big data and cognitive computing

MDPI

Article

A Comparative Evaluation of AI Approaches to Large-Scale Scientific Subject Classification

Roland Tanácsi and András Micsik *

HUN-REN Institute for Computer Science and Control (SZTAKI), 1111 Budapest, Hungary; roland.tanacsi@sztaki.hu
* Correspondence: micsik@sztaki.hu

Abstract

Background: The Hungarian Science Bibliography applies the OECD Frascati Fields of Science and Technology taxonomy for subject classification; however, approximately 80% of its records lack assigned categories. Automated large-scale classification could support retrospective completion and improve the quality of bibliographic data. Methods: We evaluated multiple artificial intelligence approaches to classifying publications into level 4 Frascati categories using only titles and keywords. Training datasets were compiled from bibliographic records and subjected to heuristic and large-language-model-based filtering to reduce noise and ambiguity. The approaches tested included statistical methods, classical machine learning classifiers, fine-tuned SciBERT models, zero-shot prompting with large language models, and a Mixture-of-Experts architecture. Results: Data quality had a stronger impact on performance than model complexity. Large-language-model-based filtering substantially improved classification results. The best-performing model, a Support Vector Classifier, achieved a weighted F1 score of 0.83, which is an outstanding result relative to state-of-the-art approaches from the literature. Conclusions: Our findings contribute new insights into classification research and may assist others in selecting appropriate solutions for real-world, large-scale bibliographic classification tasks.

Keywords: subject classification; bibliographic databases; Frascati taxonomy; scientific categorization; transformer models; SciBERT; large language models; Support Vector Classifier; data cleaning

1. Introduction

Subject classification, in the context of the scientific literature, refers to the systematic categorization of research publications according to their disciplinary focus, methodological approach, and thematic content. This taxonomic organization serves as a fundamental infrastructure for knowledge management, enabling researchers, institutions, and funding agencies to navigate the vast landscape of scientific output effectively. In large-scale bibliographic systems, subject classification also supports automated reporting, analytics, and strategic planning. The Frascati Manual, developed by the Organisation for Economic Co-operation and Development (OECD), represents one of the most widely adopted frameworks for research and development classification [1]. Named after the Italian town where it was first conceived, the Frascati classification system organizes scientific fields into a hierarchical structure spanning six levels, encompassing over 3000 distinct scientific sub-areas. Beyond Frascati, numerous other classification systems exist, including the UNESCO nomenclature for fields of science and technology, the Australian and New Zealand

Academic Editor: Xianyu Zhang

Received: 8 February 2026
Revised: 3 May 2026
Accepted: 8 May 2026
Published: 11 May 2026

Standard Research Classification (ANZSRC), the Web of Science subject categories, and discipline-specific taxonomies such as the Mathematics Subject Classification (MSC) and the Physics and Astronomy Classification Scheme (PACS).

Primarily, these systems enable systematic monitoring of research trends and emerging fields, allowing institutions and governments to identify areas of growing scientific interest and allocate resources accordingly. Classification systems facilitate evidence-based science policy-making by providing quantitative metrics for research output across different disciplines, supporting decisions about funding priorities and strategic research directions. From a practical standpoint, subject classifications enhance the discoverability of research publications, enabling more precise literature searches and reducing the time researchers spend identifying relevant work in their fields. They also support the performance of bibliometric analyses that inform research evaluation, institutional rankings, and impact assessments. Consequently, incomplete or inconsistent subject labeling limits both administrative decision-making and research evaluation processes.

On the Hungarian Science Bibliography (MTMT) platform [2], researchers can specify the thematic classification of their publications. This data can be used in various ways, but perhaps the most important use would be to monitor new developments in individual scientific fields and assess the dynamics of the area. Unfortunately, to this day, 80% of source publications (i.e., publications with Hungarian authorship) have not been classified into scientific fields, and this proportion is much worse for citing publications. This substantial coverage gap reduces the analytical value of the database and limits its usefulness for policy and evaluation purposes. We aimed to carry out mass and retrospective supplementation of these classifications, for which we examined several methods provided by artificial intelligence in terms of applicability, and we report our results here.

MTMT uses the above-mentioned Frascati taxonomy [1]. The full Frascati system is available on MTMT in both English and Hungarian. Authors or administrators can select up to five suitable scientific fields later or during data entry.

In our research, we aim to explore the available methods for automated research classification in the scenario described above and measure their performance. Unlike many previous studies, our work focuses on classification using only titles and keywords, reflecting the practical constraints of incomplete metadata availability. Through systematic experimentation, we aim to assess both the technical feasibility and practical limitations of automated subject classification in a real-world national bibliographic system.

The remainder of this paper is structured as follows. The next subsection provides an overview of the related scientific literature. Section 2 describes the datasets, preprocessing steps, and experimental setup. Section 3 presents the evaluated classification methods and quantitative results. Section 4 discusses the observed limitations, taxonomy-related challenges, and practical implications. Finally, Section 5 summarizes the main findings and outlines possible directions for future work.

Related Work

There are several relevant papers on the automated classification of scientific publications, typically relying on combinations of titles, keywords, and abstracts. However, most existing approaches incorporate abstracts or full texts to enhance semantic representation, while approaches based solely on minimal metadata (titles and keywords) remain relatively rare.

Golub [3] provides a comprehensive overview of the state of automated subject indexing in libraries as of 2021. She found that advances in machine learning and rule-based systems are promising, yet fully automated solutions remain rare. Supervised solutions give satisfactory results when at least 1000 training samples are provided for each subject

class. She states that the current key challenges are semantic understanding and the limited amount of training data.

