Enhancing Document Modeling by Means of Open Topic Models

Crossing the Frontier of Classification Schemes in Digital Libraries by Example of the DDC

Alexander Mehler and Ulli Waltinger
Faculty of Technology, Bielefeld University, Bielefeld, Germany

Abstract

Purpose:
We present a topic classification model using the Dewey Decimal Classification (DDC) as the target scheme. This is done by exploring metadata as provided by the Open Archives Initiative (OAI) to derive document snippets as minimal document representations. The reason is to reduce the effort of document processing in digital libraries. Further, we perform feature selection and extension by means of social ontologies and related web-based lexical resources. This is done to provide reliable topic-related classifications while circumventing the problem of data sparseness. Finally, we evaluate our model by means of two language-specific corpora.
This paper bridges digital libraries on the one hand and computational linguistics on the other. The aim is to make accessible computational linguistic methods to provide thematic classifications in digital libraries based on closed topic models as the DDC.

Design/methodology/approach:
text classification, text-technology, computational linguistics, computational semantics, social semantics.

Findings:
We show that SVM-based classifiers perform best by exploring certain selections of OAI document metadata.

Research limitations/implications:
The findings show that it is necessary to further develop SVM-based DDC-classifiers by using larger training sets possibly for more than two languages in order to get better F-measure values.

Practical implications:
We can show that DDC-classifications come into reach which primarily explore OAI metadata.

Originality/value:
We provide algorithmic and formal-mathematical information how to build DDC-classifiers for digital libraries.

Keywords:
closed topic models, open topic models, document modeling, document snippets, DDC, OAI

Paper type:
Research paper

1 Introduction

It is beyond any doubt that automatic content classification is of outmost interest in digital libraries (Lossau, 2004). The idea is to provide content-related add-ons which allow for improving retrieval and document processing. In this introduction, we give a short overview of competing approaches in this field of research which focus on condensed document representations as provided, for example, by keyword lists or summaries.

An early approach to clustering document summaries at different levels of thematic granularity is the scatter-gather method (Cutting *et al.*, 1992; Hearst and Pedersen, 1996). In recent years, variants of the Suffix Tree Clustering (STC) algorithm (Meyer zu Eißen, 2007; Stein and Meyer zu Eißen, 2003; Zamir and Etzioni, 1999; Stefanowski and Weiss, 2003) also attracted attention in this field of research. These variants explore common sub-phrases of documents which are judged to be similar because of their common suffix trees. An alternative approach with a focus on hierarchical document classification has been introduced by (Zhan and Dong, 2004) who explore search query snippets instead of summaries as the main source of document representation. These and related approaches form the core of search engines as, e.g., *Vivísimo* (Valdes-Perez *et al.*, 2000), *Mapuccino* (Maarek *et al.*, 2000) and *Carrot* (Osinski and Weiss, 2005), which perform post-retrieval document clustering. That is, they detect topic labels of thematic clusters based on document snippets (e.g., titles) as retrieved by search queries (Kules *et al.*, 2006). The idea behind this approach is to enhance the identification of relevant documents by eliminating the need to skim large numbers of irrelevant texts.

This approach is easily transferred to the area of digital libraries where document snippets are given by subject-related metadata. A metadata protocol which recently became more and more prominent is the *Open Archives Initiative-Protocol for Metadata Harvesting* (OAI-PMH). This protocol implements a standardized metadata model for facilitating exchange between repositories. Approaches to document clustering in digital libraries have focused, among other things, on extending search queries and metadata entries of documents (Hagedorn *et al.*, 2007; Rosenberg and Borgman, 1992). In this case, clustering is performed to detect the subject area of documents based on a predefined classification scheme, that is, a closed topic model (Newman *et al.*, 2007).

In this article, we present a topic classification model which uses the *Dewey Decimal Classification* (DDC) (OCLC, 2008) as the target scheme. Our approach is novel in two senses. On the one hand, we use metadata as provided by the *Open Archives Initiative* (OAI) to derive document snippets as minimized document representations. This is done to reduce the time and space complexity of document processing. On the other hand, we perform feature selection and feature extension by means of social ontologies and related web-based lexical resources. This is done to provide reliable topic-related classifications while circumventing the problem of data sparseness. In a nutshell, the article provides a model of topic-related document classifications whose semantics is explored by means of web-based resources of semantic relatedness and whose document model is mainly based on OAI data.

The article is structured as follows: in Section 2, we describe several reference points of document modeling in digital libraries. We do that to shed light on how to cross the frontier of classification schemes, i.e., moving from closed topic models toward open topic models. Next, in Section 3, we describe our test corpora and the representation of documents by means of OAI metadata. In Section 4, we introduce a search engine-based classifier for the DDC which integrates social semantic knowledge to enhance document representation. Further, in Section 5, we present an experiment in DDC classification using two different corpora and five different DDC-related classifiers. This experiment is discussed in detail in Section 6. Finally, Section 7 concludes and suggests prospects for future work.

2 On Reference Points of Document Modeling

When classifying a document by its topic, one has at least two possibilities: either one uses a *closed*, i.e., fixed system of categories (e.g., a classification scheme) or one uses an *open* system that changes in time. As all systems change in the long run, we have to avoid a triviality by stipulating that the time scale for open systems is less then the rate of change of classification schemes. In order to arrive at a working definition, we assume that this scale manifests a random division of time as given by the change rate of social ontologies (Mehler, 2009; Mika and Gangemi, 2004).

From this point of view, there are two reference points of system dynamics of which only the second justifies the attribute *open*:

- Firstly, because a system of topic categories may be given by a category *graph* or, more narrowly, by a *tree* (as, e.g., the DDC), its change may affect only its links leaving its vertex set untouched.

- Secondly, the vertex set of the graph may change in time by the deletion, merging, splitting or insertion of categories. In this case, a change of the set of links may, but does not need to be a consequence of the system dynamics.

Note that if we do not deal with graph-like topic systems, but with category sets (as done by the majority of approaches to text categorization (Sebastiani, 2002)), the latter distinction is irrelevant.

	closed	**open**
topic	*content classification scheme*	*emergent topics model*
genre	*genre palette*	*emergent genres model*

Table 1: Four cases of mapping categories and texts.

