Descriptor-invariant Fusion Architectures for Automatic Subject Indexing

Analysis and Empirical Results on Short Texts

Martin Toepfer
ZBW – Leibniz Information Centre for Economics
Düsternbrooker Weg 120
Kiel, Germany 24105
m.toepfer@zbw.eu

Christin Seifert
Universität Passau
Innstraße 33
Passau, Germany 94032
christin.seifert@uni-passau.de

ABSTRACT

Documents indexed with controlled vocabularies enable users of libraries to discover relevant documents, even across language barriers. Due to the rapid growth of scientific publications, digital libraries require automatic methods that index documents accurately, especially with regard to explicit or implicit concept drift, that is, with respect to new descriptor terms and new types of documents, respectively. This paper first analyzes architectures of related approaches on automatic indexing. We show that their design determines individual strengths and weaknesses and justify research on their fusion. In particular, systems benefit from statistical associative components as well as from lexical components applying dictionary matching, ranking and binary classification. The analysis emphasizes the importance of descriptor-invariant learning, that is, learning based on features, which can be transferred between different descriptors. Theoretic and experimental results on economic titles and author keywords underline the relevance of the fusion methodology in terms of overall accuracy, and adaptability to dynamic domains. Experiments show, that fusion strategies combining a binary relevance approach and a thesaurus-based system outperform all other strategies on the tested data set. Our findings can help researchers and practitioners in digital libraries to choose appropriate methods for automatic indexing.

CCS CONCEPTS

•**Computing methodologies** →*Supervised learning; Machine learning;* Natural language processing; •**Information systems** →**Digital libraries and archives;**

KEYWORDS

automatic subject indexing, meta-learning, multi-label classification, keyphrase indexing, zero-shot learning, short text

ACM Reference format:
Martin Toepfer and Christin Seifert. 2017. Descriptor-invariant Fusion Architectures
for Automatic Subject Indexing . In *Proceedings of –, –, –,* 10 pages.
DOI: --/--

–, –
2017. 978-x-xxxx-xxxx-x/YY/MM…$–
DOI: --/--

1 INTRODUCTION

Literature access is best supported by subject indexes constructed using domain-specific, controlled vocabularies and thesauri. Such structured representations enable semantic queries and discovery even across language barriers and they provide features for services like literature recommendation systems. Due to the rapid growth of scientific publications [2], scalability of the indexing process has become essential making automatic subject indexing a key technology for digital libraries.

Compared to traditional manual indexing, automatic indexing faces several challenges: First, legal restrictions might prevent the usage of publication full-text and/or abstracts, which leads to little information available to the indexing approach and thus decreases performance [7]. Second, the distribution of concepts in the training data set can be very skewed and some concepts might not appear at all [21]. This is especially likely for conceptual thesauri containing several thousands of concepts, as for example EuroVoc vocabulary[1], medical subject headings (MeSH)[2], AGROVOC[3] in the agricultural domain, or the economic thesaurus STW[4]. Concepts with little or no document coverage have to be either excluded [21] or require carefully designed feature spaces and concept representations for so-called zero-shot learning approaches [20]. Third, terminology in documents and controlled vocabularies might differ or terminology might change over time. For instance, permanent updates of the STW were performed to reflect substantial changes in economics-related literature [9]. Consider phrases like "online advertising" or "smartphone" that suddenly appeared since 1990 [18], just to give an example. Thus, automatic indexing approaches must be capable of adapting to explicit and implicit concept drift, i.e. to vanishing or emerging concepts and new types of documents containing unseen terms. Consequently, this requires descriptor-invariant learning approaches, that is, learning based on features, which can be transferred between different descriptors.

Research in the field of automatic indexing with controlled vocabularies can be broadly categorized into lexical and associative approaches. *Lexical approaches* like, for example, KEA++ [17] build upon knowledge provided by thesauri to find candidate concepts. Subsequently candidates are ranked and selected according to their relevance. As pointed out by Medelyan and Witten [17], this procedure requires only hundreds of training examples in total. But it

[1] www.eurovoc.europa.eu/ (accessed: 30.01.2017)
[2] www.nlm.nih.gov/mesh/ (accessed: 30.01.2017)
[3] www.fao.org/agrovoc (accessed: 30.01.2017)
[4] www.zbw.eu/en/stw-info/ (accessed: 30.01.2017)

comes at a cost. Lexical approaches will fail on missing candidates and incomplete vocabulary. In terms of Pouliquen et al. [21], a natural language thesaurus is required which nearly exhaustively covers the terminology of the domain. Construction and maintenance of such lexical resources is costly, thus many thesauri provide concepts but lack vocabulary entry terms, especially if multiple languages are involved. In this case, *associative approaches* may be more appropriate. They rely on associations between terms and concepts that are derived from large intellectually indexed document collections [21]. Especially, a multitude of supervised learning approaches has been proposed driven by advances in artificial intelligence and machine learning where indexing has been regarded as a multi-label learning task [10]. In essence, these approaches involve training classifiers for each concept of a thesaurus. Encouraging results have been reported in different domains, for instance, in Medicine [12, 25], Agriculture [13], Legal Texts [14], or Economics [11]. Such approaches enable automatic indexing with conceptual thesauri [21] when a lot of professionally indexed examples are available, however, they do not scale well in terms of necessary training data [17]. Researchers attempted to combine elements from associative and lexical approaches aiming to alleviate their disadvantages (e.g., [6, 12, 19, 22]) with *fusion architectures*, meta-learning, or zero-shot learning techniques. Nevertheless, fusion architectures are still an exception rather than the rule, no thorough analysis of single and fusion architectures has been performed yet, and fusion can be realized in different ways. In this paper, we aim for a detailed analysis of associative, lexical and fusion architectures supported by an empirical study of a new fusion approach in the domain of economics that especially considers dynamics in terms and concepts.

Performance of automatic subject indexing systems is influenced by several factors, raising questions about generalizability. Attempts to conduct large-scale experimentation and to empirically determine successful configurations [11] provide important feedback for researchers and practitioners, but they should be supplemented by analytical justifications if possible. Recently, there have also been concerns about just concentrating on better results on standard benchmark data and how techniques like deep learning have been applied in the field of computational linguistics. For instance, Manning wanted to "encourage everyone to think about problems, architectures, cognitive science, and the details of human language, how it is learned, processed, and how it changes, rather than just chasing state-of-the-art numbers on a benchmark task" [15, p. 706]. Following this advice, we aim to gather knowledge about reasonable architectures for automatic subject indexing systems, understanding their success and pitfalls. Our work focuses on an economic data set but it provides a detailed analysis that may help researchers and practitioners in other domains. The contributions of this paper are the following:

- We provide a detailed analysis of indexing system architectures, outlining advantages and disadvantages.
- Based on the analysis, we propose descriptor-invariant fusion to combine predictions of different indexing methods in order to mitigate their shortcomings and to handle explicit and implicit concept drift.
- We demonstrate the advantages of the proposed approach empirically in the domain of economic working papers. Our experiments especially consider scalability in terms of accuracy and performance in scenarios with explicit and implicit concept drift.

In the remainder of the paper, we will first review the background and related work in Section 2 before we analyse the existing indexing architectures in detail in Section 3. Based on the theoretical analysis we then describe our approach to a fusion architecture that combines lexical and associative characteristics in Section 4. Results of experiments on documents from the economic domain are presented in Section 5. Finally, Section 6 concludes the work and outlines directions for future research.

2 BACKGROUND AND RELATED WORK

Subject indexing is the process of selecting concepts from a controlled vocabulary like a thesaurus in accordance with certain criteria. It typically aims to cover the main topics of a document exhaustively and describe them as precisely as possible, while seeking a condensed representation that contains, for instance, about 6 concepts on average. *Automatic subject indexing* attempts to implement this task algorithmically. It is sometimes called *keyword assignment* or *keyphrase indexing* synonymously. We will also refer to concepts of the controlled vocabulary as *descriptors* which represent abstract units of thought. According to the Simple Knowledge Organization System (SKOS)[5], natural language expressions that refer to concepts are called *labels*[6]. There may also be links between concepts that encode hierarchical (broader/narrower) or associative (related-to) semantic relations. Subject indexing approaches can bee broadly categorized into *statistical associative* and *lexical* approaches. In this section, we first describe the most commonly used vocabularies and then review related work for statistical associative and lexical approaches. A characterization, analysis and comparison is provided in Section 3.

