OpenFact at CheckThat! 2024: Cross-Lingual Transfer Learning for Check-Worthiness Detection

Notebook for the CheckThat! Lab at CLEF 2024

Marcin Sawiński[1,*], Krzysztof Węcel[1] and Ewelina Księżniak[1]

[1]Department of Information Systems, Poznań University of Economics and Business, Al. Niepodległości 10, 61-875 Poznań, Poland

Abstract

This paper presents the results of the OpenFact team's experiments in the CLEF 2024 CheckThat! Lab Task 1 competition for multilingual, unimodal check-worthiness detection. Several mono- and multilingual pre-trained language models were fine-tuned using different variants of the training datasets. Cross-lingual transfer learning was applied without instance transfer and proved to be effective for Arabic and Dutch. Additionally, we tested the effectiveness of class balancing using several under-sampling methods, which, when combined with appropriate model selection and cross-lingual transfer learning, produced the second-best results for Arabic and English.

Keywords

check-worthiness, fact-checking, fake news detection, language models, cross-lingual transfer learning, BERT

1. Introduction

Check-worthiness detection refers to the process of determining which statements should be fact-checked based on their potential influence and the probability of being incorrect. This paper describes the experiments conducted as part of the preparations for the CheckThat! Lab, Task 1 for Arabic, English, and Dutch at CLEF 2024, where the task was framed as a binary text classification problem.

The study predominantly investigated the effectiveness of cross-lingual transfer learning applied through multilingual pre-trained language models and optimal dataset preparation. The comparison of the performance of multilingual pre-trained language models versus monolingual models revealed that multilingual models can perform equally well or even outperform monolingual models when fine-tuned on monolingual training datasets, and additionally improve performance when fine-tuned with multilingual datasets. Our results for check-worthiness detection at CheckThat! Lab in 2023 showed a significant impact of dataset sampling. Previous experiments demonstrated that the under-sampling method, which boosted the performance of a fine-tuned GPT-3 model from the F1 score of 0.826 to 0.898 for the positive class, did not consistently yield the same improvements for BERT models. This study collected more observations with the aim of analyzing this phenomenon.

The dataset preparation experiments focused on attempts to improve above random under-sampling by introducing additional methods based on training dynamics [1]. The experiments resulted in creating check-worthiness detection methods that were ranked as the second-best for Arabic and English on the leaderboard for CheckThat! Lab, Task 1 in 2024.[1]

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France

*Corresponding author.

†These authors contributed equally.

marcin.sawinski@ue.poznan.pl (M. Sawiński); krzysztof.wecel@ue.poznan.pl (K. Węcel); ewelina.ksiezniak@ue.poznan.pl (E. Księżniak)

https://kie.ue.poznan.pl/en/ (M. Sawiński)

0000-0002-1226-4850 (M. Sawiński); 0000-0001-5641-3160 (K. Węcel); 0000-0003-1953-8014 (E. Księżniak)

[1]https://checkthat.gitlab.io/clef2024/task1/

2. Related Work

In previous editions of CheckThat! Lab [2][3], many methods were proposed to solve the check-worthiness detection task on text data. In 2023, the dominant method involved the application of pre-trained language models fine-tuned for the classification task.

For English, the best score was achieved by team OpenFact [4], using the GPT-3 curie model fine-tuned on an under-sampled training dataset. However, DeBERTa V3 performed only marginally worse. Under-sampling was performed using an additional annotation quality flag derived from the ClaimBuster dataset[2].

Other teams used monoligual models: BERT (Fraunhofer SIT [5], CSECU-DSG [6]), RoBERTa (Accenture [7]), GigaBERT (Accenture), MARBERT (ES-VRAI [8]), a feed-forward neural network trained on embeddings (Z-Index [9]), and multilingual models: XLM-RoBERTa (DSHacker [10]), Twitter XLM-RoBERTa (CSECU-DSG). The models were mostly used for sequence classification but other methods were also applied: ensemble learning with model souping (Fraunhofer SIT [5]), BiLSTM module handle long-term contextual dependency and multisample dropout(CSECU-DSG [6]).

The dataset curation included back-translation (Accenture [7]), under-sampling (ES-VRAI [8], OpenFact [4]), instance transfer (DSHacker [10]), paraphrasing with GPT-3.5 (DSHacker [10]).

We drew the conclusion that complex model setups were not critical to achieving the best results and that a well-performing BERT-family model could achieve top results provided a sufficient dataset. Another observation was that dataset augmentations, despite showing improvements over the baseline, might be outperformed by under-sampling. The last finding from the analysis of previous submissions was that multilingual models could perform equally well or better than single-language models.

A survey on offensive language detection [11], a task that share some similarities to check-worthiness detection, presents many options for leveraging domain knowledge from high-resource languages to low-resource languages by using Cross-Lingual Transfer Learning (CLTL). The first category of CLTL, *Instance Transfer*, includes the transfer of text or label information between source and target languages. In *Text-Based Transfer* (applied by DSHacker and Accenture in 2023), machine translation is most often used. For the purposes of this research, neither *Label-Based Transfer* (annotation projection and pseudo-labeling) nor *Text Alignment* methods are relevant because the data for all languages included in the competition, although scarce, come with labels. Next category, *Feature Transfer* methods extract linguistic features from source and target languages (e.g., using Multilingual Word Embeddings) and align them into a shared feature space. Those methods are applicable for the check-worthy detection task, but they were not used for experiments. *Parameter Transfer* relies on transferring distributions of parameters between languages within one model or across separate models. Multilingual pre-trained language models are fundamental for this method, as they are pre-trained on vast datasets in many languages, sharing semantic representations across languages.

We decided to focus our experiments on this CLTL method to analyze the performance of multilingual models fine-tuned on the multilingual datasets provided by the CheckThat! Lab organizers.

3. Methodology

The study focused on application of cross-lingual transfer learning for finding the best performing solution for check-worthiness detection in Arabic, Dutch, and English. Specific research questions were formulated:

- RQ1. What was the contribution to the final score of specific features of the ClaimBuster 1:2 dataset used to create the best-performing method in the 2023 CheckThat! Lab Task 1b?
- RQ2. How effective are multilingual pre-trained language models compared to monolingual models?

[2] https://zenodo.org/record/3836810

- RQ3. How can cross-lingual transfer be leveraged to improve check-worthiness detection using training data in multiple languages?
- RQ4. Is it possible to outperform random under-sampling with methods informed by annotation quality or training dynamics?

The first research question stems from the uncertainty surrounding the root causes of effectiveness of dataset curation applied in the winning method for English in the 2023 CheckThat! Lab Task 1b. The dataset reduced the class imbalance but did not completely eliminate it (with a 1:2 ratio of positive to negative examples), and some lower quality examples were filtered out. We observed inconsistent impact during the training process: some models produced much better results (e.g., the F1 score of winning method - fine-tuned GPT-3 curie increased by 0.072), while others remained unchanged or even worsened. The experiments were planned to isolate the impact of class balancing, removal of low-quality examples, and variability arising from random model parameter initialization. The second research question aims at measuring the gap between monolingual and multilingual models, highlighting any potential performance loss when using the latter.

The third research question focuses on utilizing cross-lingual transfer not only for low-resource languages but also for improving high-resource languages by combining data from around the globe. Check-worthiness detection is part of the fact-checking process, which in many cases is global. Fake news and narratives cross geographical and language barriers. Consequently, models trained on multilingual data could potentially outperform monolingual models, even for high-resource languages.

The goal of the fourth research question is to design proxy measures that would allow for the creation of a high-quality training dataset even when explicit annotation quality feature is not available.

The study contains three parts:

1. Finding the best monolingual model to use as a baseline.
2. Preparing multilingual training dataset variants.
3. Training and evaluating mono- and multilingual models on the prepared datasets.

The study required multiple model training runs for various models, dataset preparation variants, and different random seeds to allow for more accurate comparisons of results. Each training was evaluated using the loss metric or the F1 score metric for the positive class, and tested using the F1 score metric for the positive class.

Phases of the experiments included:

1. Testing single language models using unaltered datasets.
2. Testing cross-lingual transfer learning using various concatenations of datasets.
3. Testing the impact of various structural changes to the training datasets.

Our team achieved the best score in CheckThat! Lab Subtask-1B in English in 2023 [4] using a fine-tuned GPT-3 model; however, results obtained by the DeBERTa V3 model were only marginally worse. Considering that the end goal of check-worthiness detection is large-scale application, resource consumption is a critical factor for the actual method selection. Given the significantly lower resources needed to run BERT models compared to GPT-3, we decided to limit this study to BERT models and to maximize the model performance within this constraint.

4. Models

We made an initial selection of BERT models for the experiments to use for the *sequence classification* task. We were not able to test all available models and we have not been able to establish an objectively verifiable ranking list. Instead we decided to include selected mono- and multilingual models. The subjective selection was based on preference for largest, most recent, or the best performing models according to benchmarks or previous editions of CLEF CheckThat! Lab.

For the English subtask, we tested two English models:

- DeBERTa V3 base (`microsoft/deberta-v3-base`),[3]
- DeBERTa V3 large (`microsoft/deberta-v3-large`).[4]

DeBERTa V3 base scored 0.894 in CheckThat! Lab Subtask-1B in English in 2023 [4], only 0.004 less than the winning GPT-3 but still 0.006 better than the second team [12]. Adding a larger version of the same model was expected to yield even better results.

For the Arabic subtask, we tested three variants of CAMeLBERT, choosing the best-suited model for the dataset – Modern Standard Arabic (MSA), dialectal Arabic (DA), and classical Arabic (CA):

- CAMeLBERT MSA (`CAMeL-Lab/bert-base-arabic-camelbert-msa`),[5]
- CAMeLBERT DA (`CAMeL-Lab/bert-base-arabic-camelbert-da`),
- CAMeLBERT CA (`CAMeL-Lab/bert-base-arabic-camelbert-ca`).

For the Dutch subtask, we selected two models:

- RobBERT 2023 large (`DTAI-KULeuven/robbert-2023-dutch-large`),[6]
- BERTje (`GroNLP/bert-base-dutch-cased`).[7]

Results from CheckThat! Lab Subtask-1B [3] indicated that multilingual models also have the potential to achieve top results. We decided to include two multilingual models in our experiments:

- mDeBERTa V3 base (`microsoft/mdeberta-v3-base`),[8]
- XLM-RoBERTa base (`FacebookAI/xlm-roberta-base`).[9]

Due to time and resource constraints, we were not able to extensively search for optimal hyperparameter values. We decided to use preselected values and tested multiple variants of the training dataset. We monitored the learning curves to ensure that the models did not under-fit and applied early stopping to avoid overfitting. We used step-wise evaluation strategy instead of epochs with 5000 maximum steps.

5. Datasets

5.1. Datasets Overview

CheckThat! Lab in 2024 provided participants with four datasets in Arabic, Dutch, English, and Spanish. Each dataset contained *train*, *dev*, and *dev_test* splits. For Arabic, Dutch, and English, a *test* split was also provided for use in submission.

The count of examples revealed that the Dutch dataset contained significantly fewer examples than the others (see Table 1) and that the positive class is underrepresented in all datasets (see Table 2). Results in previous editions of CheckThat! Lab inspired us to explore various sampling methods informed by data quality and training dynamics measures.

English Dataset. Analysis revealed that examples in the *train* and *dev* splits originated from the ClaimBuster dataset. A lookup on ClaimBuster files indicated that the *train* data split was fully annotated by crowd-sourcing, while the *dev* split was annotated by experts (the so-called ground-truth dataset in ClaimBuster). The *dev_test* split was equal to the *test* split delivered in the 2023 edition of CheckThat! Lab, but its origins are unknown. The *test* split was not matched with any existing dataset.

Arabic, Dutch, Spanish Datasets. The data structure revealed that examples were collected from Twitter, but the datasets were not matched with any existing datasets.

[3] https://huggingface.co/microsoft/deberta-v3-base
[4] https://huggingface.co/microsoft/deberta-v3-large
[5] https://huggingface.co/CAMeL-Lab/bert-base-arabic-camelbert-msa
[6] https://huggingface.co/DTAI-KULeuven/robbert-2023-dutch-large
[7] https://huggingface.co/GroNLP/bert-base-dutch-cased
[8] https://huggingface.co/microsoft/mdeberta-v3-base
[9] https://huggingface.co/FacebookAI/xlm-roberta-base

Table 1
Number of examples per language and dataset split

Language	*train*	*train*	*dev_test*	*test*	Total
Arabic	7333	1093	500	610	9536
Dutch	995	252	666	1000	2913
English	22501	1032	318	341	24192
Spanish	19948	5000	5000	-	29948
Total	50777	7377	6484	1951	66589

Table 2
Positive class ratios per language and dataset split

Language	*train*	*train*	*dev_test*	*test*	Total
Arabic	0.31	0.38	0.75	0.36	0.34
Dutch	0.41	0.40	0.47	0.40	0.42
English	0.24	0.23	0.34	0.26	0.24
Spanish	0.16	0.14	0.10	-	0.14
Total	0.22	0.20	0.20	0.36	0.22

5.2. Dataset Variants

5.2.1. Monolingual Dataset Variants

In the first phase, the original dataset splits were used to train language-specific models. Three main baseline variants of datasets were:

- Arabic *train*,
- Dutch *train*,
- English *train*.

For evaluation, the original *dev* dataset splits were used, and *dev_test* splits were used to calculate the F1 score (positive class) of each trained model.

5.2.2. Multilingual Dataset Variants

In the second phase, we planned experiments with cross-lingual transfer learning using six multilingual *train* datasets. New dataset variants were created by concatenating the *train* splits of the single language datasets. Similarly, the *dev* splits were concatenated to create multilingual evaluation datasets. All *dev_test* splits were used individually to calculate the F1 score. The concatenation variants included:

- **Full multilingual** – concatenation of Arabic, Dutch, English, and Spanish (later referred to as *ar+en+es+nl*).
- **Twitter multilingual** – concatenation of Arabic, Dutch, and Spanish (later referred to as *ar+es+nl*).
- **Twitter bilingual** – concatenation of Arabic and Dutch (later referred to as *ar+nl*).

We noticed a significant disproportion in the size of the train datasets: Dutch (995 examples) vs Arabic (7333), English (22501), and Spanish (19948). To address this issue, we created over-sampled versions of the datasets with Dutch examples sampled three times for *ar+nl(x3)* and *ar+es+nl(x3)*, and five times for *ar+en+es+nl(x5)*.

5.2.3. Filtering by Annotation Quality, Correctness, and Random Under-Sampling

Previously, we observed a significant improvement from balancing class counts, so we added a train dataset variant with random under-sampling applied. Another variant involved reshuffling examples in the *train* and *dev* splits before applying random under-sampling.

Our previous research [4] showed that for English, the annotation quality differed between *train* (crowd-sourced labels) and *dev* (ground-truth annotated by experts), and this difference could impact the training process. For English, the aim of reshuffling was to test if adding some higher quality examples to the *train* set from *dev*, combined with adding some lower quality examples to *dev* from *train*, would affect the results. The preparation process consisted of three steps:

1. Concatenation of *train* and *dev* splits into a single dataset.
2. Random split into new *train* and *dev* subsets with an 8:2 ratio.
3. Random under-sampling of the new *train* and *dev* sets to achieve equal class counts.

As a result, three sets of dataset variants were created: *Original* (full training datasets), *RUS* (random under-sampling applied), and *RUS & new split* (random under-sampling applied after joining *train* and *dev* and splitting again).

The total number of available training datasets for cross-lingual transfer learning was 30 (4 monolingual datasets and 6 multilingual, each in *Original*, *RUS*, and *RUS & new split* versions). Not all variants were used in experiments due to resource considerations and potential improvements in results.

In the third phase, seven additional *train* dataset variants were created for the English dataset. Leveraging additional information about annotation quality derived from the ClaimBuster dataset, individual examples were assigned a High or Low quality flag.

The authors of the ClaimBuster dataset, used for creating the English *train* dataset, introduced screening criteria to exclude low-quality labels and published three filtered datasets with class ratios of 1:2, 1:2.5, and 1:3[10]. The most balanced, 1:2 dataset was used directly in the experiment (referred to as *High Quality 1:2*). Additionally, we derived a new *High Quality* flag that was assigned to all examples included in any of the three mentioned ClaimBuster datasets. Analogously, examples not included in any of the aforementioned datasets were flagged as *Low Quality*. On top of that, we used a separate flag for *Ground Truth* indicating examples annotated by experts, while all other examples were annotated using a *crowd-sourcing* approach.

As a result, eight English train datasets based on quality were created and later referred to as:

- **Original** – Unmodified English train dataset from CheckThat! Lab 2024.
- **Ground Truth** – Selected examples annotated by experts.
- **High Quality** – Examples included in ClaimBuster files screened for quality.
- **Low Quality** – Examples excluded from ClaimBuster files screened for quality.
- **Original and GT (Ground Truth)** – Concatenation of 0.8 of Original and Ground Truth examples (0.2 hold-out for evaluation).
- **High Quality and GT (Ground Truth)** – Concatenation of 0.8 of High Quality and Ground Truth examples (0.2 hold-out for evaluation).
- **Low Quality and GT (Ground Truth)** – Concatenation of 0.8 of Low Quality and Ground Truth examples (0.2 hold-out for evaluation).
- **High Quality 1:2** - 0.8 of examples included in the ClaimBuster 1:2 file (0.2 hold-out for evaluation).

Additionally, we trained the DeBERTa V3 base model on all examples (concatenated *train* and *dev*) for 5 epochs and collected logits after each epoch to calculate training dynamics metrics: variability, confidence, and correctness as described by Swayamdipta et al. [1]. We used the correctness measure to further filter the data: examples were classified as correct (correctness equal to five) or not (correctness

[10] https://zenodo.org/record/3836810

less than five). The correctness flag was used to generate an additional set of train datasets by removing examples with correctness less than five.

As a final step, we applied random under-sampling (Random US, *RUS*) to all 16 datasets (eight splits by quality times two by correctness equal to five flag), producing 32 new final datasets (the order of filtering was quality > correctness > random under-sampling).

5.2.4. Additional Under-Sampling Methods

We created additional dataset variants using under-sampling methods informed by additional measures. These variants were assigned the following codes:

- *DUS* – Symmetrically removing the most *easy-to-learn* and *hard-to-learn* examples. All majority class examples were sorted in descending order by their ℓ_2 distance from the reference point (*variability*, *confidence*)=(0.5, 0.5) and removed until the desired class count was reached.
- *HUS* – First removing all *hard-to-learn* examples (defined as examples having an ℓ_2 distance from (*variability*, *confidence*)=(0.5, 0.5) greater than 0.35 while having a confidence < 0.5), and then removing *easy-to-learn* examples sorted by descending distance from (*variability*, *confidence*)=(0.5, 0.5) until the desired class count was reached.
- *CUS* – First removing all examples from the majority class with correctness less than five, and later, if necessary, randomly choosing examples with correctness equal to five until the desired class count was reached.

The calculation formulas (variability, confidence, and correctness) and definitions of regions (*easy-to-learn*, *hard-to-learn*, *ambiguous*) follow [1]. The results were compared to the original dataset (*Original*) and random under-sampling (*RUS*).

6. Experimental Results

6.1. Monolingual Model Selection

Several training runs for the Arabic dataset revealed that the MSA variant of the CAMeLBERT family of models is best suited for the task. The best F1 score (Positive Class) was 0.832 with a learning rate of 1e-05, and this configuration was used for other experiments (see Figure 1).

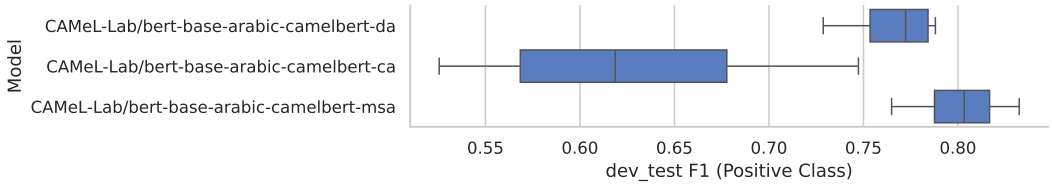

Figure 1: Results of hyperparameter sweep for Arabic models - F1 score (Positive Class).

Training runs for the Dutch dataset revealed that the RobBERT 2023 large model outperformed the BERTje model for the given task. The best F1 score (Positive Class) was 0.671 with a learning rate of 1e-05; however, we decided to include both models in other experiments (see Figure 2).

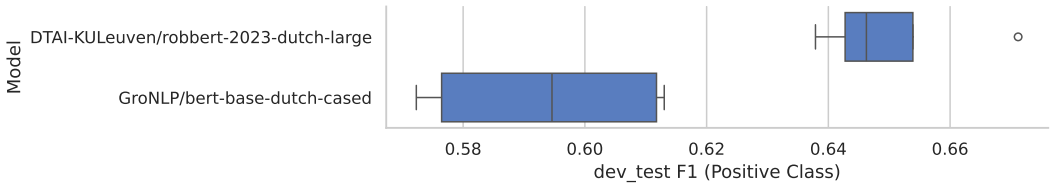

Figure 2: Results of hyperparameter sweep for Dutch models - F1 score (Positive Class).

Training runs for the English dataset revealed that the DeBERTa V3 large model outperformed the DeBERTa V3 base model for the task. The best F1 score (Positive Class) was 0.926 with a learning rate of 1e-05. We decided to mainly use the DeBERTa V3 large model for other experiments; however, we made some further comparisons with the base model as well (see Figure 3).

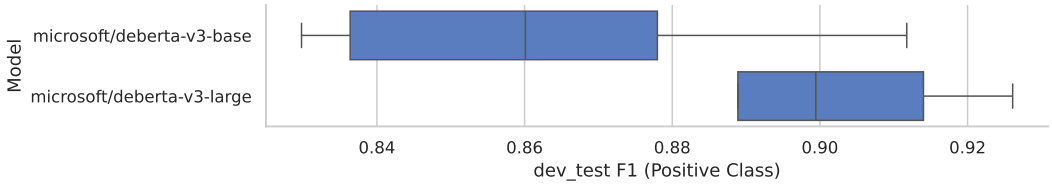

Figure 3: Results of hyperparameter sweep for English models - F1 score (Positive Class)

6.2. Cross-Lingual Transfer Learning

Experiments for cross-lingual transfer learning followed a similar pattern for all languages. We first tested and compared the performance of monolingual model training on full datasets (*Original*), Random Under-Sampling (*RUS*), and Random Under-Sampling with joined and new splits of *train* and *dev* (*RUS & new split*).

In most cases, the F1 score was higher for *RUS* and *RUS & new split*, so we dropped some of the *Original* variants for subsequent study to save on compute power. Analysis of the performance of training the monolingual models shows that the results for English (see Table 5) were significantly higher than for Arabic and Dutch (0.932 vs 0.873 and 0.671 respectively — see Tables 3 and 4).

For cross-lingual transfer, we decided to test multilingual models trained solely on the English dataset to predict on *dev_test* datasets in Arabic and Dutch. Bearing in mind resource utilization, we excluded other possibilities (e.g., testing predictions for English using a model trained solely in Arabic or Dutch) as this was not likely to improve the F1 score.

The remaining combinations of dataset variants, models, and sampling methods were applied in model training. We planned four runs with different random seeds for each combination but, due to compute constraints, not all seed values were tested. The result tables present the highest F1 score achieved (*max*) and the mean (*mean*) calculated from multiple runs of the same configuration with different random seed values.

6.2.1. Arabic

For Arabic, the highest F1 score for positive class was achieved by the mDeBERTa V3 base model trained on the largest dataset, which concatenated Arabic, Dutch, English, and Spanish datasets after applying random under-sampling on individual datasets and over-sampling Dutch data five times (*ar+en+es+nl(x5)*). The maximum F1 score was 0.901, with the mean F1 score from all runs only slightly lower at 0.894. It surpassed the best monolingual Arabic model by 0.028 for the maximum and 0.042 for the mean F1 score (0.873 and 0.852 for CAMeLBERT MSA, see Table 3). It is worth noting that using

only Arabic data for training the multilingual model also produced a higher F1 score than the dedicated Arabic model (0.021 for maximum and 0.025 for mean).

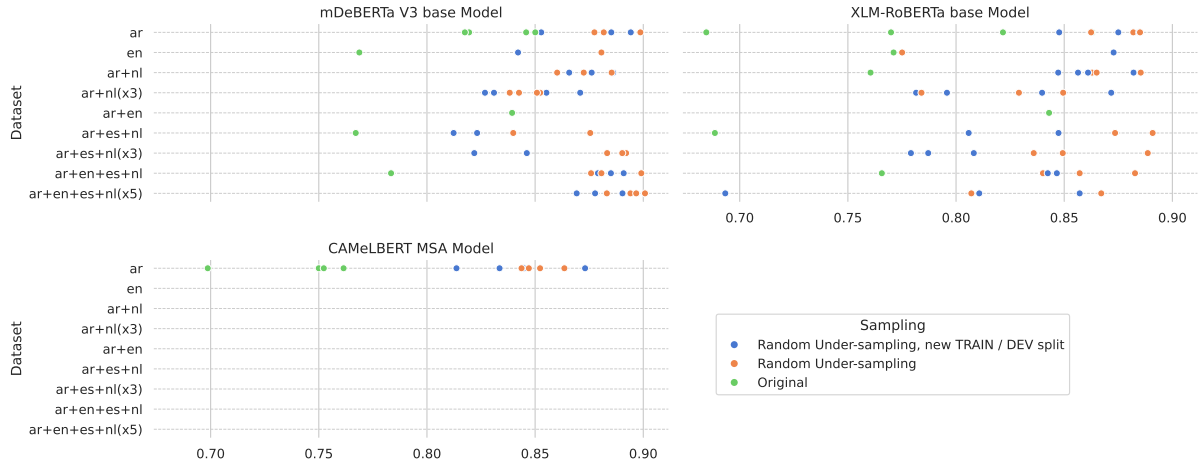

Figure 4: Results of Cross-Lingual Transfer experiments - x-axis presents F1 score (Positive Class) for the Arabic *dev_test* dataset.

6.2.2. Dutch

For Dutch, the highest F1 score (Positive Class) was also achieved by the mDeBERTa V3 base model, but the optimal training dataset was different. The best performance was achieved using a random under-sampled and reshuffled *train* and *dev* (*RUS & new split*) dataset, concatenated with Arabic and Dutch data (*ar+nl*). This configuration provided the model with the optimal training examples and resulted in both the highest maximum F1 score of 0.714 and the highest mean of all runs at 0.684. Surprisingly, adding more data (English, Spanish) or over-sampling Dutch examples lowered the F1 score. In this case, cross-lingual transfer surpassed the best monolingual model by 0.036 for the maximum and 0.016 for the mean F1 score (0.678 and 0.668 for RobBERT 2023 large, see Table 4). It is worth noting that using only Dutch data for training the multilingual model yielded lower results than dedicated Dutch models (0.017 for maximum and 0.018 for mean).

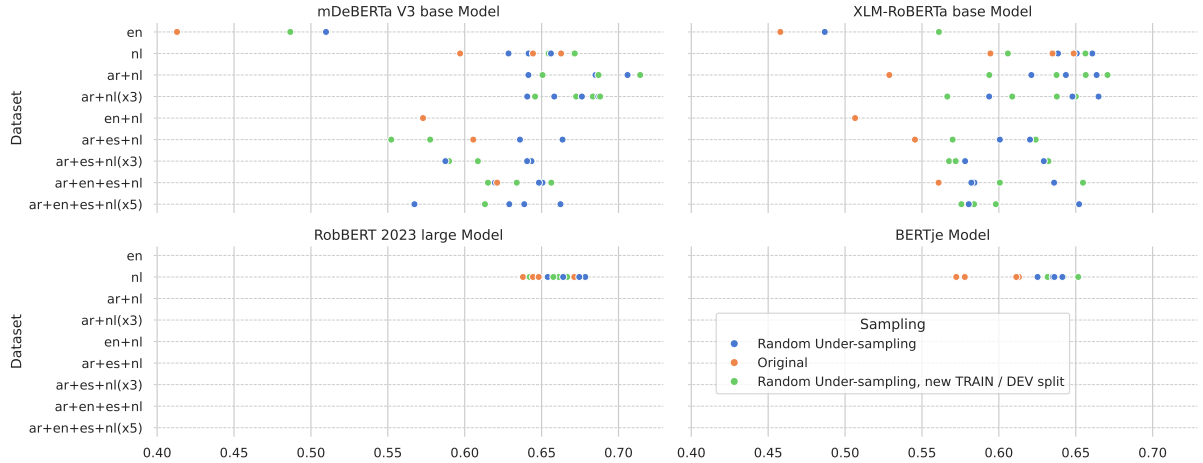

Figure 5: Results of Cross-Lingual Transfer experiments - x-axis presents F1 score (Positive Class) for Dutch *dev_test* dataset.

6.2.3. English

For English, the single highest F1 score (Positive Class) was achieved by the monolingual DeBERTa V3 large on the randomly under-sampled English dataset (0.932), but the highest mean of all runs was equal for both DeBERTa V3 large and multilingual mDeBERTa V3 base (0.899). Both results were achieved using only English examples in training. Cross-lingual transfer was not effective in this case (see Table 5) in the appendix.

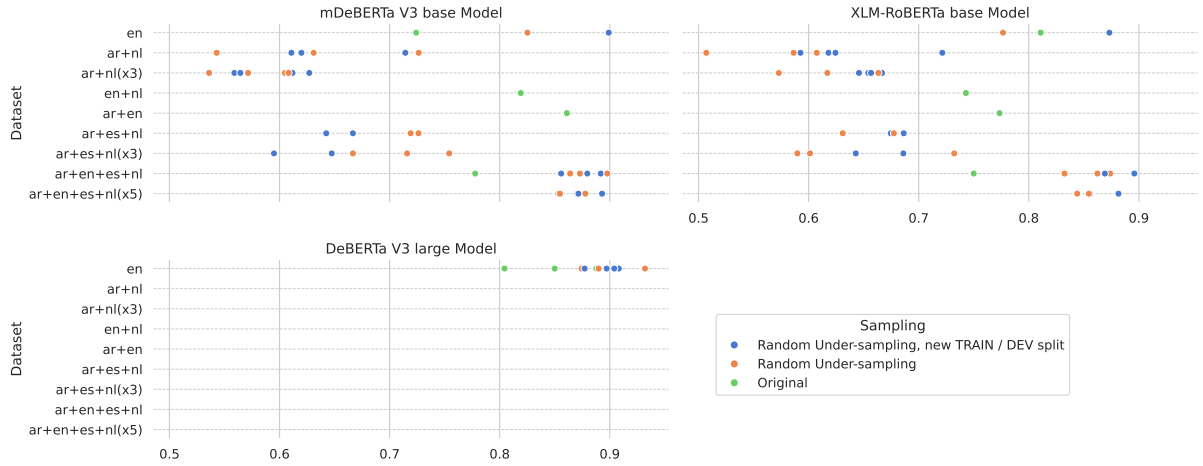

Figure 6: Results of Cross-Lingual Transfer experiments - x-axis presents F1 score (Positive Class) for English *dev_test* dataset.

6.3. Filtering by Quality and Correctness

Experiments performed on English dataset variants, filtered by annotation quality and under-sampled, showed a greater impact of class balancing over structural changes. While filtering by annotation quality was able to improve the F1 score compared to the original dataset, the improvements from class balancing were much more pronounced.

The best overall score was achieved by DeBERTa V3 large with random under-sampling on the original dataset, achieving the highest maximum score of 0.95 and a mean score of 0.939. The highest maximum score without random under-sampling was also achieved by DeBERTa V3 large on the original dataset. The highest mean score without random under-sampling was 0.90, achieved by the same model using the High Quality 1:2 dataset. It is important to note that this dataset is more balanced than the original dataset (1:2 vs 1:3.17).

Random under-sampling combined with filtering of examples with correctness less than five produced worse results than random under-sampling alone (see Figure 7). Complete results are presented in Table 6 in the appendix.

6.4. