Overview of the FoRC@NSLP2025 Shared Task: Field of Research Classification for Computational Linguistics and Natural Language Processing Publications

Maria Francis[1,2], Raia Abu Ahmad[1,*], Ekaterina Borisova[1] and Georg Rehm[1,3]

[1]*Deutsches Forschungszentrum für Künstliche Intelligenz GmbH, Berlin, Germany*
[2]*Center for Mind/Brain Sciences (CIMeC), University of Trento, Trento, Italy*
[3]*Humboldt-Universität zu Berlin, Germany*

Abstract

This paper provides an overview of the second iteration of the Field of Research Classification (FoRC) shared task, conducted as part of the Natural Scientific Language Processing (NSLP) Workshop 2025. The FoRC shared task focused on the fine-grained classification of computational linguistics and natural language processing scholarly papers using Taxonomy4CL, a hierarchical taxonomy of 170 (sub-)topics over three levels of increasing granularity. For this year's edition, we re-used the FoRC4CL dataset and extended it with over 41,000 weakly labelled articles from the ACL Anthology, using a logistic regression model for label creation. The shared task received a single submission, which employed a two-stage approach combining k-nearest neighbors over title and abstract embeddings with zero-shot prompting using large language models. Their approach achieved a micro-F1 score of 0.68 and a macro-F1 of 0.66, outperforming our baselines as well as results from last year's iteration.

Keywords

field of research classification, research topic classification, shared task, scholarly information processing

1. Introduction

In the face of an increasingly interconnected research landscape, effectively organising scholarly knowledge is an essential task. As the number of scientific publications continues to expand – doubling approximately every 17 years [1] – efficient ways to organise and navigate scholarly literature are becoming more relevant. Digital platforms, such as academic search engines, bibliographic databases, and knowledge graphs, now play a crucial role in representing both the content of individual works and the relationships among them [2, 3, 4, 5]. Such systems depend on classification schemes to provide a standardised way to group papers by their disciplinary focus, which in turn plays a role in positioning publications within their research contexts.

Field of research classification (FoRC), however, is not an easy task. The first major challenge lies in defining and maintaining a suitable taxonomy; scientific disciplines are constantly evolving, giving rise to novel subfields or intersections between fields. In addition, the boundaries between research areas are rarely clear-cut, with some often blending into each other. Furthermore, the label space is inherently large and is hierarchical in nature. While there is prior work in FoRC [6, 7, 8], progress has been limited by the absence of standardised evaluation protocols, taxonomies, or benchmarks, making the comparison of methods difficult. Additionally, broad disciplinary classifications often fail to capture the nuanced structure of research within a specific field. Fine-grained, field-specific classification schemes are essential for supporting detailed content discovery, research mapping, and trend analysis. A notable example is computational linguistics (CL) and natural language processing (NLP), where

2nd International Workshop on Natural Scientific Language Processing and Research Knowledge Graphs (NSLP 2025), co-located with ESWC 2025, June 01–02, 2025, Portorož, Slovenia

*Corresponding author.

maria.francis287@gmail.com (M. Francis); raia.abu_ahmad@dfki.de (R. Abu Ahmad); ekaterina.borisova@dfki.de (E. Borisova); georg.rehm@dfki.de (G. Rehm)

0009-0007-7638-9963 (M. Francis); 0009-0004-8720-0116 (R. Abu Ahmad); 0000-0002-3447-9860 (E. Borisova); 0000-0002-7800-1893 (G. Rehm)

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

subfields evolve rapidly and often overlap [9, 10, 11]. Yet, one of the field's primary resources, the ACL Anthology [12], does not offer a built-in topic classification system, limiting its application to structured search, exploration, and analysis.

To address this gap, we organise the second iteration of the *Field of Research Classification (FoRC) shared task* at the *Natural Scientific Language Processing (NSLP) Workshop 2025*. The previous iteration hosted two subtasks: 1) single-label classification of general academic papers into a broad research taxonomy, and 2) fine-grained multi-label classification of CL/NLP publications [13]. This year, we focus exclusively on the second subtask. We use the same dataset as the previous iteration, namely FoRC4CL – a manually annotated corpus of 1,500 ACL Anthology papers labelled using Taxonomy4CL, a hierarchical taxonomy of CL/NLP research areas [11]. To encourage participation and facilitate the exploration of low-resource learning methods, for this year's iteration, we expand the dataset by releasing a large-scale, weakly labelled extension consisting of over 41,000 ACL publications [14].[1]

The task was hosted on CodaBench [15], and although ten teams registered to participate, there was only one system submission. Nevertheless, this submission demonstrates substantial performance enhancements over last year's results, improving macro-precision from 0.39 to 0.65 and macro-F1 from 0.43 to 0.66 – an increase of 0.26 and 0.23, respectively. The shared task had the following schedule:

- Training and testing data release: February 18, 2025
- System submissions deadline: March 25, 2025
- Paper submissions: March 27, 2025
- Notification of acceptance: April 10, 2025
- Camera-ready Submission: April 17, 2025

The rest of the paper is structured as follows. Section 2 presents related work on extreme multi-label classification techniques, which achieved strong performance in last year's shared task, as well as related work in weak supervision. Section 3 provides a detailed description of the task. Section 4 outlines the methodology for constructing and weakly annotating the supplementary dataset. Section 5 presents the baseline systems and submitted models from both this and last year's iterations. Finally, Section 6 offers a discussion of the results and findings and Section 7 concludes our paper.

2. Related Work

A wide range of approaches have been proposed to address Extreme Multi-Label Classification (XMLC) [16], a task characterised by a large label space and label imbalance. Although XMLC problems typically span far more classes than seen in FoRC, the highest-performing submission from the 2024 iteration achieved good results by treating it as an XMLC problem, demonstrating the adaptability of this paradigm even in more constrained settings [17].

