ClimateSense at CheckThat! 2025: Combining Fine-tuned Large Language Models and Conventional Machine Learning Models for Subjectivity and Scientific Web Discourse Analysis

Notebook for the CheckThat! Lab at CLEF 2025

Grégoire Burel[1], Pasquale Lisena[2], Enrico Daga[1], Raphael Troncy[2] and Harith Alani[1]

[1]Knowledge Media Institute, The Open University, Milton Keynes, UK
[2]EURECOM, Sophia Antipolis, France

Abstract

These working notes present the ClimateSense team participation in the CheckThat! 2025 Lab Challenge for tasks *1* and *4a* that investigated: 1) the subjectivity of news article sentences, and; 2) the detection of scientific content in social media posts. Pre-trained Large Language Models (LLMs), conventional Machine Learning (ML) models, sentence encoders, data augmentation, and filtering techniques were leveraged by the ClimateSense team to investigate these tasks. In this paper, we detail the approaches for each task, present the methodology, and report on the performance of each submission. The fine-tuning of pre-trained models shows particularly strong results for *Task 4a*, where we achieved the first rank on the final evaluation leaderboard. This result shows that LLMs can benefit from lightweight traditional classification models when performing classification tasks.

Keywords

Large Language Models, Scientific Web Discourse, Social Media, Subjectivity

1. Introduction

In this work, we present the ClimateSense team results on two tasks of the 2025 CheckThat! Lab Challenge [1, 2]. The 2025 challenge focuses on multiple tasks related to fact-checking such as claim subjectivity (*Task 1*), claim extraction and normalisation (*Task 2*), numerical claims (*Task 3*) and scientific web discourse (*Task 4a* and *4b*). The ClimateSense team focused on *Task 1* and *Task 4a*.

The first task (*Task 1*) [3] aims to develop models to perform a binary classification on sentences from news articles. The goal of the task is to distinguish sentences that express subjective views of the author behind it (SUBJ) from objective views (OBJ). The task consists of three distinct settings:

1. *Monolingual*: Train and test on data in a given language;
2. *Multilingual:* Train and test on data comprising several languages, and;
3. *Zero-shot:* Train on several languages and test on unseen languages.

The second task (*Task 4a*) [4] focuses on identifying multiple aspects of social media posts that relate to scientific discourse. The task consists of creating a multi-label classifier that identifies three scientific aspects in X posts:

1. *Scientific claim detection*: Identify whether a social media post contains a scientific claim;
2. *Scientific reference detection*: Determine whether a social media post contains references to scientific studies or publications;

CLEF 2025 Working Notes, September 9 – 12 September 2025, Madrid, Spain
gregoire.burel@open.ac.uk (G. Burel); pasquale.lisena@eurecom.fr (P. Lisena); enrico.daga@open.ac.uk (E. Daga); raphael.troncy@eurecom.fr (R. Troncy); harith.alani@open.ac.uk (H. Alani)
0000-0003-0029-5219 (G. Burel); 0000-0003-3094-5585 (P. Lisena); 0000-0002-3184-5407 (E. Daga); 0000-0003-0457-1436 (R. Troncy); 0000-0003-2784-349X (H. Alani)

3. *Scientific mentions detection*: Determine whether a social media post mentions scientific entities (e.g., a university, a scientist).

For both tasks, we experiment with fine-tuning Large Language Models (LLMs) and conventional Machine Learning (ML) classification models (e.g., logistic regression). We achieved good results in *Task 4a*, where our approach ranked first on the final evaluation leaderboard. In the next sections, we discuss our methodology and results in more detail for each task. Our approach can be reproduced using the code at https://github.com/climatesense-project/climatesense-checkthat2025.

2. Datasets

The datasets of *Task 1* and *Task 4a* are significantly different given both the different purpose of each task and the content sources used for creating each dataset. *Task 1* dataset is collected from news articles in multiple languages whereas *Task 4a* data is obtained from X posts. *Task 1* is also a binary text classification task whereas *Task 4a* is a multi-label classification task.

2.1. Task 1: Sentence Subjectivity

The dataset used for *Task 1* contains sentences extracted from news articles and spans five different languages: Arabic, Bulgarian, German, English, and Italian. Each language folder contains a train, development (dev) and test dataset, with an additional unlabelled test set provided for the challenge submission. The training dataset contains $6,418$ sentences, the development dataset has $2,829$ sentences, and the test dataset consists of $2,332$ sentences. The language subdivision is detailed in the GitLab README for *Task 1*[1] and is listed in Table 1.