The work presented in [4] is quite similar to ours, except it was conducted in a more limited setting: research papers from Web of Science (WOS) were classified into selected scientific categories using various neural-network-based methods. The authors compared traditional deep learning approaches and pre-trained language models (PLMs), and they found that SciBERT [5], which was pre-trained on a massive corpus of scientific publications, had the best performance. A year later, they extended their experiments and improved their results [6], but, in this case, the paper abstracts were also used for classification. Kandimalla et al. [7] performed large-scale subject category classification using deep attentive neural networks trained on millions of publications from Microsoft Academic Graph. Using titles and abstracts, they reported a micro-F1 score of 0.84 across 256 subject categories with a largely unbalanced dataset. Luo et al. [8] investigated various combinations of titles, keywords and abstracts for automatic subject indexing. They reported that when using abstracts alone as the input data for the fine-tuned model, the classification accuracy was significantly better than that achieved when using keywords or titles alone. Kafi et al. [9] experimented with an ensemble of machine learning and deep learning algorithms to classify papers into 40 WOS subject categories. With titles or keywords only, the F1 score remained below 0.70 when the majority vote of the ensembled methods was used.

The above-mentioned authors implemented multi-disciplinary classification, yet their approach differs from ours because they relied on abstracts. We had limited access to abstracts, so we could only use titles and keywords (if available) for training and classification.

LLMs4Subjects [10] was a shared task pertaining to automated subject tagging for scientific and technical records in English and German using the GND taxonomy. It was organized as part of SemEval-2025, the 19th International Workshop on Semantic Evaluation. One of the winners used an LLM-ensemble approach [11], where seven LLMs were used with few-shot examples, and the answers were aggregated. The winner of the other category [12] used the Annif tool in combination with an X-Transformer. We later introduce Annif in more detail, as we also used it for experimenting. The best results achieved in LLMs4Subjects were F1 scores of 0.34 (quantitative track) and 0.41 (qualitative track), indicating substantial performance limitations in broad, real-world subject indexing scenarios similar to ours.

A classic example of automated subject classification is Springer Nature's CSO classifier [13], which uses syntactic and semantic analysis to categorize papers based on the Computer Science Ontology (CSO) using abstracts, titles, and keywords. The CSO classifier can achieve an F1 score of 0.75, albeit within a relatively narrow scientific domain supported by a well-constructed ontology. There are several measurements for an ACM dataset [14,15]: Word2Vec embeddings are used with ML classifiers (K Nearest Neighbors, Random Forest, and Decision Trees), and for various combinations of metadata, classification performance is measured. In Table 1, we refer to the results achieved with title-and-keyword pairs.

In the climate change domain, Yang et al. [16] proposed an ontology-driven subject-indexing method integrating domain knowledge with text-based similarity measures. Voskergian and his colleagues developed a topic-model-based approach for short-text classification [17], and they achieved an F1 score of 0.86 F1 after applying this model to titles only, albeit in a true/false classification scenario for a single computer science sub-area.

Sjögårde et al. [18] evaluated automated labelling in multi-level classification systems such as MeSH and analyzed the impacts of different bibliographic fields and term-weighting strategies. Their results show that combining multiple metadata fields improves performance in large-scale hierarchical settings. In our case, constructing or applying a

comprehensive taxonomy was not feasible due to the breadth of the scientific scope and the absence of a unified cross-disciplinary vocabulary.

In [19], the authors experimented with LLMs to classify documents based on titles, keywords and abstracts according to the Universal Decimal Classification (UDC). The UDC is a very general classification system, and the results show that the task is still too hard for the models. Similarly, in [20], LLMs were used to predict Library of Congress Subject Headings (LCSHs). Here, the authors applied a three-step approach combining chain-of-thought, fine-tuning and post-filtering methods to an input consisting of titles and summaries. While a recall of 0.63 was achieved, the F1 value remained 0.30. In both cases, the classified items were provided by libraries, and their topics and vocabulary were familiar to general-purpose LLMs used, as opposed to high-level scientific material. Hawalah [21] also compared several methods for classifying full documents in Arabic, and the best result was based on vector similarity. Another language-specific study [22] classified Kazakh documents using both text and images with a custom deep neural network.

Table 1. Overview of related work, sorted by best published F1 scores. The row in italics represents the results in this paper.

Ref.	Languages	Data Used	Domain	Methods	Classes	Best F1
[5]	English	Keywords, abstract	Selected WoS categories	BERT variants	3	0.98
[4]	English	Keywords	Selected WoS categories	BERT variants	3	0.94
[22]	Kazakh	Full text and images	General science	Word2Vec, Naive Bayes, CNN	5	0.88
[8]	Chinese	Title, keywords, abstract	General science	BERT, CNN, LSTM	67	0.87
[21]	Arabic	Full text	General domains	TF-IDF, SCM	107	0.87
[17]	English	Title, abstract	Selected medical and CS	Topic model	2	0.86
	English	*Title, keywords*	*General science*	*SVC*	*37*	*0.83*
[9]	English	Title, keywords, abstract	Dimension subjects	Ensemble (ANN, CNN, SVM, etc.)	40	0.83
[14]	English	Title, keywords, abstract, etc.	Computer Science	Word2Vec, KNN, Random Forest, etc.	11	0.82
[4]	English	Keywords	Selected WoS categories	BERT variants	7	0.80
[15]	English	Title, keywords	Computer Science	Word2Vec	11	0.80
[9]	English	Title, keywords, abstract	Selected WOS subjects	Ensemble (ANN, CNN, SVM, etc.)	38	0.79
[7]	English	Title, abstract	General science	Deep attentive neural networks	81	0.76
[16]	English	Full text	Climate Change	Word2Vec, Phrase-Bert	15	0.75
[13]	English	Title, abstract	Computer Science	CSO Classifier	~14,000	0.75

