RESEARCH ARTICLE

Using supervised learning to classify metadata of research data by field of study

Tobias Weber[1], Dieter Kranzlmüller[2], Michael Fromm[3], and Nelson Tavares de Sousa[4]

[1]Munich Network Management Team, Leibniz Supercomputing Centre (Germany)
[2]Munich Network Management Team, Ludwig-Maximilians-Universität München (Germany)
[3]Database Systems Group, Ludwig-Maximilians-Universität München (Germany)
[4]Software Engineering Group, Kiel University (Germany)

Keywords: fields of study, multilabel classification, research data, supervised machine learning

ABSTRACT

Many interesting use cases of research data classifiers presuppose that a research data item can be mapped to more than one field of study, but for such classification mechanisms, reproducible evaluations are lacking. This paper closes this gap: It describes the creation of a training and evaluation set comprised of labeled metadata, evaluates several supervised classification approaches, and comments on their application in scientometric research. The metadata were retrieved from the DataCite index of research data, pre processed, and compiled into a set of 613,585 records. According to our experiments with 20 general fields of study, multi layer perceptron models perform best, followed by long short-term memory models. The models can be used in scientometric research, for example to analyze interdisciplinary trends of digital scholarly output or to characterize growth patterns of research data, stratified by field of study. Our findings allow us to estimate errors in applying the models. The best performing models and the data used for their training are available for re use.

1. INTRODUCTION

By research data, we understand all digital input or output of those activities of researchers that are necessary to produce or verify knowledge in the context of the sciences and humanities[1]. Fields of study are concepts used to structure academic institutions hosting these activities into faculties or to characterize the branch of knowledge that a journal contributes to; typically, fields of study are not applied identically on every occasion, but the different classification schemes overlap substantially.

Metadata for research data often include a title or even a longer description that permits identification of the field of study of the research data item and can thus be used as a placeholder to classify it. Classification in this context can be understood in two ways: Either an item must be mapped to one, and *only one*, class out of a fixed set of classes, in which case this task is called a *multiclass* classification in the literature, or the item may be mapped to *one or several* classes, which is called *multilabel* classification. The latter approach is more appropriate for research data, because they can often be mapped to multiple fields of study and these mappings are typically *not* exclusive (research data can belong to both statistics and economics,

[1] Definition according to Weber and Kranzlmüller (2018).

an open access journal

Check for updates

Citation: Weber, T., Kranzlmüller, D., Fromm, M., & de Sousa, N. T. (2020). Using supervised learning to classify metadata of research data by field of study. *Quantitative Science Studies*, *1*(2), 525–550. https://doi.org/10.1162/qss_a_00049

DOI: https://doi.org/10.1162/qss_a_00049

Received: 15 October 2019
Accepted: 28 February 2020

Handling Editor: Ludo Waltman

Corresponding Author: Tobias Weber
mail@tgweber.de

The MIT Press

or to both medicine and biology). As both the amount and the growth of research data are too extensive for manual routines to classify them, automatic classifiers are needed for many use cases[2].

Three of these use cases illustrate the usefulness of such an automated classifier and help to specify the requirements for the methodological approach of this paper:

1. **Scientometric research**: During scientometric analyses, normalization problems arise. Metrics for publications are an example: To compare the values (citations, usage, etc.) across fields, these must be normalized to values typical for a specific field or community of research. Only automated classification procedures allow scientometricians to find these values for large and unclassified data sets. An automated classifier can also be used to take stratified samples from large collections of research data.
2. **Assistant systems**: Providers of research data services can take advantage of automated classification to assist users of their repositories. Assistant systems can suggest labels for research data based on the metadata submitted. This not only eases the work of the submitters, users, and curators but improves the overall quality of available metadata.
3. **Value-adding services**: Research data aggregation services, such as DataCite[3] or BASE[4] collect metadata across different fields of study. Enriching the collected metadata by adding classification information enables value-adding services such as a faceted search, publication alarms for specific fields, or other services relevant only to a selection of fields.

The three use cases have different requirements: When an automated classifier is used in scientometric research (e.g., for sampling research data), a wrongly assigned label has a greater impact on the application compared to a missed label. Assistant systems, on the other hand, ideally identify all correct labels; as humans can correct the suggestions in this context, a wrongly assigned label is not as bad as a missed label. The former use case therefore stresses the precision of the classifier, which is the probability that a classification is correct, while the latter use case stresses its recall, which is the probability that no correct label is missed by the classifier. In the third use case (value-adding services) both qualities are equally important.

This paper reproducibly evaluates different approaches to automatically classify research data along the requirements of the three use cases. To achieve that, the following steps have been taken:

1. Retrieval of openly available metadata for research data;
2. Crosswalks to map the different classification schemes to one common scheme;
3. Extraction of title, description, and keywords from the metadata;
4. Cleaning and preparation of the data for the application of machine learning algorithms;
5. Evaluation of different approaches along the three use cases presented above.

[2] For example, Bell, Hey, and Szalay (2009) or Peters, Kraker, et al. (2017).
[3] https://datacite.org/
[4] https://base-search.net/

As evaluation candidates we used the multilabel-enabled classifiers provided by the scikit-learn framework[5] and neural networks as supported by tensorflow[6].

The main contributions of this paper are

- A methodical evaluation of selected classification models for each of the use cases;
- The publication of the data set used for this evaluation, which allows others to reproduce or supersede our findings;
- The publication of the complete source code used to clean the data and map them to our base classification scheme;
- Suggestions on how these models can be used for scientometric research.

The remainder of this paper is structured as follows: Section 2 discusses the relations of our approach to a selection of published work. In Section 3 we summarize how the data set was retrieved and processed; the processing includes the identification of a common classification scheme and the mapping of the research data onto this scheme. The machine learning models and data vectorization workflows are introduced in Section 4. The methodological approach to evaluate the models is described in Section 5. Section 6 gives an overview of the results. These results are discussed in Section 7, which includes a comment on threats to validity. The last section concludes and suggests how to take advantage of our findings in scientometric applications.

2. RELATED WORK

This section lists approaches to realize an automated classification of research data by field of study and their common shortcomings and discusses the available data to evaluate the approaches.

2.1. Automatic Classification by Field of Study

Waltinger, Mehler, et al. (2011) use a Support Vector Machine (SVM) model to classify according to the Dewey Decimal Classification (DDC), in three hierarchy levels. The problem is described as a multilabel classification task with hierarchical predictions. Hierarchical predictions presuppose a classification scheme with different levels (e.g., science on the first level and biology, physics, etc. on the second level). The presented prediction scores are only partial due to the sparseness of the used training set on some levels, which was compiled out of the data available via the BASE service (Bielefeld Academic Search Engine)[7]. at that time: On the first level of the DDC hierarchy 5,868 English and 7,473 German metadata records were available, and on the second and third levels 20,813 English and 37,769 German metadata records were available. The English classifier had an f_1-score of 0.81 (classification over base classes; that is, 10 labels); for the deeper levels only partial data are available. It is not specified, whether the score is averaged over the whole data set (micro) or the mean scores for each label (macro). In comparison to Waltinger et al. (2011), our data set is more than 20 times larger.

[5] This framework only includes well-established models and algorithms; see Pedregosa, Varoquaux, et al. (2011).

[6] Tensorflow offers hardware support for graphic processing units (GPUs) to speed up the training of neural networks, and is generally better equipped for neural networks than scikit; see Abadi, Agarwal, et al. (2015).

[7] https://base-search.net/

Wang (2009) also discusses the application of machine learning algorithms on bibliographic data labeled with DDC numbers. They evaluate their approach with a data set comprised of publications[8], (i.e., journal articles, conference papers, books, and book chapters). Their classification scheme is limited to the DDC classes 500 and 600 (science and technology) with 88,400 mostly single-labeled records. The author suggests flattening the hierarchy to reduce the number of labels (18,462). Although the proposed classification approach achieves an accuracy score of nearly 90%, it is not a viable option for our use cases, as it is based on a multiclass approach (classification over multiple labels, which are taken as exclusive) and not fully automated. As already stated, that approach is limited to a relatively narrow selection of fields of research.