One approach for XMLC is the use of One-vs-All classifiers [18, 19, 20], which treat each label as an independent binary classification task. While conceptually simple, these methods face scalability challenges due to their high computational cost associated with training and inference across thousands of binary classifiers [20].

To mitigate this, embedding-based approaches have been proposed, which project instances and labels into a shared low-dimensional space and perform prediction via similarity search. For example, SLEEC [21] uses sparse local embeddings to capture non-linear label manifolds, while its successor, AnnexML [22], improves scalability by constructing a k-nearest neighbors (k-NN) graph of label embeddings for approximate nearest neighbor search.

Tree-based approaches, such as Parabel [23], Bonsai [24], or FastXML [25], tackle XMLC by recursively splitting the feature or label space, reducing both training and prediction complexity while maintaining competitive performance.

[1] The dataset is publicly available at https://zenodo.org/records/14901529

More recently, generative models have reframed XMLC as a sequence generation task. XLGen [26], for example, fine-tunes T5 [27] and BART [28] to generate label sequences, further enhancing output coherence by integrating hierarchical clustering to better model label dependencies.

In cases where annotated data is scarce or expensive to obtain, weak supervision has emerged as an alternative to traditional supervised learning. Weak supervision refers to strategies that generate noisy labels using inexpensive or indirect sources, for instance external knowledge bases [29], crowdsourced annotations of varying quality [30], heuristic rules [31], feature annotations [32], or predictions from pre-trained models.

To avoid poor generalisation, specialised learning strategies exist to mitigate the noise introduced by the weak labels – BOND [33] and COSINE [34], for example, apply teacher-student frameworks. Another strategy is to filter weakly labelled data for instances where labels are likely to be false – CleaR [35] preferentially exposes parameter-efficient fine-tuning modules to clean data while bypassing the noisy ones. Non-neural approaches such as bagging, boosting, outlier detection, and k-NN have also been employed to identify and discard erroneous labels [36, 37, 38, 39].

The effectiveness of weak supervision, however, has been questioned in recent work. Zhu et al. [40] demonstrate that many weakly supervised learning methods depend heavily on access to a clean validation set, and that models trained directly on this clean subset can outperform weak supervised learning (WSL) methods when even 15 cleanly labelled examples per class are available. Nonetheless, these findings are based on balanced classification tasks with up to 10 classes. In contrast, the FoRC shared task has 170 labels with a highly unbalanced label distribution, where many classes indeed have fewer than 10 annotated examples. Therefore, we find that weak supervision may still prove useful in the context of our task.

3. Task Description

The FoRC 2025 task is a fine-grained multi-label classification task with 170 total classes represented over three hierarchical levels. The task is described as follows:

> *Given an ACL publication's (meta-)data, predict all associated labels that describe the main contributions of the publication from a taxonomy of 170 (sub-)topics in CL/NLP.*

We use FoRC4CL [11] as a dataset, which consists of 1,500 CL/NLP articles extracted from the ACL Anthology [12] from the years 2016 to 2022. The data includes three splits, which were created by shuffling the corpus randomly into 70/15/15 for training, validation, and testing, respectively. The articles were manually annotated by graduate students using Taxonomy4CL, a three-level taxonomy of (sub-)topic in CL/NLP. For more detailed information on the taxonomy creation, its topics, and the annotation process of FoRC4CL, we refer the reader to our previous work [11].

Originally, the following metadata was available for each publication: ACL Anthology ID, title, abstract, author(s), URL to the PDF, publisher, publication year and month, proceedings title, DOI, and venue. This year, instead of only the URL to the full text being given, the full text of each article is directly available in the dataset. The data is additionally supplemented with 41,000 weakly labelled articles, the overall available splits and their sizes are summarised in Table 1. The task is evaluated using micro, macro, and weighted scores of precision, recall, and F1.

4. Construction of Weakly labelled Dataset

To construct the weakly labelled supplementary dataset, we start by sourcing 80,013 scholarly articles from the ACL Anthology Corpus [41]. To prevent data leakage, we exclude any articles already present in the FoRC4CL dataset. We include only papers published between 2013 and 2022, which ensures alignment with the classes in Taxonomy4CL while maintaining a dataset of sufficient size. We extract

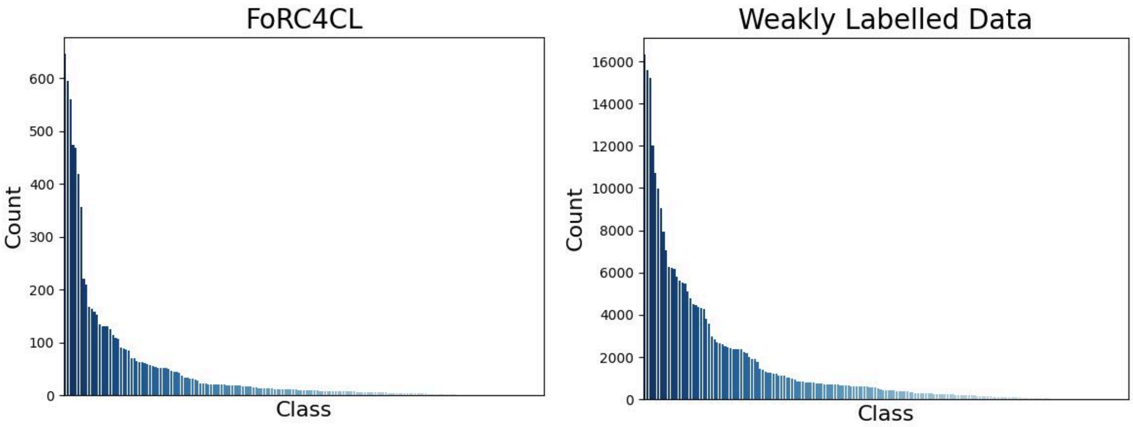

Figure 1: Class distribution of FoRC4CL and the weakly labelled dataset.

metadata fields consistent with those available in FoRC4CL, namely ACL Anthology ID, abstract, URL to the full text, publisher, year and month of publication, book title, author(s), DOI, and venue. When metadata values are missing in either FoRC4CL or the supplementary dataset, we add them using OpenAlex [5] whenever possible. The dataset for the 2025 iteration of the FoRC shared task is publicly available on Zenodo [14].