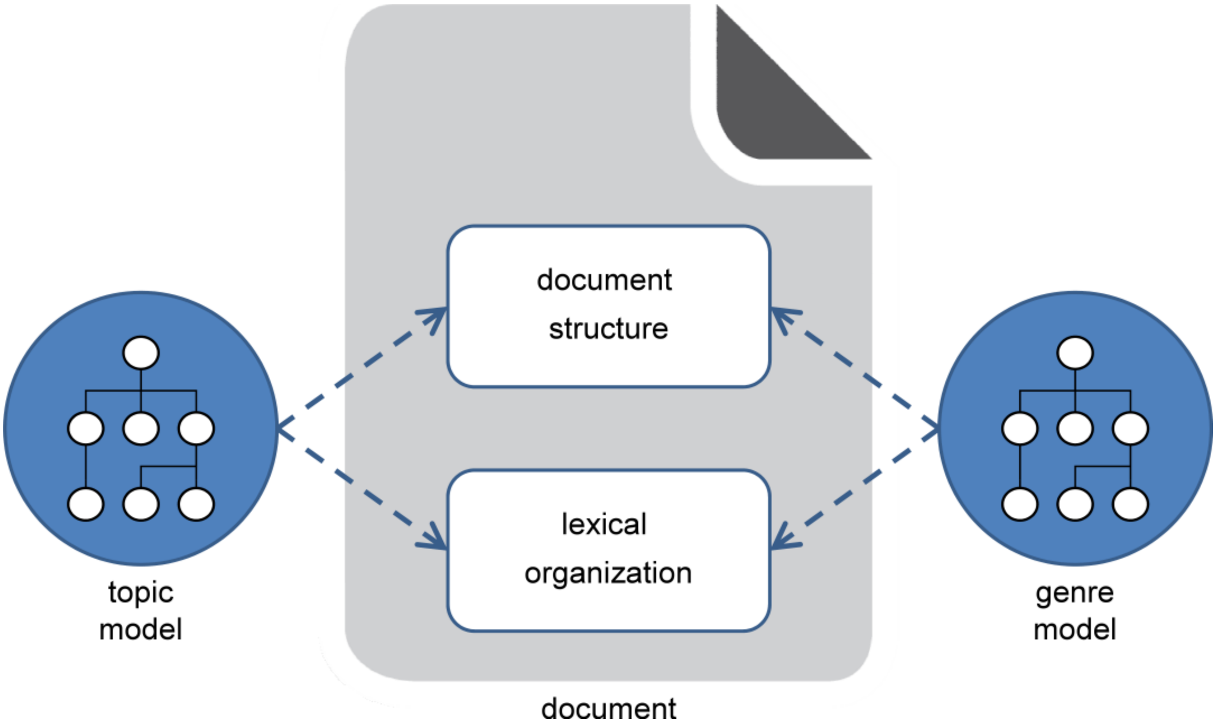


Figure 1. Realizing topic or genre by the lexical micro or the discourse macro structure of a document.

In any event, the distinction of open and closed models is related to a second, more classical difference: that is, documents may be similar by topic or by genre. Take the example of *nanotechnology* dealt with by a journal article in contrast to a doctoral thesis. In this case, the documents agree on topic but differ on genre. Conversely, we can have documents of the same genre, which differ on topic. Thus, we get two, not necessarily orthogonal views of document classification (Biber, 1995; Halliday and Hasan, 1989) and, hence, a decision matrix by which four document models can be distinguished (cf. Table 1):

- *Closed Topic Model* (CTM): As mentioned above, a topic model is closed if its composition is fixed. A CTM is given by a classification scheme (e.g., the DDC or MeSH) as a terminological ontology (Sowa, 2000) whose vertices denote conceptual types of topic areas. Such schemes are generated by a small number of selected experts whose collaboration is controlled according to the prospected target ontology communicated to its users in a one-to-many setting. The low rate of change of CTMs corresponds to a non-random time scale. This is somehow in contradiction to the dynamics and openness of the human topic universe with its ever emergent and growing topics. However, a CTM guarantees repeatability of classification results and comparability over time so that it is still usable in digital libraries. Note that closed topic models are suitable as target models of supervised learning as their fixed nature is a precondition of persistent and reliable training data.
- *Open Topic Model* (OTM): In an OTM, the topic categories are not enumerated in advance (as in supervised learning) or the result of labeling clusters found in a fixed set of empirical data (as in unsupervised learning). Rather, OTMs explore topic labels from an open, that is, ever-growing social ontology. Social ontologies as, e.g., the category system of Wikipedia, are output by social tagging (Mika and Gangemi, 2004). They emerge as a solution to a coordination problem among large groups of interacting agents (Bickhard, 2008). This relates to the sharing of a collectively structured semantic universe in the form of non-formal ontologies (Mehler, 2009). Unlike the one-to-many communication of terminological ontologies, social ontologies result from a many-to-many communication in which groups of agents interact to constitute and organize a dynamically growing universe of content units. Social ontologies provide large-scale and flexible knowledge systems for building OTMs, and these evolve according to the time scale of the topic universes of speech communities. In a nutshell: an OTM obtains its topic model from a social ontology with which it co-evolves. OTMs extend the paradigm of supervised and unsupervised learning by integrating human computation. In line with this model are web-based resources of lexical relatedness as explored by *collocation networks* (Heyer *et al.*,

2006). By analogy to OTMs, collocation networks grow due to the dynamics of human computation in the web and also dispense with predefining any semantics. This holds all the more for measures of semantic relatedness based on search engines which directly access the web as a, so to speak, universal information base. In this article, we explore these three different resources of OTMs in a single framework.

- *Closed Genre Model* (CGM): topic models are predominant in IR. However, there is rising interest in alternative retrieval models, e.g., by taking genre into account (Dillon, 2008; Mehler *et al.*, 2009; Rosso, 2008). This is mostly done by CGMs whose categories are enumerated in advance (Stein *et al.*, 2009). Although a standardization of genre categories by analogy to the stringency of CTMs is far away, the web mining community establishes such systems by so-called *genre palettes* (Santini, 2009) to guarantee comparability of classification results (Rehm *et al.*, 2008).
- *Open Genre Model* (OGM): From that perspective, one may think of genre palettes which co-evolve with some social tagging. At first glance, this seems to be a futile endeavor as textual genres change much more slowly than topics. However, if we think, e.g., of games with the purpose of annotating multimedia objects (von Ahn and Dabbish, 2008), we enter the required dynamics: if we apply this model to the area of emergent web genres, we arrive at a scenario by analogy to OTMs (see www.websitewiki.de for an example of social software used to describe websites). In this sense, OGMs emerge as a way forward that addresses the deficient coverage rate of genre palettes with respect to the dynamics of web-based communication.

So far we have related topic and genre to either open or closed category systems. To complement this picture, we distinguish, by analogy to (Halliday and Hasan, 1989), two levels of realization of topic and genre: the lexical *micro structure* of a document and its (e.g., rhetorical, functional or logical) *macro structure* (cf. Figure 1). The vast majority of approaches to IR explore easily accessible lexical structures. Only a minority utilize document macro structures (cf. Denoyer and Gallinari, 2003; Mehler *et al.*, 2007).