Another approach to use SVM models to predict DDC fields of study is presented by Golub, Hagelbäck, and Ardö (2018). The authors characterize the problem as multiclass, but their classifier honors the hierarchy of DDC. The used data set includes 143,838 records from the Swedish National Union Catalogue (joint catalogue of the Swedish academic and research libraries). They report a peak accuracy of 0.818.

In general, classifiers targeting DDC (Golub et al., 2018; Waltinger et al., 2011; Wang, 2009), face the problem that predicting the first DDC level is typically not very useful (only 10 classes, one of which is "Science"), whereas classifiers targeting DDC's second level need to provide training data for 100 labels. For the latter task to succeed, the reported data sets are too small or too sparsely populated on certain labels; the results are as a consequence partial at best. Our analysis of the DataCite index furthermore indicates that DDC is not necessarily the most used classification scheme for research data, despite its popularity among information specialists and librarians (see Table 2). These problems could be circumvented by using a classification scheme that is expressive enough in the first level, as proposed by us. As a conclusion of the review of the literature we furthermore decided to not include hierarchy predictions in our problem. Hierarchies add a second layer on top of the classification problem at hand: Classifying base classes is one layer and determine the depth in a hierarchy another. The second layer could itself be understood (recursively) as a multilabel classification problem. While we hope to contribute to the former, we do not claim to solve the latter.

All the approaches that we found in the literature have at least one of the following shortcomings:

- The classification task is characterized as multiclass, that is, selecting *only one* field of research, and not multilabel, which would include the possibility to select *more than one* field of research per research data item.
- The reported classification performance is not comparable to other approaches, because important values are missing or reported values are too unspecific.
- The domain of classification only includes publications in the classical sense (journal articles, conference papers, books, and book chapters) as opposed to the more general class of research data (also including tabular data, source code, models, etc.).
- The evaluation of the approaches is limited to a subset of the possible base classes or labels.
- The classification routine is not automated (i.e., includes human interaction).

[8] Publications are considered a subclass of research data in this paper.

Our approach shares none of the named shortcomings. The literature furthermore concentrates on linear machine learning models (most prominently SVMs), which is why we concentrated our resources on other models and excluded them from the evaluation.

2.2. Data Publications to Evaluate Classification Approaches

A general problem we found in the course of reviewing the available literature is that the reported results are often not comparable; incomplete or incommensurable performance metrics are not the only issue: The values could have been recalculated if the data were available. With one exception (Joorabchi & Mahdi, 2011), all the publications we found do not include enough information to retrieve the data used to evaluate the presented approach. Additionally, different data sets probably lead to different results; there is no single, canonical data set which is used to evaluate the different approaches.

Lösch, Waltinger, et al. (2011) present an approach to compile an annotated corpus of metadata based on the OAI-PMH standard, the Dublin Core metadata scheme, and the DDC classification scheme. The authors created a partially manual routine to determine the DDC label[9]. The resulting data set includes 52,905 English records annotated with one of the 10 top-level DDC classes. We build on their approach, but adapt it by using the DataCite metadata scheme[10], which supports qualified links to classification schemes. This allowed us to compile a larger data set with a finer set of base classes and the possibility to integrate different classification schemes into our approach. We found the resulting data set in a similar imbalance as the data set presented by Lösch et al. (2011) (see Section 3).

We hope to contribute not only with our classification approach but also by providing a large data set that can be used to evaluate future approaches and reproduce the findings of already proposed approaches.

3. DATA RETRIEVAL AND PROCESSING

This section explains the steps that were taken to create the training and evaluation set. Figure 1 lists the steps and specifies the size of the outcome in metadata records after each step. The following subsections are aligned with the workflow depicted in Figure 1.

3.1. Data Source: The DataCite Index of Metadata

The training and evaluation data have been retrieved from the DataCite index of metadata of research data[11] via OAI-PMH[12]. DataCite is a service provider that aggregated research data over more than 1,100 publishers and 750 institutions in 2017 (Robinson-Garcia, Mongeon, et al., 2017). An analysis of the publishers present in our final training and evaluation set gives evidence that the number of publishers has grown since then. DataCite metadata are openly available and they mirror a broad range of institutions, fields of study, and countries, although the distribution is skewed to data depositions in figshare and to German institutions (cf. Table 1). The heterogeneity of this data source allows us to create a training and evaluation

[9] See also Bäcker, Pietsch, et al. (2017).
[10] See DataCite Metadata Working Group (2019).
[11] https://datacite.org/
[12] http://www.openarchives.org/OAI/openarchivesprotocol.html

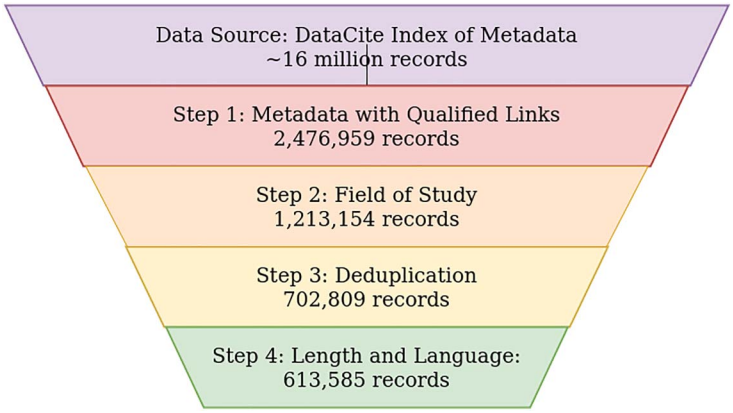

Figure 1. Overview of the retrieval and cleaning process.

set that includes data publications curated both by information professionals[13] and by researchers themselves[14].

DataCite is not only the name of the index but also the name of a metadata schema[15]. In the data retrieved from the index, a broad range of versions of this schema are present; all versions prescribe six mandatory fields (identifier, title, creator, publisher, year of publication, and type of resource). The remaining heterogeneity of the retrieved records does not carry too much weight for our purpose (to create a training and evaluation set), because there are continuities across the different versions:

- All versions prescribe a Digital Object Identifier (DOI), which allows us to easily identify some types of duplicates.
- Specifying a title is mandatory in all versions.
- The way descriptions and subjects are specified has been very similar since version 2.0[16].

The data retrieval took place in May 2019. The total number of items in the index at this time was approximately 16 million records[17]. In February 2020, the DataCite index already included more than 20 million metadata records for research data.

3.2. Step 1: Metadata with Qualified Links

The retrieved metadata include records uploaded from June 2011 to May 2019. We used a customized GeRDI-Harvester[18] to retrieve only those metadata in DataCite format that were qualified records; a qualified record is understood as a metadata record with at least one

[13] It is assumed that the contents provided by libraries (Table 1) are curated by information specialists.
[14] It is assumed that researchers usually upload their data on platforms such as figshare or the Open Science Framework (OSF) without an intermediary.
[15] Specified by DataCite Metadata Working Group (2019).
[16] All versions specify the same scheme for descriptions (including the specification of a type of description) and subjects (including the specification of a scheme of subjects). Later versions add new syntactical elements that do not break the general specification of descriptions and subjects.
[17] The exact number is not available due to the length of the time span over which the harvesting process took, in which ingests to and deprovisions from the DataCite index took place.
[18] Generic Research Data Infrastructure, https://www.gerdi-project.de/

Table 1. Research data publishers in the final training and evaluation set

Publisher	Records	Percentage
Figshare	374,903	61.10%
Leibniz Institut für Astrophysik Potsdam (AIP)	121,782	19.85%
German Medical Science GMS Publishing House	17,216	2.81%
Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik GmbH, Wadern/Saarbruecken, Germany	7,110	1.16%
Universität Stuttgart	7,032	1.15%
Technische Universität Berlin	6,477	1.06%
Universitätsbibliothek Tübingen	4,658	0.76%
Deutsches Elektronen-Synchrotron, DESY, Hamburg	3,960	0.65%
Universitätsbibliothek der Ludwig-Maximilians-Universität München	3,900	0.64%
Technische Universität Dortmund	3,224	0.53%
RWTH Aachen University	3,184	0.52%
The American Physical Society	2,587	0.42%
Universität des Saarlandes	2,581	0.42%
Universitäts- und Landesbibliothek Sachsen-Anhalt	2,350	0.38%
PsyArXiv	2,161	0.35%
Open Science Framework	1,730	0.28%
SocArXiv	1,623	0.26%
Humboldt-Universität zu Berlin, Wirtschaftswissenschaftliche Fakultät	1,492	0.24%
Kentucky Transportation Center, University of Kentucky	1,458	0.24%
Humboldt-Universität zu Berlin	1,422	0.23%
Barcode of Life Data Systems	1,383	0.23%
Universität Tübingen	1,379	0.22%
INA-Rxiv	1,313	0.21%
Technische Informationsbibliothek u. Universitätsbibliothek	1,283	0.21%
Humboldt-Universität zu Berlin, Mathematisch-Naturwissenschaftliche Fakultät I	1,229	0.20%

Note: $n = 5781$ (not checked for duplicate entries of institutions; only the 25 most occurring publishers are listed here).