Table 1
Dataset statistics by language and split for *Task 1*.

			Label	
Language	Split	Sentences	OBJ	SUBJ
English	Train	830	532	298
	Dev	462	222	240
	Dev-test	484	362	122
Italian	Train	1613	1231	382
	Dev	667	490	177
	Dev-test	513	377	136
German	Train	800	492	308
	Dev	491	317	174
	Dev-test	337	226	111
Bulgarian	Train	729	406	323
	Dev	467	175	139
	Dev-test	250	143	107
Arabic	Train	2,446	1391	1055
	Dev	742	266	201
	Dev-test	748	425	323

Each entry in the dataset contains three fields designed to enable the classification of the article sentences as either subjective (SUBJ) or objective (OBJ): 1) `sentence_id` provides a unique sentence identifier in the dataset in the UUID format; 2) `sentence` contains the sentence to annotate in plain text, and; 3) `label` indicates if the sentence is *objective* (OBJ) or *subjective* (SUBJ).

[1] *Task1*, https://gitlab.com/checkthat_lab/clef2025-checkthat-lab/-/tree/main/task1.

It is important to note that the dataset labels are not balanced and the subjective aspect of several entries is hard to determine, even for a human. Some sentences are very short (see the first two sentences in Table 2) and their meaning can be ambiguous without having access to the sentence context. For example, in Table 2), the last two sentences have a similar structure and meaning, but while one is marked as OBJ, the other is annotated as SUBJ.

Table 2
Task 1 dataset examples (English).

SID	Sentence	Label
adc7a5b9	We take that for granted.	OBJ
5ddbcceb	But the risk is real.	SUBJ
2d6c2fb0	The city, of course, is an important factor.	OBJ
ca771fd3	The pandemic and politics are two primary factors.	SUBJ

2.2. Task 4a: Scientific Web Discourse Detection

The dataset used for the scientific web discourse task (*Task 4b*) consists of posts on social media labelled in three different categories: 1) The first category identifies whether the text contains a scientific claim (`cat1`) or not; 2) The second category denotes if the social media post contains a reference to a scientific study or publication (`cat2`), and; 3) The last category identifies the mentions of scientific entities (e.g. a university, a scientist) (`cat3`).

Similarly to *Task 1*, the *Task 4a* data is divided into a training and a development dataset. These two data sets were provided during the first phase of the challenge for the development and evaluation of the multi-label classifiers, while a third, unlabelled dataset, was provided later for the final evaluation and ranking of the model. The training data contains $1,366$ entries. However, it contains two duplicated entries. After removing duplicates, the training dataset contains $1,364$ distinct posts. The development dataset contains 137 posts and the final unlabelled evaluation dataset contains 240 posts.

The training and development datasets are both unbalanced. For the training dataset, only 26.2% of the posts contain scientific claims (`cat1`), 18.3% of the posts refer to a scientific study or publication (`cat2`) and 24.9% of the posts mention scientific entities (`cat3`). Overall, we also observe that 61.4% of the data consist of posts that do not contain any scientific claim, references to scientific works and no scientific entities, and 10.9% of the posts have all three categories. Although this suggests that training a pre-classifier for such edge cases may be beneficial, we did not find it beneficial for improving the accuracy of the classification.

Each of the *Task 4a* dataset contains the following fields: 1) `index`: the index for the data sample; 2) `text`: the preprocessed tweet text (user mentions are replaced with @user), and; 3) `labels`: a list of three labels (this field is not provided in the final evaluation data), one for each category (e.g., $[0.0, 0.0, 1.0]$). More information about the task is given in the GitLab README for *Task 4a*.[2]

3. Methodology

Due to the differences in goals and data format between *Task 1* and *Task 4a*, distinct methodologies are employed for each task. This section details the various approaches used to analyse the data and create the classification models for each task. The code of both experiments is made available on the ClimateSense code repository.[3]

[2] *Task 4a*, https://gitlab.com/checkthat_lab/clef2025-checkthat-lab/-/tree/main/task4/subtask_4a.
[3] *Tasks 1* and *4b* code repository, https://github.com/climatesense-project/climatesense-checkthat2025.