Controlled Vocabularies. AGROVOC[3] covers 32,000 concepts in 27 languages in the area of food, nutrition and agriculture. All concepts have one preferred term (descriptor) and alternative labels (e.g. in different languages), concepts are organised hierarchically (skos:broader and skos:narrower) and also also contain non-hierarchical relations (skos:related). The medical subject headings (MeSH) vocabulary[2] contains approx. 28,0000 descriptors from the medical domain. Additionally to the descriptors the vocabulary provides approx. 87,000 terms and 232,000 supplementary concept records linked to descriptors. EuroVoc[1] covers multiple disciplines related to activities of the EU in 23 languages. It contains approx. 6,500 descriptors and between 150 and 13,000 non-descriptor terms (depending on the language).

In this work, we use the STW Thesaurus for Economics[4]. It is a wide-coverage bilingual resource (German and English) for economics and economics-related subject areas. The current release (9.02) has more than 6,000 subject headings, more than 20,000 synonyms, and links broader, narrower, and semantically related concepts. Descriptors are additionally categorized using a mono-hierarchy of subject groups (thsys), in the following called *categories*.

[5] www.w3.org/2004/02/skos/

[6] In related work, especially in the domain of machine learning, the term "label" is often used for classes, that is, concepts.

Automatic Subject Indexing. Ferber [7] developed a system with a linear *associative* model that was based on titles (*short text*) and co-occurrence data between words and descriptors. He reported encouraging results but noted that titles were sometimes insufficient and that it was unclear if the co-occurrence approach generalizes to different domains. Pouliquen et al. [21] investigated indexing with EuroVoc and found that only approximately one third of all training documents contained labels of the corresponding descriptors verbatim. For this reason, they distinguished between *conceptual thesauri* like EuroVoc and *natural language thesauri*. Because the former lack vocabulary terms for dictionary matching approaches, they proposed to determine associate terms, that is, statistically related terms, for descriptors with a statistical system similar to Ferber. Pouliquen et al. determined these associate lists by log-likelihood and then assigned descriptors by a linear combination of three similarity measures. They were able to apply the approach successfully to different languages, however, it frequently assigned descriptors that were semantically similar but wrong. Lauser and Hotho [13] indexed full-text documents in the agricultural domain with binary support vector machines (SVM). They explored modes (*add*, *replace*, *only*) to encode background knowledge from an ontology which modify the feature vectors by adding, replacing or restricting features to ontology concepts, respectively. Relations between concepts were used up to a maximum concept integration depth. Only adding concepts showed slight improvements in precision. Loza and Fürnkranz [14] automatically indexed legal documents of the EU using three different multi-label classification approaches based on perceptrons: binary relevance, multiclass multi-label perceptrons, and multi-label pairwise perceptrons. Pairwise classification into almost 4,000 classes of EuroVoc required almost 8,000,000 perceptrons. As a consequence, they had to solve severe scalability issues. Wilbur et al. [25] showed on a subset of MeSH headings that stochastic gradient descent applied to SVMs (SGD-SVM) performed well with a fixed number of iterations for ranking and prediction. It produced better results than several methods, including MTI, kNN-based systems, and a learning-to-rank approach.

Besides these associative approaches, automatic subject indexing has also been regarded as a controlled vocabulary extension to *keyphrase extraction*, which aims to determine the most relevant phrases of full-texts to describe the content. Systems like KEA [8] therefore rank terms according to several features like their term frequency and inverse document frequency (TF-IDF). Models that combine and weight features can be estimated when training examples are available. As shown by Medelyan et al. [17], a modified version of KEA, called controlled keyphrase indexing algorithm (KEA++), can be used for subject indexing when a thesaurus with appropriate labels is available. Their approach filters the full-text by matching of pseudo-phrases, that is, conflated versions of the documents' terms and a controlled vocabulary's labels. This approach has been evaluated on different domains (agriculture, medical, physics) and it was especially pointed out that it already performs well with little training data. KEA++ led to the development of maui [16] which can use additional features and bagged decision trees instead of a Naïve Bayes Classifier.

Große-Bölting et al. [11] evaluated several configurations for semantic document annotation on three data sets. Different annotation candidate extraction and activation methods were combined with two kinds of selection approaches: top-k and k-nearest-neighbors. Their best results on a data set with 62,924 economic documents (full-text) were produced by kNN ($k = 1$; micro-averaged F_1 value of .39).

Nam et al. [19] aimed to predict previously unseen non-terminal concepts in concept hierarchies. They proposed a joint space of instances and concepts, using hierarchical information and concept co-occurrence patterns. Experiments were conducted on two data sets. The authors stated that the regularization approach was effective to predict previously unseen classes when the tree-structure of classes is known and not complex. A pre-training strategy was proposed that empirically improved results even on large sets of classes. Recently, Sappadla et al. [22] proposed an approach in order to exploit similarities between concept labels and document terms. To predict known concepts, they used a supervised method (binary relevance), whereas unknown concepts were predicted using label word similarity by word embeddings based on Wikipedia. They evaluated their system on three fulltext data sets. The number of classes to predict were 90 (Reuters), 45 (MEDICAL), and 201 (EURLEX). The average sizes of assigned labels were 1.23, 1.24, and 2.21, respectively. These figures are close to 1, hence, close to single-label multi-class classification. Experimentally, they were able to show advantages of their approach against a supervised baseline. When labels were removed by their frequency from evaluation, using similarity knowledge led to higher macro-averaged metrics.

Research on automatic subject indexing has been very active in the (bio-)medical domain. Notably, Jimeno-Yepes et al. [12] combined different subsystems to index MEDLINE citations with medical subject headings (MeSH). Their baseline system was the Medical Text Indexer (MTI) which was compared to several machine learning approaches (Naïve Bayes, Rocchio, AdaBoostM1, Voting) and dictionary matching on titles and titles and abstracts. They learned a mapping-table that determined which method is to be used for each MeSH heading (MH). In order to select the best method, they applied significance tests. They found that more than 23,000 MHs were best indexed by MTI, while machine learning approaches were chosen for 2,712 MHs. Combinations of machine learning methods have also been applied for categorization of genomics documents by Aronson et al. [1] who used the term *fusion* in the sense of *ensemble* or *stacking* [24, 26]. Please note that it differs from fusion architectures as understood in this paper (cf. Section 3).

Erbs et al. [6] pointed out some differences between keyphrase extraction and multi-label classification (MLC) approaches, underlined certain advantages of MLC like detecting hidden synonyms and keyphrase extraction, and presented an approach which adds keyphrase extraction results to the list of terms returned by MLC. SVMs and decision trees were used for MLC and different configurations with TF-IDF for keyphrase extraction. They focused on full-text representations of German documents in the educational domain in their evaluation. The combined system reached 20% precision and 17.9% recall. Different from our approach, they investigated keyphrase extraction, that is, index terms were part of the documents' terms (uncontrolled vocabulary).

Table 1: Pros (+: advantage) and cons (-: disadvantage) of lexical (L) and associative (A) system architectures according to challenges in automatic subject indexing.

	Aspect	L	A
A1	Amount of required training data	++	-
A2	Prediction of unseen concepts	++	- -
A3	Prediction of synonyms	- -	++
A4	Ambiguity	o	+
A5	Exploitation of thesaurus relations	+	o
A6	Applicability to short texts	o	o

The problem of predicting previously unseen classes has been studied in other domains before, in so-called zero-shot learning settings. For instance, Palatucci et al. [20] presented an approach that uses a knowledge base to decode neural activity. As they pointed out, it is desirable to treat classes not separately from each other, but to create representations that apply to many, also unseen classes. How automatic subject indexing can be best realized in this regard is a current research question. Recently, some aspects have been targeted, like predicting non-terminals [19] or using label embeddings for settings that are close to single-label classification [22].

