FTD at SemEval-2023 Task 3: News Genre and Propaganda Detection by Comparing Mono- and Multilingual Models with Fine-tuning on Additional Data

Mikhail Lepekhin
MIPT / Moscow
lepehin.mn@phystech.edu

Serge Sharoff
University of Leeds / UK
s.sharoff@leeds.ac.uk

Abstract

We report our participation in the SemEval-2023 shared task on propaganda detection and describe our solutions with pre-trained models and their ensembles. For Subtask 1 (News Genre Categorisation), we report the impact of several settings, such as the choice of the classification models (monolingual or multilingual or their ensembles), the choice of the training sets (base or additional sources), the impact of detection certainty in making a classification decision as well as the impact of other hyperparameters. In particular, we fine-tune models on additional data for other genre classification tasks, such as FTD. We also try adding texts from genre-homogenous corpora, such as Panorama, Babylon Bee for satire and Giganews for for reporting texts. We also make prepared models for Subtasks 2 and 3 with fine-tuning the corresponding models first for Subtask 1. The code needed to reproduce the experiments is available.[1]

1 Introduction

Non-topical text classification includes a wide range of tasks aimed at predicting a text property that is not connected directly to a text topic, so it is not simply detected via keywords, for example, predicting a text style, politeness, difficulty level, the age or the first language of its author, etc. Automatic genre identification (Santini et al., 2010) is one of the standard problems of non-topical text classification. It can be applied in many areas such as information retrieval, language teaching or linguistic research to find texts of specific genres for any topic. SemEval Shared Task 3 consists of 3 subtasks linked to the concept of non-topical text classification. Subtask 1 (News Genre Categorisation) is aimed at predicting expression of opinions, objective news reporting, or satire on the text level. Subtask 2 is on prediction of text framing with the possibility of multiple labels per article. Subtask 3 is about detecting the kind of persuation techniques on the paragraph level.

NLP shifted dramatically when large pre-trained models emerged. Multilingual BERT (Pires et al., 2019) shows impressive results on a variety of languages and often it even manages to beat the monolingual models. XLM-RoBERTa (Conneau et al., 2019) is a more powerful multilingual model. It is trained on more than 100 languages and uses a vocabulary with a higher number of tokens - more than 200K.

A problem with many non-topical classification setups concerns robustness. When a classifier sees a text that differs drastically from any text in the train set, it tends to misclassify it. And quite often, it is accompanied by a low probability of the max probable class. A recent study by (Tsymbalov et al., 2020) represents a metric of confidence of prediction that estimates how certain the classifier is on the given text. It could be helpful to understand whether the label from classifier is trustworthy and, if not, gives signal to replace it. In some tasks, it is rational to replace the uncertain predictions with the majority class label.

Even the best classifiers have their own flaws that could result in poor performance on some set of texts. In order to reduce the flaws, it could be helpful to use ensembles of diffirent architectures. A recent study (Lepekhin and Sharoff, 2022) shows that ensembles not only have higher accuracy on diverse test sets, but also give more confident predictions.

These are some of the ideas we tested by applying them to the shared task (Piskorski et al., 2023).

2 Subtask 1: News genres

We experiment with all the languages and report our results on the dev sets. In all the models we use, the input size is restricted to k tokens (in most cases, $k = 512$). To tackle it, we take first k tokens

[1] https://github.com/MikeLepekhin/SemEval2023_task3 - our GitHub repository

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 549–555

in any text we forward to the model. We do not make any additional data splits and use the dev sets from the task statement to validate and compare our approaches. The weights for the ensembles are also selected on the corresponding dev sets. We first train separately the individual models. Then we select the coefficients for the models and make ensemble as a linear combination of the class probabilities.

2.1 Model selection

We experimented with:

mBert the original multilingual BERT;

XLM-R XLM-Roberta;

mono a respective monolingual model

- ru RuBERT for Russian
- pl Polish BERT (dkleczek/bert-base-polish-uncased-v1)
- de German BERT (bert-base-german-cased)
- deE German Electra (german-nlp-group/electra-base-german-uncased)
- Slav Slavic BERT (DeepPavlov/bert-base-bg-cs-pl-ru-cased)

Ens an ensemble of two models with their weights. The sum of the weights of all the models should be equal to 1.