3.1. Task 1: Sentence Subjectivity

To identify the most promising classification model, we conducted a first round of experiments on the Italian dataset, training on the Italian development data and testing on the Italian test data. The baseline proposed by the *Task 1* organisers consists of a logistic regression head trained on a Sentence-BERT [5] multilingual representation of the data. This solution already yielded good results, achieving an accuracy of 0.69. Iteratively, we performed experiments by changing:

1. The *representation model*: we experimented with E5 [6] (chosen for its good performance in classification tasks), ModernBERT [7] and CovidTwitterBERT [8], as they are trained on news data;
2. The *classifier*: we experimented with Support Vector Classifier (SVC) and Multi-Layer Perceptron (MLP).

Separately, we performed some experiments with the Large Language Model Zephyr [9]. The prompt consisted of the task instruction (discriminate between subjective and objective sentences), optionally the example of a SUBJ and OBJ sentence, and the request to not give any explanation in output apart from the label corresponding to the classification (SUBJ or OBJ). Our experiment included both zero-shot or one-shot prompting, respectively including or not including one example in the prompt. Nevertheless, Zephyr showed lower performances compared to the baseline (0.54), leading us to decide to put this aside. This may be due to the ambiguity of some dataset entries (see Section 2.1), that can only be acquired during the training phase.

A second round of experiments was carried out by training models on the Italian training set (instead of the development set) and testing on the same Italian test set. The results – reported in Table 3 – demonstrate the superiority of multilingual E5[4] (24 layers, embedding size = 1024) in combination with a 100-layer MLP classifier.

We empirically set the number of training epochs for the classifier to 100. We found that going beyond this number caused a drop in accuracy, likely due to overfitting. This behaviour was consistent across the other languages we experimented with, all of which also reached their peak performance around 100 epochs.

Table 3
Model performance comparison for *Task 1* for the Italian dataset.

Model	Fine Tuning Method	Nb. Epochs	Accuracy
intfloat/multilingual-e5-large-instruct	LogisticRegression	-	0.7934
intfloat/multilingual-e5-large-instruct	SVC	-	0.7602
intfloat/multilingual-e5-large-instruct	MLPClassifier	50	0.8265
intfloat/multilingual-e5-large-instruct	MLPClassifier	100	0.8343
intfloat/multilingual-e5-large-instruct	MLPClassifier	200	0.8265
intfloat/multilingual-e5-large-instruct	MLPClassifier	400	0.8031
answerdotai/ModernBERT-large	MLPClassifier	100	0.7154
answerdotai/ModernBERT-large	MLPClassifier	200	0.7135
digitalepidemiologylab/covid-twitter-bert	MLPClassifier	100	0.7135

The final submission for *Task 1* involved experimenting with the following distinct configurations:

- *Monolingual approach*: each model was trained and tested solely on data from one language;
- *Multilingual configuration*: the models were trained using data collected from all five languages. and;
- *Zero-shot*: the same multilingual model is applied to an unknown language with the text translated into English using Google Translate.

[4]We used the implementation at https://huggingface.co/intfloat/multilingual-e5-large-instruct.

All *Task 1* experiments were conducted on a Mac computer equipped with an Apple M1 processor. The M1 chip integrates an 8-core CPU with 4 performance cores and 4 efficiency cores, along with an 8-core GPU, and a 16-core Neural Engine. The system was configured with 16GB of unified memory and a 512GB SSD for storage.

3.2. Task 4a: Scientific Web Discourse Detection

As mentioned in Section 2.2, the training data contains duplicate entries. Therefore, we removed the duplicates before building the multi-label classifiers. As the main goal of the task is to obtain the best macro-average $F1$ across the labels, we optimised the training based on the provided development dataset. In particular, the development dataset was used for selecting the best-performing models and the number of training epochs.

