NotMyNarrative at SemEval-2025 Task 10: Do Narrative Features Share Across Languages in Multilingual Encoder Models?

Géraud Faye[1,2], Guillaume Gadek[1], Wassila Ouerdane[2], Céline Hudelot[2], Sylvain Gatepaille[1]

[1]Airbus Defence and Space, [2]Université Paris-Saclay - CentraleSupélec - MICS

Abstract

Narratives are a tool to propagate ideas that are sometimes well hidden in press articles. The SemEval-2025 Task 10 focuses on detecting and extracting such narratives in multiple languages. In this paper, we explore the capabilities of encoder-based language models to classify texts according to the narrative they contain. We show that multilingual encoders outperform monolingual models on this dataset, which is challenging due to the small number of samples per class per language. We perform additional experiments to measure the generalization of features in multilingual models to new languages.

1 Introduction

With the complexity of current geopolitical events, persuasion techniques have become less explicit in online content. Shared content often share a vision of the world used to interpret current events, which can influence the world vision of the readers. These are called narratives, and automatically detecting them has become a topic of interest for the machine learning community (Piskorski et al., 2025). Narratives can also be stated explicitly, but are more harmful when they are implicit in the text, like persuasion techniques are.

In this paper, we propose a multilingual approach (English, European Portuguese, Hindi, Bulgarian, and Russian) to identify whether or not a predefined narrative is present in a text and, if that is the case, what narrative it is. It is based on a standard multilingual encoder with a unique classification head for all narratives of the task.

We find that multilingual models perform better than individual monolingual models, using all of the provided data by the task organizers. While our proposed approach is not trying to be the best-performing (only in the top 50% of teams for only two languages), it relies on light language models that run on modest hardware and works the same for all languages.

2 Background

We propose a system for the Subtask 2: Narrative classification. The problem is framed as the following: given a text, identify if the text contains one, several, or none of the narratives defined by (Stefanovitch et al., 2025). The proposed narratives are part of a two-level taxonomy. However, we chose to ignore the additional information from the higher-level labels and focused directly on fine-grained narrative classification, which is the main focus of the task and on which the narrative used for the leaderboard is based. The problem is multi-class and multi-label, with 93 narratives to detect, some of which only appearing in some languages. The distribution of the number of occurrences for each class is given in Figure 1.

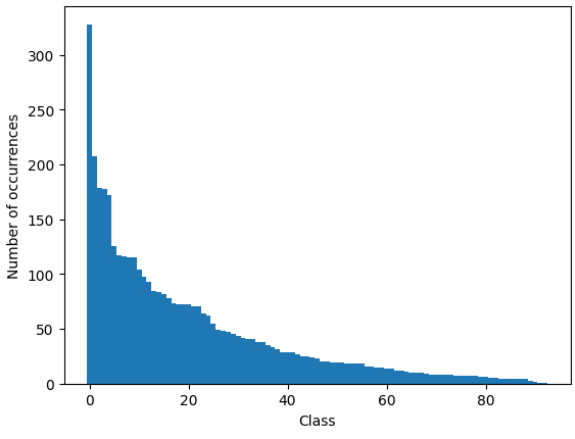

Figure 1: The number of occurrences by class, sorted in decreasing order. The distribution is unbalanced, with a median of only 23 occurrences per class.

The articles are the length of a regular news article, with about 410 words on average. They cover either news about climate change or the Ukraine-Russia war and exceptionally contain narratives

Proceedings of the The 19th International Workshop on Semantic Evaluation (SemEval-2025), pages 58–66

related to the two topics. Each article has an average of 2.3 labels and a median of 2 labels.

Given the limited number of samples per class in the dataset, we chose to work in a multilingual setting to maximize the number of samples seen by class during training.

Another challenge of the task is the multi-label constraint. Usual mono-label classification uses a Softmax activation function, outputting probabilities for each class, even when out-of-distribution. This is not possible for multi-label classification for which several labels can be applied to the same text, requiring additional steps.