3 ANALYSIS OF INDEXING SYSTEMS

In this section, we analyse architectures of indexing systems and outline strengths and weaknesses that can be derived theoretically. The analysis is independent of specific implementations. It focuses on the way how background knowledge is used and how the approaches scale with respect to growth of the controlled vocabulary. We will base our discussion on the aspects depicted in Table 1, namely (A1) the amount of training data required (low is better), (A2) wether previously unseen concepts can be predicted (desirable), (A3) wether synonyms can be predicted (desirable), (A4) whether ambiguity can be resolved (desirable), (A5) whether relations of concepts in the thesaurus are used (desirable), (A6) the applicability for short texts.

For the discussion we will use the following, small example of a document with author keywords and professional indexing terms:

> **Title:** Analysis of the German gas price from 1970 to 1980
> **Author key words:** Germany, energy pricing, gas, 70s
> **Indexing terms:** c: gas price, c:Germany

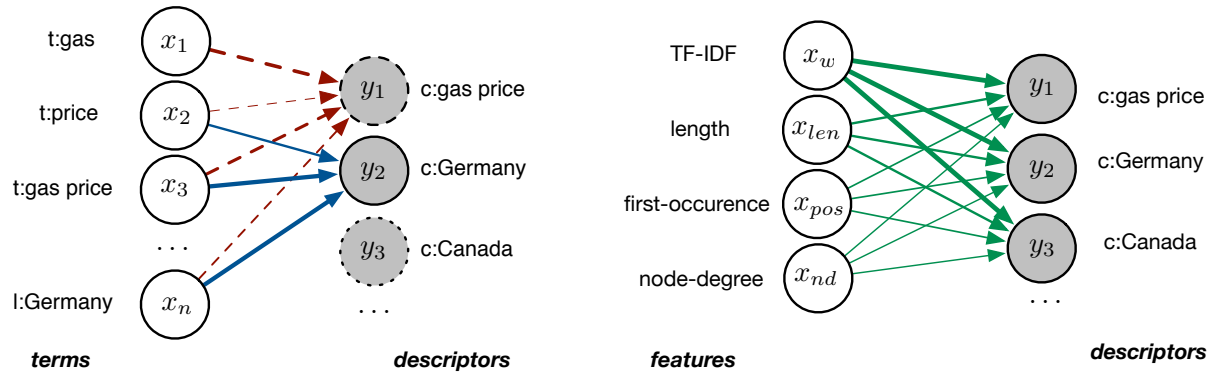
Different prefixes are used to refer to different types of features: terms / word n-grams (t), dictionary matches to labels of concepts (l), concepts, i.e. descriptors (c).

Figure 1a shows a prototypical **associative indexing** system for the example document. On the left, we can see features like the term feature "t:gas" or a match of a certain concept label "Germany" that encode the document. Typically, one feature is created for each unique n-gram of the training documents resulting in a large number of features. On the right hand side are class nodes that encode concepts that might be assigned by the system, for instance, "c:gas price". Under this representation of documents and their concepts, systems operate on sparse representation, that is in terms of the document-feature matrix most of the entries are zero. Machine learning is used to determine how features and individual concepts relate to each other. Generally, co-occurrence statistics are used to describe concepts and discriminate them from other concepts. In this methodology, it is therefore possible to derive associations from data such as that the term "t:FRG" is a positive indicator for the descriptor "c:Germany (Federal Republic)". Broadly speaking, unknown synonym expressions can be learned from the data (A3). Parameters that finally determine if a concept is assigned are learned independently for each descriptor. In Figure 1a, parameters of a classifier (encoded by color) and their weights (encoded by line thickness) are shown as arrows between terms and descriptors y_1 and y_2. No weights have been learned for y_3 (c:Canada) because no training instance was available for this concept. As a consequence, this concept can not be assigned to any document (A2). Even if we add concept features for matches against the thesaurus to the feature vector [11, 13] to encode background knowledge, descriptor-specific parameter learning makes it impossible to assign concept c:Canada when no training example is available for this descriptor. For each descriptor in the thesaurus, at least one training example is required (A1).

A prototypical **lexical indexing** system is illustrated in Figure 1b using features from KEA++ [17] as an example. Based on lexical knowledge from a thesaurus, the system determines several concepts as candidates. The output is determined by repeated application of the same classifier, as shown by duplicates (y_1, y_2, y_3) of the same node template for all concept candidates (c:gas price, c:Germany, c:Canada) in Figure 1b on the right hand side. The lexical system shares the same feature weights (green arrows) for all descriptors. This means, a lexical systems learns the best feature combinations in terms of weighting factors for the features. In the example the classification rule is $y_i = w_1 \cdot x_{\text{tf-idf}} + w_2 \cdot x_{\text{len}} + w_3 \cdot x_{\text{pos}} + w_4 \cdot x_{\text{nd}}$ with $w_1 = 2, w_2 = 1.2, w_3 = 0.7, w_4 = 0.34$ (as an example). This classification rule is applied with the same weights for each concept candidate yielding a score for each descriptor based on the features of a given input document. The final descriptor assignment is then based on this score for the concept candidates. Notably, there is only a small number of features for each candidate and the feature representation will be rather dense because each of the four features in the example has a value for each candidate. The system learns weights that are re-usable, even for previously unseen concepts like, for example, c:Canada. Consider that we apply the system to a new document that contains the term "Canada" which is recognized during concept candidate generation by dictionary matching. The system then computes TF-IDF, length, first-occurence and node-degree features for this match. Subsequently, the same parameters that have been optimized for other descriptors are utilized to decide if the descriptor of Canada should be assigned. It can successfully be added to the list of proposed descriptors (A2). Only a limited number of parameters have to be fit and only a few documents are required for training [17] (A1). But it comes at a cost. The approach is unable to learn synonymous expressions from data (A3).

Based on these insights, we propose to use *descriptor-invariant learning* and *descriptor-invariant prediction* as key properties to characterize automatic subject indexing approaches.

Definition 3.1. An automatic subject indexing system performs *descriptor-invariant learning*, when it optimizes its behavior based



(a) Associative indexing

(b) Lexical indexing

Figure 1: Comparison of architectures by example. a) In associative indexing, the learning algorithm learns relations between features, which are terms (t:) or dictionary matches (l:), and descriptors (c:) for each descriptor independently. b) In lexical indexing, features are computed for concept candidates derived from the document's terms. Feature weights are shared among all descriptors for classification.

on examples and generalizes experiences, such that previously unseen descriptors can be assigned.

Descriptor-invariant prediction is just the ability to predict previously unseen concepts and does not require behaviour to be optimized according to examples.

To underline the differences between descriptor-invariant learning in lexical systems and descriptor-specific learning (performed for each distinct descriptor separately) in associative systems, let us consider the use of relations between concepts retrieved from background-knowledge (A5), like "c:price" is broader than "c:gas price". As shown in Figure 1b, it has been proposed by Medelyan et al. [17] to compute a *node-degree* feature that measures how strong a concept candidate is connected to other candidates in the same document. Parameters are shared among descriptors and learning is therefore based on many examples. The importance of this feature can be confidently estimated and generally applied. In associative systems, concept features can be activated based on different schemes [11, 13]. Learning and prediction remain, however, restricted if only concepts from the training data can be predicted like in kNN classification or if individual classifiers are learned for each descriptor.

Natural language is inherently ambiguous (A4) and word senses have to be determined in order to understand a text. Associate approaches can learn to solve this task using arbitrary words in context, but remain limited to known concepts and words from training data. Lexical approaches depend in their performance on the controlled vocabulary. If enough candidates can be extracted, features like node-degree or descriptor co-occurrence expectations may enable to determine the sense of a phrase.

Please note that *combinations of associative systems* through strategies like voting or stacking typically *remain associative* in their design and lack descriptor-invariant characteristics. We also emphasize that with regard to descriptor-invariant prediction, the most important part of a system is the final layer. Even if components like dictionary matching are used to inject background-knowledge in early stages, building distinct classifiers with individual parameters for each descriptor or looking up candidates in a data base in the last layer or intermediate layers limits the system to the set of descriptors that are already known.

In principle, both associative and lexical approaches can be applied to short texts (A6), however, certain phenomena might be more pronounced and should be considered during configuration when only a few terms are available. For instance, the node-degree feature of lexical systems may not find enough related candidates in very short text for meaningful operation.

