraph.

In order to allow content-based classification within each of these structural segments, documents are split into sections and treated as independent pieces of information for the further classification process. This allows documents to be assigned to multiple sections in the resulting topical hierarchy without structural information overruling the content-based classification.

4. THE SELF-ORGANIZING MAP

The *Self-Organizing Map* [8] basically provides a form of cluster analysis by producing a mapping of high-dimensional input data $x, x \in \Re^n$ onto a usually 2-dimensional output space while preserving the topological relationships between the input data items as faithfully as possible. This model consists of a set of units, which are arranged in some topology where the most common choice is a two-dimensional grid. Each of the units i is assigned a weight vector m_i of the same dimension as the input data, $m_i \in \Re^n$. In the initial setup of the model prior to training, the weight vectors are filled with random values.

During each learning step, the unit c with the highest activity level, i.e. the *winner* c with respect to a randomly selected input pattern x, is adapted in a way that it will exhibit an even higher activity level at future presentations of that specific input pattern. Commonly, the activity level of a unit is based on the Euclidean distance between the input pattern and that unit's weight vector. The unit showing the lowest Euclidean distance between its weight vector and the presented input vector is selected as the winner. Hence, the selection of the winner c may be written as given in Expression (1).

$$c = \arg\min_i \{||x - m_i||\} \qquad (1)$$

Adaptation takes place at each learning iteration and is performed as a gradual reduction of the difference between the respective components of the input vector and the weight vector. The amount of adaptation is guided by a learning rate α that is gradually decreasing in the course of time. This decreasing nature of adaptation strength ensures large adaptation steps in the beginning of the learning process where the weight vectors have to be tuned from their random initialization towards the actual requirements of the input space. The ever smaller adaptation steps towards the end of the learning process enable a fine-tuned input space representation.

As an extension to standard competitive learning, units in a time-varying and gradually decreasing neighborhood around the winner are adapted, too. Pragmatically speaking, during the learning steps of the *Self-Organizing Map* a set of units around the winner is tuned towards the currently presented input pattern enabling a spatial arrangement of the input patterns such that alike inputs are mapped onto regions close to each other in the grid of output units. Thus, the training process of the *Self-Organizing Map* results in a topological ordering of the input patterns. According to [20] we may thus refer to the *Self-Organizing Map* as a neural network model performing a spatially smooth version of k-means clustering.

The neighborhood of units around the winner can be described implicitly by means of a neighborhood-kernel h_{ci} taking into account the distance—in terms of the output space—between unit i under consideration and unit c, the winner of the current learning iteration. This neighborhood-kernel assigns scalars in the range of [0, 1] that are used to determine the amount of adaptation ensuring that nearby units are adapted more strongly than units farther away from the winner. A Gaussian may be used to define the neighborhood-kernel as given in Expression (2) where $||r_c - r_i||$ denotes the distance between units c and i within the output space, with r_i representing the two-dimensional vector pointing to the location of unit i within the grid.

$$h_{ci}(t) = \exp\left(-\frac{||r_c - r_i||}{2 \cdot \delta(t)^2}\right) \qquad (2)$$

It is common practice that in the beginning of the learning process the neighborhood-kernel is selected large enough to cover a wide area of the output space. The spatial width of the neighborhood-kernel is reduced gradually during the learning process such that towards the end of the learning process just the winner itself is adapted. Such a reduction is done by means of the time-varying parameter δ in Expression (2). This strategy enables the formation of large clusters in the beginning and fine-grained input discrimination towards the end of the learning process.

In combining these principles of *Self-Organizing Map* training, we may write the learning rule as given in Expression (3). Please note that we make use of a discrete time notation with t denoting the current learning iteration. The other parts of this expression are $\alpha(t)$ representing the time-varying learning rate, $h_{ci}(t)$ representing the time-varying

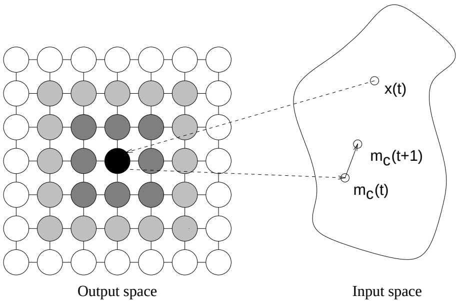


Figure 1: SOM Architecture and training process

neighborhood-kernel, $x(t)$ representing the currently presented input pattern, and $m_i(t)$ denoting the weight vector assigned to unit i.

$$m_i(t+1) = m_i(t) + \alpha(t) \cdot h_{ci}(t) \cdot [x(t) - m_i(t)] \qquad (3)$$

A graphical representation of a *Self-Organizing Map*'s architecture and its learning process is provided in Figure 1. In this figure the output space consists of a square of 49 units, depicted as circles. One input vector $x(t)$ is randomly chosen and mapped onto the grid of output units. In the second step of the learning process, the winner c showing the highest activation is selected. Consider the winner being the unit depicted as the black unit in the figure. The weight vector of the winner, $m_c(t)$, is now moved towards the current input vector. This movement is symbolized in the input space in Figure 1. As a consequence of the adaptation, unit c will produce an even higher activation with respect to input pattern x at the next learning iteration, $t+1$, because the unit's weight vector, $m_c(t+1)$, is now nearer to the input pattern x in terms of the input space. Apart from the winner, adaptation is performed with neighboring units, too. Units that are subject to adaptation are depicted as shaded units in the figure. The shading of the various units corresponds to the amount of adaptation and thus, to the spatial width of the neighborhood-kernel. Generally, units in close vicinity of the winner are adapted more strongly and consequently, they are depicted with a darker shade in the figure.