Table 1
Available dataset splits and sizes for the FoRC 2025 shared task.

Dataset	Size
FoRC4CL train	1000
FoRC4CL validation	225
FoRC4CL test	225
Weakly labelled ACL	41107
Total	42557

Prior work in last year's iteration of the FoRC shared task demonstrated that using full article texts for model training is a promising strategy, with the top-performing teams in both subtasks using this approach [42, 17]. Therefore, we retrieve the full text of each article and integrate it into both datasets. We train a One-vs-Rest Logistic Regression model on the full FoRC4CL dataset including full article texts, and we apply that model to the supplementary dataset to generate weak labels. The reason for choosing this model is based on the experiments presented in Section 4.1. As a postprocessing step, we remove any predicted labels whose corresponding superclass was not also predicted to ensure consistency within the label hierarchy.

Taxonomy4CL comprises 181 classes, of which 46 belong to the first, 109 to the second, and 26 to the third hierarchical level. Not all classes appear as labels in the FoRC4CL dataset, with 7 level 1-, 7 level-2, and 4 level-3 labels remaining unused. The set of labels present in the FoRC4CL dataset is identical to those found in the weakly labelled dataset. Both the FoRC4CL and the weakly labelled datasets exhibit significant class imbalance, with their label distributions closely adhering to Zipf's Law (see Figure 1).

4.1. Selection of Model for Weak Labeling

To select a model to use for weakly labelling the additional ACL publications, we assess the performance of various models by training and evaluating them on the FoRC4CL dataset. We evaluate the models using micro, macro, and weighted F1 scores. We assign the most importance to the macro F1 score. Our experimentation covers classic machine learning (ML) algorithms and Transformer models.

4.1.1. Classic ML Algorithms

Prior research suggests that directly training a transformer model may not be ideal for this task, and that simpler machine learning models can yield superior results [17]. Given this insight, we evaluate the effectiveness of various ensemble learning techniques, including One-vs-Rest Logistic Regression, One-vs-Rest Support Vector Classification (SVC), Random Forest, and XGBoost classifiers, across different hyperparameter configurations. Initial experiments show that One-vs-Rest Logistic Regression achieves slightly higher F1 scores when the input text is stemmed, as shown in Table 2. Thus, we adopt this preprocessing strategy for all subsequent experiments. We tokenize our input using a unigram tf-idf tokenizer. We also explore using bi- and trigram tf-idf tokenization, but observe a decrease in performance (see Table 3) alongside an increase in computational cost.

Table 2
Performance of One vs. Rest Logistic Regression model using various text preprocessing strategies.

Preprocessing	Micro-F1	Macro-F1	Weighted-F1
Removing newlines	0.32	0.03	0.25
Lowercasing	0.32	0.03	0.25
Stemming	0.34	0.03	0.27
Lowercasing + Stemming	0.34	0.03	0.27

Table 3
Performance of different vectorizers using One vs. Rest Logistic Regression model.

Vectorizer	Micro-F1	Macro-F1	Weighted-F1
Tf-Idf Unigram	0.34	0.03	0.27
Tf-Idf Unigram + Bigram	0.25	0.02	0.19
Tf-Idf Unigram + Bigram + Trigram	0.22	0.01	0.17

The results of a subset of our experiments are presented in Table 4, and a comprehensive set of experimental results is provided in Table 9 in the Appendix. All Logistic Regression and SVC models employ the One vs. Rest ensemble technique. We achieve the best performance on the test set using the One vs. Rest Logistic Regression model with a liblinear solver and an L1 penalty, which achieves a weighted F1-score of 0.65. More detailed results of micro-, macro-, and weighted-F1 scores per hierarchical level are shown in Table 5. We notice that Random Forest models achieve the worst performance – this may be because each tree in the forest only sees a portion of the total dataset, making it more likely to miss out on some uncommon labels. Training each tree on the full dataset instead may mitigate this issue and increase performance. Notably, we find that incorporating the full text of the article as part of training data increases classification performance in most cases. The final model performs better on common classes than on sparse ones: A simple linear regression analysis revealed a weak but significant positive relationship between class size in the training set and class performance on the test set ($\beta = 0.0018, SE = 0.0003, p = 1.975e - 07, R^2 = 0.141$).

4.1.2. Transformer Models

We extend our experiments to include various transformer-based models, which are trained, validated, and tested on FoRC4CL. We include BERT [43],[2] DeBERTa [44],[3] SciBERT [45],[4] SPECTER [46],[5] and SciNCL [47].[6] The latter three models are pre-trained on scientific corpora and use a vocabulary derived

[2] https://huggingface.co/google-bert/bert-base-uncased

[3] https://huggingface.co/microsoft/deberta-v3-base

[4] https://huggingface.co/allenai/scibert_scivocab_uncased

[5] https://huggingface.co/allenai/specter

[6] https://huggingface.co/malteos/scincl

Table 4
Performance comparison of various machine learning models on the FoRC task under different hyperparameter configurations. For brevity, the following settings are encoded as: (1) class_weight='balanced', (2) solver='liblinear', (3) penalty='l1', (4) kernel='linear', (5) kernel='poly', (6) kernel='sigmoid', and (7) max_depth=8. If no hyperparameter configurations are noted, the model uses the default implementation settings of Sklearn.

