Sree Krishna S at PestCLEF 2026 : Knowledge Graph Extraction from Plant Pest Documents Using BioBERT-based NER and Marker-based Relation Classification*

Notebook for the PestClefLab at CLEF 2026

Sree Krishna S[1], Varghese K James[1], Sujith M[1] and Prabavathy Balasundaram[1]

[1]Sri Sivasubramaniya Nadar College of Engineering, Chennai, India

Abstract

We describe the Sree Krishna S system submitted to the PestCLEF 2026 shared task, which frames Knowledge Graph (KG) extraction from plant pest web documents as a two-stage pipeline: Named Entity Recognition (NER) followed by relation classification. Our approach fine-tunes BioBERT for token-level entity tagging across seven domain-specific entity types, exploiting its pre-training on biomedical literature to handle the dense scientific nomenclature of the EPOP corpus. A marker-based BioBERT relation classifier then inserts special boundary tokens around candidate entity spans, allowing the encoder to produce context-aware representations anchored to each entity's position in the text rather than relying on generic sentence-level pooling. Schema-based filtering further constrains candidate pairs to biologically valid entity-type combinations, eliminating implausible relations without any model computation. We describe a system that trains on gold annotations from the EPOP corpus using this two-stage pipeline. Our final system achieves a macro-averaged F1 of 0.1403 on the development set and 0.1456 on the private test set, placing 10th on the final competition leaderboard. The NER component converges to an F1 of 0.622, demonstrating the viability of the BioBERT fine-tuning approach for this domain.

Keywords

knowledge graph extraction, named entity recognition, relation extraction, BioBERT, plant pest monitoring, PestCLEF 2026

1. Introduction

Automated extraction of structured ecological knowledge from natural language documents is a challenge for plant health surveillance. Monitoring the spread of crop pests, disease outbreaks and vector distributions across different parts of the world requires large volumes of heterogeneous web documents to be processed. It is not feasible for humans to manually process such large volumes of text. PestCLEF 2026 [2] is a shared task in the LifeCLEF [1] evaluation lab at CLEF 2026 that addresses this challenge directly.

Our system, submitted under the team identifier Sree Krishna S, adopts a two-stage pipeline architecture. In the first stage, a BioBERT token classifier identifies entity spans and their types. In the second stage, a marker-based BioBERT relation classifier predicts the predicate for each candidate entity pair produced by stage 1, based on the entity-type combinations defined in the task schema.

Our final submitted system achieves a macro-averaged F1 of 0.1456 on the private test set (approximately 82% of test documents), placing 10th on the final PestCLEF 2026 competition leaderboard.

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany

*Submitted as a working note to the PestCLEF 2026 shared task at LifeCLEF, CLEF 2026. Code and notebooks are publicly available at https://github.com/SSK166/PestClefSK2026.

sreekrishnassk166@gmail.com (S. K. S); varghesekj18@gmail.com (V. K. James); Sujithmaris@gmail.com (S. M); prabavathyb@ssn.edu.in (P. Balasundaram)

https://github.com/SSK166 (S. K. S); https://github.com/vicfic18 (V. K. James); https://github.com/sujith0613 (S. M); https://www.ssn.edu.in/computer-science-and-engineering-department/faculty/dr-b-prabavathy-associate-professor/ (P. Balasundaram)

0009-0000-2362-165X (S. K. S); 0009-0002-9977-768X (V. K. James); 0009-0001-0671-6195 (S. M); 0000-0003-3903-0410 (P. Balasundaram)

The remainder of this paper is organised as follows. Section 2 reviews related work. Section 3 describes the task and dataset. Section 4 presents the final system architecture. Section 5 describes experimental settings. Section 6 reports results and analysis. Section 7 describes how to reproduce our results. Section 8 concludes with directions for future work.