Table 2. Supported classification schemes

Name	Records (after cleaning)
Australian and New Zealand Standard Research Classification (ANZSRC)	374,948
Dewey Decimal Classification	219,946
Digital Commons Three-Tiered Taxonomy of Academic Disciplines	11,683
Basisklassifikation	7,028

Note: n = 613,585; a record can be qualified by more than one scheme.

subject field that is qualified either with a URI to a scheme (DataCite specifies a schemeURI attribute) or a name for a scheme (DataCite specifies a subjectScheme attribute).

In sum, 2,476,959 metadata records are qualified records.

3.3. Step 2: Fields of Study

Five classification schemes for fields of study are frequently used throughout the retrieved metadata; our method supports four of them (see Table 2). The scheme missing from the table is linsearch, which is a classification scheme that is derived from automatic classification (Bähr, Hannover, & Denecke, 2008; Waltinger et al., 2011). We decided to exclude the linsearch scheme to avoid amplifier effects: Data sets that are classified by machine learning models are necessarily biased toward the model used in particular and what a machine can classify in general.

The classification scheme we used (see Table 3) is identical to the base classes of the most common scheme, the Australian and New Zealand Standard Research Classification (ANZSRC), except for two adaptions: Two pairs of classes were merged to map the other classification scheme to the common classification scheme:

- "Earth Sciences" and "Environmental Sciences" are divisions 04 and 05 respectively of the ANZSRC classification scheme and became "Earth and Environmental Sciences" in the common classification scheme.
- "Engineering" and "Technology" are divisions 09 and 10 respectively of the ANZSRC classification scheme and became "Engineering and Technology" in the common classification scheme.

These merges enabled a mapping from the other classification schemes to ANZSRC without arbitrary splits or losing records due to mismatches of the schemes. The resulting classification scheme has been flattened (projection to the base classes) and therefore has no hierarchy.

The supported schemes (Table 2) were mapped in crosswalks to the classification scheme of this paper (Table 3). The crosswalks were created after the consultation of the available documentation of the schemes found in Table 2 and adapted after checking a sample of several resulting labels. A schematic overview is hard to create in the context of this paper, but the exact mapping rules (more than 1,700 lines of code) are available for analysis and improvement[19].

[19] See the file code/clean/cleanDataHelpers.py in Weber and Fromm (2019).

Table 3. Fields of study and their frequency of occurrence in the retrieved metadata

Class	1 label	2 labels	3+ labels	best	total	%	ø#labels	øwc	wc (med.)
Mathematical Sciences	9,138	13,633	23,719	45,916	46,490	7.58	2.55	111	59
Physical Sciences	138,231	8,559	13,418	138,231	160,208	26.11	1.26	48	21
Chemical Sciences	16,337	27,091	37,953	57,052	81,381	13.26	2.44	141	105
Earth and Environmental Sciences	13,327	24,747	35,344	57,042	73,418	11.97	2.48	144	97
Biological Sciences	67,861	88,643	71,184	86,305	227,688	37.11	2.11	124	66
Agricultural and Veterinary Sciences	1,851	891	420	3,125	3,162	0.52	1.62	141	75
Information and Computing Sciences	27,688	15,613	27,076	51,915	70,377	11.47	2.17	114	73
Engineering and Technology	25,096	6,453	2,195	29,845	33,744	5.50	1.35	165	146
Medical and Health Sciences	68,096	46,710	42,666	86,304	157,472	25.66	1.93	134	77
Built Environment and Design	1,795	1,107	359	3,173	3,261	0.53	1.61	147	87
Education	2,476	1,333	1,258	4,865	5,067	0.83	1.91	124	99
Economics	5,209	1,241	1,119	6,635	7,569	1.23	1.62	151	133
Commerce, Management, Tourism and Services	5,126	1,122	495	6,156	6,743	1.10	1.36	132	116
Studies in Human Society	6,653	4,178	1,284	9,219	12,115	1.97	1.65	137	129
Psychology and Cognitive Sciences	11,407	4,701	1,805	15,312	17,913	2.92	1.52	138	138
Law and Legal Studies	1,045	181	147	1,332	1,373	0.22	1.42	173	155
Studies in Creative Arts and Writing	1,106	281	324	1,497	1,711	0.28	1.58	142	107
Language, Communication and Culture	4,432	929	606	5,265	5,967	0.97	1.40	117	89
History and Archaeology	2,205	599	266	2,825	3,070	0.50	1.39	72	19
Philosophy and Religious Studies	474	724	391	1,571	1,589	0.26	2.02	124	111
Total	409,553	124,368	79,664	–	613,585	–	1.50	112	57

Note: Columns 2 labels and 3+ labels do not sum to their total as records are counted twice, or three times or more times.

After this step 1,213,154 records remained; 1,263,805 records were filtered out as "not annotatable" (i.e., there was no mapping to a field of study available). Typical reasons for a missing mapping include unclear identification of the source scheme and a different domain of the scheme, meaning the scheme is not (solely) designed to classify fields of study.

3.4. Step 3: Deduplication

The deduplication of qualified records was realized along two criteria: identity of identifier (DOI) and identity of payload[20]. After this step 702,809 records remained; 510,345 records

[20] As the data processing was parallelized, not all other records were available for comparison at all times. Some deduplications therefore only happened after step 4; step 3 and step 4 are thus conceptually separated, but overlap in the source code.

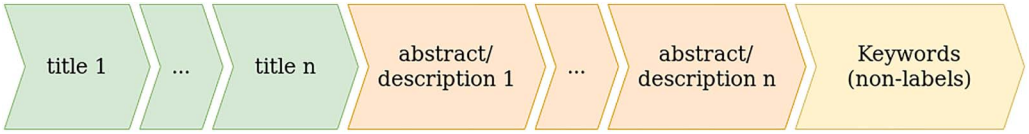

Figure 2. Concatenation of the payload parts.

were filtered out as duplicates. The high number of duplicates can be explained by the occurrence of DOI-versioning[21], which is the publication of several versions of a research data item under different DOIs and one "concept DOI," which maintains references to the different versions. As a result, a lot of payloads are identical if the title, description and subject tags do not change over versions.

3.5. Step 4: Length and Language

In the last step the payload was created from the following three components:

- one or more titles (title is a mandatory field);
- zero or more descriptions; and
- the subset of the subjects/keywords of the research data containing only those fields that have *not* been used to determine the labels.

The order of concatenation is depicted in Figure 2. Only those components are included that consist mainly of English words[22]. The language detection module relies on a seed to reproduce the same language detection[23]. If the resulting string contained fewer than 10 words (separated by white space), it was discarded.

After this step 613,585 records remained; 89,224 were filtered out as not fit for our purpose.

3.6. Overview of the Resulting Training and Evaluation Set

Table 3 provides some statistics for the resulting training and evaluation data set:

- **1/2/3+ label(s)**: all metadata records with exactly one, two, and three or more labels;
- **best**: all metadata records for which this field of study is the best label for stratification (see below);
- **total**: all metadata records labeled with this field of study (the sum of all these values is greater than the number of final records, because a record can have more than one label);
- **%**: percentage of metadata records labeled as this field, rounded to two decimal places;
- ø**#labels**: arithmetical mean of the number of labels per record of the records including that label (rounded to two decimal places);
- ø**wc**: arithmetical mean of the number of words per record with that label (rounded to a whole number); and
- **wc (med.)**: median of the number of words per record with that label.

[21] See Nowak, Ioannidis, et al. (2018).