Table 1. *Cont.*

Ref.	Languages	Data Used	Domain	Methods	Classes	Best F1
[19]	English, Slovenian	Title, keywords, abstract	General science	Zero-shot LLM	9	0.53
[18]	English	Title, keywords, abstract, etc.	General science	Statistical term weighting	5209	0.45
[11]	English, German	Bibliographic metadata	General (library catalog)	Few-shot LLM	78,741	0.41
[12]	English, German	Bibliographic metadata	General (library catalog)	Annif (Omikuji, XTransformer, etc.)	78,741	0.34
[20]	English	Title, abstract	General science	Fine-tuned LLMs	n/a	0.30

We present an overview of the selected papers in Table 1 ordered by the best F1 scores found in the results. Our best result, presented in this paper, is inserted as a row in italics. The methods noted in the rows above this row mostly require significantly more data (abstract or full text) to be used as an input, with a single exception, wherein the number of classes is only three. In other words, the results that are better than ours were obtained using methods that are either limited in terms of the number of classes or require the use of longer texts for classification.

2. Materials and Methods

The success of machine learning projects heavily depends on the quality, preparation, and cleaning of the training data. Our initial dataset consisted of records already classified in MTMT. A primary challenge was the uneven coverage of Frascati classifications. While some scientific fields included hundreds of thousands of publications, others had few or no records. Furthermore, many existing classifications were overly general, typically containing level 3 categories only (e.g., Medical and Health Sciences).

The Web of Science (WoS) platform was used to expand data in underrepresented categories. WoS categories are presented as a single, non-hierarchical list of nearly 300 items, which often correspond to level 4, 5, or 6 taxonomy elements in the Frascati hierarchy.

The final compilation of training and test data was based on specific criteria. Primarily, 37 subject terms corresponding to level 4 of the Frascati hierarchy were selected as possible scientific classification outputs, grouped under the 6 main parent themes at level 3 (Humanities, Agricultural Sciences, Engineering and Technology, Medical and Health Sciences, Social Sciences, and Natural Sciences). Finer (levels 5 and 6) classifications were mapped back to level 4. In this way, we adhered to the smaller and easily accessible Revised Fields of Science and Technology [23].

The publication titles and their associated keywords served as the training data inputs, as these are generally available in the MTMT database, unlike abstracts, which have <25% availability.

Based on the above guidelines, we collected more than 500,000 titles from MTMT and Web of Science. Due to licensing restrictions, the raw bibliographic data cannot be publicly released.

2.1. Data Preparation

The steps of the data preparation process are illustrated in Figure 1 and explained in detail in the rest of the subsection.

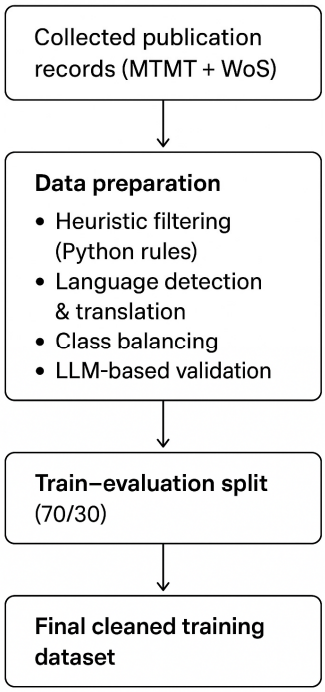

Figure 1. The process of creating the training dataset.

It was realized that the classification data from both MTMT and WoS contained unusable data that are ambiguous, incomplete, or misleading for subject classification. To eliminate this so-called noise from our dataset, two types of filters were created:

- A Python 3.10.12 script filtering data based on simple rules (e.g., too many keywords, excessively short titles, and non-English keywords);
- An LLM-based semantic validation step assessing whether the classification is unambiguous.

The script-based filtering step uses a heuristic rule set to clean the training data. Several types of anomalies were detected in the training data; these were partially due to the different practices used in the composition of titles and keywords, depending on the research domain. For example, titles in humanities tend to use quite simple wording, while in natural sciences, keyword lists can be extremely long, as they contain all related materials, equipment, and geographical regions.

Some other anomalies may have originated from erroneous data input, such as mixing languages in keywords or presenting the title and keywords in different languages.

The specific filtering rules applied to the data rows are as follows:

- Remove papers with excessively short titles (less than 3 words);
- Remove papers with excessively long titles (more than 25 words);
- Remove papers whose titles contain brackets and special characters (for example, a chemical compound);
- Remove papers that do not have enough keywords (less than 3 keywords);
- Remove papers with too many keywords (more than 10 keywords);
- Remove papers written in non-English languages (for example, Chinese or Russian).

The data from MTMT sometimes contained Hungarian titles and keywords, which first had to be detected and then translated into English. We used an XLM Roberta-based language detection model [24], and for the actual translation, the following tools provided the most accurate results: the Google Translate API and the translation model developed by the Language Technology Research Group at the University of Helsinki [25].