2. Related Work

2.1. Biomedical Named Entity Recognition

Pre-trained language models have substantially advanced the state of the art in biomedical NER. Lee et al. introduced BioBERT [4], a domain-specific BERT model pre-trained on PubMed abstracts and PubMed Central full texts. BioBERT demonstrated consistent improvements over general-domain BERT on three biomedical benchmarks: NER, relation extraction, and question answering, with the NER improvement of 0.62% F1 attributed to the model's exposure to biomedical terminology and Latin species names during pre-training. We used BioBERT for the NER stage of the pipeline as the EPOP corpus contains dense scientific nomenclature for plant pathogens and host species.

2.2. Relation Extraction with Entity Markers

A key design choice in our system is the use of entity boundary markers for relation classification. This technique was introduced by Baldini Soares et al. [5], who proposed inserting special tokens at the boundaries of each entity span so that the BERT encoder can attend directly to the entity positions rather than pooling a generic sentence-level representation. Their experiments on FewRel, SemEval 2010 Task 8, and TACRED demonstrated that the entity-start representation (the hidden state at the opening marker) significantly outperforms CLS-based pooling for relation classification. While the original method uses only the subject-start hidden state, our implementation concatenates the hidden states at both the subject-start and object-start markers to represent both entities and we further append three numbers to this combined representation: the normalised token distance between the two markers, and the normalised absolute positions of each marker in the sequence, providing the classifier with positional context about how far apart the two entities appear in the document.

Pipeline-based relation extraction — where NER and relation classification are performed sequentially as separate models — is a paradigm reviewed in depth by Zhao et al. [6]. While joint models can reduce error propagation from the NER stage, pipeline approaches remain competitive when annotated data is limited, as independent training allows each component to be tuned separately. We opted for the pipeline approach as the EPOP training set is small (110 documents).

2.3. Agricultural Knowledge Graph Construction

Knowledge graph construction from agricultural texts has received growing attention [7, 8]. Most relevant to our work, Yao et al. [9] found that even the state of the art LLMs struggle with the relational structure of the EPOP dataset — a challenge addressed by our pipeline.

3. Task and Dataset

3.1. Task Definition

PestCLEF 2026 requires participants to extract a set of KG triples representing ecological interactions involving plant pests for each input document. Each triple consists of a subject entity, a predicate, and an object entity. The evaluation metric is macro-averaged F1 over all documents in the test set, where per-document F1 is computed as the harmonic mean of precision and recall over the set of predicted triples matched against the gold KG.

3.2. The EPOP Corpus

The EPOP corpus [3] comprises 247 web documents focused on 20 monitored pest species, split into training (110 documents), development (55 documents), and test (82 documents) sets. Documents vary considerably in length and genre. Each document is annotated with named entities, relations between entities, and a document-level KG.

3.3. Entity and Relation Schema

The task defines seven entity types: *Pest*, *Plant*, *Disease*, *Vector*, *Dissemination_pathway*, *Location*, and *Date*. Seven relation predicates are defined (excluding *NO_RELATION*): *Affects*, *Causes*, *Dispersed_by*, *Found_on*, *Located_in*, *Occurs_on*, and *Transmits*. Not all subject–object type pairs are valid for each predicate; the schema defines 18 valid (subject type, object type) combinations. The implementation was refined to ensure all the 18 pairs are present, including the (Pest, found_on, dissemination_pathway) relation. For example, only a *Vector* can be the subject of a *Transmits* relation, and only a *Disease* or *Pest* can be its object. This correction was made in response to reviewer feedback.

4. Final System Architecture

Our final system consists of three components: a BioBERT NER model, a marker-based BioBERT relation classifier, and a KG assembly module. The thresholds $\tau_{ent} = 0.45$ and $\tau_{rel} = 0.35$ were established through an iterative empirical calibration on the development set, where we manually adjusted values to maximize the macro-averaged F1 metric. $\tau_{ent} = 0.45$ was chosen to improve the precision of initial entity spans and prevent the propagation of noisy candidates, while $\tau_{rel} = 0.35$ was calibrated to balance the precision-recall trade-off during graph assembly.