5. THE GROWING HIERARCHICAL SELF-ORGANIZING MAP

While the *SOM* has proven to be a very suitable tool for detecting structure in high-dimensional data and organizing it accordingly on a two-dimensional output space, some shortcomings have to be mentioned. These include its inability to capture the inherent hierarchical structure of data. Furthermore, the size of the map has to be determined in advance when proper insight into the characteristics of an (unknown) data distribution might not be available. These drawbacks have been addressed separately in several modified architectures of the *SOM*, such as the Incremental Grid Growing or Growing Grid architectures or Hierarchical Feature Maps [1, 5, 15]. However, none of these approaches provide an architecture which fully adapts itself to the characteristics of the input data. To overcome the limitations of both fix-sized and non-hierarchically adaptive architectures we developed the *Growing Hierarchical Self-Organizing Map* (*GHSOM*), which dynamically fits its multi-layered architecture according to the structure of the data [4].

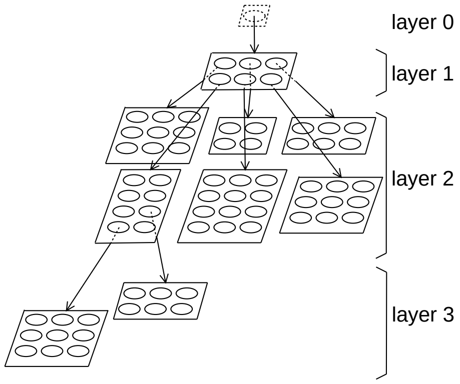


Figure 2: GHSOM: The *GHSOM* evolves to a structure of *SOMs* reflecting the hierarchical structure of the input data.

The *GHSOM* has a hierarchical structure of multiple layers, where each layer consists of several independent growing *Self-Organizing Maps*. Starting from a top-level map, each map, similar to the *Growing Grid* model, grows in size in order to represent a collection of data at a specific level of detail. After a certain improvement of the granularity of data representation is reached, the units are analyzed to see whether they represent the data at a specific minimum level of granularity. Those units that have too diverse input data mapped onto them are expanded to form a new small growing *SOM* at a subsequent layer, where the respective data shall be represented in more detail. These new maps again grow in size until a specified improvement of the quality of data representation is reached. Units representing an already rather homogeneous set of data, on the other hand, will not require any further expansion at subsequent layers. The resulting *GHSOM* thus is fully adaptive to reflect, by its very architecture, the hierarchical structure inherent in the data, allocating more space for the representation of inhomogeneous areas in the input space.

A graphical representation of a *GHSOM* is given in Figure 2. The map in layer 1 consists of 3×2 units and provides a rather rough organization of the main clusters in the input data. The six independent maps in the second layer offer a more detailed view on the data. The input data for one map is the subset which has been mapped onto the corresponding unit in the upper layer. Two units from one of the second layer maps have further been expanded into third-layer maps to provide sufficiently granular input data representation. It has to be noted that the maps have different sizes according to the structure of the data, which relieves us from the burden of predefining the structure of the architecture. The layer 0 serves as a representation for the complete data set and is necessary for the control of the growth process.

5.1 Initial setup

Prior to the training process a "map" in layer 0 consisting of only one unit is created. This unit's weight vector m_0 is initialized as the average of all input vectors and its mean quantization error mqe_0 is computed.

Basically, the mean quantization error mqe_i of a unit i is the deviation between its weight vector and the input vectors mapped onto this very unit. It is calculated as the mean Euclidean distance between its weight vector m_i and the $n_\mathcal{C}$ input vectors x_j which are elements of the set of input vectors $\mathcal{C}_i$ that are mapped onto this unit i:

$$mqe_i = \frac{1}{n_\mathcal{C}} \cdot \sum_{x_j \in \mathcal{C}_i} \|m_i - x_j\|, \qquad n_\mathcal{C} = |\mathcal{C}_i| \tag{4}$$

where $|\cdot|$ denotes the cardinality of a set.

Specifically, the mean quantization error of the single unit at layer 0 is computed as detailed in Expression 5, where $n_\mathcal{I}$ is the number of all input vectors x of the input data set $\mathcal{I}$.

$$mqe_0 = \frac{1}{n_\mathcal{I}} \cdot \sum_{x_i \in \mathcal{I}} \|m_0 - x_i\|, \qquad n_\mathcal{I} = |\mathcal{I}| \tag{5}$$

The value of mqe_0 can be regarded as a measurement of the dissimilarity of all input data. It will play a vital role during the growth process of the neural network, as will be described in the next section.

5.2 Training and growth process of a map

Beneath the layer 0 map a new growing SOM is created with a size of initially 2×2 units. This first layer map is trained according to the standard SOM training procedure as described in Section 4. After a fixed number of training iterations the *mqe's* of all units are analyzed. A high *mqe* shows that for this particular unit the input space is not represented accurately enough. Therefore, new units are needed to increase the quality of input space representation. The unit with the highest *mqe* is thus selected and is denoted as the error unit e. A new row or column of units is inserted in between the error unit and its most dissimilar neighbor. The weight vectors of the new units are initialized as the average of their corresponding neighbors.

