DataDumplings at SimpleText 2026: Hierarchical Scientific Domain Classification with Fine-Tuned SciBERT and Taxonomy-Guided Inference*

Sujaudeen Nannai John[1], Srinidhi Rajagopal[1] and Yadushree Venkatesh[1]

[1]*Department of Computer Science and Engineering, Sri Sivasubramaniya Nadar College of Engineering, Chennai 603110, Tamil Nadu, India*

Abstract

This paper presents a two-stage hierarchical classification system for assigning scientific papers to predefined field and discipline categories. We use the title and abstract of each paper as input and fine-tune SciBERT models to perform the classification. The first model predicts a broad field area, while the second predicts a more specific discipline area. Since each discipline belongs to a particular field, we incorporate taxonomy-based constraints during inference to ensure that discipline predictions remain consistent with the predicted field. The proposed system is evaluated using Exact Match, Levenshtein String Distance, and Embedding Distance metrics. This approach combines the strengths of domain-specific language models with simple hierarchical constraints to produce valid and interpretable predictions.

Keywords

scientific document classification, hierarchical classification, SciBERT, taxonomy constraints, transformer models, field prediction, discipline prediction

1. Introduction

The rapid growth of scientific publications has made it increasingly difficult to organize and categorize the areas of research manually. Digital libraries, search engines, and research databases rely heavily on accurate subject labels to help users discover relevant work. As the number of published articles continues to increase, automated classification systems have become an important tool for efficiently managing scientific literature.

The goal here, is to automatically classify scientific papers into two levels of a predefined taxonomy. Given a paper's title and abstract, a system must predict both a broad *field area* and a more specific *discipline area*. The task is inherently hierarchical because each discipline belongs to a particular field. As a result, a valid prediction must not only be accurate but also be in accordance with the structure of the taxonomy.

This task presents several challenges. Scientific abstracts often contain specialized terminology, papers from different domains may use similar language, and some disciplines can be closely related, making classification difficult. In addition, treating field and discipline prediction as completely independent tasks can lead to inconsistent outputs, where a discipline is assigned to a field under which it does not belong.

To address these challenges, we use SciBERT, a transformer model specifically pre-trained on scientific literature. We fine-tune separate SciBERT classifiers for field and discipline prediction using the title

CLEF 2026: Conference and Labs of the Evaluation Forum, September 21–24, 2026, Jena, Germany
sujaudeenn@ssn.edu.in (S. N. John); srinidhi2310238@ssn.edu.in (S. Rajagopal); yadushree2310494@ssn.edu.in (Y. Venkatesh)

and abstract of each paper. During inference, we apply a taxonomy based constraint that restricts discipline predictions to those that are valid under that particular field. This simple strategy makes sure that the predictions are structurally valid.

The remainder of this paper is organised as follows. Section 2 discusses previous work relevant to scientific document classification. Section 3 describes the dataset and taxonomy. Section 4 presents our proposed system, while Sections 5 and 6 discuss the experimental setup and results. Finally, Section 7 concludes the paper.

2. Related Work

Document classification has traditionally been approached using machine learning methods such as Naive Bayes, Support Vector Machines, and Random Forests trained on handcrafted features and bag-of-words representations. While these methods achieved reasonable performance, they often struggled to capture the semantic meaning and context present in text.

The introduction of transformer based language models significantly improved text classification performance. BERT demonstrated that when pre training is done on a large scale and it is followed by fine tuning which is specific to the task, it can achieve state of the art results across a wide range of NLP tasks. However, general purpose language models are trained primarily on generic text and may not fully capture the specialized vocabulary used in scientific documents.

To address this limitation, Beltagy et al. introduced SciBERT, a BERT-based model pre-trained on a large collection of scientific publications. Because SciBERT is exposed to scientific terminology during pre training, it has been shown to outperform other language models which are general purpose on several scientific NLP tasks, including classification and information extraction. This makes it a natural choice for scientific document classification.

Hierarchical text classification has also been widely studied. Existing approaches range from training separate classifiers for different levels of the hierarchy to building models that jointly predict labels while explicitly modelling hierarchical relationships. In this work, we adopt a simple yet effective approach: independent classifiers are trained for field and discipline prediction, and taxonomy constraints are applied during inference to ensure consistency between the two predictions.