3 System overview

Our proposed system is based on encoder language models trained solely on the provided data for the task. The choice of encoder models is motivated by their wide use in text classification tasks, especially for misinformation detection (Pelrine et al., 2021). The encoder produces an embedding that is then processed by a two-layer classification head with a sigmoid activation function and 94 output neurons, one for each class plus one for the absence of narrative.

During training, we consider that each neuron with an activation over 50% is activated. However, preliminary experiments showed that this setting could not be kept for inference, with all neurons activating at values below this threshold for almost all test samples. To solve this problem, we propose an adaptative threshold for multi-label classification based on the activation of the `No narrative` class neuron. If this neuron is the most activated, it means that the absence of a narrative is more plausible than the presence of any narrative seen during training. Each narrative corresponding to a neuron more activated than the `No narrative` neuron is considered present in the text. This neuron could be considered as the *neutral* or *control* class, determining if one of the training classes is found in the text. Figure 2 shows a global schema of the system.

One point of interest for our study is the multilingualism of model embeddings for narrative classification. Several types of state-of-the-art models were used:

- Multilingual models: experiments are done on models supporting all provided languages using all training data. For this type of models, we chose two models, the widely used XLM-RoBERTa-large[1] (Conneau et al., 2019) (561M parameters, noted RoBERTa in experiments) and mDeBERTa-v3-base[2] (He et al., 2021) (86M parameters, noted mDeBERTa in experiments).

- Monolingual models: we chose ModernBERT[3] (Warner et al., 2024) for English, Albertina PT-PT[4] (Rodrigues et al., 2023) for Portuguese, MuRIL[5] (Khanuja et al., 2021) for Hindi, and for lack of strictly monolingual models, SlavicBERT[6] (Arkhipov et al., 2019) for Bulgarian and Russian. These models were chosen as they are the state-of-the-art specialized monolingual models for each language at the time of writing.

Monolingual models are trained with the corresponding language data. Multilingual models were used for two types of experiments:

- A first one with all training data, to measure if using samples from multiple languages improves performance over using only one language.

- A second one with all training data except one language. This will allow us to measure how narrative embeddings transfer to new languages and if models trained with additional data can function in new languages. The five provided languages are a good opportunity for this experiment, as they cover three different alphabets (Latin, Hindi, and Cyrillic).

4 Experimental setup

The given train data is split in two with a random 80/20 split. Models are trained on the first 80% and evaluated on the remaining 20% at the end of each epoch. Models are trained for a maximum of 100 epochs, and an early stopping strategy with a patience of 5 is used. If the F1 score on the fine narratives on the 20% of data does not improve for

