HUMBOLDT-UNIVERSITÄT ZU BERLIN
INSTITUT FÜR BIBLIOTHEKS- UND INFORMATIONSWISSENSCHAFT

BERLINER HANDREICHUNGEN
ZUR BIBLIOTHEKS- UND
INFORMATIONSWISSENSCHAFT

HEFT 496

AUTOMATIC CLASSIFICATION OF THE BERLINER HANDREICHUNGEN ZUR BIBLIOTHEKS- UND INFORMATIONSWISSENSCHAFT

VON
JULIANE KÖHLER

Automatic Classification of the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft

von

Juliane Köhler

Berliner Handreichungen zur
Bibliotheks- und Informationswissenschaft

Begründet von Peter Zahn
Herausgegeben von
Vivien Petras
Humboldt-Universität zu Berlin

Heft 496

Köhler, Juliane

Automatic Classification of the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft/ von Juliane Köhler. – Berlin : Institut für Bibliotheks- und Informationswissenschaft der Humboldt-Universität zu Berlin, 2022. – 139 S. : 17 graph. Darst. – (Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft ; 496)

ISSN 14 38-76 62

Abstract:

Classification systems are one of the most established methods of knowledge organization with many advantages and yet, the collection of the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft (BHR) is missing a classification scheme. Therefore, an objective of the thesis at hand is to achieve a classification system for the collection and to potentially use Machine Learning (ML) methods for the automatic allocation of the BHR documents to the obtained classification system. The research questions that will be answered, are whether the JITA Classification System of Library and Information Science (JITA) is an appropriate classification system for the BHR and if automatic classification with ML can be applied to allocate the documents of the collection to a classification system without a using BHR data in the training dataset. To evaluate JITA an evaluation checklist was created based on recommendations of the cited literature. Using this checklist, it was concluded that JITA is not suitable as classification system of the BHR. Thus, using the same checklist as a reference, a new classification system was created. No expert evaluations nor user studies were conducted, which is a clear limitation of the thesis at hand. After a suitable classification scheme for the BHR was created, titles and abstracts of documents from different sources were scraped to use them as the training set for the ML experiments. Naïve Bayes, SVM, and Logistic Regression classifiers as well as Deep Learning classifiers, using the FLAIR framework, were tested. None of the obtained models yielded satisfying results, which is why no further experiments classifying the BHR documents were conducted. It was concluded that an automatic classification of the BHR documents is not possible without a BHR training set. Several limitations, especially during the creation of the training set, could have led to the unsatisfactory results which will be discussed in this thesis, which offers a basis for future studies that aim to evaluate classification schemes or for further Text

Diese Veröffentlichung geht zurück auf eine Masterarbeit im Studiengang Information Science, M. A. an der Humboldt-Universität zu Berlin.

Eine Online-Version ist auf dem edoc Publikationsserver der Humboldt-Universität zu Berlin verfügbar.

Acknowledgement

I would like to thank Ján M. Hanes for realizing my ideas in the form of programming code and for always supporting me.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AUC	Area Under the Curve
BERT	Bidirectional Encoder Representations from Transformers
BHR	Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft
CNN	Convolutional Neural Network
DABI	DABI - Datenbank Deutsches Bibliothekswesen
DC	(Automatic) Document Classification
DL	Deep Learning

e-LiS	e-LiS : e-prints in library and information science
FE	Feature Extraction
FS	Feature Selection
IBI	Berlin School of Library and Information Science
JITA	JITA Classification System of Library and Information Science
KBHR	Klassifikationssystem für die Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft; Classification System for the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft
KNN	k-Nearest Neighbor
LIS	Library and Information Science
ML	Machine Learning
MLC	Multi-Label Classification
NB	Naïve Bayes
NN	Artificial Neural Network
o-bib	o-bib. Das offene Bibliotheksjournal
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristic
Springer	SpringerLink
SVM	Support Vector Machine
TC	(Automatic) Text Classification
TFIDF	Term Frequency – Inverse Document Frequency

1 Introduction

Classification systems are one of the most established methods of knowledge organization (Lorenz, 2018; Manecke, 2004; Oberhauser, 2005; Pfeffer & Schöllhorn, 2018). With their hierarchical structure they map relationships between classes (Lorenz, 2018) and yield benefits that the commonly used search bar does not provide (Hall et al., 2014): Classification systems can be of aid to users who are unable to formulate their information need, searching for a complex topic, or trying to obtain an overview of the according collection of entities. Furthermore, they make it easy for the user to narrow or broaden their search and to gain a context of the classes (Matveyeva, 2002). In addition, they offer solutions to problems like multilingualism and ambiguity (Oberhauser, 2005). Overall, the hierarchical structure is simply intuitive for users (Lawrie et al., 2001). Therefore, Manecke (2004) claims that the organization that classification systems offer, satisfies a basic need of human beings.

In regard to text documents, the automatic classification is often preferred to manual classification since the 1990s in Germany (Oberhauser, 2005). This is because automatic document classification (DC) saves time and manpower (Labrador et al., 2020; Sharma et al., 2018). Especially Deep Learning (DL) – a subset of Machine Learning (ML) – algorithms, like Feed-Forward Neural Networks, Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) show excellent results for automatic classification (Akhter et al., 2020; Behera et al., 2019; Zheng & Zheng, 2019). However, if the dataset is small, traditional ML algorithms like Naive Bayes (e.g., Caruana & Niculescu-Mizil, 2006; Dwivedi & Arya, 2016; Ting et al., 2011), Support Vector Machines (SVMs; e.g., Caruana & Niculescu-Mizil, 2006; Dwivedi & Arya, 2016; Spirovski et al., 2018), Random Forest (e.g., Kowsari et al., 2019; Spirovski et al., 2018), Logistic Regression (e.g., Caruana & Niculescu-Mizil, 2006; Kowsari et al., 2019), and k-Nearest-Neighbor (kNN; e.g., Akhter et al., 2020; Kowsari et al., 2019) are also commonly applied.

The ‘Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft’ (Berliner Handouts to Library and Information Science; BHR); a collection of research papers, theses, lectures etc. from members of the Berlin School of Library and Information Science (IBI); is missing a classification system. Since the collection is curated by the IBI of the Humboldt-Universität zu Berlin, that among others teaches the optimization of information processes and systems (Humboldt-Universität zu Berlin, 2018), a classification system for the BHR

should be provided. The collection of the BHR is small. Therefore, a classification scheme is the optimal organizational system for it, according to Gaus (2005). An objective of this thesis is to achieve a classification system for the BHR and to potentially use ML methods for the automatic allocation of the BHR documents to the obtained classification system. A known classification system for Library and Information Science (LIS) topics is the 'JITA Classification System of Library and Information Science' (JITA). JITA is used as classification system for the e-LiS : e-prints in library and information science (e-LiS) database[1]. To save time and resources, the initial aim is to discover if JITA can be reused as classification scheme for the BHR. If JITA is suitable for reuse, the open access documents in e-LiS can be used as training set for the experiments with ML algorithms. If it is not suitable, documents that are accessible online must be extracted, since no pre-labeled BHR dataset is available. Thus, this paper aims to answer the following two research questions:

I. Is the ‚JITA Classification System of Library and Information Science' an appropriate classification system for the BHR?
II. Can automatic classification via ML be applied to allocate the BHR to a classification system without a BHR training dataset?

If a satisfactory classification system for the BHR can be found, it can be used on the official website of the BHR collection[2] and the open access server of the Humboldt-Universität zu Berlin[3]. Moreover, the thesis can serve as an example to other institutions that aim to evaluate a classification system.

The thesis is structured as follows: Chapter 2 explains different terms that stand in relation to knowledge organization. Chapter 3 gives an overview of the most relevant research about the evaluation of classification systems and insights about automatic text classification (TC) using ML or DL. In Chapter 4, a description of the BHR collection and JITA is given as well as a discussion of suitable ML algorithms for experiments in this thesis. Chapter 5 explains how the evaluation of JITA, and the ML experiments were conducted. The according results will be reported in Chapter 6 and discussed in Chapter 7. Finally, the insights of the conducted research are summarized in Chapter 8.

[1] http://eprints.rclis.org/ [September 25, 2021]
[2] http://www.ib.hu-berlin.de/inf/handrei.htm [September 25, 2021]
[3] https://edoc.hu-berlin.de/ [September 25, 2021]

2 Essential Definitions for Organizing Knowledge

Within the research field of organizing knowledge, there are multiple terms and notions that seem to share the same meaning and yet are slightly different. In the following, unambiguous terms are defined to build a base for the following chapters.

The most central term of this research is 'classification'. To define classification, it is necessary to firstly know what a *class* is: According to Dahlberg (1974) a class is a set of elements that are grouped together based on one common feature. This definition will be followed in this thesis with the additional clarification of Gaus (2005), who states, that classes are the different domains that derive from the segmentation of the subject area that is ought to be documented.

Subsequently, Jacob (2004) defines *classification* as a system, a group or class, or a process. Manecke (2004) has a similar definition. He distinguishes classification as three different concepts: the process of creating classes, the resulting classification system, and the process of assigning objects to the classes in a classification system. In the study at hand, the focus is on the latter definition of Manecke (2004) and therefore classification is defined as assigning entities to predefined classes according to their features (Jacob, 2004; Kadhim, 2019; Manecke, 2004; Sharma et al., 2018). The classification system itself, usually a hierarchy of ordered non-overlapping classes, is then defined as *classification system* or *classification scheme* (Gantert, 2016; Gaus, 2005; Jacob, 2004). One of its advantages are the relationships between the classes that give more context about a topic to the user (Gaus, 2005; Jacob, 2004; Lorenz, 2018). Generally, classification systems are divided into *monohierachies* and *polyhierarchies*. In monohierarchies, every term only has one broader term, while terms can have two or more broader terms in polyhierarchies (Gaus, 2005; Manecke, 2004).

The word classification is often used interchangeably with *categorization*. However, Jacob (2004) strongly declares herself in favor of a distinction of those two terms. Similar to classification, she defines categorization as "the process of dividing the world into groups of entities whose members are in some way similar to each other" (Jacob, 2004, p. 518). However, while classifications are more rigorous and stable in their definition of classes and relationships to each other, categorizations are more flexible. Nonetheless, other authors used categorization (almost) as a synonym for classification (Golub et al., 2016; Lewis et

al., 2004; Weigend et al., 1999) or as a verb in a sense of dividing elements into groups (Koller & Sahami, 1997). Thus, in this thesis, categorization will also be seen as a synonym for classification.

Another term that appears often in the knowledge organization literature is *taxonomy*. Traditionally, taxonomy is a synonym for a classification system as well. Nowadays, it means the science behind classification processes and systems and other organizational structures (Bedford, 2013; Jacob, 2004). Following the example of the Council on Library and Information Resources (CLIR, 2017), Fraunhofer ISST and]init[(2009), and Bedford (2013), taxonomy will be used as a synonym for a classification system just like categorization.

Ontology is another organizational structure. The definition of ontologies can be really broad and therefore is sometimes hard to distinguish from other knowledge organization systems (Gómez-Pérez, 1996; Stuart, 2016). Stuart (2016, p. 12) defines an ontology as "a formal representation of knowledge with rich semantic relationships between terms". The key aspect that distinguishes ontologies from classification systems is the wording of rich semantic relationships. An ontology provides more information about the data than classification schemes do that only store the data in hierarchal list with classes (van Rees, 2003). Madsen and Thomsen (2009) further distinguish the purpose of those two systems: While an ontology is a model that aims to represent simplified knowledge concepts, classification aims to bring structure and order into the data.

A *thesaurus*, following the definition of Stuart (2016), provides different relationships between terms including hierarchical, equivalence and associative relationships. According to van Rees (2003), a thesaurus offers additional information for classification schemes, taxonomies and ontologies and mainly deals with words and their relations to each other. This structured list of words helps an indexer to describe documents by following the cross-references between the terms (Broughton, 2006).

Finally, *knowledge organization* is the broader term for classification, categorization, taxonomy, and ontology (Stuart, 2016). CLIR (2017) acknowledges that all those concepts have subtle differences, but in their core, they all divide elements into groups with a overlying topic and hence are often used interchangeably. Thus, in the next chapter, if authors

evaluate one of those knowledge organizations, it will also be seen as potentially valuable for the evaluation of classification systems.

3 Literature Review

3.1 Evaluation of Classification Systems

A classification system is a hierarchically organization of documents, which are set in relation, structured and ordered in a way that simplifies the search and the knowledge acquisition of the user (Lorenz, 2018; Matveyeva, 2002). These are important functions and hence a high quality of classification schemes is vital. Yet, to the best of the author's knowledge, there is only a limited number of approaches, respectively documentation of such approaches, on evaluating an existing classification system. One of the few scientific contributions putting the focus solely on this topic is a presentation held by Kwaśnik (2021). She mentions ten useful criteria that can be employed for the purpose of the evaluation of a classification scheme.

However, only the minority of scientific publications puts its focus on the evaluation of classification systems only, as Kwaśnik (2021) did, but center different topics, such as the (automatic) creation of hierarchical structures or metadata (Fernando et al., 2012; Hall et al., 2014; Lawrie et al., 2001; Stoica et al., 2007), similarity measurements (Maedche & Staab, 2002) or (technologies for) knowledge sharing (Gómez-Pérez, 1996; Gruber, 1995). Moreover, even though well-designed classification schemes are as important as ever, the literature sources are often several years if not decades old. Yet, even though more up-to-date literature would be desirable, the research results are still valuable. In the following, important insights from the literature about the quality measurement of classification schemes will be summarized:

Hall et al. (2014) list four main approaches including the following references: The gold-standard (Maedche & Staab, 2002), criteria-based evaluation (Gómez-Pérez, 1996), expert evaluation (Stoica et al., 2007) and statistically evaluation (Lawrie et al., 2001). Those approaches will be discussed later in this chapter. Hall et al. (2014) themselves call the methods historic. Furthermore, they criticize that expert knowledge is needed to use these approaches and claim that classification systems are viewed independently from the problem they are built to solve. Therefore, they propose their own user-centered approach by extending their former suggestion (Fernando et al., 2012): Hall et al. (2014) evaluate the classification according to its cohesion, the parent-child-relationships, whether it provides an overview and its context. In other words, they test if items in one class are similar, but

different to the items in other classes; if users recognize and can name the kind of parent-child-relationship in the classification system; if users get an overview over the collection of items and if the items are well-placed in the hierarchy.

Stoica et al. (2007), that were referenced by Hall et al. (2014), evaluate hierarchies with the help of people as well. Yet, instead of end-users, they recruited experts: 34 information architects were asked to compare the outputs of different hierarchy-creating algorithms and to evaluate certain aspects of the hierarchies. They were given questions such as if they would add more categories or if the categories are meaningful. In contrast, in Maedche and Staab's study (2002) only one expert created a gold standard, that was used to evaluate the quality of ontologies that were created by people without expert knowledge. A gold standard is a set of correctly classified documents or a classification system created by an expert (Golub et al., 2016; Maedche & Staab, 2002). Golub et al. (2016) also address the topic of the gold standard and evaluate the term indexing quality. According to them, term indexing can be seen as a broader term for classification and thus, their work also contributes to the research about the evaluation of classification systems. They propose a framework for evaluating the indexing quality in three different contexts based on a literature review: evaluation through either a professional evaluator or a comparison with a gold standard, evaluation in the context of indexing workflow, and evaluation through retrieval performance. They also elaborate problems in the gold standard approach and finally recommend a mix of an expert and a user evaluation. Not only this recommendation, but all of their proposals require user or expert groups.

The remaining two approaches mentioned by Hall et al. (2014) do not rely on user or expert evaluation: Lawrie et al. (2001) use statistical means like the expected mutual information measure or ANOVA for evaluation. Gómez-Pérez (1996) on the other hand, created a framework for the evaluation of ontologies. She divides the quality check into three levels: The verification of the ontologies, the software, and the documentation. The former is the most interesting for this research and will be explained further: To verify the architecture of an ontology, Gómez-Pérez refers to the design criteria of Gruber (1993). Also listed in Gruber (1995), the five design criteria are clarity, coherence, extendibility, minimal encoding bias and minimal ontological commitment. Clarity is defined as objective and clear definition of terms. Coherence means that conclusions based on the definitions of the ontology cannot interfere with conclusions of other definitions. Extendibility says that it

should be possible to extend the ontology by new terms without interfering with existing definitions. Minimal encoding bias implies that choices should not be made because of encoding (notation or implementation) conveniences. Lastly, minimal ontological commitment means that the ontology should be built with the least assumptions about the domain as possible to give users the freedom to adapt the ontology. For the verification of lexis and syntax, Gómez-Pérez (1996) proposes to use a parser (syntax) and a scanner (lexis). To examine the quality of definitions and axioms, she defines multiple functions and constraints. With the help of those, a verification of a hierarchy and its classes is possible.

Another approach is suggested very early on by Ranganathan (1937). The researcher lists 13 canons, which he explains with examples, in order to compare and evaluate classification schemes. Ranganathan (1967) revised, reordered and added more information to the list of canons. A central term in his definitions is ‘characteristics’. Characteristics are, according to Ranganathan (1937, 1967), one or more distinct attributes of entities that makes them comparable or distinguishable. Based on that definition, multiple canons for the evaluation of a classification system are drawn from both sources, Ranganathan (1937) and Ranganathan (1967):

1) Canons for Characteristics:
    a. Differentiation: The characteristic element of a class should be a differentiating attribute to other classes and entities.
    b. Relevance: All the characteristics are ought to be relevant regarding the classification purpose.
    c. Ascertainability: Characteristics should be definite and clearly determinable.
    d. Permanence: Characteristics should be definable and fixed unless the purpose of the classification changes.
2) Canons for Successions of Characteristics:
    a. Concomitance: Two characteristics of classes should not overlap in a way that they give rise to the same subjects.
    b. Relevant Sequence/Succession: The characteristics should be ordered in a logical manner that is relevant to the classification purpose.

c. Consistency: The chosen characteristics and their sequence of importance should be followed and not be changed unless the purpose of the classification changes.

3) Canons for Arrays:
    a. Exhaustiveness: The subordinate classes of a main class should be exhaustive in terms of the universe of the classification system.
    b. Exclusiveness: The subordinate classes of a main class should be exclusive in a way that an entity can only be allocated to no more than one class.
    c. Helpful Order: The order of subordinate classes of a main class should not be random but follow principles that serve the purpose of the intended users of the classification system.
    d. Consistent Order: If comparable subordinate classes appear in different main classes, their order should be the same or comparable in all the main classes.
4) Canons for Chains:
    a. Intension: In a sequence of classes the classes get narrower. That means the last link should have the most characteristics and the least allocated entities.
    b. Modulation: There should be no gaps in a sequence of classes.

Many authors do not focus on the evaluation of an existing classification system, but on how to build a new one. Yet, doing that, they mention principles on how a good classification system should look like, which can be used for an evaluation as well: Umlauf (1999) lists ten requirements for classification of openly displayed media in a library. It is possible to use most of the requirements for digital libraries too.

In his book, Gaus (2005) explains information systems in great detail and hence also classification schemes. Included are features of classification systems as well as formulas to calculate the number of required classes or the desirable average occupancy of classes through entities. Manecke (2004) explains logical rules for hierarchies and three features a classification systems should exhibit. According to him, classification systems need to be universal, continuous, and up to date. He also mentions several other qualities a classification system should have, that can be used as evaluation factors*:* I. e., the classes must be disjoint, the structure of the classification system should be consistent, and the use over a longer

period of time should be possible. Dahlberg (1974), one of the former leading experts for classification in Germany (Ohly, 2020), also lists and explains five features an universal classification should have: objective term fixation [*Sachadäquatheit*], use of system principals [*Formadäquatheit*], universality, flexibility, compatibility and computerization [*Computerrisierbarkeit*].

Fraunhofer ISST and]init[(2009) collected several guidelines and good practices for taxonomies. They emphasize that reusing an existing taxonomy should always be considered, but also state that a reused taxonomy usually requires modifications. They also reference five purposes of a taxonomy respectively controlled vocabulary listed by the National Information Standards Organization (NISO, 2005). Those five purposes are:

1) Provide a vocabulary and the means to create such a vocabulary for indexing and retrieval
2) Consistency in the sense of using the same format and rule for the assignment of terms
3) "Indicate semantic relationships among terms" (NISO, 2005, p. 11)
4) "Provide consistent and clear hierarchies […]" (NISO, 2005, p. 11)
5) Help users during their searching process

Derived from those purposes they give recommendations on terms and their relationships. Furthermore, they give advice about multilingual taxonomies referring to the work of the International Federation of Library Associations and Institutions (IFLA, 2009). Their explanations are, among others, useful for the development of a multilingual classification system.

Hjørland (2013) stresses the importance of subject knowledge of the domains that are covered by the classification system. This specific knowledge is needed to build and evaluate a system. According to him, this cannot be avoided by conducting user studies, trying to use common sense, or using other methods.

In the following chapter the assumption is that a high-quality classification system already exists. The problem that must be solved is the automatic allocation of documents to that classification system. Different approaches of ML will be discussed.

3.2 Automatic Document Classification using Machine Learning

In the literature, TC, that is part of the field of Natural Language Processing, is used in different contexts (Sebastiani, 2002). One application of TC and the focus of this thesis is DC (Akhter et al., 2020). The definition of what a document is, is not unimportant (Buckland, 1997, Reprint/1998). Thus, to avoid confusion, in this thesis a document is referred to as a text that consists of more than one sentence. There are also other applications of TC than DC, such as sentiment analysis or sentence classification. However, these are not subject of the thesis at hand and will not be further discussed. Furthermore, in the remainder of this study, TC and DC will be used interchangeably, because definitions that apply to DC also apply to TC.

DC is defined as the assignment of documents to predefined classes or categories (Akhter et al., 2020; Golub et al., 2016; Lewis et al., 2004; Spirovski et al., 2018). As a mathematical definition, it "is the task of assigning a Boolean value to each pair $\langle d_j, c_i \rangle \in D \times C$, where D is a domain of documents and $C = \{c_1, \dots, c_{|C|}\}$ is a set of predefined categories" (Sebastiani, 2002, p. 3) using an unknown target function $\breve{\Phi}$: $D \times C \{T, F\}$ (Spirovski et al., 2018). A special subcategory of this classification task is the problem of multi-label classification. While in single-label (or 'multi-class') classification each document is assigned exactly one label (one class), multi-label classification is defined as the task of assigning one or more labels to a document (Guibin Chen et al., 2017; Pintas et al., 2021; Sebastiani, 2002). The latter is more challenging since the combination of possible classes grows exponentially. Yet, multi-label classification can be translated into a simplified version by reducing the problem to several classification tasks – one for each class.

TC is not a new concept and is already conducted employing ML since the 1990s (Oberhauser, 2005; Pong et al., 2008; Sebastiani, 2002). ML and especially DL seem to be the most popular method in the recent literature. DL is a subfield of ML and includes all ML algorithms that are inspired by the human brain, i.e., all kinds of Artificial Neural Networks (NNs) with more than one layer of neurons (Denuit et al., 2019; Forsyth, 2019). Insights of both research areas, ML and DL, will be introduced in the following paragraphs.

One of the standard references for TC using ML is Sebastiani (2002). He defines the many meanings of TC, explains its uses, and introduces the basic main steps of the classification pipeline as well as the general functionality of different types of classifiers. A classifier is a

ML algorithm that analyzes a text and its features and produces a class label for them (Forsyth, 2019; Shah et al., 2020). There are many algorithms for classifying documents. Many are addressed and especially compared in the literature. Most here listed papers are referring to English text classification, since – to the best of the author's knowledge and also according to Kass (2019) – there is only limited research being done using German or other non-English datasets.

3.2.1 The Dataset

A pre-labeled dataset is required for DC using ML (supervised learning). Mostly such datasets already existed, before different classifiers were trained and tested on them (e.g., Kamath et al., 2018; Lai et al., 2015; Miao et al., 2018) and some were manually created by experts that labeled the documents in question (Banerjee et al., 2019).

Most studies compare different traditional classifiers with each other or with a newly introduced classifier (Akhter et al., 2020; Banerjee et al., 2019; Kamath et al., 2018; Kass, 2019; Miao et al., 2018; Pong et al., 2008; Romanov et al., 2019; Shah et al., 2020; Spirovski et al., 2018; Ting et al., 2011). Not only the classification results of traditional ML classifiers are compared to other traditional ML classifiers (Miao et al., 2018; Pong et al., 2008; Shah et al., 2020), but also the results of traditional ML algorithms with DL algorithms (Akhter et al., 2020; Banerjee et al., 2019; Kamath et al., 2018; Kass, 2019; Romanov et al., 2019; Spirovski et al., 2018; Ting et al., 2011) or DL algorithms with DL algorithms (Banerjee et al., 2019; Lai et al., 2015). The results often vary, because there is no algorithm that works equally well on all datasets, according to Caruana and Niculescu-Mizil (2006) and Dwivedi and Arya (2016). The datasets for DC studies can vary immensely. For example, Pong et al. (2008) applied KNN and Naïve Bayes to classify library documents, when others used emails (Uysal & Gunal, 2014), scientific texts (Lai et al., 2015; Romanov et al., 2019), medical free-text reports (Banerjee et al., 2019), news articles (Akhter et al., 2020; Miao et al., 2018; Shah et al., 2020; Spirovski et al., 2018) or other kind of text (Lai et al., 2015) as datasets. Moreover, the input can differ. Thus, Galke et al. (2017) evaluated if the title alone is enough for multi-label classification or if full-texts are required. Even though the best results were still achieved with the full text, using only the title also was possible with a high quality.

Most datasets are in English (Banerjee et al., 2019; Galke et al., 2017; Lai et al., 2015; Pong et al., 2008; Shah et al., 2020; Ting et al., 2011), but some are also in non-English languages (Akhter et al., 2020; Lai et al., 2015; Miao et al., 2018; Romanov et al., 2019; Spirovski et al., 2018) or bi-lingual (Uysal & Gunal, 2014). Literature on the automatic classification of German text documents or related studies are sparse. Reiner (2010) writes in her article in 2010 that the tested classifiers are not useable for German professional purposes yet. Most German results stem from university publications in the form of reports based on a thesis (Kass, 2019) or actual theses (Cabrera Granados, 2014; Scherer, 2003) only.

3.2.2 Preprocessing & Dimensionality Reduction

Not only the dataset has an influence on the performance of a classifier. In the study of Uysal and Gunal (2014) the subject was not the classification itself, but the influence of different preprocessing methods. Their overall result confirms that there is no general combination of processing tasks that always exhibits the best results for every domain and language. Therefore, they suggest testing different processing variations for every study. Ting et al. (2011) propose the same. One of the purposes of using preprocessing techniques, such as stop-word removal, stemming, lemmatization, spelling correction and others, is dimensionality reduction. Dimensionality reduction reduces the chances of overfitting (Sebastiani, 2002) and results in the reduction of time complexity, computational resources, and memory consumption through shrinking the feature space (Kowsari et al., 2019; L. Liu & Liang, 2011). Features are used for pattern recognition and can be any extracted measurable numeric or symbolic characteristics of the data (Schalkoff, 2007). For TC such properties can be characters, words, phrases, or even entire documents represented as a numeric value in a vector (called feature vector). The process of creating feature vectors is called vectorization. For vectorization, tokenization is necessary. Tokenization chops a text into useful semantic elements called tokens (Kowsari et al., 2019; Manning et al., 2008; Uysal & Gunal, 2014). Tokens are the representation of the properties that are turned into numeric values in the vectorization step and are sometimes also referred to as terms. Even though terms and tokens are slightly different as explained in Manning et al. (2008), they will be seen as equivalent in this thesis. The feature space then is yielded by available feature vectors. Two widely used methods in the TC pipeline aim to reduce the dimensionality of the feature space: Feature Selection (FS) and Feature Extraction (FE, also called Feature Transformation or Feature Projection).

Two definitions of FE can be found in the literature: The first one states that the purpose of this step is to obtain features from the raw text by transforming it into a vector (Kass, 2019; Kowsari et al., 2019; Spirovski et al., 2018; Uysal & Gunal, 2014) and thus is equal to vectorization. Two methods can be distinguished (Kowsari et al., 2019): weighted word techniques (do not capture the relationship between terms) and embeddings (capture relationship). The other definition of FE says that a smaller set of features is created out of the original feature set in a way that it preserves the relevant information (Aggarwal & Zhai, 2014; Khalid et al., 2014; Pintas et al., 2021; Sebastiani, 2002). Typical FE techniques are Term Frequency, Term Frequency – Inverse Document Frequency (TFIDF), fastText, GloVe, or Word2Vec (Kass, 2019; Kowsari et al., 2019; Miao et al., 2018).

FS on the other hand is not about creating new features or a new feature set. Instead it reduces the size of the feature space by creating a subset of relevant features that work best for predictions from the original set (Aggarwal & Zhai, 2014; Khalid et al., 2014; Manning et al., 2008; Pintas et al., 2021; Ting et al., 2011). It not only aims to reduce the dimensionality of the data, but also to boost the classification accuracy through reducing the noise (Manning et al., 2008). Filters, wrapper, and embedded methods are three different types of FS (Khalid et al., 2014; Sebastiani, 2002; Steinwendner & Schwaiger, 2020). Popular techniques include principal component analysis, mutual information, information gain, gini index or chi-square (Aggarwal & Zhai, 2014; Gayathri & Marimuthu, 2013; Khalid et al., 2014; Lai et al., 2015; Sebastiani, 2002; Uysal & Gunal, 2014; Yang & Pedersen, 1997). Kass (2019), Lai et al. (2015) and Aggarwal and Zhai (2014) also count preprocessing steps such as lowercasing, stop-word removal, and stemming as FS.

3.2.3 Traditional Machine Learning Classifiers

Finally, the choice of a classifier is an important decision as well. As implied above, there is no classifier that works well on all datasets and the classification result further differs depending on the chosen preprocessing methods. However, all classifiers also contribute to the classification result with different advantages and disadvantages that makes the choice not an arbitrary one.

Naïve Bayes, for instance, which is based on the probability theory of the Bayes theorem (Kamath et al., 2018; Pong et al., 2008), is simple to understand as well as to build (Akhter et al., 2020; Miao et al., 2018; Rennie et al., 2003), fast (Kass, 2019; Miao et al., 2018;

Rennie et al., 2003), inexpensive (Kowsari et al., 2019), and works for multi-label classification. Because of these advantages, it used to be really popular (Kowsari et al., 2019). Naïve Bayes is a linear classifier (Manning et al., 2008) that is built to calculate the probability that a given document belongs to a given class. It is called naïve, because it assumes that all features are statistically independent from each other (Pong et al., 2008; Rennie et al., 2003). Even though this assumption is rarely true in realistic data, it works well on many applications (Miao et al., 2018) and for several TC tasks (Ting et al., 2011). However, another problem occurs due to its simplicity: Naïve Bayes does not correctly classify documents of classes with only a few instances in an unbalanced dataset (Pong et al., 2008; Rennie et al., 2003). Yet, Rennie et al. (2003) claim to have found a solution to this and other problems of the Naïve Bayes classifier that achieves an accuracy like a SVM.