Earlier iterations of this pipeline contained data-loading errors that prevented meaningful learning from gold annotations; we omit a detailed comparison of these intermediate versions here, as the errors reflected implementation bugs rather than substantive modeling choices, and instead report only the corrected final system.For the same reason, we do not tabulate development-set F1 for the earlier versions: because V1 and V2 did not train on gold annotations at all, their scores would reflect the severity of the respective data-loading bug rather than a meaningful modeling comparison

4.1. Stage 1: Named Entity Recognition

Model. We fine-tune `dmis-lab/biobert-base-cased-v1.2` for token classification using the Hugging Face `transformers` library. The model produces a distribution over 15 BIO entity labels (B- and I- prefixes for each of the seven entity types, plus O) for every input token. The token-level classification head is a linear layer initialised randomly and trained from scratch on top of the frozen-then-unfrozen BioBERT encoder.

Input preparation. Raw document text is pre-processed by collapsing repeated whitespace and removing very short lines. Gold entity annotations, provided as character offsets, are converted to word-level BIO labels by matching each entity span to whitespace-tokenised words. Sub-word tokenisation by BioBERT's WordPiece tokeniser is handled by aligning labels to the first sub-word of each word and masking continuation tokens with `-100` during loss computation.

Long document handling. Because EPOP documents frequently exceed BioBERT's 512-token context limit, we apply sliding-window inference with a window size of 512 tokens and a stride of 462 tokens. Window predictions are stitched back to the full token sequence by position, with later windows overwriting earlier ones in the overlap region.

Class imbalance. Entity tokens constitute approximately 12.4% of training tokens; the remaining 87.6% are labelled O. To counteract this imbalance, we use weighted cross-entropy loss, computing inverse-frequency weights per class and capping them at 5× to avoid extreme over-correction. Training uses the `WeightedTrainer` class, a thin subclass of the Hugging Face `Trainer` that overrides the loss computation.

Inference and filtering. During inference, BioBERT produces a probability distribution over BIO tag set for each token. The entity-level confidence score is computed via a minimum-score strategy. The confidence score of a span is defined by the minimum softmax probability of the tokens constituting the span. Spans with a confidence score below $\tau_{ent} = 0.45$, spans shorter than 3 characters and pure digit tokens are discarded. A curated blocklist of generic geographic terms (*global*, *world*, *country*, *international*, *area*, *province*, *recently*) is used to suppress common false positives from the Location and Date entity classes. Duplicate spans of the same type are deduplicated, and the entity list is capped at 50 entities per document, retained in the order they are decoded from the token sequence, so the retained set favors entities that appear earlier in the text. This limits the quadratic growth of candidate pairs in the subsequent stage. From this, all the valid candidate pairs are generated and the invalid entity pairs are filtered out. The NER model's performance is evaluated by the entity-level micro-averaged F1 score, calculated using the standard seqeval implementation.

4.2. Stage 2: Relation Classification

Model. The relation classifier uses a custom `RelationClassifier` module built on top of `dmis-lab/biobert-base-cased-v1.2`. For each candidate entity pair (subject s, object o), the model receives a modified version of the document text with four special boundary tokens inserted:

```
[SUBJ_START] <subject span> [SUBJ_END] ... [OBJ_START] <object span>
    [OBJ_END]
```

This marker-based encoding, introduced by Baldini Soares et al. [5], allows the BioBERT encoder to attend directly to the entity boundaries. The hidden states at the `[SUBJ_START]` and `[OBJ_START]` positions are extracted from the final encoder layer and concatenated to form a 1536-dimensional representation. Three scalar distance features are appended: the normalised absolute token distance between the two marker positions, and the normalised absolute positions of each marker in the sequence. The resulting 1539-dimensional vector is passed through the classifier head:

$$\text{Dropout}(0.15) \to \text{Linear}(1539 \to 768) \to \text{GELU} \to \text{LayerNorm}(768) \to \text{Dropout}(0.15) \to \text{Linear}(768 \to 8)$$

The output is an 8-dimensional logit vector (one per predicate including *NO_RELATION*). Softmax probabilities are computed at inference; the predicted label is the argmax and its probability is the confidence score.