[22] We used a python port of the langdetect library (see citetlangdetect) to determine the language of the fields: https://pypi.org/project/langdetect/

[23] We used the randomly created seed 1914088464.

The distribution of the different fields of study is unbalanced, which is in accordance with the findings in the literature[24]. The imbalance of labels necessitates additional thought on the selection of evaluation metrics in Section 5 and on the configuration of the different models.

Payloads labeled as "Physical Sciences" and "History and Archaeology" are noteworthy outliers, because they have fewer words compared with records from other fields. The average number of labels is highest in "Mathematical Sciences"; this can be explained by the fact that statistics is part of this category: Using certain statistical methods justifies this label, and those methods are used in a broad variety of fields. With one exception all the fields of study have a distribution of word counts that is skewed to shorter payloads (median < mean), meaning that there are lengthwise outliers. "Psychology" shows the same mean as median.

The label cardinality (average number of labels per record) is 1.5; the label density (average proportion of labels per record) is 0.07[25]. The data set includes 952 different labelsets (combinations of labels for a research data item); this number is relatively small compared to the 2^{20} theoretically possible labelsets.

A challenging issue is the stratified split of training and evaluation set to guarantee the same distribution of fields of study in both subsets: 171 labelsets occur only once, which makes a stratified split along the 952 labelsets impossible. To enable stratified splitting we followed a "best label" approach:[26]

1. All records with only one label are assigned that label.
2. Iterating over the remaining records, the "best" label out of the labelset is selected, which is the label that is selected the least often at the current state of the loop.

These "best" labels are used in stratified sampling and feature selection (see following section), but *not* as labels for the training itself.

4. VECTORIZATION AND MODEL SELECTION

In this section the vectorization methods and evaluated models are presented. We evaluated two ways to vectorize the payloads described in the last section, they are presented in Section 4.1. The models we used on these vectorized payloads are described in Section 4.2. As a result three different combinations are evaluated:

1. "Classic" machine learning models combined with bag of words vectorization; approaches that have already been evaluated (linear models) or that do not support multilabel classification natively have been excluded;
2. Multilayer perceptrons, combined with bag of words vectorization;
3. A model used in contemporary Natural Language Processing, the Long Short Term Memory (LSTM) model, combined with word embedding vectorization.

[24] See, for example, Golub et al. (2018), Kraker, Lex, et al. (2015), Peters, Kraker, et al. (2016), or Waltinger et al. (2011).

[25] See Tsoumakas and Katakis (2009) for a formal definition.

[26] Sechidis, Tsoumakas, and Vlahavas (2011) propose an algorithm to realize a "relaxed interpretation of stratified sampling for multilabel data." The base idea is to distribute the data items over several subsets, starting with all data items labeled with the least common label (greedy approach). As our data include a substantial part with only one label, we took the following approach, which is easier to implement.

The section concludes with a short description of the hyper-parameter tuning of the model/vectorization combinations.

4.1. Vectorization

Before vectorization, the data are split into a training set (552,226 records) and an evaluation set (61,359 records) with a ratio of 9:1. The split is stratified (i.e., the distribution of the "best" labels in the test and training sets is approximately identical). For the deep learning models the training set is again split by the same approach (497,003 training and 55,223 validation records)[27]. Both vectorized data sets (see following paragraphs) are split identically (i.e., they were derived from the same data sets to gain comparable results). The split and the vectorization are executed three times, for small (s), medium (m), and large (l) vectorized representation of the payloads; the definitions of the sizes will be given in the following sections.

4.1.1. Bag of Words (BoW)

One way to vectorize the corpus of all documents (i.e., all payloads of the metadata records) is the "Bag of *n*-grams"-approach; that is, each document is treated as a row in a matrix in which the columns are the terms (1-grams and 2-grams). Some terms were filtered out by a stop word list[28]. A stop word list is designed to filter out noninformative parts of the documents. We chose to create our own stop word list (with 240 entries). The list of stop words includes

- words that are generally considered unspecific (e.g., "the," "a," "is");[29]
- numbers and numerals ≤ 10;
- words that are unspecific in the context of data (e.g., "kb," "file," "metadata," "data"); and
- words that are unspecific in the academic context (e.g., "research," "publication," "finding").

For each term *t* in a document—that is, for each cell of the matrix of documents and terms—the term frequency-inverted document frequency (tf-idf) was calculated using the default settings of scikit-learn[30] (except for the stop word list). Tf-idf does not solely rely on occurrences of any given term, but also takes its frequency of occurrence throughout the document space into calculation. Therefore, the final td-idf value of a term decreases if the term occurs in multiple documents. This reduces the weight of terms with broader usage.

The vectorization resulted in 4,087,639 possible features.

[27] Deep learning models typically need validation sets during training. These validation sets are also stratified by "best" label (see above).

[28] Nothman, Qin, and Yurchak (2018) voice concerns with regard to stop word lists in vectorizing text data (controversial words, incompatible tokenization rules, incompleteness). Additionally, the contextuality of stop word lists is a problem: If the context of a set of documents is given, certain words are likely to lose discriminatory potential, although they would not qualify as stop words in a more general context. We decided to extend an existing stop word list, but specify it for reproducibility. This allows us to take advantage of the context of data.

[29] These stop words are a subset of the English stop words list of the nltk software package; see Bird, Klein and Loper (2009).

[30] Pedregosa et al. (2011).

The best features were selected in three modes:

- **s**: 1,000 features per label, 20,000 features in total;
- **m**: 2,500 features per label, 50,000 features in total; and
- **l**: 5,000 features per label, 100,000 features in total.

The selection is based on an ANalysis Of VAriance (ANOVA) of the features[31]. This allows us to identify the features that are best suited to discriminate between the classes. This is the second and last time that the "best" labels were used.

4.1.2. Word embeddings

Another approach for vectorization is using word representations, like word2vec[32]. Such an approach can utilize either continuous bag-of-words (CBOW)[33] or continuous skip-gram[34]. The CBOW model predicts the current word from a window spanning over context words. The skip-gram model uses the current word to predict the context surrounding the word. In our work we used pretrained word2vec embeddings[35] that were trained on the Google News data set (about 100 billion words). The embeddings were trained with the CBOW approach and consist of 300-dimensional word vectors representing three million words. Compared to BoW, word embeddings provide a low-dimensional feature space and encode semantic relationship among words.

To apply the vectorization method, each document needs to be tokenized:

- **s**: up to 500 words of the document are tokenized;
- **m**: up to 1,000 words of the document are tokenized; and
- **l**: up to 2,000 words of the document are tokenized.

4.2. Machine Learning Models with Support for Multilabel Classification

4.2.1. Classic machine learning models

We did not include linear models in our evaluation, because preliminary tests did not indicate that they could reach the performance of neural networks. As linear models such as Support Vector Machines (SVM) have already been used for the task at hand, we decided to exclude them for the evaluation. The published data and source code allow others to supersede our results with a linear model and a combination of hyperparameters that we might have missed[36].

- *DecisionTreeClassifier*[37]: This classifier uses a decision tree to find the best suited classes for each record. The nodes of the tree are used to split the records into two sets based on a feature, eventually resulting in a leaf that ideally represents a certain class or a certain labelset. The training consists in finding the best features to split the data set. Multilabel classification of unseen data works by following the decision path until a leaf is reached. All labels that are present in the majority of the data items in the leaf are returned as the classification result.

[31] Fisher (1973).
[32] Mikolov, Sutskever, et al. (2013).
[33] citetcbow.
[34] Mikolov et al. (2013b).
[35] https://code.google.com/archive/p/word2vec/
[36] See Duan and Keerthi (2005) for an approach to testing SVMs.
[37] Breiman, Friedman, et al. (1984).

- *RandomForestClassifier*[38]: This classifier is based on the DecisionTreeClassifier, by building an ensemble of multiple decision trees. The general idea is that a bias that trees typically have by overfitting a training set is remedied by building multiple decision trees based on different random subsets of the features. In ideal cases the bias in different directions corresponds to different aspects of the data. The training consists in fitting *n* trees by using a random subset of features and a random subset of the training set. The classification is then achieved by a voting procedure among the trees in the ensemble.
- *ExtraTreesClassifier*[39]: This classifier is similar to the RandomForestClassifier. Both are ensembles of trees, but this classifier is based on the ExtraTreeClassifier (note the missing *s*) which introduces more randomness by selecting the feature to split by totally at random. (DecisionTreeClassifiers in random forests, by contrast, select the best feature out of a subset sampled at random).

4.2.2. Multilayer Perceptron (MLP)

A multilayer perceptron (MLP)[40] is a neural network and consists of one input layer, one output layer and *n* intermediate layers of perceptrons. The input layer corresponds to the vectorized data (e.g., a vector of 20,000 values in the s-sized BoW approach) and the output layer has the shape of the labels (i.e., a vector with 20 elements). Backpropagation is used to train the the model for the given data; we used the Adam optimizer for this task[41]. By using a sigmoid function as the activation function of the output layer, the MLP can predict the probability for multiple labels (multilabel).

4.2.3. Recurrent network

Recurrent networks form a class of neural networks that are used to process sequential data of different length. This design furthermore allows it to make use of temporal dynamic behavior (e.g., recurrences of terms in a text). The recurrent network architecture we use in our work is an Bidirection Long Short-Term Memory (BiLSTM) model[42]. We use word2vec[43] embeddings as described in Section 4.1.2 as input to the model[44]. The embeddings are frozen and not further trained in the classification process. On top of the BiLSTM layer we use a dense layer with a sigmoid activation function to classify multilabels. The BiLSTM layer and the dense layer are trained by an Adam optimizer.

4.3. Weights and Hyper-Parameter Tuning

An important parameter in the training of multilabel classification problems based on unbalanced training sets is the weight given to each label. All the classes of algorithms we used allow us to give more weight to underrepresented labels. We calculated the weights based on the label frequencies found in the training set:

$$\text{weight}(\text{label}) = \frac{1}{\frac{\text{frequencies}(\text{label})}{\max(\text{frequencies})}}.$$

[38] Breiman (2001).
[39] Geurts, Ernst, and Wehenkel (2006).
[40] Rumelhart, Hinton, et al. (1988).
[41] Kingma and Ba (2014).
[42] Hochreiter and Schmidhuber (1997).
[43] Mikolov et al. (2013).
[44] Any other kind of word embeddings can be used too.

For each model we executed semi automated parameter tuning, following this procedure:

1. List selected hyper-parameters in the order of expected impact to the evaluation metrics.
2. For each hyper-parameter or combination thereof, execute a grid search to find the best candidate(s).
3. Fix the selected parameters and repeat step 2 with the next parameter or parameter combination.

The space of possible solutions is too big to be exhaustively searched with a reasonable use of resources, so it might be that using a different combination of hyper-parameters results in better scores than reported by us.

5. EVALUATION PROCEDURE

For each use case (see Section 1) an evaluation metric has been identified that takes the imbalance of the label distribution into account, as stated in Section 3. The use cases differ in the weight they put on recall and precision. Recall for label *l* is the ratio between true positives and positives for label *l* (sum of true positives and false negatives for *l*). Precision for label *l* is the ratio between true positives and predicted positives (sum of true positives and false positives for *l*)[45].

These values alone are easy to game, which is why they should be combined: The *fβ* -score puts them in a relation to each other that allows us to modify the weights we give to precision and recall respectively:

$$f_\beta = \left(1+\beta^2\right) \cdot \frac{\text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}}.$$

The value of β controls the weight give to recall and precision:

- If $\beta < 1$, precision is highlighted; a value of 0.5 has been chosen for the "scientometric research" use case.
- If $\beta > 1$, recall is highlighted; a value of 2 has been chosen for the "assistant system" use case.
- If $\beta = 1$, precision and recall are treated equally; this is chosen for the "value-adding services" use case.

Precision and recall are calculated for each label, and the arithmetical mean over all labels is taken as the input for the calculation of the f_β-score. This *macro* average approach takes the imbalance of the base classes (and therefore of the labels) into account. It can be interpreted as the chance of a correct classification when a stratified sample is drawn. This is the basis for the evaluation of the approaches for the presented use cases.

The *micro* average approach averages the values over all data sets, without the intermediate aggregation over the labels. It can be interpreted as the chance of a correct classification when a completely random sample is drawn. In unbalanced scenarios, micro scores tend to be skewed by the predominant labels; because these often perform better (more training data), micro scores are often too optimistic when the performance of the model with regard to all

[45] Formal definitions are provided by Sorower (2010).

labels is the target. Although microaverages are not used in this paper to evaluate the models, we nevertheless report them for the sake of comparability.

We refrain from reporting accuracy (ratio of the sum of true positives and true negatives for label *l* to the size of the evaluation set), as it is biased toward negative classifications, which in our case are much more frequent than positive classifications.

The final evaluation is based on the f_β-macroscores calculated on all three evaluation sets. In this way, each model is tested against the same unseen set of data (see Section 3).

6. RESULTS

6.1. Model Performance

For each of the evaluated model and data pairs the best performance results are shown in Table 4. The macro and micro scores are derived from the same models and the models are selected by their best $f_{0.5}$, f_1, and f_2 scores, respectively.

For all three use cases the MLPClassifier was the best performing model according to our tests (cf. Table 4). The MLPClassifiers trained on the l-sized data are in almost all cases better performing than the model trained on the m-sized data, with one exception (the value-adding service use case/f_1) and only by a small amount. With one exception (the best performing f_1-models trained on s-sized data), all LSTMClassifiers perform worse compared to their MLPClassifier counterparts, and they took essentially longer to train than the MLP models. The lead of the MLPClassifier might be explained by the number of short payloads and the missing structure of the payloads in the data set: The median word count (57 words) is left of the mean (111.9 words), and 25 words is the value of the first quartile[46]. As shown in Figure 2, title(s), description(s), and keyword(s) are simply concatenated—the resulting payload therefore has no "macro" structure typical for texts. Many of the records' payloads might be too short and too unstructured for the LSTM model to play out its ability to detect semantic relationships beyond the statistical approach used by the MLP based on BoW.

The results of tree-based models are far behind the results achieved by the deep learning models. Trees and Ensembles perform best on s-sized data, and with the exception of DecisionTreeClassifiers, all models performed better in terms of precision than in recall ($f_{0.5}$ scores are greater than f_2 scores).

6.2. Use Cases

6.2.1. Scientometric research

The performance scores of the best model for the use case "scientometric research" ($f_{0.5}$-scores) are shown in Table 5;[47] this table lists the scores for each field of study. The $f_{0.5}$-score correlates positively with the number of records (total in Table 3): 0.747 (Pearson correlation). None of the fields with more than 10,000 labeled payloads scored a smaller $f_{0.5}$ value than 0.78, while on the other side of the scale (fewer than 5,000 labeled payloads) no comparable tendency could be detected. The results for this use case ($f_{0.5}$-values) show the best scores compared to the other use cases. With the exceptions of "Biological Sciences" and "Chemical Sciences," all field-specific values for precision are larger than the recall scores.

[46] 25% of the records were at most 57 words long, with 10 words being the minimum.

[47] The (hyper)parameters for the best-$f_{0.5}$-MLPClassifier can be found in Weber, Fromm, and de Sousa (2019) (evaluation.csv) filtering for the pHash value 57508ee7a55735685fa52312df873dadcba64ac79f8bde3ac995bef84eca71e7.

Table 4. Best performance scores for each model + size combination

Model	Size	$f_{0.5}$ (macro)	$f_{0.5}$ (micro)	f_1 (macro)	f_1 (micro)	f_2 (macro)	f_2 (micro)
DecisionTreeClassifier	s	0.278	0.501	0.302	0.520	0.354	0.542
DecisionTreeClassifier	m	0.248	0.461	0.264	0.483	0.306	0.507
DecisionTreeClassifier	l	0.255	0.477	0.273	0.493	0.317	0.509
ExtraTreesClassifier	s	0.483	0.700	0.341	0.547	0.277	0.449
ExtraTreesClassifier	m	0.453	0.686	0.316	0.520	0.255	0.419
ExtraTreesClassifier	l	0.422	0.673	0.289	0.495	0.232	0.392
RandomForestClassifier	s	0.567	0.738	0.424	0.624	0.352	0.540
RandomForestClassifier	m	0.534	0.728	0.392	0.601	0.323	0.511
RandomForestClassifier	l	0.508	0.719	0.365	0.581	0.299	0.488
LSTMClassifier	s	0.779	0.851	0.739	0.833	0.704	0.816
LSTMClassifier	m	0.756	0.822	0.684	0.796	0.630	0.771
LSTMClassifier	l	0.765	0.831	0.701	0.804	0.651	0.780
MLPClassifier	s	0.780	0.845	0.736	0.833	0.703	0.822
MLPClassifier	m	0.792	0.859	0.753	0.845	0.720	0.832
MLPClassifier	l	0.798	0.861	0.751	0.850	0.721	0.847

The reported recall and precision values allow us to estimate errors when the classifier is applied to a large input set (see Section 7.3).