We also made a couple of experiments with the respective larger models:

mBert-l the original multilingual BERT;

XLM-R-l XLM-Roberta

2.2 Additional data

For each experiment, we use the data of the SemEval shared task. However, we also made experiments with additional datasets:

ML + using training data for more languages, for example, FI for a combination of French and Italian added to the English traning dataset;

FTD an existing dataset in English and Russian developed for the purposes of genre classification (Sharoff, 2018);

GN a small portion of the Giga News dataset for English (Cieri and Liberman, 2002);

BB Babylon Bee [2] is an american site with satiristic news.

P Panorama [3] is a Russian site with satiristic news, analogue to the American Babylon Bee. It was added to tackle the shortage of the *satire* texts in the training set.

One of the sources for additional data is FTD Table 1 developed for text genre classification. For the English and the Russian languages we can use models initially fine-tuned on the corresponding FTD datasets and fine-tune them on the shared task data, since Subtask 1 is similar to the task of text genre classification. Moreover, some of its classes more or less correspond to the classes of Subtask 1 of this shared task (A1 = opinion, A8 = reporting, A4 $\sim$ satire). Because of it, it is potentially useful to use additional information the model can extract from the FTD corpus via double fine-tuning.

2.3 Confidence of prediction

To estimate robustness of our predictions at the inference stage, we use a *confidence* metric, which equals $1 - uncertainty$, as introduced in (Tsymbalov et al., 2020). The key idea is that a classifier can only be confident on a text example if it predicts the correct labels for texts with *similar* embeddings. If the genre predicted by a classifier is not the same for many texts in the neighbourhood of an original text, we cannot consider the classifier confident. To simulate variation in embeddings, we apply dropout with probability 0.1 to all of the model layers, including the embedding layer and the final dense classification layer (Sun et al., 2019). A softmax classifier returns a probability distribution of the possible text labels. Normally, the label with the maximal probability is considered the answer of a classifier. Application of dropout to the classifier disturbs the probability distribution. We perform dropout n times, and thereby we generate the corresponding probability distributions $p_1, ..., p_n$. Then we pool them into a single distribution $\widehat{p}$. The maximum value of probability in $\widehat{p}$ is called *the confidence of prediction*. The intuition of this metric is that if a classifier is confident in the predicted text label, it is unlikely to distribute the likelihood on other class labels significantly often. In our study, we use $n = 10$ - the same as used by the authors of (Tsymbalov et al.,

[2] https://babylonbee.com/ – Babylon Bee
[3] http://panorama.pub – Panorama

Genre short	Genre label	Prototypes	EN		RU	
			train	test	train	test
A1	Argument	Argumentative blogs or opinion pieces	276	77	207	77
A4	Fiction	Novels, myths, songs, film plots	69	28	62	23
A7	Instruction	Tutorials, FAQs, manuals	141	50	59	17
A8	News	Reporting newswires	114	37	379	103
A9	Legal	Laws, contracts, terms&conditions	56	17	69	13
A11	Personal	Diary entries, travel blogs	72	19	126	49
A12	Promotion	Adverts, promotional postings	218	66	222	85
A14	Academic	Academic research papers	59	23	144	49
A16	Information	Encyclopedic articles, definitions, specifications	131	38	72	33
A17	Review	Reviews of products or experiences	48	22	107	34
	Total		10341	3288	1447	483

Table 1: Training and testing corpora FTD

2020), since this provides a good balance between assessing the confidence value and the speed of computation.

With this estimate we can calculate the confidence value for our models in order to identify texts for which the prediction is less reliable. We can either refuse to classify such texts or we can replace such predictions with the majority class (this is *opinion* for all languages in this dataset). The experiments in which we use the replacement are marked with the *majority* in the tables. We also tried different thresholds for the confidence estimate. For each model, we select a threshold value among the confidence percentiles calculated based on the dev set, e.g., by refusing to classify 10% of least reliable texts.

2.4 Other parameters

UP upsampling the training data in order to tackle the class inbalance; the most popular class is *opinion*, hence we upsample the texts of classes *satire* and *reporting* to make the number of texts of these classes more or equal to the number of opinion texts.