In the second filtering process, an LLM was used to further refine the data that was already passed through the initial heuristic filter. The prompt instructs the LLM to analyze each article's title, keywords, and current subject tag while also considering a list of all possible subject tags. Entries judged to be ambiguous or incorrectly labeled were removed from the training set.

A small portion of the articles in MTMT are multidisciplinary (~5% of all data) and thus can fall under more than one category. Thus, we considered a multi-label classification approach where an article could fall under several categories. However, this could present practical challenges, such as the collection of adequate and ample examples for all the different multi-label combinations. Given that there was a small proportion of such papers and that most other papers were also suitable for the multi-class classification approach, we ultimately aimed to filter out entries that were ambiguously classifiable.

The LLM-based validation was executed using HUN-REN Cloud's GenAi4Science service [26], where several state-of-the-art models are available with parallel processing. We experimented with several models, including Llama3.1:8B, Llama3.2:3B, and Qwen3:32B. The total runtime for this cleaning step was highly variable: while the smaller Llama models typically completed the entire process over a weekend, the larger Qwen3 model required 2 to 3 weeks of continuous operation.

In the following result tables, the term *cleaned* refers to training data filtered using these two methods. Table 2 shows the sizes of our datasets. In all three cases, cleaning through the LLM phase removed ca. 40% of the items from the dataset.

Table 2. The datasets used in the experiments.

Dataset Name	Items per Subject	Item Total After Balancing and Basic Cleaning	Item Total After LLM Cleaning
MTMT_small	800–1200	46,810	27,412
MTMT_medium	4000–6000	217,676	127,549
MTMT_large	8000–12,000	435,352	255,063

Initially, we collected training data from the MTMT database, but the data obtained this way was very imbalanced. Then, we complemented the dataset using available Web of Science (WoS) subjects for papers in MTMT, resulting in a large dataset. For this, we created a mapping from WoS subject categories to Frascati subjects.

Finally, to determine how training data size affects the method's performance, a medium-sized dataset was created by halving the large dataset for each subject. For the balancing of datasets, we used a Python script to address class imbalance by downsampling majority classes. We also experimented with oversampling via Easy Data Augmentation (EDA) [27]. The main goal was to generate new, synthetic data points for underrepresented classes by paraphrasing existing titles and extracting new keywords from these paraphrased titles. However, this augmented dataset made the model's performance a little worse.

2.2. Evaluations of Dataset Preparation

As we had several options and decision points when creating the dataset to be used for the classification experiments, we wanted to ensure the right choices were made with the following measurements.

Table 3 assures that larger training sets produce better results, although the difference between the medium-sized and large datasets is merely 2%. In our next experiments, we investigated how cleaning the dataset would affect the quality of the model based on the cleaning methods described in Section 2.1. For the level 3 categorization, basic, heuristic cleaning improved the F1 by 0.06, and LLM cleaning added a further 0.03 points to the F1

score. However, at level 4, the increase in F1 after LLM cleaning varied between 0.09 and 0.21 depending on the science subdomain. Furthermore, we returned 10% of the previously filtered data (called noise) into the holdout test set to further check the usefulness of our cleaning methods. Adding 10% noise to the datasets did not change the average of the F1 scores.

Table 3. Performance of a fine-tuned SciBERT model reported by training dataset size. Metrics are given as weighted averages (Train/Eval ratio was 70–30%).

Dataset	Size	Topics	F1-Score
MTMT_small	27,412	All (37)	0.64
MTMT_medium	127,549	All (37)	0.76
MTMT_large	255,063	All (37)	0.78

3. Results

In this section, we present and analyze our experiments. The methods are introduced in the order in which they influenced our research decisions. We first evaluate simpler classification approaches before moving to transformer-based models and hybrid architectures.

3.1. Annif

Annif is an open-source software developed by the National Library of Finland for automated classification [28]. Its advantage is flexibility, supporting multiple algorithms and models (also via external libraries). Furthermore, these algorithms can be combined using ensemble models; for example, the nn-ensemble model combines algorithm suggestions using predefined weights.

The TF-IDF (Term Frequency-Inverse Document Frequency) backend implements a baseline algorithm for automated subject indexing based on the frequency with which terms appear. Omikuji is an implementation of a family of efficient machine learning algorithms for multilabel classification based on the idea of partitioned label trees, including Parabel and Bonsai [29]. We attained the best results using the Bonsai-style configuration with the parameters cluster_balanced = False, cluster_k = 100, and max_depth = 3.

We used the automated hyperparameter optimization (hpo) feature built into Annif to find the best weights used in ensemble models. Based on our experiments, the most effective combination was a 1:8 ratio combination of TF-IDF and Omikuji-Bonsai.

3.2. SciBERT

SciBERT [5] is an LLM built on Google's BERT architecture, specialized specifically for processing scientific texts. Because it was trained on domain-specific corpora, it handles scientific terminology more effectively than general-purpose language models.

Initial experiments with BERT and ModernBERT [30] showed limited performance in the science domain. After switching to SciBERT, we observed a substantial improvement. As a next step, we wanted to determine the degree to which fine-tuning SciBERT could improve its classification performance.

Fine-tuning was implemented using PyTorch 2.9.1 and Hugging Face Transformers. Balanced inverse-frequency weighting was applied to mitigate residual class imbalance. We applied early stopping, monitoring the F1 value, typically stopping the process after 10–12 epochs. Training was using an NVIDIA V100 GPU with 16GB of VRAM, with training time averaging anywhere from 30–45 min to 10–12 h or even more (depending on the size of the training dataset and the early-stopping mechanism).