Model	Parameters	Fulltext	Micro-F1	Macro-F1	Weighted-F1
Logistic Regression		✗	0.34	0.03	0.27
	1	✗	**0.64**	0.26	0.62
	1	✓	0.63	0.30	0.63
	1, 2, 3	✗	0.60	0.29	0.63
	1, 2, 3	✓	0.62	**0.33**	**0.65**
SVC		✗	0.39	0.05	0.32
	1	✗	0.51	0.10	0.44
	1, 4	✗	0.61	0.21	0.58
	1, 5	✗	0.15	0.01	0.13
	1, 6	✗	0.62	0.24	0.60
	1, 6	✓	**0.64**	0.30	0.64
Random Forest		✗	0.21	0.02	0.17
One vs. Rest Random Forest		✗	0.31	0.03	0.25
		✓	0.31	0.03	0.25
	1	✗	0.27	0.02	0.21
XGBClassifier		✗	0.55	0.17	0.51
		✓	0.55	0.17	0.51
	7	✗	0.56	0.17	0.51

Table 5
Performance of the best-performing model (One vs. Rest Logistic Regression) on the FoRC4CL test set, by hierarchical level.

Level	Micro-F1	Macro-F1	Weighted-F1
Level 1	0.67	0.45	0.68
Level 2	0.55	0.30	0.58
Level 3	0.61	0.23	0.64
Overall	0.62	0.33	0.65

from scientific texts. The results of these experiments are presented in Table 6.

We test our models on two loss functions – Binary Cross-Entropy with Logits (BCE Loss) and Focal Loss. Focal Loss handles strong class imbalances by down-weighting classes that are easy to predict [48]. All models use AdamW [49] as an optimiser, and a learning rate of 4e-5, which we selected based on preliminary trials on SciBERT and SciNCL. The results of these trials are shown in Figure 2. To prevent overfitting, we implement early stopping with a patience of 3 on the validation loss.

The best-performing transformer model, SPECTER with BCE loss, trained for 30 epochs before early stopping and achieved a weighted F1-score of 0.4. While this performance is not necessarily poor, we manually check the outputs of the model and find that these models predict only a small subset of the possible labels. SPECTER + BCE uses only 30 of the 170 possible classes, predominantly the most common ones. This adherence to common labels is also reflected in the low macro-F1 score compared to weighted-F1. Furthermore, none of the models produced predictions from the second or third levels of the label hierarchy. These findings suggest that these models are not well-suited for generating useful weakly labelled instances for this task.

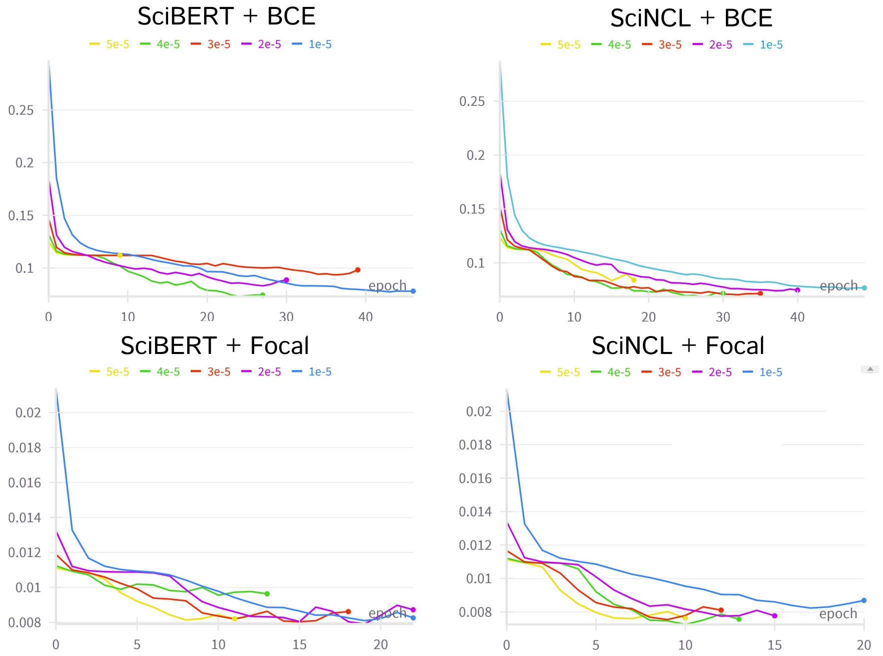

Figure 2: Average validation loss by epoch using SciBERT and SciNCL.

Table 6
Performance comparison of different transformer models on the FoRC task. #Classes denotes the number of classes that appear in the model's predictions.

Loss	Model	Epochs	Micro-F1	Macro-F1	Weighted-F1	#Classes
BCE	BERT	25	0.51	0.11	0.38	24
	DeBERTa	13	0.32	0.02	0.15	4
	SciBERT	12	0.35	0.04	0.23	12
	SPECTER	30	0.51	**0.16**	**0.40**	30
	SciNCL	34	0.51	0.16	0.40	31
Focal	BERT	17	0.474	0.136	0.361	23
	DeBERTa	19	0.35	0.05	0.24	15
	SciBERT	13	0.48	0.13	0.36	28
	SPECTER	10	0.48	0.13	0.36	25
	SciNCL	13	**0.52**	0.15	0.40	27

5. System Descriptions and Results

The following section provides an overview of all systems evaluated in the current and previous edition of the FoRC shared task, including both baseline and submitted models. The results for each system are presented in Table 7. We include models from the 2024 task to contextualise progress, particularly in light of the limited number of submissions in the 2025 edition. While results from 2024 are reported in terms of micro and macro scores, we additionally report weighted scores for systems from 2025.