SVMs might be one of the most popular traditional ML classifiers for TC and most studies in the literature use them to some extent (Banerjee et al., 2019; Kamath et al., 2018; Kass, 2019; Miao et al., 2018; Romanov et al., 2019; Spirovski et al., 2018; Ting et al., 2011). They are popular, because they are often the most accurate classifiers (Aggarwal & Zhai, 2014; Gayathri & Marimuthu, 2013; Sebastiani, 2002) and work well on text data (Joachims, 1998). They function on linear and non-linear data and very effective in a high-dimensional space (Akhter et al., 2020; Gayathri & Marimuthu, 2013). SVMs aim to find a hyper-plane, that can be understood as a separating plane, with maximum margin (aggregated distances of the closest datapoint to the hyper-plane of each class) that separates the instances of two classes (Dwivedi & Arya, 2016; Sebastiani, 2002; Spirovski et al., 2018). In its original form, it is an binary classifier (Dwivedi & Arya, 2016; Kamath et al., 2018). The output of the SVMs are normalized distances to the hyperplane (Caruana & Niculescu-Mizil, 2006). If the data is non-linear, so-called kernel functions are used to map the data into a higher-dimensional space to make them linear separable (Kowsari et al., 2019; Manning et al., 2008). There are different kernels and choosing the right one is important and difficult (Dwivedi & Arya, 2016). Hsu et al. (2016) recommend to use a linear kernel if the feature set is large, which is the case for TC problems. The drawback of SVMs is that they require a lot of training time (Dwivedi & Arya, 2016; Gayathri & Marimuthu, 2013; Spirovski et al., 2018; Ting et al., 2011) which is why it is recommended to use them only on small datasets (Kass, 2019; Miao et al., 2018).

An alternative to SVMs is Logistic Regression, because it is more computational efficient (Pawar & Gawande, 2012). Logistic Regression is easy to implement: It uses a sigmoid curve to predict the probability that a document belongs to one of two classes (Kamath et al., 2018; Kowsari et al., 2019; Shah et al., 2020). Hence, Logistic Regression is a binary classifier as well, but can be extended to Multinomial Logistic Regression for multiclass classification problems (Kowsari et al., 2019). It is also a linear classifier and thus it follows that it cannot solve non-linear problems. Furthermore, it requires uncorrelated independent variables.

A non-linear classifier is the k-Nearest Neighbors (KNN) algorithm. The instance-based algorithm is easy to understand and implement: For the basic KNN, an yet unclassified datapoint is assigned to the class that most of its k nearest neighbors are labeled with by comparing their predefined distance-metrics (Akhter et al., 2020; Pong et al., 2008). An extension of this basic algorithm is to also consider similarity measures. Simplicity is the major advantage of KNN and it also works for multi-class tasks (Gayathri & Marimuthu, 2013; Kowsari et al., 2019). However, it is computationally expensive for big datasets, because the vector of a new document has to be compared to all vectors of the training set (Akhter et al., 2020; Gayathri & Marimuthu, 2013; Pong et al., 2008). Furthermore, finding the optimal k and distance-metric for a dataset is challenging (Kowsari et al., 2019).

The final last traditional ML classifiers that are often mentioned in the literature and are to be named in this review are Decision Trees and their extension Random Forest. A Decision Tree is built in the training phase by creating hierarchical rules based on features of the data (Akhter et al., 2020; Dwivedi & Arya, 2016; Kowsari et al., 2019). Branches represent feature values and leaves represent classes. Documents are allocated by following the rules from the root to a leaf and accepting the class represented by that leaf. Both, training, and prediction are fast (Aggarwal & Zhai, 2014) and Decision Trees are suitable for binary and multi-class problems. However, features at a higher level of a tree are given more importance than features at a lower level (Aggarwal, 2014). This causes problems for TC since a single feature alone usually holds little information about the correct class. Furthermore, due to imbalanced trees and rare occurrences of terms in a text, the classification results are poor. Aggarwal (2014) states that it is possible to improve the classification effectiveness through multivariate splits, but at a computational cost. Nonetheless, Decision Trees are very sensitive to noise in the data as well as often overfitted to the training data (Kowsari et al., 2019). That is why Random Forests have advantages over Decision Trees (Kamath et al.,

2018). A Random Forest consists of several Decision Trees that are randomly created using n random features from the data (Kamath et al., 2018; Spirovski et al., 2018). The classification result is a calculated combination of the outputs of each Decision Tree in the forest (Spirovski et al., 2018). In this way, Random Forests are more robust to overfitting than a single Decision Tree. It is still fast to train and only influenced by two factors: the number of trees and the number of features considered for a split (Kowsari et al., 2019; Spirovski et al., 2018). Yet disadvantages remain: Overfitting can still be an issue, the prediction time increases with the number of trees in the forest, and Random Forests are hard to interpret (Kowsari et al., 2019).

In general it is said that traditional ML algorithms are less expensive, easy to understand, do not require a big dataset, but more feature engineering (FS & FE) efforts (Akhter et al., 2020; Pintas et al., 2021). DL algorithms on the other hand, yield state-of-the-art results, require less feature engineering efforts, but are more expensive in matters of training time and computational requirements and it is difficult to interpret their results. Nevertheless, they often work better than traditional ML methods on complex problems (Géron, 2018).

3.2.4 Deep Learning Classifiers

The foundation of DL methods is a NN. Their functionality is based on biological neurons and neural pathways (Denuit et al., 2019; Spirovski et al., 2018). For a basic NN, there are three main layers of neurons: input layer, hidden layer and output layer (Aggarwal, 2014). A neuron is any element in the network which holds some kind of input (Skansi, 2018). The input layer simply stores the input and delivers it forward to the other layers (Aggarwal, 2014; Skansi, 2018). Accordingly, the input layer has as many neurons as there are input features (Spirovski et al., 2018). The output layer does the predictions which are determined by the output neuron with the highest value (or a softmax function with probabilities) and hence has as many neurons as there are labels for the classification problem (Aggarwal, 2014; Géron, 2018; Kowsari et al., 2019). In between those two main layers is the hidden layer, where most of the computation is done (Aggarwal, 2014; Aggarwal & Zhai, 2014). Within the hidden layer there can be again one or several layers of neurons. If there is more than one of such a layer then the network is called Deep Neural Network, otherwise it is called Shallow Network (Denuit et al., 2019; Géron, 2018). All neurons are connected by channels that are assigned weights (Géron, 2018). They influence what numerical value the subsequent neuron receives as an input. This input is called the weighted sum (Géron, 2018;

Kamath et al., 2018). Additionally, neurons can have a bias that is added to this input. A so-called activation function is used on the final value, that determines if the neuron transmits its data to the next layer or not. Through the calculation of an error and backpropagation, the weights of the neurons are adjusted to reduce the error (gradient descent), and the model gets trained. One of the simplest versions of a NN is a (single layer) perceptron (Aggarwal, 2014). It is an algorithm used for binary classification problems and just uses one layer including inputs, weights, the weighted sum, and the activation function.

More popular variations of NNs are Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs). In the former, the neurons have circular connections to themselves (Denuit et al., 2019). RNNs can handle sequential data and capture long-term dependencies well (P. Liu et al., 2016; Zheng & Zheng, 2019). However, for long sequences, to avoid the problem of exploding or vanishing gradients, variants of a RNN such as a LSTM (Hochreiter & Schmidhuber, 1997) or GRU (Cho, van Merrienboer, Bahdanau, & Bengio, 2014; Cho, van Merrienboer, Gulcehre, et al., 2014) are required. RNNs are popular for NLP tasks, because of their ability to process semantic information of text of variable length (Behera et al., 2019; Lai et al., 2015; P. Liu et al., 2016).

Even though CNN are often used for image and audio processing (Albawi et al., 2017; Steinwendner & Schwaiger, 2020), they can also be applied in Natural Language Processing (Akhter et al., 2020; Banerjee et al., 2019; Kim, 2014; Lai et al., 2015; Pham et al., 2016). They are especially useful for pattern detection. The hidden layer in a CNN includes the following sublayers: a convolutional layer that consist of several filters (also called kernels), an activation layer and a pooling layer (Steinwendner & Schwaiger, 2020). Those sublayers usually are stacked several times on top of each other and finally followed by a normal feed-forward NN that calculates the final prediction (Géron, 2018). The basic idea of the convolutional layer is a sliding (convoluting) window over a document matrix (Kamath et al., 2018; Steinwendner & Schwaiger, 2020). The matrix consists of feature vectors and the window detects different patterns depending on the respective filters it applies. Multiple filters can and should be applied. The result of each filter is called a feature map. Through those, CNNs capture the best text representation with the most influential features (Zheng & Zheng, 2019). The typical activation function of the activation layer of a CNN is the rectified linear unit (ReLU) function that can do calculations fast (Steinwendner & Schwaiger, 2020). The purpose of the pooling layer is to reduce the size of an output from one layer to another,

usually through maxpooling, and thus is a form of dimensionality reduction (Albawi et al., 2017; Kowsari et al., 2019; Steinwendner & Schwaiger, 2020). Zheng and Zheng (2019) state that a CNN trains faster than a RNN, but cannot capture features in a long sequence as well.

The most recent developments in DL are transformers. A transformer, introduced by Google employees in 2017 (Vaswani et al., 2017), is a deep learning model that reduces training time in comparison to the former state-of-the-art RNNs. It uses sequences of data, like RNNs do, but does not require a strict order as before.

3.2.5 Evaluation Methods

Even though, the accuracy measurement of the classifiers was the most popular evaluation method in the literature (Kamath et al., 2018; Kass, 2019; Romanov et al., 2019; Shah et al., 2020; Spirovski et al., 2018; Ting et al., 2011), it is risky to rely on accuracy as evaluation measurement only. It tells the percentage of correctly classified documents in comparison to all documents in the dataset (Dwivedi & Arya, 2016; Shah et al., 2020). That means that if the data is really unequally distributed such as that 95 % of the data are in one class only, the classifier will have an accuracy of 95 % if it classifies every instance to this one class (Kass, 2019). Therefore, there are other evaluation metrics:

Precision tells how many classified documents were allocated correctly to a class out of all documents that were allocated to that class by the classifier, or in other words, how often was the classifier correct when predicting the class for the documents (Dwivedi & Arya, 2016; Joorabchi & Mahdi, 2011; Miao et al., 2018). Recall (also true positive rate or sensitivity) returns how many documents were allocated correctly to a class out of all documents with this class label. As with accuracy, precision and recall should not be used as the only evaluation metrics (Pong et al., 2008). The maximum recall of a class can be achieved by assigning all documents in the dataset to that class, but the Precision would suffer. Furthermore, precision and recall are often contradictory (Miao et al., 2018). Hence, a metric that combines those two can be applied: The F1 score calculates the weighted harmonic mean of precision and recall (Manning et al., 2008; Shah et al., 2020). The score is only high if the values of precision and recall are also high (Géron, 2018). Yet, it also rewards equal values of precision and recall, which

Table i: The structure of a confusion matrix

		Reality	
		In class	*Not in class*
Prediction	*In class*	True Positives	False Positives
	Not in class	False Negatives	True Negatives

is problematic if one of them is more important than the other for a classification problem. However, it is more robust to class imbalance than accuracy which is why it is popular in the TC field (Forman & Scholz, 2010). The introduced evaluation metrics are binary measurements (Guibin Chen et al., 2017) and are described through the following formulars (Sebastiani, 2002):

$$Accuracy = \frac{True\ Positives + True\ Negatives}{True\ Positives + True\ Negatives + False\ Positives + False\ Negatives}$$

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall}$$

The formular values are based on the so-called confusion matrix. A confusion matrix is a matrix that includes the number of documents for each combination of the predicted class and the actual class as depicted in Table i (Kass, 2019; Kowsari et al., 2019). Three other binary measurements that are applied for evaluation are illustrated as a graph: The receiver operating characteristic (ROC) curve visualizes the recall against the false positive rate (L. Liu & Liang, 2011). The false positive rate is the proportion of falsely allocated documents to a class of all documents that do not belong to that class. It is calculated as follows (Géron, 2018):

$$False\ Positive\ Rate = 1 - Recall$$

or

$$False\ Positive\ Rate = \frac{False\ Positives}{False\ Positives + True\ Negatives}$$

The area under the curve (AUC) is as meaningful as the ROC curve itself. It can be used to compare different classifiers with each other by comparing its size (Géron, 2018; L. Liu & Liang, 2011). Similarly to the ROC curve, the precision-recall curve contrasts recall and precision (Géron, 2018). The precision-recall curve should be preferred over the ROC curve if positive datapoints are rare or if false positives are more important than false negatives (Géron, 2018). The AUC of the precision-recall curve can also give insights about the skill of a classifier as the ROC AUC does.

To apply evaluation on multiclass problems, sometimes metrics as macro and micro average measurements are calculated (Behera et al., 2019; Guibin Chen et al., 2017). "Macro average[d] refers to the average performance (Precision, Recall and F1 score) over labels, while [micro average] counts all true positives, true negatives, false positives and false negatives first among all labels and then has a binary evaluation for its overall counts" (Guibin Chen et al., 2017, p. 2381).

4 Preliminary Examinations

4.1 Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft

The BHR are a collection of publications, lectures, theses etc. of members from the IBI. The collection is curated by the very same school.

The BHR document files were scraped from the edoc-server (the open access server of the Humboldt-Universität zu Berlin) on March 20, 2021 at 11:11 a.m. In addition, the metadata of the documents was scraped from their individual edoc-server websites four minutes later as well. The scrapers can be found in Appendix III. The extracted information for the description is the first author, the title, the language, the publication type, and the subjects of the publications. Later the abstract was added for classification predictions.

The scraper crawled 419 documents and their corresponding metadata. However, three BHR (129, 189 and 362) appeared multiple times in the data. The duplicates were deleted, and 416 entries remained. Additionally, small mistakes in the original data were noticed, such as misspellings (BHR 130) or wrong order (BHR 239) of the author's first and last name. Since they did not influence the analysis, they were not corrected.

At the time of scraping, there were 466 BHR listed on the BHR official website. The difference between the number of scraped documents and BHR represented on the website

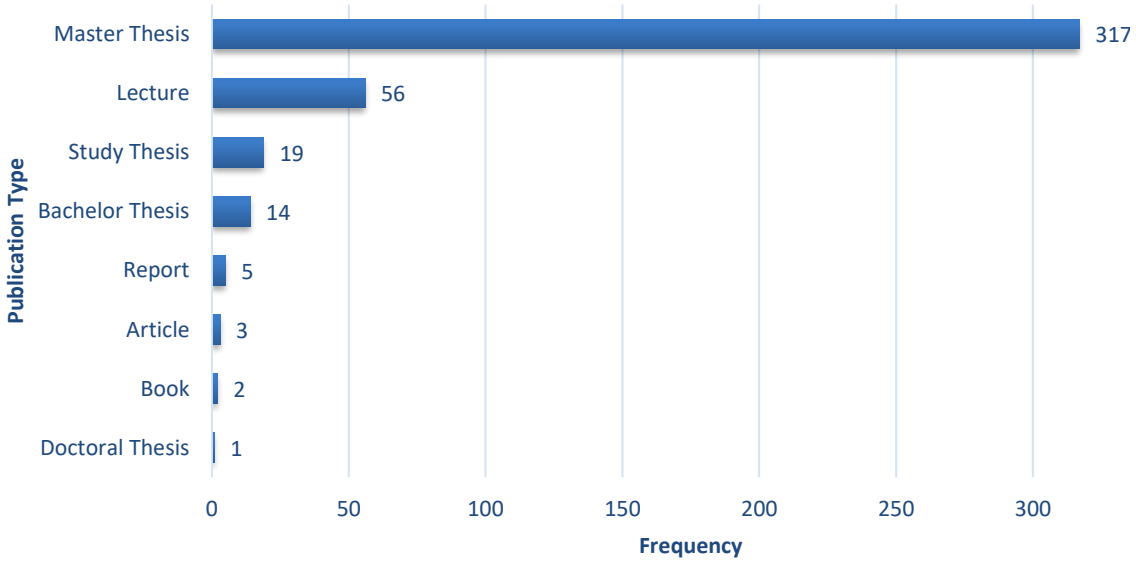


Figure 1: Distribution of the types of publication of the scraped BHR documents

is due to missing digital copies on the edoc-server and missing data within the BHR: BHR 148, 201 and 371 are not assigned any publication. Thus, the BHR collection consists of 463 BHR only. After checking the missing 47 BHR, it was confirmed that those documents cannot be found on the edoc-server (anymore) and thus no metadata could be used for the analysis, except for one BHR to which metadata was added to the data. Following, the final corpus of 417 documents represents 90.1 %[4] of the BHR at that time. The corpus together with its analysis can be found in Appendix I.

The description level of the documents varied a lot. While one document was described by 37 subject keywords (BHR 330), others had only one (e.g., BHR 34). 95.2 % of the scraped BHR are written in German, 4.8 % in English. No other language was represented. More than three quarters (76.0 %) of the analyzed BHR are master theses. The second-biggest group with 13.4 % are lectures, followed by study theses (4,6 %). Other publication types and their absolute frequencies are depicted in Figure 1.

After cleaning the data concerning the subjects of the BHR (for further information see codebook in the sheet 'Subject Cleaning' in the file 'BHR_corpus_and_analysis.xlsx' in Appendix I), that were mostly written in German, 1,091 individual subjects remained. The cleaning process was not completely flawless: Subjects were united that represented the same concepts, e.g., the German and the English version of a word (e.g., 'Classification' and 'Klassifikation') or an abbreviation of a concept and the written-out version (e.g., 'IBI' and 'Institut für Bibliotheks- und Informationswissenschaft'). The frequency of the unified subject term then was calculated by summing up the frequencies of the original terms. It was not reviewed if two terms were used to describe only one or several BHR. Thus, it is possible that the frequencies do not exclusively represent the number of BHR that were denoted with a certain subject. However, these cases should be in the minority.

The most frequent subject is *Library and Information Science* (in the data 'Bibliotheks- and Informationswissenschaft') with 428 occurrences. Here the above-mentioned limitation becomes obvious, because the code occurred more often than there are BHR in the corpus. Since this subject was a clear outlier to the rest of the subjects, it was excluded for clarity from Figure 2, which shows the other 24 most frequent subjects, translated into English and

[4] The numbers are shown rounded up to one digit after the decimal symbol

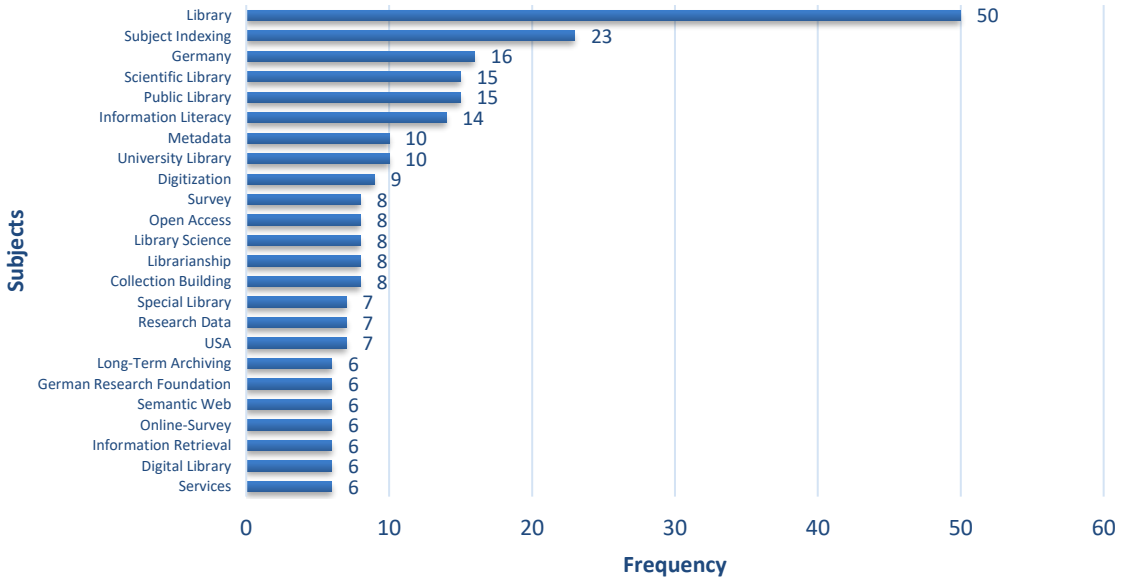


Figure 2: The 24 most frequently used subjects translated into English (‘Library and Information Science' excluded; frequency > 5)

with a frequency higher than 5. As can be drawn from Figure 2, a predominating theme in the BHR are different types of libraries and library science topics. This theme became even more apparent after freely coding the subjects, to make it possible to assign them to groups. The coding process resulted in 28 subject groups that are listed in Table ii ordered by the number of included subjects per group. Once more, the subject *Library and Information Science* was excluded of this analysis for clarity.

There were a lot of subjects that were either too unique (e.g., January 2006; ‘Januar 2006’) or too broad (e.g., quantity; ‘Anzahl’) to allocate them to any of the groups. Therefore, they were put into a group called *Others* (Table ii). Especially in this group it is obvious, that subject do not necessarily perfectly represent the topic of a BHR, but also describe the context of a BHR publication. In the following only the English translations of subjects and not the original subjects will be used as examples.

The groups *Library Science & Librarianship* (e.g., RDA) and *Libraries & Information Facilities* (e.g., Public Library) illustrate, that topics concerning any kind of library, librarianship activities or the science around libraries, are numerous – in both, accumulated frequencies, and number of subjects within the two groups – within the collection. The group *Education* covered almost exclusively subjects with a frequency of one. In this group, subjects illustrate different degrees (e.g., Bachelor), educational institutions (e.g., School),

study forms (e.g., postgraduate distance learning), resources etc. concerning the topic of education (e.g., Webinar). *Other Disciplines & Related Fields* includes subjects that represent other fields outside of or related to the LIS scope (e.g., Electrochemistry).

Management, Marketing & Financing (e.g., Fundraising) could be seen as part of the latter group. However, this thematical complex appeared so often in the data, that an own group was build.

Another very numerous subject group was *Publication Types & Genres* which includes subjects of the content that the group name suggests (e.g., Lyric). Other examples of groups with contents equivalent to their category name are *Services and Software* (e.g., Discovery Service, Social Intranet), *Locations* (e.g., Nepal), *Institutions (e.g.,* Robert Koch Institute), *History* (e.g., History 1945-2007), *Languages* (e.g., Russian), *Information Systems* (e.g., Current Research Information System), *Archival Science & Museology* (e.g., long-term archiving), *Publishing* (e.g., Open Peer Review) and *Knowledge Organization Systems* (e.g., Thesaurus).

The group *Methodology* contains subjects regarding scientific methods or tools (e.g., Logfile Analysis). *Information Retrieval* (e.g., Retrieval Test), *Information Behavior* (e.g., Information Need), *Information Management* (e.g., Research Data Management) and *Information Processing & Analytics* (e.g., Automatic Identification) include subjects that are connected to the research fields of the according chairs of IBI. *Foundations and cross-disciplinary topics of Information Science* unites all subjects that apply to several fields of Information Science (e.g., Information Literacy) or are seen as a basic element of it (e.g., information). *Internet & Technology* is a group of subjects directly related to the internet (e.g., Semantic Web) or technical problems (e.g., chatbots). *Scientific Practices* groups all subjects together that do not specifically belong to *Methodology* but describe scientific concepts and data (e.g., empirical studies). *Laws & Justice* not only includes law topics (e.g., youth protection), but also licenses (e.g., creative commons). *People* contains subjects that serves as a portrayal of actual real-life people (e.g., Herwarth Walden), while *Actors* represents job roles (e.g., songwriter) and specific groups of people (e.g., refugees). Finally, the group *User* includes all the subjects which only address users and no other group (e.g., user orientation).

Table ii: Freely coded groups of the subjects of the BHR ordered by the number of included subjects per group

Free Codes	Accumulated frequency of all subjects in this group	Number of included subjects in this group
Other	212	183
Library Science & Librarianship	229	138
Libraries & Information Facilities	170	64
Education	55	54
Other Disciplines & Related Fields	66	53
Management, Marketing & Financing	66	51
Methodology	78	50
Publication Types & Genres	66	50
Information Retrieval	81	48
Services and Software	61	47
Locations	76	46
Foundations and cross-disciplinary topics of LIS	72	38
Internet & Technology	62	36
Scientific Practices	37	29
Institutions	38	28
Publishing	31	21
Knowledge Organization Systems	33	19
Information Processing & Analytics	31	19
Information Behavior	29	17
History	27	17
Information Management	26	16
Laws & Justice	18	14
Languages	13	11
People	11	11
Actors	10	8
User	9	8
Information Systems	8	8
Archival Science & Museology	12	6

Even though the number of Library Science related subjects is high, other Information Science topics, such as ‘Information Retrieval’ (n = 6) or ‘Long-Term Archiving’ (n = 6), also occurred in the data. Since the BHR collect documents from both the Library, and the Information Science field, every BHR that is not related to Library Science must be connected to Information Science. Therefore, it is necessary to include both fields in one

classification system. In the next chapter, JITA will be introduced and described to assess whether it is suitable for reuse for the BHR.

4.2 JITA Classification System of Library and Information Science

e-LiS : e-prints in library and information science (e-LiS) is the largest digital repository for open access documents from the LIS field (e-LiS, n. d.–a). Since the repository is curated by a team of volunteer editors from 35 countries, it is only natural, that works can be uploaded in all languages (e-LiS, n. d.–b). The international archive is using the 'JITA classification system of Library and Information Science'. It is a two-level-mono-hierarchical classification system. The notation consists of one to two capital Latin characters in alphabetically order, which divides JITA into twelve main classes with in total 141 subclasses. The main classes and their corresponding subclasses are based on three main areas (Dal Porto & Marchitelli, 2006; De Robbio & Subirats Coll, 2014; Dupriez, 2013; Shaheen, 2013):

- Theoretical and General (general level; main classes A-B): "theoretical and general aspects of libraries and information; information use and the sociology of information." (De Robbio & Subirats Coll, 2014, pp. 15–16)
- User-oriented, directional, and management functionalities (intermediate level; main classes C-G): "Socio-economic and legal issues are included here. This divides into: users, literacy and reading; libraries and information repositories; publishing and legal issues; management; industry, profession and education." (De Robbio & Subirats Coll, 2014, p. 16)
- Objects, Pragmatics and Technicalities (specific level; main classes H-L): "[…] information sources, supports and channels; information treatment for information services; technical services in libraries, archives and museums; housing technologies; information technology and library technology." (De Robbio & Subirats Coll, 2014, p. 16)

On the date of the analysis of the data set, the March 20, 2021, at 3:39 p.m., the JITA classification system contained 22,748 documents. The documents are either allocated to one or more of the 141 subordinate classes or to one or more of the twelve main classes or to both. The main classes are:

A) Theoretical and general aspects of libraries and information (2,809 documents)

B) Information use and sociology of information (6,730 documents)

C) Users, literacy and reading (2,602 documents)

D) Libraries as physical collections (3,820 documents)

E) Publishing and legal issues (2,203 documents)

F) Management (2,076 documents)

G) Industry, profession and education (2,669 documents)

H) Information sources, supports, channels (5,982 documents)

I) Information treatment for information services (3,746 documents)

J) Technical services in libraries, archives, museums (1,932 documents)

K) Housing technologies (280 documents)

L) Information technology and library technology (4,327 documents).

As illustrated in Figure 3, the documents in JITA are unevenly distributed among the main classes. Especially, main class K 'Housing technologies' includes very little documents in comparison to the other main classes. The excel file that was used for the analysis of the contents of JITA can be found in Appendix I.

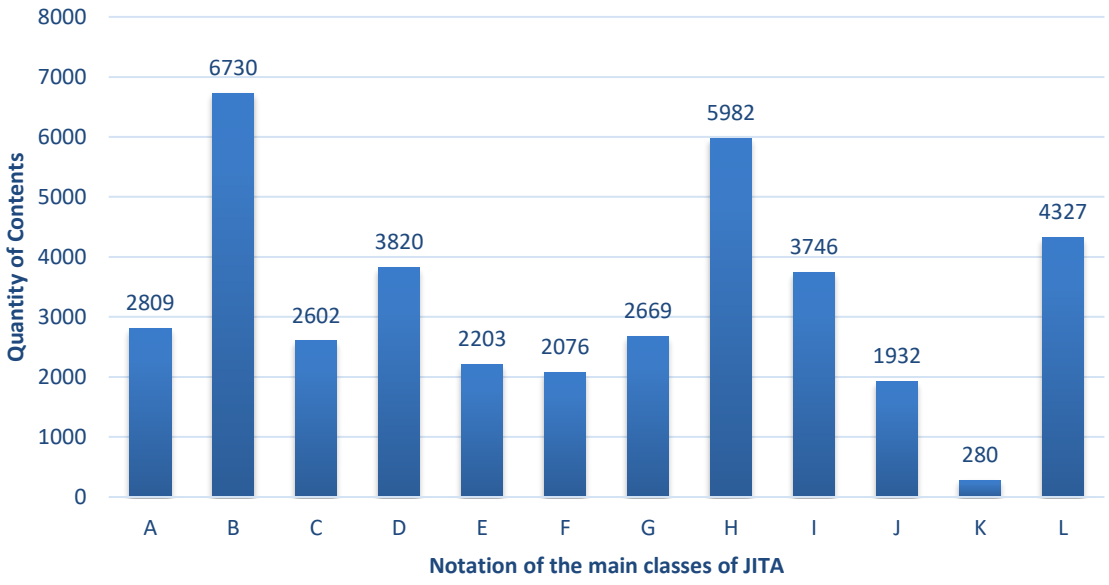


Figure 3: Quantity of contents of the main categories of JITA

According to De Robbio and Subirats Coll (2014), JITA was not created to be a comprehensive classification scheme. The classes are broad and do not go into detail. Thus, JITA's simple task is to facilitate the search for documents through a browsing mode (Dal Porto & Marchitelli, 2006). A suitable classification scheme for the BHR should not go into too much detail either, since only a few classes are required (Chapter 5.1). To save costs and for better quality, the reuse of JITA should be considered (Fraunhofer ISST &]init[, 2009). Thus, JITA needs to be evaluated whether it is suitable as classification scheme for the BHR collection. In Chapter 5.1, the tools and methods to do so will be discussed.

4.3 Discussion of suitable Machine Learning Algorithms

After a final classification system is found, the goal is to examine if an automatic allocation of BHR documents to this classification system without a pre-existing BHR training set is possible. For the experiments, a set of ML classifiers is required. In this chapter, it will be discussed which ML algorithms are the most suitable for this purpose and can be applied within the scope of this master thesis.

As expounded in Chapter 3.2, there is no universal combination of preprocessing methods, dimensionality reduction methods and classifiers for all datasets. Hence, different options must be tested, but due to time and resource constraints it is not possible to test more than four classifiers in this thesis. As the literature suggest (Kowsari et al., 2019; P. Liu et al., 2016) DL algorithms work well on big datasets. The BHR corpus is relatively small. Therefore, traditional ML algorithms seem to be the better alternative. Nonetheless, DL often yielded better results than ML algorithms (Kowsari et al., 2019; Zheng & Zheng, 2019). For this reason, it was decided to also use one DL and three ML algorithms for the automatic classification of the BHR.

To choose appropriate algorithms, it is important to know attributes of the training data and the specific classification goal first: The dataset, that will be used for training, is unbalanced and will be available in two versions: English and German (see Chapter 5.2.1). The documents consist of a title and an abstract that is between 150 and 5,000 characters long. The classification problem that needs to be solved is a multi-label problem with a defined maximum of three labels. Therefore, a probability output for each class is preferred. Based on this knowledge, the suitability of different classifiers for the problem can be discussed.

To build the models from scratch would go beyond the scope of this thesis. Instead, public frameworks and libraries were ought to be used. A very popular open-source framework for NLP tasks via DL is FLAIR[5] (Akbik et al., 2019). FLAIR claims to be simple to use and is curated by members of the Humboldt-Universität zu Berlin. Hence, it seemed to be a good choice for a DL classifier. FLAIR uses a RNN for the DC. A RNN provides, as described in Chapter 3.2.4, a good architecture for TC, since it capture the semantics of the text as well as dependencies between terms (Behera et al., 2019; Lai et al., 2015; P. Liu et al., 2016; Zheng & Zheng, 2019). One of FLAIR's advantages is a broad range of language models. To make use of one of these powerful features, Flair will be applied on the German dataset. For the traditional ML algorithms, the English dataset will be used since most studies are only conducted on English text and thus the classifiers are expected to work better on English data.