E the number of epochs; we vary it from 1 to 3, because it would be unpractical to use greater values on such a small training dataset. Even E3 leads to overfitting.

Model	f1 macro	f1 micro
en mBERT E2	0.156	0.265
en mBERT up E2	0.287	0.301
en mono E2	0.181	0.181
en mono up E2	0.228	0.337
en mBERT up E2	0.287	0.301
en mBERT up FTD E2	0.243	0.349
en XLM-R up E2	0.224	0.313
en XLM-R up FTD E2	0.271	0.386
en mBERT up E2	0.287	0.301
en mBERT up BB E2	0.202	0.301
en mBERT up + GN E2	0.400	0.614
en XLM-R up E2	0.224	0.313
en XLM-R up + GN E1	0.278	0.265
FI XLM-R up	0.297	0.422
FI XLM-R up + GN 2E	0.227	0.325
en XLM-R up E2	0.224	0.313
FI XLM-R up E2	0.297	0.422
ru+en XLM-R up E2	0.299	0.422
en mBERT up + GN E2	0.401	0.614
en mBERT up + GN 2K E2	0.339	0.482
FI mBERT up + GN E2	0.419	0.458
en mBERT up E2	0.287	0.301
all en mBERT up E2	0.344	0.361
en mBERT up + GN E1	0.349	0.494
en mBERT up + GN E2	0.401	0.614
en mBERT up + GN E3	0.127	0.217
mBERT-l + GN E2	0.422	0.639
Ens[0.7 FI mBERT up + GN, 0.3 FI XLM-R up]	0.434	0.470
Ens[0.65 FI mBERT up + GN, 0.35 FI XLM-R up]	0.437	0.470
Ens[0.65 FI mBERT up + GN, 0.35 FI XLM-R up] + major perc 10%	0.454	0.566
Ens[0.65 FI mBERT up + GN, 0.35 FI XLM-R up] + major perc 20%	**0.487**	**0.602**

Table 2: Metrics on the English dev set for Subtask 1

Model	f1 macro	f1 micro
fr mono	0.658	0.796
fr mono up	0.718	0.778
fr mBERT	0.513	0.778
fr mBERT up	0.495	0.426
fr XLM-R up	0.262	0.648
fr+it+en+finetune fr mBERT + GN E2	0.420	0.704
fr+it+en XLM-R	0.462	0.685
all fr XLM-R up	0.477	0.556
fr mBERT FTD E2	0.499	0.741
fr+it XLM-R	0.515	0.593
romance all fr up	0.519	0.630
all + XLM-R GN E2	0.534	0.796
fr+it+en+finetune fr XLM-R	0.541	0.815
fr+it+en mBERT	0.553	0.704
fr+it+en XLM-R + GN E2	0.566	0.833
fr+it+en+finetune fr mBERT	0.571	0.722
all + sharoff news mBERT	0.580	0.593
fr+it+en mBERT + GN E2	0.584	0.667
fr XLM-R FTD E2	0.601	0.722
all fr XLM-R FTD E2	0.632	0.778
all fr mBERT up	0.691	0.833
Ens[0.9 fr mono up, 0.1 fr+it+en+finetune fr XLM-R]	0.751	0.815
Ens[0.7 fr mono up, 0.3 fr+it+en+finetune fr XLM-R]	0.808	0.852
Ens[0.8 fr mono up, 0.2 fr+it+en+finetune fr XLM-R]	**0.826**	**0.870**