6.2.2. *Assistant systems*

Table 6 shows the performance scores analogous to Table 5, but for the use case "assistant systems" (f_1-scores)[48]. A field-by-field comparison of the two tables reveals that some fields drop more than others. One of the common features of those "dropping" fields is their relatively low number of total payloads. The effect of the imbalance of the fields is thus smaller on precision than on recall; in accordance with this finding is a stronger correlation between the number of records and f_2-scores: 0.819 (Pearson correlation). Assistant systems based on the proposed models are possible, although their acceptance by users seems doubtful if they fail to suggest obvious labels. They might be acceptable in contexts in which the worst-scoring fields "Agricultural and Veterinary Sciences" and "Philosophy and Religious Studies" play no or minor roles.

6.2.3. *Value-adding services*

The performance scores of the best performing model for the use case "valued-adding services" are listed in Table 7[49]. Unsurprisingly, the f_1-scores of the best models lie between their

[48] The (hyper)parameters for the best-f_2-MLPClassifier can be found in Weber et al. (2019) (evaluation.csv) filtering for the pHash value fa5f1c09aa4dfc69680b0e35bb5255c31d4c6aa1def78dbc61b2054f89cb7c85.

[49] The (hyper)parameters for the best-f_1-MLPClassifier can be found in Weber et al. (2019) (evaluation.csv) filtering for the pHash value fa5f1c09aa4dfc69680b0e35bb5255c31d4c6aa1def78dbc61b2054f89cb7c85.

Table 5. Best model for scientometric research ($f_{0.5}$): MLPClassifier (l-sized)

Label	Value	Recall	Precision
Mathematical Sciences	0.79	0.72	0.80
Physical Sciences	0.96	0.93	0.97
Chemical Sciences	0.82	0.82	0.82
Earth and Environmental Sciences	0.80	0.78	0.80
Biological Sciences	0.89	0.90	0.88
Agricultural and Veterinary Sciences	0.70	0.40	0.86
Information and Computing Sciences	0.82	0.78	0.82
Engineering and Technology	0.78	0.72	0.79
Medical and Health Sciences	0.84	0.83	0.85
Built Environment and Design	0.76	0.58	0.83
Education	0.78	0.61	0.84
Economics	0.74	0.60	0.79
Commerce, Management, Tourism and Services	0.72	0.51	0.80
Studies in Human Society	0.79	0.63	0.84
Psychology and Cognitive Sciences	0.85	0.77	0.87
Law and Legal Studies	0.80	0.53	0.92
Studies in Creative Arts and Writing	0.80	0.52	0.93
Language, Communication and Culture	0.79	0.62	0.84
History and Archaeology	0.80	0.61	0.88
Philosophy and Religious Studies	0.73	0.44	0.88

neighboring extremes,[50] but slightly closer to the f_2-scores than to the $f_{0.5}$-scores. This is due to the fact that the f_1-score is the harmonic mean between recall and precision, which tends to stress the lower values. Analogous to the previous use case, value-adding services based on the proposed model should be tested by interaction studies to determine whether users will accept their performance.

7. DISCUSSION

7.1. Field-Related Differences

There are fields of research that are in general easier for the models to detect, above all "Physical Sciences," which is the field of study with the highest number of records. Besides

[50] As does the correlation of number of records with f_1-scores: 0.766.

Table 6. Best model for value-adding services (f_2): MLPClassifier (l-sized)

Label	Value	Recall	Precision
Mathematical Sciences	0.74	0.72	0.80
Physical Sciences	0.94	0.93	0.96
Chemical Sciences	0.82	0.82	0.82
Earth and Environmental Sciences	0.76	0.75	0.82
Biological Sciences	0.89	0.89	0.89
Agricultural and Veterinary Sciences	0.53	0.50	0.72
Information and Computing Sciences	0.80	0.79	0.82
Engineering and Technology	0.74	0.72	0.80
Medical and Health Sciences	0.82	0.81	0.86
Built Environment and Design	0.65	0.64	0.69
Education	0.69	0.67	0.79
Economics	0.68	0.68	0.69
Commerce, Management, Tourism and Services	0.60	0.58	0.72
Studies in Human Society	0.71	0.70	0.77
Psychology and Cognitive Sciences	0.82	0.81	0.82
Law and Legal Studies	0.65	0.63	0.79
Studies in Creative Arts and Writing	0.62	0.60	0.77
Language, Communication and Culture	0.69	0.66	0.79
History and Archaeology	0.69	0.67	0.79
Philosophy and Religious Studies	0.59	0.56	0.73

this apparent correlation of number of records with the f_β-scores, other correlations of the f_β-scores with variables of the data set lead to the following insights:

- **Number of words per payload**: All f_β-scores correlate negatively with the median number of words per payload (Pearson correlation): –0.405 ($f_{0.5}$), –0.379 (f_1), –0.32 (f_2). These numbers indicate that the use of a concise vocabulary in the metadata improves the chance to be classified correctly. In the BoW approach, fields of research with a smaller vocabulary have a better relative representation in the set of terms finally selected. With regard to the LSTM/embedding approach, a possible explanation for the correlation is that shorter texts are easier for the model to "digest," meaning that further textual content does not improve the performance if it does not contain an equivalent semantic surplus.
- **Number of labels per payload:** The Pearson correlation between the mean number of labels and the f_β-scores is rather weak: 0.017 ($f_{0.5}$), 0.155 (f_1), 0.242 (f_2). The number of

Table 7. Best model for assistant systems (f_1): MLPClassifier (m-sized)

Label	Value	Recall	Precision
Mathematical Sciences	0.75	0.71	0.80
Physical Sciences	0.95	0.93	0.96
Chemical Sciences	0.81	0.80	0.83
Earth and Environmental Sciences	0.78	0.77	0.80
Biological Sciences	0.89	0.89	0.89
Agricultural and Veterinary Sciences	0.56	0.44	0.78
Information and Computing Sciences	0.80	0.76	0.85
Engineering and Technology	0.76	0.74	0.78
Medical and Health Sciences	0.83	0.79	0.86
Built Environment and Design	0.66	0.56	0.80
Education	0.74	0.68	0.81
Economics	0.68	0.60	0.78
Commerce, Management, Tourism and Services	0.66	0.59	0.74
Studies in Human Society	0.74	0.71	0.77
Psychology and Cognitive Sciences	0.83	0.78	0.89
Law and Legal Studies	0.71	0.66	0.77
Studies in Creative Arts and Writing	0.72	0.62	0.86
Language, Communication and Culture	0.75	0.70	0.81
History and Archaeology	0.78	0.74	0.83
Philosophy and Religious Studies	0.67	0.54	0.87

labels is therefore no explanation for the performance in general, at least not from the aggregated perspective[51].

7.2. Discussion of Misclassifications

This section discusses explanations of the reported misclassifications and presents strategies to mitigate the identified problems or improve the performance by trying out other approaches than presented in this paper.

The following sections focus on different aspects of the presented approach:

- data processing; and
- vectorizing approaches.

[51] The same applies to the ratio of 1-labeled payloads to total payloads average number of labels (Pearson correlation): 0.028 ($f_{0.5}$), −0.109 (f_1), −0.187 (f_2).

While we discuss some possible limitations of our approach in this context, a third approach to reduce misclassification is left out: model and/or hyper-parameter selection. As already stated, the solution space (models + hyper-parameter combinations) is too vast to be exhaustively searched with reasonable efficiency. However, we publish sufficient information and resources along with this paper for others to use them to supersede our findings or use them to benchmark other approaches (see Section 7.4).