More formally, the growth process of a map can be described as follows. Let $\mathcal{C}_i$ be the subset of vectors x_j of the input data that is mapped onto unit i, i.e. $\mathcal{C}_i \subseteq \mathcal{I}, \mathcal{C}_i \neq \emptyset$; and m_i the weight vector of unit i. Then, the error unit e is determined as the unit with the maximum mean quantization error:

$$e = \arg\max_i \left(\frac{1}{n_\mathcal{C}} \cdot \sum_{x_j \in \mathcal{C}_i} \|m_i - x_j\| \right), \qquad n_\mathcal{C} = |\mathcal{C}_i| \tag{6}$$

Following the selection of the error unit, its most dissimilar neighbor d is determined. This is done by comparing the weight vectors of all neighboring units with the weight vector of the error unit e. A complete row or column of units is inserted in between d and e. The weight vectors of these new units are initialized as the means of their respective neighbors.

The growth process continues until the map's mean quantization error, referred to as MQE in capital letters, reaches a certain fraction τ_1 of the mqe_u of the corresponding unit u in the upper layer, i.e. the unit constituting the layer 0 map for the first layer map. The MQE of a map is computed as the mean of all units' mean quantization errors mqe_i (cf. Expression 4) of the subset $\mathcal{U}$ of the maps' units onto which data is mapped:

$$MQE_m = \frac{1}{n_\mathcal{U}} \cdot \sum_{i \in \mathcal{U}} mqe_i, \qquad n_\mathcal{U} = |\mathcal{U}| \tag{7}$$

In general terms, the stopping criterion for the training of a single map m is defined as:

$$MQE_m < \tau_1 \cdot mqe_u \tag{8}$$

where mqe_u is the mean quantization error of the corresponding unit u in the upper layer. Obviously, the smaller the parameter τ_1 is chosen the longer the training will last and the larger the resulting map will be. In case of the first layer map the stopping criterion for the training process is $MQE_1 < \tau_1 \cdot mqe_0$. The parameter τ_1 thus serves as the control parameter for the final size of each map by defining the degree to which each map has to represent the information mapped onto the unit it is based upon in greater detail.

5.3 Reflection of the hierarchical structure

When the training of the map is finished, every unit has to be checked for expansion, i.e. whether or not this unit shall be further refined in a map on the next layer. This means that for units representing a set of too diverse input vectors a new map in the next layer will be created. The threshold for this expansion decision is determined by a second parameter τ_2 which defines the data representation granularity which has to be met by every unit. Expression 9 thus constitutes a global stopping criterion for the training process by defining a minimum quality of data representation required for all units as a fraction of the dissimilarity of all input data described by mqe_0. All units satisfying this criterion do not require any further expansion.

$$mqe_i < \tau_2 \cdot mqe_0 \tag{9}$$

If a unit i fails to meet the stopping criterion specified in Expression 9, i.e. $mqe_i \geq \tau_2 \cdot mqe_0$, then a new small map in the next layer will be created, whereas if the stopping criterion given in Expression 9 holds true for a given unit, no further expansion is required. Please note, that, unlike the stopping criterion for horizontal growth of a map determined by τ_1 and the *mqe* of the according upper layer unit, this second criterion is based solely on mqe_0, i.e. the *mqe* of layer 0, for every unit on all maps.

The input vectors to train the newly added map are the ones mapped onto the unit which has just been expanded. This map will again continue to grow following the procedures detailed in the previous subsection. The whole process is repeated for the subsequent layers until the criterion given in Expression 9 is met by all units in the lowest layers.

The parameter τ_2 thus defines the minimum quality of representation for all units in the lowest layer of each branch. This guarantees, that the quality of data representation fulfills a minimum criterion for all parts of the input space, with the *GHSOM* automatically providing the required number of units in the respective areas.

At the transition from one layer to the next, the number of input vectors used for training a particular map decreases to the subset of vectors mapped onto the respective upper-layer unit. Additionally, the input vectors may be shortened on the transition from one layer to the next. This is due to the fact, that some input vector components, i.e. features of the data set, may be expected to be (almost) identical for all input vectors mapped onto a specific unit. These features may be omitted for training the respective lower layer maps, resulting in an increasingly smaller vector dimensionality and thus reduced training time.

6. DESCRIBING CLUSTERS USING THE LABELSOM TECHNIQUE

With no a priori knowledge on the data, even obtaining information on the cluster boundaries as it is provided by a number of cluster boundary analysis methods for the *SOM* [12, 27], does not reveal information on the relevance of single attributes for the clustering and classification process. In the *LabelSOM* approach we determine those vector elements (i.e. features of the input space) that are most relevant for the mapping of an input vector onto a specific unit. This is basically done by determining the contribution of every element in the vector towards the overall Euclidean distance between an input vector and the winners' weight vector, which forms the basis of the *SOM* training process.