5.1. Baseline Systems

Due to its simplicity and strong performance in the preliminary experiments for constructing the weakly labelled dataset, we adopt Logistic Regression as the baseline model for this year's iteration. The model is trained with balanced class weights, using the liblinear solver and an L1 regularisation penalty. We train the baseline on both the FoRC4CL training set and the weakly labelled supplementary dataset. For completeness, we train once using full texts and once without. Otherwise, all available metadata are used in training for both settings.

The baseline model for the 2024 iteration of the task was SciNCL fine-tuned on the train split of the FoRC4CL dataset. The input features were article titles and abstracts, and taxonomy labels were multi-hot encoded. Hierarchical information was not used during training and instead was flattened. The baseline was trained for three epochs with BCEWithLogits as the loss function and AdamW as the optimiser. All other hyperparameters were left at their default values as defined in Hugging Face's AutoModelForSequenceClassification class[7].

5.2. Submitted Systems

Submission from the 2025 iteration. The 2025 iteration of the shared task received one submission from the team **KBOGAS**, who explored three approaches to the FoRC classification problem: 1) k-NN over article embeddings, 2) graph neural networks (GNNs), and 3) zero-shot classification using large language models (LLMs). Their best-performing approach combines the first and third strategies in a two-stage pipeline, first retrieving likely candidate labels for a given query via k-NN, then refining the selection using zero-shot LLMs. The k-NN approach alone, which constitutes the baseline, achieves high recall, while the LLM boosts precision. To describe in more detail: the k-NN approach uses Sentence Transformers [50] to embed the titles and abstracts of the training set. Then, the query item is embedded, and the k most similar embeddings by cosine similarity are selected. A subset of their labels are chosen as the final classification using a distance-weighted voting scheme with a voting threshold of 0.3. With increasing values of k, recall increases, converging around 90% at k=20. The team choose a fixed k of 19 because of this. The GNN approach constructs a graph using embedding-based similarity and applies a node classification model, but this does not outperform the baseline. For the LLM component, the team experimented with several prompting strategies, varying the selection of possible labels that are passed to the LLM. Either the full set of Taxonomy4CL labels was included in the prompt, or only a smaller set of labels that were deemed as likely candidates using the aforementioned k-NN algorithm was included. The authors also experiment with the inclusion of hierarchical information. All components were trained and evaluated using only titles and abstracts; the provided weakly labelled data and full texts were not utilised. KBOGAS' best model achieves a micro-F1 of 0.68, macro-F1 of 0.66, and a weighted-F1 of 0.69, surpassing baselines and last year's results in almost all metrics.

Submissions from the 2024 iteration. Two systems were submitted for the 2024 iteration of the shared task: one by **CAU&ZBE** [17] and another by **CUFE**. CAU&ZBE outperform CUFE on all metrics. As CUFE did not provide any system description, we proceed to describe the system submitted by CAU&ZBE. Given the large label set, the imbalanced class distribution, and the hierarchical structure of the taxonomy, CAU&ZBE approach FoRC4CL as an XMLC task. Accordingly, they experimented with models commonly used in the XMLC literature: Parabel [23] and X-Transformer [51]. Parabel is a tree-based approach that deals with label imbalance by recursively dividing the label space to create balanced clusters. Two versions of Parabel were trained: Once using only the FoRC4CL training set, and once additionally using the article full texts. Interestingly, the model trained with full texts underperformed compared to the model trained without, indicating that full texts do not always provide useful signal for the task. For the X-Transformer, fine-tuning proceeds in three phrases: a) clustering the label space, b) assigning the input publication to one of the clusters by relevance, and c) using a ranker to score individual labels within the selected cluster based on their relevance to the input. Two X-Transformer variants were trained, one on the FoRC4CL dataset without full texts, and another, referred to as the weak X-Transformer, trained on the same data augmented with approximately 70,000 additional ACL Anthology articles. The extra 70,000 articles are weakly labelled using a simple tf-idf classifier. Their best results were achieved using the X-Transformer model with the weakly supervised data.

[7] https://huggingface.co/transformers/v3.0.2/model_doc/auto.html#automodelforsequenceclassification

Table 7
Comparison of performance of all baseline and submitted models from the 2024 and 2025 iterations of FoRC. Best scores are bolded, while runners-up are underlined. Models from the 2025 iteration are additionally evaluated with weighted metrics.

Model	P (micro)	R (micro)	F1 (micro)	P (macro)	R (macro)	F1 (macro)	P (W.)	R (W.)	F1 (W.)
Baselines									
with fulltext (2025)	0.46	**0.90**	0.60	0.27	0.59	0.35	0.58	**0.90**	0.67
without fulltext (2025)	0.53	0.89	0.67	0.31	0.59	0.39	0.63	0.89	**0.72**
without fulltext (2024)	0.36	0.33	0.34	0.02	0.05	0.02	–	–	–
2024 Submissions									
CAU&ZBW	0.44	0.76	0.56	0.39	0.56	0.43	–	–	–
CUFE	0.40	0.37	0.39	0.10	0.07	0.06	–	–	–
2025 Submission									
KBOGAS	**0.62**	0.75	**0.68**	**0.65**	**0.74**	**0.66**	**0.68**	0.75	0.69