In this article we introduce a document representation model which combines a closed with an open topic model. This is done by exploring the lexical structure of a document subject to a highly restricted representation of its macro structure. As a closed topic model, we utilize the DDC in combination with, among other things, Wikipedia as the operative social ontology. More specifically, we explore the OAI metadata of a document as a highly condensed document representation where Wikipedia and web-based lexical resources are used to circumvent the problem of data sparseness and to secure the usage of topic-related, that is, semantic document features. In a nutshell, this article crosses the border of closed topic models into the direction of open topic models to profit from both the openness and covering rate of the latter and the systematicity of the former. This is what we subsume under the notion of *social semantics for digital libraries.*

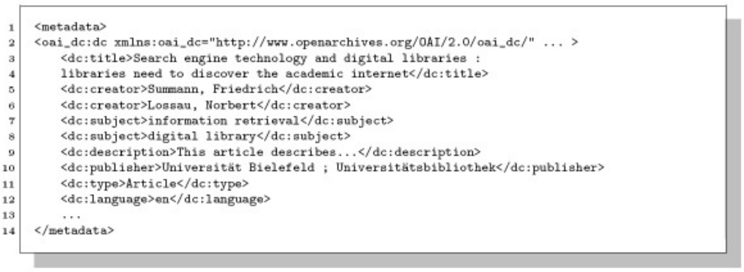
No	Label
000	Computer science, information & general works
100	Philosophy & psychology
200	Religion
300	Social sciences
400	Language
500	Science
600	Technology
700	Arts & recreation
800	Literature
900	History & geography

No	Label
000	Computer science, knowledge & systems
010	Bibliographies
020	Library & information sciences
030	Encyclopedias & books of facts
040	[Unassigned]
050	Magazines, journals & serials
060	Associations, organizations & museums
070	News media, journalism & publishing
080	Quotations
090	manuscripts & rare books

Table 2. The 10 top categories of the DDC (top) and the 10 subdivisions of the class 000 (bottom) (OCLC 2008).

3 Building a Test Corpus for Closed Topic Models

In this section, we describe the CTM used for document classification, i.e., the *Dewey Decimal Classification* (DDC) (cf. Section 3.1). Further, we describe the preprocessing of input documents by their OAI metadata (cf. Sec. 3.2 and 3.3).



```
<metadata>
<oai_dc:dc xmlns:oai_dc="http://www.openarchives.org/OAI/2.0/oai_dc/" ... >
    <dc:title>Search engine technology and digital libraries :
    libraries need to discover the academic internet</dc:title>
    <dc:creator>Summann, Friedrich</dc:creator>
    <dc:creator>Lossau, Norbert</dc:creator>
    <dc:subject>information retrieval</dc:subject>
    <dc:subject>digital library</dc:subject>
    <dc:description>This article describes...</dc:description>
    <dc:publisher>Universität Bielefeld ; Universitätsbibliothek</dc:publisher>
    <dc:type>Article</dc:type>
    <dc:language>en</dc:language>
    ...
</metadata>
```

Figure 2. Outline of the OAI metadata of OCLC, 2008. Dots indicate omitted content.

3.1 The DDC as a Closed Topic Model

As a CTM we use the DDC (OCLC, 2008) which is the most common classification scheme for subject cataloguing in libraries. The DDC starts from 10 main classes which are subdivided on two levels such that each subdivision is structured into 10 areas (cf. Table 2). As a result, a forest of 10 trees is spanned each of which has 111 vertices — a remarkably artificial ontology as is typical for CTMs. We access the DDC by means of the *Bielefeld Academic Search Engine* (BASE) (Pieper and Summann, 2006) which provides OAI metadata of input documents. Among other things, this includes their DDC-related classifications. BASE has mapped documents on all levels of the DDC, with up to 100,000 training candidates processed so far. These classifications allow us to build a training corpus for the top-level categories of the DDC, and these, in turn, are used finally to evaluate our approach.

3.2 Minimizing Representation Effort by Exploring OAI Snippets

A central demand of digital libraries concerns the fast, online and reliable classification of documents. In order to guarantee reliability, the documents need to be processed carefully. This requirement is opposed by the space complexity of processing documents up to the length of theses. On the other hand, if one reduces the data to be processed, the problem of data sparseness is raised which may produce misclassifications. In order to balance the prevention of the latter risk against the former requirement, we do not process input documents directly, but explore their *Open Archives Initiative* (OAI) metadata (cf. Table 3 for a sample document representation). More specifically, we use the *OAI-Protocol for Metadata Harvesting* (OAI-PMH) to access document snippets in the form of OAI metadata fields. This allows us to explore document data without the need to parse the entire document. Generally speaking, OAI represents a document by a title, creator, subject, description, publisher, type, format, and language tag (cf. Figure 2).

Info	**English**	**German**
Summary: arithmetic mean	84.45	15.43
Summary: standard deviation	9.36	10.81
Keywords: arithmetic mean	7.74	12.75
Keywords: standard deviation	7.03	10.51

Table 3. Corpus statistics: standard deviation and arithmetic mean of comprised OAI-Snippet summary and keyword information by single tokens.

These tags are mainly based on the Dublin Core metadata element set extended by a small set of OAI-specific tags. For the task of document representation, we further reduce this set so that each document is finally represented by three (types of) tags: that is, `title`, `subject`, and `description`. From that perspective, our classification hypothesis reads as follows:

> **H1** *The topic of a scientific document is reliably classified by processing its title, subject fields and short description.*

In relation to the DDC as the operative CTM we get the following target statement to be evaluated experimentally: *you shall know the top-level DDC class of a scientific document by its OAI snippets*. Note that the subject field of the English documents included into our experimental corpus contains 84.45 tokens on average, while its description field contains on average 9.36 tokens (cf. Table 3) — *this is a remarkably small set of tokens in relation to documents of the length of articles or even books*. Note also that apart from the title, none of the OAI fields taken into consideration is necessarily extracted from the underlying document. Rather, these fields may depart in their lexical structure from the lexical structure of the document itself. Our minimization procedure reduces the space complexity of document representation and saves processing time. However, it also raises the risk of data sparseness. To face this risk, that is, to secure semantic reliability of the features which are finally explored to classify documents, we cross the border to open topic models and utilize a social ontology in conjunction with web-based resources of lexical relatedness. This is described *in extenso* in Section 4. First, however, we describe the preprocessing of the document snippets.

3.3 Preprocessing OAI Snippets

The preprocessing of OAI document snippets is performed in the usual way. That is, we perform language identification, segmentation of the logical document structure (including sentence boundary detection), lemmatization of lexical units, part of speech tagging and named entity recognition (Mehler *et al.*, 2008; Waltinger and Mehler, 2008). This allows for filtering out non-lexical tokens as well as function words. As a result, we get a linguistically tagged input stream of lexical features per document snippet which is input to the next step: *feature selection*. This step is performed by mapping an input stream of lexical units of the OAI snippet representation x of a document X onto a fuzzy set X where, for any lexical item a in x, the membership value $\mu_{\mathrm{X}}(a)$ is computed as the frequency of a in x standardized by the frequency of the most frequent item in x.