The *LabelSOM* method is built upon the observation, that, after SOM training, the weight vector elements resemble as far as possible the corresponding input vector elements of all input signals that are mapped onto this particular unit as well as to some extent those of the input signals mapped onto neighboring units. Vector elements having about the same value within the set of input vectors mapped onto a certain unit describe the unit in so far as they denominate a common feature of all data signals of this unit. If a majority of input signals mapped onto a particular unit exhibit a highly similar input vector value for a particular feature, the corresponding weight vector value will be highly similar as well. We can thus select those weight vector elements, which show, by and large, the same vector element value for all input signals mapped onto a particular unit to serve as a descriptor for that very unit. This is done by calculating the so-called *quantization error vector*. It is computed for every unit i as the accumulated distance between the weight vector elements of all input signals mapped onto unit i and the unit's weight vector elements. More formally, this is done as follows: Let C_i be the set of input patterns $x_j = (\xi_{i_1}, ..., \xi_{i_n}) \in \Re^n$ mapped onto unit i. Summing up the distances for each vector element k over all the vectors x_j ($x_j \in C_i$) and the corresponding weight vector $m_i = (\mu_{i_1}, ..., \mu_{i_n})$ yields a quantization error vector q_i for every unit i (Equation 10).

$$q_{i_k} = \frac{1}{|C_i|}\sqrt{\sum_{x_j \in C_i} (\mu_{i_k} - \xi_{j_k})^2}, \qquad k = 1..n \tag{10}$$

The quantization error for all individual features serves as a guide for their relevance as a class label. Selecting those weight vector elements that exhibit a corresponding quantization error of close to 0 thus results in a list of attributes that are shared by all input signals on the respective unit and thus describe the characteristics of the data on that unit. These attributes thus serve as candidate labels for regions of the map for data mining applications. Based on the ranking provided by the quantization error vector, we can decide to select either a set of labels exhibiting a quantization error vector value below a certain threshold λ_1 as labels or simply choose a set of up to n labels for every unit.

While this selection of labels may be used for standard data mining applications, in the text mining arena we are usually faced with a further restriction. Due to the high dimensionality of the vector space and the characteristics of the $tf \times idf$ representation of the document feature vectors, we usually find a high number of input vector elements that have a value of 0, i.e. there is a large number of terms that is not present in a group of documents. These terms obviously yield a quantization error value of 0 and would thus be chosen as labels for the units. Doing that would result in labeling the units with attributes that are *not* present in the data on the respective unit. While this may be useful for some data analysis tasks, where even the absence of an attribute is a distinctive characteristic, it is definitely not the goal in text mining applications where we want to describe the present features that are responsible for a certain clustering rather than describe a cluster via the features that are *not* present in its data. Hence, we need to determine those vector elements from each weight vector which, on the one hand, exhibit about the same value for all input signals mapped onto that specific unit as well as, on the other hand, have a high overall weight vector value indicating their importance. To achieve this we define a threshold λ_2 in order to select only those attributes that, apart from having a very low quantization error, exhibit a corresponding weight vector value above λ_2. In these terms, λ_2 can be thought of indicating the minimum importance of an attribute with respect to the $tf \times idf$ representation to be selected as a label.

7. EXPERIMENTS

7.1 Text representation

The test environment for our approach comprises three text collections of the most important documents of European case law in English, German and French in HTML format. For reasons of easier evaluation, most tests have been done with the German text collection of 399 documents. Some documents were included twice in different HTML formats. The documents have been taken from the databases CELEX and EUR-Lex with the permission of the Office for Publications of the European Communities in Luxembourg.

In former research, we have tested the feasibility of classification and labeling for legal text corpora using *SOM* and *LabelSOM* [23, 24]. Extensive experiments with three text corpora of about 580 documents in three languages have shown that some improvements of the binary or weighted vector representation are required due to the diversity of legal documents [24].

In this test circle, we thus have focused on the segmentation of documents, splitting documents into independent section, each of which was further split if it exceeded a certain size. Splitting the documents resulted in the following sections: 1. name of the court decision, 2. summary, 3. parties, 4. issue, 5. grounds, 6. costs and 7. disposition (operative part). Unfortunately, we did not succeed in segmenting the grounds according to the numbering of paragraphs due to insufficient and inconsistent mark-up of paragraphs in the HTML sources. Next, all structural parts were segmented in parts of 2000 bytes. Parts smaller than 1000 bytes were discarded with the exception of the disposition. We thus ended up with 4992 document segments. Documents are named by using the CELEX number with a suffix indicating the structural part (no. 1 to 7) and two letters for the various parts within the segment (aa to zz, e.g. 4aa, 4ab, 4ac, ... 4zz). We represent each document by a vector with a length of 8.890 words without prior application of any stemming techniques. We automatically removed words appearing in less than 0.11% or in more than 12% of all documents. Terms are further weighted using a *tf* × *idf* weighting scheme [21].

Our experiments have shown a fine classification and a usable but sometimes insufficient labeling. We have compared the labeling with a manually produced list of all relevant words describing the cluster. We are well aware that this method has to be evaluated with a much larger document collection before commercial exploitation. For the existing challenge of an improvement of the vector representation of legal content a qualitative evaluation seems to be more appropriate. As the standard IR evaluation method of recall and precision is unsuitable for exploratory data analysis, we have chosen the so-called Delphi method [11] with two lawyers and a review of two computer scientists to assess the quality of labeling. The given examples of the maps and the labeling describe this qualitative evaluation and focus on existing potential and future challenges.

7.2 Topical Organization using SOMs

Two different forms of *Self-Organizing Maps* were used: a flat *Self-Organizing Map* with a 40 x 10 map and a *Growing Hierarchical Self-Organizing Map* which evolved into a structure of up to three layers in depth, starting with a map that grew to a size of 6 × 6 units in the first layer. Due to space restrictions, we can present only parts of the full maps. We will give selected examples of clusters and of the corresponding labels. Although the original tests were done in German, we have translated the descriptors into English for reasons of convenience.