The baseline provided by the challenge organisers was based on a fine-tuned `DeBERTaV3` model [10] and reported a macro-average $F1$ of 0.8375 ($F1$ for `cat1`: 0.8214, `cat2`: 0.7925 and `cat3`: 0.8986). This result suggests that fine-tuned LLMs perform well. However, using LLMs that have been trained with social media data or scientific content could lead to improved results. We also observe that `cat2` appears to be harder to classify accurately. This is confirmed when looking at the final task leaderboard.[5]

Three main approaches were considered when building the multi-label classifiers:

1. *LLM fine-tuning*: Similarly to the baseline, a pre-trained LLM is fine-tuned on the multi-label task.
2. *Embedding classifier*: We use a conventional ML classifier (e.g. SVM or logistic regression) on top of a sentence encoder without fine-tuning it.
3. *Efficient Few-shot Learning with Sentence Transformers (SetFit)*: We fine-tune sentence encoders using the training data before using a conventional classification head.

We also investigated whether training separate classifiers yielded better results than using a single multi-label classifier, and found that creating a single model with different classification heads for each task provided the best results. We also looked at zero-shot classifiers using generative models (`deepseek-r1:7b`), but the results were lower than the baseline.

Since the training data is unbalanced we considered two main approaches for alleviating the risk of overfitting on the majority classes. First, we used a weighted loss function to improve the classification. Second, we created a paraphraser model that generated new minority-class examples by paraphrasing existing minority examples. This approach only works when creating a separate classifier for each category.

The oversampling model used `deepseek-r1:7b` and Ollama[6] for rewriting the sentences. The prompt used a set of *personas* to generate the new examples based on the following prompt: "*Just answer with the text and nothing else, generate an alternate version of the following tweet as if you were* `P`" where `P` is one of the following *personas*: *friendly, formal, humorous, poetic, sarcastic, dramatic, scientific, mysterious, adventurous, romantic, philosophical, historical, technical, casual, business-like, playful, empathetic, authoritative, inquisitive, optimistic, pessimistic, cynical, realistic, idealistic, whimsical, nostalgic, sophisticated, down-to-earth, witty, charming, enigmatic, intellectual* and *artistic*. The oversampling method was only used with the embedding models, where separate classifiers were trained for each category.

Due to the data and task overlap between *Task 4a* and the research conducted in SciTweets [11], we also investigated the integration of the heuristic methods of SciTweets into the classifiers developed as part of *Task 4a*. We based our code on the code provided in the SciTweets GitHub repository.[7]

All *Task 4a* experiments were conducted on a GPU server with an Intel Xeon Silver 4114 processor with two NVIDIA Tesla P40 with 24GB memory and 269GB RAM. The following sentence encoders and models for the embedding and SetFit models were considered: `all-minilm` and `bge-m3` [12]. These models were selected based on the computing resources available for training

[5] *Task 4a* leaderboard, https://codalab.lisn.upsaclay.fr/competitions/22355.
[6] Ollama, https://ollama.com/.
[7] SciTweets GitHub heuristics, https://github.com/AI-4-Sci/SciTweets/tree/main/heuristics.

the models and the MTEB leaderboard rankings [13]. After some initial experiments, we used the `twitter-roberta-base-2022-154m` [14] model since it is trained on data similar to the data found in the dataset.

The main results on the development dataset and the final model results are discussed in Section 4.2.

4. Results and Discussion

In this section, we present the results for *Task 1* and *Task 4a* for the development datasets and the rankings for the best performing model for the test datasets.

4.1. Task 1: Sentence Subjectivity

The results of the challenge indicate varying performance for different configurations for the test data. The monolingual models show a range of scores, with Mono-German and Mono-English achieving the highest F1 score of 0.72. The multilingual model achieved a score of 0.65. For the zero-shot configurations, performance varies significantly, with zero-shot Romanian achieving the highest score of 0.74, while zero-shot Greek has the lowest at 0.41. We achieved a notable third place for Ukrainian, with a score of 0.64. These results suggest that the effectiveness of multilingual and zero-shot approaches is highly dependent on a specific language and context. It is also possible that the performance of automatic translations influences the results, even though the translations appeared accurate upon manual review.

Table 4
Results and rankings for *Task 1*, with macro $F1$ scores for short and long sentences. $F1$ scores higher than the overall $F1$ are shown in bold.