This allows us to build a ranked feature list per input document X, where the higher $\mu_{\mathrm{X}}(a)$, the higher the rank of feature a. From a technological point of view, X is an algebraic representation of a sparse matrix as represented, e.g., in $\mathtt{SVM}^{\mathtt{light}}$ (Joachims, 2002). Thus, we easily derive a vector representation of X, weight it in the usual way (Salton and Buckley, 1988), derive a *Vector Space Model* (VSM) (Salton, 1989) of the input corpus and finally make this an input to *Latent Semantic Analysis* (LSA) (Landauer and Dumais, 1997) which serves as a baseline scenario in our experiment (cf. Section 5).

4 Two Novel DDC Classifiers

As a novel method of classifying OAI-Metadata according to the DDC we now introduce a generalised *Search Engine Quotient* (SEQ). This classifier extends the so-called *Google quotient* (Cilibrasi and Vitanyi, 2007; Cramer, 2008) by combining distance measuring with a category feature model based on co-occurrence statistics. Since we do not focus on measuring the relatedness of pairs of tokens but on classifying documents by means of OAI snippets, we need to explore for features in these snippets (Section 4.1) as significant content descriptors (Section 4.2). Further, we need to define a separate feature model for each of the categories to be classified in order to relate them to our document feature models (Section 4.3). Next, we have to implement a search engine-based quotient — in the present case by means of the Wikipedia and the search engine Yahoo — in order to map OAI snippets to DDC classes (Section 4.4). By this procedure we get a classification value for each main class of the DDC which expresses the relatedness of a given OAI-input stream to the selected class.

4.1 Building Document Models

Generally speaking, we view the OAI data assigned to a document as a highly condensed representation of that document. More specifically, apart from function words we view any lexical constituent a_i of the snippet S_j assigned to a document D_j as a candidate feature of the content of that document. By ranking these lexical constituents according to their standardized *term-frequency* (tf) in descending order we get information about the most significant content terms of D_j.

That is, terms a_i are ranked with respect to documents D_j by computing their frequency index tf_{ij} where f_{ij} is the frequency of a_i in D_j and $L(D_j)$ is the set of all lexical constituents of D_j (note that function words are excluded):

$$tf_{ij} = \frac{f_{ij}}{\max_{a_k \in L(D_j)} f_{kj}} \in (0,1] \text{ (Equation 1)}$$

Additionally, we take multi-word units and phrases into account by ranking them according to their standardized *phrase-frequency* (pf). Note that we explore frequent phrases by means of n-grams of tokens. This allows us to rank phrases p_i with respect to documents D_j by computing their frequency index pf_{ij}

$$pf_{ij} = \frac{f_{ij}}{\max_{p_k \in P(D_j)} f_{kj}} \in (0,1] \text{ (Equation 2)}$$

where f_{ij} is the frequency of p_i in D_j and $P(D_j)$ is the set of all n-grams of D_j. By the rank-frequency lists of words and phrases assigned to a document D_j we can select the topmost ranked features of both lists. This is done by means of two stacks: the *word stack* WS_j and the *phrase stack* PS_j which list all words and phrases of D_j in descending order of their standardized frequencies tf_{ij} and pf_{ij}, respectively. In a nutshell: WS_j (PS_j) is the list of all lexical (phrasal) items of D_j in descending order of their significance as content descriptors of D_j where significance is measured in terms of frequency.

4.2 Feature Verification

By means of the feature stacks WS_j and PS_j assigned to a document D_j we can select the topmost ranked lexical and phrasal features of D_j. This is done in order to secure reliable search results when using document features as search terms of a search engine-based query. The number N of features to be selected in this step has to be carefully chosen. By selecting too few features, search results get unspecific, while too many features can distract the search from the actual content of D_j.

In order to reduce the risk of choosing the wrong number N of selected features, we utilize and refine the approach of (Waltinger *et al.*, 2008). More specifically, we initially set $N = \max(|WS_j|, |PS_j|)$ where $|S|$ is the length of stack S. Then, we perform a search in our reference search engine — in the present case *Wikipedia*. This is initially done by using all N topmost ranked lexical and phrasal features. Next, we decrement N, that is, $N \leftarrow N - 1$, and repeat the latter search till we get at least one Wikipedia article as a search result. That way, we select the $N \leq |WS_j|$ topmost ranked lexical and the $N \leq |PS_j|$ topmost ranked phrasal features *without* the need to preset N. In other words: we align the number N of significant features to the characteristics of the given input document D_j. We denote this threshold by N_j. As a result, each document D_j is represented by its N_j most significant lexical and N_j most significant phrasal features extracted from its OAI metadata representation. The reason to do this is to filter out irrelevant features even if they are frequent. The final feature set is denoted by

$$F(D_j) = \{F \in WS_j \mid \text{rank}(F) \leq N_j\} \cup \{F \in PS_j \mid \text{rank}(F) \leq N_j\} \quad \text{(Equation 3)}$$

which is the feature set representation of document D_j where $\text{rank}(F)$ is the rank of feature F in the corresponding stack as determined by the frequency index of F — we write $F \in WS_j$ to denote that feature F is on stack WS_j.

4.3 Building Topic Models

So far we have shown how to represent documents by subsets of lexical and phrasal constituents. Now, we turn to the task of learning separate feature models for each of the 10 main classes of the DDC. More specifically, we represent each main class of the DDC by means of two resources of feature extraction: (i) the titles of the divisions and sections dominated by the corresponding class and (ii) web-based co-occurrence data related to these titles. As an example, consider the first class of the DDC: 000 Computer Science, Information & General Works. This class dominates 10 divisions on the second level of the DDC (i.e., Bibliographies, Library, Encyclopedia, …) and 100 sections on the third level (i.e., Knowledge, The book, Systems, Data processing, …). Each of these division and section titles is added to the representation model of the class 000. That is, each of the 10 DDC classes is represented by 110 lexical items (including multiword terms). In the second step, we enrich this feature model by extending each feature by its most significant co-occurrence neighbor.

In our experiments we retrieved this neighborhood information by the Web service of the *Leipziger Wortschatz* (http://corpora.informatik.uni-leipzig.de/) (Heyer *et al.*, 2006). In the case of class 000 we enriched, for example, the feature *book* by *published* and the feature *system* by *operating* as these are the most significant lexical neighbors of both features in the latter co-occurrence network. This approach overcomes problems of data sparseness by exploring co-occurrence data as it relies on two related feature resources: *taxonomical information* provided by the DDC and *word association information* provided by the co-occurrence network. Note that we add only one feature per DDC division and section title so that each main class is represented by 220 content descriptors. As a result, we get a feature set $F(C_i)$ of 220 descriptors per DDC main class C_i. The next step is to compute for each class C_i and each document D_j an index of overlap of their representation models $F(C_i)$ and $F(D_j)$. As we deal with linguistic features we do that by accounting for composite units. More specifically, two (lexical or phrasal) features F, G are said to overlap, that is, $F \sim G$, if either $F = G$ or if F is a substring of G. Then, as an index of overlap of $F(C_i)$ with $F(D_j)$ we compute

$$\text{overl}(F(C_i), F(D_j)) = |\{F \mid \exists G \in F(C_i) \exists H \in F(D_j) : F \sim G \wedge F \sim H\}| \quad \text{(Equation 4)}$$

Equation 4 tells us the degree to which features of a given input document D_j occur in the feature representation of class C_i. This is the starting point of performing the final classification as explained in the next section.