[1] https://huggingface.co/FacebookAI/xlm-roberta-large

[2] https://huggingface.co/microsoft/mdeberta-v3-base

[3] https://huggingface.co/answerdotai/ModernBERT-large

[4] https://huggingface.co/PORTULAN/albertina-900m-portuguese-ptpt-encoder

[5] https://huggingface.co/google/muril-large-cased

[6] https://huggingface.co/DeepPavlov/bert-base-bg-cs-pl-ru-cased

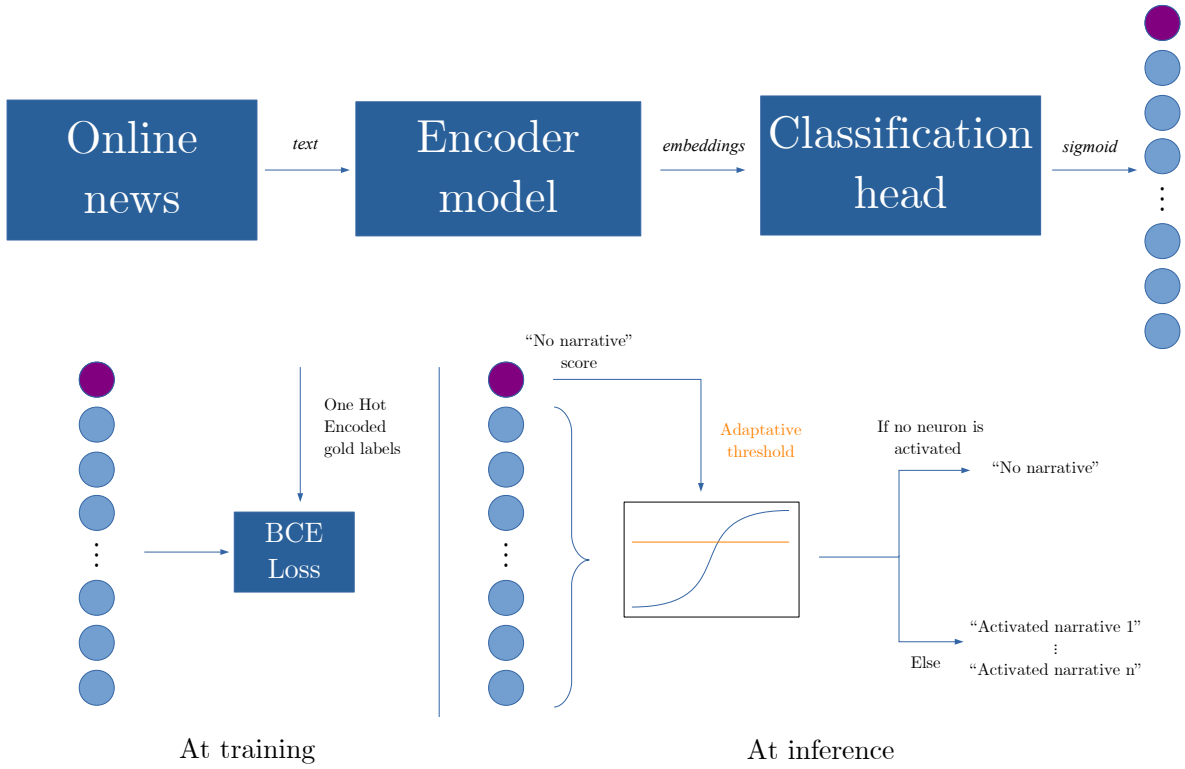

Figure 2: The global system architecture and label computation.

five epochs, the model is restored to its state with the best F1 score. The dev data has been used for evaluation, and the reported results are computed for this split. The final models used for the test submission are chosen per language, based on the configuration giving the best F1 score on the fine narratives on the dev split.

We report the F1 scores on the coarse and the fine narratives for each experiment.

Each model is trained with a batch size of 8 and a learning rate of 10^{-5} with an AdamW optimizer (Loshchilov and Hutter, 2019), which is a common default choice for such models. Models come from HuggingFace and the transformers library.

The model and classification head are wrapped in a PyTorch Lightning[7] `LightningModule`. Because classes are unbalanced, we use a sampler from the pytorch-multilabel-balanced-sampler module[8], and more specifically the `LeastSampledClassSampler`, which returns a random sample with a label from the least sampled class at each moment.

5 Results

5.1 General results on the task

Firstly, we report results on the dev dataset for model selection in Table 1. Overall, multilingual

[7] https://github.com/Lightning-AI/pytorch-lightning

[8] https://github.com/issamemari/pytorch-multilabel-balanced-sampler

	Model	EN		PT		HI		BG		RU	
		Coarse	Fine	Coarse	Fine	Coarse	Fine	Coarse	Fine	Coarse	Fine
All languages	RoBERTa	**0.432**	**0.325**	0.318	0.208	0.162	0.161	**0.361**	**0.234**	0.296	0.100
	mDeBERTa	0.362	0.309	**0.442**	**0.270**	**0.238**	**0.168**	0.309	0.211	**0.276**	**0.148**
Language split	ModernBERT	0.268	0.268	-	-	-	-	-	-	-	-
	AlBERTina	-	-	0.345	0.235	-	-	-	-	-	-
	Muril	-	-	-	-	0.176	0.148	-	-	-	-
	Slavic-bert	-	-	-	-	-	-	0.243	0.116	-	-
	Slavic-bert	-	-	-	-	-	-	-	-	0.174	0.070