3.3. General-Purpose LLMs

General-purpose LLMs can be instructed to classify scientific publications using specific system prompts without training. The advantage is immediate applicability, while the disadvantages include significant hardware requirements for local execution and expensive subscriptions.

The related experiments were structured as several groups, testing zero-shot classification capabilities across different datasets and cleaning levels. Table 4 lists the results of the experiment for all LLMs tested on the small dataset. Due to time limits and the fact that all LLMs had very similar performance, for the large dataset, only the LLM Qwen was run (Table 5).

Table 4. Classification performance of various methods on a single topic with 7 subtopics.

Classification Method	Dataset	Topics	Micro F1	Weighted F1
Annif	MTMT_small	Natural sciences (7)	0.811	-
SciBERT	MTMT_large_qwen3	Natural sciences (7)	0.918	0.918
SVC	MTMT_large_qwen3	Natural sciences (7)	**0.920**	**0.920**
Gemma3:27B	MTMT_small	Natural sciences (7)	0.800	0.810
Llama3.3:70B	MTMT_small	Natural sciences (7)	0.800	0.800
Mistral-small3.1:24B	MTMT_small	Natural sciences (7)	0.800	0.800
Deepseek-R1:70B	MTMT_small	Natural sciences (7)	0.780	0.780
QWQ:32B	MTMT_small	Natural sciences (7)	0.800	0.800

Bold formatting indicates the best result.

Table 5. Classification performance of various methods for all subfields of science.

Classification Method	Dataset	Topics	Micro F1	Weighted F1
MoE-SciBERT	MTMT_large_qwen3	All (37)	0.77	0.76
SciBERT	MTMT_large_qwen3	All (37)	0.79	0.78
SVC	MTMT_large_qwen3	All (37)	**0.83**	**0.83**
Annif	MTMT_large_qwen3	All (37)	0.75	-
Qwen3	MTMT_large_llama31	All (37)	0.56	0.55

Bold formatting indicates the best result.

3.4. MoE-SciBERT

In this experiment, we implemented a simple script mimicking a Mixture-of-Experts (MoE) architecture. The core components of this system are specialized SciBERT models, each fine-tuned for a particular Frascati subject domain. Our MoE system used two distinct model types:

- Router model: This is a SciBERT classifier trained to predict one of the six level 3 Frascati domains. It directs the input to the appropriate Expert model.
- Expert models: These constitute six separate SciBERT classifiers, each trained exclusively on level 4 subfields within one domain.

The motivation for this architecture was a desire to decompose the problem, first into a broad category selection (6 fields) and then into a specific classification within a focused domain (5–10 subfields). We hoped to achieve better performance than a single, complex

model attempting to classify everything at once would yield. The results of this MoE experiment are presented in Table 5.

3.5. Embedding-Based Classification

In this experiment, we evaluated classical machine learning classifiers from Scikit-learn [31]. These models are less complex than large language models and much quicker to train.

Instead of token-level input, each publication was represented by a single embedding vector. We then trained various ML classifier models using these embedding vectors and compared their performance.

The classifiers selected from Scikit-learn 1.7.2 included LogisticRegression, RandomForestClassifier, LinearSVC (svm_linear), SVC (svm_rbf), GradientBoostingClassifier, ExtraTreesClassifier, MLPClassifier, KNeighborsClassifier, GaussianNB and SGDClassiifier.

Across most datasets, the Support Vector Classifier (SVC) consistently achieved the best performance, likely due to its suitability for high-dimensional feature vectors. The small dataset showed exceptions to this rule, where other models performed better for certain categories: KNN for 'Natural sciences', Logistic Regression for 'Medical and health sciences,' and MLPClassifier for 'Agricultural sciences'. Despite these exceptions, SVC was the clear overall winner across all other datasets and scenarios.

3.6. Quantitative Results

Figure 2 contains an overview of how the classification methods are used to provide a topic prediction output. First, the measurements of a smaller classification task are presented, with 7 possible topics; then, we move on to the full classification task for 37 possible topics.

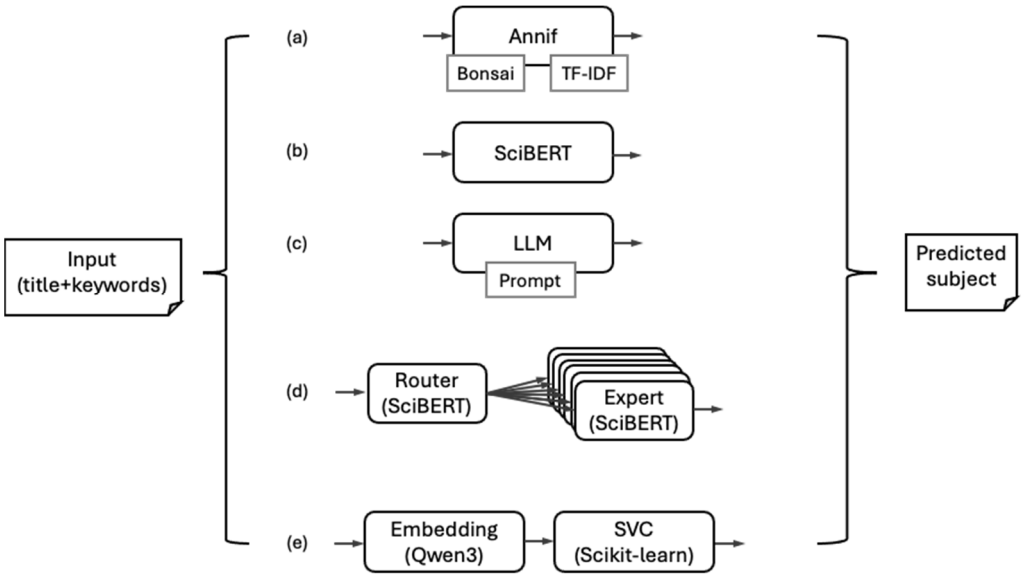

Figure 2. The prediction of scientific topics during evaluation of the methods: (a) Annif—Section 3.1; (b) SciBERT—Section 3.2; (c) general-purpose LLMs—Section 3.3; (d) MoE-SciBERT—Section 3.4; and (e) Support Vector Classifier (SVC)—Section 3.5.