4.4 The Classification Rule

The overlap index in Equation 4 relates only those classes and documents whose feature sets actually overlap. As a matter of fact, such an overlap is a strong indicator of class membership but occurs rather infrequently. Therefore, we need a fall-back strategy which covers all cases in which this overlap does not occur or is not large enough to be indicative of class membership. As such, we utilize a search engine-based quotient which is computed as follows: for DDC class C_i and any of its features $F_k \in F(C_i)$ we compute the search engine-based relatedness $\mathrm{rel}(F_k, D_j)$ of F_k and document D_j by

$$\mathrm{rel}(F_k, D_j) = 2 \times \frac{g(F(D_j), F_k)}{g(F(D_j)) + g(F_k)} \text{ (Equation 5)}$$

where $g(F(D_j), F_k)$ is the number of Yahoo hits one gets when using all features in $F(D_j)$ *together with* F_k as search terms while $g(F(D_j))$ and $g(F_k)$ are the corresponding numbers of hits one gets when searching by the features in $F(D_j)$ and by F_k separately. Next, we sum up these values for different features $F_k \in F(C_i)$ to relate C_i and D_j as a whole:

$$\mathrm{Rel}(C_i, D_j) = \sum_{F_k \in F(C_i)} \mathrm{rel}(F_k, D_j) \text{ (Equation 6)}$$

Because it requires too much search effort to sum over all 220 features of class C_i we only consider the 10 division titles assigned to C_i — note that all other features are accounted for by the overlap index in Equation 4. Thus, for the 10 main classes and 10 divisions per class we perform 10 × 10 × 3 = 300 search queries in the course of classifying a given document D_j (note that $g(F(D_j), F_k)$, $g(F(D_j))$ and $g(F_k)$ are computed separately). Next, we compute an overall classification value which takes the index of search-engine-based relatedness of document D_j and class C_i into account as well as their overlap as computed by Equation 4:

$$\mathrm{SEQ}(C_i, D_j) = \alpha \cdot \mathrm{overl}(F(C_i), F(D_j)) + (1 - \alpha) \cdot \mathrm{Rel}(C_i, D_j) \text{ (Equation 7)}$$

This index explores four resources: (i) Wikipedia as a source of document feature selection, (ii) the DDC hierarchy as a source of category feature extraction, (iii) a web-based co-occurrence network for feature enrichment, and (iv) a search engine to provide a fall-back strategy. Note that α allows us to balance these different resources of computing the class membership of a document. However, in order to reduce the parameter set of our study we set $\alpha = .5$. Finally, we classify document D_j by the class C_i for which

$$C_i = \underset{C_k \in C}{arg\,max} \{\mathrm{SEQ}(C_k, D_j)\} \text{ (Equation 8)}$$

where $C = \{C_1, ..., C_{10}\}$ is the set of the 10 main classes of the DDC.

4.5 Utilizing a Wikipedia-based OTM to Build a DDC-Related Classifier

To get a second DDC-related classifier, we explore the Wikipedia as a social-ontological resource of lexical features for modeling documents and topics. In contrast to the SEQ-based classifier, the Wikipedia-based classifier omits the feature verification step.

That is, it uses all lexical features to compute the membership value of a document to a DDC category. In this context, a reduced vector representation of the Wikipedia data set is used to measure the semantic relatedness of a lexical feature F of the OAI snippet of a document D to the corresponding DDC category C. We define the relatedness score $WR_{\mathrm{X}}(F,C)$ of feature F with respect to category C as follows:

$$WR_{\mathrm{X}}(F,C) = 1 - \left(\frac{\max\{\log(f_{\mathrm{X}}(F)), \log(f_{\mathrm{X}}(C))\} - \log(f_{\mathrm{X}}(F \wedge C))}{\log M - \min\{\log(f_{\mathrm{X}}(F)), \log(f_{\mathrm{X}}(C))\}} \right) \text{ (Equation 9)}$$

where $f_{\mathrm{X}}(x)$ is the document frequency, that is, the number of documents of the Wikipedia document collection X in which x occurs, and M is the cardinality of this document collection. This score is either based on exploring the article graph — in this case $\mathrm{X} = art$ — or on the category graph of the Wikipedia — i.e., $\mathrm{X} = cat$. By balancing both sources of relatedness, that is, WR_{art} and WR_{cat}, with the help of a parameter $\beta \in [0,1]$, we get the following formula as an overall measure of the relatedness of F and C:

$$WR(F,C) = \beta \cdot WR_{art}(F,C) + (1-\beta) \cdot WR_{cat}(F,C) \text{ (Equation 10)}$$

This allows us to finally compute the relatedness of a document D and a DDC category C by a mean value:

$$WR(D,C) = |L(D)|^{-1} \sum_{F \in L(D)} WR(F,C) \text{ (Equation 11)}$$

where $L(D)$ is the set of OAI-based features of document D. Finally, we derive a classification rule by analogy to Expression 8:

$$C_i = \underset{C_k \in C}{arg\,max}\{WR(C_k, D_j)\} \text{ (Equation 12)}$$

See (Waltinger and Mehler, 2009) for a thorough description of this approach.

5 Experimentation

In this section, we evaluate the classifiers of Section 4 in relation to baseline scenarios. This is done by classifying documents with respect to the top-level DDC categories based on OAI metadata representations. In this comparative study, we put special emphasis on the SEQ-and the Wikipedia-based classifier as approaches to crossing the frontier of classification schemes into the direction of OTMs.

Further, we subdivide this experiment in two parts (cf. Table 5). In the first part we focus on English documents, while in the second part, we deal with German documents. As we will see in Section 6, the outcomes for both parts are quite different. However, this does not reflect a linguistic divergence, but is caused by a difference in the quality of the OAI metadata of the corresponding input documents. For each language-specific part of our experiment, we evaluate five different classification algorithms:

- Firstly, we build *Support Vector Machine*s (SVM) with the help of SVM$^{\text{light}}$ (Joachims, 2002). We start by stemming tokens and filtering function words to generate a classical VSM (see above). In the case of the German corpus we perform a full lemmatization. Based on the resulting VSM we derive more than 16,000 lexical features to represent input documents. The next step is to learn a separate SVM for each of the 10 target categories (cf. Table 6 and 10). This is done in a one-against-all setting by training linear kernels. The evaluation is performed by means of the leave-one-out method. Note that we decided to use a linear kernel to save training effort — this leaves plenty room for improving our approach.
- Secondly, we start from the same VSM to perform a latent semantic analysis (LSA) in conjunction with k-means clustering, k = 10 (cf. Table 7 and 11). Note that we average the results of k-means clustering over 10 repetitions, while we select 300 main components within the *single value decomposition* step of the LSA. Other than the SVM-based classifier, this approach does not need any training, but only knowledge about the number of target classes. The idea of performing LSA is to come up with a reduced feature matrix which ideally represents more explicitly inherent semantic relations of lexical features. See (Kules *et al.*, 2006) for a thorough description of this approach.
- Thirdly, we vary the latter approach by including frequent phrases as additional features (cf. Table 7 and 11).