Schema-based candidate filtering. Before encoding any pair, the system checks whether the (subject type, object type) combination appears in the `VALID_PAIRS` schema. Pairs not in the schema are skipped entirely. This constraint eliminates biologically implausible candidates without consuming model compute, and substantially reduces the false positive rate.

Training data construction. Positive examples are extracted directly from the gold relation annotations in the training and development splits. Hard negative examples are constructed by enumerating schema-valid entity pair permutations that do not appear as positives within the same document and whose entity mentions are within 300 characters of each other. Hard negatives are sampled at a ratio of 3:1 negative-to-positive per document. The final training set contains 7,758 examples: 2,874 positive and 4,884 negative.

Class weighting. Predicate frequency varies substantially across the training set. We compute log-normalised inverse-frequency weights (`torch.log1p`) and clip them to the range [0.5, 10.0]. The resulting weights give the rare predicates (*Causes*, *Transmits*, *Affects*) weights near 8.79 and the majority class (*NO_RELATION*) a weight near 0.95.

4.3. Stage 3: KG Assembly

After relation prediction, candidate edges with a confidence score below $\tau_{rel} = 0.35$ are discarded. Remaining edges are passed to build_kg(). Entity surface forms are mapped to canonical IDs using a dictionary built from gold annotations during training/development data loading, which pairs each surface string with its gold canonical ID. At inference, a predicted entity's surface text is looked up in this dictionary; if a match exists, the gold canonical ID is used. For unseen text — which is expected on the test set — the system falls back to using the raw surface string itself as the canonical ID. Edge keys are normalised the same way, and a seen-set is used to remove duplicate edges. The final KG for each document is serialised as a JSON array of {subject, predicate, object} triples and written to the submission CSV.

Test-set fallback. During test-set inference, entity spans are generated using the BioBERT NER model, with spans filtered by a confidence threshold of $\tau_{ent} = 0.45$. To avoid sparse output in documents where the NER model fails to identify at least two entities, a rule-based silver labeller is invoked as a fallback. The labeller uses curated term dictionaries matched against the text via case-insensitive regular expressions, with longest-match prioritization to reduce noise.

5. Experimental Setup

5.1. Training Configuration

NER model. We train for 8 epochs with batch size 16, learning rate 3×10^{-5}, weight decay 0.01, and a linear warmup over the first 10% of total training steps. Mixed-precision training (FP16) is used on GPU. The best checkpoint is selected by development-set F1 at the end of each epoch.

Relation model. We train for 5 epochs with batch size 16, learning rate 2×10^{-5}, weight decay 0.01, and gradient clipping at 1.0. A linear learning rate schedule with 10% warmup is applied. The best checkpoint across epochs is saved based on macro-averaged F1 over positive predicates on a held-out 15% of the training examples (rounded to 1,164 validation examples, with 6,594 used for training).

5.2. Hardware

All experiments were conducted on Google Colab with an NVIDIA T4 GPU (16 GB VRAM). BioBERT was loaded from the Hugging Face model hub (`dmis-lab/biobert-base-cased-v1.2`). Approximate wall-clock times were 55–70 minutes for NER training (8 epochs) and 30–40 minutes for relation model training (5 epochs).

6. Results and Analysis

6.1. NER Performance

Table 1 shows the NER model's entity-level micro-averaged F1 across epochs. The model improves steadily to an F1 of 0.622 at the final epoch, with precision 0.544 and recall 0.726 at the best checkpoint. The entity token density of 12.4% (6,568 entity tokens out of 52,947 total) indicates that the EPOP documents contain a reasonably high concentration of entities, validating the use of weighted loss.

Table 1
NER development-set performance per epoch.

Epoch	Precision	Recall	F1
1	0.000	0.000	0.000
2	0.438	0.092	0.153
3	0.438	0.460	0.449
4	0.473	0.635	0.542
5	0.470	0.671	0.553
6	0.513	0.697	0.591
7	0.544	0.716	0.619
8	0.544	0.726	**0.622**

Table 2
Summary of development-set KG evaluation.

Metric	Value
Total documents	55
Documents with F1 = 0.0	46
Documents with partial F1 (>0, <1)	5
Documents with F1 = 1.0	4
Total predicted edges	125
Average edges per document	2.27
Macro-averaged F1	0.1403

6.2. Relation Model Performance

The relation classifier achieves a best macro-averaged F1 of 0.4145 on the validation split (epoch 4 of 5). Training loss decreases consistently from 1.0525 to 0.3767, indicating that the model is learning well. The class weights assigned to rare predicates (up to 8.79×) successfully prevent the model from predicting NO_RELATION for all inputs during training.

6.3. Development Set KG Evaluation

Table 2 summarises the pipeline's performance on the 55 development documents. The system produces 125 total edges across 55 documents (average 2.27 edges per document), achieving a macro-averaged F1 of **0.1403**.

6.4. Test Set Result

For the PestCLEF 2026 task, we submitted a single system which was selected for official judging. Table 3 summarizes the performance of this submission on both our local development split and the official private test set. Our system achieved a macro-averaged F1 score of 0.1456 on the private test set, corresponding to a 10th-place rank on the final competition leaderboard.[1]

Table 3
Official evaluation results for the Sree Krishna S submission.

Phase	**Team Identifier**	**Metric**	**Score**	**Rank**	**Setting**
Development	Sree Krishna S	Macro-F1	0.1403	-	Local Split
Private Test	Sree Krishna S	Macro-F1	0.1456	10th	Official Leaderboard

[1] https://www.kaggle.com/competitions/pest-clef-2026/leaderboard

The close agreement between our local development F1 (0.1403) and the official private test F1 (0.1456) suggests that our pipeline generalizes consistently across data splits and avoids significant overfitting to the development set.

6.5. Analysis and Discussion

The schema-based filtering removes invalid entity-type pairs before any model computation, reducing false positives. The marker-based encoding directs the BioBERT encoder to focus on entity boundaries rather than the whole sentence, which matters in EPOP documents where multiple entities of the same type appear close together. The weighted loss in both models prevents them from always predicting the most common class, as seen in the NER recall of 0.726 and the relation model F1 of 0.4145.

The gap between per-component performance and end-to-end KG F1 is characteristic of pipeline systems where errors propagate between stages: entity spans not recovered by NER cannot contribute to any relations, and the final KG F1 is bounded above by the NER recall. The consistent performance across splits suggests that fixing the data-loading errors was an important step toward a working pipeline, though we did not test this with an ablation study.

The marker-based BioBERT approach offers several advantages for this domain. First, it requires no additional feature engineering beyond inserting four special tokens per candidate pair. Second, the valid entity-type pairs are defined in a simple lookup table that domain experts can easily inspect and modify. Third, independent training of NER and RE components allows each to be improved separately in future work without retraining the entire system.

We also found that many short documents (fewer than 200 words) produced no entities above the confidence threshold, resulting in an empty KG. Adjusting the confidence threshold for short documents is a key direction for future work.

We do not report per-entity-type or per-predicate breakdowns, nor relation-extraction performance before versus after schema-based candidate filtering, in this version of the notes. Both require re-running evaluation over the full pipeline and are left as immediate follow-up analysis rather than a rerun of training.

7. Reproducibility

To facilitate reproduction of our results, all code and notebooks are provided in the public GitHub repository mentioned above. The final corrected system, as described in this paper, is implemented in PestClefSubmissionSKVSH.ipynb. Setup instructions and data configuration details are provided in the README. Earlier iteration notebooks are retained in the repository for transparency but are not required for reproduction

8. Conclusion

We presented the Sree Krishna S system for PestCLEF 2026, a two-stage pipeline combining BioBERT-based NER with marker-based relation classification for Knowledge Graph extraction from plant pest documents.