7.2.1 Flat SOM

Due to experiences gained from our previous experiments using the *Self-Organizing Map*, the flat *SOM* was seen as the testbed for the feasibility of the segmentation of the documents. Classification indeed produced better results but the labeling was in some cases quite broad. Many units are described only by one or two words. However, the words were a quite good description of the content showing clearly the advantage of the segmentation solution. The various parts of the documents are usually located on the same or neighboring units. Orientation in the flat SOM was quite difficult due to the size of the map.

The map (see Figure 3) shows very interesting concentrations of similar document segments. In the upper left corner, the units contain the document segments for costs and dispositions. The lower left corner contains the headings of the court decisions. The unit (1,10)[1] in the upper right corner represents a fine cluster on bananas from third countries describing 39 parts of the cases 695J0068 (Port IV), 693J0280 (Bananas), 693J0465 (Atlanta III) and 695J0122 (Germany/Council). In close neighborhood, similar units are found. The descriptions for the case 693J0415 (Bosman) were quite good (units (14,6), (14,7)). The chosen labels are *player, association*, and *player, association, urbsfa, bosman, uefa*, respectively. Another good example are the cases concerning the annulment of legal acts with special reference to the airport of Gibraltar (32,4). The description that was given for these parts of the cases 686J0097 (Asteris), 689j0298 (Gibraltar), 689j0309 (Codoníu) are *airport, gibraltar.* Quite illustrative is also the description of unit (40,10) concerning the case 697J0140 (state liability - Rechberger) with labels *organizer, bankruptcy, insolvency, traveling agent, organizing activity.* Still, the word *state liability* is missing in the list of automatically selected labels, but the facts of the case concerning the insolvency of a traveling agent are properly described.

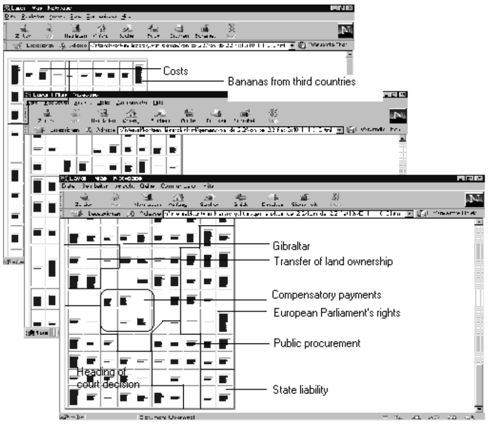

Costs
Bananas from third countries
Gibraltar
Transfer of land ownership
Compensatory payments
European Parliament's rights
Public procurement
Heading of court decision
State liability

Figure 3: Flat SOM: The upper, middle and lower parts of the map.

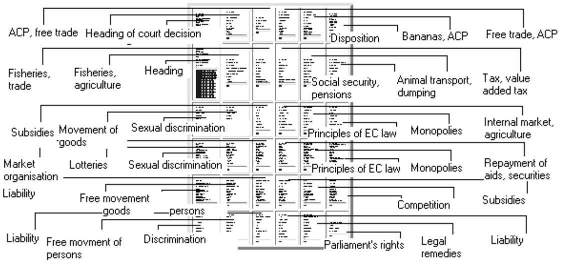


Figure 4: GHSOM: The top-layer map of the hierarchy.

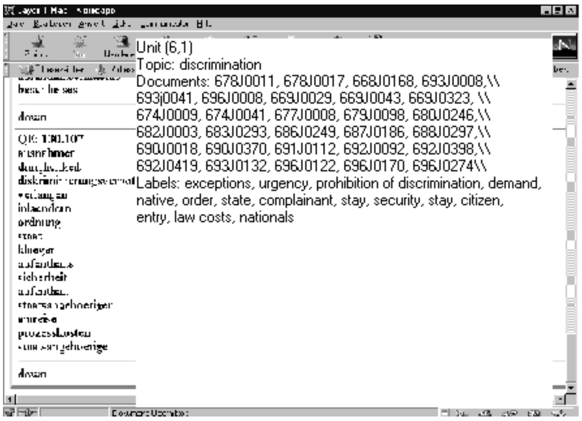
Unit (33,1) concerns restrictions on the transfer of land ownership. The parts of the documents 697J0302 (TGVG), 686J0031 (LAISA) and 696J0122 (Saldanha) were described with *Austria, accession treaty, tgvg* (abbreviation for Land Property Transfer Act of Tyrol). The cases *LAISA* and *Soldanha* were also classified in this unit because they are also related to special exception in the accession treaty.

7.2.2 GHSOM

The *Growing Hierarchical SOM* can deal quite effectively with the very useful technique of generality and specialty most wanted in legal classification. In general, classification and labeling of the *Growing Hierarchical SOM* showed even further improvements compared to former research circles.

The top-layer map of the resulting hierarchy depicted in Figure 4 has grown to a size of 6 × 6 units and gives an overview of the document collection. For example, we find in the upper left corner all case headings, being distinctly

[1] We use the notation (x, y) to refer to the unit in row x and column y, starting with $(1, 1)$ in the upper left corner.



Unit (6,1)
Topic: discrimination
Documents: 678J0011, 678J0017, 668J0168, 693J0008,\\
693j0041, 696J0008, 669J0029, 669J0043, 669J0323, \\
674J0009, 674J0041, 677J0008, 679J0098, 680J0246,\\
682J0003, 683J0293, 686J0249, 687J0186, 688J0297,\\
690J0018, 690J0370, 691J0112, 692J0092, 692J0398,\\
692J0419, 693J0132, 696J0122, 696J0170, 696J0274\\
Labels: exceptions, urgency, prohibition of discrimination, demand, native, order, state, complainant, stay, security, stay, citizen, entry, law costs, nationals

Figure 5: Second Layer Map

different in terms of content from the other sections of the documents. The same applies to the rulings, forming a separate branch of the *GHSOM* structure originating in the upper right corner of the map. The branches representing the various topical areas are organized according to their similarity on the top-level map, with, for example, the sexual discrimination cases being located in the upper left area of the map, or competition cases, located in the lower right section. A branch on discrimination-related cases is located in the lower left corner of the map. Figure 5 gives an overview on the document but also the description.

The documents mapped onto this unit are represented at a somewhat more detailed level in the subsequent layer *SOM*, which grew to a size of 8 × 4 units. The various topical sections on this map, as identified by their labels, are provided in Table 1. Again, we find a topological organization of the various facets of discrimination-related cases, with e.g. discrimination concerning the costs of judicial procedure being located in the upper left area. Down the left side of the map we find, on neighboring units, the rights to residence, next to the freedom of persons, down to exceptions to the regulations mentioned above, located in the lower left corner. Further interesting topical clusters are related to discrimination concerning study fees, income tax and language-issues.

Further topical branches are clustered by content quite similarly. We will pick a few examples from these to take a closer look at the quality of the automatically selected labels for content description. In the following examples, the first unit coordinates denote the according top-layer unit, whereas the second coordinate pair refers to the unit on the second layer map for which a manually assigned topic description, the mapped articles, as well as the automatically assigned labels are provided.

Example 1 - Unit (3,1) (1,3)
Topic: discrimination in the ground of sexual orientation
Document: 696J0249 (Grant)
Labels: child, pregnancy, since, years, tribunal, allowed, two, employer, women, travel concessions, clause, employees, industrial, discrimination, mother

Example 2 - Unit (4,3) (1,4)
Topic: Use of evidence of breath-analysis apparatus that was not properly standardized
Document: 697J0226 (Lemmens)
Labels: imposed, obligation, notify, driving, evidence, effect, accused, breach, charged, approved, to be relied upon, technical, notified, driving while under the influence of alcohol, obtained

costs of judicial procedures, seizure	security for the costs of judicial procedures	security for the costs of judicial procedures	security for the costs of judicial procedures
criminal procedure, seizure, security for the costs of judicial procedures	copyright, language, security for the costs of judicial procedures	security for the costs of judicial procedures	security for the costs of judicial procedures
study fees	study fees	jobs in public administration, licenses	income tax
right to residence, right to remain	right to residence, right to remain	jobs, licenses, reciprocity	freedom of persons, nationals
right to residence, right to remain	right to residence, right to remain	freedom of persons, airport transit	passive freedom of services, language
freedom of persons for spouses	freedom of persons, nationals	freedom of persons, third countries (Morocco)	language (South Tyrol)
freedom of persons	freedom of persons, nationals	free	freedom of persons, nationals, UN-Covenant II
free movement of workers, public security exception, right to residence	free movement of workers, public security exception, directive 64/221	directive 64/221, legal remedies	directive 64/221, legal remedies

Table 1: Layer 2 Map for Unit (6/1) on Layer 1

Example 3 - Unit (2,2) (4,7)
Topic: No transposition of an EC regulation
Document: document: 677j0094 (Fratelli Zerbone)
Labels: import, applicable, day, Italian, each, export, schedule, monetary compensatory amounts, reference, business action, conjunctural policy

Example 4 - Unit (1,4) (5,5)
Topic: Import of bananas from ACP countries
Documents: 687J0247 (Star Fruit), 693J0280 (Bananas), 693J0469 (Chiquita)
Labels: bananas, ACP

Example 5 - Unit (5,1) (1,3)
Topic: Free movement of workers
Documents: 686J0222 (Heylens), 665J0061 (Vaassen-Goebbels), 673J0152 (Sotgiu), 683J0180 (Moser), 691J0237 (Kus)
Labels: security, health, free movement, access, workers, Moser (name of complainant), diploma

In general, most facets of the cases were given. Quite evident is the lack on a meta descriptor covering all subjects. Example 1 is illustrative. The meta descriptor - *discrimination in the ground of sexual orientation* is missing but the context is quite clearly described. Examples 2 and 3 show a quite good picture of the disputed issue. Example 4 shows a short but precise description of the issue of the preference of ACP bananas. Example 5 gives an overview of the importance of the public security exception clause for the free movement of workers.

A very particularity of legal texts makes labeling of legal documents a very difficult task. Important words or phrases are used only once after long arguing.

Example 6 - Unit (2,4) (2,8)
Topic: no review of legality of legal acts for privileged complainants after two months
Document: 691J0074
Labels: appeal, tax allowance, directives, legal relations, May, April, planes, tax allowances, effects, instead of, value added tax, value added tax system, cumulate, turnovers
The main topic of review of legality of legal acts is not described properly by the labels. Only the descriptors concerning the VAT give some hint about the issue of the case. This is due to the fact, that the crucial keywords are only mentioned very few times in the according documents, which is entirely different from the situation in conventional texts, where the most important topic is frequently stressed. Furthermore, to some extent, the effects of strong neighboring clusters have negative effects on the quality of labels for weaker topical clusters. While the classification remains correct the strong units in the neighborhood with many documents change the vector on a unit by the learning procedure.