Task					
Configuration	Language	$F1$ (Overall)	Rank	$F1$ (Short)	$F1$ (Long)
Monolingual	Italian	0.68	12/15	0.61	0.70
	Arabic	0.51	9/15	0.36	0.52
	German	0.72	12/17	1.00	0.72
	English	0.72	11/24	0.64	0.72
Multilingual	All	0.65	12/17	0.62	0.65
Zero-shot	Polish	0.55	12/15	0.51	0.55
	Ukrainian	0.64	3/15	0.67	0.63
	Greek	0.41	10/15	0.73	0.41
	Romanian	0.74	9/15	0.88	0.72

To further investigate the impact of sentence length on model performance, we conducted a dedicated experiment comparing the classification $F1$ scores for short sentences (less than 40 characters) and long sentences (more than 40 characters) across all languages. The results are reported in Table 4. In most cases, the models performed better on long sentences, suggesting that longer inputs provide richer linguistic and semantic cues that facilitate the detection of subjectivity. This is likely because subjective discourse often relies on nuanced expressions, modifiers, or contextual markers that are absent in very short statements, leading to the observed drop in score.

This observation is valid for all the monolingual configurations[8]. In the multilingual and zero-shot configurations, this trend is less pronounced and even reversed for Romanian, Greek, and Ukrainian, where the highest performances are achieved on short sentences. A possible explanation is that the training process learns to handle longer sentences more effectively, but this ability does not transfer well in a zero-shot (translated) scenario, where shorter sentences tend to be simpler and, consequently, easier to classify.

[8]The perfect $F1$ score observed on short German sentences is not statistically significant due to the extremely small number of samples (only six), while in any other configurations they are more than 30.

4.2. Task 4a: Scientific Web Discourse Detection

The results for *Task 4a* for the testing data are presented in Table 5. When using multiple individual experiments, we only focused on the individual classification results and did not compute the micro-averaged $F1$ for these models, as we only used the results to select the best-performing model for the evaluation on the unlabelled data.

Table 5
Model performance for *Task 4a* on the development dataset.

					F1		
Method	Base Model	Head	Features	MA-*F1*	`Cat1`	`Cat2`	`Cat3`
Embed	all-minilm	SVC / LogisticRegression / GaussianNB	Text	–	0.71	0.75	0.85
		SVC / NuSVC / ExtraTreesClassifier	Text + Oversampling	–	0.75	0.76	0.86
		BernoulliNB / Perceptron / RidgeClassifier	Text + Heuristics	–	0.67	0.77	0.88
		KNeighborsClassifier / SVC / SVC	Text + Heuristics + Oversampling	–	0.65	0.82	0.82
Embed	bge-m3	SVC / LogisticRegression / GaussianNB	Text	–	0.71	0.75	0.85
		SVC / NuSVC / ExtraTreesClassifier	Text + Oversampling	–	0.75	0.76	0.86
		KNeighborsClassifier / SGDClassifier / SVC	Text + Heuristics	–	0.70	0.82	0.78
		NuSVC / NuSVC / SVC	Text + Heuristics + Oversampling	–	0.74	0.84	0.84
SetFit	all-minilm	LogisticRegression	Text	0.67	0.70	0.62	0.71
	bge-base-en-v1.5	LogisticRegression	Text	0.62	0.63	0.40	0.84
	twitter-roberta-large-2022-154m	LogisticRegression	Text	0.83	0.80	0.82	0.81
FT	twitter-roberta-base-2022-154m		Text	0.87	0.83	0.90	0.88
	twitter-roberta-large-2022-154m	NearestCentroid / GaussianNB / NearestCentroid	Text	0.89	0.87	0.92	0.87

Although the best model achieved the third position on the development leaderboard (micro-average $F1 = 0.8887$), it achieved first position on the final evaluation dataset (micro-average $F1 = 0.7998$), suggesting less overfitted results compared to the other models. The model may also benefit from using a weighted loss function.

While the use of specific classification heads for each category showed a small improvement over the regression head used by the SetFit models and the fine-tuned models, the approach may have benefited the final model rankings given the small difference between the first three results (< 0.01)

Besides ranking first for the micro-average F1 across the categories, the model showed strong results for each category, ranking third for the identification of scientific claims (`cat1`, $F1 = 0.7932$) and second for both the identification of scientific references (`cat2`, $F1 = 0.7736$) and the identification of scientific entities (`cat3`, $F1 = 0.8325$).

The embedding models benefited from using the oversampling method described in Section 3.2, but during tests, the impact on the fine-tuned model was mostly negative. Therefore, it was not integrated into the fine-tuned models. Similarly, the heuristics helped with the accuracy of specific models (e.g.

`all-minilm`) but did not provide advantages when used with fine-tuned models.