SVMs are popular classifiers that showed very good performance in the past (Gayathri & Marimuthu, 2013; Kowsari et al., 2019; Pawar & Gawande, 2012; Sebastiani, 2002). Forsyth (2019) states that SVMs are the go-to classifier. Behera et al. (2019) even calls them vital for automatic TC. This comes at a cost of a long running-time and non-interpretability of the data (Dwivedi & Arya, 2016; Gayathri & Marimuthu, 2013; Kowsari et al., 2019; Spirovski et al., 2018; Ting et al., 2011). Since it is more important to have the BHR documents accurately classified than to have transparency for the results, this drawback is negligible. However, the long running-time is an issue in terms of time resources. Spirovski et al. (2018) even call it unacceptable. Therefore, Kass (2019) and Miao et al. (2018) recommend on using it on small datasets only. Even though the training dataset for the BHR is not small, it was decided to use the SVM classifier nonetheless, since the training time is not expected to be longer than the training time of the DL model. SVMs are binary classifiers (Dwivedi & Arya, 2016; Kamath et al., 2018). This opposes the goal of assigning between one and three labels to each document. Therefore, the python library scikit-learn[6] will not only be applied for constructing a linear SVM classifier (LinearSVC[7]), but also to use the model for multi-label classification. The library provides a MultiOutputClassifier[8] strategy that constructs one classifier for each label automatically. The LinearSVC is a faster version

[5] https://github.com/flairNLP/flair [September 25.2021]
[6] https://scikit-learn.org/stable/ [September 25, 2021]
[7] https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html [September 25, 2021]
[8] https://scikit-learn.org/stable/modules/generated/sklearn.multioutput.MultiOutputClassifier.html [September 25, 2021]

of scikit-learn's SVC classifier with a linear kernel and is therefore preferred in this study. A linear kernel is chosen, because it is recommended for a high number of features, which is usually the case for text classification (Hsu et al., 2003/2016). Another disadvantage is that SVMs do not give out probabilities without making changes to the model (Spirovski et al., 2018). Hence, in case of sufficient test results, appropriate changes will have to be made to classify the BHR. Those can be time expensive.

Some studies suggest Logistic Regression as an alternative to SVMs, because the classifier yields similar results, but requires less training time (Forsyth, 2019; Kass, 2019; Pawar & Gawande, 2012). Therefore, it will be employed for the experiments of this thesis. Logistic Regression is a binary classifier as well, but unlike SVMs the classifier naturally gives out probabilities as an output through the use of a Sigmoid function (Forsyth, 2019; Kamath et al., 2018; Y. Y. Liu et al., 2011; Shah et al., 2020). Via the library scikit-learn, probabilities can easily be accessed as output if desired. It assumes that the data is linear separable (Kowsari et al., 2019). For the implementation the LogisticRegression module of scikit-learn[9] is used.

Finally, Naïve Bayes was chosen as a last traditional classifier for the experiments. In Ting et al. (2011) Naïve Bayes even yields slightly better results than the SVM classifier and is also less expensive. Even though many report of non-satisfying results of Naïve Bayes (Kass, 2019; Pong et al., 2008), it is still a popular classifier because it is fast, does not require a lot of memory and easy to implement also for large and high dimensional datasets (Akhter et al., 2020; Forsyth, 2019; Kowsari et al., 2019; Miao et al., 2018; Rennie et al., 2003; Ting et al., 2011). Therefore, it will be used as a baseline classifier. To avoid the bias for an unbalanced dataset a Complemented Naïve Bayes as suggested by Rennie et al. (2003) is applied via the module ComplementNB of scikit-learn[10].

Other traditional ML classifiers are not employed due to time and resource constraints of this thesis, but also due to the characteristics of the individual algorithms: KNN is computationally expensive for larger datasets (Akhter et al., 2020; Gayathri & Marimuthu, 2013) and usually does not yield better results than DL methods or SVMs (Miao et al., 2018;

[9] https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html [September 25, 2021]
[10] https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.ComplementNB.html [September 25, 2021]

Shah et al., 2020). Decision Trees are easily overfit and are not popular for TC (Aggarwal, 2014; Kowsari et al., 2019) while Random Forests require increased time for the prediction step (Kowsari et al., 2019).

Each classifier has hyperparameters that must be selected as well. The detailed descriptions of the procedures of the experiments will therefore be elaborated in the next chapter.

5 Methodology

5.1 Evaluation of JITA & Creation of a Classification System

To save time it was considered to reuse the JITA as classification system for the BHR. To do so, the suitability of JITA for this purpose had to be examined and JITA needed to be evaluated.

In Chapter 3.1 research results from the literature concerning the evaluation of classification systems or comparable knowledge organization system were explored. Since user studies in any form would go beyond the scope of this thesis, this evaluation method could not be used. Hjørland (2013) demands subject knowledge to tackle the task of evaluation. The author of this thesis studies Information Science at IBI for more than five years, took part in the scientific exchange (Köhler, 2020b) and also published her bachelor thesis in the scope of the BHR (Köhler, 2020a). Thus, her subject knowledge is sufficient to evaluate JITA. Additionally, an evaluation checklist for the JITA classification was created using the introduced literature. The checklist, documented in Table iii, is a list of evaluation criteria derived from rules and advices of Dahlberg (1974), Fernando et al. (2012), Gantert (2016), Gaus (2005), Gruber (1995), Hall et al. (2014), Kwaśnik (2021), Manecke (2004), Ranganathan (1937, 1967) and Umlauf (1999).

Since user studies were not feasible, criteria 1, 2 and 3 of Table iii were addressed by defining the needs of the users as the requirements of a classification system for the BHR (as described later in this chapter). They were then judged by the impression of the author of this thesis, since she is a user of JITA too. To check if JITA is in accordance with the recent research topics (criteria 9), the results of the studies of G. Liu and Yang (2019) and Ma and Lund (2021) were employed. They analyzed popular research topics of LIS publications of the past decade (G. Liu & Yang, 2019) and their development by observing the publications in the years 2006, 2012 and 2018 (Ma & Lund, 2021).

Furthermore, to evaluate criteria 1, the 28 proposals for a classification scheme for Information Science collected by Zins (2007a), were compared to each other to obtain a structured collection of Information Science subjects that meets a broad consensus of Information Science experts that then can be compared to JITA. Zins (2007b) himself created a 10-facet hierarchical model using the data from Zins (2007a). It is a very powerful tool,

Table iii: Checklist to evaluate the JITA classification system

	Criteria	Corresponding references	
1.	The classification corresponds with the research field and the probable needs of the users.	*Umlauf, 1999*	Use
2.	The classification system is understandable to the users and must provide an overview of the domain or any kind of new knowledge.	*Fernando et al., 2012; Hall et al., 2014; Kwaśnik, 2021*	
3.	The classification has a hierarchical, easy-to-remember notation and is memorable itself.	*Kwaśnik, 2021; Umlauf, 1999*	
4.	The semantic scope of the broader term must cover the semantic scope of its narrower terms.	*Fernando et al., 2012; Hall et al., 2014; Manecke, 2004; Ranganathan, 1937, 1967*	Hierarchy
5.	The narrower classes must be disjoint.	*Gaus, 2005; Manecke, 2004; Ranganathan, 1937, 1967*	
6.	No jumps in the classification hierarchy.	*Manecke, 2004; Ranganathan, 1937, 1967*	
7.	Items are well placed in the hierarchy.	*Fernando et al., 2012; Hall et al., 2014*	
8.	The order of subordinate classes should follow principles. If subordinate classes appear in more than one main class, their order should be the same or similar in all those main classes.	*Ranganathan, 1937, 1967*	
9.	The classification and its contents must be up to date.	*Gantert, 2016; Manecke, 2004*	Time Aspect
10.	It should be possible to extend the classification in future so that it can be used for a longer period.	*Dahlberg, 1974; Gruber, 1995; Kwaśnik, 2021; Manecke, 2004; Umlauf, 1999*	
11.	The classification system is focused on clarity and seems elegant by avoiding unnecessary details. That means only sufficient information should be included. As few assertions about the modelled domain as possible should be made.	*Gruber, 1995; Kwaśnik, 2021*	Other
12.	All classes contain around the same number of items.	*Umlauf, 1999*	
13.	Classes are formulated and defined as broad and clear as possible and in an objective manner. They should be exhaustive regarding the universe of the classification system. The level of expressiveness is assimilated to the purpose of the classification system.	*Dahlberg, 1974; Gruber, 1995; Kwaśnik, 2021; Ranganathan, 1937, 1967*	
14.	The requirements for a classification system for the BHR are fulfilled.	---	BHR

but its organization is not applicable for comparison since it does not go into enough detail. Thus, instead of using the knowledge map of Zins (2007b), the data from Zins (2007a) was reused. The 28 classification systems were analyzed using the software MAXQDA.

The included concepts were collected, summarized, and finally rearranged into twelve groups. More information about the coding process and the allocation of terms can be found in Appendix II.

Furthermore, for evaluating JITA, the requirements for a classification system for the BHR, based on the analysis of the BHR dataset, were taken into consideration: Because only 463 BHR were published since 1992 to the date of the analysis, the dataset is rather small and not many classes are needed (Gaus, 2005). According to Umlauf (1999) 15-30 items per class are enough for a document set of this size. In other words, a small classification system with 30 classes is sufficient for 900 documents. If the publication rate continuous at the same rate as before, a classification system with 30 classes should have enough space for documents for more than 20 years. Yet, following the formular of Gaus (2005) for calculating the number of required classes with defined values of 463 BHR, an average overlapping factor of 2 and an average class occupancy of 25; the classification system would require 37 classes already at the time this thesis was written. Consequently, it was specified that the final classification system for the BHR should have between 30 and 40 classes.

Moreover, even though Library Science topics are the predominant subjects in the BHR, the main research fields of all chairs of the IBI should be included in the final classification system. In March 2021, five chairs existed at the IBI: Information Behavior, Information Management, Information Processing and Analytics, Information Retrieval, and Information Science. Those chairs address topics beyond the Library Science scope and should be included in the classification system. Especially, since the BHR mostly consist of theses written for one of the IBI chairs. Hence, a classification for the BHR requires a focus on the research fields of the IBI professors and chairs that are representing fields of the interdisciplinary (Borko, 1968; Luft, 2015) LIS universe. To ensure this, the professors of the IBI at the time and in some cases their research assistants were contacted via email. They were asked, if a list of research fields retrieved from their webpages on the IBI website was a good representation of their chair in a classification system for the BHR. Their suggestions

Table iv: Requirements of a classification system for the BHR

Criteria	Characteristics	
Number of Classes:	Between 30 and 40 classes	
Universe:	LIS with a focus on the research fields of the IBI	
Subjects that were asked for by the professors and research assistances of the IBI:	Information Behavior	▪ Theories, Models & Framework of Information Behavior ▪ Information Seeking ▪ Information Use ▪ Human-Computer Interaction & User-Experience ▪ Information Need
	Information Management	▪ Digital Curation ▪ Digital Preservation ▪ Open Science (including Open Access and Open Data) ▪ Digital Repositories ▪ Risk, Privacy, and Ethics [as overarching topics of Information Science] ▪ Archives and Archival Theory [as own category]
	Information Processing and Analytics	▪ Recommender Systems ▪ Social Bookmarking ▪ Data Mining & Machine Learning ▪ Web Archiving ▪ Open Science ▪ Natural Language Processing
	Information Retrieval	▪ Information Systems Evaluation ▪ information Retrieval Evaluation ▪ Scientometrics / Bibliometrics ▪ Multilingual Information Retrieval (MLIR) ▪ Interactive Information Retrieval (IIR) ▪ Digital Libraries (including Metadata, Development, Interoperability, Quality & Managing Heterogeneity) ▪ Cultural Heritage Systems ▪ Information Literacy & Digital Skills ▪ Knowledge Organization ▪ Electronic Publishing ▪ Information Management ▪ Research Data Management ▪ Open Access ▪ Open Science
	Information Science	▪ Philosophy of Information ▪ Information Ethics ▪ Definitions of Information ▪ Human-Computer Interaction ▪ Personal Information Management ▪ Information Organization and Retrieval ▪ Knowledge Representation ▪ Metadata ▪ Information Literacy ▪ Web and Information Systems Design ▪ Database Design ▪ Bibliometrics

and improvements were included in the list of necessary requirements of BHR classification system. However, not all inquiries could be considered in the final classification system since they did not match the BHR dataset. It was requested, for instance, to have an own class for 'Archives and Archival Theory'. However, as can be seen in Table ii, this discipline was barely addressed in the documents of the BHR.

The goal of the evaluation of JITA was to see if JITA can be reused for a classification system of the BHR. After getting to the conclusion that this is not a suitable solution, which will be further elucidated in Chapter 6.1, an alternative classification system had to be created. This was done by using the evaluation criteria of Table iii and Table iv as well as the information of Chapter 4.1. The literary warrant (Kwaśnik, 2021) was followed – that means that the classification scheme was based on the document corpus of the BHR.

After receiving the first version, a sanity check was conducted by the editor of the BHR, Professor Dr. Vivien Petras. In consultation with her, the classification system was adapted several times until it met the requirements of a classification scheme for the BHR. Furthermore, 101 random BHR were manually classified to the new classification system to obtain a test set to evaluate how well the ML and DL algorithms work on the BHR and to make sure that the documents distribute evenly in the classification system. The results will be presented in Chapter 6.2.

5.2 Traditional Machine Learning Classification

A typical framework for ML tasks concerning TC consists of the following steps: Preprocessing of the data, FE, FS, the classification process itself and finally the evaluation (Pong et al., 2008; Uysal & Gunal, 2014). This is illustrated in Figure 4. It should be noted that TC is an iterative process, meaning that individual steps are adapted and repeated at any stage of the framework. For simplicity reasons, this is not depicted in Figure 4.

In the following subchapters each step of the architecture of the study at hand is introduced. For each of those steps, the author of this thesis received help writing the programming code from another party, namely Ján M. Hanes. Yet, all decisions about the architecture of the models were made solely by the author.

5.2.1 Creating the Dataset

For every ML (or DL) task, a dataset, the machine can train on, is needed (Aggarwal & Zhai, 2014). For DC this means a set of already labeled documents. The labels are the respective classes the documents need to be assigned to. Since the BHR did not have a classification system with documents so far and JITA could not be used as classification scheme, no such training set existed. Instead, documents from four different databases were extracted: *DABI - Datenbank Deutsches Bibliothekswesen* (DABI), e-LiS, *SpringerLink* (Springer) and *o-bib. Das offene Bibliotheksjournal* (o-bib). Due to access constraints only abstract, title and language of a document were scraped instead of the entire full text. To avoid null values and false data, it was decided that abstracts must consist of at least 150 characters. Additionally, the name of the databases where the metadata was scraped from and the respective classes in the BHR classification system were stored as well. The latter were determined in different ways depending on the database: For e-LiS and DABI a mapping was created from classes of their classification systems to the classes of the BHR classification system. The documents within the respective classes of those two databases were allocated to the BHR classification classes according to this mapping. Springer and o-bib do not offer a LIS classification scheme. Therefore, customized queries in German for o-bib and in English and German for Springer were written for each subclass of the BHR classification system. No query was

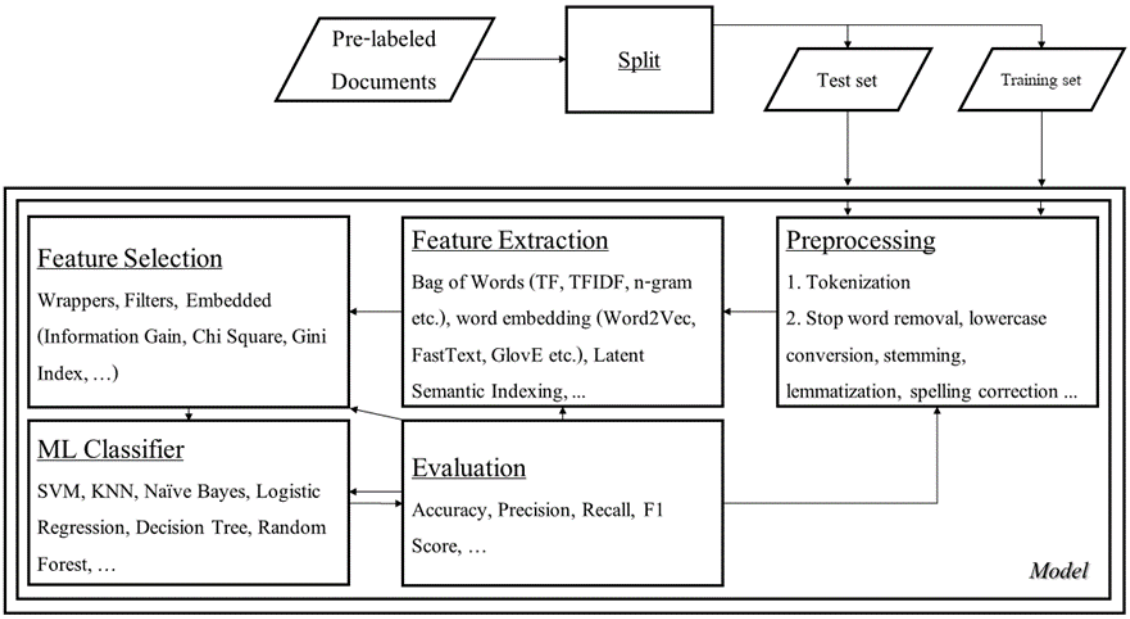


Figure 4: Basic architecture of TC using ML

written for main classes of the BHR classification system because those are too broad to formulate queries and furthermore are intended to function as an ‘Others’ class and should only be used as last instance. Mappings, queries, and codes for the scrapers can be found in Appendix III. The described information that serves as the dataset was stored in a CSV file and is attached in Appendix III.

The scrapers finished running on July 23, 2021. The raw dataset consisted of 175,508 documents. Most of the documents by far were scraped from Springer with 170,895 documents (97.37 %); followed by 3,294 documents from e-LiS (1.88 %); 1,030 documents from DABI (0.59 %) and finally 289 documents from o-bib (0.16 %). A first step of data cleaning was done by deleting null and false values: 57 documents had no assigned BHR classification system class and 3 documents had no or false entries in the language column. Moreover, 1,080 documents had titles with less or equal to 10 characters (e.g. ‚Education‘ or ‚Goethe‘). Documents with titles of that length were not considered as holding much or correct information for the classification and were deleted from the dataset. The number of assigned classes of the BHR classification system of the remaining documents ranged from 1 to 25 classes. Since a high number of classes seemed inaccurate for a reliable classification, any document with more than five classes was deleted from the dataset (10,769 documents). After the deletion only the first three classes of the remaining documents were considered and thus, the columns of class 4 to 25 deleted (12,038 documents influenced). This was done because documents should only be assigned to a maximum of three classes in the BHR classification system. It was examined beforehand if BHR classes with only a few documents would lose a considerable, and potentially valuable number of documents, but this was not the case.

As will be described in Chapter 5.3, a variant of BERT was used for DL. BERT has a maximum sequence length of 512 tokens. Thus, all documents that have more words in title and abstract combined were deleted as well (78 documents). This did not completely solve the problem of a maximum of 512 tokens, since not the original dataset, but a translation was used in FLAIR and because BERT applies WordPiece tokenization. The WordPiece tokenization uses the longest-match-first strategy, which means if a word is not in the vocabulary of the tokenizer, it searches for subwords in the vocabulary with the same strategy and the token is divided into several subtokens (Song et al., 2020). Consequently, even if documents with more than 512 words were deleted, the number of tokens per

document could be higher than 512. However, documents with more than 512 words would be truncated in any case, which is why they were deleted in the data cleaning step.

Finally, in the translating step, which will be described below, the maximum number of characters is 5000 characters. Following, all documents with a totalized character length of greater than 5000 characters in title and abstract were excluded from further processing. However, there were none that matched this criterion.

The final dataset consisted of 163,521 instances. Of those documents 97.39 % were scraped from Springer, 1.96 % from e-LiS. 0.63 % from DABI and 0.03 % from o-bib. The distribution of the scraped documents over the BHR classification system is depicted in Figure 5. The documents are unevenly distributed which demands extra steps in the ML pipeline. Those will be discussed in the following chapters as well. The final dataset as well as the Jupyter Notebook that was written for the data cleaning can be found in Appendix IV.

84.94 % (138,901 documents) of the documents are written in English, 15.06 % in German (24,620 documents). This did not represent the BHR documents well but not surprising due to limited German LIS publications. To simplify the classification task, Google Translate was used to translate the titles and abstracts of the documents. Google Translate was rated as an efficient or even the most accurate translator in Vanjani and Aiken (2020), Ziganshina et al. (2021), and Zulfiqar et al. (2018).

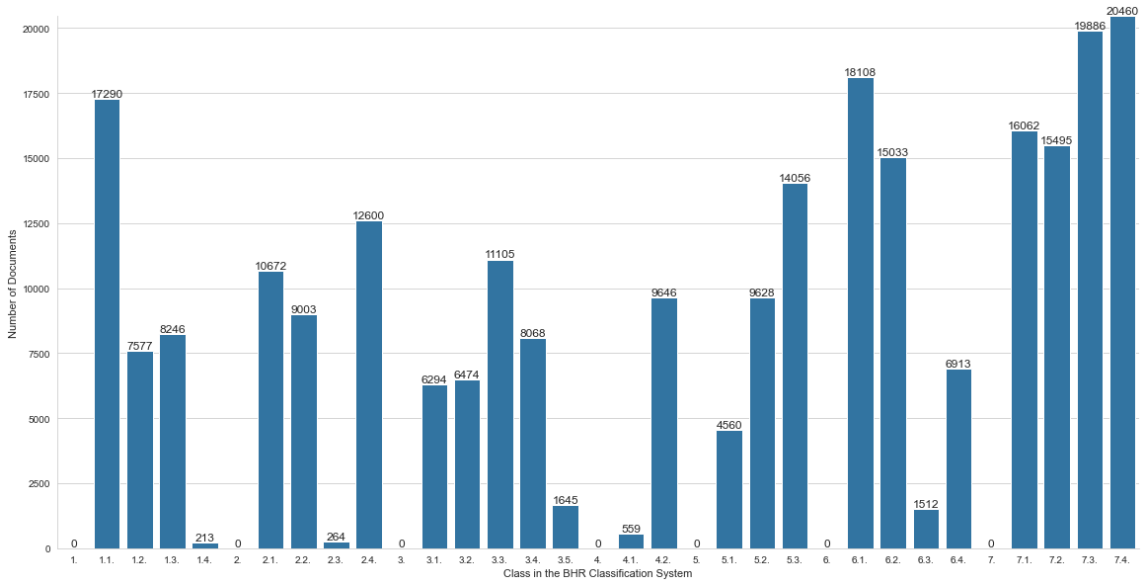


Figure 5: The distribution of the scraped documents on the BHR classification system

It was decided to create two new datasets based on the cleaned data: One German dataset and one English dataset. For the translation, title and abstract were merged into one text. Furthermore, in the German dataset eight documents were lost. The translated datasets and the python code for the translation can be found, in Appendix IV as well. Both datasets are used for the automatic classification depending on the classifier which will be elaborated later in the following subchapters.

5.2.2 Splitting the Data & Dimensionality Reduction

To be able to examine the quality of the classification process, a split into training and test set is necessary. To avoid overfitting and support the selection of parameters, a third so-called validation set can be created (Sebastiani, 2002). For the study at hand, all three sets were used. The classifiers were trained on the training set, optimized using the validation set and finally tested on the test set. The train/test split ratio is often a number between 70,0 %/30,0 % and 80,0 %/20,0 % where the higher relative share denotes the proportion of training data (Shah et al., 2020; Spirovski et al., 2018; Ting et al., 2011). Thus, in this study it was decided to split the data in the ratio of around 70,0 %/15,0 %/15,0 %, meaning that around 70,0 % of the data was used to train the model, and around 15,0 % each for testing and validation. Since the scraped dataset, as described in Chapter 5.2.1 is rather imbalanced, stratification was applied. Stratified sets draw appropriate proportions of datapoints for each class according to the class size. Thus, the distribution is preserved in training, test and validation set. Because some documents have more than one class, a perfect split of 70,0 %/15,0 %/15,0 % was not possible.

The distribution of the English dataset is presented in Table v. The split algorithm was used on the German and on the English dataset and for all classifiers. The python code for the dataset split can be found in Appendix IV.

As for the preprocessing, different combinations of methods were compared. For the English dataset respectively the traditional ML algorithms, the following preprocessing steps were compared:

1) Stop-word-removal (including punctuation) and lowercase conversion
2) Stop-word-removal (including punctuation), lowercase conversion and lemmatization
3) Stop-word-removal (including punctuation), lowercase conversion and stemming

Table v: Distribution of the scraped data on training, test and validation set in absolute and relative numbers

Class label	Training Absolute	Training Relative	Test Absolute	Test Relative	Validation Absolute	Validation Relative
1.	0	0,00	0	0,00	0	0,00
1.1.	12103	70,00	2593	15,00	2594	15,00
1.2.	5304	70,00	1136	14,99	1137	15,01
1.3.	5772	70,00	1237	15,00	1237	15,00
1.4.	149	69,95	32	15,02	32	15,02
2.	0	0,00	0	0,00	0	0,00
2.1.	7470	70,00	1601	15,00	1601	15,00
2.2.	6303	70,01	1350	15,00	1350	15,00
2.3.	187	70,83	37	14,02	40	15,15
2.4.	8820	70,00	1890	15,00	1890	15,00
3.	0	0,00	0	0,00	0	0,00
3.1.	4408	70,03	942	14,97	944	15,00
3.2.	4623	71,41	880	13,59	971	15,00
3.3.	7775	70,01	1664	14,98	1666	15,00
3.4.	5787	71,73	1071	13,27	1210	15,00
3.5.	1163	70,70	235	14,29	247	15,02
4.	0	0,00	0	0,00	0	0,00
4.1.	395	70,66	80	14,31	84	15,03
4.2.	6757	70,05	1442	14,95	1447	15,00
5.	0	0,00	0	0,00	0	0,00
5.1.	3192	70,00	684	15,00	684	15,00
5.2.	6817	70,80	1367	14,20	1444	15,00
5.3.	9839	70,00	2109	15,00	2108	15,00
6.	0	0,00	0	0,00	0	0,00
6.1.	12676	70,00	2716	15,00	2716	15,00
6.2.	10523	70,00	2255	15,00	2255	15,00
6.3.	1058	69,97	227	15,01	227	15,01
6.4.	4839	70,00	1037	15,00	1037	15,00
7.	0	0,00	0	0,00	0	0,00
7.1.	11243	70,00	2410	15,00	2409	15,00
7.2.	10846	70,00	2325	15,00	2324	15,00
7.3.	13920	70,00	2983	15,00	2983	15,00
7.4.	14322	70,00	3069	15,00	3069	15,00

Stemming reduces a word or token to its stem, meaning its morphological root by removing affixes (Moral et al., 2014). In contrast, lemmatization gives back the dictionary form of a word (Manning et al., 2008). For the English stop-word-removal the default 'english' stop word list within the CountVectorizer[11] of scikit-learn was used. This list was originally created by the Glasgow Information Retrieval Group[12]. The hyperparameter 'lowercase' was set to True to enable lowercase conversion. The default tokenizer of the CountVectorizer, that was applied for the first English preprocessing option, automatically removes punctuation, but in a preceding step, the special characters were removed manually anyway. This was necessary, because for the following two preprocessing options, lemmatization and stemming, the tokenizer was customized using word_tokenize package[13] of the NLTK library[14] which does not include punctuation removal. For stemming, the English Snowball stemmer[15] developed by Martin F. Porter was applied using the NLTK library as well. For lemmatization the WordNetLemmatizer[16] of the NLTK library was used.

As FE, the TFIDF method was chosen for weighting the terms, because it is very popular for TC (Kass, 2019; Miao et al., 2018; Shah et al., 2020; Spirovski et al., 2018). To reduce training time and noise, FS was also applied through the chi-square test. The top 1500 features were selected. The chi-square test is a statistical method that can be used to measure the independency of two variables (Aggarwal & Zhai, 2014; Yang & Pedersen, 1997). Reversely, for FS in TC tasks chi-square is used to determine whether a term and a class are dependent. If that is the case, the term contains relevant information and will be selected for further processing. Yet, since FS removes features from the feature vector and Forsyth (2019) states that classifiers with a poor performance potentially can be improved by adding features, experiments without FS were also performed. If FS was used, it was done by using the package SelectKBest[17] of sklearn with a k value of 1,500, meaning the top 1,500 will be selected for further processing.

[11] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html [September 25, 2021]
[12] http://ir.dcs.gla.ac.uk/resources/linguistic_utils/stop_words [September 25, 2021]
[13] https://www.nltk.org/api/nltk.tokenize.html [September 25, 2021]
[14] https://www.nltk.org/index.html [September 25, 2021]
[15] http://snowball.tartarus.org/algorithms/english/stemmer.html [September 25, 2021]
[16] https://www.nltk.org/_modules/nltk/stem/wordnet.html [September 25, 2021]
[17] https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html [September 25, 2021]

5.2.3 Classifiers, their Hyperparameters & Evaluation Methods

As discussed in Chapter 4.3, SVM, Logistic Regression and Naïve Bayes were selected as the most suitable classifiers for this study. All of them can get different hyperparameters assigned. In the following, packages of scikit-learn for the classifiers and their given hyperparameters will be introduced.

Naïve Bayes is the classifier with the easiest implementation. Because of the underlying unbalanced dataset, its variation, the Complement Naïve Bayes (Rennie et al., 2003) was used via the package ComplementNB of scikit-learn. The only hyperparameter given to the classifier and tested on the validation set is the Laplace smoothing hyperparameter alpha. Laplace smoothing is a method to avoid calculation errors caused by zero values (Ramadhani et al., 2016). The following list of possible alpha values was tested: 0, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2. To yield multi-label results, the MultiOutputClassifier package of scikit-learn is used for this and all other traditional ML classifiers.

For Logistic Regression, the LogisticRegression[18] package of scikit-learn was utilized. As hyperparameters 'saga' as solver; L1 for penalty; one CPU core and two lists, one for inverse of regularization strength (a hyperparameter to reduce overfitting) and one for the maximum number of iterations, were chosen deliberately. For the inverse of regularization strength, the following values were tested: 0.1, 0.15, 0.2, 0.25, 0.3, 0.5, 0.8, 1, 3, 5, 10, 30, 60, 100, 150, 500, 1000. For the maximum number of iterations, the following numbers were tested: 100, 150, 200, 250, 500, 750, 1000, 1250, 1500, 3000, 4000, 5000. All other hyperparameters were given the default value.

For the SVM classifier the same two lists for regularization and maximum number of iterations were chosen. To save runtime, the LinearSVC package of scikit-learn was chosen deliberately with the hyperparameters L2 for penalty and dual optimization problem. All other hyperparameters remained with the default value.

Even though accuracy is not an evaluation measurement without risks, it was decided to use it as an evaluation output, because it is an established evaluation metric as reported in

[18] https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html [September 25, 2021]

Chapter 3.3.5. Nevertheless, the F1 score is a more reliable value because the dataset is unbalanced. Hence, the F1 score, precision and recall, and macro-averaged values were also calculated. Additionally, the precision-recall curves and the runtime will be evaluated. Only the 26 best models of each classifier, based on the F1 score, will be discussed in Chapter 6.3 and Chapter 7.2.