When evaluating a classification experiment, three different aggregate metrics are calculated to assess performance:

- Macro F1, which averages the metric across all classes equally, effectively penalizing poor performance on rare classes;

- Micro F1, which calculates the metric globally across all predictions, with larger classes having a greater effect on the score;
- Weighted F1, which calculates an average wherein each class's score is weighted by its sample count (support), offering a balanced view that accounts for class imbalance.

We found the weighted F1 to be the most expressive measure, and therefore we selected it whenever possible to save space in the tables.

Table 4 contains the measurements used to classify the seven subfields of natural sciences (mathematics, computer and information sciences, physical sciences, chemical sciences, earth and environmental sciences, biological sciences, other natural sciences). Compared to other fields, it is relatively simple for a human to differentiate the subfields of natural sciences.

As shown in Table 4, five smaller LLMs were tested with zero-shot prompting, and they produced the worst performance. A slightly better result was achieved by Annif using the ensemble of TF/IDF and Omikuji methods in a 1:8 ratio. For Annif's internal parameter optimization, the weighted F1 calculation was not available. The winner in this case was a Support Vector Classifier (SVC) trained on the embedding vectors of the input. SciBERT, with a small difference, became the second-best performer.

As our main goal, we wanted to compare solutions for the classification according to the Revised Fields of Science and Technology (2007) [23], which is equivalent to levels 3 and 4 in the Frascati hierarchy [1]. The tested methods had to find the best choice from among 37 topics. The MoE approach was tested with all three types of LLM cleaning (shown in the dataset name), and the Qwen3 cleaning came first, yet it was still behind the single SciBERT model approach. When a fine-tuned SciBERT model was used to generate the embedding vectors for SVC training, it achieved an F1 score of only 0.76, but after Qwen3-Embedding-8B was used for embedding generation and the embedding dimensions were increased from 768 to 4096, the SVC solution became the winner.

4. Discussion

The classification task presented in this paper proved to be more difficult than expected. Although text classification is often regarded as a mature and largely solved problem in controlled benchmark settings, our experiments demonstrate that real-world, large-scale scientific-subject indexing remains highly challenging. This is because of various aspects of the problem, mostly related to the training data.

In order to better understand our results, we examined the confusion matrix for our best-performing model. Table 6 contains the fields where the proportion of correct predictions was less than 60%. It reflects the idea that it is hard to decide what belongs to "other". There is also a chance that some of the human categorizations for "other" were inaccurate, confusing the model during learning.

Table 6. Scientific fields for which correct prediction was below 60%.

Topic	True Positives
Other humanities	42%
Other engineering and technologies	40%
Other social sciences	34%

Table 7 elaborates on the details of confusion, showing the more frequent pairs of correct and predicted topics. These pairs show that the classifier often prefers exact domains such as Arts or Sociology over Other categories. Furthermore, we see some confusion in the areas of biology and medicine and in the areas of Information engineering and Computer

and information science. We feel that these symptoms point beyond our data and methods, relating to the subject taxonomy formulation. Neither Web of Science nor Scopus has Other categories, so it is almost impossible to find good training samples for these. Moreover, Other categories function as residual containers rather than semantically coherent groups, making them inherently unsuitable for supervised learning.

Table 7. Scientific fields for which the false prediction proportion was above or equal to 10% (arranged alphabetically).

Topic	Predicted Topic	Occurrence
Biological sciences	Basic medicine	16%
Clinical medicine	Basic medicine	11%
Clinical medicine	Health sciences	10%
Electrical engineering, Electronic engineering, Information engineering	Computer and information sciences	10%
Materials engineering	Chemical sciences	10%
Other agricultural sciences	Agriculture, Forestry, and Fisheries	13%
Other engineering and technologies	Chemical sciences	13%
Other engineering and technologies	Mechanical engineering	10%
Other humanities	Arts	20%
Other humanities	Sociology	11%
Other social sciences	Sociology	18%

As described in Section 2.2, data cleaning is clearly beneficial for the model's performance, and open-source LLMs provide an economic solution to filtering out ambiguous items from the training data. When the model encounters fewer ambiguous or unclassifiable items during training, its performance increases by 25% (on average). However, the LLM-cleaning phase filters out 40% of the training samples (on average), which means that even more raw samples must be collected. This highlights a trade-off between dataset size and semantic consistency: aggressive cleaning improves model reliability but increases the cost of corpus construction.

As for the generative models, the main performance factor has been shown to be the basic training material for the model. SciBERT was trained on scientific text, and it clearly outperformed the other general BERT models and LLMs. Unfortunately, fine-tuning an LLM with lots of scientific text was not feasible in this project, but future work could explore whether domain-adapted LLMs narrow the observed performance gap.