3. Task and Data

3.1. Task Definition

CLEF SimpleText 2026 Task 3 focuses on hierarchical classification of scientific papers. Each instance consists of a title and an abstract, and the objective is to predict two labels:

- Field Area – the broad research domain of the paper.
- Discipline Area – a more specific sub domain within the predicted field.

The labels are organized in a predefined taxonomy, i.e. only certain disciplines are valid for each field.

3.2. Classification Taxonomy

The classifications of taxonomy containing valid field–discipline pairs are given. This taxonomy defines the hierarchical relationship between the two levels. Once a field is predicted, the set of valid discipline labels is known in advance. We use this information during inference to restrict discipline predictions and prevent invalid combinations of field and disciplines.

3.3. Training and Test Data

The training dataset contains paper titles, abstracts, field labels, and discipline labels. To provide the model with richer contextual information, we combine the title and abstract into a single input sequence separated by the `[SEP]` token. This combined text is then tokenised using the SciBERT tokenizer.

Field and discipline labels are converted into numerical representations using label encoding. The data is divided into training and validation subsets which are in the ratio of 85:15.

The test dataset contains only titles and abstracts. After training, the models are used to generate field and discipline predictions for these unseen instances, which are then formatted into the final submission file.

4. System Architecture

4.1. Text Encoder

The two models have identical pre-trained backbone: `allenai/scibert_scivocab_uncased` which is a 12-layer BERT pre-trained on roughly 3.1 million scientific papers using the Semantic Scholar dataset [1]. The backbone has scientific domain vocabulary of size 30,000 subword pieces. This allows for better coverage of domain-specific terms compared to standard BERT vocabulary.

4.2. Two-Stage Classification

Both *field area* and *discipline area* predictions are trained using two separate sequence classification tasks on top of SciBERT. Each classifier is a simple linear layer which maps the representation from `[CLS]` token to the number of labels available in its corresponding label space. Separate classifiers are used instead of a shared multi-task system because this makes training simpler and prevents negative knowledge transfer among heterogeneous label sets.

Input to both models is a concatenation of the title and abstract text:

$$x = \texttt{[CLS]}\ \text{title}\ \texttt{[SEP]}\ \text{abstract}\ \texttt{[SEP]} \tag{1}$$

Inputs are truncated to 512 tokens and padded as necessary.

4.3. Taxonomy-Constrained Inference

The critical aspect of the architecture used in our approach involves enforcing the taxonomy hierarchy during the inference stage. For any given field logit vector $\mathbf{f} \in \mathbb{R}^{|\mathcal{F}|}$ and discipline logit vector $\mathbf{d} \in \mathbb{R}^{|\mathcal{D}|}$, one would normally perform an independent argmax to obtain the output classification. This approach, however, may result in structural inconsistency because the discipline predicted by the model is not necessarily a sub-category of the predicted field.

Our decoding process is performed under the following constraints:

1. Predict the field: $\hat{f} = \arg\max_i \ \mathbf{f}_i$.
2. Retrieve the set of valid disciplines for $\hat{f}$ from the taxonomy: $\mathcal{D}_{\hat{f}} \subseteq \mathcal{D}$.
3. Construct a masked logit vector $\tilde{\mathbf{d}}$ where $\tilde{\mathbf{d}}_j = \mathbf{d}_j$ if $j \in \mathcal{D}_{\hat{f}}$, and $\tilde{\mathbf{d}}_j = -\infty$ otherwise.
4. Predict the discipline: $\hat{d} = \arg\max_j \ \tilde{\mathbf{d}}_j$.

This masking technique ensures that all output pairs $(\hat{f}, \hat{d})$ are always valid members of the taxonomy. It does not add any additional parameters and has no significant computation time, since it works directly on precomputed logit vectors.

Figure 1 illustrates the end-to-end pipeline.

