Explainable Fine-Grained Document Classification for Mathematical Texts

Philipp Scharpf[1], Moritz Schubotz[2], Corinna Breitinger[3], André Greiner-Petter[4], Klaus Lippert[5], Konrad Foerstner[6], Olaf Teschke[7], and Bela Gipp[8]

[1,3]University of Göttingen, Germany ({last}@gipplab.org)
[2,7]FIZ-Karlsruhe, Germany ({first.last}@fiz-karlsruhe.de)
[4]University of Göttingen, Germany (greinerpetter@gipplab.org)
[5,6]ZB MED - Information Centre Life Sciences, Germany ({last}@zbmed.de)
[8]University of Göttingen, Germany ({last}@cs.uni-goettingen.de)

December 16, 2024

Abstract

To structure digital libraries and to allow readers to search for articles on specific topics, reliable and fine-grained document subject classification is essential. Currently, research article classification is largely performed by human domain experts. Semi-supervised Machine Learning algorithms can support experts by exploiting the labeled data to predict subject classes for unclassified new documents. However, previous research indicates that these algorithms are effective, i.e., produce meaningful results, only when the ratio of training examples per class is sufficiently high. Furthermore, in the domain of mathematical documents, the widely adopted Mathematical Subject Classification (MSC) scheme presents multiple challenges: The classification is 1) multi-label, 2) hierarchical, 3) fine-grained, and 4) sparsely populated at the lowest level. Specifically, the current MSC scheme contains 63 two-digit classifications, 529 three-digit classifications, and 6,006 five-digit classifications. In this paper, we extract and leverage the class-entity relations of mathematical texts for the first time to facilitate multi-label hierarchical and fine-grained category predictions. We analyze the relationships between specific subject classes and keyword entities using the zbMATH Open service that yields the largest dataset of classified mathematical publications with over 4 million documents. Moreover, we compare fine-grained MSC prediction on the zbMATH corpus of mathematical texts to MeSH prediction on the PubMed corpus of medical texts provided by ZB MED. Finally, we present approaches to explain

the classification suggestions, thus easing the work of human reviewers in either accepting or rejecting suggestions. The results show we can predict MSCs from keywords with a precision-recall curve close to the human baseline. A demo of our fine-grained MSC prediction explainer (using text and keywords) and an interactive notebook to reproduce our experiments are available at `https://automscexplainer.wmcloud.org`. We also provide a public API for the fine-grained MSC prediction at `https://automscpredictor.wmcloud.org`.

1 Introduction

Since the earliest known classification scheme by the Greek Callimachus, a librarian of the Library of Alexandria, there have been countless efforts to classify subject categories in document collections [16]. Libraries must be sorted so readers can search for literature in specific areas or topics of interest. With the rise of digital libraries, machine-readable documents, and Machine Learning methods, human expert classifiers are being supported by computers.

In the case of mathematical literature, the world's most comprehensive record of bibliographic data, reviews, and abstracts is the zbMATH Open service.[1] zbMATH Open contains over 4M mathematical documents, individually assigned to 6k hierarchically organized classes, known as the Mathematical Subject Classification (MSC). In contrast to commercial applications like Amazon's book categories,[2] digital libraries containing academic texts have fewer human resources and fewer development resources available. Additionally, an impractically large number of domain experts would be required to classify the highly technical and specialized literature in extensive fields that contain many subdisciplines, such as mathematics. Thus, classifying academic literature as automated, fine-grained, and as accurately as possible with limited resources is desirable.

Unfortunately, automated Machine Learning document classifiers require sufficient training data with a high ratio of examples per class. If the classification scheme is fine-grained and multi-label, the classification is often challenging[3] and not feasible using a Machine Learning classifier due to class sparsity (refer to the 'curse of dimensionality' problem [26]). Connecting category classes with keyword concept entities mitigates this problem by employing a class-entity knowledge graph to augment classification training data [1].

Using the class-entity knowledge graph, it is possible to build an Artificial Intelligence (AI) model that can predict fine-grained hierarchical class labels. The prediction can then be used for automatic unsupervised labeling or supervised label suggestions to human expert classifiers. This considerably accelerates the labeling process for unclassified documents that are constantly being added to digital libraries. Moreover, we can use the class-entity (or category-concept) knowledge graph to provide class-prediction explanations for humans.

[1] `https://zbmath.org`

[2] `https://blog.reedsy.com/guide/kdp/amazon-book-categories/`

[3] `https://nfdi4ds.github.io/nslp2024/docs/forc_shared_task.html`

Document classification is often a black box and needs more transparency and explainability for humans to understand AI decisions. The topic of explainable AI (XAI) has recently gained increasing interest in Machine Learning applications, e.g., in medicine or jurisdiction [11], where AI decisions determine the fate of individuals. Methods of XAI aim to reverse-engineer a class-entity (or label-feature) correspondence.

In this paper, we show that by using concept keywords, we can perform explainable, multi-label, and fine-grained hierarchical document classifications for 6k class labels in the zbMATH Open mathematical library with precision and recall close to the human baseline. By linking concept keywords to the Wikidata knowledge graph, we directly establish relations between feature Wikidata entity QIDs[4] and MSC subject class labels. In analogy to named entities for natural language, keyword entities are mathematical notions, such as 'path integral' or 'variational principle.' Mapping keywords to hierarchical classifications, when compared to LLM-based approaches, offers several distinct benefits, particularly in terms of explainability. In this way, explainability is augmented by references to unique identifiers. This degree of transparency is a decisive advantage over LLMs, which often function as "black boxes," offering limited insight into the reasoning behind their classifications. While LLMs provide advanced capabilities in handling unstructured, nuanced text, the approach of mapping keywords to hierarchical classifications excels in areas of explainability, consistency, and the potential for creating rich, structured knowledge representations like ontologies.