Table 1: Results for several standard encoder models. Each row represents one experiment, and results are given for all languages used during training. The best results for each language (regarding the F1 score on fine narratives) are in bold.

RoBERTa	EN		PT		HI		BG		RU	
	Coarse	Fine	Coarse	Fine	Coarse	Fine	Coarse	Fine	Coarse	Fine
All languages	0.432	0.325	0.318	0.208	0.162	**0.161**	0.361	0.234	0.296	0.100
No EN	0.420	0.319	0.210	0.131	0.143	0.073	0.291	0.213	0.238	0.109↑
No PT	**0.438**↑	0.323	**0.383**↑	**0.220**↑	0.145	0.093	0.389↑	0.274↑	**0.307**↑	0.119↑
No HI	0.426	0.321	0.291	0.160	**0.167**↑	0.128	0.391↑	**0.292**↑	0.238	0.096
No BG	0.403	**0.346**↑	0.259	0.139	0.121	0.069	0.247	0.180	0.296	0.098
No RU	0.347	0.289	0.289	0.151	0.116	0.077	**0.430**↑	0.257↑	0.256	**0.153**↑

Table 2: Generalization study to new languages with XLM-RoBERTa-large. Grayed results are results obtained on languages seen during training. The best approach has been selected based on the F1-score on the fine narratives. Results are marked with ↑ when results are better than the results when trained on all languages.

mDeBERTa	EN		PT		HI		BG		RU	
	Coarse	Fine	Coarse	Fine	Coarse	Fine	Coarse	Fine	Coarse	Fine
All languages	**0.362**	**0.309**	**0.442**	**0.270**	**0.238**	**0.168**	0.309	0.211	0.276	0.148
No EN	0.303	0.249	0.437	0.249	0.100	0.059	0.326↑	0.218↑	0.198	0.101
No PT	0.302	0.239	0.332	0.206	0.160	0.128	0.260	0.164	**0.300**↑	0.109
No HI	0.343	0.300	0.212	0.133	0.114	0.097	0.228	0.152	0.270	**0.151**↑
No BG	0.299	0.240	0.349	0.187	0.170	0.121	0.339↑	0.226↑	0.193	0.144
No RU	0.346	0.274	0.379	0.193	0.162	0.120	**0.372**↑	**0.289**↑	0.261	0.133

Table 3: Generalization study to new languages with mDeBERTa-v3-base. Grayed results are results obtained on languages seen during training. The best approach has been selected based on the F1-score on the fine narratives. Results are marked with ↑ when results are better than the results when trained on all languages.

models perform better than monolingual models with this little data for each class. There is no clear winner between XLM-RoBERTa-large and mDeBERTa-v3-base, but the latter is 6.5 times lighter. Moreover, mDeBERTa-v3-base performs better on average than XLM-RoBERTa-large, with a mean F1 score of 0.344 versus 0.257 on fine narratives. In addition, the two models seem to perform worse for non-West-European languages. The same observation can be made for specialized models, which could also be explained by data distribution for these specific languages.

Quantitatively, when compared to other systems on the final test submissions, simple encoder models are not the best for identifying narratives but still beat the baseline for all languages. The official leaderboard[9] allows to compare models performance directly. Our model performed 13/28 in English, 12/14 in Portuguese, 8/14 in Hindi, and 9/12 in Bulgarian, and would have performed 13/16 in Russian (results were not submitted on time).