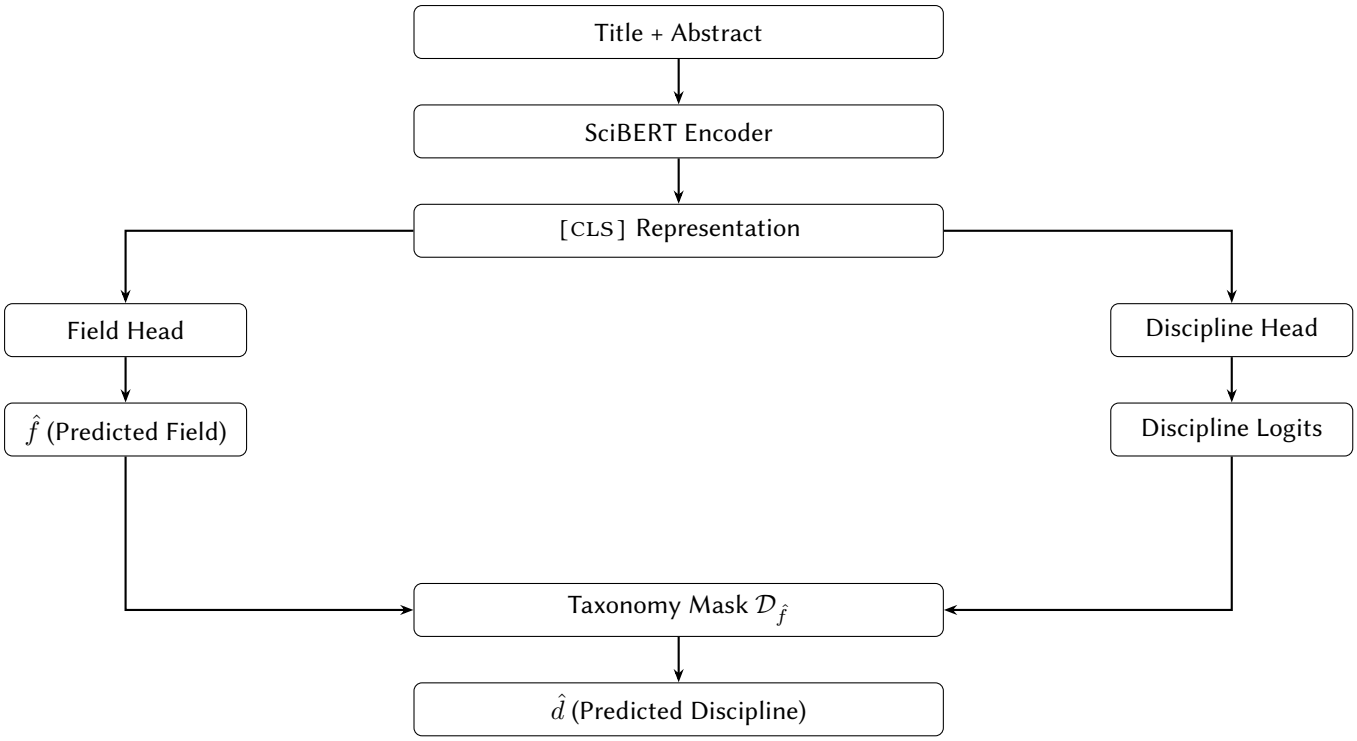

Figure 1: End-to-end pipeline. Both classifiers share a SciBERT backbone. At inference, the predicted field constrains the discipline logits via taxonomy masking before the final discipline prediction is taken.

5. Experimental Setup

5.1. Training Configuration

Both classifier models were trained using the same set of hyperparameters, listed in Table 1.

Table 1
Training configuration for both classifiers.

Component	Setting
Base model	`allenai/scibert_scivocab_uncased`
Max sequence length	512 tokens
Batch size	16
Epochs	4
Learning rate	2×10^{-5}
Warmup ratio	0.1 (linear schedule)
Gradient clipping	1.0
Weight decay	0.01
Optimizer	AdamW
Validation split	15% (stratified)
Checkpoint criterion	Validation accuracy
Hardware	Single NVIDIA T4 GPU (Google Colab)

The model used AdamW optimiser with linear decay on a learning rate and a 10% warm-up phase. Gradient clipping was done at 1.0. Validation accuracy served as the criterion to choose the best model. The optimal model was stored for further use, and re-training was required only in case the data altered.

5.2. Evaluation Metrics

Performance is measured using three complementary metrics, computed separately for field and discipline predictions.

Exact Match (EM) is the strictest metric, requiring the predicted label to exactly equal the gold label after case-insensitive whitespace normalisation.