Scalability often considers the processing time of an algorithm in relation to amount of data to be processed. In automatic subject indexing, not only the number of documents, but also the number of concepts relevant for the documents, grows. Practitioners at digital libraries are therefore also interested in how methods adapt to changes in the terminology and how they deal with descriptors for which only a few examples are available. Please recall that concepts often follow a *power law* [6], thus, there are few concepts that occur quite frequently while the majority of concepts occurs rarely. In this context, learning separate parameters for each concept is subject to miss training data in too many cases. If a system is, however, able to re-use the same parameters that have been learned for other, similar concepts, no further learning might be necessary at all.

In summary, we conclude, that descriptor-invariant lexically-based classification and associative classification provide distinct capabilities in order to achieve accuracy and scalability. A comprehensive overview of advantages and disadvantages of both systems can be found in Table 1. Descriptor-invariant learning is of utmost importance to enable prediction of previously unseen descriptors.

4 DESCRIPTOR-INVARIANT FUSION

In the last section, we have seen that approaches that are solely lexical or solely associative fail on some challenges of automatic indexing but also have individual strengths. Therefore it seems reasonable to attempt a fusion of both approaches by combining

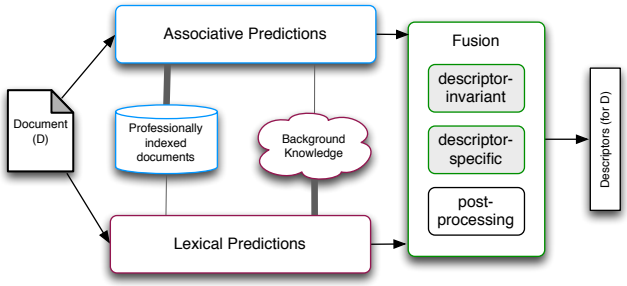


Figure 2: Schema of a fusion system.

the individual predictions. The interesting questions are, however, how fusion is actually realized and which pitfalls have to be avoided.

The top level design of the proposed fusion architecture is depicted in Figure 2. First, different candidate sets are produced: either by an associative component (center, top) that leverages a large set of professionally indexed documents or by a lexical system (center, bottom) that relies on background knowledge from a thesaurus. Then, the fusion layer (right) is responsible for combining these predictions. The most interesting property of this layer is the *descriptor-invariant decision function*, i.e. a function that can perform predictions for all (also unseen) descriptors. Optionally, the fusion module may additionally consult the knowledge base or the professionally indexed documents for its decisions and use a descriptor-specific fusion component.

Within the fusion layer, it is crucial how the predictions are combined. On the one hand, one may learn on a basis of descriptors (descriptor-specific fusion), for example, learning mapping tables [12] using confidence tests. In a similar but different manner, we can simply compute for each descriptor c and method m the support (number of documents with c assigned by m) and confidence (number of c correctly proposed by m divided by its support) for each descriptor c based on held-out data of the training set. Descriptors that surpass a minimum support and a minimum confidence may then be added by m to the final output in a production setting (testing). This simple strategy, in the following referred to as *Rhack*, is slightly different from mapping tables that map descriptors to methods [12]. While the latter may learn that the concept "theory" is better predicted by the associative component than by the lexical component and therefore will choose to *always* handle it by the associative system, Rhack will simply join their predictions and assume that both are reliable. We suggest that both kinds of behavior are not optimal in general because they are again restricted to the set of known descriptors from the training documents. They will not be able to determine a suitable predictor for the term "Canada" if this term is not present in the training documents.

Therefore, a fusion decision function should be implemented that is invariant to descriptors. In order to investigate the potential of the proposed design, we construct a very straight-forward system. We study the union of predictions per document, which we denote by $f_\circ(*)$ in the following. This strategy is derived from the idea of setting the above-mentioned minimum confidence and support to zero in the fusion layer, but expands predictions to previously unknown concepts. Each subsystem may, however, still filter its predictions by an individual confidence threshold. This is indeed essential to guarantee high precision in the fusion system. The union approach is straightforward, however, it has some interesting aspects and especially enables us to explore if higher recall can be reached by fusion. Following the discussion of existing architectures in the previous section, we observe that:

- Associative systems may suffer from low recall, because the data they learn from is likely to be insufficient. Terms and concepts follow power laws, hence, many concepts and terms are infrequent.
- Lexical systems may suffer from low recall, when the knowledge base lacks synonymous expressions, especially when texts are short and therefore less candidates are generated per document.

For these reasons, gaining recall in the fusion layer seems to be crucial and it may be a promising way to join predictions for better overall performance.

Beside choosing between descriptor candidates from the subsystems, we also investigate *post-processing* aspects of the fusion layer. During fusion, systematic errors of individual modules might be corrected with supervision that builds upon predictions, professionally indexed documents, and background-knowledge from the thesaurus. Inspired by ideas from transformation-based error-driven learning [4], we investigate a *transformation-rule learning* module. For each pair of categories (k_1, k_2) in the thesaurus, it counts cases on held-out data of the training examples where a descriptor $c_1 \in k_1$ was predicted erroneously while a related descriptor $c_2 \in k_2$ was missed at the same time. It then attempts to increase performance on the training data with a transformation rule (switch every prediction of c_1 by c_2). If it succeeds, this rule is added to a list of rules that are used in production to index new documents. For instance, we may learn a rule that replaces candidates c_1 by c_2 if c_1 is a geographic adjective or language (e.g. "German"), c_1 and c_2 are related concepts as defined in the thesaurus, and c_2 is a geographic name (e.g. "Germany"). Interestingly, such transformation rules may predict previously unseen concepts when they consider types of descriptors instead of descriptors themselves; the example rule above applies to "Canadian" even when "Canada" was not part of the training data.

In the presented framework, associative predictions and lexically-based prediction modules may be implemented by different methods. In the experiments, we will especially consider two state-of-the-art approaches: maui [16] to produce predictions with lexical background knowledge and SGD-SVM [25] for prediction in an associative way. Inside of the lexical layer, maui [16] provides a mature thesaurus-based system with a rich set of features that goes beyond simple dictionary matching. In our case, it can, however, be assumed that different features are required to realize the full potential of short texts like titles or author keywords. For instance, maui's span feature aims to weight terms higher that are mentioned in the abstract and the conclusions, which are however not accessible in this case. We leave the invention and integration of new features for future work and suspect that maui's supervised learning method (bagged decision trees [3]) will still be able to create a robust prediction component, even when applied to short texts.

5 EXPERIMENTS

With the experiments we wanted to answer the following three experimental questions: i) How do fusion systems compare to associative and lexical approaches in terms of overall accuracy? Are the approaches robust to ii) explicit concept drift and iii) implicit concept drift? Explicit concept drift is modelled by a test data set containing descriptors from certain categories that are not present in the training data set. To assess implicit concept drift we evaluate the trained models on an unknown series of documents, which may cover different topics. We performed the experiments on short texts from the economic domain and using the STW thesaurus (cf. Section 2).

5.1 Data Set

Our data set consists of documents represented by their titles and author keywords only. This information is available even in indexing scenarios where abstracts or full-texts are either missing (in the case of books) or not accessible due to legal aspects. We represent the documents as described in Section 3. The complete sample contains 20,195 documents, indexed by professional indexers. Indexers assigned 5.85 (SD = 1.84) descriptors per document on average. 94% (19,054) of the documents have a unique combination of descriptors assigned.

To compare i) the overall performance of the different approaches we split the data set randomly into training and test sets using 5-fold cross-validation (data set denoted by $\mathcal{D}_{\text{shuffle}}$).

In order to measure the influence of ii) *explicit concept drift*, we created data sets denoted $\mathcal{D}_{\text{cat}}$, where we split the documents according to certain subthesauri (categories), that is, subject fields. We used sets of classification scheme codes ("thsys" codes) of the STW for which we ensured that they were not used during training. For instance, one training set of $\mathcal{D}_{\text{cat}}$ does not contain documents with descriptors from the field "marketing" (thsys: B.07), but all test documents cover some descriptors from this category, for instance, market share, competition, or customers. Consequently, this setting emphasizes the zero-shot learning task.

To investigate the influence of iii) *implicit concept drift*, we split documents into sets $\mathcal{D}_{\text{series}}$ according to publication series. For example, one single working paper series which covers subjects like regional business growth programmes or human capital may be omitted from training. The corresponding test set includes only documents from this series.

Table 2 provides an overview of the different data sets. The average number of assigned concepts is the same on training and testing for the random splits $\mathcal{D}_{\text{shuffle}}$, but it differs on $\mathcal{D}_{\text{cat}}$ and $\mathcal{D}_{\text{series}}$, respectively. The explicit and implicit concept drift settings have larger training sets on average, but the size of the corresponding test sets varies. For instance, the test subsets of ${\mathcal{D}_{\text{series}}}^{(\text{test})}$ contain 4742, 748, 415, 385, and 385 documents, respectively.

5.2 Evaluation Metrics

We use common metrics [23] which can be computed in total (micro-average), per concept (macro-average), or per document (sample-based average): precision (correctly predicted descriptor assignments divided by all predicted descriptor assignments), recall (correctly predicted descriptor assignments divided by all descriptors assigned by professional indexers), F_1 score (harmonic mean of precision and recall).

Because the macro-averaging metrics are not weighted by concept count, they show if concepts are recognized accurately independently of their frequencies in the test sets.

Table 2: Properties of settings with respect to professional indexing. $|\{\mathcal{D}_i\}|$: number of different partitions (folds). $|\bar{\mathcal{D}}|$: average number of documents. $|\bar{\mathcal{L}}|$: average number of unique descriptors. $|\bar{\mathcal{Y}}|$: average number of assigned descriptors per document.

Setting	$\lvert\{\mathcal{D}_i\}\rvert$	$\lvert\bar{\mathcal{D}}\rvert$	$\lvert\bar{\mathcal{L}}\rvert$	$\lvert\bar{\mathcal{Y}}\rvert$
${\mathcal{D}_{\text{shuffle}}}^{(\text{train})}$	5	16,156	3,848.8	5.85
${\mathcal{D}_{\text{shuffle}}}^{(\text{test})}$	5	4,039	2,777.2	5.85
${\mathcal{D}_{\text{cat}}}^{(\text{train})}$	5	17,490	3,812.8	5.78
${\mathcal{D}_{\text{cat}}}^{(\text{test})}$	5	2,705	1,946.0	6.26
${\mathcal{D}_{\text{series}}}^{(\text{train})}$	5	18,860	3,950.0	5.82
${\mathcal{D}_{\text{series}}}^{(\text{test})}$	5	1,335	1,205.4	6.54

5.3 Configurations

As two basic **lexical systems**, we implemented dictionary matching approaches: a simple matching algorithm that only considers phrases between stop words, denoted *dict*, and *monq* which accesses a dictionary matching library[7] that considers morphological variants of terms and which was used in related work [12]. As a strong lexical baseline, we chose *maui*[8] [16]. Maui's maximum number of concepts to assign was set to $k = 15$ and the minimum confidence was set to $c = 0.1$. Please note, however, that maui is typically applied to full text rather than short text.

Associative systems were realized by binary relevance (BR) approaches. We chose to use $\text{BR}^{(\text{LR})}$ (logistic regression classifier) and $\text{BR}^{(\text{SVM})}$ (support vector machines) trained by stochastic-gradient-descent as described by Wilbur et al. [25]. Both, $\text{BR}^{(\text{LR})}$ and $\text{BR}^{(\text{SVM})}$, were configured with word n-gram features between stop-words.

Rhack (cf. Section 4) is a meta-learning approach which is similar in mind to [12]. We configured it to enrich predictions made by $\text{BR}^{(\text{LR})}$ with dictionary matching of *dict*, adding only confident *dict* predictions to the list of descriptors created by $\text{BR}^{(\text{LR})}$. On the training data, it therefore determines all concepts with minimum support (min.sup = 20) and minimum confidence (min.conf = 50%). These estimates for dict predictions per concept rely on training data and implicitly measure a degree of association between terms and descriptors. As a consequence, it belongs to the associative system architectures.

Fusion approaches combining lexical and associative characteristics have been realized by combining the predictions of $\text{BR}^{(\text{LR})}$ and *dict* as well as of $\text{BR}^{(\text{LR})}$ and $\text{BR}^{(\text{SVM})}$ with *maui* using the *union* strategy described in Section 4. These fusion systems are denoted $f_\circ : \text{BR}^{(\text{LR})}$ + dict, $f_\circ : \text{BR}^{(\text{LR})}$ + maui, and $f_\circ : \text{BR}^{(\text{SVM})}$ + maui with their short forms *BRLR+D*, *BRLR+M*, and *BRSVM+M*, respectively.

[7] https://github.com/HaraldKi/monqjfa
[8] https://github.com/zelandiya/maui

For *dict*, $f_\circ : \text{BR}^{(\text{LR})}$ + maui and $f_\circ : \text{BR}^{(\text{SVM})}$ + maui, we additionally applied the **transformation** described in Section 4 which led to systems in the following denoted by the suffix T or *transform*. Due to the runtime of the quickly realized implementation of transformation rule learning[9], transformations were only determined based on the dict method on the first fold and restricted to high-level categories of the thesaurus. Because the number of examples per category is expected to be high, we suspected that these rules are representative for all data sets and settings.

For the experiments we mostly used python and the scikit-learn library[10] which support $\text{BR}^{(\text{LR})}$ and $\text{BR}^{(\text{SVM})}$. For *Rhack*, we additionally used a script written for the R statistics package. Maui and *monq* were applied with Java.

5.4 Results

Table 3 lists the results for all data sets and approaches, supplemented by Figure 3 which focuses on sample-based averages and gives a visual impression of how systems perform.

Best values are marked bold in the table, showing that fusion approaches (arch.: F) that combine binary relevance approaches and maui were superior to lexical (arch.: L) and associative approaches (arch.: A) on all settings in terms of sample-based F_1 score and concept-based F_1 score. In almost all cases[11] this difference is statistically significant (paired t-test to the best performing algorithm), as indicated by arrows in the table. Across all settings, associative approaches (binary relevance methods and Rhack) achieved often significantly higher precision than other methods, however, they only predicted less than 3 descriptors per document on average. Recall of fusion systems outperformed associative as well as lexical approaches. These differences can also be recognized in Figure 3. Associative approaches (indicated in blueish colors) dominate the top, while fusion approaches (greenish) are further to the right with respect to the setting (symbol).

When training data and test data were selected to reflect explicit concept drift (experimental question ii), the associative systems were deteriorated considerably while maui was more stable. This is also reflected in Figure 3.

The implicit concept drift setting (experimental question iii) showed results that are similar to $\mathcal{D}_{\text{shuffle}}$, however, they seem to be more diverse, as can be seen in Figure 3. For instance, green circles are spread wider than green triangles.

Figure 4 illustrates a constrained evaluation which shows that 1) F_1 measure of the associative approach (A) vanished for the zero-shot tasks, 2) the fusion approaches combined predictions (A+L) for the better, and 3) modifications by transformation-rules lead to improvements under special circumstances, for instance, with regard to category G.01 (Europe) and the zero-shot setting (top right panel).

5.5 Discussion

Considering the questions i)-iii) posed in the beginning of Section 5, results showed the following:

[9] several hours on several thousand documents

[10] www.scikit-learn.org

[11] In some cases, the data was not shown to be normally distributed (Shapiro-Wilk test, $p < 0.05$), thus the assumptions for t-tests were not met.