- Fourthly, we compute *SEQ*-based classifiers as described in Section 4.4 (cf. Table 8 and 12).
- Fifthly and finally, we implement the Wikipedia-based classifier of Section 4.5 which instead of a classical search engine uses the Wikipedia to derive information about the semantic relatedness of terms (cf. Table 9 and 13).

To evaluate these approaches we compute the *F*-measure or *F*-score (the F-score of a classification is the harmonic mean of its precision and recall) as a standard evaluation technique in IR (Rijsbergen, 1975). The results of this experiment are discussed subsequently.

Class Name	**English**	**German**
DDC 000: Computer science, information	111	100
DDC 100: Philosophy & psychology	115	100
DDC 200: Religion	46	100
DDC 300: Social sciences	45	100
DDC 400: Language	105	100
DDC 500: Science	104	100
DDC 600: Technology	100	100
DDC 700: Arts & recreation	33	100
DDC 800: Literature	24	100
DDC 900: History & geography	36	100
Overall	719	1000

Table 5. Corpus size by language used for evaluation.

DDC	**Precision**	**Recall**	***F*-Score**
000	.889	.943	.915
100	.893	.958	.925
200	.814	.977	.888
300	.829	.918	.871
400	.847	.952	.896
500	.908	.936	.922
600	.675	.895	.770
700	.181	.857	.299
800	.655	.950	.775
900	.222	.888	.355
Overall	.691	.927	.761

Table 6. Results of SVM-based classification of English OAI data.

	Baseline	Phrase	Term
***F*-Score:**	.171	.500	.469

Table 7. *F*-measure results of term- and phrase-based LSA of English OAI data. The baseline classification is performed by a random mapping of input objects to 10 target classes where the classifier is informed about the correct extension of the target classes.

DDC	Precision	Recall	*F*-Score
000	.516	.874	.649
100	.691	.739	.714
200	.730	.852	.786
300	.529	.446	.484
400	.645	.848	.733
500	.786	.740	.762
600	.878	.360	.511
700	.833	.303	.444
800	.706	.500	.585
900	.888	.444	.593
Overall	.720	.611	.626

Table 8. Results of SEQ-based classification of English OAI data.

DDC	Precision	Recall	*F*-Score
000	.525	.563	.543
100	.361	.496	.418
200	.667	.296	.410
300	.500	.278	.357
400	.640	.305	.413
500	.568	.760	.650
600	.439	.290	.349
700	.429	.182	.255
800	.394	.542	.456
900	.143	.444	.216
Overall	.467	.416	.407

Table 9. Results of Wikipedia-based classification of English OAI data.

DDC	Precision	Recall	*F*-Score
000	.911	.720	.804
100	.691	.380	.490
200	.682	.580	.627
300	.564	.310	.400
400	.825	.470	.599
500	.694	.430	.531
600	.509	.290	.369
700	.778	.700	.737
800	.605	.460	.523
900	.625	.300	.405
Overall	.689	.464	.549

Table 10. Results of SVM-based classification of German OAI data.

	Baseline	Term	Phrase
***F*-Score:**	.148	.398	.468

Table 11. *F*-measure results of term- and phrase-based LSA of German OAI data. The baseline classification is performed by a random mapping of input objects to 10 target classes where the classifier is informed about the correct extension of the target classes.

DDC	Precision	Recall	*F*-Score
000	.277	.520	.361
100	.279	.410	.332
200	.315	.640	.422
300	.285	.370	.322
400	.226	.260	.242
500	.500	.180	.265
600	.394	.130	.195
700	.276	.080	.124
800	.370	.272	.314
900	.282	.129	.177
Overall	.320	.299	.275

Table 12. Results of SEQ-based classification of German OAI data.

DDC	Precision	Recall	*F*-Score
000	.546	.650	.594
100	.422	.430	.426
200	.737	.730	.734
300	.410	.160	.230
400	.738	.620	.674
500	.514	.370	.430
600	.431	.690	.531
700	.735	.360	.483
800	.332	.760	.462
900	.319	.150	.204
Overall	.518	.492	.477

Table 13. Results of Wikipedia-based classification of German OAI data.

6 Discussion

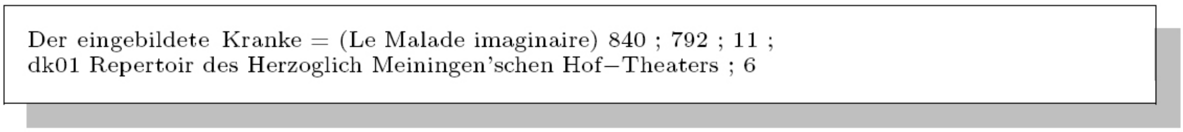
Looking at the results of classifying English documents by their OAI data (cf. Tables 6–9), it is evident that Wikipedia-and LSA-based classifiers produce the lowest *F*-scores (F = .407 and F = .469, respectively). Of course, this finding is conditioned by the scenario under consideration. Notwithstanding this result, we observe that by enhancing the vector space model with the help of frequent phrases we raise the *F*-score to .5. Further, we see that the SEQ-based classifier (cf. Table 8) outperforms the Wikipedia-and LSA-based approach with an *F*-score of .626. This value is much above the outcome of the corresponding random baseline scenario (cf. Table 7), that is, .171. In other words, in the case of more than 60% of the input documents the DDC main class is correctly assigned by exploring OAI metadata snippets — this is much better than a random classifier which is informed about the extension of the target classes. Not surprisingly, the SVM-based classifier performs best (cf. Table 6). With an overall *F*-score of .761, SVMs provide an adequate DDC-related method to classify documents based on their OAI metadata.

That is, in up to 75% of the documents, the classification is correct, *however at the cost of a much higher training effort than induced by the less expensive SEQ-based classifier.*

Regarding the German corpus data (cf. Table 10–13), a poorer performance is observed by means of all five different classifiers. The SEQ-based classifier performs now with an overall *F*-score of .275 — the worst result among all candidates (which is only under-run by the random baseline scenario). In contrast to the English case, the Wikipedia-based classifier (cf.

Table 13) performs now much better, that is, with an *F*-score of .477. Once more, we observe that the LSA-based classifier is enhanced by including frequent phrases into the selection of features. Further, the SVM-based classifiers outperform again all other approaches by an overall *F*-score of .549. In a nutshell, while in the case of the English data, the Wikipedia-based classifier is the poorest performer, the German data is a test case where this classifier is the second-best approach.