String Distance (SD) measures normalised character-level similarity using the Levenshtein edit distance:

$$\mathrm{SD}(s_1, s_2) = 1 - \frac{\mathrm{lev}(s_1, s_2)}{\max(|s_1|, |s_2|)} \tag{2}$$

where s_1, s_2 are the two strings being compared, $\mathrm{lev}(s_1, s_2)$ is the Levenshtein distance between them, and $\max(|s_1|, |s_2|)$ normalises the score to $[0, 1]$.

Embedding Distance (ED) measures semantic similarity using BERT sentence embeddings. Both the predicted and gold labels are encoded as vector embeddings, and cosine similarity is computed between them:

$$ED = \cos(\mathbf{v}_{\text{model}}, \mathbf{v}_{\text{actual}}) = \frac{\mathbf{v}_{\text{model}} \cdot \mathbf{v}_{\text{actual}}}{\|\mathbf{v}_{\text{model}}\| \, \|\mathbf{v}_{\text{actual}}\|} \tag{3}$$

A prediction is counted as a match if the similarity score S exceeds a threshold:

$$\text{Match} = \begin{cases} 1, & \text{if } S > 0.7 \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

6. Results and Discussion

6.1. Main Results

Table 2
Validation-set results for field and discipline classification. EM = exact match; SD = string distance; ED = embedding distance.

Setting	Discipline Area			Field Area		
	EM	SD	ED	EM	SD	ED
SciBERT + taxonomy masking	0.5983	0.6909	0.7836	0.8448	0.9080	0.9242

6.2. Effect of Taxonomy-Constrained Inference

The taxonomy-based masking operation (Section 4.3) is specifically devised to address structurally incorrect predictions. Without this constraint, the discipline classifier may assign a discipline to a field under which it does not exist in the taxonomy. The masking step eliminates all such cases by construction, since only valid discipline candidates are considered during prediction. As observed on the validation data, the constrained approach produces improvements across EM, SD, and ED metrics, since all predictions are drawn from a structurally correct candidate set.

6.3. Error Analysis

Field Classifier Confusions Confusion between adjacent fields with similar terminologies, such as Computer Science vs. Mathematics and Biology vs. Medicine, is the most frequent type of errors observed in the field classifier. These field confusions propagate straight to the discipline classifier, since the taxonomy mask is generated based on the (potentially wrong) field classification by the

field classifier. As future directions, it might be interesting to investigate alternative methods for the construction of the taxonomy masks using a soft approach based on top-k predictions for each field.

Another less frequent cause of errors is associated with inter-disciplinary papers. Even when papers belong to one field according to gold-standard annotations and are classified as part of another equally likely field, this will result in errors for all three scores.

7. Conclusion

The proposed system introduces a hierarchical scientific document classification pipeline that fine-tunes SciBERT for classifying documents into fields and disciplines and enforces taxonomy consistency during inference by performing logit masking. The taxonomy-enforced decoding mechanism is computationally trivial, parameter-free, and guarantees that every prediction produced is structurally valid according to the taxonomy. Experiments on the CLEF SimpleText 2026 Task 3 dataset demonstrate that the proposed approach produces coherent hierarchical predictions with strong performance across EM, SD, and ED metrics.

Future research could explore soft top-k field constraints for handling interdisciplinary papers, as well as incorporating citation context or section-level features to further improve classification accuracy.

Acknowledgments

We thank the CLEF 2026 SimpleText Track organizers for their effort in coordinating the shared task and for their valuable feedback during the review process.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools in the preparation of this work.

References

[1] I. Beltagy, K. Lo, A. Cohan, SciBERT: A pretrained language model for scientific text, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2019, pp. 3615–3620. URL: https://aclanthology.org/D19-1371.

[2] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of NAACL-HLT 2019, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423.

[3] T. Joachims, Text categorization with support vector machines: Learning with many relevant features, in: Proceedings of the European Conference on Machine Learning (ECML), Springer, 1998, pp. 137–142.

[4] C. N. Silla Jr., A. A. Freitas, A survey of hierarchical classification across different application domains, Data Mining and Knowledge Discovery 22 (2011) 31–72.

[5] L. Ermakova, P. Bellot, J. Biryukov, A. Nardini, F. Pitel, G. SanJuan, E. Seo, I. Ovchinnikova, Overview of the CLEF 2022 SimpleText lab: Automatic simplification of scientific texts, in: Proceedings of the 13th International Conference of the CLEF Association (CLEF 2022), 2022.