5.3 Deep Learning Classifier

For conducting a DL classification, the FLAIR framework was employed (Akbik et al., 2019). FLAIR is an open-source deep learning framework that is used for NLP tasks and aiming to be easy in use. One of its core concepts is the wide range of embeddings and one of its strengths a broad range of language models. To make use of one of this powerful feature, FLAIR was applied on the German dataset. It also offers its own state-of-the-art word embeddings using a pre-trained character-level language model as described in Akbik et al. (2018). However, in the official FLAIR tutorials, it is recommended to use the fine-tuned TransformerDocumentEmbeddings for embedding and most text classification tasks, which requires a transformer instead of word embeddings. Since BERT (Bidirectional Encoder Representations from Transformers) yields state-of-the-art results (Devlin et al., 2018), the 'bert-base-german-dbmdz-cased'[19] transformer was chosen for the experiments. BERT was developed by Google members and is a language-model based on transformers. It offers a training of the transformer in a way that it processes the text from left to right and from right to left at the same time. This allows a deeper understanding of the text. 'bert-base-german-dbmdz-cased' is a German-language model trained on various data sources such as Wikipedia text, subtitles, news and more. BERT has a limit of 512 tokens as input and uses WordPiece tokenization.

As an optimizer, that means a method to reduce the loss function, Adam (Kingma & Ba, 2014) was used. Adam is a gradient descent algorithm that is widely applied (Guibin Chen et al., 2017; Kass, 2019; Kowsari et al., 2019; Spirovski et al., 2018). Since this algorithm was also used for the experiments with BERT in Devlin et al. (2018) and for the development of transformers in Vaswani et al. (2017), Adam was considered a good choice for an optimizer for the study at hand as well. The loss is a cost function that is used to evaluate errors and its result is meant to be kept as low as possible (Forsyth, 2019).

[19] https://github.com/dbmdz/berts [September 25, 2021]

Hyperparameters that must be chosen are the learning rate, a mini batch size, and the maximum number of epochs. For most experiments only the learning rate was changed while mini batch size and maximum number of epochs remained the same with a value of 20 for the mini batch size and 150 for the maximum number of epochs. The following learning rates were tested: 0.07, 0.03, 0.1, 0.75, 0.65, 20. For a learning rate of 0.03 and 0.75, a maximum number of 500 epochs was tested as well.

FLAIR does not require any preprocessing (Akbik et al., 2019) and therefore the German training, test and validation set were taken as input without any additional changes. For evaluation, FLAIR outputs different values from which runtime, precision, recall, F1 score, and their macro averages will be analyzed.

5.4 Experimental Setup

Some of the chosen methods showed state-of-the-art results in the past. This comes at the cost that they are computationally expensive. Therefore, the following experimental setup including powerful equipment**,** was used:

The Humboldt-Universität zu Berlin offers its members access to different servers. For this thesis the machine learning and deep learning experiments are performed on the servers gruenau1 and gruenau2[20]. Both provide the same hard and software: They are a Dell R740xd with a SuSE Leap 15 operating system, a Xeon 6254 CPU, 756 GB RAM and a Nvidia RTX6000 GPU (130 Tera Operations per Second, 24 GB RAM, 1770 MHz GPU Speed and 4608 CUDA cores). On those two servers, all classifiers were tested, since especially FLAIR and the SVM were expected to require a long runtime and to be computationally costly.

The other server, Jerry[21], was provided by the Information Processing and Analytics research group at IBI. Jerry offers an Ubuntu 18.04.5 LTS as operating system, an AMD EPYC 7351P 16-Core Processor CPU and 131.9 GB RAM. On Jerry, the scraping of the data, the translation, and the split took place. Furthermore, some experiments and models were

[20] https://www.informatik.hu-berlin.de/de/org/rechnerbetriebsgruppe/dienste/hpc/computeserver [September 25, 2021]
[21] https://www.ibi.hu-berlin.de/de/service/rechen-und-datenressourcen [September 25, 2021]

conducted on Jerry as well to save time. This makes it hard to compare runtimes, but since the models only have to be trained once, the runtimes only played a minor role.

The overall architecture of the experiment is visualized in Figure 6. The Laplace smoothing for Naïve Bayes was abbreviated with alpha and for SVM and Logistic Regression the (inverse of) regularization strength was abbreviated with C and the maximum number of iterations with max_iter.

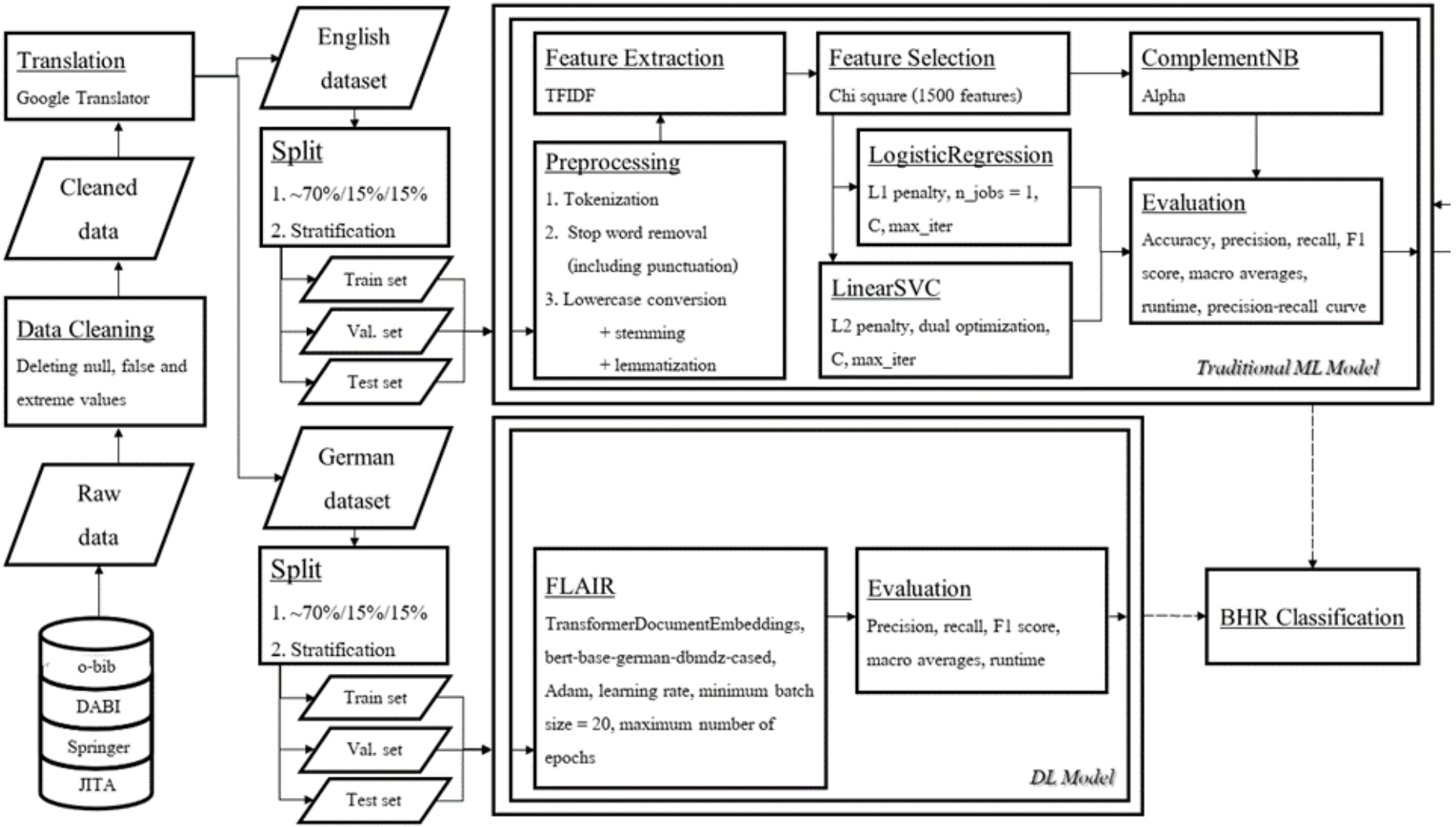


Figure 6: General Experiment Architecture with four different classifier

6 Results

6.1 Evaluation of JITA

JITA was first evaluated using the checklist from Table iii in Chapter 5.1. In the following, every criterion mentioned in the table will be evaluated one by one.

The first set of evaluation criteria is about the use of a classification scheme. For criterion 1, the needs of the users were equated with the requirements of Table iv and the dataset of the BHR itself, which was described in Chapter 4.1. Furthermore, a structured collection of Information Science subjects in Appendix II was compared to JITA to verify JITAs correspondence with the Information Science research field: JITA claims to be a classification scheme for Library and Information Science. The structured collection consists of terms derived from classification schemes only for Information Science, but since Library Science and Information Science are closely related, topics of Library Science also occur. Nonetheless, the primary focus is on Information Science. Thus, it was expected that JITA does not cover all concepts listed in the structured collection in Appendix II. Yet, JITA seems to be rather Library Science focused: Main classes 'D. Libraries as physical collections.' and 'J. Technical services in libraries, archives, museum.' are solely focused on libraries or memory institutions in general. It is assumed that main classes 'C. Users, literacy and reading.'; 'I. Information treatment for information services'; 'F. Management.' and 'H. Information sources, supports, channels.' are rather leaned towards these institutions as well. The remaining seven main classes are either a mix of Information Science and Library Science topics or not really focused on either of these two research fields (i.e., classes 'K. Housing technologies.'). In comparison with the structured collection of Information Science subjects (Appendix II), it became apparent that topics such as evaluation; research methods; information seeking; digital libraries; information systems; and ethical, historical and philosophical aspects of Information Science are not or only loosely present in JITA. However, one may argue that those concepts could be allocated to other classes, e.g., the evaluation of a search engine could be allocated to class 'LS. Search engines.'. Moreover, a lot of the remaining concepts from structured collection of Information Science subjects are covered (Appendix II).

Furthermore, not all aspects of LIS must be relevant to a classification system for the BHR, because not all subfields match the requirements. Thus, it is more important to evaluate if

JITA provides classes to cover the requirements of Table iv and the BHR dataset than covering the entirety of LIS. In Figure 7 the requested research fields of the professors and research assistants of the IBI are marked in yellow if they are covered completely by a class in JITA and gray if they are partially covered by a class. The notation of the respective classes is given in square brackets as well. As can be drawn from this figure, it is possible to allocate most research fields to one class in JITA: 3 out of 5 requests of Information Behavior (60,0 %); 4 out of 4 requests of Information Management (100,0 %); 6 out of 6 requests of Information Processing & Analytics (100,0 %); 13 out of 14 requests of Information Retrieval (92,9 %) and 11 out of 12 requests of Information Science (91,7 %) are fulfilled or somewhat fulfilled.

However, most concepts do not have a satisfying representation (marked in gray): 2 out of 3 fulfilled requests of Information Behavior (66,7 %); 2 out of 4 fulfilled requests of Information Management (50,0 %); 3 out of 6 fulfilled requests of Information Processing & Analytics (50,0 %); 10 out of 13 fulfilled requests of Information Retrieval (76,9 %) and 5 out of 11 fulfilled requests of Information Science (45,5 %) are not completely satisfactory.

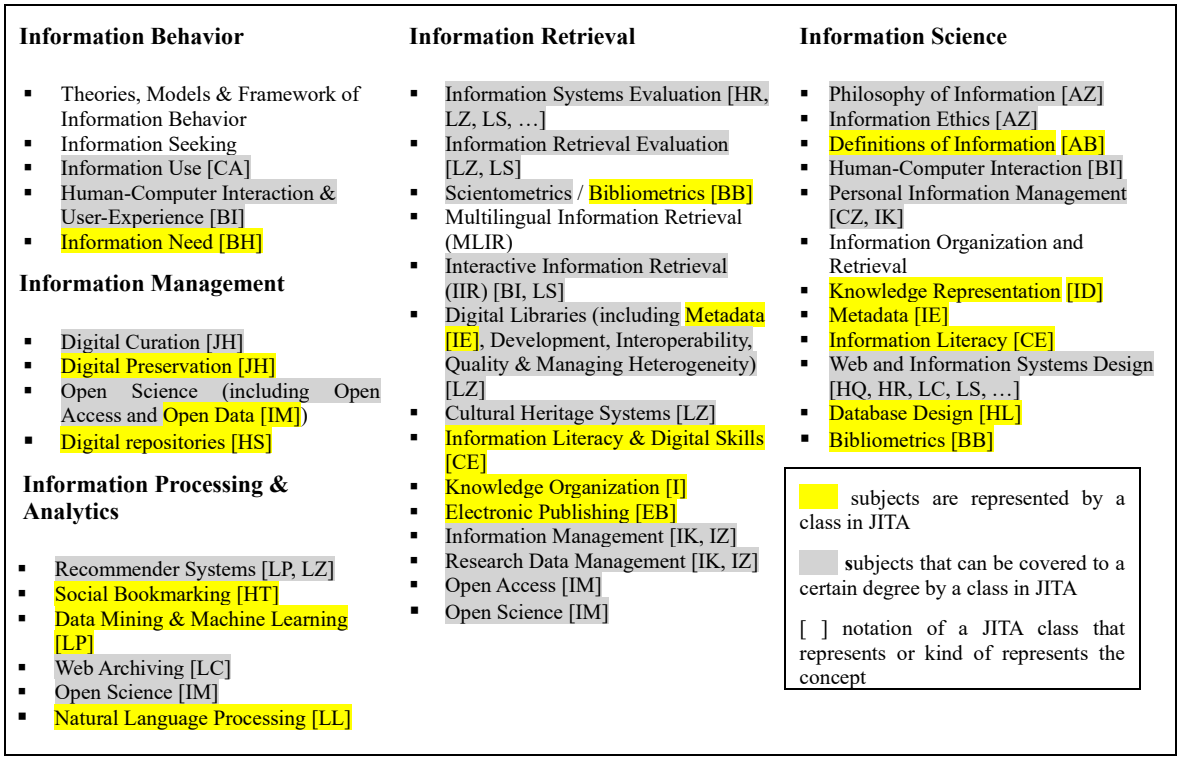


Figure 7: Requested research fields of the IBI professors and research assistants marked by their occurrence in JITA

The often requested issues of Open Science, Open Access and Open Data for instance, could potentially be assigned to JITAs class 'IM. Open data'. Yet, Open Science is a superordinate rather than a subordinate of Open Data.

Furthermore, four research fields like the one of Information Seeking, do not have an appropriate counterpart in JITA. Examples like that prove that JITA is only partially fulfilling the first point on the evaluation checklist (Table iii). On the other hand, when the BHR subjects are compared to the JITA classes, it seems possible to classify most of them (Figure 8): 19 out of 28 subjects (67,9 %) are addressed or partially addressed; 5 out of those 19 subjects are only partially addressed (26,3 %). 'Actors', 'Institutions', 'Languages', 'Other' and 'People' are groups of subjects that are very specific and therefore it is no surprise that they cannot be assigned to any JITA class. Therefore, 'History' and 'Scientific Practices', that cannot be appropriately allocated to any class, should be focused on more.

Overall, it seems that criterion 1 from the evaluation checklist is only partially fulfilled. Even though JITA covers a lot of topics from the LIS field (Appendix II), it is not optimal for the BHR specific topics: 67,9 % of the subjects of the BHR are covered, but important fields (according to the requirements of the IBI) are missing or not covered satisfactorily, such as Information Retrieval or Information Behavior disciplines. Thus, criterion 1 is evaluated as half fulfilled.

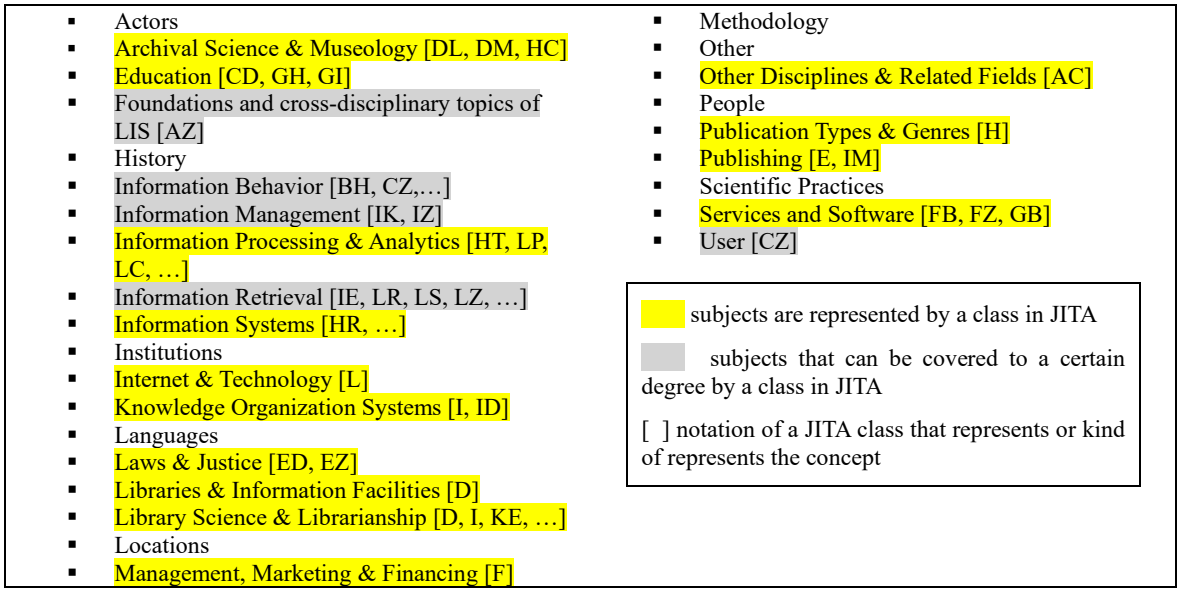


Figure 8: Groups of BHR subjects marked by their occurrence in JITA

The next criterion of a good classification system is its comprehensibility and its ability to provide the user with an overview or new knowledge about the domain. The first impression of the author of the thesis at hand was that parts of JITA are not understandable. This impression was enhanced after asking an independent person from the personal network of the author for their opinion. It was resembling the author's stance: The class names could be more detailed. For instance, it is not obvious what kind of users are meant in class 'C. Users, literacy and reading.' or to which research field of LIS, class 'F. Management.' is referring. Furthermore, to mention other examples, it is not clear how main class 'K. Housing technologies.' is related to the other main classes. A final confusing example is that the number of documents in the main classes does not equal the summed-up number of the documents of their subclasses. Despite those irritating discrepancies, an overview over general concepts of LIS is given (as also proven by the comparison with the structured collection of Information Science subject in Appendix II), even though commonly used terms for fields like Information Management or Knowledge Organization are not used in the classification system. However, as elaborated above, the disciplines of Information Retrieval and Information Behavior need more representation in JITA, and some classes could be named in a broader manner (such as 'IM. Open Data'). Overall, once more, criterion 2 is seen as only half fulfilled, because, ignoring the discrepancies, the rest of the classification system can be understandable to users, and they receive an overview of the domain.

Criterion 3 is fulfilled. The JITA notation is easy and logical. The only possible critic is that due to the notation, the number of possible classes are limited: There is a maximum number of 26 main classes and per main class a maximum of 25 free subordinate classes, since the last one (…Z) is always reserved for 'None of these, but in this section.'.

Criterion 4 is given for most classes. However, there are exceptions: 'DL. Archives.' and 'DM. Museums.' are not libraries and hence should not be a subordinate of 'D. Libraries as physical collections.'. Furthermore, it is arguable if e.g., 'KC. Furniture.' should be a part of 'K. Housing technologies.'. The name of the main category implies that the category 'KC. Furniture.' includes documents about a scientific process concerning furniture, e.g., the techniques or methods that are deduced by scientific knowledge to produce a chair. This however is not within the range of the LIS field.

Criterion 5 is not fulfilled. Some subordinate classes show intersections: ‘CB. User studies.’ and ‘CC. User categories: children, young people, social groups.’ are likely to share the same documents. Likewise, ‘HD. Rare books and manuscripts.’ and e.g., ‘HE. Print materials.’; and ‘HG. Non-print materials.’ and most other subclasses in ‘H. Information sources, supports, channels.’ (e.g., ‘HI. Electronic Media.’, ‘HJ. CD-ROM.’, ‘HK. Online hosts.’ etc.) overlap. ‘HP. e-resources.’ collides with ‘HO. e-books.’ and ‘HN. e-journals.’. ‘KB. Library, archive and museum buildings.’ and ‘KE. Architecture.’; ‘KE. Architecture.’ and ‘KF. Planning, Design, Removal.’; and ‘LL. Automated language processing.’ and ‘LM. Automatic text retrieval.’ also overlap, to only name a few. Finally, it is confusing that documents can be allocated to the main class too, which makes the last subordinate class ‘[…]Z. None of these, but in this section.’ of each main class redundant. Since Information Science is interdisciplinary, most problems are related to each other. Yet, the subcategories should not overlap as proposed in criterion 5.

Even though a low hierarchy is used for JITA, criterion 6 is not completely fulfilled. The subclasses are sometimes on different description levels. Taken class ‘LE. Scanners.’ and class ‘LL. Automated language processing.’ for instance, this becomes clear. Scanners are physical objects while automated language processing is an entire research field. Still, they both are on the same level in the same main class. It is similar with class ‘HT. Web 2.0, Social networks’ and the rest of the subclasses in main class ‘H. Information sources, supports, channels.’ or ‘GA. Information industry.’ and ‘GF. Biographies.’. Yet, most of the time, there are no jumps in the hierarchy. Therefore, it was decided to mark criterion 6 as partially fulfilled.

The same applies to criterion 7. Most of the items, with a few exceptions, are well placed. One exception was already mentioned earlier: Subclasses ‘DL. Archives.’ and ‘DM. Museums.’ are not covered by the main category ‘D. Libraries as physical collections.’. However, they are closely related to the other subclasses in this category. Thus, it could be said that they are well placed, but the title of the main category is not broad enough. Another exception is “GE. Staff.” which seems lost in class ‘G. Industry, profession and education.’ and more as part of subcategory ‘FE. Personnel management.’ or ‘GI. Training.’. Moreover, subclass ‘GF. Biographies.’ also could be placed in another main class. It is understood as a very specific subclass of main category ‘H. Information sources, supports, channels.’.

For the subcategories, no order is recognizable. If at all, the concepts were ordered according to the order of their respective research fields in the title of the main class (e.g., main class 'C. Users, literacy and reading.' and its subclasses). This was just rarely the case, however, which is why it was decided to rate criterion 8 as half fulfilled.

To examine if JITA is up to date, first, the results from the comparison of the classification system with the structured collection of Information Science subjects (Appendix II) were used once more. As stated above, topics such as evaluation; research methods; information seeking; digital libraries; information systems and ethical, historical, and philosophical aspects of Information Science are missing or not directly addressed. Furthermore, the research results of G. Liu and Yang (2019) and Ma and Lund (2021) were used. The most recent popular research topics in LIS from 2008 to 2017 in 41 peer-reviewed journals, according to G. Liu and Yang (2019) and the top research topics according to Ma and Lund (2021) are listed in Figure 9. They are marked in yellow if they are completely covered by a class in JITA and in gray if partially covered. Most of the topics are included in JITA (Figure 9): 21 out of 28 topics (75,0 %) detected by G. Liu and Yang (2019) and 7 out of 10 topics (70,0 %) found by Ma and Lund (2021) are covered or somewhat covered by JITA. 10 out of those 21 covered topics (47,6 %) by G. Liu and Yang (2019) and 3 out of 7 covered topics (42,9 %)

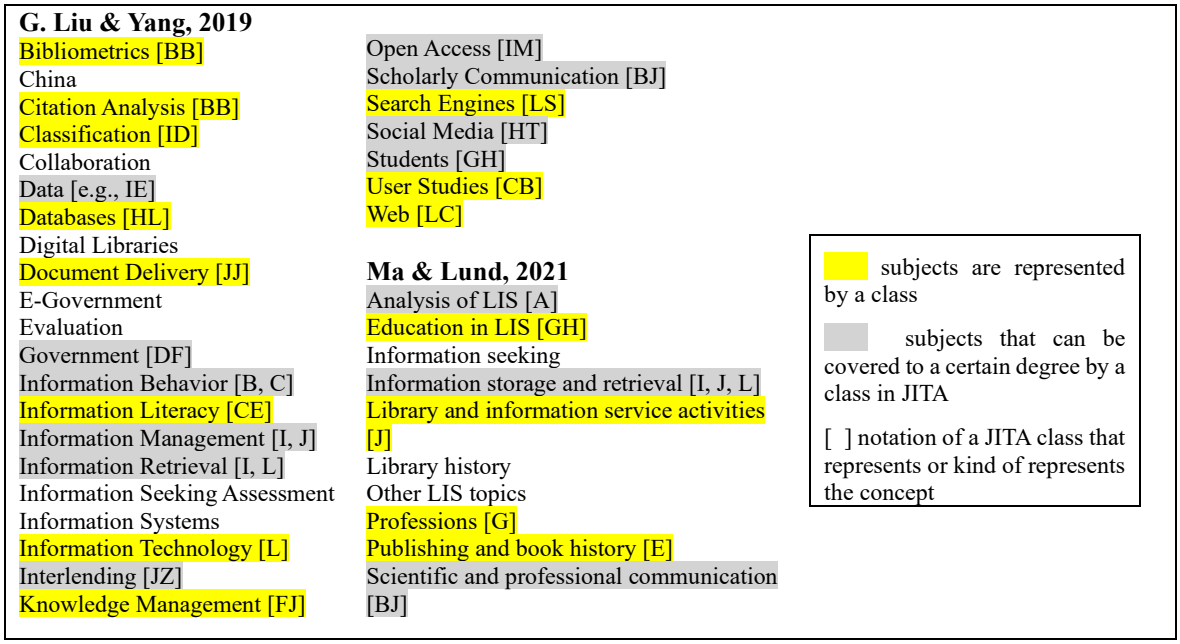


Figure 9: The most recent LIS research topics according to G. Liu and Yang (2019) and Ma and Lund (2021) marked by their occurrence in JITA

by Ma and Lund (2021) are only partially addressed in JITA. Consequently, some research fields are missing in JITA: Since ‘China’ and ‘Other LIS topics’ are too specific or broad it is no surprise that they are not represented by a class in JITA. Furthermore, documents about ‘Evaluation’ could be assigned to the specific item that is evaluated. ‘Collaboration’, ‘Digital Libraries’, ‘E-Government’, ‘Information Seeking Assessment’, ‘Information Systems’, ‘Information Seeking’ and ‘Library History’ are hard to assign to classes. This does not seem to be a problem of topicality, but rather a problem concerning criterion 1. Since most other topics are present in JITA, criterion 9 was rated as fulfilled.

For a long-lasting classification system, it should be possible to extend it or in other words to add concepts. The advantage of a missing order is that it is easy to add new classes to JITA. The only limitation is created by the notation, as mentioned earlier: Using Latin letters as notation, implies that there is a maximum of 26 superordinate classes and 25 free subordinate classes for each main class (the Z is always reserved for ‘None of these, but in this section.’). This means that 14 main classes could still be added to JITA and in all existing superordinate classes, subordinate classes could still be extended. Even though the number of addable classes is limited, it is more than enough for several years. Thus, criterion 10 is fulfilled.

In contrast, criterion 11 is not fulfilled. This is due to unnecessary and confusing details such as the text ‘(A and I, class.)’ in the class name ‘IB. Content analysis (A and I, class.)’ or the questionable importance of class ‘K. Housing technologies.’ and its subclasses. As depicted in Figure 3 the number of documents per superordinate class varies. Especially the class ‘K. Housing technologies.‘ is a clear outlier. It contains the least documents (280) of all main classes. This might be because class K contains very specific subordinate classes, such as ‘KC. Furniture’ or ‘KD. Vehicles.’, which are the categories with the least allocated documents. Other examples of categories with only a few documents are ‘LG. Photocopiers.’, ‘LE. Scanners.’ or ‘HF. Microforms.’. Those classes have in common that they cover very small and specific areas. This is another reason why criterion 11 is not fulfilled. Neither is criterion 12 based on Figure 3.

Criterion 13 can be seen as addition to criterion 11. It adds that the classes must be formulated broad enough, to cover the universe to which the classification system refers. The universe of JITA is LIS. Therefore, in this case, the universe is equal to the research field of LIS and this criterion can be evaluated like criterion 1 in combination with criterion 11.

Thus, since JITA is missing some research fields (e.g., Information Seeking) and is sometimes too detailed (e.g., main class K), criterion 13 is not fulfilled.

It was already discussed in the evaluation of criterion 1 if the requirements based on the research fields of the IBI chairs are met: Most topics are included in JITA, but many in a rather unsatisfactory manner. Another requirement is that the final classification scheme of the BHR should have between 30 and 40 classes. JITA has in total 153 main- and subclasses. To match the BHR requirements, more than two thirds of the classes would have to be excluded. Therefore criterion 14 is not fulfilled.

In Table vi, the evaluation checklist from Table iii is filled out by marking the criteria with a checkmark if the criterion was met or with a X if it was not met. If a criterion was only partially fulfilled, it was marked with a checkmark with dashed lines. Out of 14 Features, five were marked as not fulfilled while the leftover nine were documented as fulfilled or rather fulfilled. Out of these nine, six criteria were not fulfilled to complete satisfaction. Hence, only three criteria are fully satisfied. Based on those results, it was decided that JITA cannot be reused as classification system for the BHR. Instead, a new one was created. It will be introduced in the following chapter.

6.2 Creation of the Classification System for the Berliner Handreichungen

After the decision was made that JITA cannot be reused for the BHR, a completely new classification system had to be created. The creation process was heavily based on the numbers of Table ii in Chapter 4.1. After revising the first draft of the classification scheme four times in consultation with Professor Dr. Vivien Petras – the editor of the BHR – the final scheme with the name ‘Klassifikationssystem für die Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft’ (English name: ‘Classification System for the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft’; KBHR) was established. The English version is shown in Figure 10. The German version can be found in Appendix V. In the following, its individual features will be introduced. At the same time, to assure a good quality of the newly introduced classification system, the criteria of the evaluation checklist in Table iii are examined once more.