To ensure reproducibility, our research results (tables and explanations), data (raw and indexed), and code are publicly available.[5] In addition, we provide an interactive notebook to easily run the code[6] and a demonstration user interface of the fine-grained MSC prediction recommender and recommendation explainer using text and keywords.[7] Finally, a public API for the fine-grained explainable MSC prediction from mathematical texts can be accessed.[8]

2 Related Work

To highlight the existing research gap, we first review the state of the art in fine-grained document classification and mathematical document classification. This paper then addresses the challenges of limited training data and the absence of explainability in Machine Learning document classifiers for mathematical texts. It introduces an advanced, explainable, multi-class, hierarchical, fine-grained mathematical document classification system that utilizes class-entity relationships. Our research builds on and extends our previous publication [20].

[4] Unique Wikidata concept item identifiers (URIs) with corresponding URLs.
[5] `https://github.com/AnonymousCSResearcher/FineGrainedMSCPred`
[6] `https://purl.org/fine-class`
[7] `https://automscexplainer.wmcloud.org`
[8] `https://automscpredictor.wmcloud.org`

2.1 Fine-Grained Document Classification

The rapid increase in digital documents has called for the development and employment of methods for Automatic Document Classification (ADC) [21]. While human domain expert labeling is costly, tedious, and time-consuming, effective ADC approaches categorize documents at scale, instantaneously and at a lower cost. Furthermore, the algorithms are portable and can be applied to various other applications, such as spam filtering, sentiment analysis, product categorization, speech categorization, author and text genre identification, essay grading, word sense disambiguation, and hierarchical categorization of web pages [12].

Document classification can be divided into supervised, unsupervised, and semi-supervised approaches. Semi-supervised approaches still require labeling; however, they require less manual labeling of data. Further, classification approaches can be divided into few-shot learning (FSL), one-shot learning (OSL), and zero-shot learning (ZSL) [13], where the number of 'shots' signifies the number of examples for which the model needs to predict new labels. Due to the large number of classes in fine-grained classification schemes, there are often only zero, one, or a few examples available for each class. This motivates using ZSL, OSL, and FSL in this scenario. If the example per class ratio is very low and classical Machine Learning does not work, we can employ class-entity relation statistics to perform ZSL. We demonstrate the effectiveness of this adaptive approach in Section 4.3.

There are single-label and multi-label, as well as coarse-grained and fine-grained (hierarchical) classification problems. Automated fine-grained multi-label classification is much more challenging than coarse-grained single-label classification [27]. Nonetheless, only a few recent approaches have addressed the current research problems. Eykens et al. present and evaluate methods for fine-grained multi-label classification of social science journal articles using textual data. They achieve F1-scores of up to 0.55 when classifying 113,909 records into 31 sub-disciplines from three main disciplines [4].

In contrast to learning-based Extreme Multi-Label Text Classification (XMLC) methods, which typically employ Deep Learning transformer models [25], domain ontology-based methods enhance 1) classification interpretability, 2) efficiency in sparse scenarios, 3) reduction of noise in the label space [30]. On the other hand, state-of-the-art XMLC methods [31] mostly lack prediction explainability and exhibit complexity and computational demand due to their reliance on large transformer architectures.

In the life sciences and medicine, fine-grained Medical Subject Headings (MeSH)[9] are used to index journal articles and books, most prominently in the online Medical Literature Analysis and Retrieval System (MEDLINE)[10] and the National Library of Medicine PubMed[11] database. The MeSH scheme includes

[9] `https://www.nlm.nih.gov/mesh/meshhome.html`
[10] `https://www.nlm.nih.gov/bsd/medline.html`
[11] `https://pubmed.ncbi.nlm.nih.gov`

more than 28,000 entries[12] used to index and catalog medical literature. Each entry can have numerous subheadings, allowing for very fine-grained indexing. The hierarchy includes multiple layers for detailed categorization of medical topics, including diseases, drugs, and procedures. Methods for automated assignment of MeSHs [15] exploit abstract similarity and citations [8], as well as knowledge graphs and hierarchical structure [29]. Approaches for medical document classification based on the domain ontology MeSH find that employing concept keywords enhances classification performance [3].

In this paper, we test the hypothesis of keyword enhancement for a fine-grained mathematical document classification scheme, which we introduce in the following.

2.2 Mathematical Document Classification

zbMATH Open[1] provides a catalog with abstracts and reviews for mathematical documents, sorted and labeled using the fine-grained hierarchical 'Mathematics Subject Classification (MSC)' scheme.[13] The MSC has a long history[14] with major version publications every ten years (e.g., 2000, 2010, 2020). The MSC scheme is also developed and used [2] by the American Mathematical Society (AMS) Mathematical Reviews (MR) electronic bibliographic database MathSciNet.[15]

In 2008, Watt examined relative symbols and expression frequencies to classify a mathematical document according to the MSC scheme [28]. He found that the particular use of symbols and expressions, i.e., their frequency ranking, varies from area to area between different top-level subjects of the MSC 2000. However, the 'pattern of relative frequencies for the most popular symbols' was noted to remain the same. It was claimed (but not verified) that the symbol frequency 'fingerprints' for the different MSC areas could be used to classify given mathematical documents. Kusmierczyk et al. compared hierarchical mathematical document clustering against the hierarchical MSC classification tree [9]. They postulated that the hierarchy was highly correlated with the document content. Using publications from the zbMATH database, they aimed to reconstruct the original MSC tree based on document metadata. For the comparison, they developed novel tree similarity measures. The best results were obtained for 3-level hierarchical clustering using bigram encodings.

In 2014, Schöneberg et al. discussed part-of-speech (POS) Tagging and its applications for mathematics [22]. They aimed to adapt NLP methods to the special requirements for mathematical document content analysis. They presented a mathematics-aware POS tagger for mathematical publications. The tagger was trained using keyphrase extraction and classification of documents from the zbMATH database. The results showed that while precision was sufficient (for 26 of the 63 top-level classes higher than 0.75 and only for 4 classes

[12] `https://libguides.umsl.edu/pubmed/mesh`
[13] `https://zbmath.org/static/msc2020.pdf`
[14] `http://www.mi.uni-koeln.de/c/mirror/www.ams.org/msc/msc-changes.html`
[15] `https://mathscinet.ams.org`

smaller than 0.5), recall was very low.