Table 3: Metrics on the French dev set for Subtask 1

Model	f1 macro	f1 micro
lXLM-R	0.551	0.711
de mBERT	0.563	0.733
de mBERT upsampled	0.600	0.600
all de XLM-R upsampled	0.594	0.600
all de mBERT up	0.601	0.578
de mBERT up ftd	0.642	0.689
all de XLM-R FTD	0.616	0.711
de XLM-R up ftd	0.657	0.756
all de mBERT up	0.601	0.578
all + GN mBERT	0.622	0.667
all de XLM-R FTD	0.616	0.711
all + GN XLM-R	0.640	0.733
Ens[0.6 de mono up, 0.4 all de XLM-R up]	0.631	0.689
Ens[0.7 de mono up, 0.3 all de xlm-r up]	0.676	0.711
ge mono up	0.687	0.711
Ens[0.9 de mono up, 0.1 all de xlm-r up]	0.687	0.711
Ens[0.7 de mono up, 0.3 all de mBERT up]	0.706	0.733
Ens[0.9 de mono up, 0.1 all de mBERT up]	0.726	0.756
deE	**0.772**	**0.800**
Ens[0.8 deE, 0.2 de mono up]	0.716	0.756
Ens[0.9 deE, 0.1 de mono up]	0.772	0.800

Table 4: Metrics on the German dev set for Subtask 1

Model	f1 macro	f1 micro
fr+it+en bert funetune it GN E2	0.365	0.779
it mBERT	0.372	0.792
all it mBERT up	0.406	0.506
it mBERT up	0.408	0.766
romance all it up	0.443	0.545
all XLM-R GN E2	0.443	0.792
fr+it+en XLM-R	0.446	0.714
fr+it+en XLM-R GN E2	0.451	0.792
it mBERT FTD	0.470	0.779
fr+it+en mBERT	0.473	0.727
it mono up	0.502	0.649
fr+it+en+finetune it mBERT	0.514	0.753
all mBERT GN E2	0.518	0.688
fr+it+en+finetune it XLM-R	0.527	0.779
fr+it+en mBERT GN E2	0.548	0.675
fr+it XLM-R	0.564	0.662
all it XLM-R FTD	0.588	0.753
it XLM-R FTD	0.706	0.857
all it xlm-roberta upsampled	**0.652**	**0.805**
Ens[0.1 all it XLM-R up, 0.9 all it mBERT up]	0.469	0.727
Ens[0.7 all it XLM-R up, 0.3 all it mBERT up]	0.642	0.792
Ens[0.9 all it XLM-R up, 0.1 all it mBERT up]	**0.652**	**0.805**

Table 5: Metrics on the Italian dev set for Subtask 1

Model	f1 macro	f1 micro
pl XLM-R	0.481	0.700
pl XLM-R up	0.496	0.740
pl mBERT up	0.537	0.780
pl mBERT	0.636	0.800
pl mono	0.805	0.860
pl mono up	0.857	0.900
all pl XLM-R up	0.715	0.760
all pl XLM-R FTD up	0.787	0.820
pl XLM-R up	0.496	0.740
pl XLM-R FTD up	0.789	0.860
pl mBERT up	0.537	0.780
pl mBERT up FTD	0.641	0.780
pl Slav	0.367	0.400
all Slav	0.633	0.640
pl mBERT up	0.537	0.780
all mBERT up	0.687	0.800
all mBERT + GN up	0.679	0.780
all mBERT up	0.687	0.800
all pl XLM-R up	0.715	0.760
all XLM-R + GN up	0.771	0.840
Ens[0.9 pl mono up, 0.1 all pl XLM-R up]	0.844	0.880
Ens[0.8 pl mono up, 0.2 all pl XLM-R up]	**0.872**	**0.900**
Ens[0.7 pl mono up, 0.3 all pl XLM-R up]	0.857	0.900

Table 6: Metrics on the Polish dev set for Subtask 1

Model	f1 macro	f1 micro
ru XLM-R up E2	0.363	0.347
ru XLM-R E2	0.464	0.694
ru mono up E2	0.472	0.755
ru mono E2	0.502	0.755
ru mBERT E2	0.471	0.714
ru mBERT up E2	0.578	0.633
ru XLM-R up E2	0.363	0.347
ru XLM-R up FTD E2	0.650	0.694
ru mBERT up E2	0.578	0.633
ru mBERT up FTD E2	0.768	0.837
ru mono up E2	0.472	0.755
ru mono up FTD E2	0.657	0.735
ru+en mBERT up FTD E2	0.426	0.653
ru+en+ru mBERT up FTD E2	0.458	0.694
ru XLM-R genres up FTD E2	0.650	0.694
ru+en XLM-R up FTD E2	0.507	0.776
all XLM-R up E2	0.466	0.551
all + GN XLM-R up E2	0.480	0.735
all mBERT up E2		
all + GN mBERT up E2	0.442	0.653
XLM-R up E2	0.363	0.347
XLM-R up + P E2	0.604	0.653
XLM-R up FTD E2	0.497	0.735
XLM-R up FTD + P E2	0.487	0.714
ru mBERT up FTD E2	**0.768**	**0.837**
ru mBERT up FTD majority 10% E2	0.762	0.837
ru mBERT up FTD majority 25% E2	0.725	0.816
Ens[0.7 * mBERT up FTD, 0.3 * ruBert up FTD]	0.721	0.776
Ens[0.9 * mBERT up FTD, 0.1 * ruBert up FTD]	0.735	0.796