Table vi: Filled out evaluation checklist for JITA

		Feature	
1.	☑	The classification corresponds with the research field and the probable needs of the users.	Use
2.	☑	The classification system is understandable to the users and must provide an overview of the domain or any kind of new knowledge.	
3.	☑	The classification has a hierarchical, easy-to-remember notation and is memorable itself.	
4.	☑	The semantic scope of the broader term must cover the semantic scope of its narrower terms.	Hierarchy
5.	☒	The narrower classes must be disjoint.	
6.	☑	No jumps in the classification hierarchy.	
7.	☑	Items are well placed in the hierarchy.	
8.	☑	The order of subordinate classes should follow principles. If subordinate classes appear in more than one main class, their order should be the same or similar in all those main classes.	
9.	☑	The classification and its contents must be up to date.	Time Aspect
10.	☑	It should be possible to extend the classification in future so that it can be used for a longer period.	
11.	☒	The classification system is focused on clarity and seems elegant by avoiding unnecessary details. That means only sufficient information should be included. As few assertions about the modelled domain as possible should be made.	Other
12.	☒	All classes contain around the same number of items (for small datasets 15-30 items).	
13.	☒	Classes are formulated and defined as broad and clear as possible and in an objective manner. They should be exhaustive regarding the universe of the classification system. The level of expressiveness is assimilated to the purpose of the classification system.	
14.	☒	The requirements for a classification system for the BHR are fulfilled.	BHR

1. Foundations & Related Fields
   1.1. Definitions, Theories, Models, Methods & Standards
   1.2. Historical Aspects
   1.3. Education & Training
   1.4. Related Fields & Disciplines
2. Knowledge Organization & Collection Management
   2.1. Knowledge Organization Systems
   2.2. Cataloging & Indexing
   2.3. Collection Development & Design
   2.4. Collection Acquisition & Evaluation
3. Memory Institutions & Information Infrastructure
   3.1. Types
   3.2. Networks & Cooperations
   3.3. Architecture & Technology
   3.4. Management
   3.5. Information Services
4. Information Society
   4.1. Information Literacy
   4.2. Social Participation
5. Human Information Behavior
   5.1. Information Use
   5.2. Information Seeking & Need
   5.3. Human-Computer Interaction & User Experience
6. Information & Research Data Management
   6.1. Scientific Publishing
   6.2. Open Science
   6.3. Data Curation & Preservation
   6.4. Informetrics & Science Studies
7. Information Systems & Information Processing
   7.1. Design, Implementation & Management of Information Systems
   7.2. Evaluation of Information Systems
   7.3. Internet Technologies & Services
   7.4. Automation, Data Mining & Artificial Intelligence

Figure 10: The Classification System for the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft

The classification system in Figure 10 is, just as JITA, a two-level mono-hierarchical classification system. It contains 7 superordinate and 26 subordinate classes. Consequently, it consists of in total 33 classes, which matches one of the requirements of the BHR (Table iv; criterion 14 fulfilled). It is possible to only assign documents to the main class which therefore also serves as 'Other' class. For the notation, Arabic numerals are used. Every level is separated with a dot between the numbers. This makes it easily possible to extend the KBHR without limitations to the number of classes or levels, if it is necessary in future (hence, criteria 3 and 10 are fulfilled). There is both, a German, and an English version of the KBHR. The multilingualism is supposed to make the BHR more accessible for international interested parties. Furthermore, it seemed appropriate since 4.8 % of the BHR are written in English and this share is likely to increase with two English speaking natives as IBI professors. For each category there is only one equivalent in the respective other language which makes the two versions identical. It should be noted that for the German version it was intended to use as many German equivalents as possible. However, since a few English terms ('Human-Computer Interaction', 'User Experience' and 'Data Mining') are used, even in a German context, they were not translated.

As reported in Chapter 4.1, the most subjects of the BHR are referring to Library Science and librarianship or libraries & information facilities. Thus, it was clear that a big emphasize of the classification system had to be put on those two fields. Consequently, classes 2 and 3

were created to cover most of those topics. With classes 4, 5, 6 and 7 it was then aimed to cover the research fields of the IBI chairs as best as possible in accordance with the given BHR data. Class 1 includes subjects that are related to research and LIS in general.

The order of the seven main classes was based on the following deliberation: The first class should represent the foundations of LIS (class 1) as this is also the knowledge a potential user must know or learn first to do research in the LIS field. Afterwards, the specific subjects of LIS should follow. Since Library Science is named before Information Science in the frequently employed abbreviation LIS and in the name of the IBI, the same order should be represented in the classification scheme (class 2 and 3, before class 4 to 7) even though it is difficult to perfectly separate those two fields. Within the scope of Library Science, once again, the fundamentals (class 2) should precede the practical use (class 3). Within the Information Science field, the order is supposed to resemble the same concept: Theoretical as well as social applications before more applied and less social applications. It was aimed to follow the same order concepts (foundations to applications, social to non-social) in the subclasses and that the semantic scope of the superordinate classes covers the semantic scope of the subordinate classes (criterion 4 and 8 fulfilled).

One of the quality checks was done by examining the distribution of a subgroup of the BHR dataset. The subgroup was created by picking 100 random documents from the BHR dataset. BHR Nr. 458 was added to this test set, because it is the bachelor thesis of the author of the thesis at hand (Köhler, 2020a). Thus, without reading the content, it could be assigned with 100 % certainty to classes in the KBHR. For the other 100 BHR in the set, only the title, the abstract and partially the table of contents were read to allocate them to classes. A minimum of one class and a maximum of three classes for the assignment were defined beforehand. The result is illustrated in Figure 11. The Library Science related classes, 2 and 3, are studded in comparison to the other classes (criterion 12 not fulfilled). This was expected based on the description in Chapter 4.1.

Figure 12 shows the codes used for the subjects of the BHR as introduced in Chapter 4.1. Most subjects (18 out of 28 subjects; 64,2 %) are fully covered by KBHR which is marked in yellow in the figure. ‘Actors’, ‘Institutions’, ‘Languages’, ‘Locations’, ‘Other’, ‘People’ and ‘Publication Types & Genres’ are either too specific or too broad to be covered by a class or they contain subjects that were used to describe metadata of the publication and thus can also not be covered by a classification scheme. The three gray underlined categories

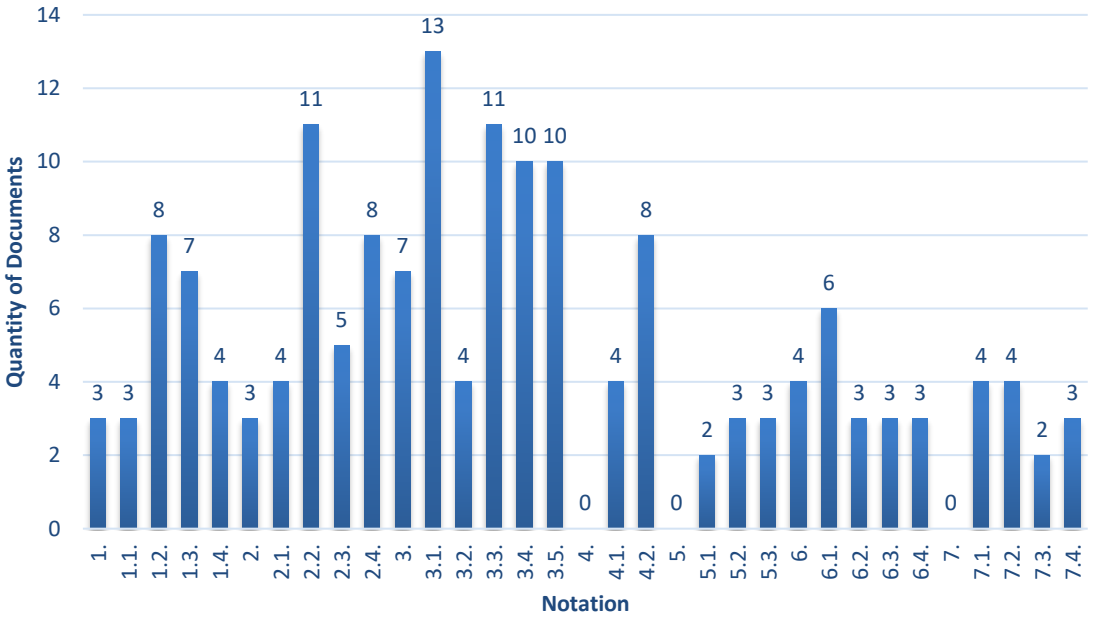


Figure 11: Test set, consisting of 101 BHR, assigned to the Classification System for the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft

- Actors
- Archival Science & Museology [2., 3., 6.]
- Education [1.3.]
- Foundations and cross-disciplinary topics of LIS [1.1., 1.4.]
- History [1.2.]
- Information Behavior [5.]
- Information Management [6.]
- Information Processing & Analytics [7., 7.3., 7.4.]
- Information Retrieval [7., 7.1., 7.2.]
- Information Systems [7.1., 7.2.]
- Institutions
- Internet & Technology [7.3.]
- Knowledge Organization Systems [2.1.]
- Languages
- Laws & Justice [1.4., 4., 6.1.]
- Libraries & Information Facilities [3.1.]
- Library Science & Librarianship [2., 3.]
- Locations
- Management, Marketing & Financing [3.4.]
- Methodology [1.1.]
- Other
- Other Disciplines & Related Fields [1.4.]
- People
- Publication Types & Genres
- Publishing [6.1.]
- Scientific Practices [1.1., 6.1., 6.2., 6.4.]
- Services and Software [3.3., 3.5.]
- User [3.5, 4.2., 5.]

subjects are represented by a class in in the KBHR

subjects that can be covered to a certain degree by a class in the KBHR

[ ] notation of a class in the KBHR that represents or kind of represents the concept

Figure 12: Groups of BHR subjects marked by their occurrence in the KBHR

(3 subjects; 10,7 %) are not specifically addressed in the BHR classification scheme but can be covered by some classes depending on the specific use case. Overall, 75,0 % of the BHR subject groups are addressed in the KBHR. This is 7,1 % or two subject groups more than JITA addressed. Furthermore, almost all the requests by the professors of the IBI are directly addressed in the proposed classification scheme as depicted in Figure 13 (criterion 1 fulfilled): 100,0 % of Information Behavior and Information Management; 83,3 % of Information Processing & Analytics; 92,9 % of Information Retrieval; and 66,7 % of

Information Science. The inquiries ‘Cultural Heritage Systems’, ‘Philosophy of Information’, ‘Information Ethics’, ‘Metadata’ and ‘Database Design’ are not represented by a specific class but can be allocated to the ones available. Therefore, 100,0 % of the requests are presented in the KBHR. This is also an improvement in comparison to JITA.

In terms of topicality, the KBHR was once more compared to the problems collected in the structured collection of Information Science subjects **(**Appendix II**)**. Since the collection is much bigger than the KBHR has capacities for, most subjects are only partially covered. This is because the KBHR is more general and does not go into detail and therefore only provides an overview (criteria 2 and 13 fulfilled). However, most of the topics are somewhat covered. Topics that miss representation for the most part are ’Legal, Ethical, Educational & Social Issues’ (apart from educational issues); information professions in the group of ‘Information Professions, Information Services & Applied IS’; ‘Information Industries, Economy & Management’; partly ‘Information Technology’ and ‘Others’. The structured collection in Appendix II is a portrayal of the consensus of Information Science experts about topics of Information Science. The universe of this collection is therefore the whole Information Science spectrum. The KBHR on the other hand aims to cover the LIS field with a focus on the representation of the topics of the BHR and the research fields of the chairs

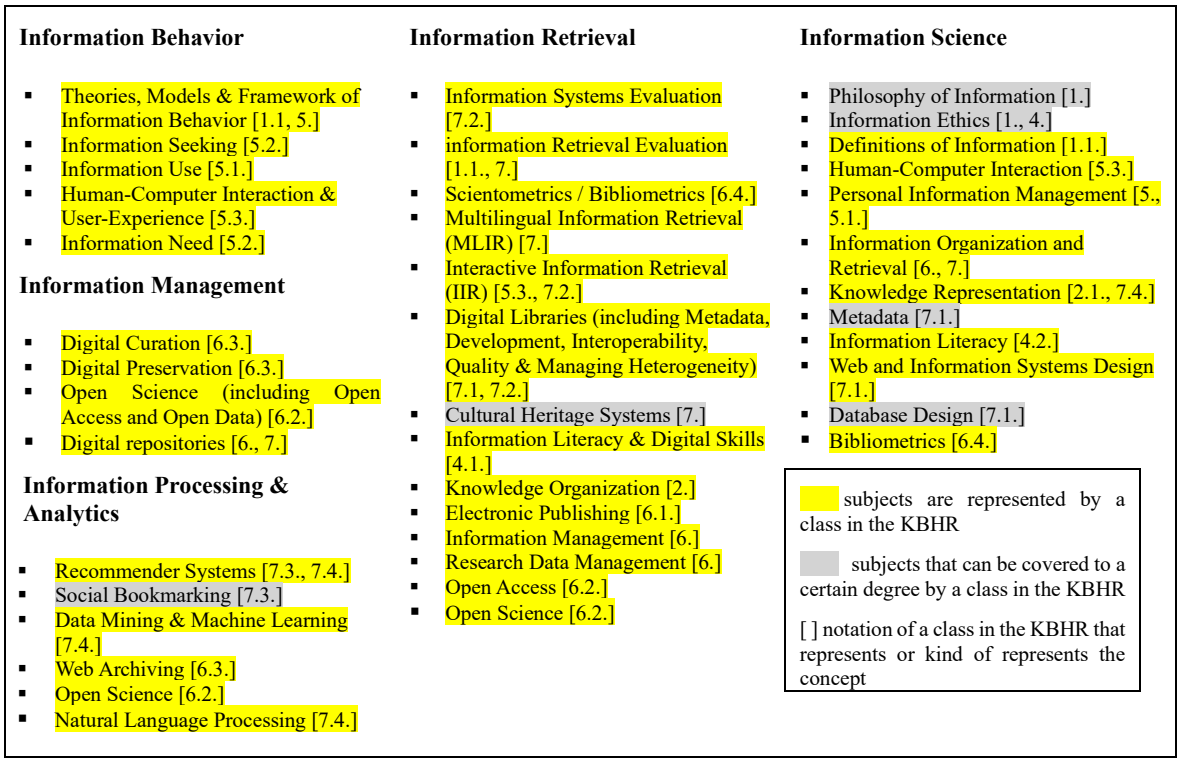


Figure 13: Requested research fields of the IBI professors and research assistants marked by their occurrence in the KBHR

of the IBI. The topics of the BHR and of the IBI chairs do not meet the mentioned missing topics. Moreover, in comparison with the most recent LIS topics in the literature (G. Liu & Yang, 2019; Ma & Lund, 2021; Figure 14) the KBHR can cover virtually all of the latest topics completely or partially (criteria 9 fulfilled): 27 out of 28 topics (96,4 %) of G. Liu and Yang (2019) and 10 out of 10 topics (100,0 %) by Ma and Lund (2021). Even though 8 out of those 27 covered topics (29,62 %) from G. Liu and Yang (2019) and 2 out of those 10 covered topics (20,0 %) by Ma and Lund (2021) are only partially covered.

A criterion that is only partly fulfilled is criterion 7. Research topics of the chair of Information Retrieval and the chair of Information Processing & Analytics had to be joined to save space (class 7). It would have been better, in terms of elegance, to create a whole class for each of them. However, this would not have been in accordance with the BHR documents and it would have cost more space. Nonetheless, the topics of those two chairs are related and hence, it is possible to combine them.

Criterion 11 is evaluated as partly fulfilled as well. Even though, the KBHR is very general and thus does not include unnecessary detail, it combines several topics under one class name (e.g., '1.1. Definitions, Theories, Models, Methods & Standards'). Since this is not very elegant, criterion 11 was judged as partly fulfilled.

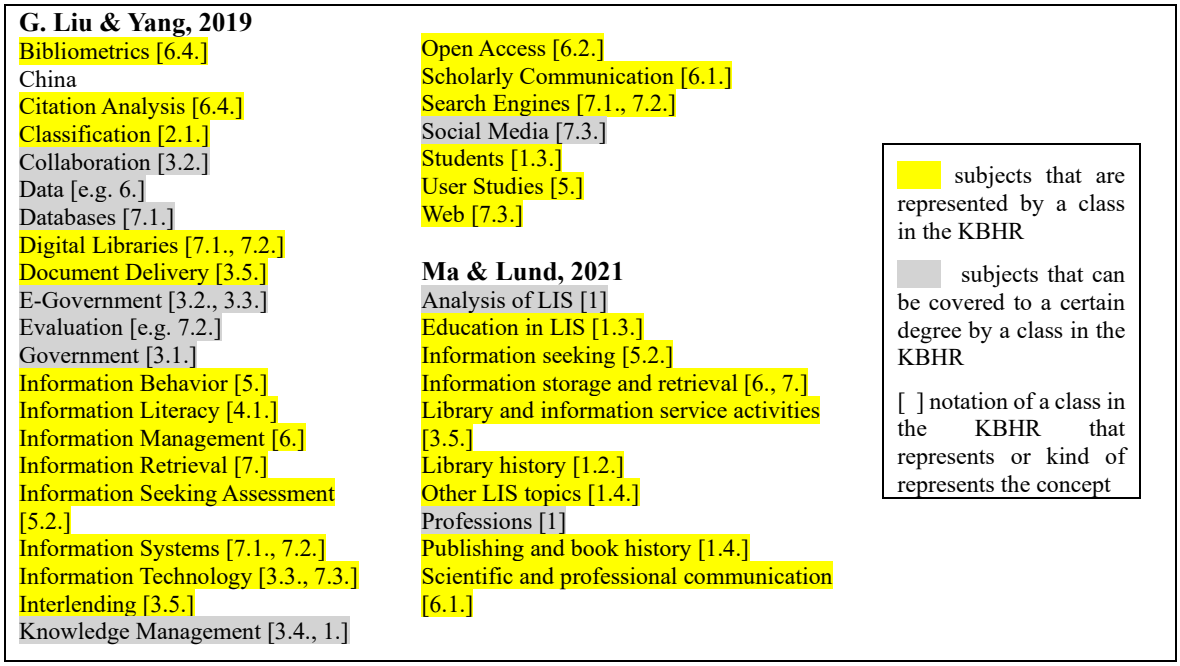


Figure 14: The most recent LIS research topics according to G. Liu and Yang (2019) and Ma and Lund (2021) marked by their occurrence in the KBHR

Table vii: Filled in evaluation checklist of Classification System for the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft

		Feature	
1.	☑	The classification corresponds with the research field and the probable needs of the users.	Use
2.	☑	The classification system is understandable to the users and must provide an overview of the domain or any kind of new knowledge.	
3.	☑	The classification has a hierarchical, easy-to-remember notation and is memorable itself.	
4.	☑	The semantic scope of the broader term must cover the semantic scope of its narrower terms.	Hierarchy
5.	☒	The narrower classes must be disjoint.	
6.	☒	No jumps in the classification hierarchy.	
7.	☑	Items are well placed in the hierarchy.	
8.	☑	The order of subordinate classes should follow principles. If subordinate classes appear in more than one main class, their order should be the same or similar in all those main classes.	
9.	☑	The classification and its contents must be up to date.	Time Aspect
10.	☑	It should be possible to extend the classification in future so that it can be used for a longer period.	
11.	☑	The classification system is focused on clarity and seems elegant by avoiding unnecessary details. That means only sufficient information should be included. As few assertions about the modelled domain as possible should be made.	Other
12.	☒	All classes contain around the same number of items (for small datasets 15-30 items).	
13.	☑	Classes are formulated and defined as broad and clear as possible and in an objective manner. They should be exhaustive regarding the universe of the classification system. The level of expressiveness is assimilated to the purpose of the classification system.	
14.	☑	The requirements for a classification system for the BHR are fulfilled.	BHR

Criteria 5 and 6 are not fulfilled. ‘6.1. Scientific Publishing’ can correlate with ‘6.2. Open Science’ and ‘3.3. Architecture & Technology’ can overlap with ‘3.5. Information Services’. Even though criterion 5 is violated by this design, it seemed logical to keep the classes like that, because they either represent big research fields on their own or are represented in the BHR data a lot. Since multi-labeling is allowed for the KBHR and Information Science is a highly interdisciplinary field, harming criterion 5 was judged as an acceptable tradeoff.

Criterion 6 was harmed, because it was tried to incorporate as many research fields as possible within the least number of classes and furthermore, because most BHR subjects should be respected. The criterion was violated in class ‘3. Memory Institutions & Information Infrastructure’ with the subclasses ‘3.1. Types’ and ‘3.5. Information Services’. These are very different concepts but still appear in the same class. It was tried to reduce this effect through the positioning of classes, but it cannot change, that criterion 6 is not fulfilled.

As for JITA, the evaluation checklist was filled out for the KBHR as well. In Table vii, the evaluation checklist from Table iii is filled out by marking the criteria with a checkmark if the criterion was met or with a X if it was not met. If a criterion was only partially fulfilled, it was marked with a checkmark with dashed lines. Out of 14 Features, 3 were marked as not fulfilled while the leftover 11 were documented as fulfilled or rather fulfilled. Out of these 11, 2 criteria were not fulfilled to complete satisfaction. Hence, 9 criteria are fully satisfactory. Consequently, more criteria are fulfilled to complete satisfaction with the KBHR than with JITA.

6.3 Training & Tests on the Scraped Training Set

Based on the macro-average F1 scores, a variation of the Naïve Bayes classifiers and the Logistic Regression classifier yielded the best results. Both showed a macro average F1 score of 0.23. In the following, the outputs of different variations of the Naïve Bayes, Logistic Regression, SVM, and FLAIR classifiers will be reported. The results will be discussed in Chapter 7.2.2.

6.3.1 Naïve Bayes

The Naïve Bayes classifier was the quickest classifier and required 2.26 hours to run all possible models and 0.71 hours (42.87 minutes) for the best models. Almost all the best models had an alpha value of 0.3 (21 models; 80.8 % of the best models), except for three models with an alpha value of 0.0 (11.5 % of the best models) and two models with an alpha value of 0.2 (7.7 % of the best models). Thus, Laplace smoothing was adopted for most models, but small values only. Eleven models achieved the best results using lemmatization (42.3 % of the models), nine with stemming (34.6 % of the models), and six models used neither of these two preprocessing options (23.1 % of the models). The first tests were done using chi-square FS. Even though the average accuracy over all best models is 0.82, the macro-averaged F1 score is only 0.18. All models had an F1 score below 0.36, a precision

value below 0.24 and a recall value below 0.89. The macro-averaged precision and recall over all best models are 0.11 (precision) and 0.58 (recall).

A second test without FS was applied to investigate if the classification improves. All remaining hyperparameters remained unaltered. The runtime for the best models was reduced to 0.49 hours (29.65 minutes) and 2.07 hours for all models. For the best models without FS the hyperparameters changed: Most models had an alpha value of 0.2 (15 models; 57.7 % of the best models), followed by models with an alpha value of 0.0 (10 models; 38.5 % of the best models) and one model with an alpha value of 0.3 (3.8 % of the best models). This time, the preprocessing option without stemming or lemmatization was applied the most (14 models; 53.8 % of the best models), followed by seven models with stemming (26.9 % of the best models) and five models using lemmatization (19.2 % of the best models). The average accuracy, improved to 0.93 as wekk as for the macro averaged F1 score to 0.22 and the macro average precision to 0.21. The macro-average recall however decreased to 0.26.

Because both variations of the Naïve Bayes classifier outputted several errors that stop words from the stop word list are not represented in the data, the algorithm was altered and run again with and without FS: This time the stop word list was also lemmatized or stemmed manually if the documents were as well. The macro average F1 score for Naïve Bayes with FS did not change (0.18) and for Naïve Bayes without FS it improved slightly from 0.22 to 0.23. The macro average recall decreased from 0.58 to 0.57 for Naïve Bayes with FS and increased from 0.26 to 0.27 for Naïve Bayes without FS. The macro average precision and the average accuracy values remained the same. The calculations required more runtime: For the variant with FS, all models took in total 2.98 hours to train and 2.97 hours without FS. For the best models the runtime was in total 0.91 hours (54.66 minutes) for the classifier with FS and 0.79 hours (47.63 minutes) without. For the best models with FS the applied preprocessing methods and alpha values remained mostly the same. Only the model for KBHR class 4.2. used an alpha value of 0.3 instead of 0.2 and the model for KBHR class 3.2. did not employ stemming or lemmatization anymore. For the best models without FS, the alpha values remained the same for every model. However, the preprocessing option of the model for KBHR class 1.3. changed from using neither stemming nor lemmatization to applying lemmatization and for the model of KBHR class 6.3. from using neither stemming nor lemmatization to applying stemming.

Comparing the macro average F1 scores of the four variations of the Naïve Bayes classifier (no altered stop word list and FS, no altered stop word list and no FS, altered stop word list and FS, altered stop word list and no FS), the variations without FS performed better than the ones with FS and the Naïve Bayes classifier without FS and with altered stop word list yielded the best F1 score of 0.23.

6.3.2 SVM

The LinearSVC (with FS) had a runtime of 0.88 hours (52.92 minutes) for the best models and 36.52 hours for all models. In a first experiment the FS method, chi-square, was applied. Most of the best models had a regularization value of 500 (10 models; 38.5 % of the best models), followed by the value 150 (7 models; 26.9 % of the best models) and 100 (4 models; 15.4 % of the best models). Three models (11.5 % of the best models) employed a regularization value of 1,000, one model with a regularization of 60 and one model with 30 (3.8 % of the best models each). The maximum number of iterations also varied between the best models: A value of 200; 300, and 1,000 was used by six models each (23.1 % of the best models). A value of 400; 1,500 or 2,000 was used by two models each (7.7 % of the best models). Finally, a maximum number of 500 or 3,000 was applied by one model each (3.8 % of the best models). Stemming was used by half of the models (13 models; 50.0 %), followed by the models using lemmatization (8 models; 30.8 %) and by the models that did not apply either of these (5 models; 19.2 %). The average accuracy was 0.95, the macro average F1 score 0.19, the macro average precision 0.28, and the macro average recall 0.17.

The LinearSVC was also run again without FS. The training times reduced to 0.08 hours (5.00 minutes) for the best models and 33.12 hours for all models. A broad range of regularization values were applied in the different models: Five models each (19.2 % of the best models) applied a value of 0.8 or 1.0; three models each (11.5 % of the best models) a value of 3.0 or 10.0; two models each (7.7 % of the best models) a value of 0.1, 0.3 or 5.0; and one model each (3.8 % of the best models) a value of 0.15, 0.2, 0.5 or 100.0. Almost all best models (24 models; 92.3 % of the best models) used a maximum number of iterations of 200 and the two remaining best models (7.7 % of the best models each) a maximum number of 300. One of the best models (7.7 % of the best models) employed lemmatization, the rest (25 models; 96.2 % of the best models) used neither lemmatization nor stemming. Average accuracy as well as macro average F1 score and precision improved to the following

values: 0.96 (accuracy), 0.20 (F1 score) and 0.44 (precision). The macro average recall value decreased to 0.15.

The same stop word list errors occurred for the SVM as they did for Naïve Bayes. Therefore, both variants were run again with adapted stop word lists: Based on the macro-average F1 score these changes did not improve the final classification result as the macro-average F1 score remained at 0.19 for the SVM with FS and at 0.20 without FS. The average accuracy did not change for the SVM without FS either (0.96) and improved slightly for the SVM with FS from 0.95 to 0.96. For the SVM without FS, the macro-average precision (0.44) and recall (0.15) values remained the same as well (in comparison to the SVM without FS and unchanged stop word list). For the SVM with FS, they changed slightly with a macro-average precision of 0.29 (0.01 higher than without changed stop word list) and a macro-average recall of 0.15 (0.02 less than without changed stop word list). The individual hyperparameters showed more alterations:

For the SVM with changed stop word list and FS, the applied regularization values ranged from 0.1 to 1,000. The regularization that was used the most was 500 (8 models, 30.8 % of the best models), followed by 1,000 (6 models; 23.1 % of the best models) and 100 (5 models; 19,2 % of the best models). The applied maximum number of iterations ranged from 200 to 3,000. Most models employed a maximum number of 200 iterations (5 models; 19.2 % of the best models), followed by 400 and 1,500 (4 models each; 15.4 % of the best models each). The preprocessing combined with lemmatization was applied in eleven models (42.3 % of the best models). Preprocessing combined with stemming was used in seven models (26.9 % of the best models) and only stop word removal and lowercase conversion were applied in eight models (30.8 % of the best models).

For the SVM with changed stop word list and without FS, the applied regularization ranged from 0.1 to 100. Most often the regularization value was 0.8 or 1.0 (5 models each; 19.2 % of the best models each) and 3.0 or 10.0 (3 models each; 11.5 % of the best models each). The applied maximum number of iterations remained the same as for the SVM with unchanged stop word list: 24 models (92.3 % of the best models) employed a maximum number of 200 iterations and 2 models (7.7 % of the best models) a value of 300. The applied preprocessing options also remained unchanged: Almost all models (25 models; 96.2 %) neither applied lemmatization or stemming and only one model (3.8 %; the model for KBHR class 6.3.) used lemmatization.

6.3.3 Logistic Regression

The Logistic Regression classifier took the longest to run all calculations. The variation with FS and without altered stop word list required 1.75 hours (105.59 min) for calculating the best models. The total run time was lost, because the code had to be restarted several times due to server issues. However, it was running for roughly six weeks. To test other variations of the Logistic Regression classifier was not feasible within the scope of this research. After running for roughly six weeks, the Logistic Regression classifier without FS finished 148 of 612 models (24.2 %). Finishing the calculations would probably have required more than 18 weeks. Therefore, only the variation of Logistic Regression with FS and without altered stop word list will be analyzed.

Many of the best models (12 models; 46.2 %) applied a maximum of 100 iterations, followed by 150, 200, and 750 maximum iterations for three models each (11.5 % of the best models). The inverse of regularization strength was more broadly distributed. Together, 30 (6 models; 23.1 % of the best models), 150 (5 models, 19.2 % of the best models), and 500 (5 models, 19.2 % of the best models) were applied the most for the inverse of regularization strength. The three different preprocessing options were employed almost equally often: Nine models each (34.6 % of the best models each) applied either only stop word removal and lowercase conversion or stop word removal, lowercase conversion, and stemming. Stop word removal, lowercase conversion, and lemmatization was used by eight models (30.8 % of the best models).

The average accuracy of the Logistic Regression classifier with FS was 0.96. The macro average F1 score was 0.23. It is influenced by the precession and recall values. The macro average precision was 0.31 and the macro average recall 0.19.

6.3.4 FLAIR

FLAIR was tested with different learning rates. All FLAIR experiments required a runtime between 1.74 days (41.96 hours; a learning rate of 0.03 and a maximum of 500 epochs) and 3.99 days (95.95 hours; a learning rate of 20 and a maximum of 150 epochs). All according models showed zero values for most class labels and evaluation metrics. The model with the least zero values had a learning rate of 0.75. KBHR class 1.1., 2.4., 7.1. and 7.3. were assigned values with a value of 150 for maximum number of epochs. For all those classes

the recall was 1.00. In contrast, their precision values were 0.07 (class 1.1.), 0.05 (class 2.4.), 0.06 (class 7.1.) and 0.08 (class 7.4.). Accordingly, their F1 scores were 0.13 (class 1.1.), 0.10 (class 2.4.), 0.12 (class 7.1.) and 0.15 (class 7.4.). The macro average F1 score over all best models was 0.02, while precision and recall had a macro average value of 0.01 (precision) and 0.15 (recall). For a value of 500 for maximum number of epochs and the same learning rate, the models for KBHR classes 6.1., 7.1., 7.2. and 7.3. showed values higher than zero. These models yielded a recall value of 1.00 each as well. The precision values were 0.07 (class 6.1.), 0.06 (classes 7.1. and 7.2.), and 0.08 (class 7.4.). The F1 scores illustrates these values accordingly with scores of 0.14 (class 6.1.), 0.12 (classes 7.1. and 7.2.), and 0.15 (class 7.4.). The macro average F1 score, precision and recall value were the same as for a learning rate of 0.75 with a maximum number of epochs of 150.

Even though other tested learning rates yielded less models that had values higher than 0.00, different KBHR classes were covered by these models than the models with a learning rate of 0.75: For a learning rate of 0.03 and a maximum number of 150 epochs, for KBHR class 3.1. other values than null were outputted by the classifier (0.05 F1 score, 1.00 recall, 0.03 precision). For this learning rate, another experiment with a maximum number of epochs of 500 was conducted. The results remained unchanged, however. The following learning rates were only tested with a maximum number of 150 epochs: The learning rate 0.85 yields results for KBHR classes 6.1. (0.14 F1 score, 1.00 recall, 0.07 precision), 7.3. (0.15 F1 score, 1.00 recall, 0.08 precision) and 7.4. (0.15 F1 score, 1.00 recall, 0.08 precision). All values for the model with a learning rate of 20 are the same as for the model with a learning of 0.85, except for KBHR class 6.1. which yielded zero values.