In 2017, Suzuki and Fujii presented a structure-based method for Mathematical Document Classification [24]. They included structures of mathematical expressions (ME) as classification features combined with the text. They hypothesized that ME would hold important information about mathematical concepts, being a central part of communication in Science, Technology, Engineering, and Mathematics (STEM) fields. Employing 3,339 Q&A threads from MathOverflow[16] and 37,735 papers from arXiv,[17] they achieved classification F-measures of 0.68 on the text and 0.71 on the combined text and math encodings.

In 2019, Ginev and Miller performed a supervised Scientific Statement Classification over arXiv.org [5]. Exploring 50 author-annotated categories, they grouped 10.5 million annotated paragraphs into 13 classes. Using a BiLSTM encoder-decoder model, they achieved a maximum F1-score of 0.91. Further, they introduced a lexeme serialization for mathematical formulae and discuss the limitations of both data and task design, highlighting the lacking capacity to provide a live human evaluation of the classification results.

In 2020, Scharpf et al. presented large-scale experiments for classification and clustering of arXiv documents, sections, and abstracts comparing encodings of natural and mathematical language [21]. They evaluated the performance and runtimes of selected algorithms, achieving classification accuracies of up to 82.8% and cluster purities of up to 69.4%. Further, they observed a relatively low correlation between text and math similarity, indicating a potential independence of text and formula document features. Moreover, they demonstrated that the computer outperformed a human expert in classification performance.

2.3 Research Gap and Delta

Previously, Schubotz et al. introduced 'AutoMSC' - a system for the automatic assignment of Mathematics Subject Classification (MSC) labels [23]. Evaluating the performance of automatic methods in comparison to a human baseline, they found that their best-performing method achieved an F1-score of 77.2%. The authors claim that their models could reduce manual classification effort by 86.2% without losing classification accuracy.

However, using Machine Learning, Schubotz et al. could only classify the main MSC (primary subject) label (63 available labels). The novel approach we present in this paper addresses this shortcoming through its ability to predict 6k fine-grained low-level MSCs by employing class-entity relations. Furthermore, our classifications are explainable. This means that for each predicted class (label), the distribution of keywords (features) is extracted and visualized.

[16] `https://mathoverflow.net`
[17] `https://arxiv.org`

3 Methods

In this paper, we address the lack of explainable and interpretable fine-grained hierarchical mathematical document subject category classification. Specifically, we design a solution for the following challenges of the Mathematical Subject Classification (MSC),[18] in which the classification is 1) multi-label, 2) hierarchical, 3) fine-grained, and 4) sparsely populated at the lowest level. The new MSC contains 63 two-digit classifications, 529 three-digit classifications, and 6,006 five-digit classifications. As described in Section 2, Machine Learning approaches for the coarse-grained two-digit classification already exist, while the fine-grained three-digit and five-digit classifications still need to be done manually. Figure 1 demonstrates the hierarchical structure of the MSC, highlighting its progression from broad, coarse-grained categories to highly specific fine-grained ones. This organization introduces several challenges: the dependencies between hierarchical levels require consistency across predictions, the sparsity of data at fine-grained levels makes accurate modeling difficult, and the overlapping nature of categories necessitates handling multi-label assignments effectively. Additionally, the complexity of navigating over 6,000 categories underscores the importance of explainable methods to ensure the results are interpretable and usable for domain experts.

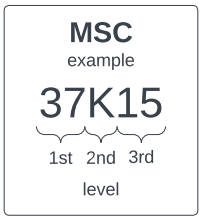

Figure 1: Example of the MSC document classification level hierarchy scheme.

3.1 Research Tasks

To address these challenges, we analyze the relationships between categories (labels) and entities (features) of a mathematical document. Our developed approaches are guided by advances in the medical domain for the automated assignments of medical document classification labels using a domain ontology, as described in Section 2.1. Further, our methods build on the recent results and datasets presented in Section 2.2. No other baselines for fine-grained Mathematical Subject Class (MSC) document classification exist to date. Furthermore, we are also not aware of any baselines for similar fine-grained hierarchical multi-label classification problems.

[18]The following exemplary descriptions focus on MSC prediction from mathematical texts. The MeSH prediction from medical texts is done analogously.

To predict fine-grained MSC class labels, we use the abstract (text), keywords, and references[19] as input. We define the following research tasks:

1. Develop methods for fine-grained subject classifications for sparse training data using document (abstract) text or concept keywords.
2. Compare the subject class predictions to Machine Learning and human baselines.
3. Evaluate entity linking of text entities to a knowledge base to persist class-entity relations that enable a fine-grained label prediction.

3.2 Research Method

The problem of data sparsity prevents the feasibility of multi-class hierarchical fine-grained classification of mathematical documents using Machine Learning. Two conceivable methods could be used to address this problem:

1. Predicting missing data using augmentations from an existing knowledge graph and subsequently executing Machine Learning classification algorithms on the resulting, more densely populated, dataset;
2. Directly predicting the sparsely populated fine-grained category classes using a knowledge graph created from available class-entity relations.

In our case (MSC prediction from text and keywords), no existing knowledge graph is available. Thus, our approach is to use the second method to create a knowledge graph using extracted class-entity (or category-concept)[20] relations stored in an index. Employing the created knowledge graph makes it possible to use class-entity or entity-class co-occurrence frequency statistics to predict the fine-grained and sparsely populated classes. We use mathematical documents with both category class labels and concept entity features available to retrieve the co-occurrence frequency relations.

This results in ranked lists, descending by co-occurrence frequency, where for each class, a number of correlated entities exists, and vice versa. For the class predictions, the assignment of category classes for each keyword entity up to a specified prediction cutoff (maximum number of classes per entity) provides respective prediction confidence scores.

Figure 2 illustrates index creation and reversion from class-entity to entity-class. Let us explain the index creation using a general template with example numbers. The class-entity co-occurrence frequency index could be:
`{'class 1': {'entity 1': 15, 'entity 2': 5, ...},`
`'class 2': {'entity 7': 8, ...}}`.
In this case, for `'class 1'`, `'entity 1'` co-occurs 15 times,
`'entity 2'` 5 times, etc.

[19] The class labels of the references.
[20] The terms class or category and entity or concept can be used interchangeably.

Subsequently, the index is reversed to
`{'entity 1': {'class 1': 15, 'class 3': 6, ...},`
`'entity 2': {'class 1': 5, ...}}`.
The reverse entity-class index enables a multi-label prediction of classes from entities.

In this example, the prediction confidence score for `'class 1'` given `'entity 1'` is three times higher than given `'entity 2'`. The exact confidence score is computed by normalizing the co-occurrence frequencies (dividing by their respective maxima).

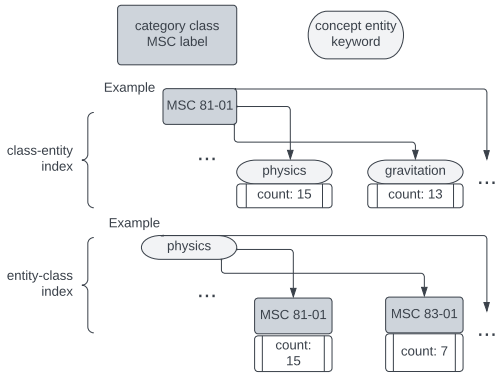

Figure 2: Illustration of the class-entity and reverse entity-class index that is used (as a knowledge graph) for multi-label hierarchical fine-grained classification with co-occurrence frequency confidence.

Given a novel document labeled with a list of keyword entities, the procedure is as follows:

1. In the first step, the reverse entity-class index is employed to compute a ranked list of all co-occurring classes for each entity.
2. In the second step, the resulting ranking of the collected classes yields the confidence of the prediction.
3. In a single-label classification, the class with the highest frequency is predicted.
4. In a multi-label classification, the first m classes from the ranked list with descending frequency numbers are predicted.

So, using the class-entity co-occurrence frequency statistics index, we can predict a specified number n of MSCs (category classes) given keyword entities for a given mathematical text document.

Keyword Entity Linking to Wikidata To persist the class-entity or entity-class correspondence relations in a knowledge graph [6], we employ entity linking ('Wikification') of zbMATH concept entity keywords to Wikidata[21] QIDs. Wikidata QIDs are unique identifiers (URIs) with corresponding concept item URLs (often also linked to a set of Wikipedia articles in different languages). For example, the concept 'speed' can be linked to a Wikidata item[22] with the QID Q3711325. This allows us to make the fine-grained class-entity mapping entirely unique and numeric since both MSCs and QIDs are not ambiguous. We employ two Wikidata retrieval sources to link keywords (n-grams) to QIDs. For each keyword, we predict Wikidata QIDs using both Pywikibot[23] and SPARQL.[24] The final unique class-entity or entity-class knowledge graph with URLs is then seeded to the respective Wikidata concept entity items as 'Mathematics Subject Classification ID' (P3285) property statements.

3.3 Implementation

We now outline the implementation of the data processing pipeline to apply and evaluate our methods. Data and algorithms (code) for our experiments can be found in the paper repository.[5] Due to the large size of the classification indexes (part of our contribution), we publish them separately (anonymously).[25]

The script `evaluate_classification.py` contains all required steps in the data processing pipeline. The steps are explained in detail in an interactive notebook.[6]

The procedure is as follows.[26]

1. Load train table: The full 'zbMATH Open Mathematics Subject Classification Dataset' is downloaded from its open source.[27]

2. Generate mapping indexes: The MSC-keyword-MSC or MSC-reference-MSC index is created from the train table. The indexes are dumped to disk to be reloaded to provide MSC predictions.

3. Index statistics: Average index key entry numbers and index distribution entropies are computed to illustrate that entities are much more sharply defined by classes than classes by entities.

4. Load test table: The full MR-MSCs baseline, the 'Mathematics Subject Classification interrater agreement dataset' is downloaded from its open source.[28]

[21] `https://www.wikidata.org`
[22] `https://www.wikidata.org/wiki/Q3711325`
[23] `https://www.mediawiki.org/wiki/Manual:Pywikibot/de`
[24] `https://query.wikidata.org`
[25] `https://zenodo.org/records/10251194`
[26] For the exact script and function names, see the README in the repository.
[27] The dataset is publicly available at `https://zenodo.org/record/6448360`.
[28] The dataset is publicly available at `https://zenodo.org/record/5884600`.

5. Predict MSCs: For each document in the input table, a ranked list of MSCs is predicted from its (abstract) text, keywords, and references using the indexes with a specified prediction cutoff at the nth MSC (see Section 3.2).

6. Evaluate MSC predictions: The MSC ranking quality of the predicted MSCs is evaluated in comparison to selected baselines (e.g., MR-MSCs) in terms of Discounted Cumulative Gain (DCG) scores.

7. Precision-recall curves: The prediction (from text, keywords, and references) quality is evaluated in terms of precision and recall in comparison to a human baseline.

8. MSC prediction explainer demo: A user interface is provided to test the MSC prediction from (abstract) text or keyword entities.