Table 7: Metrics on the Russian dev set for Subtask 1

2.5 Results

For all the languages Tables 2, 3, 4, 5, 6, 7 (the codes are as in the description above) show that upsampling of a limited data sample is crucial for this subtask. The confidence estimates for rejecting unreliable predictions were found to be the most important for the English dataset, which is likely to be the noisiest one.

We find that for most pairs (language, model) the optimal number of epochs is 2. It is more or less consistent with the findings of Sun et al. (2019). In that work, the optimal number of epochs is between 2 and 4, depending on the amount of the training data. The learning rate and other hyperparameters are taken from (Sun et al., 2019) since they seem to be universal and there is no need to select them for each pair (language, model) separately.

The results for the English language Table 2 show that sometimes using ensembles is beneficial for the target metrics. In addition, we try to replace the predictions of low confidence with the major class. And it results in a significant growth of the f1 macro score on the dev set.

Table 7 shows results for the Russian models on the Subtask 1. The best result on the dev set was achieved with application of a mBERT first fine-tuned on the Russian FTD and further fine-tuned on the Russian data for Subtask 1. Our attempts to use ensembles for Russian did not improve the results. For this language, we also try to replace the predictions of low confidence with the major class. As a potential threshold we select various percentiles of the confidence on the dev set. We try different values of the hyperparameters but none of them helps to improve the metrics of the best model. When using the majority class with a threshold by confidence of prediction, we get f1 macro score 0.329. If we use the max probability instead of confidence, we get 0.310. It shows that confidence of prediction is a more reliable metric that max probability.

For both English and Russian, the best result was obtained with a multilingual model. We can make a similar conclusion for the French and Italian languages. For Polish Table 6 and German Table 4, the results are different. The best model for German is monolingual German Electra. The second best model is monolingual German BERT, which is likely to be related to the importance of language-specific tokenisation for the German compounds. But sometimes, the best result can be achieved with

an ensemble of a multilingual model and a monolingual one. It works well for Polish.

For the Polish language, we can see it clearly that the best ensemble on the dev set attains accuracy that does not differs much from that of the most accurate individual model, i.e. Polish monolingual BERT. On the test set, f1 macro for the monolingual Polish BERT is 0.737, while the accuracy for the ensemble is 0.786. Such a big gap confirms our assumption that ensembles are in general more reliable than its individual components.

By default, we use the base configurations of the pre-trained models for all the models and languages. We have also tried larger configurations (BERT large, XLM-R Large, RuBERT large) for English and Russian but they do not manage to improve over the base versions.

To understand the cases on which a classifier often makes mistakes, it is useful to look at the confusion matrix. We train XLM-R on concatenation of the French, Italian, and English texts and apply it to the English dev dataset. The model without additional training data (Table 8) appears to be biased in favor of the opinion texts. The model corrected by adding Giga-News data still shows a bias towards the opinion texts but in a much softer way, thus allowing for some recognition of satire in the dev set.

The confusion matrix for the best French model 9 seems to be better balanced with respect to recognition of other genres. The same is true for most of the languages apart from English, which might indicate possible biases in the English dataset.

2.6 Surprise languages

In the shared task, there are three surprise languages - Georgian, Greek and Spanish, - for which the train and dev sets are unavailable. These languages are added to the shared task in order to promote language-agnostic approaches.

For these languages, we take a multilingual BERT and train it on concatenation of the train data for all the available languages. This simple technique reaches the 3rd place on the Georgian language and the 5th place on Greek on the test set.