Our models tend to make cautious predictions, and in a little more than 40% of dev samples, no narrative was detected when it should have been, which leads to lower scores overall.

[9] https://propaganda.math.unipd.it/semeval2025task10/leaderboardv3.html

5.2 Generalization on new languages

After the final submission, additional experiments were run to measure how well the tested multilingual models would generalize to other languages. To this end, we train the same models several times with a whole language left out each time. Reported results are computed on the dev set and given in Table 2 and 3. Results are grayed when computed on a language seen during training, an arrow is displayed when the ablated model performs better than the same model trained with all languages, and bold results are the best obtained for a specific language among all tested models.

In most cases, performance does not drastically change on one language if it is removed from the training languages (-3.475% for mDeBERTa and +0.75% for XLM-RoBERTa on average).

Performance increased for XLM-RoBERTa due to strange behaviors in Portuguese and Russian. Counterintuitively, removing these languages increases performance on the dev set. In general, for XLM-RoBERTa, removing a language improves performance in at least one other language. This hints that while this model can process multiple languages, features are not shared evenly across languages. Portuguese features rely on other languages, as performance improves with No PT. Moreover, removing Portuguese also helps performance in Bulgarian and Russian, showing that Portuguese disturbs the features of other languages.

Also, removing Russian improves Bulgarian (close in vocabulary but different in grammar) performance, showing that the model may confound the two languages.

The same observation can be done with mDeBERTa. In most cases, mDeBERTa performs better than XLM-RoBERTa, except for English. mDeBERTa seems more balanced between languages and shows good transfer capabilities, with the model performing better when trained on all data for all languages but Slavic ones. This generalization is possible at the cost of the performance in English.

In conclusion, we observe that XLM-RoBERTa generalizes better than mDeBERTa on new languages, but that if given data in multiple languages, mDeBERTa is the model that will be the best to leverage all the information from all languages.

5.3 Error analysis

To further understand how our models performed, we chose to do an error analysis on the dev set for mDeBERTa trained on all languages, our best-performing model on average. It misses many narratives on the dev set. All articles with no narratives were correctly labeled, but 72 were false negatives for the absence of narratives (over the 178 articles in the dev set). In this sense, the model is conservative and when unsure, does not try to guess a narrative. The following analysis has been done on the part of the dev split for which the model predicted at least one narrative.

There is no simple way of showing a confusion matrix for multi-label problems, as the recommendation would be to plot as many label-specific confusion matrices as there are labels. To simplify our analysis, we propose a "confusion-like" matrix to check for common errors in the predictions, which detailed computations are given in Appendix A.

To summarize computations, accurate predictions are counted as usual, but the wrong predictions are only partially counted, sharing a weight of 1 among wrongly predicted labels and unpredicted gold labels. Generally, the idea of this matrix is to perform qualitative error analysis, which is done in this Section. The confusion-like matrix global form is in Figure 3, and the whole matrix with labels is in Appendix A, in Figure 4.

There is a clear split between climate change (CC) and war-related (URW) narratives (the first 40 narratives for CC and the last 48 ones for URW). Moreover, some rows (resp. columns) are filled with zeros, corresponding to a lack of data in the dev (resp. training) split.

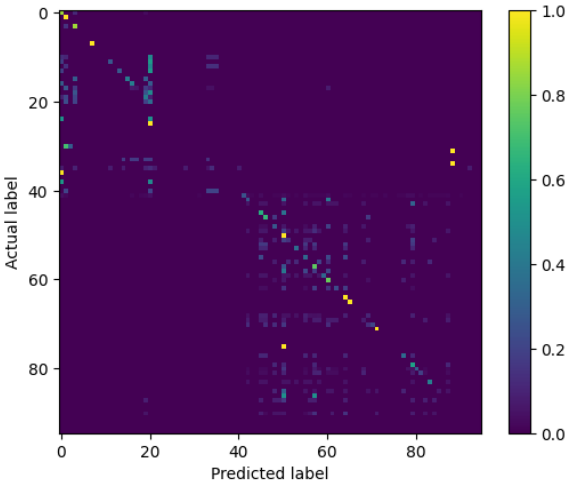

Figure 3: Confusion-like matrix general form. It can be used to identify clusters of wrong predictions quickly. A more detailed confusion matrix with the labels is given in Appendix A.