6. Discussion

The performance of all models is shown in Table 7. KBOGAS achieves the strongest overall performance across nearly all evaluation metrics, with the exception of micro-recall, weighted-recall, and weighted-F1, where the 2025 baselines remain slightly ahead. The fact that the most notable gains made from KBOGAS' approach are found in the macro-averaged scores suggests that its improvements are concentrated on the underrepresented labels, rather than the majority classes. This is additionally reflected in the weighted scores, where KBOGAS only improves in precision over the baseline. In general, KBOGAS performs remarkably well in precision, improving by over 0.25 in macro-precision over the next best model, CAU&ZBW. We find this indicative of the model's ability to make semantically informed predictions, rather than simply predicting more common labels. Overall, KBOGAS' approach yields impressive results, driving progress in the areas of the task that have proved most difficult in the past. By avoiding direct fine-tuning on the data, their approach mitigates the overfitting that typically occurs in such low-resource settings. We expect that with a carefully-curated set of manually labelled instances, this approach could be improved even further.

Despite its simplicity, the Logistic Regression baseline performs surprisingly well on micro and weighted metrics. In comparison, it underperforms in macro scores, which suggests that the baseline handles common labels effectively, but struggles with rare ones. Comparing the results shown here to those in Section 4.1, we observe that the additional training on weakly labelled data does indeed lead to improvements in macro- and weighted-F1, compared to training on clean data alone. Interestingly, the utility of full-text inputs seems to depend on data quality. While adding full-texts to training data improved performance when training on the smaller, clean FoRC4CL dataset, it degraded performance when training with the weakly labelled dataset. This suggests that, in the context of weak supervision, full-texts introduce more noise than they do signal.

One of the conclusions of last year's FoRC shared task was the difficulty posed by the limited availability of high-quality annotated data for training. These challenges are compounded by the large number of labels and the heavy class imbalance, which makes the classification of rare labels particularly difficult. Despite these constraints – and with only a single system submission – the 2025 edition saw substantial progress in the task, especially in improving classification performance on underrepresented labels, all without access to any additional annotated data. To guide future research towards improvement in this task, a qualitative analysis of the predictions made by the KBOGAS model may provide insights towards its weaknesses.

7. Conclusion

In this paper, we presented an overview of the second iteration of the FoRC shared task, which was held under NSLP 2025. This year, the shared task focused on the fine-grained classification of papers in computational linguistics as a hierarchical, multi-label classification problem. The taxonomy used was Taxonomy4CL, which includes 170 (sub-)topics, and the dataset was FoRC4CL, a manually annotated corpus of 1500 papers from the ACL Anthology. This year, we additionally provided 41,000 weakly labelled ACL Anthology papers, which were labelled using a simple Logistic Regression model. One of the main challenges of the task is the highly unbalanced nature of both datasets, making classification of underrepresented labels particularly difficult. While only one team, KBOGAS, participated in the task, their system outperformed last year's winners by a large margin, particularly excelling in precision and in the classification of uncommon labels. Their method combines k-NN clustering over paper embeddings with zero-shot prompting of large language models – the first stage alone achieves high recall, while the second stage improves precision by refining the final prediction. This year's baseline model achieves surprisingly high scores, particularly in recall, and improves in performance through additional training on the weakly labelled dataset. Both datasets are publicly available, and we hope to support future work on FoRC through this contribution.

Acknowledgments

This work was supported by the consortium NFDI for Data Science and Artificial Intelligence (NFDI4DS)[8] as part of the non-profit association National Research Data Infrastructure (NFDI e. V.). The consortium is funded by the Federal Republic of Germany and its states through the German Research Foundation (DFG) project NFDI4DS (no. 460234259).

Declaration on Generative AI

During the preparation of this work, the author(s) used GPT-3.5 and GPT-4 in order to: Grammar and spelling check. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] L. Bornmann, R. Haunschild, R. Mutz, Growth rates of modern science: a latent piecewise growth curve approach to model publication numbers from established and new literature databases, Humanities and Social Sciences Communications 8 (2021) 1–15.

[2] M. Y. Jaradeh, A. Oelen, K. E. Farfar, M. Prinz, J. D'Souza, G. Kismihók, M. Stocker, S. Auer, Open research knowledge graph: next generation infrastructure for semantic scholarly knowledge, in: Proceedings of the 10th international conference on knowledge capture, 2019, pp. 243–246.

[3] A. D. Wade, The semantic scholar academic graph (s2ag), in: Companion Proceedings of the Web Conference 2022, 2022, pp. 739–739.

[4] G. Hendricks, D. Tkaczyk, J. Lin, P. Feeney, Crossref: The sustainable source of community-owned scholarly metadata, Quantitative Science Studies 1 (2020) 414–427.

[5] J. Priem, H. Piwowar, R. Orr, Openalex: A fully-open index of scholarly works, authors, venues, institutions, and concepts, arXiv preprint arXiv:2205.01833 (2022).

[6] J. Eykens, R. Guns, T. C. Engels, Fine-grained classification of social science journal articles using textual data: A comparison of supervised machine learning approaches, Quantitative Science Studies 2 (2021) 89–110.

[8] https://www.nfdi4datascience.de

[7] M. Daradkeh, L. Abualigah, S. Atalla, W. Mansoor, Scientometric analysis and classification of research using convolutional neural networks: A case study in data science and analytics, Electronics 11 (2022) 2066.

[8] F. Hoppe, D. Dessì, H. Sack, Deep learning meets knowledge graphs for scholarly data classification, in: Companion proceedings of the web conference 2021, 2021, pp. 417–421.

[9] X. Chen, H. Xie, X. Tao, Vision, status, and research topics of natural language processing, 2022.

[10] A. A. Salatino, T. Thanapalasingam, A. Mannocci, F. Osborne, E. Motta, The computer science ontology: A large-scale taxonomy of research areas, in: International Semantic Web Conference, 2018, pp. 187–205.