SetFit models were slow to train due to the number of pairs generated from the training data. As a result, the training of these models was limited and the results were not as good as the fine-tuned models.

Overall, fine-tuning performed best, but the top results were obtained when specific classification heads were used for each category. This approach shares some similarities with SetFit, but instead of fitting a sentence encoder and then training a classifier on the embeddings, a model is fine-tuned, and then its embeddings are used for training a conventional classifier (or in our case, multiple classifiers).

5. Conclusions and Future Work

In this paper, we have presented the ClimateSense's team results for the CheckThat! 2025 Lab Challenge *Task 1* and *Task 4a*. Our results highlight the effectiveness of fine-tuning pre-trained language models, particularly for *Task 4a*, where ClimateSense secured first place on the final evaluation leaderboard. This result suggests that combining fine-tuned LLMs with more traditional classification models can be beneficial compared to simply relying on fine-tuning LLMs with a classification head.

While our approaches have yielded competitive results, several avenues remain for further exploration and improvement. For *Task 1* (Sentence Subjectivity), the performance limitations observed with Zephyr deserve further study. Future work should explore the use of larger language models and alternative prompting strategies, such as *chain of thought* or *chain of verification* techniques, to potentially enhance zero-shot capabilities. Additionally, in the zero-shot scenario, it may be beneficial to reverse the current approach by creating language-specific training sets through the translation of English sentences into target languages such as Ukrainian, Greek, and others, rather than translating those languages into English. Finally, given that short or isolated sentences tend to be the most misleading and difficult to classify, future experiments could evaluate the impact of systematically excluding these instances from the dataset. During the development of the *Task 4a* models, we identified data from SciTweet [11] that could improve the identification of scientific claims (`cat 1`). Future work should investigate its potential as additional training data for *Task 4a*. Our experiments with zero-shot models and paraphrasing were constrained by available computational resources. This limitation also influenced the types of models we could fine-tune. Similarly to *Task 1* future work, future research should explore more powerful LLMs. Future experiments could also investigate optimising the fine-tuning process on limited resources (e.g., quantisation, PEFT [15]) to improve the performance of the classification models.

Finally, it would be beneficial to evaluate the generalisation of the proposed models across additional social media platforms and news sources to assess the robustness of the models in different environments. Investigating explainability methods for both tasks could also enhance the interpretability and trustworthiness of the proposed approaches, particularly for fact-checking applications.

Acknowledgments

This work was supported by the European CHIST-ERA program within the ClimateSense project (Grant ID ANR-24-CHR4-0002, EPSRC EP/Z003504/1).

Declaration on Generative AI

During the preparation of this work, the authors used MistralAI's Le Chat and Google Gemini in order to: *Grammar and spelling check*, *Paraphrase and reword.* After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CLEF 2025, Madrid, Spain, 2025.

[2] F. Alam, J. M. Struß, T. Chakraborty, S. Dietze, S. Hafid, K. Korre, A. Muti, P. Nakov, F. Ruggeri, S. Schellhammer, V. Setty, M. Sundriyal, K. Todorov, V. Venktesh, Overview of the CLEF-2025 CheckThat! Lab: Subjectivity, fact-checking, claim normalization, and retrieval, in: J. Carrillo-de Albornoz, J. Gonzalo, L. Plaza, A. García Seco de Herrera, J. Mothe, F. Piroi, P. Rosso, D. Spina, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Sixteenth International Conference of the CLEF Association (CLEF 2025), 2025.

[3] F. Ruggeri, A. Muti, K. Korre, J. M. Struß, M. Siegel, M. Wiegand, F. Alam, R. Biswas, W. Zaghouani, M. Nawrocka, B. Ivasiuk, G. Razvan, A. Mihail, Overview of the CLEF-2025 CheckThat! lab task 1 on subjectivity in news article, in: [1], 2025.

[4] S. Hafid, Y. S. Kartal, S. Schellhammer, K. Boland, D. Dimitrov, S. Bringay, K. Todorov, S. Dietze, Overview of the CLEF-2025 CheckThat! lab task 4 on scientific web discourse, in: [1], 2025.

[5] N. Reimers, I. Gurevych, Sentence-BERT: Sentence embeddings using Siamese BERT-networks, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3982–3992. URL: https://aclanthology.org/D19-1410/. doi:10.18653/v1/D19-1410.