3.4 Classification Explainability

Both the class-entity and the reverse entity-class index can be used to provide explainability and interpretability of the prediction by displaying the respective distributions to a human reader or reviewer. This is a decisive advantage over classical Machine Learning models. Currently, only a few very simple and lean classifiers support recent explainers, such as LIME [17] or SHAP [10]. In our approach, using the entity-class index, it is possible to trace back the exact prediction entity source for each predicted class label. This can help both to explain semi-supervised suggestions to human annotators and to understand the bottlenecks of the model in the performance evaluation. Our demo user interface provides MSC prediction explanations as class distribution charts (see Figure 3). Finally, our fine-grained prediction can be employed in production using our public API provided.[8]

4 Evaluation

In this section, we present and discuss our evaluation results. Our data, code, and result tables are publicly available.[5] We present and evaluate our novel approach to provide fine-grained hierarchical multi-label prediction of MSC labels. Moreover, we exploit class-entity correlations to achieve classification explainability. Finally, we assess entity linking of MSCs to Wikidata URLs to persist the class-entity relations as a knowledge graph.

In our experiments, we vary numerous *evaluation parameters*, such as included subject classes and granularity, ranking size, distribution type, n-gram length, and text cleaning. We evaluate our experimental results using several different *evaluation metrics*, such as class prediction accuracy, entity prediction relevance, and ranking quality using nDCG, as well as precision, recall, and F1-score.

4.1 Dataset Statistics

In the following, we describe the two employed datasets, corpora of mathematical and medical texts, both with fine-grained classification labels, respectively. The train-test split is done according to the available human baseline dataset (interrater agreement) as the independent test set (yielding a ratio of approximately 10:1).

The 'zbMATH Open Mathematics Subject Classification Dataset'[27] contains 4,374,874 documents labeled both with MSCs and keywords. The columns contain the document number, DOI, MSC, keywords, title, abstract, and references. The generated class-entity index contains 6,679 MSC classes correlated with keywords, while the entity-class index contains 2,481,029 keywords (composite n-grams). On average, each MSC class is attributed to two keyword entities, whereas each keyword entity is attributed to six MSC classes. The average entropy of the individual class-entity distributions (6.02) is about six times larger than that of the individual entity-class distributions (1.18). Both average index entry length and distribution entropy indicate that, as expected, entities are much more sharply defined by classes than classes by entities.

The medical corpus is a randomized subset with a training set and test set of the same size as the mathematical corpus, taken from the complete PubMed dataset provided by the National Library of Medicine.[29] For ease of processing, the data was taken from the corpus behind the Search Portal [14] LIVIVO[30] provided by the German library ZB MED.[31]

4.2 Evaluation Metrics

To explore the potential of fine-grained multi-label classification using a class-entity knowledge graph, we compare MSC predictions based on zbMATH Open keywords to the predictions based on references and compare both against a human baseline from the AMS Mathematical Reviews (MR) journal annotators.[27] We evaluate the respective predictions using the normalized Discounted Cumulative Gain (nDCG) ranking performance measure, which can be calculated according to [7] as

$$\mathrm{nDCG} = \frac{DCG}{IDCG}$$

with the ideal IDCG (predictions exactly match the baseline) and the individual DCGs given by

$$\mathrm{DCG_p} = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)}$$

where rel_i is the relevance (here 0, 1, or 2) at position i, and p is the ranking scale cutoff (here position 10).

[29] `https://pubmed.ncbi.nlm.nih.gov/download`
[30] `https://www.livivo.de/app?LANGUAGE=en`
[31] `https://www.zbmed.de/en`

Both the quality of the fine-grained MSC predictions and the entity linking of keywords to Wikidata are evaluated using classical information retrieval (IR) metrics, such as precision, recall, and F1-score that are derived from the number of true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN). We use the IR metrics to calculate a precision-recall curve.

4.3 Research Results

In the following, we present the results of our experiments for 1) fine-grained MSC prediction from document abstract text, keywords, and references (nDCG, precision-recall curve), and 2) Entity linking of keywords to Wikidata items (F1 measure).

Fine-grained MSC prediction from abstract text, keywords, and references Table 1 shows the ranking performance of the MSC predictions for different MSC number cutoffs (first column). The results show that the fine-grained MSC predictions from keywords almost reach the nDCG scores of the human zbMATH Open baseline while outperforming the state-of-the-art prediction from reference MSCs [23]. In the medical corpus, MeSHes are unranked (no nDCG).

Table 1: Comparison of nDCG for MSC predictions from keywords and references for different numbers of assigned MSCs (prediction cutoffs). Mathematical Reviews (MR) annotations are used as human baseline (Baseline-MSCs).

Nr. MSCs	Baseline-MSCs	Keyword-MSCs	References-MSCs
10	0.61	0.53	0.40
5	0.61	0.49	0.32
3	0.58	0.44	0.27
1	0.35	0.26	0.16

Figure 4 shows the precision-recall curve of the MSC predictions from text, keywords, or references compared to the human baseline. The results indicate that the quality of the MSC predictions from keywords is close to the human baseline (inter-reviewer agreement) for a prediction cutoff of up to three MSCs with both precision and recall around 0.5 (F1 measure 0.5). The prediction from text achieves lower scores. This could be attributed to the fact that the abstract text contains a wider range and greater number of keywords, thus introducing entropy.

Figure 5 shows the precision-recall curve of the MSC predictions from text or keywords (references or a human baseline of independent annotators are unavailable). The results show that the quality of the MSC predictions is much better than that of the MeSH predictions. For both MSCs and MeSHes, the prediction from keywords outperforms the prediction from (abstract) text.

One potential explanation for the strikingly low recall of MeSH prediction from text (both in absolute terms and relative to MSC prediction from text) is the difference in annotator freedom between the two corpora. In the MSC

corpus, annotators have more flexibility in selecting keywords, allowing them to choose words that frequently appear as entity names in the abstract. Conversely, in the MeSH corpus, the selection of keywords is constrained by the MeSH ontology, resulting in more conceptual terms that may not closely match the concrete language used in the abstract text.

Overall, the concept keywords are a valuable predictor of the MSC category classes (`predicted_keywords`). The lower performance of fine-grained MSC prediction from reference MSCs
(`predicted_references`) differs from the results of Schubotz et al. for coarse-grained MSC classification [23]. In their experiments, the prediction from reference MSCs (F1-score 0.74) performs better than the one from text features (F1 = 0.70) and is closer to the human baseline (F1 = 0.81).