[11] R. A. Ahmad, E. Borisova, G. Rehm, Forc4cl: a fine-grained field of research classification and annotated dataset of nlp articles, in: Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), 2024, pp. 7389–7394.

[12] S. Bird, R. Dale, B. Dorr, B. Gibson, M. Joseph, M.-Y. Kan, D. Lee, B. Powley, D. Radev, Y. F. Tan, The acl anthology reference corpus: A reference dataset for bibliographic research in computational linguistics, in: Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), 2008.

[13] R. Abu Ahmad, E. Borisova, G. Rehm, FoRC@NSLP2024: Overview and insights from the field of research classification shared task, in: International Workshop on Natural Scientific Language Processing and Research Knowledge Graphs, Springer, 2024, pp. 189–204.

[14] M. Francis, R. Abu Ahmad, E. Borisova, G. Rehm, FoRC@NSLP2025 Dataset, 2025. URL: https://doi.org/10.5281/zenodo.14901529. doi:10.5281/zenodo.14901529.

[15] Z. Xu, S. Escalera, A. Pavão, M. Richard, W.-W. Tu, Q. Yao, H. Zhao, I. Guyon, Codabench: Flexible, easy-to-use, and reproducible meta-benchmark platform, Patterns 3 (2022) 100543. URL: https://www.sciencedirect.com/science/article/pii/S2666389922001465. doi:https://doi.org/10.1016/j.patter.2022.100543.

[16] J. Liu, W.-C. Chang, Y. Wu, Y. Yang, Deep learning for extreme multi-label text classification, in: Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval, 2017, pp. 115–124.

[17] L. R. Bashyam, R. Krestel, Advancing automatic subject indexing: combining weak supervision with extreme multi-label classification, in: Proceedings of the 1st International Workshop on Natural Scientific Language Processing and Research Knowledge Graphs (NSLP 2024). Hersonissos, Crete, Greece, volume 27, 2024.

[18] R. Babbar, B. Schölkopf, Dismec: Distributed sparse machines for extreme multi-label classification, in: Proceedings of the tenth ACM international conference on web search and data mining, 2017, pp. 721–729.

[19] I. E. Yen, X. Huang, W. Dai, P. Ravikumar, I. Dhillon, E. Xing, Ppdsparse: A parallel primal-dual sparse method for extreme classification, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017, pp. 545–553.

[20] I. E.-H. Yen, X. Huang, P. Ravikumar, K. Zhong, I. Dhillon, Pd-sparse: A primal and dual sparse approach to extreme multiclass and multilabel classification, in: International conference on machine learning, PMLR, 2016, pp. 3069–3077.

[21] K. Bhatia, H. Jain, P. Kar, M. Varma, P. Jain, Sparse local embeddings for extreme multi-label classification, Advances in neural information processing systems 28 (2015).

[22] Y. Tagami, Annexml: Approximate nearest neighbor search for extreme multi-label classification, in: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, 2017, pp. 455–464.

[23] Y. Prabhu, A. Kag, S. Harsola, R. Agrawal, M. Varma, Parabel: Partitioned label trees for extreme classification with application to dynamic search advertising, in: Proceedings of the 2018 World Wide Web Conference, 2018, pp. 993–1002.

[24] S. Khandagale, H. Xiao, R. Babbar, Bonsai: diverse and shallow trees for extreme multi-label classification, Machine Learning 109 (2020) 2099–2119.

[25] Y. Prabhu, M. Varma, Fastxml: A fast, accurate and stable tree-classifier for extreme multi-label learning, in: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, 2014, pp. 263–272.

[26] T. Jung, J.-K. Kim, S. Lee, D. Kang, Cluster-guided label generation in extreme multi-label classification, arXiv preprint arXiv:2302.09150 (2023).

[27] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, Journal of machine learning research 21 (2020) 1–67.

[28] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, arXiv preprint arXiv:1910.13461 (2019).

[29] R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, D. S. Weld, Knowledge-based weak supervision for information extraction of overlapping relations, in: Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies, 2011, pp. 541–550.

[30] M.-C. Yuen, I. King, K.-S. Leung, A survey of crowdsourcing systems, in: 2011 IEEE third international conference on privacy, security, risk and trust and 2011 IEEE third international conference on social computing, IEEE, 2011, pp. 766–773.

[31] A. Awasthi, S. Ghosh, R. Goyal, S. Sarawagi, Learning from rules generalizing labeled exemplars, arXiv preprint arXiv:2004.06025 (2020).

[32] G. S. Mann, A. McCallum, Generalized expectation criteria for semi-supervised learning with weakly labeled data., Journal of machine learning research 11 (2010).

[33] C. Liang, Y. Yu, H. Jiang, S. Er, R. Wang, T. Zhao, C. Zhang, Bond: Bert-assisted open-domain named entity recognition with distant supervision, in: Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining, 2020, pp. 1054–1064.

[34] Y. Yu, S. Zuo, H. Jiang, W. Ren, T. Zhao, C. Zhang, Fine-tuning pre-trained language model with weak supervision: A contrastive-regularized self-training approach, arXiv preprint arXiv:2010.07835 (2020).

[35] Y. Kim, J. Kim, S. Lee, Clear: Towards robust and generalized parameter-efficient fine-tuning for noisy label learning, arXiv preprint arXiv:2411.00873 (2024).

[36] V. Wheway, Using boosting to detect noisy data, in: Advances in Artificial Intelligence. PRICAI 2000 Workshop Reader: FourWorkshops held at PRICAI 2000 Melbourne, Australia, August 28-September 1, 2000 Revised Papers 6, Springer, 2001, pp. 123–130.