The general decline from the English to the German test results can be explained by the descriptive gap induced by a loss of tokens used to build the metadata under consideration. In the case of the English corpus, an OAI summary consists on average of 84 tokens, while in the German texts there are only about 15 — this is a loss of more than 75% of the descriptive units used to induce lexical features. As an example, take the following OAI of a German document (Figure 3):



```
Der eingebildete Kranke = (Le Malade imaginaire) 840 ; 792 ; 11 ;
dk01 Repertoir des Herzoglich Meiningen'schen Hof−Theaters ; 6
```

Figure 3. OAI of a German document.

Evidently, such an input is a little on the short side which makes it too difficult to classify the corresponding document correctly. Of course, the document belongs to literature (i.e., DDC class 800). However, lemmata as *Kranke(r)/invalid* associate unrelated content which may disturb the classifier (note that "Der eingebildete Kranke" is the German title of Molière's "The Hypochondriac").

From this point of view, the classification results produced using the German corpus are remarkably high, especially in the case of the Wikipedia-based classifier, while the search engine-based classifier fails because of the loss of lexical descriptors in the German corpus. Moreover, we may also conclude that the minimum number of lexical descriptors should range above 80 per OAI summary. Of course, to give an exact statement about this number requires further research.

The results of our DDC-related document classification are in a sense promising, in that we may think of DDC classifiers which solely explore document metadata. However, we also learn that these metadata should be extensive enough to prevent misclassifications — *beyond* *what given DDC metadata snippets provide*. Thus, control of the quality of OAI metadata becomes crucial when it comes to building metadata-based classifiers based on OTMs as approached by the SEQ and the Wikipedia based classifiers of Section 4. Actually, we can imply an optimal metadata extension much below the range of full documents, but also above the extension of the German document samples collected in our corpus. Search engine- and Wikipedia-based classifiers are promising candidates to realize this approach. But first and foremost, SVM-based classifiers produce the best result — *at the cost of significantly increased training time and resources*. Although the SEQ-based classifier does not need training, it produces a total amount of 719×300 = 215,700 search queries — this is a secondary source of expense to be considered carefully. Note that the text resources of OAI snippets are research papers, presentations or dissertations with up to 100 pages and more. Therefore, we can expect that the title, summary and keywords of a document will provide sufficient information to classify this document supposed that this feature resource is extended by its abstract. Of course, this is a feasible extension, so we are optimistic about improving the existing range of DDC classifiers — *in support of Hypothesis H1*.

7 Conclusion

In this article we introduced and evaluated several content-related classifiers used in digital libraries. The classifiers explore OAI metadata as a source of document representation and focus on the DDC as a system of target categories. To overcome problems of data sparseness, our approach explores web-based resources such as, e.g., the Wikipedia, to enhance feature extraction and selection. Our evaluation proves the potential of using OAI-metadata-based document representations. However, the *F*-scores of our approach which are below 90% indicate plenty room for improvement. Finally, our evaluation emphasizes the need for controlling the quality of this metadata and for enhancing it by additional document-related information as provided, for example, by abstracts.

Acknowledgement

We gratefully acknowledge financial support of the German Research Foundation (DFG) through the EC 277 *Cognitive Interaction Technology*, the Research Group 437 *Text Technological Information Modeling* and the DFG-LIS-Project *P2P-Agents for Thematic Structuring* *and Search Optimization in Digital Libraries* at Bielefeld University. We also thank Bielefeld University Library which kindly provided the test data used in this article.

References

Biber, D. (1995), *Dimensions of Register Variation: A Cross-Linguistic Comparison*, Cambridge University Press, Cambridge.

Bickhard, M. H. (2008), “Social ontology as convention”, *Topoi*, Vol. 27, No. 1-2, pp. 139–149.

Cilibrasi, R. L. and Vitanyi, P. M. B. (2007), “The google similarity distance”, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 19, No. 3, pp. 370–383.

Cramer, I. (2008), ”How Well Do Semantic Relatedness Measures Perform? A Meta-Study.”, in Bos, J. and Delmonte, R. (Eds.), *Semantics in Text Processing. STEP 2008 Conference Proceedings*, volume 1 of *Research in Computational Semantics*, College Publications, pp. 59–70.

Cutting, D. R., Karger, D. R., Pedersen, J. O. and Tukey, J. W. (1992), “Scatter/gather: A cluster-based approach to browsing large document collections”, in Belkin, N. (Ed.), *Proceedings of the 15th Annual International Conference on SIGIR ’92*, ACM Press, New York, NY, pp. 318–329.

Denoyer, L. and Gallinari, P. (2003), “A belief networks-based generative model for structured documents. An application to the XML categorization”, in Perner, P. (Ed.), *Proceedings of Machine Learning and Data Mining in Pattern Recognition, Third International Conference, MLDM 2003 (Leipzig, Germany)*, volume 2734 of *LNCS*, Springer, Berlin, pp. 328–342.

Dillon, A. (2008), “Bringing genre into focus: Why information has shape”, *Bulletin of the American Society for Information Science and Technology*, Vol. 34, No. 5, pp. 17-19.

Hagedorn, K., Chapman, S. and Newman, D. (2007), “Enhancing search and browse using automated clustering of subject metadata.”, *D-Lib Magazine*, Vol. 13, No. 7.

Halliday, M. A. K. and Hasan, R. (1989), *Language, Context, and Text: Aspects of Language in a Socialsemiotic Perspective,* Oxford University Press, Oxford.

Hearst, M. A. and Pedersen, J. O. (1996), “Reexamining the cluster hypothesis: Scatter/gather on retrieval results”, in Frei, H.-P. and Harman, D. (Eds.) *Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Hartung-Gorre, Konstanz, pp. 76–84.

Heyer, G., Quasthoff, U. and Wittig, T. (2006), *Text Mining: Wissensrohstoff Text*, W3L, Herdecke.

Joachims, T. (2002), *Learning to classify text using support vector machines*, Kluwer, Boston.

Kules, B., Kustanowitz, J. and Shneiderman, B. (2006), “Categorizing web search results into meaningful and stable categories using fast-feature techniques”, in *JCDL ’06: Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries*, ACM, New York, NY, pp. 210–219.

Landauer, T. K. and Dumais, S. T. (1997), “A solution to Plato’s problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge”, *Psychological Review*, Vol. 104, No. 2, pp. 211–240.

Lossau, N. (2004), “Search engine technology and digital libraries: Libraries need to discover the academic internet”, *D-Lib Magazine*, Vol. 10, No. 6.

Maarek, Y. S., Fagin, R., Ben-Shaul, I. Z. and Pelleg, D. (2000), “Ephemeral document clustering for web applications”, Technical report RJ 10186, IBM Research.

Mehler, A. (2009), “A quantitative graph model of social ontologies by example of Wikipedia”, in Mehler, A., Sharoff, S. and Santini, M. (Eds.), *Genres on the Web: Computational Models and Empirical Studies,* Springer, Berlin/New York, pp. 291-352.