Example 7 - Unit (2,2) (3,4)
Topic: applicability of rules for public works contracts for telecommunication enterprises (British Telecom)
Document: 693J0392
Labels: register, wine year, seizure, coercive measure, fishing boats, internal, regulation, October, Danish, proprietor, sea waters, flag, Germany, table wine

The selected labels obviously do not describe the topic of this unit appropriately. Rather, these labels properly describe the neighboring unit (2,2) (3,5) with the document 688J0217 and the unit (2,2) (3,3) with the document 690J0286. The units above (2,2) (2,4) and below (2,2) (4,4) deal also with fishing and high sea. These strong neighboring clusters overrule the labels for the weaker cluster on unit (3,4). In Example 8 the procedural context has been predominant in labeling.

Example 8 - Unit (2,6) (2,3)
Topic: land ownership transfer act of Tyrol
Document: 697J0302 (in two HTML versions)
Descriptions: additional payment, payment in kind, health insurance, dependent, systems, payment, scope of validity, contribute, legal rules, compensation, social security insurance, national law, applicant, security, social

7.3 Meta description of a document

In addition to the maps, the user can be provided with a meta description of the document containing the meta tags of all units in which a part of the document can be found. As an example, we have chosen the cases 696J0274 *(Bickel/Franz)* and 681J0267 *(Petrolifera)* concerning the right of foreigners to use the minority language in courts and applicability of GATT in EC law, respectively.

Case: 696J0274 (Bickel/Franz)
Units with meta tags:
(6,1) discrimination
(6,1) (2,2) copyright, language, security for the costs of judicial procedures
(6,1) (5,4) freedom to receive services, language
(6,1) (6,4) language (South Tyrol)
(1,1) (12,2) issue part of the court decision

Case 681j0267 (Petrolifera)
Topic: applicability of GATT in EC law
Units with meta descriptors:
(2,3) Binding force of treaties in EC law, fisheries, maritime resources, transit goods, trade/GATT, customs tariff
(2,3) (10,1) Transit of goods, GATT, transit agreement
(2,3) (11,1) GATT and transit
(2,3) (12,1) GATT applicability in EC law
(2,3) (12,2) GATT and EC customs tariff
(2,3) (12,3) direct applicability of GATT law

As the examples have shown, the meta description of the documents works quite well and gives a comprehensive overview. It should be noted that the labels are also quite helpful as additional information.

8. CONCLUSIONS AND FUTURE WORK

In summarizing the results, we can establish some important steps in our research. The segmentation has opened the door to a more sophisticated analysis of legal segments. There exists a strong difference between ordinary language and legal language in case of statistical analysis. Quite often, the most important words are used only once following long and diverse argumentation. The *Self-Organizing Map* is quite appropriate dealing with this problem because it takes into account the co-occurrences in a very high-dimensional feature space. Lawyers are highly trained text analyzers and expect a higher degree of quality. Therefore, the presented tool may be very helpful for a legal researcher but further improvements of the labeling quality are necessary.

As the segmentation has been quite successful for improved indexing, using available XML structure will also provide more quality. Especially helpful would be the numbering of the paragraphs of court decisions and the paragraphs or articles of statutes. Besides some reluctance, we will change from traditional binary or $tf \times idf$ vector representation to a modified vector representation taking into account also descriptors of legal thesauri but also more important parts of the documents. In practice, we will change the $tf \times idf$ value in order to giver higher value to an important legal term or words occurring very often in the reasoning of courts.

A major advantage may be the screening of huge document collections for particular problems showing connections between documents that are not detected by manual research. The main problem remains the development of the knowledge base of lexica and rules. A long step would be the use of fully-developed ontologies, cf. [28]. The developed formalization offers a high potential of improvement but the manual development of ontologies requires high resources. Therefore, we will focus on more simple ontologies like in the FLEXICON or KONTERM project.

Acknowledgements

This research was supported by the Jubiliäumsfonds der Österreichischen Nationalbank, Vienna, research project no. 6888. We have to thank the Office for Publications of the European Communities for providing us with a document collection. Finally, the very valuable evaluation work of G. Prutsch, A. Simek, B. Rendl, I. Pellech and J. Grabner should be acknowledged.

9. REFERENCES

[1] J. Blackmore and R. Miikkulainen. Incremental grid growing: Encoding high-dimensional structure into a two-dimensional feature map. In *Proc IEEE Int'l Conference on Neural Networks (ICNN'93)*, volume 1, pages 450–455, San Francisco, CA, USA, 1993.

[2] S. Brünninghaus and K. Ashley. Toward adding knowledge to learning algorithms for indexing legal cases. In *Proc Int'l Conf Artificial Intelligence and Law (ICAIL'99*, pages 9–17, 1999.

[3] J. Daniels and E. Rissland. Finding legally relevant passages in case opinions. *Proc Int'l Conf Artificial Intelligence and Law (ICAIL'95)*, pages 39–46, 1995.

[4] M. Dittenbach, D. Merkl, and A. Rauber. The growing hierarchical self-organizing map. In *Proc Int'l Joint Conference on Neural Networks (IJCNN 2000)*, volume VI, pages 15 – 19, Como, Italy, July 24. – 27. 2000. IEEE Computer Society.