7.2.1. Data issues

There are two promising explanations for misclassifications based on a critical review of the training/evaluation set:

- Some payloads may be labeled wrongly, which means that in such a case the model performs better than the person who originally labeled the research data item. There are structural explanations available for this assumption that go beyond simple classification errors: Some repositories might only allow or encourage one field of research per data set or data sets are not curated over time (e.g., by adding fields after submission). Unfortunately, there is no approach known to us to automate a procedure to identify and correct such types of misclassifications other than manual checks:
  - Ordering of all false positives by the probability the model assigns to the misclassified labels in decreasing order: The higher the probability, the more likely the model's classification is to be correct.
  - Manual relabeling if the machine's classification seems warranted.
- Some payloads could be of insufficient quality for both model *and* human expert to unambiguously classify the research data item. This means that the model would be "justified" in a false negative, because the payload in question would be too short and/or too unspecific. As with the previous idea, there is only a manual approach known to us to detect and handle such cases:
  - Ordering of all false negatives by the probability the model assigns to the misclassification and number of words (both in increasing order): The lower the probability and shorter the payload, the more likely the model's classification is to be warranted.
  - Manual assessment of the payload (does it contain enough information for a human expert to make a sound classification decision?) If the payload is not found to be sufficient for a classification, it can be ignored by future training runs.

Both these approaches focus on the quality of the training/evaluation data, whereas another idea is to enlarge these data by using additional sources, such as the Bielefeld Academic Search Engine (BASE)[52], Crossref[53], or the new records included in the DataCite index since our data retrieval.

All three approaches lead to a newer version of the training and evaluation data set, which necessitates a retraining of the models to assess the improvements by the baseline presented in this paper.

[52] https://base-search.net/
[53] https://crossref.org/

7.2.2. Vectorization issues

- **BoW**: The large number of input features for the best performing model (100,000) slows down prediction and increases the size of the model. Aside from these rather technical issues, 3,987,639 features are not exploited to increase the performance of the models. With dimension-reduction mechanisms such as Principal Component Analysis (PCA), this unused information might be exploited to further reduce the misclassifications, while the dimension of the input layer of the network is reduced at the same time (with beneficial consequences for the prediction latency and model size). Another approach is to test what impact the selection of another list of stop words would have.
- **Embeddings**: Using another embedding data set to vectorize the payloads might allow us to reduce the number of misclassifications, if these embeddings were trained on a context closer to the use cases of this paper than the context of the Google News data set. Another approach is to train the embeddings from scratch or update an existing embedding model during the classification process.

7.3. Estimates of Classification Errors

The best performing MLPClassifier and LSTMClassifier models are available for reuse in a python package[54]. If the models are used to explore big, unclassified sets of metadata, the reported precision and recall values for each field of study can be used to estimate the errors of the used classifier.

The precision for each field of study can be used to estimate the number of records that the model falsely classified for a given label:

$$\text{Wrong predictions(label)} = \sharp \text{ predicted values(label)} \cdot (1 - \text{precision(label)}).$$

This value can be used to determine the lower bound of the expected error. If the best performing model (MLPClassifier trained on l-sized data) results in 200 records classified as "Physical Science," the (rounded) expected number of wrongly classified records for this label would be six and the (rounded) lower error bound would be 194.

The recall for each field of study can be used to estimate the number of records the model missed to classify as a given label:

$$\text{Missed predictions(label)} = \sharp \text{ predicted values(label)} \cdot (1 - \text{recall(label)}).$$

This value can be used to determine the upper bound of the expected error. If the best performing model (MLPClassifier trained on l-sized data) results in 200 records classified as "Physical Science," the (rounded) expected number of missed records would be 14 and the (rounded) upper error bound would be 214.

7.4. Notes on Reproducibility

The presented procedure to retrieve, clean, and vectorize the data, and evaluate different machine learning models on the results is based on several assumptions, which were motivated in this paper. In general, the solution space is so vast that it is currently unreasonable to check every hyper-parameter combination or promising learning algorithm considering the resources necessary.

[54] https://github.com/tgweber/fosc/

Besides the explanation for the choices we gave in this paper, we followed an approach that allows us to retrace each step taken, so different configurations and hyper-parameters can be tested. Nevertheless, to guarantee comparability, it is crucial to keep those parts of the data processing and the learning pipeline fixed that do *not* diverge from our approach. To ease such a procedure, the data, source code, and all configurations, are made publicly available:

- Raw retrieved data: Weber (2019c);
- Cleaned and vectorized training and evaluation data:
  - small: Weber (2019d);
  - medium: Weber (2019b);
  - large: Weber (2019a);
- Source code with submodules for the steps presented in this paper, Weber and Fromm (2019), with the following components:
  - code/retrieve, corresponding to Section 3.1;
  - code/clean, corresponding to Sections 3.2–3.5;
  - code/vectorize, corresponding to Section 4.1;
  - code/evaluate, corresponding to Section 5;
  - config, includes all configurations and the stop word list; and
- Statistical data for this paper and evaluation data of all training evaluation runs: Weber et al. (2019).

7.5. Threats to Validity

The decision for a classification scheme is necessarily a political statement. How borders between fields are drawn, which research activities are aggregated under a label, and which fields are considered to be "neighbors" should be open to reflection and debate. The selection of the common classification scheme for this paper was steered by technical reasoning to minimize the effort while maximizing the classifiers' performance (see Section 3). The methodological approach, and therefore the code, allows the use of another classification scheme if mappings are provided from the found schemes to this alternative target scheme.

Another potentially arbitrary feature of the presented approach lies both in the subject classifications found in the raw data and in the process to streamline them during the cleaning step. DataCite aggregates over many sources, and therefore over many curators and scientists who make the first type of decision; Robinson-Garcia et al. (2017) report that 762 worldwide organizations were included as data centers in the DataCite index in April 2016; our findings indicate that this number has grown essentially since then (cp. Table 1). This bandwidth hopefully leads to a situation where prejudice and error are averaged out. Even if some cultural and socialized patterns in classification remain and some classification schemes are more prominent than others (ANZSRC, DDC), this approach is to our knowledge the best available. By filtering out metadata that were clearly classified by other automatic means (e.g., the linsearch classification scheme), we hope to minimize an amplifier effect (models trained on data classified by other models). We managed the second type of arbitrariness (mapping the found labels to a common scheme) by making each of the 613,585 mapping decisions transparent and reproducible, so that possibly existing biases and mistakes are correctable.

8. CONCLUSION AND OUTLOOK

This paper reports how training and evaluation metadata describing research data are retrieved, cleaned, labeled, and vectorized in order to test the best machine learning model to classify the metadata by the fields of study of the research data.

As this is a multilabel problem, both training and evaluation procedure must be aligned with technical best practices, such as stratified sampling or using macro averaged scores.

MLP models perform well enough for use in the context of scientometric research, while the use of the evaluated models in assistant systems and value-adding services of research data providers should only be considered after user interaction studies. Ideas for improvement of performance are mostly targeted at the training and evaluation data.

Ideas for scientometric application of the models on data sources such as DataCite and BASE include, but are not limited to, answering the following questions:

- Is research becoming more or less interdisciplinary? This question can be answered by using the automated classifiers on time-indexed data sets (such as DataCite, which includes the year of publication as a mandatory field). The classified data sets will display a trend: whether the number of labels increases, decreases, or stays stable.
- How does each field contribute to the growth of research data? This question can be answered by analyzing data sources such as DataCite or BASE after their contents have been classified by the presented models.

This paper provides the means to quantify the expected error in such investigations, based on the reported precision and recall scores (see Section 7.3). The models are published and wrapped in a Python module, which can be used without background in the development of machine learning modules[55].

Our understanding is that it is still an open question how models honoring the hierarchical nature of most classification schemes could be trained and implemented at scale for the use cases at hand.

ACKNOWLEDGMENTS

We, as the authors of this work, take full responsibility for its content; nevertheless, we want to thank the LRZ Compute Cloud team for providing the resources to run our experiments, Martin Fenner from the DataCite team for comments on an early draft of this paper, the staff at the library of the Ludwig-Maximilians-Universität München (especially Martin Spenger), for support in questions related to library science, and the Munich Network Management Team for comments and suggestions on the ideas of this paper. We thank the team at CERN responsible for Zenodo for their collaboration in testing the classifiers. The reviewers of an earlier version of this paper offered good advice on how to adapt the paper to the target audience and we thank them for thorough feedback.