A final experiment was conducted with the attempt to make the problem less complex: The problem was reduced to forwarding only one label for each document to the FLAIR classifier. As a learning rate, 0.07 was chosen and a maximum of 500 epochs. Because only one class model yielded results higher than zero, the macro average F1 score was only 0.04, while the macro average precision was 0.00 and the macro average recall was 0.04. The only model that showed results higher than zero was the model for KBHR class 7.4. (0.15 F1 score, 1.00 recall, 0.08 precision).

Because the test results of all classifiers were not promising, no experiments automatically classifying the BHR were conducted. The findings and the limitations of this research will be discussed in Chapter 7.2.

7 Discussion

7.1 Evaluation of JITA and Creation of a New Classification System

One objective of this thesis was to obtain a classification system for the BHR. To do so, one possibility was the reuse of JITA. The research question, if JITA is an appropriate classification system for the BHR, had to be answered. In Chapter 6.1 this question was investigated by using a checklist consisting of criteria for a good classification system as mentioned in different sources. Furthermore, the requirements of a classification system for the BHR were compared to JITA. After a thorough evaluation, it was concluded that JITA cannot be reused for the BHR collection. Instead, a new classification scheme was created considering the requirements of the BHR and the criteria of the evaluation checklist. Based on the latter, the new classification system was more suitable for the BHR than JITA.

As described in Chapter 3.1, there is only limited current literature about the evaluation of classification systems. Even though people face hierarchical structures every day on the internet and offline, there are only limited up-to-date, universal guidelines on how to evaluate them. The common suggestion for quality checks in the literature, are expert evaluations (Golub et al., 2016; Maedche & Staab, 2002; Stoica et al., 2007) or user studies (Golub et al., 2016; Hall et al., 2014). However, there are many scenarios in which neither experts nor the resources to conduct user studies are available. Since this was also the case for the study at hand, another approach had to be found. The collected criteria from the literature, as summarized in Table iii, are a helpful tool that is universal and thus can be used in any future project with the need to evaluate a classification scheme. For criterion 14, instead of the BHR requirements the requirements of the institution in question should be used, however. The checklist does not depend on the scope or subject of the classification system that needs to be evaluated. Nonetheless, if possible, it is recommended to consult an expert and conduct user studies as well. The fact that this was not feasible within the scope of the study at hand is a clear limitation of this thesis.

The criteria should rather be seen as guidelines than as mandatory rules. E.g., Umlauf (1999) states that all classes should contain around the same number of items. In most cases, this will not be possible, and the underlying data will usually be unevenly distributed. Yet, a tradeoff of practicability and ideal should always be the goal.

The level of importance of the individual criteria can vary depending on the evaluator and the cause for evaluation. It is a drawback of the evaluation checklist, that all criteria appear to be equally important, and it is difficult to stress the importance of some criteria more than others. One of the most important aspects that should be kept in mind when evaluating or creating a new classification scheme is that it should be designed for durability but at the same time represent the current state of the domain well. These two criteria are seperated by a fine line, especially since classification systems are rather inflexible hierarchical structures (Manecke, 2004): For the KBHR, it was tried to keep that rule. Therefore, even though most BHR are written about Library Science topics, the research fields of the chairs are also included. It was challenging to find categories that represent the IBI research fields, but at the same time embody the data. Other researchers might handle the issue differently and thus would obtain a different classification system. In the end, it is not possible to find a perfect classification system (Kwaśnik, 2021), but the needs of the users should be prioritized.

Users of the KBHR will be LIS interested parties, but also especially students and employees of the IBI. It was assumed that they expect to find a representation of the IBI chairs in the classification system (but no scientifically founded statement can be made without appropriate studies). It is a political decision to include all chairs in the final classification system even though there are not enough publications in the BHR yet that represents them. Representation and inclusiveness are influencing factors when creating a classification system and should not be underestimated. As political is the decision that must be made about the order of different research fields. This issue was also addressed by Kwaśnik (2021), Hjørland (2013) and Zins (2007b). These decisions are made by the person creating the classification and thus incorporate personal interpretations and views and cannot be neutral (Hjørland, 2013; Zins, 2007b). Yet, including all chairs and hence more LIS fields also serves the purpose to portray more knowledge about LIS research and puts the fields into context. To give an overview about the contents and their relationship to each other is one advantage of a classification system (Fernando et al., 2012; Gantert, 2016; Hall et al., 2014; Kwaśnik, 2021). However, due to a lack of respective BHR documents, some chairs had to be merged to one main class (Information Retrieval and Information Processing and Analytics) and the main class names are not equal to the chair names. This is another indication that it is not possible to include all aspects, fields, and concepts equally well (Kwaśnik, 2021). Therefore, a careful decision must be made about what to in- and what to exclude, since this will also influence the way, users will see the research structures at IBI.

Especially in the LIS field, sacrifices must be made due to its interdisciplinary background (Borko, 1968; Luft, 2015). In Zins (2007a), he collected 28 different classification schemes for Information Science, which shows that organizing LIS fields is a highly controversial topic. In the same paper, a comment of Maria Pinto says:

> *"It is almost impossible to elaborate an Information Science tree with clearly defined branches, because Information Science, as many others [sic] fields, does not have a tree structure, but rather a network structure. Therefore, dependencies and overlapping are an essential constituent of this multi-paradigmatic domain." (Zins, 2007a, p. 665)*

The author of the study at hand agrees with the statement – it reflects the issue stated above. Due to this interdisciplinarity, the classes in the KBHR are not strictly disjunct as usually demanded of classification systems (Lorenz, 2018), but the KBHR is a classification system with superimpositions as defined by Gaus (2005).

It is likely that the contents of the BHR documents change slightly in the next years. Because a literary warrant was followed, changes in the collection will also influence the classification system (Kwaśnik, 2021). E.g., philosophy is not a topic that is addressed often within the BHR publications and therefore had to be left out. This might change with further developments of the chair of Information Science under Professor Jesse Dineen and is also an often named topic in the classification schemes collected in Zins (2007a; Appendix II). The notation of the KBHR makes it easy to add classes. Yet, simply adding classes to the end of a list is likely to destroy the order principle of the classification. Since all documents exist in a digital format however, the effort of changing the notation and thus the location of the documents should be tolerable. Nonetheless, changes that could lead to confusions of the users should be avoided, unless it is truly necessary. A question that remains is who has the right to make adaptions to the classification system (Kwaśnik, 2021).

Regarding the availability of the KBHR in English and German, it is a nice feature to offer the classification system in two languages, but it also entails risks. Due to political and cultural differences and differences in the meaning of research terms, the understanding and quality of a classification scheme could change (Fraunhofer ISST &]init[, 2009; IFLA, 2009). Yet, it is more inclusive and therefore it was decided to translate the KBHR in two languages. To avoid confusion and make manual classification easier, in future, class

descriptions should be added for each KBHR class (Bedford, 2013). Missing descriptions also made it more difficult to evaluate JITA or to create the collection of Information Science topics out of Zins (2007a) collection of classification systems.

A final limitation of this part of the thesis is the evaluation of the KBHR. A classification system should not be judged by its own creator, because they designed it to the best of their abilities and tend to overlook their mistakes. Therefore, it is strongly recommended to let other experts check the quality of the classification scheme and to conduct user studies. Other than with the editor of the BHR, Professor Dr. Vivien Petras, no other sanity check was implemented and the evaluation of the KBHR was done by the author of this thesis herself. Hence, especially criteria 2, 6 and 7 are difficult to assess for the author. In future studies, this step should be performed by an independent party with potentially other standards, world views and values.

7.2 Automatic Text Classification

7.2.1 Creation of a Training Set

Since the KBHR was newly created, there was no BHR training set that could have been used for automatic classification. Because automatic classification saves resources (Labrador et al., 2020; Sharma et al., 2018), it was to be evaluated if automatic classification via ML algorithms can be applied. For ML tasks, a training set is necessary (Aggarwal & Zhai, 2014). Therefore, a training set had to be created. The underlying research question was whether TC on the BHR documents without a BHR training set is possible. This was tested by using different classification algorithms.

The insights of the literature reveal that DL mostly yields better results than ML (Akhter et al., 2020; Behera et al., 2019; Zheng & Zheng, 2019), but also requires more data (Behera et al., 2019; Kowsari et al., 2019; P. Liu et al., 2016). To obtain a big training set, titles and abstracts of documents were scraped from four different databases. Since there are access restrictions, only the public available titles and abstracts were scraped from the websites. In future studies, it could be of interest to use full-text documents or to additionally crawl subject metadata for the TC.

Originally, it was intended to scrape data from *Google Scholar*, *Library and Information Science Abstracts* and *DeGruyter* as well. This was not possible, however, due to messy data,

individual query IDs that could not be recreated for scraping or complicated query structures. Instead, e-LiS, DABI, Springer and o-bib were chosen as data sources. This came with limitations: DABI retrieved a maximum of 100 results for each category of its classification system[22], JITA's and o-bib's documents sometimes did not have an abstract in the language of the document[23] or no abstract at all[24] and Springer, among others, retrieves unsuitable documents even though the queries were formulated as specific as possible. For example, the Springer query "(("Knowledge Organization System*" OR "Knowledge Organization" OR "Ontolog*" OR ("Classification system" OR "classification scheme") OR "Thesaur*" OR "Semantic Net*" OR "Taxonom*") AND ("LIS" OR "Library Science" OR "Information Science" OR "Library and Information Science"))" for KBHR class '2.1. Knowledge Organization Systems' also retrieved an article with the title "Identifying the research focus of Library and Information Science institutions in China with institution-specific keywords" (Guo Chen et al., 2015). Furthermore, some documents had abstracts with less than 150 characters and were therefore not employed for further processing.

Another possible limitation could be the way the queries and mappings were designed. They were written to the best of the author's knowledge. Yet, maybe other queries would have yielded better or more results. It is surprising that in the scraped dataset, classes like '4.1. Information Literacy' and '6.3. Data Curation & Preservation' only hold very little data points in comparison to the other classes, even though they are major research topics in the LIS field (G. Liu & Yang, 2019; Ma & Lund, 2021). This could be a result of poorly formulated queries. The mappings could contain flaws as well: Sometimes classes from one classification system only partially overlapped with classes from the other. E.g., JITA's class 'IK. Design, development, implementation and maintenance' was mapped to '7.1. Design, Implementation & Management of Information Systems' even though it is not completely clear what class IK is referring to. Its superordinate class is called 'I. Information treatment for information services'. Since no additional information for the classes and subclasses in JITA was given, it was hard to interpret them.

The quality of the data was probably further reduced through the translation process. Even though, the Google Translator was rated as the best translator by Vanjani and Aiken (2020), Ziganshina et al. (2021), and Zulfiqar et al. (2018), every translation can only be equally

[22] E.g.: http://dabi.ib.hu-berlin.de/cgi-bin/dabi/suche.pl?modus=html&notation=3a2a2 [September 25, 2021]
[23] E.g.: http://eprints.rclis.org/17824/ [September 25, 2021]
[24] E.g.: https://www.o-bib.de/article/view/5599 [September 25, 2021]

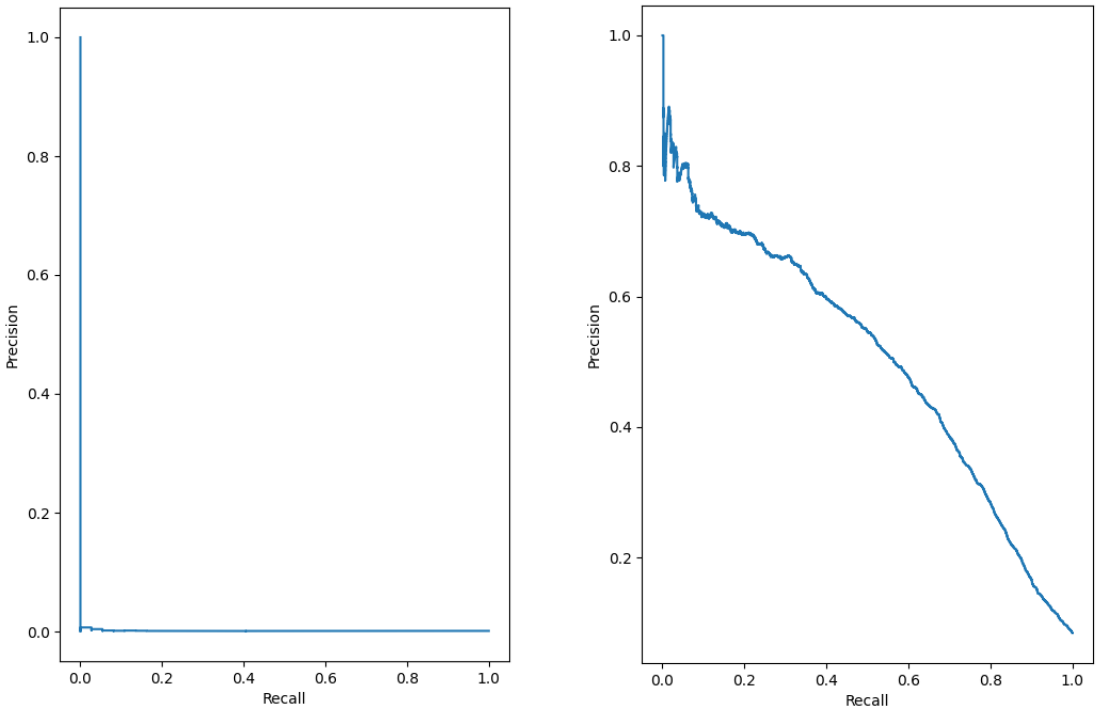
good or worse than the original. E.g., the German title ‘Das Buch und sein Haus […]’ (id ab131edd6dec21a48dedece324d92408 in Abstract IV) was translated to ‘The book and his house […]’ (Abstract IV). The correct translation would be ‘The book and its house’. It is only a minor mistake that is also context dependent, but it potentially changes the features in the feature vector and thus has an influence on the classification. Moreover, the German dataset is expected to be of worse quality than the English dataset because more documents (84.94 % of the documents in the original dataset) had to be translated. Yet, Zulfiqar et al. (2018) confirm that Google Translate is a reliable tool for translating at least the central ideas of German scientific texts and Ziganshina et al. (2021) come to the same conclusion for English to Russian translations. Hence, the translation step in the study at hand could be sufficient for the purpose of classification, but it is likely that the classifiers yielded poorer results than they would have with a manually translated dataset. Finally, another disputable aspect of the translation step was that the title and abstract were merged to one text. Since titles often compress the topic of a document into one sentence, they might hold more information than the abstract and should be kept separate. This needs to be tested in future studies.

7.2.2 Interpretation of the Classification Results for each Classifier

The results show that without an appropriate BHR training set, no classification is possible. All the classifiers yielded in no way sufficient results. Since the ML algorithms created one model for each KBHR class, there are differences not only between the classifiers and their individual settings, but also between the individual classes. To evaluate every single model would therefore go beyond the scope of this thesis. However, in general it can be said that none of the achieved models are suitable for an automatic classification of the BHR. In the following the most markable results will be elaborated.

7.2.2.1 Naïve Bayes

In Figure 15 the precision-recall-curve for the model of the KBHR class with one of the worst performances and the model with one of the best performances of the Naïve Bayes classifier (with FS, without adapted stop word list) are confronted. The AUC for the model of KBHR class 2.3. (Figure 15, left side) is almost non-existing which means that it is not skillful and the best model of KBHR class 7.4 shows a curve (Figure 15, right side) that almost resembles a diagonal**,** meaning it is slightly better than random guessing, but far from



KBHR Class 2.3. KBHR Class 7.4.

Figure 15: The precision-recall-curve of the model with the worst and the best performance of the Naïve Bayes classifier with FS and without adapted stop word list

the optimum**.** Thus, all the models of this classifier are not sufficient at all. KBHR class 2.3. is one of the classes with the least documents while class 7.4. is the class with the highest number of documents (Figure 5). Even though the models for both classes are not skillful, the results suggest that more information is needed to improve classification. To increase the features, in a second experiment FS was taken out of the ML pipeline. The remaining hyperparameters were unaltered. The macro average F1 score improved slightly from 0.18 to 0.22. This is because the macro average precision almost doubled from 0.11 to 0.22, but the macro average

recall value suffered extensively and was reduced from 0.58 to 0.26. This means that the classifier became more selective in a sense that it is more often correct when it predicts that a document belongs to the class in question, but also assigns less documents in general to that class. Thus, FS might have excluded important features for the TC from the feature vector. The changes in the stop word list influenced the classification results only slightly.

In comparison to the other classifiers Naïve Bayes had overall a higher recall, but lower precision values. Hence, Naïve Bayes allocated more documents that belong to a class correctly than the other classifiers did. Yet, at the same time, many of the documents that the

classifier assigned to a class were wrongly classified and in reality did not belong to the class.

Another change to the classifiers was made by adapting the stop word list through lemmatization or stemming if these methods were applied on the documents as well. This had only a slight influence on the classification results. Therefore, it can be assumed that the stop word removal did not have a big impact on the TC. Nonetheless, overall, the best classifier among the Naïve Bayes classifiers as well as among the other classifiers was the Naïve Bayes classifier without FS and with an adapted stop word list, based on its macro average F1 score of 0.23. All F1 scores for each model and classifier are listed in in the excel file 'Summary_Outputs.xlsx' in Appendix VI. The best F1 scores for each row are written in bold type and underlined. Through this table it becomes apparent that the Naïve Bayes classifiers did not result in the best models for each KBHR class, but sometimes the SVM or the Logistic Regression models performed better.

7.2.2.2 SVM

The accuracy and precision were better for the SVM classifiers than for Naïve Bayes with FS: The average accuracy of the SVM with FS and without adaption of the stop word list was 0.95 (a difference of 0.13 to Naïve Bayes with FS and without adaption of the stop word list) and the macro-averaged precision was 0.28 (a difference of 0.17 to Naïve Bayes). The highest precision value was achieved with the model for KBHR class 6.4. with a value of 0.73. However, the model for the same class yielded only a recall value of 0.4. This means that the model is very selective in the sense that it is often correct when it classifies a document to class 6.4. but makes this prediction often even if it would not have been the right choice. This effect increased when FS was removed for class 6.4. and most other classes. Overall, for the best models, the macro average precision increased a lot after running the experiments again without FS and changed from 0.28 to 0.44. The macro average recall decreased from 0.17 to 0.15.

With adapted stop word lists, the overall macro average F1 scores of the SVM with and without FS did not change. Neither did the macro average precision nor the macro average recall value for the SVM without FS, which yielded slightly better results than the SVM with

FS. Hence, it seems like the stop words did not have a major influence on the calculations of the SVM either.

In general, the macro average recall value of the best models for the LinearSVC classifier (0.17; no altered stop word list, with FS) was much worse than of Naïve Bayes (0.58; no altered stop word list, with FS) with a difference of 0.41. This discrepancy is also noticeable in the macro F1 score (0.19) that is almost the same as for the Naïve Bayes classifier (0.18), even though LinearSVC had much better precision values. Overall, the SVM with FS performed a little bit worse than Naïve Bayes with FS based on the comparison of the F1 scores.

If the recall values of Naïve Bayes and the precision values of SVM would have been satisfactorily high, the models of the two classifiers could have been used in combination: In a first step, the documents would be given to the Naïve Bayes classifier that yielded good recall results, but bad precision values. In a second step, the documents that Naïve Bayes predicted as belonging to a class, would be taken as input of the SVM classifier with good precision, but bad recall values. The final output would be the output of the SVM. However, the results were not efficient enough for those experiments.

7.2.2.3 Logistic Regression

Surprisingly, Logistic Regression required the longest runtime even though it was chosen because it was expected to require very little runtime. It is not clear what caused the long calculation times. It is possible that removing FS would have improved the macro average F1 score of 0.23 as it did for the Naïve Bayes classifiers. This should be tested in future studies with more time resources as well as more variations of the classifier.

As the SVM classifiers, Logistic Regression yielded better precision (macro average precision 0.31) than recall (macro average recall 0.19). However, the macro average precision of the SVM classifiers without FS with 0.44 (for altered and non-altered stop word list) are higher than the results of Logistic Regression. Yet, the F1 score for the Logistic Regression classifier is higher (0.23) than for the SVM classifiers (0.20), because their macro average recall with 0.15 is lower than the one of Logistic Regression.

In Table viii the classifiers of the best models for each KBHR class and their F1 scores are listed. The best Logistic regression models for a few KBHR classes, like 1.3. or 2.2., showed the best F1 scores among all the classifiers. For other classes' models other classifiers yielded better results. In comparison with the number of documents per class (Figure 5), no pattern could be recognized when a classifier performed better than the others. Using the individual best variation of preprocessing methods, hyperparameters, and classifiers for each KBHR class would result in the best possible classification result. However, not a single model showed a F1 score higher than 0.52 (Logistic Regression, KBHR class 6.4.). This is not satisfactory and thus, no experiments of automatically classifying the BHR to the KBHR with a traditional ML classifier were made.

7.2.2.4 FLAIR

The advantage of using FLAIR is that the framework does not require many inputs from the user. The downside is that the user has not much influence on the results. Even though different hyperparameters were tested, only four labels showed results higher than 0.00 for the best model.

The precision, recall and F1 score values were often much worse than the scores for the traditional ML algorithms. Furthermore, the best value were achieved for the learning rate was 0.75. This is surprising as the optimal learning rate was calculated beforehand and a learning rate of 0.03 should have yielded a lower loss. This is depicted in Figure 16, in which the loss is put in contrast to the learning rate. The same calculations were run for the experiment with only one label per document. Based on Figure 17, it was decided to use a learning rate of 0.07, which did not yield sufficient results nonetheless.

Since FLAIR is a framework with components that yield state-of-the-art results (Akbik et al., 2019; Devlin et al., 2018), the insufficient results might be an indicator that the dataset for training the models is not suitable for this TC task. This could either be due to the translation errors or errors in relation to the scraping process as discussed above. Furthermore, even though FLAIR offers powerful language-models, potentially the English dataset could have led to better results. Either because most documents were originally written in English or because more research was done on English datasets. All in all, it is assumed that the dataset is not suitable for the TC of the BHR.

Table viii: The classifiers that yielded the best F1 scores for each KBHR class

KBHR class	Best F1 Score	Classifiers
1.	---	---
1.1.	0,21	Naïve Bayes with FS; Naïve Bayes with altered stop word list and with FS
1.2.	0,24	Naïve Bayes with altered stop word list and without FS
1.3.	0,19	Logistic Regression with FS
1.4.	0,04	Naïve Bayes without FS; Naïve Bayes with altered stop word list and without FS
2.	---	---
2.1.	0,31	Logistic Regression with FS; SVM with altered stop word list and with FS
2.2.	0,23	Logistic Regression with FS
2.3.	0,00	---
2.4.	0,32	Naïve Bayes without FS; Naïve Bayes with altered stop word list and without FS; Logistic Regression with FS
3.	---	---
3.1.	0,36	Naïve Bayes with altered stop word list and without FS
3.2.	0,16	Naïve Bayes with FS
3.3.	0,47	Logistic Regression with FS
3.4.	0,14	Naïve Bayes with altered stop word list and with FS
3.5.	0,03	Naïve Bayes with FS; Naïve Bayes without FS; Naïve Bayes with altered stop word list and with FS; Naïve Bayes with altered stop word list and without FS
4.	---	---
4.1.	0,25	Logistic Regression with FS
4.2.	0,26	SVM with FS
5.	---	---
5.1.	0,06	Naïve Bayes without FS; Naïve Bayes with altered stop word list and without FS
5.2.	0,12	Naïve Bayes with altered stop word list and without FS
5.3.	0,50	SVM without FS; SVM with altered stop word list and without FS
6.	---	---
6.1.	0,30	Naïve Bayes with FS; Naïve Bayes with altered stop word list and with FS
6.2.	0,49	Naïve Bayes without FS; Naïve Bayes with altered stop word list and without FS
6.3.	0,03	Naïve Bayes with FS; Naïve Bayes with altered stop word list and with FS
6.4.	0,52	Logistic Regression with FS; SVM without FS; SVM with altered stop word list and without FS
7.	---	---
7.1.	0,33	Logistic Regression with FS
7.2.	0,33	Naive Bayes without FS; Naive Bayes with altered stop word list and without FS
7.3.	0,26	SVM with altered stop word list and with FS
7.4.	0,51	SVM without FS; SVM with altered stop word list and without FS

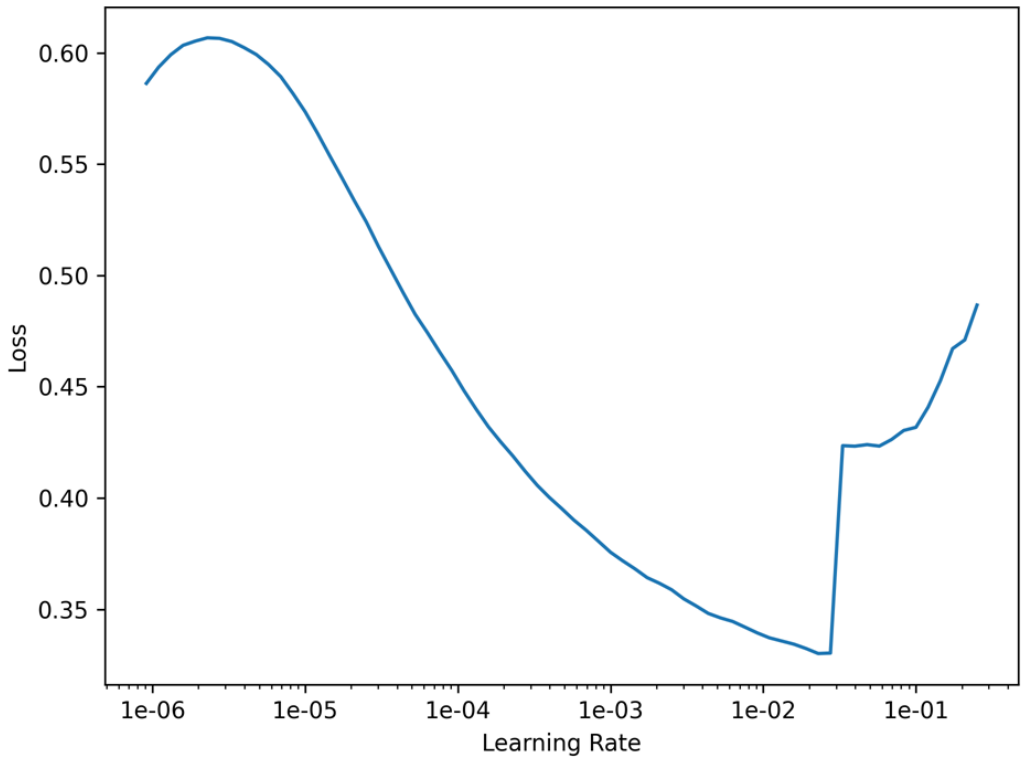


Figure 16: Contrast of the learning rate and the loss for the multilabel TC problem of the provided training set

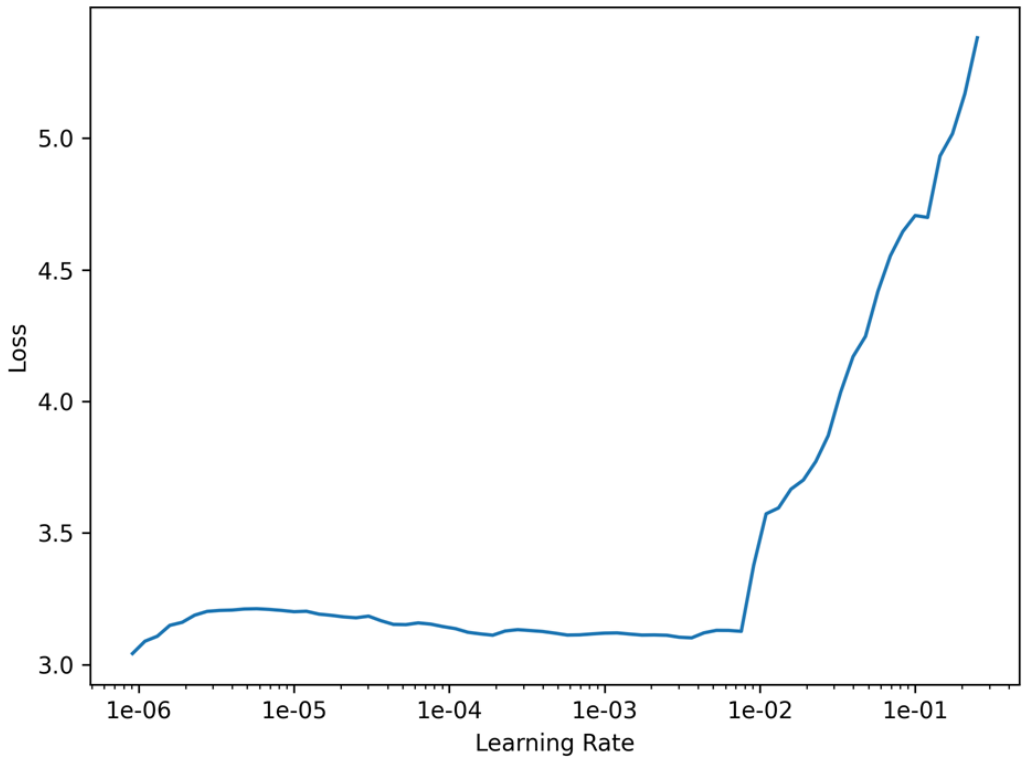


Figure 17: Contrast of the learning rate and the loss for the one label TC problem of the provided training set

7.2.3 General Remarks

With the results of this thesis, it was proven that accuracy alone is not a good evaluation metric as stated by Kass (2019). All the traditional ML classifiers showed satisfying average accuracies of 0.82 or higher, but their macro average F1 scores were very low. This is due to of the unbalanced distribution of the documents. The F1 score is therefore preferred for comparison. For all classifiers and the F1 score was zero for the models of KBHR class 2.3. KBHR class 2.3. was after KBHR class 1.4. the class with the least number of training documents. It is assumed that more information is needed for training the classifiers. However, even though much better F1 scores were achieved with the models for KBHR class 7.4., that contained the most training documents, than for KBHR class 2.3. the scores were not sufficient either.

The quality of the data influences the quality of the TC. It is likely that the dataset in general was insufficient for that kind of TC and reason for the unsatisfactory results. However, there are many components that influence the classification: Preprocessing, FE, FS, and classification methods can improve or worsen the results (Sebastiani, 2002). It is preferred to test as many combinations of methods as possible. Since the author of this thesis has no computer science background and therefore lacking experience and because of time restrictions, commonly used methods were chosen for FE and FS as well as for the hyperparameters of the classifiers. It is possible, however, that other settings would have worked better. Furthermore, more adaptions to the training set could have been made to make it less skewed, like deleting documents from the classes with the most documents or duplicating documents from classes with very little instances. However, the Complement Naïve Bayes proposed by Rennie et al. (2003) that was used in the experiments, was supposed to avoid this bias and yet yielded only slightly better results than the other classifiers. This observation leads to the assumption that the training set is of low quality and not suitable for this TC problem.

Afterall, automatic classification is not absolutely necessary for the BHR. 466 documents can be manually assigned to classes in a short amount of time. After this is done, classifiers can be trained and tested in future studies on those labeled BHR documents again. Joorabchi and Mahdi (2011) state that the classification accuracy of ML algorithms decreases the more classes the classification scheme has. In this thesis, the labels represented the 26 subordinate classes of the KBHR. 23 classes more than in most studies (e.g., Akhter et al., 2020; Banerjee

et al., 2019; Miao et al., 2018) which makes the TC problem more complicated. However, creating binary classifiers for each subclass should reduce this problem.