Another interesting conclusion is that decomposing the task into smaller classification subtasks using a Mixture-of-Experts (MoE) architecture did not improve performance; in fact, it resulted in slightly lower F1 scores. This suggests that error propagation between routing and expert stages may offset the theoretical advantages of hierarchical decomposition.

Although SVC became the clear winner, there may be space for further improvement via specialized solutions for Other categories and through the constant development of LLMs and neuro-symbolic reasoning.

5. Conclusions

In this study, we examined the feasibility of large-scale automated subject classification of scientific publications in the Hungarian Science Bibliography using only titles and keywords. By comparing statistical approaches, classical machine learning models,

transformer-based models, general-purpose LLMs, and hybrid architectures, some important lessons emerged, and we succeeded in filling a gap regarding classification experiments with an outstanding F1 score.

The most decisive factor influencing performance was training data quality. Removing ambiguous and noisy samples proved essential in a setting with overlapping scientific fields, although this required discarding 40–60% of the original data. Dataset size also mattered: performance improved steadily when moving from the small to large datasets, though gains diminished at higher volumes, suggesting that conceptual clarity of labels is as important as quantity.

Among the tested models, the embedding-based Support Vector Classifier (SVC) method clearly outperformed all alternatives with respect to the full 37-class task, achieving a weighted F1 of 0.83. The embedding dimension played an important role here.

Interestingly, general-purpose LLMs exhibited the lowest performance among all the tested solutions. Another unexpected result was that the Mixture-of-Experts architecture did not surpass a single fine-tuned SciBERT model, but classical ML classifiers trained on high-quality embeddings performed competitively.

Error analysis revealed that many misclassifications occurred between semantically adjacent or inherently ambiguous categories, particularly in engineering, medical sciences, and "Other" fields.

We feel that a fully automated retrospective classification of millions of records is not yet advisable with the current solution, but with small improvements and careful planning, a partial batch classification with human introspection might be possible, and we will continue our work in this direction. In the meantime, the performance achieved is sufficient for decision-support applications, such as providing ranked subject suggestions during data entry or assisting administrators with validation.

Although text classification is often considered a largely solved problem in contemporary AI research, our results fill a gap in the current literature and may help others to move in the right direction when dealing with real-world and large-scale bibliographic classification tasks.

Author Contributions: Conceptualization, A.M.; methodology, A.M. and R.T.; software, R.T.; validation, A.M. and R.T.; formal analysis, A.M. and R.T.; investigation, A.M. and R.T.; resources, A.M. and R.T.; data curation, R.T.; writing—original draft preparation, A.M. and R.T.; writing—review and editing, A.M.; visualization, R.T.; supervision, A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Hungarian Ministry of Innovation and Technology NRDI Office within the framework of the Artificial Intelligence National Laboratory Program (MILAB, RRF-2.3.1-21-2022-00004).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: We deposited a dataset containing key assets of our research in the ARP Data Repository at https://hdl.handle.net/21.15109/ARP/VWQFD2 (accessed on 3 May 2026). The dataset contains code and data samples used to train and evaluate models, evaluation results for experiments (including the full confusion matrix for Tables 6 and 7) in CSV format, and the model with best evaluation results.