AUTHOR CONTRIBUTIONS

Tobias Weber: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Visualization, Writing—original draft, Writing—review & editing. Dieter Kranzlmüller: Funding acquisition, Resources, Supervision. Michael Fromm: Methodology, Validation, Writing—review & editing. Nelson Tavares de Sousa: Conceptualization, Writing—review & editing.

COMPETING INTERESTS

The authors have no competing interests.

[55] https://github.com/tgweber/fosc#quick-start

FUNDING INFORMATION

This work was supported by the DFG (German Research Foundation) with the GeRDI project (Grants No. BO818/16-1 and HA2038/6-1). This work has also been funded by the DFG within the project Relational Machine Learning for Argument Validation (ReMLAV), Grant Number SE 1039/10-1, as part of the Priority Program "Robust Argumentation Machines (RATIO)" (SPP-1999).

DATA AVAILABILITY

All data are made publicly available. See Section 7.4 for more details.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., … Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org. Retrieved from https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/45166.pdf

Bäcker, A., Pietsch, C., Summann, F., & Wolf, S. (2017). BASE (Bielefeld Academic Search Engine). Eine Suchmaschinenlösung zur Indexierung wissenschaftlicher Metadaten. *Datenbank-Spektrum*, *17*(1), 5–13.

Bähr, T., Hannover, T., & Denecke, K. (2008). *LINSearch—Linguistisches Indexieren und Suchen Chancen und Risiken im Grenzbereich zwischen intel lektuel ler Erschließung und automatisch gesteuerter Klassifikation*. TIB Hannover/Forschungszentrum L3S Hannover. Retrieved from https://www.researchgate.net/publication/259938650_LINSearch_-Linguistisches_Indexieren_und_Suchen_Chancen_und_Risiken_im_Grenzbereich_zwischen_intellektueller_Erschliessung_und_automatisch_gesteuerter_Klassifikation

Bell, G., Hey, T., & Szalay, A. (2009). Beyond the data deluge. *Science*, *323*(5919), 1297–1298. https://doi.org/10.1126/science.1170411

Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python: Analyzing text with the Natural Language Toolkit*. Sebastopol, CA: O'Reilly Media, Inc.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. (1984). In L. Breiman (Ed.), *Classification and regression trees*. The Wadsworth statistics, probability series. Belmont, CA: Wadsworth International Group.

DataCite Metadata Working Group. (2019). *DataCite Metadata Schema for the Publication and Citation of Research Data, version 4.3*. DataCite e.V. https://doi.org/10.14454/f2wp-s162

Duan, K.-B., & Keerthi, S. S. (2005). Which is the best multiclass SVM method? An empirical study. In N. C. Oza, R. Polikar, J. Kittler, & F. Roli (Eds.), *Multiple Classifier Systems* (pp. 278–285). Berlin, Heidelberg: Springer.

Fisher, R. A. (1973). *statistical methods for research workers*. New York, NY: Hafner.

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, *63*(1), 3–42. https://doi.org/10.1007/s10994-006-6226-1

Golub, K., Hagelbäck, J., & Ardö, A. (2018). Automatic classification using DDC on the Swedish Union catalogue. In *CEUR Workshop Proceedings* (Vol. 2200, pp. 4–16). Porto, Portugal: CEUR.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computating*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

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

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. Published as a conference paper at the *3rd International Conference for Learning Representations*, San Diego, 2015. Retrieved from http://arxiv.org/abs/1412.6980

Kraker, P., Lex, E., Gorraiz, J., Gumpenberger, C., & Peters, I. (2015). Research data explored II: The anatomy and reception of figshare. In *Proceedings of the 20th International Conference on Science and Technology Indicators (STI 2015)*.

Lösch, M., Waltinger, U., Horstmann, W., & Mehler, A. (2011). Building a DDC-annotated corpus from OAI metadata. *Journal of Digital Information*, *12*(2).

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems 26* (pp. 3111–3119). Curran Associates, Inc. Retrieved from http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf

Nothman, J., Qin, H., & Yurchak, R. (2018). Stop word lists in free open-source software packages. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)* (pp. 7–12). Retrieved from https://aclweb.org/anthology/W18-2502

Nowak, K., Ioannidis, A., Bigarella, C., & Nielsen, L. H. (2018). DOI versioning done right. https://doi.org/10.5281/zenodo.1256592

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830. Retrieved from http://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf

Peters, I., Kraker, P., Lex, E., Gumpenberger, C., & Gorraiz, J. (2016). Research data explored: An extended analysis of citations and altmetrics. *Scientometrics*, *107*(2), 723–744. https://doi.org/10.1007/s11192-016-1887-4

Peters, I., Kraker, P., Lex, E., Gumpenberger, C., & Gorraiz, J. I. (2017). Zenodo in the spotlight of traditional and new metrics. *Frontiers in Research Metrics and Analytics*, *2*, 13. https://doi.org/10.3389/frma.2017.00013

Robinson-Garcia, N., Mongeon, P., Jeng, W., & Costas, R. (2017). Datacite as a novel bibliometric source: Coverage, strengths and

limitations. *Journal of Informetrics*, *11*(3), 841–854. https://doi.org/10.1016/j.joi.2017.07.003

Rumelhart, D. E., Hinton, G. E., Williams, R. J., et al. (1988). Learning representations by back-propagating errors. *Cognitive Modeling*, *5*(3), 1.

Sechidis, K., Tsoumakas, G., & Vlahavas, I. (2011). On the stratification of multi-label data. In D. Gunopulos, T. Hofmann, D. Malerba, & M. Vazirgiannis (Eds.), *Machine Learning and Knowledge Discovery in Databases* (pp. 145–158). Berlin, Heidelberg: Springer.

Sorower, M. S. (2010). *A Literature Survey on Algorithms for Multi-label Learning*. Oregon State University.

Tsoumakas, G., & Katakis, I. (2009). Multi-label classification: An overview. *International Journal of Data Warehousing and Mining*, *3*, 1–13. https://doi.org/10.4018/jdwm.2007070101

Waltinger, U., Mehler, A., Lösch, M., & Horstmann, W. (2011). Hierarchical classification of OAI metadata using the DDC taxonomy. In R. Bernardi, S. Chambers, B. Gottfried, F. Segond, & I. Zaihrayeu (Eds.), *Advanced Language Technologies for Digital Libraries* (pp. 29–40). Berlin, Heidelberg: Springer.

Wang, J. (2009). An extensive study on automated dewey decimal classification. *Journal of the American Society for Information Science and Technology*, *60*(11), 2269–2286. https://doi.org/10.1002/asi.21147

Weber, T. (2019a). *l-sized Training and Evaluation Data for Publication "Using Supervised Learning to Classify Metadata of Research Data by Field of Study."* https://doi.org/10.5281/zenodo.3490460

Weber, T. (2019b). *m-sized Training and Evaluation Data for Publication "Using Supervised Learning to Classify Metadata of Research Data by Field of Study."* https://doi.org/10.5281/zenodo.3490458

Weber, T. (2019c). *Raw Data for Publication "Using Supervised Learning to Classify Metadata of Research Data by Field of Study."* https://doi.org/10.5281/zenodo.3490329

Weber, T. (2019d). *s-sized Training and Evaluation Data for Publication "Using Supervised Learning to Classify Metadata of Research Data by Field of Study."* https://doi.org/10.5281/zenodo.3490396

Weber, T., & Fromm, M. (2019). *Source Code and Configurations for Publication "Using Supervised Learning to Classify Metadata of Research Data by Field of Study."* https://doi.org/10.5281/zenodo.3757464

Weber, T., Fromm, M., & de Sousa, N. T. (2019). *Statistics and Evaluation Data for Publication "Using Supervised Learning to Classify Metadata of Research Data by Field of Study."* https://doi.org/10.5281/zenodo.3757468

Weber, T., & Kranzlmüller, D. (2018). How FAIR can you get? Image retrieval as a use case to calculate FAIR metrics. In *2018 IEEE 14th International Conference on E-science* (pp. 114–124). https://doi.org/10.1109/eScience.2018.00027