Mehler, A., Geibel, P. and Pustylnikov, O. (2007): “Structural classifiers of text types: Towards a novel model of text representation”, *Journal for Language Technology and Computational Linguistics (JLCL)*, Vol. 22, No. 2, pp. 51–66.

Mehler, A., Gleim, R., Ernst, A. and Waltinger, U. (2008), “WikiDB: Building interoperable wiki-based knowledge resources for semantic databases”, *Sprache und Datenverarbeitung. International Journal for Language Data Processing*, Vol. 32, No. 1, pp. 47–70.

Mehler, A., Sharoff, S. and Santini, M. (Eds.) (2009), *Genres on the Web: Computational Models and Empirical Studies*, Springer, Berlin/New York.

Meyer zu Eißen, S. (2007), *On Information Need and Categorizing Search*, Dissertation, University of Paderborn, Feb 2007, available at: http://ubdata.uni-paderborn.de/ediss/17/2007/meyer_zu/ (accessed 19 June 2009).

Mika, P. and Gangemi, A. (2004), "Descriptions of social relations", in *Proceedings of the 1st Workshop on Friend of a Friend, Social Networking and the (Semantic) Web*.

Newman, D., Hagedorn, K., Chemudugunta, C. and Smyth, P. (2007), "Subject metadata enrichment using statistical topic models", in *JCDL '07: Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, ACM, New York, NY, pp. 366–375.

OCLC (2008), "Dewey decimal classification summaries. A brief introduction to the Dewey Decimal Classification", available at: http://www.oclc.org/dewey/resources/summaries/default.htm (accessed 15 February 2009).

Osinski, S. and Weiss, D. (2005), "Carrot^2: Design of a flexible and efficient web information retrieval framework", in Szczepaniak, P. S., Kacprzyk, J. and Niewiadomski, A. (Eds.), *AWIC*, volume 3528 of *Lecture Notes in Computer Science*, Springer, Berlin / Heidelberg, pp. 439–444.

Pieper, D. and Summann, F. (2006), "Bielefeld academic search engine (BASE): An end-user oriented institutional repository search service", *Library Hi Tech*, Vol. 24, No. 4, pp. 614–619.

Rehm, G., Santini, M., Mehler, A., Braslavski, P., Gleim, R., Stubbe, A, Symonenko, S., Tavosanis, M. and Vidulin, V. (2008), "Towards a reference corpus of web genres for the evaluation of genre identification systems", in *Proceedings of the 6th Language Resources and Evaluation Conference (LREC 2008), Marrakech (Morocco)*, ELRA, Paris (CD-ROM).

Rosenberg, J. B. and Borgman, C. L. (1992), "Extending the Dewey Decimal Classification via keyword clustering: the science library catalog project", in *ASIS '92: Proceedings of the 55th annual meeting on Celebrating change: information management on the move*, American Society for Information Science, Silver Springs, MD, pp. 171–184.

Rosso, M. A. (2008), "Bringing genre into focus: Stalking the wild web genre (with apologies to euell gibbons)", in *Bulletin of the American Society for Information Science and Technology*, Vol. 34, No. 5, pp. 20-22.

Salton, G. (1989), *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*, Addison Wesley, Reading, Massachusetts.

Salton, G. and Buckley, C. (1988), “Term weighting approaches in automatic text retrieval”, *Information Processing Management*, Vol. 24, No. 5, pp. 513–523.

Santini, M. (2009), “Cross-testing a genre classification model for the web”, in Mehler, A., Sharoff, S. and Santini, M. (Eds.), *Genres on the Web: Computational Models and Empirical Studies,* Springer, Berlin/New York.

Sebastiani, F. (2002), “Machine learning in automated text categorization”, *ACM Computing Surveys*, Vol. 34, No. 1, pp. 1–47.

Sowa, J. F. (2000), *Knowledge Representation: Logical, Philosophical, and Computational Foundations*, Brooks/Cole, Pacific Grove.

Stefanowski J. and Weiss, D. (2003), “Carrot^2 and language properties in web search results clusterings”, in Menasalvas, E. *et. al.* (Eds.), *Advances in web intelligence. Proceedings of the First International Atlantic Web Intelligence Conference, Madrid, Spain*, Lecture notes in computer science 2663, pp. 240-249.

Stein, B. and Meyer zu Eißen, S. (2003), “Automatic Document Categorization: Interpreting the Performance of Clustering Algorithms”, in Günter, A., Kruse, R. and Neumann, B. (Eds.), *KI 2003: Advances in Artificial Intelligence*, volume 2821 LNAI of *Lecture Notes in Artificial Intelligence*, Springer, Berlin, Heidelberg, New York, pp. 254–266.

Stein, B., Meyer zu Eissen, S. and Lipka, N. (2009), “Web genre analysis: Use cases, retrieval models, and implementation issues.”, in Mehler, A., Sharoff, S. and Santini, M. (Eds.), *Genres on the Web: Computational Models and Empirical Studies,* Springer, Berlin/New York.

Valdes-Perez, R., Pesenti, J. and Palmer, C. (2000), “Vivísimo, inc. — enterprise search, federated search and clustering”, available at: http://vivisimo.com/ (accessed 19 June 2009).

van Rijsbergen, C. J. (1975), *Information Retrieval*, Butterworths, London, Boston.

von Ahn, L. and Dabbish, L. (2008), “Designing games with a purpose”, *Communications of the ACM*, Vol. 51, No. 8, pp. 58–67.

Waltinger, U. and Mehler, A. (2008), “Who is it? Context sensitive named entity and instance recognition by means of Wikipedia”, in *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence (WI-2008)*, IEEE Computer Society, pp. 381–384.

Waltinger, U. and Mehler, A. (2009), “Social semantics and its evaluation by means of semantic relatedness and open topic models”, in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, September 15-18, Milano.*

Waltinger, U., Mehler, A. and Heyer, G. (2008), “Towards automatic content tagging: Enhanced web services in digital libraries using lexical chaining”, in *4th International Conference on Web Information Systems and Technologies (WEBIST ’08)*, Springer, Berlin, Heidelberg, pp. 231–236.

Zamir, O. and Etzioni, O. (1999), “Grouper: a dynamic clustering interface to web search results”, in *Proceedings of the Eighth International WWW Conference, Toronto*, pp. 283-296.

Zhang, D. and Dong, Y. (2004), “Semantic, hierarchical, online clustering of web search results”, in *Advanced web technologies and applications. Proceedings of the 6th Asia Pacific Web Conference (APWEB), Hangzhou, China*, Springer, Berlin, pp. 69-78.

About the authors

Alexander Mehler is Professor in the Faculty of Technology, Bielefeld University, Bielefeld, Germany. Alexander Mehler is the corresponding author and can be contacted at: alexander.mehler@uni-bielefeld.de

Ulli Waltinger is a Scientific Assistant in the Faculty of Technology, Bielefeld University, Bielefeld, Germany.