[6] L. Wang, N. Yang, X. Huang, B. Jiao, L. Yang, D. Jiang, R. Majumder, F. Wei, Text embeddings by weakly-supervised contrastive pre-training, arXiv (2024). URL: https://arxiv.org/abs/2212.03533. arXiv:2212.03533.

[7] B. Warner, A. Chaffin, B. Clavié, O. Weller, O. Hallström, S. Taghadouini, A. Gallagher, R. Biswas, F. Ladhak, T. Aarsen, N. Cooper, G. Adams, J. Howard, I. Poli, Smarter, Better, Faster, Longer: A Modern Bidirectional Encoder for Fast, Memory Efficient, and Long Context Finetuning and Inference (2024). URL: https://arxiv.org/abs/2412.13663. arXiv:2412.13663.

[8] M. Müller, M. Salathé, P. Kummervold, COVID-Twitter-BERT: A natural language processing model to analyse COVID-19 content on Twitter, Frontiers in Artificial Intelligence 6 (2023). doi:10.3389/frai.2023.1023281.

[9] L. Tunstall, E. E. Beeching, N. Lambert, N. Rajani, K. Rasul, Y. Belkada, S. Huang, L. Von Werra, C. Fourrier, N. Habib, et al., Zephyr: Direct Distillation of LM Alignment, in: First Conference on Language Modeling, 2024.

[10] P. He, J. Gao, W. Chen, Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, 2021. arXiv:2111.09543.

[11] S. Hafid, S. Schellhammer, S. Bringay, K. Todorov, S. Dietze, SciTweets-A Dataset and Annotation Framework for Detecting Scientific Online Discourse, in: Proceedings of the 31st ACM International Conference on Information & Knowledge Management, 2022, pp. 3988–3992.

[12] J. Chen, S. Xiao, P. Zhang, K. Luo, D. Lian, Z. Liu, Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation, 2024. arXiv:2402.03216.

[13] K. Enevoldsen, I. Chung, I. Kerboua, M. Kardos, A. Mathur, D. Stap, J. Gala, W. Siblini, D. Krzemiński, G. I. Winata, S. Sturua, S. Utpala, M. Ciancone, M. Schaeffer, G. Sequeira, D. Misra, S. Dhakal, J. Rystrøm, R. Solomatin, Ömer Çağatan, A. Kundu, M. Bernstorff, S. Xiao, A. Sukhlecha, B. Pahwa, R. Poświata, K. K. GV, S. Ashraf, D. Auras, B. Plüster, J. P. Harries, L. Magne, I. Mohr, M. Hendriksen, D. Zhu, H. Gisserot-Boukhlef, T. Aarsen, J. Kostkan, K. Wojtasik, T. Lee, M. Šuppa, C. Zhang, R. Rocca, M. Hamdy, A. Michail, J. Yang, M. Faysse, A. Vatolin, N. Thakur, M. Dey, D. Vasani, P. Chitale, S. Tedeschi, N. Tai, A. Snegirev, M. Günther, M. Xia, W. Shi, X. H. Lù, J. Clive, G. Krishnakumar, A. Maksimova, S. Wehrli, M. Tikhonova, H. Panchal, A. Abramov, M. Ostendorff, Z. Liu, S. Clematide, L. J. Miranda, A. Fenogenova, G. Song, R. B. Safi, W.-D. Li, A. Borghini, F. Cassano, H. Su, J. Lin, H. Yen, L. Hansen, S. Hooker, C. Xiao, V. Adlakha, O. Weller, S. Reddy, N. Muennighoff, Mmteb: Massive multilingual text embedding benchmark, arXiv preprint arXiv:2502.13595 (2025).

URL: https://arxiv.org/abs/2502.13595. doi:10.48550/arXiv.2502.13595.

[14] D. Loureiro, K. Rezaee, T. Riahi, F. Barbieri, L. Neves, L. E. Anke, J. Camacho-Collados, Tweet insights: A visualization platform to extract temporal insights from twitter, arXiv preprint arXiv:2308.02142 (2023).

[15] S. Mangrulkar, S. Gugger, L. Debut, Y. Belkada, S. Paul, B. Bossan, Peft: State-of-the-art parameter-efficient fine-tuning methods, https://github.com/huggingface/peft, 2022.