Most CC narratives were predicted as "Criticism of climate movement" and "Criticism of climate policies," which are the main topics of CC narratives globally. The second main group of CC narrative predictions is on the first narratives of the ontology, hinting that geopolitical agendas behind climate policies are hidden. The same observation can be made on URW narratives, with most predictions covering the "Discrediting Ukraine" and "Discrediting the West" narratives and the central narratives of the URW topic. Some outliers appear in the matrix, but they only represent one sample each, highlighting them in the row-normalized matrix.

Overall, the system is able to detect large categories of narratives, but struggles for fine narratives, showing a bias for well-represented narratives from the training set. More specific encoders should be used with less fine narratives to detect to be able to better detect these fine narratives.

6 Conclusion and Future Works

In this paper, we explored the capabilities of multilingual encoder-based models for the task of narrative classification. We proposed a method with an adaptative threshold for multi-label classification tasks and showed that it performs reasonably well, especially for high-resource languages.

Additional experiments on language ablations showed differences between models' behavior,

with XLM-RoBERTa generalizing better on unseen languages, but mDeBERTa generally performing better when trained with all languages.

The proposed approach could be enhanced by using data augmentation and hierarchical classification; ideas proposed by (Singh et al., 2025; Assis et al., 2025; Huayang Li, 2025). For real use cases, performance on the coarse labels may be more important to detect the presence or absence of narratives before using more specialized models if needed. The main challenge for our model was the limited number of samples by class, which the addition of new annotated data could alleviate. In addition to that, the proposed system only works with a pre-defined set of initially defined narratives. It could be possible to reuse the adaptative threshold idea to detect when new narratives appear in new articles. Moreover, other thresholding strategies could be used, by instance by adding a margin around the adaptative threshold in order to maximize either precision or recall, depending on the use case.

Acknowledgments

We would like to thanks the anonymous reviewers for their time and precious feedback on the paper.

EUCINF project is co-funded by European Union under grant agreement N°101121418. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the granting authority can be held responsible for them.

The PhD project of the first author was provided with computing AI and storage resources by GENCI at CINES thanks to the grant 2024-AD011015826 on the supercomputer Adastra's MI250x partition.

References

Mikhail Arkhipov, Maria Trofimova, Yuri Kuratov, and Alexey Sorokin. 2019. Tuning multilingual transformers for language-specific named entity recognition. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 89–93, Florence, Italy. Association for Computational Linguistics.

Gabriel Assis, Lívia de Azevedo, João Vitor de Moraes, Laura Alvarenga, and Aline Paes. 2025. Irapuarani at semeval-2025 task 10: Evaluating strategies combining small and large language models for multilingual narrative detection. In *Proceedings of the 19th International Workshop on Semantic Evaluation*, SemEval 2025, Vienna, Austria.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing. *CoRR*, abs/2111.09543.

Jingjie Zeng Huayang Li, Pengyuan Du. 2025. Duttask10 at semeval-2025 task 10: Thoughtflow: Hierarchical narrative classification via stepwise prompting. In *Proceedings of the 19th International Workshop on Semantic Evaluation*, SemEval 2025, Vienna, Austria.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, Shruti Gupta, Subhash Chandra Bose Gali, Vish Subramanian, and Partha Talukdar. 2021. Muril: Multilingual representations for indian languages. *Preprint*, arXiv:2103.10730.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Kellin Pelrine, Jacob Danovitch, and Reihaneh Rabbany. 2021. The surprising performance of simple baselines for misinformation detection. In *Proceedings of the Web Conference 2021*, WWW '21, page 3432–3441, New York, NY, USA. Association for Computing Machinery.