8 Conclusion

The objective of this thesis was to obtain a classification system for the collection of the BHR of the IBI. Furthermore, it was to be answered if JITA is an appropriate classification system for the BHR and if automatic TC using ML methods can be applied even if there is no BHR training set.

The first research question was investigated by summarizing several evaluation criteria from the literature in a checklist and evaluating JITA with this list. Moreover, a thorough description of the documents in the BHR and JITA was provided. Based on the collected information, it was concluded that JITA is not a suitable classification scheme for the BHR. The research at hand contributed to the very few recent publications about the evaluation of classification systems. On the one hand, the results can be used by the IBI to organize the BHR. On the other hand, the checklist can be employed by all other institutions that want to evaluate a classification system, but do not have the resources for expert evaluations or user studies. However, it cannot replace valuable insights from these two methods and that they could not be applied within the scope of this research is a clear limitation. Another limitation is that the quality check of the newly created classification system was done by the creator which should be avoided in future studies.

Moreover, using the created evaluation checklist, a completely new classification system for the BHR was introduced. The ‘Classification System for the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft’ can be used on the official website of the BHR as classification scheme. It offers users a broad overview of research that is conducted at the IBI.

The second part of the thesis addressed the second research question. Since no BHR training set was available, documents – consisting of title and abstract – were scraped from four publicly available databases (Springer, e-LiS, DABI, and o-bib). The documents were cleaned, translated into German and English and split into training, test, and validation set. Different combinations of preprocessing steps, classifiers and their hyperparameters were tested. After a discussion of different properties of multiple classifiers and their suitability for the TC of the BHR, a Complement Naïve Bayes classifier, a LinearSVC classifier and a Logistic Regression classifier were chosen as traditional ML classifiers. Furthermore, the FLAIR framework was used to apply a DL classifier as well. Since all results were not

satisfactory, it was concluded that an automatic classification of the BHR without an appropriate BHR training set is not possible. In future studies, more combinations of various preprocessing, FS and FE methods as well as classification algorithms should be tested to validate or reject those results. Furthermore, different modifications of the training set should be applied to avoid the bias of unbalanced data. Finally, after BHR documents are classified manually, future research can be based and conducted on those allocated documents. The thesis at hand provides a basis for all those possible future studies and additionally provides an overview about the general pipeline for TC problems. Especially newcomers to the ML field can benefit from these insights.

References

Aggarwal, C. C. (2014). An Introduction to Data Classification. In C. C. Aggarwal (Ed.). *Data classification: Algorithms and applications*. *Chapman & Hall/CRC Data Mining and Knowledge Discovery Series* (1-36). CRC Press.

Aggarwal, C. C., & Zhai, C. (2014). Text Classification. In C. C. Aggarwal (Ed.). *Data classification: Algorithms and applications*. *Chapman & Hall/CRC Data Mining and Knowledge Discovery Series* (287-336). CRC Press.

Akbik, A., Bergmann, T., Blythe, D., Rasul, Kashif, Schweter, S., & Vollgraf, R. (2019). FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstration*, 54–59. Retrieved September 25, 2021, from https://www.aclweb.org/anthology/N19-4010.pdf

Akbik, A., Blythe, D., & Vollgraf, R. (2018). Contextual String Embeddings for Sequence Labeling. *Proceedings of the 27th International Conference on Computational Linguisti*, 1638–1649. Retrieved September 25, 2021, from https://aclanthology.org/C18-1139

Akhter, M. P., Jiangbin, Z., Naqvi, I. R., Abdelmajeed, M., Mehmood, A., & Sadiq, M. T. (2020). Document-Level Text Classification Using Single-Layer Multisize Filters Convolutional Neural Network. *IEEE Access*, *8*, 42689–42707. https://doi.org/10.1109/ACCESS.2020.2976744

Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. *2017 International Conference on Engineering and Technology (ICET)*, 1–6. https://doi.org/10.1109/ICENGTECHNOL.2017.8308186

Banerjee, I., Ling, Y., Chen, M. C., Hasan, S. A., Langlotz, C. P., Moradzadeh, N., Chapman, B., Amrhein, T., Mong, D., Rubin, D. L., Farri, O., & Lungren, M. P. (2019). Comparative effectiveness of convolutional neural network (CNN) and recurrent neural network (RNN) architectures for radiology text report classification. *Artificial intelligence in medicine*, *97*, 79–88. https://doi.org/10.1016/j.artmed.2018.11.004

Bedford, D. (2013). Evaluating classification schema and classification decisions. *Bulletin of the American Society for Information Science and Technology*, *39*(2), 13–21. https://doi.org/10.1002/bult.2013.1720390206

Behera, B., Kumaravelan, G., & Kumar B., P. (2019). Performance Evaluation of Deep Learning Algorithms in Biomedical Document Classification. *2019 11th International Conference on Advanced Computing (ICoAC)*, 220–224. https://doi.org/10.1109/ICoAC48765.2019.246843

Borko, H. (1968). Information science: What is it? *American Documentation*, *19*(1), 3–5. https://doi.org/10.1002/asi.5090190103

Broughton, V. (2006). *Essential thesaurus construction*. Facet. https://doi.org/10.29085/9781856049849

Buckland, M. K. (1997). What is a “document”? *Journal of the American Society for Information Science*, *48*(9), 804–809. https://doi.org/10.1002/(SICI)1097-4571(199709)48:9%3C804::AID-ASI5%3E3.0.CO;2-V

Buckland, M. K. (1998). What is a "digital document"? *Document Numérique*, *2*(2), 221–230. Retrieved September 25, 2021, from https://www.marilia.unesp.br/Home/Instituicao/Docentes/EdbertoFerneda/what-is-a-digital-document.pdf (Reprint)

Cabrera Granados, M. F. (2014). *Applications of Deep Learning in Natural Language Processing for Information Extraction on German Language Documents* [Master Thesis]. Technische Universität München, München. Retrieved September 25, 2021, from https://babel.banrepcultural.org/digital/api/collection/p17054coll23/id/453/download

Caruana, R., & Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. *ICML '06: Proceedings of the 23rd international conference on Machine learning*, 161–168. https://doi.org/10.1145/1143844.1143865

Chen, G [Guibin], Ye, D., Xing, Z., Chen, J., & Cambria, E. (2017). Ensemble application of convolutional and recurrent neural networks for multi-label text categorization. *2017*

International Joint Conference on Neural Networks (IJCNN), 2377–2383. https://doi.org/10.1109/IJCNN.2017.7966144

Chen, G [Guo], Xiao, L., Hu, C., & Zhao, X. (2015). Identifying the research focus of Library and Information Science institutions in China with institution-specific keywords. *Scientometrics*, *103*(2), 707–724. https://doi.org/10.1007/s11192-015-1545-2

Cho, K., van Merrienboer, B., Bahdanau, D., & Bengio, Y. (2014, September 3). *On the Properties of Neural Machine Translation: Encoder-Decoder Approaches*. Retrieved September 25, 2021, from http://arxiv.org/pdf/1409.1259v2

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014, September 3). *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*. Retrieved September 25, 2021, from http://arxiv.org/pdf/1406.1078v3

Council on Library and Information Resources (CLIR). (2017, February 5). *1. Knowledge Organization Systems: An Overview*. Retrieved September 25, 2021, from https://www.clir.org/pubs/reports/pub91/1knowledge/

Dahlberg, I. (1974). *Grundlagen universaler Wissensordnung: Probleme und Möglichkeiten eines universalen Klassifikationssystems des Wissens*. *DGD-Schriftenreihe: Vol. 3*. Verlag Dokumentation. https://doi.org/10.1515/9783111412672

Dal Porto, S., & Marchitelli, A. (2006). The Functionality and Flexibility of Traditional Classification Schemes Applied to a Content Management System (CMS): Facets, DDC, JITA. *Knowledge Organization*, *33*(1), 35–44. https://doi.org/10.5771/0943-7444-2006-1-35

De Robbio, A., & Subirats Coll, I. (2014). E-LIS: Unique Model for Subject Specific Open Access Repository. *Informatics Studies*, *1*(1), 8–29. Retrieved August 19, 2021, from http://hdl.handle.net/10760/23244

Denuit, M., Hainaut, D., & Trufin, J. (2019). *Effective Statistical Learning Methods for Actuaries III: Neural Networks and Extensions* (1st ed.). *Springer Actuarial Lecture Notes*. Springer.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018, October 11). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* Google. Retrieved September 25, 2021, from https://arxiv.org/pdf/1810.04805

Dupriez, C. (2013). *Welcome to the initiative of the E-LIS community to improve JITA !* ASKOSI. Retrieved March 23, 2021, from http://www.destin-informatique.com/ASKOSI/Wiki.jsp?page=JITA%20Maintenance

Dwivedi, S. K., & Arya, C. (2016). Automatic Text Classification in Information retrieval: A Survey. *Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies*, Article 131, 1–6. https://doi.org/10.1145/2905055.2905191

e-LiS. (n. d.–a). *About Us*. Retrieved September 25, 2021, from http://eprints.rclis.org/information.html

e-LiS. (n. d.–b). *Guidelines: E-LIS Submission guidelines*. Retrieved September 25, 2021, from http://eprints.rclis.org/guidelines.html

Fernando, S., Hall, M., Agirre, E., Soroa, A., Clough, P., & Stevenson, M. (2012). Comparing Taxonomies for Organising Collections of Documents. *Proceedings of COLING 2012*, 879–894. Retrieved September 25, 2021, from https://www.aclweb.org/anthology/C12-1054/

Forman, G., & Scholz, M. (2010). Apples-to-apples in cross-validation studies: pitfalls in classifier performance measurement. *ACM SIGKDD Explorations Newsletter*, *12*(1), 49–57. https://doi.org/10.1145/1882471.1882479

Forsyth, D. (2019). *Applied Machine Learning*. Springer.

Fraunhofer ISST, &]init[. (2009, October 26). *Guidelines and Good Practices for Taxonomies* (1.3). Semantic Interoperability Centre Europe (SEMIC.EU). Retrieved September 25, 2021, from https://joinup.ec.europa.eu/collection/semantic-interoperability-community-semic/document/guidelines-and-good-practices-taxonomies

Galke, L., Mai, F., Schelten, A., Brunsch, D., & Scherp, A. (2017). Using Titles vs. Full-text as Source for Automated Semantic Document Annotation. *Proceedings of the*

Knowledge Capture Conference on - K-CAP 2017, Article 20, 1–4. https://doi.org/10.1145/3148011.3148039

Gantert, K. (2016). *Bibliothekarisches Grundwissen* (9th fully updated and expanded ed.). De Gruyter Saur. https://doi.org/10.1515/9783110321500

Gaus, W. (2005). *Dokumentations- und Ordnungslehre: Theorie und Praxis des Information Retrieval* (5th rev. ed.). *eXamen.press*. Springer. https://doi.org/10.1007/3-540-27518-5

Gayathri, K., & Marimuthu, A. (2013). Text Document Pre-Processing with the KNN for Classification Using the SVM. *2013 7th International Conference on Intelligent Systems and Control (ISCO)*, 453–457. https://doi.org/10.1109/ISCO.2013.6481197

Géron, A. (2018). *Praxiseinstieg Machine Learning mit Scikit-Learn und TensorFlow: Konzepte, Tools und Techniken für intelligente Systeme* (K. Rother, Trans.) (1st ed.). O'Reilly.

Golub, K., Soergel, D., Buchanan, G., Tudhope, D., Lykke, M., & Hiom, D. (2016). A framework for evaluating automatic indexing or classification in the context of retrieval. *Journal of the Association for Information Science and Technology*, *67*(1), 3–16. https://doi.org/10.1002/asi.23600

Gómez-Pérez, A. (1996). Towards a framework to verify knowledge sharing technology. *Expert Systems with Applications*, *11*(4), 519–529. https://doi.org/10.1016/S0957-4174(96)00067-X

Gruber, T. R. (1993). *Toward principles for the design of ontologies used for knowledge sharing* (KSL Reports 93-04). Stanford University.

Gruber, T. R. (1995). Toward principles for the design of ontologies used for knowledge sharing? *International Journal of Human-Computer Studies*, *43*(5-6), 907–928. https://doi.org/10.1006/ijhc.1995.1081

Hall, M. M., Fernando, S., Clough, P. D., Soroa, A., Agirre, E., & Stevenson, M. (2014). Evaluating hierarchical organisation structures for exploring digital libraries. *Information Retrieval*, *17*(4), 351–379. https://doi.org/10.1007/s10791-014-9242-y

Hjørland, B. (2013). Theories of Knowledge Organization—Theories of Knowledge. *Knowledge Organization*, *40*(3), 169–181. https://doi.org/10.5771/0943-7444-2013-3-169

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hsu, C.-W., Chang, C.-C., & Lin, C.-J. (2016, May 19). *A Practical Guide to Support Vector Classification*. Taipei. National Taiwan University. Retrieved September 25, 2021, from https://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf

Humboldt-Universität zu Berlin. (2018, June 05). *Fachspezifische Studien- und Prüfungsordnung für den Masterstudiengang Information Science: Überfachlicher Wahlpflichtbereich für andere Masterstudiengänge*, *27*(39). Retrieved September 25, 2021, from https://gremien.hu-berlin.de/de/amb/2018/39/39_2018_ma-information-science_druck.pdf

International Federation of Library Associations and Institutions. (2009). *Guidelines for Multilingual Thesauri* (IFLA Professional Reports 115). Retrieved September 25, 2021, from https://archive.ifla.org/VII/s29/pubs/Profrep115.pdf

Jacob, E. K. (2004). Classification and Categorization: A Difference that Makes a Difference. *Library Trends*, *52*(3), 515–540. Retrieved September 25, 2021, from http://hdl.handle.net/2142/1686

Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. In J. G. Carbonell, J. Siekmann, G. Goos, J. Hartmanis, J. van Leeuwen, C. Nédellec, & C. Rouveirol (Eds.). *Machine Learning: ECML-98*. *Lecture Notes in Computer Science: Vol. 1398* (pp. 137–142). Springer. https://doi.org/10.1007/BFb0026683

Joorabchi, A., & Mahdi, A. E. (2011). An unsupervised approach to automatic classification of scientific literature utilizing bibliographic metadata. *Journal of Information Science*, *37*(5), 499–514. https://doi.org/10.1177/0165551511417785

Kadhim, A. I. (2019). Survey on supervised machine learning techniques for automatic text classification. *Artificial Intelligence Review*, *52*(1), 273–292. https://doi.org/10.1007/s10462-018-09677-1

Kamath, C. N., Bukhari, S. S., & Dengel, A. (2018). Comparative Study between Traditional Machine Learning and Deep Learning Approaches for Text Classification. *Proceedings of the ACM Symposium on Document Engineering 2018*, Article 14, 1–11. https://doi.org/10.1145/3209280.3209526

Kass, M. (2019). *Textklassifikation mit neuronalen Netzen und klassischen Modellen* (Technische Berichte des Fachbereichs Elektrotechnik und Informatik, Hochschule Niederrhein 01). Hochschule Niederrhein. Retrieved September 25, 2021, from https://www.hs-niederrhein.de/fileadmin/dateien/FB03/Technische_Berichte/fb03-tb-2019-01.pdf

Khalid, S., Khalil, T., & Nasreen, S. (2014). A survey of feature selection and feature extraction techniques in machine learning. *2014 Science and Information Conference*, 372–378. https://doi.org/10.1109/SAI.2014.6918213

Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. *arXiv preprint arXiv:1408.5882*. Retrieved September 25, 2021, from https://arxiv.org/pdf/1408.5882

Kingma, D. P., & Ba, J. (2014, December 22). *Adam: A Method for Stochastic Optimization*. Retrieved September 25, 2021, from https://arxiv.org/pdf/1412.6980

Köhler, J. (2020a). *Rechercheverhalten von Geflüchteten: Eine Videoanalyse* (Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft 458). Berlin. Institut für Bibliotheks- und Informationswissenschaft der Humboldt-Universität zu Berlin. https://doi.org/10.18452/21728

Köhler, J. (2020b). Seeking Employment in a Non-Native Language: Online Information-Seeking Behavior of Refugees in Germany. *The International Journal of Information, Diversity, & Inclusion (IJIDI)*, *4*(2), 108–115. https://doi.org/10.33137/ijidi.v4i2.33144

Koller, D., & Sahami, M. (1997). *Hierarchically classifying documents using very few words* (Technical Report). Stanford InfoLab. Retrieved September 25, 2021, from http://ilpubs.stanford.edu:8090/291/

Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., & Brown, D. (2019). Text Classification Algorithms: A Survey. *Information*, *10*(4), Article 150. https://doi.org/10.3390/info10040150

Kwaśnik, B. H. (2021, January 28). *KO-ED Theoretical Perspectives: Critical Description and Evaluation of Classification Scheme* [Presentation Slides]. International Society for Knowledge Organization (ISKO UK).

Labrador, V., Peiró, Á., Garrido, Á. L., & Mena, E. (2020). LEDAC: Optimizing the Performance of the Automatic Classification of Legal Documents through the Use of Word Embeddings. *Proceedings of the 22nd International Conference on Enterprise Information Systems - Volume 1: ICEIS*, 181–188. https://doi.org/10.5220/0009421001810188

Lai, S., Xu, L., Liu, K., & Zhao, J. (2015). Recurrent Convolutional Neural Networks for Text Classification. *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2267-2273. Retrieved September 25, 2021, from http://zhengyima.com/my/pdfs/Textrcnn.pdf

Lawrie, D., Croft, W. B., & Rosenberg, A. (2001). Finding topic words for hierarchical summarization, *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 349–357. https://doi.org/10.1145/383952.384022

Lewis, D. D., Yang, Y., Rose, T. G., & Li, F. (2004). RCV1 A New Benchmark Collection for Text Categorization Research. *Journal of Machine Learning Research*, *5*, 361-397. Retrieved September 25, 2021, from https://www.jmlr.org/papers/volume5/lewis04a/lewis04a.pdf

Liu, G., & Yang, L. (2019). Popular research topics in the recent journal publications of library and information science. *The Journal of Academic Librarianship*, *45*(3), 278–287. https://doi.org/10.1016/j.acalib.2019.04.001

Liu, L., & Liang, Q. (2011). A high-performing comprehensive learning algorithm for text classification without pre-labeled training set. *Knowledge and Information Systems*, *29*(3), 727–738. https://doi.org/10.1007/s10115-011-0387-3

Liu, P., Qiu, X., & Huang, X. (2016). *Recurrent Neural Network for Text Classification with Multi-Task Learning*. Retrieved September 25, 2021, from http://arxiv.org/pdf/1605.05101v1

Liu, Y. Y., Yang, M., Ramsay, M., Li, X. S., & Coid, J. W. (2011). A Comparison of Logistic Regression, Classification and Regression Tree, and Neural Networks Models in Predicting Violent Re-Offending. *Journal of Quantitative Criminology*, *27*(4), 547–573. https://doi.org/10.1007/s10940-011-9137-7

Lorenz, B. (2018). Zur Theorie und Terminologie der bibliothekarischen Klassifikation. In H. Alex, G. Bee, & U. Junger (Eds.). *Klassifikationen in Bibliotheken: Theorie – Anwendung – Nutzen*. *Bibliotheks- und Informationspraxis: Vol. 53* (pp. 1–22). De Gruyter Saur.

Luft, J. (2015). The Challenges of Being a Fox – Library and Information Science as an Applied Discipline. *Bibliothek Forschung und Praxis*, *39*(2), 132–137. https://doi.org/10.1515/bfp-2015-0015

Ma, J., & Lund, B. (2021). The evolution and shift of research topics and methods in library and information science. *Journal of the Association for Information Science and Technology*, *72*(8), 1059–1074. https://doi.org/10.1002/asi.24474

Madsen, B. N., & Thomsen, H. E. (2009). Ontologies vs. classification systems. *NEALT Proceedings Series*, *7*, 27–32. Retrieved September 25, 2021, from http://beta.visl.sdu.dk/~eckhard/cdrom/workshop3_wordnets.pdf#page=31

Maedche, A., & Staab, S. (2002). Measuring Similarity between Ontologies. In A. Gómez-Pérez & V. R. Benjamins (Eds.). *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web. EKAW 2002*. *Lecture Notes in Computer Science: Vol. 2473* (pp. 251–263). Springer. https://doi.org/10.1007/3-540-45810-7_24

Manecke, H.-J. (2004). Klassifikation, Klassieren. In R. Kuhlen, T. Seeger, & D. Strauch (Eds.), *Grundlagen der praktischen Information und Dokumentation: Band 1: Handbuch zur Einführung in die Informationswissenschaft und -praxis - Band 2: Glossar* (5th ed., pp. 127–140). De Gruyter. https://doi.org/10.1515/9783110964110.127

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

Matveyeva, S. J. (2002). A Role for Classification: The Organization of Resources on the Internet. *MLA Forum*, *1*(2). Retrieved September 25, 2021, from http://hdl.handle.net/10057/1264

Miao, F., Zhang, P., Jin, L., & Wu, H. (2018). Chinese News Text Classification Based on Machine Learning Algorithm. *2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, 48–51. https://doi.org/10.1109/IHMSC.2018.10117

Moral, C., Antonio, A. de, Imbert, R., & Ramírez, J. (2014). A survey of stemming algorithms in information retrieval. *Information Research*, *19*(1). Retrieved September 25, 2021, from http://informationr.net/ir/19-1/paper605.html

National Information Standards Organization. (2005, July 25). *Guidelines for the Construction, Format, and Management of Monolingual Controlled Vocabularies* (Z39.19-2005). Bethesda, Maryland, U.S.A. Retrieved September 25, 2021, from https://ils.unc.edu/courses/2015_fall/inls151_002/Readings/NISO.pdf

Oberhauser, O. (2005). Automatisches Klassifizieren und Bibliothekskataloge. In H. Hrusa (Ed.), *Bibliothek Technik Recht: Festschrift für Peter Kubalek zum 60. Geburtstag* (pp. 119–131). Manz. Retrieved September 25, 2021, from http://eprints.rclis.org/6789/

Ohly, P. (2020). Ingetraut Dahlberg (1927-2017). *Knowledge Organization*, *47*(2), 173–182. https://doi.org/10.5771/0943-7444-2020-2-173

Pawar, P. Y., & Gawande, S. H. (2012). A Comparative Study on Different Types of Approaches to Text Categorization. *International Journal of Machine Learning and Computing (IJMLC)*, *2*(4), 423–426. https://doi.org/10.7763/IJMLC.2012.V2.158

Pfeffer, M., & Schöllhorn, K. (2018). Praktische Nutzung von Klassifikationssystemen. In H. Alex, G. Bee, & U. Junger (Eds.). *Klassifikationen in Bibliotheken: Theorie – Anwendung – Nutzen*. *Bibliotheks- und Informationspraxis: Vol. 53* (pp. 207–234). De Gruyter Saur. https://doi.org/10.1515/9783110299250-007

Pham, N.-Q., Kruszewski, G., & Boleda, G. (2016). Convolutional Neural Network Language Models. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1153–1162. https://doi.org/10.18653/v1/D16-1123

Pintas, J. T., Fernandes, L. A. F., & Garcia, A. C. B. (2021). Feature selection methods for text classification: a systematic literature review. *Artificial Intelligence Review.* Advance online publication. https://doi.org/10.1007/s10462-021-09970-6

Pong, J. Y.-H., Kwok, R. C.-W., Lau, R. Y.-K., Hao, J.-X., & Wong, P. C.-C. (2008). A comparative study of two automatic document classification methods in a library setting. *Journal of Information Science*, *34*(2), 213–230. https://doi.org/10.1177/0165551507082592

Ramadhani, R. A., Indriani, F., & Nugrahadi, D. T. (2016). Comparison of Naive Bayes smoothing methods for Twitter sentiment analysis. *2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 287–292. https://doi.org/10.1109/ICACSIS.2016.7872720

Ranganathan, S. R. (1937). *Prolegomena to library classification*. The Madras Library Association. Retrieved August 25, 2021, from http://hdl.handle.net/10973/19232

Ranganathan, S. R. (1967). *Prolegomena to Library Classification* (3rd ed.). *Ranganathan Series in Library Science: Vol. 20*. Asia Publishing House.

Reiner, U. (2010). Automatische DDC-Klassifizierung. *Dialog mit Bibliotheken*, *22*(1), 23–29. Retrieved September 25, 2021, from https://nbn-resolving.org/urn:nbn:de:101-2016110378

Rennie, J. D., Shih, L., Teevan, J., & Karger, D. R. (2003). Tackling the Poor Assumptions of Naive Bayes Text Classifiers. *Proceedings of the Twentieth International Conference*

on Machine Learning (ICML-2003), 616–623. Retrieved September 25, 2021, from https://www.aaai.org/Papers/ICML/2003/ICML03-081.pdf

Romanov, A., Lomotin, K., & Kozlova, E. (2019). Application of Natural Language Processing Algorithms to the Task of Automatic Classification of Russian Scientific Texts. *Data Science Journal*, *18*(1), Article 37, 1–17. https://doi.org/10.5334/dsj-2019-037

Schalkoff, R. J. (2007). Pattern Recognition. In B. W. Wah (Ed.), *Wiley Encyclopedia of Computer Science and Engineering.* John Wiley & Sons, Inc. https://doi.org/10.1002/9780470050118.ecse302

Scherer, B. (2003). *Automatische Indexierung und ihre Anwendung im DFG-Projekt „Gemeinsames Portal für Bibliotheken, Archive und Museen (BAM)“* [Master Thesis]. Universität Konstanz, Konstanz. Retrieved September 25, 2021, from http://nbn-resolving.de/urn:nbn:de:bsz:352-opus-9965

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, *34*(1), 1–47. https://doi.org/10.1145/505282.505283

Shah, K., Patel, H., Sanghvi, D., & Shah, M. (2020). A Comparative Analysis of Logistic Regression, Random Forest and KNN Models for the Text Classification. *Augmented Human Research*, *5*(1), Article 12. https://doi.org/10.1007/s41133-020-00032-0

Shaheen, S. K. (2013). A Proposed Knowledge. Map for Library, Archives and Information Science from an Academic-Professional View Highlighting Cairo University. *The International Journal of Social Sciences*, *7*(1), 110–129. Retrieved September 25, 2021, from https://www.tijoss.com/7th%20volume/sherif.pdf

Sharma, K., Gaikwad, A., Patil, S., Kumar, P., & Salapurkar, D. P. (2018). Automated Document Summarization and Classification Using Deep Learning. *International Research Journal of Engineering and Technology (IRJET)*, *5*(6), 1182–1185. Retrieved August 25, 2021, from https://www.academia.edu/download/57105427/IRJET-V5I6222.pdf

Skansi, S. (2018). *Introduction to Deep Learning: From Logical Calculus to Artificial Intelligence*. Springer. https://doi.org/10.1007/978-3-319-73004-2

Song, X., Salcianu, A., Song, Y., Dopson, D., & Zhou, D. (2020, December 31). *Linear-Time WordPiece Tokenization*. Retrieved September 25, 2021, from http://arxiv.org/pdf/2012.15524v1

Spirovski, K., Stevanoska, E., Kulakov, A., Popeska, Z., & Velinov, G. (2018). Comparison of different model's performances in task of document classification. *Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics*, Article 10, 1–12. https://doi.org/10.1145/3227609.3227668

Steinwendner, J., & Schwaiger, R. (2020). *Neuronale Netze programmieren mit Python* (2nd updated and rev. ed.). Rheinwerk Computing.

Stoica, E., Hearst, M. A., & Richardson, M. (2007). Automating Creation of Hierarchical Faceted Metadata Structures. *Proceedings of NAACL HLT 2007*, 244–251. Retrieved September 25, 2021, from https://aclanthology.org/N07-1031.pdf

Stuart, D. (2016). *Practical ontologies for information professionals*. Facet. https://doi.org/10.29085/9781783301522

Ting, S. L., Ip, W. H., & Tsang, A. H. (2011). Is Naïve Bayes a Good Classifier for Document Classification? *International Journal of Software Engineering and Its Applicatio*, *5*(3), 37–46. Retrieved September 25, 2021, from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.643.6611&rep=rep1&type=pdf

Umlauf, K. (1999). *Einführung in die bibliothekarische Klassifikationstheorie und -praxis: Mit Übungen* (Berliner Handreichungen zur Bibliothekswissenschaft 67). Berlin. Institut für Bibliotheks- und Informationswissenschaft der Humboldt-Universität zu Berlin. https://doi.org/10.18452/18436

Uysal, A. K., & Gunal, S. (2014). The impact of preprocessing on text classification. *Information Processing & Management*, *50*(1), 104–112. https://doi.org/10.1016/j.ipm.2013.08.006

van Rees, R. (2003). *Clarity in the usage of the terms ontology, taxonomy and classification* (CIB Report 284). Retrieved September 25, 2021, from https://www.cs.auckland.ac.nz/research/conferences/w78/papers/W78-37.pdf

Vanjani, M., & Aiken, M. (2020). A Comparison of Free Online Machine Language Translators. *Journal of Management Science and Business Intelligence*, *5*(1), 26–31. https://doi.org/10.5281/ZENODO.3961085

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017, June 12). *Attention Is All You Need*. Retrieved September 25, 2021, from http://arxiv.org/pdf/1706.03762v5

Weigend, A. S., Wiener, E. D., & Pedersen, J. O. (1999). Exploiting Hierarchy in Text Categorization. *Information Retrieval*, *1*(3), 193–216. https://doi.org/10.1023/A:1009983522080

Yang, Y., & Pedersen, J. O. (1997). A Comparative Study on Feature Selection in Text Categorization. *Proceedings of the Fourteenth International Conference on Machine Learning*, 412–420. Retrieved September 25, 2021, from http://nyc.lti.cs.cmu.edu/yiming/Publications/yang-icml97.pdf

Zheng, J., & Zheng, L. (2019). A Hybrid Bidirectional Recurrent Convolutional Neural Network Attention-Based Model for Text Classification. *IEEE Access*, *7*, 106673–106685. https://doi.org/10.1109/ACCESS.2019.2932619

Ziganshina, L. E., Yudina, E. V., Gabdrakhmanov, A. I., & Ried, J. (2021). Assessing Human Post-Editing Efforts to Compare the Performance of Three Machine Translation Engines for English to Russian Translation of Cochrane Plain Language Health Information: Results of a Randomised Comparison. *Informatics*, *8*(1), Article 9. https://doi.org/10.3390/informatics8010009

Zins, C. (2007a). Classification schemes of Information Science: Twenty-eight scholars map the field. *Journal of the American Society for Information Science and Technology*, *58*(5), 645–672. https://doi.org/10.1002/asi.20506

Zins, C. (2007b). Knowledge map of information science. *Journal of the American Society for Information Science and Technology*, *58*(4), 526–535. https://doi.org/10.1002/asi.20505

Zulfiqar, S., Wahab, M. F., Sarwar, M. I., & Lieberwirth, I. (2018). Is Machine Translation a Reliable Tool for Reading German Scientific Databases and Research Articles? *Journal of Chemical Information and Modeling*, *58*(11), 2214–2223. https://doi.org/10.1021/acs.jcim.8b00534

Appendix I – Description of the Berliner Handreichungen & JITA

The data was uploaded to Zenodo[25] with the following doi: https://doi.org/10.5281/zenodo.6957077. There the following files can be found:

- *BHR_corpus_and_analysis.xlsx* – This file contains the scraped BHR documents and their metadata as well as the analysis of the BHR collection.
- *JITA_analysis.xlsx* – This file contains the analysis of JITA

[25] Köhler, J. (2022). Description of the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft & JITA Classification System of Library and Information Science. Zenodo. https://doi.org/10.5281/zenodo.6957077

Appendix II – Analysis of collected Classification Systems by Zins

The data was uploaded to Zenodo[26] with the following doi: https://doi.org/10.5281/zenodo.6957587. There the following files can be found:

- *Consensus_Information_Science_Subjects.pdf* – A structured collection of Information Science subjects that meets a broad consensus of Information Science experts that was obtained after coding the 28 classification systems collected by Zins (2007a). Codes that only occurred once (see *Organizing_the_Subclasses.xlsx*) were deleted from this list. The list will also be presented below.
- *Consensus_x_JITA.docx.pdf* – A comparison of the structured collection of Information Science subjects with the JITA Classification System of Library and Information Science. If a topic is completely covered by a class in JITA it is marked in yellow. A topic is marked in gray if it is only partially covered by a class in JITA. The notation of the respective classes is given in square brackets behind the topic. The comparison will also be presented below.
- *Consensus_x_KBHR.pdf* – A comparison of the structured collection of Information Science subjects with the Classification System for the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft. If a topic is completely covered by a class in the KBHR it is marked in yellow. A topic is marked in gray if it is only partially covered by a class in KBHR. The notation of the respective classes is given in square brackets behind the topic. The comparison will also be presented below.
- *Organizing_the_Main_Classes.xlsx* – In the attached excel file, a detailed description of the organizing process of the codes for the main classes of the classification systems collected by Zins (2007a) can be found.
- *Organizing_the_Subclasses.xlsx* – In the attached excel file, a detailed description of the organizing process of the codes for the subclasses of the classification systems collected by Zins (2007a) can be found. This organizing process resulted in the final structured collection of Information Science subjects (*Consensus_Information_Science_Subjects.pdf*).