In conclusion, the superior predictive performance achieved through the utilization of keyword entities is a positive outcome. This result suggests that, particularly in fine-grained scenarios, entities prove to be more valuable than references. This finding underscores the significance of employing our keyword-MSC index prediction method.

Entity linking of keywords to Wikidata items Finally, we assess the prediction of Wikidata QIDs or concept item URLs from zbMATH Open keywords, also known as Entity Linking or 'Wikification' for Knowledge Graph Population. This step is necessary if no keywords are available for a given document. In this case, the Entity Linker extracts them before the fine-grained classification. Table 2 shows the evaluation of entity linking for 500 mathematical concept keywords from 100 randomly selected documents. The complete evaluation lists of results can be found in the repository. For the performance assessment of the entity linking methods, a manual annotation benchmark is created.

Table 2: Manual evaluation of automatically linking 500 random mathematical concept keywords from zbMATH Open to Wikidata QIDs. The manual benchmark can be reused to evaluate other approaches on this dataset.

Method	**Precision**	**Recall**	**Specificity**	**F1**
Pywikibot	1.0	0.875	1.0	0.933
SPARQL	0.987	0.570	0.667	0.723

The results in Table 2 show that both entity linkers are highly precise, with only one false positive for Pywikibot and even zero for the SPARQL retrieval. However, there are a number of false negatives for both, leading to a lower recall. This can be traced back to linked disambiguation pages (both for Wikidata items and Wikipedia articles). Moreover, some concepts have items and articles named by their synonyms or aliases. Overall, the Pywikibot linker performs better than the SPARQL retrieval in terms of precision, recall, and especially true negative rate (TNR) and F1-score.

5 Discussion

In this final section, we recall our findings, discuss and summarize our contribution, and outline future research directions.

5.1 Conclusion

Our research shows that using the relations between mathematical keyword entities and category classes in mathematical documents, we can achieve fine-grained mathematical subject class category (MSC) predictions. To demonstrate the effectiveness of this approach, we employed a dataset containing 4.4M documents from zbMATH Open, labeled using MSC-keyword correspondences.

Our experiments indicate that we can predict fine-grained MSC subjects in the zbMATH Open dataset using document keywords with a performance in terms of normalized Discounted Cumulative Gain and precision-recall curve close to the human baseline. The lower performance of the MSC prediction from (abstract) text indicates that this method should primarily be employed if keywords are unavailable.

Finally, we can link keywords to Wikidata items (Wikipedia articles) with a precision of 1.0, recall of 0.88, and F1-score 0.93 on a collection of 500 manually assessed samples. The linking allows the fine-grained class-entity mapping to consist only of unique and numeric identifiers (MSCs and QIDs).

In summary, we show that by exploiting the relationships between category classes and concept entities, we can address fine-grained classification with data sparsity and provide high classification explainability. Our work is the first to apply entity linking to Wikidata in mathematical texts by using MSC classification labels.

Impact The research we present in this paper has already impacted the mathematical and research community. Our public API[8] is used in production by the mathematical library zbMath Open, which is being accessed around 40 million times per year.[32]

5.2 Future Work

In the future, we will seed additional entity linking relations between MSC category classes and concept entity Wikidata QIDs to the Wikidata knowledge graph and persist entity-category linkings to evaluate the quality of knowledge graph labeling. For example, the Wikidata item for the keyword 'least squares method' (Q74304)[33] is linked to the MSC '62J02' via the property 'Mathematics Subject Classification ID' (P3285). We seeded the concept entities[34] into Wiki-

[32] https://www.fiz-karlsruhe.de/sites/default/files/FIZ/Dokumente/Jahresberichte/\\Jahresbericht-2022.pdf

[33] https://www.wikidata.org/wiki/Q74304

[34] https://github.com/AnonymousCSResearcher/FineGrainedMSCPred/blob/main/src/entitylinking/qid-msc_wikidata-seeding.csv

data, for which there was a humanly verified concept benchmark. For batch processing, we used the Wikidata 'quickstatements' toolforge.[35]

Since there are often not enough training examples per class for fine-grained classification, a classification using knowledge graph relations (linking entity keywords to subject classes) may be very beneficial to label small classes with little training data. If expedient, we will also employ costly supervised entity and category annotation in an active learning framework, a process for which we will first need to develop guidelines. Finally, we plan to explore how mathematical entity linking (MathEL) [19, 18] of formula entities to concept keyword entity names can support the classification of mathematical documents.

Our approach to enabling explainable fine-grained hierarchical multilabel classification of mathematical and medical documents using concept keywords can be generalized and applied to other libraries and schemes. Having tackled the presented classification schemes in the mathematical and medical domains, we plan to extend our research to arXiv's Category Taxonomy. Our approach will foster explainability and transparency. The transparency provided by mapping keywords to hierarchical classifications is especially crucial in fields where understanding the basis of classification decisions is essential, such as in legal or medical contexts.

Acknowledgements

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – grant 437179652, the Lower Saxony Ministry of Science and Culture, and the VW Foundation. This work was supported by the DAAD (German Academic Exchange Service) - 57515245.

preprint/short

References

[1] M. Bayer, M. Kaufhold, and C. Reuter. "A Survey on Data Augmentation for Text Classification". In: *CoRR* abs/2107.03158 (2021). arXiv: 2107.03158.

[2] E. Dunne and K. Hulek. "Mathematics subject classification 2020". In: *EMS Newsletter* 115 (2020), pp. 5–6.

[3] Z. Elberrichi, A. Belaggoun, and M. Taibi. "Medical Documents Classification Based on the Domain Ontology MeSH". In: *CoRR* abs/1207.0446 (2012). arXiv: 1207.0446.

[4] J. Eykens, R. Guns, and T. C. E. Engels. "Fine-grained classification of social science journal articles using textual data: A comparison of supervised machine learning approaches". In: *Quant. Sci. Stud.* 2.1 (2021), pp. 89–110. DOI: 10.1162/qss_a_00106.