[5] B. Fritzke. Growing Grid – A self-organizing network with constant neighborhood range and adaption strength. *Neural Processing Letters*, 2(5):1 – 5, 1995.

[6] U. Hahn and I. Mani. The challenges of automatic summarization. *IEEE Computer*, 33(11):29–35, November 2000.

[7] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43, 1982.

[8] T. Kohonen. *Self-organizing maps.* Springer-Verlag, Berlin, 1995.

[9] T. Kohonen, S. Kaski, K. Lagus, J. Salojärvi, J. Honkela, V. Paatero, and A. Saarela. Self-organization of a massive document collection. *IEEE Transactions on Neural Networks*, 11(3):574–585, May 2000.

[10] V. Konstantinou, J. Sykes, and G. Yannopoulos. Can legal knowledge be derived from legal texts? In *Proc Int'l Conf Artificial Intelligence and Law (ICAIL'93)*, pages 218–227, 1993.

[11] H. Linstone and M. Turoff. (ed.) The Delphi Method: Techniques and Applications. Addison-Wesley, Massachusetts, 1975.

[12] D. Merkl and A. Rauber. Alternative ways for cluster visualization in self-organizing maps. In T. Kohonen, editor, *Proceedings of the Workshop on Self-Organizing Maps (WSOM97)*, pages 106–111, Espoo, Finland, 1997. Helsinki University of Technology.

[13] D. Merkl and A. Rauber. Document classification with unsupervised neural networks. In F. Crestani and G. Pasi, editors, *Soft Computing in Information Retrieval*, pages 102–121. Physica Verlag, 2000.

[14] D. Merkl and E. Schweighofer. The exploration of legal text corpora with hierarchical neural networks: A guided tour in public international law. In *Proc Int'l Conf Artificial Intelligence and Law (ICAIL'97)*, pages 98–105, Melbourne, Australia, 1997. ACM.

[15] R. Miikkulainen. Script recognition with hierarchical feature maps. *Connection Science*, 2:83 – 101, 1990.

[16] M. Moens. *Automatic Indexing and Abstracting of Document Texts.* Kluwer Academic Publishers: Boston, 1999.

[17] A. Rauber. LabelSOM: On the labeling of self-organizing maps. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN'99)*, Washington, DC, July 10. - 16. 1999.

[18] A. Rauber and D. Merkl. The SOMLib Digital Library System. In S. Abiteboul and A. Vercoustre, editors, *Proceedings of the 3. European Conference on Research and Advanced Technology for Digital Libraries (ECDL99)*, LNCS 1696, pages 323–342, Paris, France, Springer.

[19] A. Rauber, E. Schweighofer, and D. Merkl. Text classification and labelling of document clusters with self-organising maps. *Journal of the Austrian Society for Artificial Intelligence (ÖGAI)*, 19(3):17–23, October 2000.

[20] B. Ripley. *Pattern Recognition and Neural Networks.* Cambridge University Press, Cambridge, UK, 1996.

[21] G. Salton. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer.* Addison-Wesley, Reading, MA, 1989.

[22] E. Schweighofer. *Legal Knowledge Representation.* Number 7 in Law and Electronic Commerce. Kluwer Law International: The Hague, 1999.

[23] E. Schweighofer and D. Merkl. A learning technique for legal document analysis. In *Proc Int'l Conf Artificial Intelligence and Law (ICAIL'99)*, pages 156–163, Oslo, Norway, 1999. ACM.

[24] E. Schweighofer, A. Rauber, and D. Merkl. Some remarks on vector representation of legal documents. In A. Tjoa, R. Wagner, and A. Al-Zobaidie, editors, *DEXA Workshop Proceedings of the Workshop on Legal Information Systems (LISA 2000)*, pages 1087 – 1091, Greenwich, UK, IEEE Computer Society Press.

[25] S. Simits. *Informationskriese des Rechts und Datenverarbeitung (Information Crisis of Law).* Müller, Karlsruhe, Germany, 1970.

[26] J. Smith, D. Maccrimmon, B. Atherton, J. McClean, J. Shinehoft, and L. Quintana. Artificial intelligence and legal discourse: The flexlaw legal text management system. *AI & Law*, 3(1):55–95, 1995.

[27] A. Ultsch. Self-organizing neural networks for visualization and classification. In O. Opitz, B. Lausen, and R. Klar, editors, *Information and Classification. Concepts, Methods and Application*, Studies in Classification, Data Analysis, and Knowledge Organization, pages 307–313. Springer, Dortmund, Germany, 1992.

[28] A. Valente. *A Modelling Approach to Legal Knowledge Engineering.* IOS Press: Amsterdam, 1995.

[29] P. Visser and R. Winkels, editors. *First International Workshop on Legal Ontologies.* 1997.

Automatic Text Representation, Classification and Labeling in European Law

Erich Schweighofer (1), Andreas Rauber (2), Michael Dittenbach (2)
(1) Institute of Public International Law, University of Vienna Research Center for Computers and Law, Universitätsstr. 2, A-1090 Vienna, Austria, Erich.Schweighofer@univie.ac.at
(2) Institute for Software Technology, Vienna University of Technology, Favoritenstr. 9-11 / 188, A-1040 Vienna, Austria, {andi,mbach}@ifs.tuwien.ac.at.

In: Proceedings of the International Conference on AI and Law 2001, May 21-25, 2001, St. Louis, MI