The final system achieves a NER token-level F1 of 0.622, a relation classification validation F1 of 0.4145, a development-set KG macro F1 of 0.1403, and a private test-set macro F1 of 0.1456, placing 10th on the final competition leaderboard. The close match between development and test scores demonstrates that the pipeline generalises well across document splits.

Future work will focus on three directions: (i) lowering and cross-validating the relation confidence threshold, particularly for short documents that currently produce empty KGs; (ii) replacing sentence-level relation training with full-document or cross-sentence encoding to better capture the long-range ecological relations characteristic of EPOP documents; and (iii) incorporating entity normalisation to reduce mismatches between different surface forms of the same entity during KG evaluation.

Acknowledgments

The authors thank the PestCLEF 2026 organisers at INRAE Paris-Saclay for providing the EPOP corpus and the evaluation infrastructure.

Declaration on Generative AI

During the preparation of this work, the author(s) used Claude (Anthropic): for grammar checking, code documentation, and preparation of this working note. After using this tool, the author(s) reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] Lukas Picek, Lukáš Adam, Stefan Kahl, Robert Bossy, Laura Chrobak, Hervé Goëau, Kostas Papafitsoros, Holger Klinck, Willem-Pier Vellinga, Robert Planqué, Tom Denton, Kevin Barnard, Claire Nédellec, Louise Deléger, Marine Courtin, Giulio Martellucci, Ilyass Moummad, Fabrice Vinatier, Pierre Bonnet, and Alexis Joly. Overview of LifeCLEF 2026: AI Challenges for Biodiversity Understanding and Ecosystem Management. *International Conference of the Cross-Language Evaluation Forum for European Languages (CLEF)*, Springer, 2026.

[2] Robert Bossy, Marine Courtin, Louise Deléger, Xinzhi Yao, and Claire Nédellec. Overview of PestCLEF 2026: Knowledge Graph Extraction from Plant Health Literature. *Working Notes of CLEF 2026 - Conference and Labs of the Evaluation Forum*, 2026.

[3] Claire Nédellec, Marine Courtin, Xinzhi Yao, Marie Grosdidier, Isabelle Pieretti, Sandy Duperier, and Robert Bossy. EPOP: A benchmark corpus for Assessing NLP Models on Structured Information Extraction in Plant Health. *Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026)*, 11(16):1331–1340, 2026.

[4] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.

[5] Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2895–2905, Florence, Italy, July 2019. ACL. doi:10.18653/v1/P19-1279.

[6] Weinan Zhao, Xin Ye, Meishan Yang, Zhenjun Lei, Zhengdao Zhang, and Rong Zhao. A comprehensive survey on deep learning for relation extraction: Recent advances and new frontiers. *ACM Computing Surveys*, 2024. doi:10.1145/3674501.

[7] Kun Wang, Yifei Miao, Xin Wang, Yu Li, Fang Li, and Hao Song. Research on the construction of a knowledge graph for tomato leaf pests and diseases based on the named entity recognition model. *Frontiers in Plant Science*, 15:1482275, 2024. doi:10.3389/fpls.2024.1482275.

[8] Hao Wu, Nengfu Xie, Xiaoli Wang, Jingchao Fan, Yonglei Li, and Zhibo Meng. Crop GraphRAG: pest and disease knowledge base Q&A system for sustainable crop protection. *Frontiers in Plant Science*, 16:1696872, 2025.

[9] Xinzhi Yao, Claire Nédellec, Jingbo Xia, and Robert Bossy. Consistency–accuracy correlation in hard-prompted LLMs for entity and relation extraction: empirical findings from plant-health data. *Genomics & Informatics*, 24(1):3, 2026.