[35] https://quickstatements.toolforge.org/\#/batch/85955

[5] D. Ginev and B. R. Miller. “Scientific Statement Classification over arXiv.org”. In: *LREC*. European Language Resources Association, 2020, pp. 1219–1226.

[6] F. Hoppe, D. Dessì, and H. Sack. “Deep Learning meets Knowledge Graphs for Scholarly Data Classification”. In: *WWW (Companion Volume)*. ACM, 2021, pp. 417–421.

[7] K. Järvelin and J. Kekäläinen. “Cumulated gain-based evaluation of IR techniques”. In: *ACM Trans. Inf. Syst.* 20.4 (2002), pp. 422–446.

[8] A. K. Kehoe and V. I. Torvik. “Predicting Medical Subject Headings Based on Abstract Similarity and Citations to MEDLINE Records”. In: *Proc. ACM/IEEE JCDL*. Ed. by N. R. Adam et al. ACM, 2016, pp. 167–170. DOI: 10.1145/2910896.2910920.

[9] T. Kusmierczyk et al. “Comparing Hierarchical Mathematical Document Clustering against the Mathematics Subject Classification Tree”. In: *Intelligent Tools for Building a Scientific Information Platform*. Vol. 467. Springer, 2013, pp. 365–392.

[10] S. M. Lundberg et al. “Explainable AI for Trees: From Local Explanations to Global Understanding”. In: *CoRR* abs/1905.04610 (2019).

[11] C. J. Mahoney et al. “A Framework for Explainable Text Classification in Legal Document Review”. In: *IEEE BigData*. IEEE, 2019, pp. 1858–1867.

[12] M. Mironczuk and J. Protasiewicz. “A recent overview of the state-of-the-art elements of text classification”. In: *Expert Syst. Appl.* 106 (2018), pp. 36–54.

[13] F. G. Mohammadi, M. H. Amini, and H. R. Arabnia. “An Introduction to Advanced Machine Learning : Meta Learning Algorithms, Applications and Promises”. In: *CoRR* abs/1908.09788 (2019).

[14] B. Müller et al. “LIVIVO - the Vertical Search Engine for Life Sciences”. In: *Datenbank-Spektrum* 17.1 (2017), pp. 29–34.

[15] S. J. Nelson et al. “Automated Assignment of Medical Subject Headings”. In: *AMIA 1999, American Medical Informatics Association Annual Symposium, Washington, DC, USA, November 6-10, 1999*. AMIA, 1999.

[16] H. Phillips. “Great Library of Alexandria”. In: *Library philosophy and practice* Aug. (2010).

[17] M. T. Ribeiro, S. Singh, and C. Guestrin. “"Why Should I Trust You?": Explaining the Predictions of Any Classifier”. In: *KDD*. ACM, 2016, pp. 1135–1144.

[18] P. Scharpf, M. Schubotz, and B. Gipp. “Fast Linking of Mathematical Wikidata Entities in Wikipedia Articles Using Annotation Recommendation”. In: *Proceedings of the Web Conference (WWW) 2021*. ACM, Apr. 2021. DOI: 10.1145/3442442.3452348.

[19] P. Scharpf, M. Schubotz, and B. Gipp. “Mathematics in Wikidata”. In: *Wikidata@ISWC*. Vol. 2982. CEUR-WS.org, 2021.

[20] P. Scharpf, M. Schubotz, and B. Gipp. "Towards Explaining STEM Document Classification using Mathematical Entity Linking". In: *CoRR* abs/2109.00954 (2021).

[21] P. Scharpf et al. "Classification and Clustering of arXiv Documents, Sections, and Abstracts, Comparing Encodings of Natural and Mathematical Language". In: *JCDL*. ACM, 2020, pp. 137–146.

[22] U. Schöneberg and W. Sperber. "POS Tagging and Its Applications for Mathematics - Text Analysis in Mathematics". In: *CICM*. Vol. 8543. Springer, 2014, pp. 213–223.

[23] M. Schubotz et al. "AutoMSC: Automatic Assignment of Mathematics Subject Classification Labels". In: *CICM*. Vol. 12236. Springer, 2020, pp. 237–250.

[24] T. Suzuki and A. Fujii. "Mathematical Document Categorization with Structure of Mathematical Expressions". In: *JCDL*. IEEE Computer Society, 2017, pp. 119–128.

[25] A. N. Tarekegn, M. Ullah, and F. A. Cheikh. "Deep Learning for Multi-Label Learning: A Comprehensive Survey". In: *CoRR* abs/2401.16549 (2024).

[26] G. V. Trunk. "A Problem of Dimensionality: A Simple Example". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 1.3 (1979), pp. 306–307. DOI: 10.1109/TPAMI.1979.4766926.

[27] Y. Wang et al. "Towards Coarse and Fine-grained Multi-Graph Multi-Label Learning". In: *CoRR* abs/2012.10650 (2020). arXiv: 2012.10650.

[28] S. M. Watt. "Mathematical document classification via symbol frequency analysis". In: *Towards Digital Mathematics Library. Birmingham, United Kingdom, July 27th, 2008* (2008), pp. 29–40.

[29] W. Wei, Z. Ji, and L. Ohno-Machado. "Using the hierarchical structure of the Medical Subject Headings (MeSH) for automatic MeSH term assignment". In: *AMIA 2016, American Medical Informatics Association Annual Symposium, Chicago, IL, USA, November 12-16, 2016*. AMIA, 2016.

[30] I. Yelmen, A. Gunes, and M. Zontul. "Multi-Class Document Classification Using Lexical Ontology-Based Deep Learning". In: *Applied Sciences* 13.10 (2023), p. 6139.

[31] R. Zhang et al. "Exploiting Local and Global Features in Transformer-based Extreme Multi-label Text Classification". In: *CoRR* abs/2204.00933 (2022).