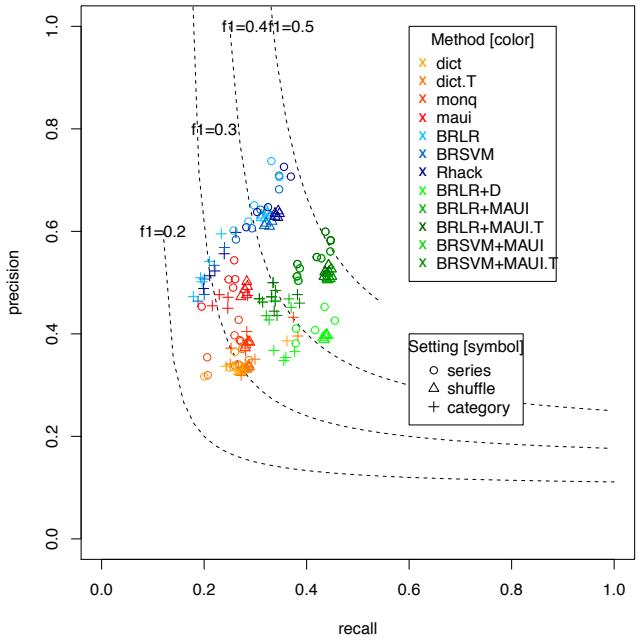


Figure 3: Sample-based average precision and recall. Symbols encode data sets ($\mathcal{D}_{\text{shuffle}}$, $\mathcal{D}_{\text{cat}}$, $\mathcal{D}_{\text{series}}$), colors encode approaches – Figure is best viewed in color.

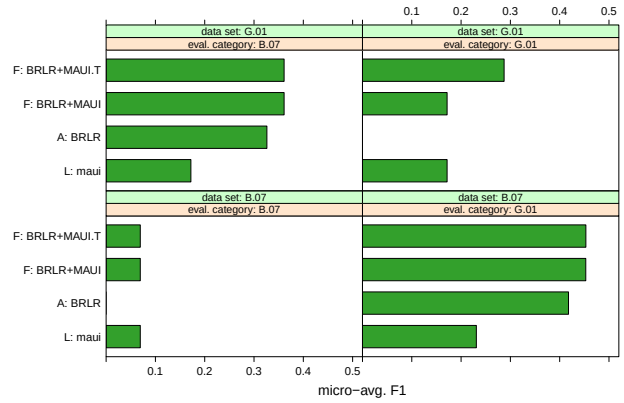


Figure 4: Constrained evaluation showing effects of explicit concept-drift and transformation rules. Results regard two categories and their corresponding concept-drift data sets.

The proposed descriptor-invariant fusion is i) superior to the associative and lexical systems in terms of F_1. Firstly, the union of individually proposed descriptors per document increased the overall recall. Hence, concepts proposed by the systems are at least partly non-overlapping. With the union strategy, the average number of assigned descriptors comes closer to how professional indexers act. Secondly, the union also retains high precision assignments, especially from the associative component.

Table 3: Comparison of approaches (averaged over 5 test sets). Architecture: L=lexical, A=associative, F=fusion. Bold type: highest values per setting and metric. Superscript $^{\downarrow}$: significantly smaller than maximum (bold) value (paired t-test, $p < .05$).

	Method		sample-based avg.			concept-based avg.			
Data	Name	Arch.	F_1	prec.	rec.	F_1	prec.	rec.	$\overline{\lvert\mathcal{Y}_{\text{pred}}\rvert}$
$\mathcal{D}_{\text{shuffle}}$	dict	L	0.277$^{\downarrow}$	0.329$^{\downarrow}$	0.273$^{\downarrow}$	0.222$^{\downarrow}$	0.451$^{\downarrow}$	0.265$^{\downarrow}$	4.92
$\mathcal{D}_{\text{shuffle}}$	dict.T	L	0.286$^{\downarrow}$	0.334$^{\downarrow}$	0.285$^{\downarrow}$	0.223$^{\downarrow}$	0.450$^{\downarrow}$	0.267$^{\downarrow}$	5.07
$\mathcal{D}_{\text{shuffle}}$	monq	L	0.307$^{\downarrow}$	0.381$^{\downarrow}$	0.285$^{\downarrow}$	0.245$^{\downarrow}$	0.475$^{\downarrow}$	0.285$^{\downarrow}$	4.41
$\mathcal{D}_{\text{shuffle}}$	maui	L	0.332$^{\downarrow}$	0.486$^{\downarrow}$	0.280$^{\downarrow}$	0.256$^{\downarrow}$	0.459$^{\downarrow}$	0.291$^{\downarrow}$	3.28
$\mathcal{D}_{\text{shuffle}}$	BR$^{(\text{LR})}$	A	0.391$^{\downarrow}$	0.632	0.318$^{\downarrow}$	0.206$^{\downarrow}$	**0.558**	0.181$^{\downarrow}$	2.69
$\mathcal{D}_{\text{shuffle}}$	BR$^{(\text{SVM})}$	A	0.394$^{\downarrow}$	0.617$^{\downarrow}$	0.326$^{\downarrow}$	0.208$^{\downarrow}$	0.510$^{\downarrow}$	0.187$^{\downarrow}$	2.90
$\mathcal{D}_{\text{shuffle}}$	Rhack	A	0.413$^{\downarrow}$	**0.633**	0.342$^{\downarrow}$	0.211$^{\downarrow}$	0.553$^{\downarrow}$	0.190$^{\downarrow}$	2.98
$\mathcal{D}_{\text{shuffle}}$	$f_\circ$:BR$^{(\text{LR})}$ + dict	F	0.392$^{\downarrow}$	0.395$^{\downarrow}$	0.436$^{\downarrow}$	0.279	0.426$^{\downarrow}$	0.351$^{\downarrow}$	6.55
$\mathcal{D}_{\text{shuffle}}$	$f_\circ$:BR$^{(\text{LR})}$ + maui	F	0.449$^{\downarrow}$	0.521$^{\downarrow}$	0.439$^{\downarrow}$	0.303	0.433$^{\downarrow}$	0.366$^{\downarrow}$	4.91
$\mathcal{D}_{\text{shuffle}}$	$f_\circ$:BR$^{(\text{LR})}$ + maui.T	F	**0.449**	0.521$^{\downarrow}$	0.439$^{\downarrow}$	**0.303**	0.433$^{\downarrow}$	0.367$^{\downarrow}$	4.91
$\mathcal{D}_{\text{shuffle}}$	$f_\circ$:BR$^{(\text{SVM})}$ + maui	F	0.449	0.512$^{\downarrow}$	0.444$^{\downarrow}$	0.300$^{\downarrow}$	0.417$^{\downarrow}$	0.369$^{\downarrow}$	5.08
$\mathcal{D}_{\text{shuffle}}$	$f_\circ$:BR$^{(\text{SVM})}$ + maui.T	F	0.449	0.512$^{\downarrow}$	**0.445**	0.300$^{\downarrow}$	0.416$^{\downarrow}$	**0.370**	5.09
$\mathcal{D}_{\text{cat}}$	dict	L	0.292	0.344	0.285$^{\downarrow}$	0.206$^{\downarrow}$	0.420	0.261$^{\downarrow}$	5.29
$\mathcal{D}_{\text{cat}}$	dict.T	L	0.300$^{\downarrow}$	0.349$^{\downarrow}$	0.298$^{\downarrow}$	0.208	0.418	0.263$^{\downarrow}$	5.47
$\mathcal{D}_{\text{cat}}$	monq	L	0.320$^{\downarrow}$	0.393$^{\downarrow}$	0.295$^{\downarrow}$	0.225$^{\downarrow}$	0.441	0.279$^{\downarrow}$	4.76
$\mathcal{D}_{\text{cat}}$	maui	L	0.300$^{\downarrow}$	0.466$^{\downarrow}$	0.245$^{\downarrow}$	0.233$^{\downarrow}$	0.436	0.279$^{\downarrow}$	3.26
$\mathcal{D}_{\text{cat}}$	BR$^{(\text{LR})}$	A	0.273$^{\downarrow}$	0.524$^{\downarrow}$	0.202$^{\downarrow}$	0.150$^{\downarrow}$	**0.467**	0.139$^{\downarrow}$	2.21
$\mathcal{D}_{\text{cat}}$	BR$^{(\text{SVM})}$	A	0.277$^{\downarrow}$	0.510$^{\downarrow}$	0.210$^{\downarrow}$	0.151$^{\downarrow}$	0.425$^{\downarrow}$	0.146$^{\downarrow}$	2.42
$\mathcal{D}_{\text{cat}}$	Rhack	A	0.298$^{\downarrow}$	**0.536**	0.226$^{\downarrow}$	0.159$^{\downarrow}$	0.465	0.154$^{\downarrow}$	2.50
$\mathcal{D}_{\text{cat}}$	$f_\circ$:BR$^{(\text{LR})}$ + dict	F	0.350$^{\downarrow}$	0.365$^{\downarrow}$	**0.374**	0.235$^{\downarrow}$	0.377$^{\downarrow}$	0.316$^{\downarrow}$	6.57
$\mathcal{D}_{\text{cat}}$	$f_\circ$:BR$^{(\text{LR})}$ + maui	F	0.366	0.469$^{\downarrow}$	0.332$^{\downarrow}$	0.253$^{\downarrow}$	0.388$^{\downarrow}$	0.326	4.56
$\mathcal{D}_{\text{cat}}$	$f_\circ$:BR$^{(\text{LR})}$ + maui.T	F	0.371	0.472$^{\downarrow}$	0.339$^{\downarrow}$	**0.253**	0.388$^{\downarrow}$	0.326	4.60
$\mathcal{D}_{\text{cat}}$	$f_\circ$:BR$^{(\text{SVM})}$ + maui	F	0.366	0.458$^{\downarrow}$	0.338$^{\downarrow}$	0.249$^{\downarrow}$	0.371$^{\downarrow}$	0.328	4.75
$\mathcal{D}_{\text{cat}}$	$f_\circ$:BR$^{(\text{SVM})}$ + maui.T	F	**0.371**	0.461$^{\downarrow}$	0.344$^{\downarrow}$	0.249$^{\downarrow}$	0.371$^{\downarrow}$	**0.328**	4.79
$\mathcal{D}_{\text{series}}$	dict	L	0.268$^{\downarrow}$	0.338$^{\downarrow}$	0.247	0.189	0.417$^{\downarrow}$	0.241$^{\downarrow}$	4.89
$\mathcal{D}_{\text{series}}$	dict.T	L	0.277$^{\downarrow}$	0.343$^{\downarrow}$	0.259$^{\downarrow}$	0.191	0.417$^{\downarrow}$	0.245$^{\downarrow}$	5.05
$\mathcal{D}_{\text{series}}$	monq	L	0.293	0.390$^{\downarrow}$	0.255	0.206$^{\downarrow}$	0.445$^{\downarrow}$	0.256$^{\downarrow}$	4.32
$\mathcal{D}_{\text{series}}$	maui	L	0.308	0.500$^{\downarrow}$	0.244	0.222$^{\downarrow}$	0.464$^{\downarrow}$	0.264$^{\downarrow}$	3.11
$\mathcal{D}_{\text{series}}$	BR$^{(\text{LR})}$	A	0.387$^{\downarrow}$	0.663	0.304$^{\downarrow}$	0.218$^{\downarrow}$	**0.639**	0.205$^{\downarrow}$	2.70
$\mathcal{D}_{\text{series}}$	BR$^{(\text{SVM})}$	A	0.389$^{\downarrow}$	0.645$^{\downarrow}$	0.312$^{\downarrow}$	0.224$^{\downarrow}$	0.598$^{\downarrow}$	0.217$^{\downarrow}$	2.87
$\mathcal{D}_{\text{series}}$	Rhack	A	0.409$^{\downarrow}$	**0.665**	0.327$^{\downarrow}$	0.229$^{\downarrow}$	0.628$^{\downarrow}$	0.223$^{\downarrow}$	2.99
$\mathcal{D}_{\text{series}}$	$f_\circ$:BR$^{(\text{LR})}$ + dict	F	0.394$^{\downarrow}$	0.416$^{\downarrow}$	0.413	0.259$^{\downarrow}$	0.435$^{\downarrow}$	0.344$^{\downarrow}$	6.54
$\mathcal{D}_{\text{series}}$	$f_\circ$:BR$^{(\text{LR})}$ + maui	F	0.448	0.556$^{\downarrow}$	0.413$^{\downarrow}$	0.284	0.467$^{\downarrow}$	0.354$^{\downarrow}$	4.79
$\mathcal{D}_{\text{series}}$	$f_\circ$:BR$^{(\text{LR})}$ + maui.T	F	**0.449**	0.556$^{\downarrow}$	0.414$^{\downarrow}$	0.284	0.467$^{\downarrow}$	0.355$^{\downarrow}$	4.80
$\mathcal{D}_{\text{series}}$	$f_\circ$:BR$^{(\text{SVM})}$ + maui	F	0.447	0.544$^{\downarrow}$	0.418$^{\downarrow}$	0.285	0.454$^{\downarrow}$	0.362$^{\downarrow}$	4.95
$\mathcal{D}_{\text{series}}$	$f_\circ$:BR$^{(\text{SVM})}$ + maui.T	F	0.447	0.544$^{\downarrow}$	**0.419**	**0.285**	0.454$^{\downarrow}$	**0.363**	4.96

With regard to question ii) and iii), fusion makes predictions more robust against concept drift as can be seen in Figure 3. It is backed up by Maui [16], which seems to be a robust choice for the lexical component of the system. Implicit and explicit concept drift were handled with lower variance by Maui ($F_1 \approx 0.3$ on $\mathcal{D}_{\text{shuffle}}$, $\mathcal{D}_{\text{cat}}$, $\mathcal{D}_{\text{series}}$) compared to associative systems. In particular, BR$^{(\text{LR})}$ and BR$^{(\text{SVM})}$ were considerably deteriorated by explicitly induced concept drift ($F_1 < 0.28$ on $\mathcal{D}_{\text{cat}}$, $F_1 > 0.39$ on $\mathcal{D}_{\text{shuffle}}$).

Among the different fusion configurations, it seems that $f_\circ$: BR$^{(\text{LR})}$ + maui and $f_\circ$:BR$^{(\text{SVM})}$ + maui are on par with each other. The effect of post-processing by transformation rules appeared to be small. Maybe the constraint to high-level categories was to strict.

Our experiments gave an impression on how the approaches may behave in practical settings when methods are applied to new domains. They are in line with our expectations from the analysis of system architectures. Similar to the results from Jimeno-Yepes et al. [12], we also found improvements by meta-learning according to specific concepts (Rhack). In our setting, it was, however, considerably affected by concept drift.

A direct comparison to figures reported on full-texts in the economics domain [11] (micro-avg. $F_1 = .39$, based on random order) is difficult because our results base on less training data (ours: $\approx$ 20k vs. theirs: > 60k) and short text (titles and author-keywords). In general, multiple factors influence the absolute performance including data set characteristics and the calculation of F_1 scores (i.e., the type of averaging). Finally, please note that even professional indexers, which we use as ground-truth, do not agree on all indexing terms. For instance, Medelyan and Witten [17] reported an inter-indexer agreement of 39%. Albeit their values are not directly comparable to our work because of differences in data sets, thesauri, and indexing rules, they might give a rough overall impression.

6 CONCLUSION

Analysis and experimental results of our work underline that special consideration of the system architecture is essential for the success of automatic subject indexing systems, especially in order to scale with the rapid growth of scientific publishing and dynamic subject areas. In this regard, we proposed descriptor-invariant fusion of associative and lexical approaches. Experiments in the economic domain on texts, shorter than abstracts, showed that our fusion approach is superior to state-of-the-art methods for lexically-based and associative indexing. Fusion improved F_1 scores, in particular even when concept drift was induced explicitly or implicitly. Beyond numbers, we emphasized the relevance of descriptor-invariant prediction for scalable automatic indexing. Our work supported the German National Library of Economics (ZBW) – Leibniz Information Centre for Economics to find suitable solutions for their practical setting and it may help researchers and practitioners in other digital libraries as well. In future work, we will further investigate the fusion of different architectures and extend features, classifiers, regularization and learning algorithms.

REFERENCES

[1] Alan R. Aronson, Dina Demner-Fushman, Susanne M. Humphrey, Jimmy J. Lin, Patrick Ruch, Miguel E. Ruiz, Lawrence H. Smith, Lorraine K. Tanabe, W. John Wilbur, and Hongfang Liu. 2005. Fusion of Knowledge-Intensive and Statistical Approaches for Retrieving and Annotating Textual Genomics Documents. In *Proceedings of the Fourteenth Text REtrieval Conference, TREC 2005, Gaithersburg, Maryland, USA, November 15-18, 2005*, Ellen M. Voorhees and Lori P. Buckland (Eds.), Vol. Special Publication 500-266. National Institute of Standards and Technology (NIST). http://trec.nist.gov/pubs/trec14/papers/nlm-umd.geo.pdf

[2] Lutz Bornmann and Rüdiger Mutz. 2015. Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references. *Journal of the Association for Information Science and Technology* 66, 11 (2015), 2215–2222. http://EconPapers.repec.org/RePEc:bla:jinfst:v:66:y:2015:i:11:p:2215-2222

[3] Leo Breiman. 1996. Bagging Predictors. *Machine Learning* 24, 2 (1996), 123–140. DOI: http://dx.doi.org/10.1007/BF00058655

[4] Eric Brill. 1995. Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging. *Computational Linguistics* 21, 4 (1995), 543–565.

[5] Caterina Caracciolo, Armando Stellato, Ahsan Morshed, Gudrun Johannsen, Sachit Rajbahndari, Yves Jaques, and Johannes Keizer. 2013. The AGROVOC Linked Dataset. *Semantic Web* 4, 3 (2013), 341–348.

[6] Nicolai Erbs, Iryna Gurevych, and Marc Rittberger. 2013. Bringing Order to Digital Libraries: From Keyphrase Extraction to Index Term Assignment. *D-Lib Magazine* 19, 9/10 (sep 2013). DOI: http://dx.doi.org/10.1045/september2013-erbs

[7] Reginald Ferber. 1997. Automated indexing with thesaurus descriptors: A co-occurrence based approach to multilingual retrieval. In *Research and Advanced Technology for Digital Libraries*. Springer Science + Business Media, 233–252. DOI: http://dx.doi.org/10.1007/bfb0026731

[8] Eibe Frank, Gordon W. Paynter, Ian H. Witten, Carl Gutwin, and Craig G. Nevill-Manning. 1999. Domain-Specific Keyphrase Extraction. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, IJCAI 99, Stockholm, Sweden, July 31 - August 6, 1999. 2 Volumes, 1450 pages*, Thomas Dean (Ed.). Morgan Kaufmann, 668–673. http://ijcai.org/Proceedings/99-2/Papers/002.pdf

[9] Manuela Gastmeyer, Max-Michael Wannags, and Joachim Neubert. 2016. Relaunch des Standard-Thesaurus Wirtschaft - Dynamik in der Wissensrepräsentation. *Inf. Wiss. & Praxis* 67, 4 (2016). DOI: http://dx.doi.org/10.1515/iwp-2016-0039

[10] Eva Gibaja and Sebastián Ventura. 2015. A Tutorial on Multilabel Learning. *ACM Comput. Surv.* 47, 3 (2015), 52:1–52:38. DOI: http://dx.doi.org/10.1145/2716262

[11] Gregor Große-Bölting, Chifumi Nishioka, and Ansgar Scherp. 2015. A Comparison of Different Strategies for Automated Semantic Document Annotation. In *Proceedings of the 8th International Conference on Knowledge Capture (K-CAP 2015)*. ACM, New York, NY, USA, Article 8, 8 pages. DOI: http://dx.doi.org/10.1145/2815833.2815838

[12] Antonio Jimeno-Yepes, James G. Mork, Dina Demner-Fushman, and Alan R. Aronson. 2012. A One-Size-Fits-All Indexing Method Does Not Exist: Automatic Selection Based on Meta-Learning. *JCSE* 6, 2 (2012), 151–160. DOI: http://dx.doi.org/10.5626/JCSE.2012.6.2.151

[13] Boris Lauser and Andreas Hotho. 2003. Automatic Multi-label Subject Indexing in a Multilingual Environment. In *Research and Advanced Technology for Digital Libraries, 7th European Conference, ECDL 2003, Trondheim, Norway, August 17-22, 2003, Proceedings (Lecture Notes in Computer Science)*, Traugott Koch and Ingeborg Sølvberg (Eds.), Vol. 2769. Springer, 140–151. DOI: http://dx.doi.org/10.1007/978-3-540-45175-4_14

[14] Eneldo Loza Mencía and Johannes Fürnkranz. 2010. Efficient Multilabel Classification Algorithms for Large-Scale Problems in the Legal Domain. In *Semantic Processing of Legal Texts – Where the Language of Law Meets the Law of Language* (1 ed.), Enrico Francesconi, Simonetta Montemagni, Wim Peters, and Daniela Tiscornia (Eds.). Lecture Notes in Artificial Intelligence, Vol. 6036. Springer-Verlag, 192–215. DOI: http://dx.doi.org/10.1007/978-3-642-12837-0_11 accompanying EUR-Lex dataset available at http://www.ke.tu-darmstadt.de/resources/eurlex.

[15] Christopher D. Manning. 2015. Computational Linguistics and Deep Learning. *Computational Linguistics* 41, 4 (Sept. 2015), 701–707. DOI: http://dx.doi.org/10.1162/COLI_a_00239

[16] Olena Medelyan, Eibe Frank, and Ian H. Witten. 2009. Human-competitive Tagging Using Automatic Keyphrase Extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3 - Volume 3 (EMNLP '09)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 1318–1327. http://dl.acm.org/citation.cfm?id=1699648.1699678

[17] Olena Medelyan and Ian H. Witten. 2008. Domain-independent automatic keyphrase indexing with small training sets. *Journal of the American Society for Information Science and Technology* 59, 7 (2008), 1026–1040. DOI: http://dx.doi.org/10.1002/asi.20790

[18] Jean-Baptiste Michel, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K. Gray, The Google Books Team, Joseph P. Pickett, Dale Holberg, Dan Clancy, Peter Norvig, Jon Orwant, Steven Pinker, Martin A. Nowak, and Erez Lieberman Aiden. 2010. Quantitative Analysis of Culture Using Millions of Digitized Books. *Science* (2010). http://www.sciencemag.org/content/331/6014/176.full

[19] Jinseok Nam, Eneldo Loza Mencía, Hyunwoo J. Kim, and Johannes Fürnkranz. 2015. Predicting Unseen Labels Using Label Hierarchies in Large-Scale Multi-label Learning. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2015, Porto, Portugal, September 7-11, 2015, Proceedings, Part I*. 102–118. DOI: http://dx.doi.org/10.1007/978-3-319-23528-8_7

[20] Mark Palatucci, Dean Pomerleau, Geoffrey Hinton, and Tom M. Mitchell. 2009. Zero-shot Learning with Semantic Output Codes. In *Proceedings of the 22Nd International Conference on Neural Information Processing Systems (NIPS'09)*. Curran Associates Inc., USA, 1410–1418. http://dl.acm.org/citation.cfm?id=2984093.2984252

[21] Bruno Pouliquen, Ralf Steinberger, and Camelia Ignat. 2003. Automatic annotation of multilingual text collections with a conceptual thesaurus. *Proceedings of the Workshop Ontologies and Information Extraction at the Summer School 'The Semantic Web and Language Technology - Its Potential and Practicalities' (EUROLAN'2003)* abs/cs/0609059 (2003). http://arxiv.org/abs/cs/0609059

[22] Prateek Veeranna Sappadla, Jinseok Nam, Eneldo Loza Mencía, and Johannes Fürnkranz. 2016. Using semantic similarity for multi-label zero-shot classification of text documents. (2016). http://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2016-174.pdf 24th European Symposium on Artificial Neural Networks.

[23] Fabrizio Sebastiani. 2002. Machine learning in automated text categorization. *Comput. Surveys* 34, 1 (2002), 1–47. citeseer.ist.psu.edu/sebastiani02machine.html

[24] Kai Ming Ting and Ian H. Witten. 1999. Issues in Stacked Generalization. *J. Artif. Intell. Res. (JAIR)* 10 (1999), 271–289. DOI: http://dx.doi.org/10.1613/jair.594

[25] W John Wilbur and Won Kim. 2014. Stochastic Gradient Descent and the Prediction of MeSH for PubMed Records. *AMIA ... Annual Symposium proceedings. AMIA Symposium* 2014 (2014), 1198–1207. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4419959/

[26] David H. Wolpert. 1992. Stacked generalization. *Neural Networks* 5, 2 (1992), 241–259. DOI: http://dx.doi.org/10.1016/S0893-6080(05)80023-1