[26] Köhler, J. (2022). Analysis of collected Information Science Classification Systems by Zins. Zenodo. https://doi.org/10.5281/zenodo.6957587

- *Zins_(2007)_Classification_Systems_Comparison.mx20* – The MAXQDA file contains the data and the codes that were used to create a structured collection of Information Science subjects.

Structured Collection of Information Science Subjects

A structured collection of Information Science subjects that meets a broad consensus of Information Science experts that was obtained after coding the 28 classification systems collected by Zins (2007a). Codes that only occurred once were deleted from this list.

History, Philosophy & Foundations Of Information Science	Disciplines & Related Fields	Legal, Ethical, Educational & Social Issues
History History Of Library Science History Of Librarianship History Of IS Philosophy IS Epistemology Philosophy Of Information Science Philosophy Of Information Philosophy Of Computers Philosophy Of Librarianship Foundations Of Information Science Theories Information Science Theory Information Theory Library Science Theory Librarianship Theory Cognition Theory Message Theory Communication Theory Documentation	Librarianship Metalibrarianship Library Science Archival Science Museology Communication Scientific Communication Grey Literature Computer Mediated Communication Social Communication Chemical Documentation Mathematics & Logic Informatics Aviation Informatics Health/Biomedical Informatics Bioinformatics Community Informatics Environment Cognition Science Linguistics & Logic Semantics Semiotics Computational Operations Research Memetics	Law Information Policies Public Information Policies Privacy Copyright Data Privacy Censorship Filtering Ethics Free Access To Information Freedom Of Information Intellectual Property IS Education & Training Information Literacy Info & IT Literacy Courses & Curricula Training Courses Information Skills User Education Continuing Education Lifelong Learning E-Learning Educational Information Social & Cultural Aspects In The Information Society The Information Society Information Communities Futures Scenarios Social Information Information Politics E-Government Sociology Of Knowledge Information In Traditional & Transitional Societies Information Cultures

Information Professions, Information Services & Applied IS	Information Industries, Economy & Management	Information Technology
Professions Information Brokers Professional Organizations Information Services Libraries & Information Centres Library Facilities Opacs Digital Libraries Digital & Virtual Libraries, Hybrid Libraries State & National Libraries Public Libraries Academic Libraries Government Libraries Special Libraries Library Management Library Automation & Operations Museums Archives Web Web Pages Transmission Scientific Information	Information Industry Market Electronic Information Industry Newspapers Marketing Publishing Electronic Publishing E-Books E-Journals Print Labor In Information Systems Economics Of Information Management Information Management Knowledge Management Document Management Digitization Collection Management Records & Archives Management Competitive Intelligence Human Resource Management Financial Management	Information Technology Technological Information Preservation Technologies Software Artificial Intelligence Intelligent Agents Pattern Recognition Programming Languages Hardware Telecommunications Internet Technologies Internet Search Engines Hypermedia Hypertext Systems Directories Multimedia Networks Technologies Intranets Portals And Gateways Communication & Computer Networks Information Networks Digital Security Access Control Authentication Encryption (Digital Watermarking) Data Mining Mobile Information Technologies

Information Systems	Information Use & Users	Information Processing & Retrieval
Information Systems Systems Analysis Access Systems Information Retrieval Systems Document Delivery Systems Interlibrary Loan High-Density Book Storage Systems Information Architecture Information Structures Information Design Mass Media Distributed Networked Environments	Human Information Behavior Information Use And User Users User Studies Readership Studies Information Use Information Utilization Information Usability Information Uses & Applications Information Seeking Behavior Information Need Production Of Knowledge Behavior Human Computer Interaction Design Issues Cognition Aspects Of Information Transfer	Information Processing Information Dissemination Information Products Bibliography Preservation Digital Preservation Information Storage Automatic Processing Of Language Information Manipulation Information Retrieval Search Methods Online Searching Techniques Image Retrieval Music-Information-Retrieval Databases

Information & Knowledge Organization	Measuring, Evaluation & Research	Others
Knowledge Organization Knowledge Knowledge Representation Information Representation Knowledge Structures Cataloging Abstracting Indexing Subject Analysis Domain Analysis Tools For Knowledge Organization The Semantic Web Categorization & Classification Classification Systems Classification Theory Classification Schemes Vocabulary Control Thesauri Taxonomies Ontology Metadata Discipline Area Concepts Terminology	Theories & Methodologies Of IS Webometrics Informetrics Bibliometrics Scientometrics Citation Analysis Evaluation Research Evaluation Information Quality Evaluation Of Information Quality Standards Usability Studies Quality Assurance Of Software Evaluation Of Information Systems Research Methods Qualitative Research Quantitative Research Information Acquisition Diffusion Studies Information Diffusion User Needs Studies	Reference Work Biographies Data Documents Message Information Information Sources Information Genres

Comparison of a Structured Collection of Information Science Subjects and JITA

A comparison of the structured collection of Information Science subjects with the JITA Classification System of Library and Information Science. If a topic is completely covered by a class in JITA it is marked in yellow. A topic is marked in gray if it is only partially covered by a class in JITA. The notation of the respective classes is given in square brackets behind the topic.

History, Philosophy & Foundations Of Information Science	Disciplines & Related Fields	Legal, Ethical, Educational & Social Issues
History History Of Library Science [AA, AZ] History Of Librarianship [AZ] History Of IS [AA, AZ] Philosophy IS Epistemology Philosophy Of Information Science [AA] Philosophy Of Information [AA] Philosophy Of Computers [LD] Philosophy Of Librarianship [AA] Foundations Of Information Science [AA] Theories [AB] Information Science Theory [AB] Information Theory [AB] Library Science Theory [AB] Librarianship Theory [AB] Cognition Theory Message Theory Communication Theory [BJ] Documentation	Librarianship Metalibrarianship Library Science [AA] Archival Science [DL] Museology [DM] Communication [BJ] Scientific Communication [E] Grey Literature [HB] Computer Mediated Communication [GC] Social Communication [BJ] Chemical Documentation [AC] Mathematics & Logic [AC] Informatics [AC] Aviation Informatics [AC] Health/Biomedical Informatics [AC] Bioinformatics [AC] Community Informatics [AC] Environment [AC] Cognition Science [AC] Linguistics & Logic [AC] Semantics [AC] Semiotics [AC] Computational [AC] Operations Research [AC] Memetics [AC]	Law Information Policies [BF] Public Information Policies [BF] Privacy Copyright [ED] Data Privacy Censorship [EF] Filtering [IJ] Ethics Free Access To Information Freedom Of Information Intellectual Property IS Education & Training [GH, GI] Information Literacy [CE] Info & IT Literacy [CE] Courses & Curricula [GG] Training Courses [GI] Information Skills [CE] User Education [CD] Continuing Education [GH, GZ] Lifelong Learning [CD, GH] E-Learning [CD, GH] Educational Information [CD, GH] Social & Cultural Aspects In The Information Society The Information Society [BD] Information Communities [BD] Futures Scenarios Social Information Information Politics [BD, BF] E-Government [BD, BF] Sociology Of Knowledge Information In Traditional & Transitional Societies Information Cultures

Information Professions, Information Services & Applied IS	Information Industries, Economy & Management	Information Technology
Professions [G] Information Brokers [GZ] Professional Organizations [GD] Information Services [I] Libraries & Information Centres [D] Library Facilities [D] Opacs [HM] Digital Libraries Digital & Virtual Libraries, Hybrid Libraries [DZ] State & National Libraries [DB] Public Libraries [DC] Academic Libraries [DD] Government Libraries [DF] Special Libraries [DH] Library Management [F] Library Automation & Operations [LQ] Museums [DM] Archives [DL] Web Web Pages [HQ] Transmission Scientific Information [AZ]	Information Industry Market Electronic Information Industry [GB, GC] Newspapers [HA] Marketing [FB] Publishing [E; EB] Electronic Publishing [EB] E-Books [HO] E-Journals [HN] Print [EB; HE] Labor In Information Systems Economics Of Information [BE] Management [F] Information Management Knowledge Management [FJ] Document Management Digitization [JG] Collection Management Records & Archives Management [DL] Competitive Intelligence Human Resource Management [FE] Financial Management [FC]	Information Technology [L] Technological Information Preservation Technologies [JF, JH, LE] Software [LJ] Artificial Intelligence Intelligent Agents [LP] Pattern Recognition Programming Languages Hardware Telecommunications [LA] Internet Technologies [LC] Internet [LC] Search Engines [LS] Hypermedia [IG] Hypertext Systems [IG] Directories Multimedia [HH] Networks Technologies [LB] Intranets Portals And Gateways [HR] Communication & Computer Networks [LA, LB] Information Networks Digital Security [LH] Access Control [LI] Authentication [LI] Encryption (Digital Watermarking) [LH] Data Mining Mobile Information Technologies [LT]

Information Systems	Information Use & Users	Information Processing & Retrieval
Information Systems Systems Analysis Access Systems Information Retrieval Systems Document Delivery Systems [JJ] Interlibrary Loan [JK] High-Density Book Storage Systems Information Architecture Information Structures [IE] Information Design [IK] Mass Media [EA] Distributed Networked Environments	Human Information Behavior Information Use And User [BZ, CZ] Users User Studies [CB] Readership Studies Information Use [BZ, CZ] Information Utilization [CA] Information Usability [BI] Information Uses & Applications [CA] Information Seeking Behavior Information Need [BH] Production Of Knowledge Behavior Human Computer Interaction Design Issues [IK] Cognition Aspects Of Information Transfer [IF]	Information Processing Information Dissemination [BG] Information Products Bibliography Preservation [JF, JH] Digital Preservation [JH] Information Storage Automatic Processing Of Language [LL] Information Manipulation Information Retrieval Search Methods Online Searching Techniques Image Retrieval [IH] Music-Information-Retrieval Databases [HL, LN]

Information & Knowledge Organization	Measuring, Evaluation & Research	Others
Knowledge Organization Knowledge Knowledge Representation [ID] Information Representation Knowledge Structures Cataloging [IA] Abstracting [IA] Indexing [IA] Subject Analysis Domain Analysis Tools For Knowledge Organization The Semantic Web [IL] Categorization & Classification Classification Systems Classification Theory Classification Schemes Vocabulary Control Thesauri Taxonomies Ontology Metadata [IE] Discipline Area Concepts Terminology	Theories & Methodologies Of IS Webometrics Informetrics Bibliometrics [BB] Scientometrics Citation Analysis Evaluation Research Evaluation Information Quality Evaluation Of Information Quality Standards Usability Studies [BI] Quality Assurance Of Software Evaluation Of Information Systems Research Methods Qualitative Research Quantitative Research Information Acquisition [JA] Diffusion Studies [BG] Information Diffusion [BG] User Needs Studies [CB]	Reference Work [IJ] Biographies [GF] Data [IE] Documents Message Information Information Sources [H, HZ] Information Genres

Comparison of a Structured Collection of Information Science Subjects and the KBHR

A comparison of the structured collection of Information Science subjects with the Classification System for the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft. If a topic is completely covered by a class in the KBHR it is marked in yellow. A topic is marked in gray if it is only partially covered by a class in KBHR. The notation of the respective classes is given in square brackets behind the topic.

History, Philosophy & Foundations Of Information Science	Disciplines & Related Fields	Legal, Ethical, Educational & Social Issues
History [1.2.] History Of Library Science [1.2.] History Of Librarianship [1.2.] History Of IS [1.2.] Philosophy IS Epistemology Philosophy Of Information Science Philosophy Of Information Philosophy Of Computers Philosophy Of Librarianship Foundations Of Information Science [1.] Theories [1.1.] Information Science Theory [1.1.] Information Theory [1.1.] Library Science Theory [1.1.] Librarianship Theory [1.1.] Cognition Theory [1.1.] Message Theory [1.1.] Communication Theory [1.1.] Documentation	Librarianship [1.4.] Metalibrarianship [1.4.] Library Science [1.4.] Archival Science [1.4.] Museology [1.4.] Communication [1.4.] Scientific Communication [6.1.] Grey Literature Computer Mediated Communication [1.4.] Social Communication [1.4., 4.] Chemical Documentation [1.4.] Mathematics & Logic [1.4.] Informatics [1.4.] Aviation Informatics [1.4.] Health/Biomedical Informatics [1.4.] Bioinformatics [1.4.] Community Informatics [1.4.] Environment [1.4.] Cognition Science [1.4.] Linguistics & Logic [1.4.] Semantics Semiotics Computational Operations Research [1.4.] Memetics [1.4.]	Law [1.4., 4.] Information Policies Public Information Policies Privacy Copyright Data Privacy Censorship Filtering Ethics [4.] Free Access To Information Freedom Of Information Intellectual Property IS Education & Training [1.3.] Information Literacy [4.1.] Info & IT Literacy [4.1.] Courses & Curricula [1.3.] Training Courses [1.3.] Information Skills [4.1.] User Education [1.3.] Continuing Education [1.3.] Lifelong Learning [1.3.] E-Learning [1.3.] Educational Information [1.3.] Social & Cultural Aspects In The Information Society [4.] The Information Society [4.] Information Communities [4.] Futures Scenarios [4.2.] Social Information Information Politics [4.] E-Government Sociology Of Knowledge Information In Traditional & Transitional Societies Information Cultures [4.]

Information Professions, Information Services & Applied IS	Information Industries, Economy & Management	Information Technology
Professions Information Brokers Professional Organizations Information Services [3.5.] Libraries & Information Centres [3.1.] Library Facilities [3.1.] Opacs Digital Libraries [7.] Digital & Virtual Libraries, [7.] Hybrid Libraries [3.1.] State & National Libraries [3.1.] Public Libraries [3.1.] Academic Libraries [3.1.] Government Libraries [3.1.] Special Libraries [3.1.] Library Management [3.4.] Library Automation & Operations [3.3.] Museums [3.1.] Archives [3.1.] Web [7.3.] Web Pages Transmission Scientific Information	Information Industry Market Electronic Information Industry Newspapers Marketing [3.4.] Publishing [6.1.] Electronic Publishing [6.1.] E-Books E-Journals Print Labor In Information Systems Economics Of Information [3.4.] Management [3.4.] Information Management [2., 3.4.] Knowledge Management [2.] Document Management [2.] Digitization Collection Management [2.] Records & Archives Management [2.] Competitive Intelligence Human Resource Management [3.4.] Financial Management [3.4.]	Information Technology Technological Information Preservation Technologies [6.3.] Software Artificial Intelligence [7.4.] Intelligent Agents [7.4.] Pattern Recognition [7.4.] Programming Languages Hardware Telecommunications Internet Technologies [7.3.] Internet [7.3.] Search Engines [7.] Hypermedia Hypertext Systems Directories Multimedia Networks Technologies Intranets [7.3.] Portals And Gateways [7.3.] Communication & Computer Networks [7.3.] Information Networks [7.3.] Digital Security Access Control Authentication Encryption (Digital Watermarking) Data Mining [7.4.] Mobile Information Technologies

Information Systems	Information Use & Users	Information Processing & Retrieval
Information Systems [7.] Systems Analysis [7.2.] Access Systems [7.] Information Retrieval Systems [7.] Document Delivery Systems Interlibrary Loan High-Density Book Storage Systems Information Architecture Information Structures Information Design [7.1.] Mass Media Distributed Networked Environments	Human Information Behavior [5.] Information Use And User [5.1.] Users User Studies [5.] Readership Studies Information Use [5.1.] Information Utilization [5.1.] Information Usability [5.1., 5.3.] Information Uses & Applications [5.1.] Information Seeking Behavior [5.2.] Information Need [5.2.] Production Of Knowledge Behavior Human Computer Interaction [5.3.] Design Issues Cognition Aspects Of Information Transfer	Information Processing [7.] Information Dissemination Information Products Bibliography Preservation [6.3.] Digital Preservation [6.3.] Information Storage [6.] Automatic Processing Of Language [7.4.] Information Manipulation [7.] Information Retrieval [7.] Search Methods [5.2.] Online Searching Techniques Image Retrieval Music-Information-Retrieval Databases [7.]

Information & Knowledge Organization	Measuring, Evaluation & Research	Others
Knowledge Organization [2.] Knowledge Knowledge Representation Information Representation Knowledge Structures Cataloging [2.2.] Abstracting [2.2.] Indexing [2.2.] Subject Analysis Domain Analysis Tools For Knowledge Organization [3.1.] The Semantic Web [3.1.] Categorization & Classification [3.1.] Classification Systems [3.1.] Classification Theory [1.1.] Classification Schemes [3.1.] Vocabulary Control [2.2.] Thesauri [3.1.] Taxonomies [3.1.] Ontology [3.1.] Metadata Discipline Area Concepts Terminology [1.1.]	Theories & Methodologies Of IS [1.1.] Webometrics [6.4.] Informetrics [6.4.] Bibliometrics [6.4.] Scientometrics [6.4.] Citation Analysis [6.4.] Evaluation Research Evaluation Information Quality [7.2.] Evaluation Of Information Quality[7.2.] Standards [1.1.] Usability Studies [5.3.] Quality Assurance Of Software Evaluation Of Information Systems [7.2.] Research Methods [1.1.] Qualitative Research Quantitative Research Information Acquisition Diffusion Studies [1.4.] Information Diffusion User Needs Studies [5.2.]	Reference Work Biographies Data Documents Message Information Information Sources Information Genres

Appendix III – Scrapers, Queries and Mappings

The data was uploaded to Zenodo[27] with the following doi: https://doi.org/10.5281/zenodo.6957760. There the following files can be found:

- *BHR*
  - *BHRAbstractScraper.py* – This python code represents a scraper that extracts the following information from the open access server of the Humboldt-Universität zu Berlin about the documents in the BHR collection: author, title, language, identifier, abstract. For each BHR publication one CSV file will be created.
  - *BHRPDFscraper.py* – This python code represents a scraper that extracts and stores the PDF files of the BHR publications from the open access server of the Humboldt-Universität zu Berlin.
  - *BHRScraper.py* – This python code represents a scraper that extracts the following information from the open access server of the Humboldt-Universität zu Berlin about the documents in the BHR collection: author, title, language, type, subject, identifier. For each BHR publication one CSV file will be created.
  - *csvCombi.py* – This python code combines the CSV files created by *BHRAbstractScraper.py* or *BHRScraper.py* to one CSV file.
- *DABI*
  - *DABI_Mapping.txt* – This text file contains the mapping of the classes in the DABI classification system to the classes of the KBHR. The first string of numbers of each line represents a class in the DABI classification system. After a pipe symbol, the matching KBHR classes are listed.
  - *DABIScraper.py* – This python code represents a scraper that stores the following information from and about the publications in DABI: URL, database name, title, abstract, respective KBHR classes. For each publication one CSV file will be created.

[27] Köhler, J. (2022). Scrapers, Queries and Mappings for the Creation of a Dataset for Automatic Classification to the Classification System for the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft. Zenodo. https://doi.org/10.5281/zenodo.6957760

- *e-LiS*
  - *JITA_Mapping.txt* – This text file contains the mapping of the classes in JITA to the classes of the KBHR. The first string of numbers of each line represents a class in the KBHR. After a pipe symbol, the matching JITA classes are listed.
  - *JITAScraper.py* – This python code represents a scraper that stores the following information from and about the publications in e-LiS: URL, database name, title, abstract, respective KBHR classes. For each publication one CSV file will be created.
- *o-bib*
  - *obib_Anfragen.txt* – This text file contains the German queries for the search engine on o-bib and for each KBHR class. The first string of numbers of each line represents a class in the KBHR. After a semicolon, the matching queries are listed. Pluses are used instead of spaces, because the queries can directly be incorporated in the URL for the scraping process.
  - *ObibScraper.py* – This python code represents a scraper that stores the following information from and about the publications in o-bib: URL, database name, title, abstract, respective KBHR classes. For each publication one CSV file will be created.
- *Springer*
  - *Springer_Anfragen.txt* – This text file contains the German queries for the search engine on SpringerLink and for each KBHR class. The first string of numbers of each line represents a class in the KBHR. After a semicolon, the matching queries are listed. Pluses are used instead of spaces, because the queries can directly be incorporated in the URL for the scraping process.
  - *Springer_Queries.txt* – This text file contains the English queries for the search engine on SpringerLink and for each KBHR class. The first string of numbers of each line represents a class in the KBHR. After a semicolon, the matching queries are listed. Pluses are used instead of spaces, because the queries can directly be incorporated in the URL for the scraping process.
  - *SpringerScraper.py* – This python code represents a scraper that stores the following information from and about the publications in Springer: URL, database name, title, abstract, respective KBHR classes. For each publication one CSV file will be created.

- *Database_CSVCombiner.py* – This python code combines the CSV files created by *DABIScraper.py, JITAScraper.py, ObibScraper.py* and *SpringerScraper.py* to one CSV file.
- *Original_Dataset.csv* – The combined CSV files of all scraped documents from DABI, e-LiS, o-bib and Springer.

Appendix IV – Data Cleaning, Translation & Split

The data was uploaded to Zenodo[28] with the following doi: https://doi.org/10.5281/zenodo.6957842. There the following files can be found:

- *Data Cleaning*
  - *Cleaned_Dataset.csv* – The combined CSV files of all scraped documents from DABI, e-LiS, o-bib and Springer.
  - *Data_Cleaning.ipynb* – The Jupyter Notebook with python code for the analysis and cleaning of the original dataset.
- *Split*
  - *dataSplitDe* – This folder contains the German training, test and validation set.
    - *ger_test.csv* – The German test set as CSV file.
    - *ger_train.csv* – The German training set as CSV file.
    - *ger_validation.csv* – The German validation set as CSV file.
  - *dataSplitEn* – This folder contains the German training, test and validation set.
    - *en_test.csv* – The English test set as CSV file.
    - *en_train.csv* – The English training set as CSV file.
    - *en_validation.csv* – The English validation set as CSV file.
  - *splitting.py* – The python code for splitting a dataset into train, test and validation set.
- *Translation*
  - *DataSetTrans_de.csv* – The final German dataset as a CSV file.
  - *DataSetTrans_en.csv* – The final English dataset as a CSV file.
  - *translation.py* – The python code for translating the cleaned dataset.

[28] Köhler, J. (2022). Data Cleaning, Translation & Split of the Dataset for the Automatic Classification of Documents for the Classification System for the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft. Zenodo. https://doi.org/10.5281/zenodo.6957842

Appendix V – Klassifikationssystem für die Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft

1. Grundlagen & verwandte Gebiete
   1.1. Definitionen, Theorien, Modelle, Methoden & Standards
   1.2. Historische Aspekte
   1.3. Bildung & Ausbildung
   1.4. Verwandte Forschungsfelder & Disziplinen
2. Informationsorganisation & Bestandsmanagement
   2.1. Wissensorganisationssysteme
   2.2. Erschließung & Indexierung
   2.3. Bestandsaufbau & -konzeption
   2.4. Erwerbung & Bestandsevaluation
3. Gedächtnisinstitutionen & Informationsinfrastrukturen
   3.1. Arten
   3.2. Verbünde & Kooperationen
   3.3. Architektur & Technologien
   3.4. Management
   3.5. Informationsdienstleistungen
4. Informationsgesellschaft
   4.1. Informationskompetenz
   4.2. Gesellschaftliche Teilhabe
5. Informationsverhalten
   5.1. Informationsnutzung
   5.2. Informationssuche & -bedürfnis
   5.3. Human-Computer Interaction & User Experience
6. Informations- & Forschungsdatenmanagement
   6.1. Wissenschaftliches Publizieren
   6.2. Open Science
   6.3. Datenkuration & Langzeitarchivierung
   6.4. Informetrie & Wissenschaftsforschung
7. Informationssysteme & Informationsverarbeitung
   7.1. Design, Implementation & Management von Informationssystemen
   7.2. Evaluation von Informationssystemen
   7.3. Internettechnologien & -services
   7.4. Automatisierung, Data Mining & Künstliche Intelligenz

Appendix VI – Automatic Classification

The data was uploaded to Zenodo[29] with the following doi: https://doi.org/10.5281/zenodo.7043867. There the following subfolders and files can be found:

- *FLAIR* – This folder contains the python code for the FLAIR classifier as well as the output files for each setting.
    - *FLAIR_classifier.py* – The python code for the classifier using the FLAIR framework.
    - *FLAIR_one_label_classifier.py* – The python code for the one label classifier using the FLAIR framework.
    - *LR_0.03_maxE_150_Logs.out* – The output file of test of the FLAIR classifier with a learning rate of 0.03 and a maximum number of epochs of 150.
    - *LR_0.03_maxE_500_Logs_Logs.out* – The output file the test of the FLAIR classifier with a learning rate of 0.03 and a maximum number of epochs of 500.
    - *LR_0.75_maxE_150_Logs.out* – The output file the test of the FLAIR classifier with a learning rate of 0.75 and a maximum number of epochs of 150.
    - *LR_0.75_maxE_500_Logs.out* – The output file the test of the FLAIR classifier with a learning rate of 0.75 and a maximum number of epochs of 500.
    - *LR_0.85_maxE_150_Logs.out*– The output file the test of the FLAIR classifier with a learning rate of 0.85 and a maximum number of epochs of 150.
    - *LR_20_maxE_150_Logs.out* – The output file the test of the FLAIR classifier with a learning rate of 20 and a maximum number of epochs of 150.
    - *One_Label_LR_0.007_maxE_500_Logs.out* – The output file the test of the one label FLAIR classifier with a learning rate of 0.007 and a maximum number of epochs of 500.
- *Logistic Regression* – This folder contains the python code for Logistic Regression classifier as well as the output files for each setting.
    - *figures* – This folder contains the ROC and precision-recall-curves for the test results of each best model of the Logistic Regression classifier as png file.
    - *LogReg.py* – The python code for the Logistic Regression classifier.
    - *LogReg_Logs.out* – The output file of the test of Logistic Regression classifier.

[29] Köhler, J. (2022). Classification Experiments for the Automatic Classification of the Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft. Zenodo. https://doi.org/10.5281/zenodo.7043867

- *Naïve Bayes* – This folder contains the python code for the Naïve Bayes classifier as well as the output files for each setting.
    - *NB_SW_with_FS* – This folder contains the python code and the output file for the Naïve Bayes classifier that uses Feature Selection and specifically lemmatizes or stems the stop word list.
        - *figures* – This folder contains the ROC and precision-recall-curves for the test results of each best model of the Naïve Bayes classifier with Feature Selection and adapted stop word list as png file.
        - *NB_SW_with_FS.py* – The python code for the Naïve Bayes classifier with Feature Selection and adapted stop word list.
        - *NB_SW_with_FS_Logs.out* – The output file of the test of the Naïve Bayes classifier with Feature Selection and adapted stop word list.
    - *NB_SW_without_FS* – This folder contains the python code and the output file for the Naïve Bayes classifier that does not use Feature Selection and specifically lemmatizes or stems the stop word list.
        - *figures* – This folder contains the ROC and precision-recall-curves for the test results of each best model of the Naïve Bayes classifier without Feature Selection and adapted stop word list as png file.
        - *NB_SW_without_FS.py* – The python code for the Naïve Bayes classifier without Feature Selection and adapted stop word list.
        - *NB_SW_without_FS_Logs.out* – The output file of the test of the Naïve Bayes classifier with Feature Selection and adapted stop word list.
    - *NB_with_FS* – This folder contains the python code and the output file for the Naïve Bayes classifier that uses Feature Selection.
        - *figures* – This folder contains the ROC and precision-recall-curves for the test results of each best model of the Naïve Bayes classifier with Feature Selection as png file.
        - *NB_with_FS.py* – The python code for the Naïve Bayes classifier with Feature Selection.
        - *NB_with_FS_Logs.out* – The output file of the test of the Naïve Bayes classifier with Feature Selection.
    - *NB_without_FS* – This folder contains the python code and the output file for the Naïve Bayes classifier that does not use Feature Selection.

        - *figures* – This folder contains the ROC and precision-recall-curves for the test results of each best model of the Naïve Bayes classifier without Feature Selection as png file.
        - *NB_without_FS.py* – The python code for the Naïve Bayes classifier without Feature Selection.
        - *NB_without_FS_Logs.out* – The output file of the test of the Naïve Bayes classifier without Feature Selection.
- *SVM* – This folder contains the python code for the SVM classifier as well as the output files for each setting.
    - *SVM_SW_with_FS* – This folder contains the python code and the output file for the SVM classifier that uses Feature Selection and specifically lemmatizes or stems the stop word list.
        - *figures* – This folder contains the ROC and precision-recall-curves for the test results of each best model of the SVM classifier with Feature Selection and adapted stop word list as png file.
        - *SVM_SW_with_FS.py* – The python code for the SVM classifier with Feature Selection and adapted stop word list.
        - *SVM_SW_with_FS_Logs.out* – The output file of the test of the SVM classifier with Feature Selection and adapted stop word list.
    - *SVM_SW_without_FS* – This folder contains the python code and the output file for the SVM classifier that does not use Feature Selection and specifically lemmatizes or stems the stop word list.
        - *figures* – This folder contains the ROC and precision-recall-curves for the test results of each best model of the SVM classifier without Feature Selection and adapted stop word list as png file.
        - *SVM_SW_with_FS.py* – The python code for the SVM classifier without Feature Selection and adapted stop word list.
        - *SVM_SW_with_FS_Logs.out* – The output file of the test of the SVM classifier without Feature Selection and adapted stop word list.
    - *SVM_with_FS* – This folder contains the python code and the output file for the SVM classifier that uses Feature Selection.
        - *figures* – This folder contains the ROC and precision-recall-curves for the test results of each best model of the SVM classifier with Feature Selection as png file.

        - *SVM_with_FS.py* – The python code for the SVM classifier with Feature Selection.
        - *SVM_with_FS_Logs.out* – The output file of the test of the SVM classifier with Feature Selection.
    - *SVM_without_FS* – This folder contains the python code and the output file for the SVM classifier that does not use Feature Selection.
        - *figures* – This folder contains the ROC and precision-recall-curves for the test results of each best model of the SVM classifier without Feature Selection as png file.
        - *SVM_without_FS.py* – The python code for the SVM classifier without Feature Selection.
        - *SVM_without_FS_Logs.out* – The output file of the test of the SVM classifier without Feature Selection.
- *Summary_Outputs.xlsx* – An excel file in which the outputs of the classifiers (such as accuracy, F1 score, precision, recall, and run timTo e) are summarized.