Acknowledgments: We are grateful that were provided with the opportunity to use the GenAI4Science service and the GPUs of HUN-REN Cloud [26] (https://science-cloud.hu/en, accessed on 3 May 2026), which helped us achieve the results published in this paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. OECD. *Frascati Manual 2015: Guidelines for Collecting and Reporting Data on Research and Experimental Development*; OECD: Paris, France, 2015. [CrossRef]
2. Holl, A.; Makara, G.; Micsik, A.; Kovács, L. MTMT: The Hungarian Scientific Bibliography. In *5th Samos Summit on ICT-Enabled Governance*; Samos: Samos Island, Greece, 2014. Available online: https://eprints.sztaki.hu/8020/ (accessed on 3 May 2026).
3. Golub, K. Automated Subject Indexing: An Overview. *Cat. Classif. Q.* **2021**, *59*, 702–719. [CrossRef]
4. Rostam, Z.R.K.; Kertész, G. Fine-Tuning Large Language Models for Scientific Text Classification: A Comparative Study. In *2024 IEEE 6th International Symposium on Logistics and Industrial Informatics (LINDI)*; IEEE: New York, NY, USA, 2024; pp. 000233–000238. [CrossRef]
5. Beltagy, I.; Lo, K.; Cohan, A. SciBERT: A Pretrained Language Model for Scientific Text. *arXiv* **2019**. [CrossRef]
6. Rostam, Z.R.K.; Kertész, G. Advancing Scientific Text Classification: Fine-Tuned Models with Dataset Expansion and Hard-Voting. In *Proceedings of the 2025 IEEE 23rd World Symposium on Applied Machine Intelligence and Informatics (SAMI)*; IEEE: New York, NY, USA, 2025. [CrossRef]
7. Kandimalla, B.; Rohatgi, S.; Wu, J.; Giles, C.L. Large Scale Subject Category Classification of Scholarly Papers with Deep Attentive Neural Networks. *Front. Res. Metr. Anal.* **2021**, *5*, 600382. [CrossRef] [PubMed]
8. Luo, X.; Mutalib, S.; Aris, S.R.S. Chinese Paper Classification Based on Pre-Trained Language Model and Hybrid Deep Learning Method. *IAES Int. J. Artif. Intell. (IJ-AI)* **2025**, *14*, 641–649. [CrossRef]
9. Kafi, A.A.; Banshal, S.K.; Sultana, N.; Gupta, V. Source Recommendation System Using Context-Based Classification: Empirical Study on Multi-Level Ensemble Methods. *J. Scientometr. Res.* **2024**, *13*, 475–484. [CrossRef]
10. D'Souza, J.; Sadruddin, S.; Israel, H.; Begoin, M.; Slawig, D. SemEval-2025 Task 5: LLMs4Subjects-LLM-based Automated Subject Tagging for a National Technical Library's Open-Access Catalog. *arXiv* **2025**. [CrossRef]
11. Kluge, L.; Kähler, M. DNB-AI-Project at SemEval-2025 Task 5: An LLM-Ensemble Approach for Automated Subject Indexing. *arXiv* **2025**. [CrossRef]
12. Suominen, O.; Inkinen, J.; Lehtinen, M. Annif at SemEval-2025 Task 5: Traditional XMTC augmented by LLMs. *arXiv* **2025**. [CrossRef]
13. Salatino, A.; Osborne, F.; Motta, E. CSO Classifier 3.0: A Scalable Unsupervised Method for Classifying Documents in Terms of Research Topics. *Int. J. Digit. Libr.* **2022**, *23*, 91–110. [CrossRef]
14. Mustafa, G.; Usman, M.; Afzal, M.; Shahid, A.; Koubaa, A. A Comprehensive Evaluation of Metadata-Based Features to Classify Research Paper's Topics. *IEEE Access* **2021**, *9*, 133500–133509. [CrossRef]
15. Mustafa, G.; Usman, M.; Yu, L.; Afzal, M.; Sulaiman, M.; Shahid, A. Multi-Label Classification of Research Articles Using Word2Vec and Identification of Similarity Threshold. *Sci. Rep.* **2021**, *11*, 21900. [CrossRef]
16. Yang, H.; Wang, N.; Yang, L.; Liu, W.; Wang, S. Research on the Automatic Subject-Indexing Method of Academic Papers Based on Climate Change Domain Ontology. *Sustainability* **2023**, *15*, 3919. [CrossRef]
17. Voskergian, D.; Bakir-Gungor, B.; Yousef, M. TextNetTopics Pro, a Topic Model-Based Text Classification for Short Text by Integration of Semantic and Document-Topic Distribution Information. *Front. Genet.* **2023**, *14*, 1243874. [CrossRef]
18. Sjögårde, P.; Ahlgren, P.; Waltman, L. Algorithmic Labeling in Hierarchical Classifications of Publications: Evaluation of Bibliographic Fields and Term Weighting Approaches. *J. Assoc. Inf. Sci. Technol.* **2020**, *72*, 853–869. [CrossRef]
19. Borovič, M.; Tomovski, E.; Dobnik, T.L.; Majninger, S. Evaluating Proprietary and Open-Weight Large Language Models as Universal Decimal Classification Recommender Systems. *Appl. Sci.* **2025**, *15*, 7666. [CrossRef]
20. Liu, J.; Song, X.; Zhang, D.; Thomale, J.; He, D.; Hong, L. A Hybrid Framework for Subject Analysis: Integrating Embedding-Based Regression Models with Large Language Models. *Proc. Assoc. Inf. Sci. Technol.* **2025**, *62*, 445–457. [CrossRef]
21. Hawalah, A. Semantic Ontology-Based Approach to Enhance Arabic Text Classification. *Big Data Cogn. Comput.* **2019**, *3*, 53. [CrossRef]
22. Bogdanchikov, A.; Ayazbayev, D.; Varlamis, I. Classification of Scientific Documents in the Kazakh Language Using Deep Neural Networks and a Fusion of Images and Text. *Big Data Cogn. Comput.* **2022**, *6*, 123. [CrossRef]
23. Fields of Science and Technology. Wikipedia. 2025. Available online: https://en.wikipedia.org/wiki/Fields_of_Science_and_Technology (accessed on 3 May 2026).
24. Hugging Face. Simoneteglia/Xlm-Roberta-Europarl-Language-Detection. Available online: https://huggingface.co/simoneteglia/xlm-roberta-europarl-language-detection (accessed on 3 May 2026).
25. Hugging Face. Helsinki-NLP/opus-mt-hu-en. Available online: https://huggingface.co/Helsinki-NLP/opus-mt-hu-en (accessed on 3 May 2026).
26. Héder, M.; Rigó, E.; Medgyesi, D.; Lovas, R.; Tenczer, S.; Török, F.; Farkas, A.; Emődi, M.; Kadlecsik, J.; Mező, G.; et al. The Past, Present and Future of the ELKH Cloud. *Inf. Társad.* **2022**, *22*, 128. [CrossRef]
27. Wei, J.; Zou, K. EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. *arXiv* **2019**. [CrossRef]

28. Suominen, O. Annif: DIY automated subject indexing using multiple algorithms. *LIBER Q. J. Assoc. Eur. Res. Libr.* **2019**, *29*, 1–25. [CrossRef]
29. Khandagale, S.; Xiao, H.; Babbar, R. Bonsai: Diverse and shallow trees for extreme multi-label classification. *Mach. Learn.* **2020**, *109*, 2099–2119. [CrossRef]
30. Warner, B.; Chaffin, A.; Clavié, B.; Weller, O.; Hallström, O.; Taghadouini, S.; Gallagher, A.; Biswas, R.; Ladhak, F.; Aarsen, T.; et al. Smarter, Better, Faster, Longer: A Modern Bidirectional Encoder for Fast, Memory Efficient, and Long Context Finetuning and Inference. *arXiv* **2024**. [CrossRef]
31. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.