[37] B. Sluban, D. Gamberger, N. Lavrač, Ensemble-based noise detection: noise ranking and visual performance evaluation, Data mining and knowledge discovery 28 (2014) 265–303.

[38] S. J. Delany, N. Segata, B. Mac Namee, Profiling instances in noise reduction, Knowledge-Based Systems 31 (2012) 28–40.

[39] J. Thongkam, G. Xu, Y. Zhang, F. Huang, Support vector machine for outlier detection in breast cancer survivability prediction, in: Advanced Web and Network Technologies, and Applications: APWeb 2008 International Workshops: BIDM, IWHDM, and DeWeb Shenyang, China, April 26-28, 2008. Revised Selected Papers 10, Springer, 2008, pp. 99–109.

[40] D. Zhu, X. Shen, M. Mosbach, A. Stephan, D. Klakow, Weaker than you think: A critical look at weakly supervised learning, arXiv preprint arXiv:2305.17442 (2023).

[41] S. Rohatgi, Acl anthology corpus with full text, Github, 2022. URL: https://github.com/shauryr/ACL-anthology-corpus.

[42] F. Ruosch, R. Vasu, R. Wang, L. Rossetto, A. Bernstein, Single-label multi-modal field of research classification, in: International Workshop on Natural Scientific Language Processing and Research Knowledge Graphs, Springer Nature Switzerland Cham, 2024, pp. 224–233.

[43] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers), 2019, pp. 4171–4186.

[44] P. He, X. Liu, J. Gao, W. Chen, Deberta: Decoding-enhanced bert with disentangled attention,

arXiv preprint arXiv:2006.03654 (2020).

[45] I. Beltagy, K. Lo, A. Cohan, Scibert: A pretrained language model for scientific text, arXiv preprint arXiv:1903.10676 (2019).

[46] A. Cohan, S. Feldman, I. Beltagy, D. Downey, D. S. Weld, Specter: Document-level representation learning using citation-informed transformers, arXiv preprint arXiv:2004.07180 (2020).

[47] M. Ostendorff, N. Rethmeier, I. Augenstein, B. Gipp, G. Rehm, Neighborhood contrastive learning for scientific document representations with citation embeddings, arXiv preprint arXiv:2202.06671 (2022).

[48] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980–2988.

[49] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, arXiv preprint arXiv:1711.05101 (2017).

[50] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, arXiv preprint arXiv:1908.10084 (2019).

[51] J. Zhang, W.-C. Chang, H.-F. Yu, I. Dhillon, Fast multi-resolution transformer fine-tuning for extreme multi-label text classification, Advances in Neural Information Processing Systems 34 (2021) 7267–7280.

A. Model Experiments

During our experiments with different transformer architectures, we also tried various dropout rates. These modifications did not lead to any improvements in performance. The results of these experiments are presented in Table 8.

Table 8
Experimentation using Dropout with transformer models on FoRC task.

Model	Dropout	Loss	Epochs	Batch Size	Macro-F1	Weighted-F1
BERT	–	BCEWithLogits	3	8	0.018	0.156
BERT	0.2	BCEWithLogits	3	8	0.018	0.156
BERT	0.8	BCEWithLogits	3	8	0.020	0.158
BERT	–	FocalLoss	3	8	0.019	0.163
BERT	0.8	FocalLoss	3	8	0.016	0.120
SciNCL	–	BCEWithLogits	3	8	0.018	0.157
SciNCL	0.3	BCEWithLogits	3	8	0.018	0.156
SciNCL	–	FocalLoss	8	8	0.071	**0.317**
SciNCL	0.3	FocalLoss	8	8	0.072	0.311
SciNCL	–	FocalLoss	8	16	**0.073**	0.316

Table 9
Performance comparison of different machine learning models under various hyperparameter settings on the classification task. For brevity, the following settings are encoded as: (1) class_weight=‘balanced’, (2) solver=‘liblinear’, (3) penalty=‘l1’, (4) kernel=‘linear’, (5) kernel=‘poly’, (6) kernel=‘sigmoid’, (7) max_depth=8, (8) probability=True, (9) n_estimators=300, (10) max_features=’sqrt’, (11) objective=’binary:logistic’, and (12) learning_rate=1. If no hyperparameter configurations are noted, the model uses the default implementation settings of Sklearn.

Model	Parameters	Fulltext	Micro-F1	Macro-F1	Weighted-F1
Logistic Regression		✗	0.34	0.03	0.27
	1	✗	**0.64**	0.26	0.62
	1	✓	0.63	0.30	0.63
	1, 2, 3	✗	0.60	0.29	0.63
	1, 2, 3	✓	0.62	**0.33**	**0.65**
SVC		✗	0.39	0.05	0.32
	1	✗	0.51	0.10	0.44
	1, 4	✗	0.61	0.21	0.58
	1, 4	✓	0.63	0.26	0.60
	1, 5	✗	0.15	0.01	0.13
	1, 6	✗	0.62	0.24	0.60
	1, 6	✓	0.64	0.30	0.64
	1, 4, 8	✗	0.61	0.21	0.58
	1, 4, 8	✓	0.55	0.21	0.52
Random Forest		✗	0.21	0.02	0.17
One vs. Rest Random Forest		✗	0.31	0.03	0.25
		✓	0.31	0.03	0.25
	1	✗	0.27	0.02	0.21
	9	✗	0.32	0.03	0.26
	9, 10	✗	0.32	0.03	0.26
XGBClassifier		✗	0.55	0.17	0.51
		✓	0.55	0.17	0.51
	11	✗	0.55	0.17	0.51
	7	✗	0.56	0.17	0.51
	12	✗	0.51	0.17	0.48