Jakub Piskorski, Tarek Mahmoud, Nikolaos Nikolaidis, Ricardo Campos, Alípio Jorge, Dimitar Dimitrov, Purificação Silvano, Roman Yangarber, Shivam Sharma, Tanmoy Chakraborty, Nuno Guimarães, Elisa Sartori, Nicolas Stefanovitch, Zhuohan Xie, Preslav Nakov, and Giovanni Da San Martino. 2025. SemEval-2025 task 10: Multilingual characterization and extraction of narratives from online news. In *Proceedings of the 19th International Workshop on Semantic Evaluation*, SemEval 2025, Vienna, Austria.

João Rodrigues, Luís Gomes, João Silva, António Branco, Rodrigo Santos, Henrique Lopes Cardoso, and Tomás Osório. 2023. Advancing neural encoding of portuguese with transformer albertina pt-*. *Preprint*, arXiv:2305.06721.

Iknoor Singh, Carolina Scarton, and Kalina Bontcheva. 2025. Gatenlp at semeval-2025 task 10: Hierarchical three-step prompting for multilingual narrative classification. In *Proceedings of the 19th International Workshop on Semantic Evaluation*, SemEval 2025, Vienna, Austria.

Nicolas Stefanovitch, Tarek Mahmoud, Nikolaos Nikolaidis, Jorge Alípio, Ricardo Campos, Dimitar Dimitrov, Purificação Silvano, Shivam Sharma, Roman Yangarber, Nuno Guimarães, Elisa Sartori, Ana Filipa Pacheco, Cecília Ortiz, Cláudia Couto, Glória Reis de Oliveira, Ari Gonçalves, Ivan Koychev, Ivo Moravski, Nicolo Faggiani, Sopho Kharazi, Bonka Kotseva, Ion Androutsopoulos, John Pavlopoulos, Gayatri Oke, Kanupriya Pathak, Dhairya Suman, Sohini Mazumdar, Tanmoy Chakraborty, Zhuohan Xie, Denis Kvachev, Irina Gatsuk, Ksenia Semenova, Matilda Villanen, Aamos Waher, Daria Lyakhnovich, Giovanni Da San Martino, Preslav Nakov, and Jakub Piskorski. 2025. Multilingual Characterization and Extraction of Narratives from Online News: Annotation Guidelines. Technical Report JRC141322, European Commission Joint Research Centre, Ispra (Italy).

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *Preprint*, arXiv:2412.13663.

A Full multi-label confusion matrix with labels

For clarity, we provide the pseudo-code used to compute the "confusion-like" matrix for multi-label classification problems in Algorithm 1

The full confusion-like matrix with narrative labels is given in Figure 4.

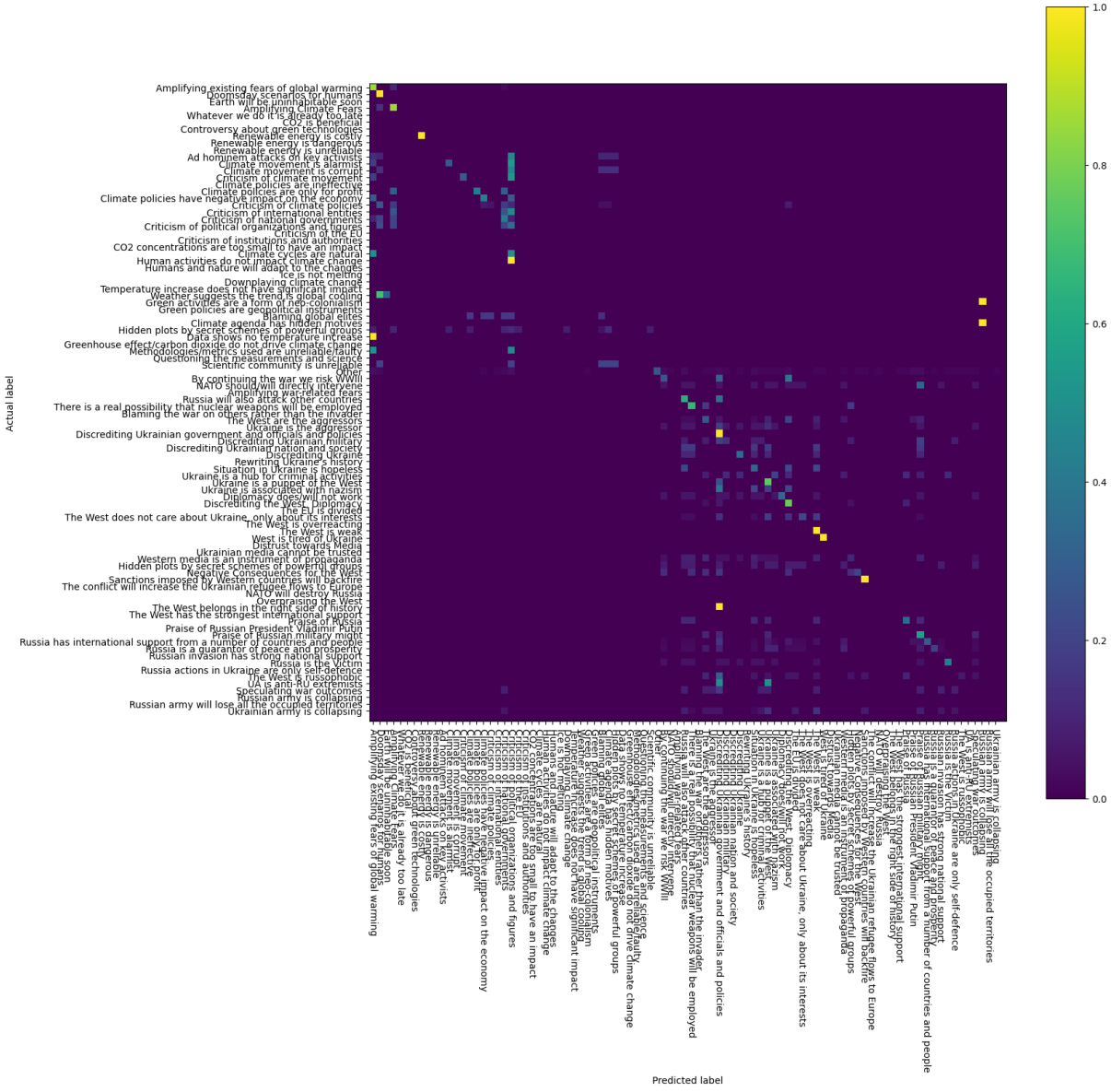

Figure 4: Confusion matrix for predictions of narratives (predictions giving no narratives are ignored).

Algorithm 1: Confusion-like matrix computations

```
begin Compute confusion-like matrix
    confusion_matrix = zeros(n_nar,n_nar);
    for sample in dataset do
        predictions ⇐ model(sample);
        wrong_predictions ⇐ predictions;
        not_predicted ⇐ gold_labels(sample);
        for prediction in predictions do
            if prediction ∈ gold_labels(sample) then
                confusion_matrix[prediction,prediction] += 1;
                wrong_predictions.remove(prediction);
                not_predicted.remove(prediction);
        for prediction in wrong_predictions do
            for label in not_predicted do
                confusion_matrix[label, prediction] += 1 / size(not_predicted);
        for label in not_predicted do
            for prediction in wrong_prediction do
                confusion_matrix[label, prediction] += 1 / size(wrong_prediction);
    normalize_by_row(confusion_matrix);
    return confusion_matrix;
```