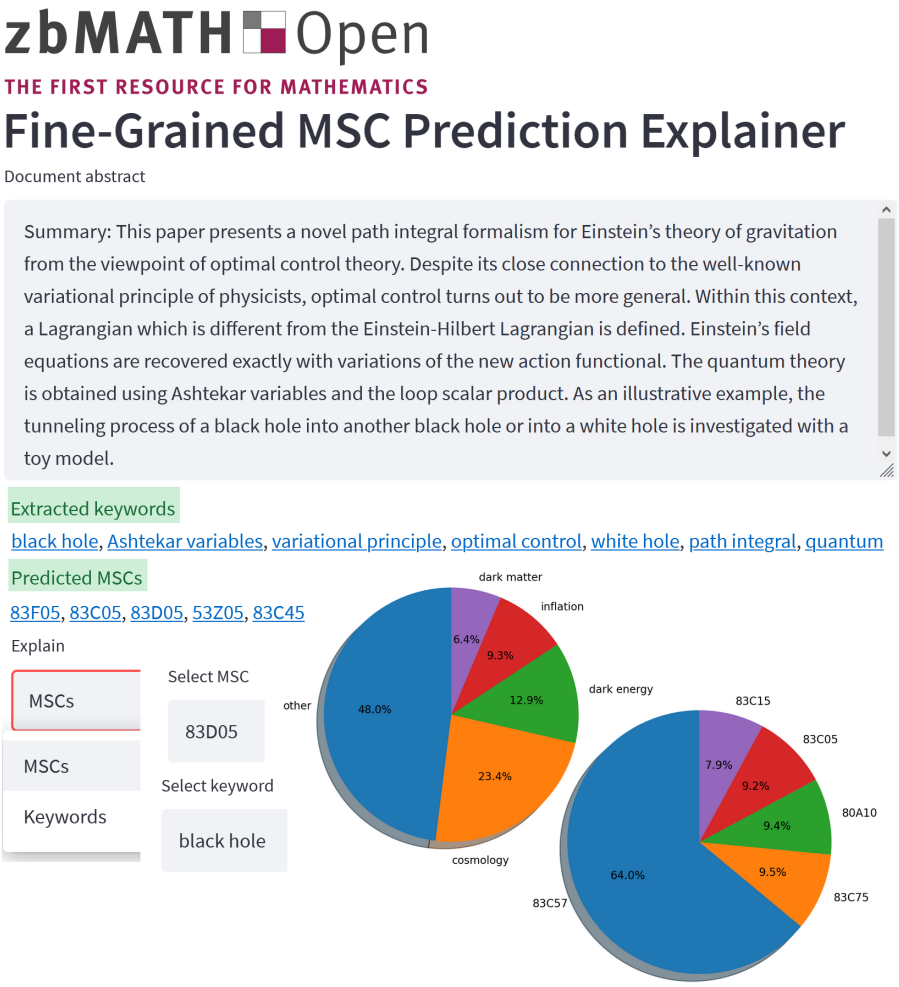

Figure 3: Demonstration interface for explainable MSC classification using an example document abstract from `https://zbmath.org`. Class-entity distributions are visualized as pie charts, allowing readers or reviewers to understand the classification reason. The demo is available at `https://automscexplainer.wmcloud.org` with an API to implement the predictions in production at `https://automscexplainer.wmcloud.org`.

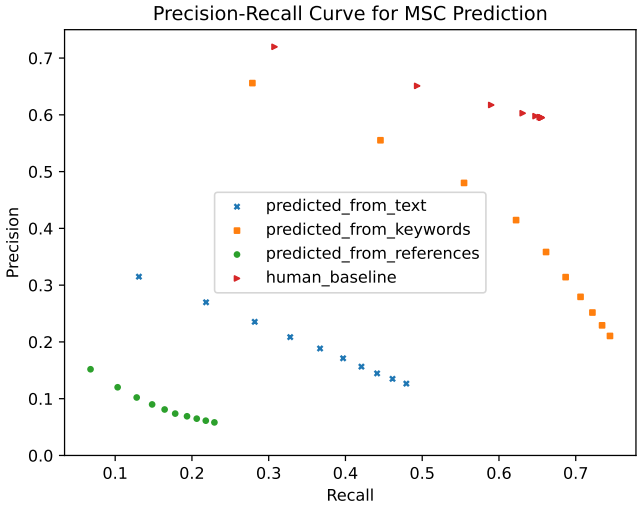

Figure 4: Mathematical texts: Precision-recall curve for MSC predictions based on text (abstract), keywords, and references compared to the human annotator baseline. The MSC prediction cutoff varies from 1 (upper left) to 10 (lower right). Figure best viewed in color.

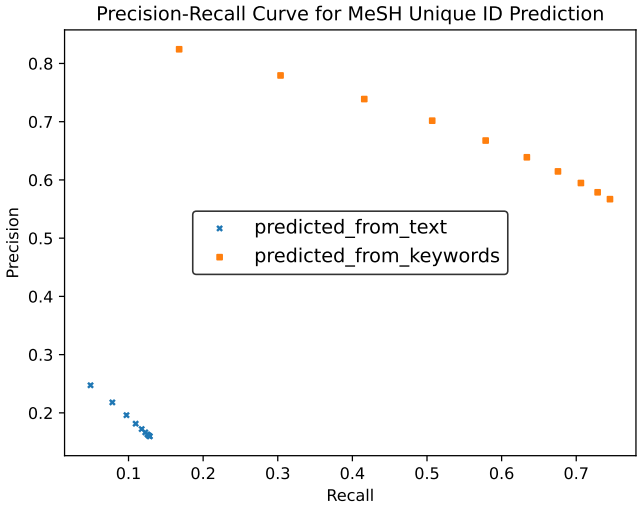

Figure 5: Medical texts: Precision-recall curve for MeSH predictions based on text (abstract) and keywords. References or a human baseline of independent annotators are unavailable. The MeSH prediction cutoff varies from 1 (upper left) to 10 (lower right). Figure best viewed in color.