3 Subtask 2

In this subtask, we use the architectures that show the best results on Subtask 1 among the individual models. It is mBERT for Russian, English, French and Italian, Polish BERT for Polish, German Electra for German. Unlike in Subtask 1, we have not fine-tuned the models on the additional data.

In multilabel classification it is crucial to select the optimal thresholds for the classes. We experimented with a range of thresholds from 0.1 to 0.6 and selected the best one on the dev set. In order to simplify selection of the optimal hyperparameters, we use the same threshold all the labels. However, of course, it could be potentially improved by choosing different thresholds for each label. We found out the following threshold to be the best: 0.3 - for Italian, French and Russian, 0.2 - for Polish and German, 0.1 - for English.

4 Subtask 3

We use the best architectures pre-trained on Subtask 1 and fine-tune them further on the training set for Subtask 3. It improves the f1 score compared to the baseline, though the difference is not dramatic. For instance, the macro f1 score for the English language rises from 0.22 to 0.25.

The process of selection of the optimal thresholds is identical to that for Subtask 2.

5 Conclusion

In our study, we have tried multiple techniques including

1. usage of additional relevant corpora for fine-tuning,
2. usage of BERT-based classifiers pre-trained on a different classification task,
3. computing of confidence of prediction and replacing the predictions with low condifence,
4. ensembles of classifiers.

The available training corpora were not sufficient. When an additional relevant corpus is available (either with additional data in a specific genre, such as news, or a general purpose corpus, such as FTD), it always helps. We show that upsampling also improves the metrics for every language in the dataset. This implies that in tasks with few labeled texts available it is crucial for training data to be balanced. Another important conclusion is that for most languages multilingual models are better than monolingual models. The only exceptions are German and Polish. We also show that for most languages (4 out of 6) ensembles give higher performance than the individual models.

True/predicted	reporting	opinion	satire
reporting	0.370	0.630	0.000
opinion	0.250	0.750	0.000
satire	0.111	0.889	0.000

True/predicted	reporting	opinion	satire
reporting	0.444	0.463	0.093
opinion	0.250	0.500	0.250
satire	0.222	0.333	0.444

Table 8: Confusion matrices for English without adding GN (left) and with GN (right)

True/predicted	reporting	opinion	satire
reporting	0.800	0.200	0.000
opinion	0.057	0.914	0.029
satire	0.000	0.250	0.750

Table 9: Confusion matrix for the best French model

We show that there is no technique or approach that works well for each language. It makes the classification task more complicated and less language-agnostic. There is also suspicion of language-specific variation in data, such as comparatively low scores of all of the models obtained for English, while the same models are successful for other languages.

References

Christopher Cieri and Mark Liberman. 2002. Language resources creation and distribution at the Linguistic Data Consortium. In *Proc LREC*, pages 1327–1333. Las Palmas, Spain.

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

Mikhail Lepekhin and Serge Sharoff. 2022. Estimating confidence of predictions of individual classifiers and their ensembles for the genre classification task. *arXiv preprint arXiv:2206.07427*.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert? *arXiv preprint arXiv:1906.01502*.

Jakub Piskorski, Nicolas Stefanovitch, Giovanni Da San Martino, and Preslav Nakov. 2023. Semeval-2023 task 3: Detecting the category, the framing, and the persuasion techniques in online news in a multi-lingual setup. In *Proceedings of the 17th International Workshop on Semantic Evaluation*, SemEval 2023, Toronto, Canada.

Marina Santini, Alexander Mehler, and Serge Sharoff. 2010. Riding the rough waves of genre on the web. In Alexander Mehler, Serge Sharoff, and Marina Santini, editors, *Genres on the Web: Computational Models and Empirical Studies*. Springer, Berlin/New York.

Serge Sharoff. 2018. Functional text dimensions for the annotation of Web corpora. *Corpora*, 13(1):65–95.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune BERT for text classification? *arXiv preprint arXiv:1905.05583*.

Evgenii Tsymbalov, Kirill Fedyanin, and Maxim Panov. 2020. Dropout strikes back: Improved uncertainty estimation via diversity sampling. *arXiv preprint arXiv:2003.03274s*.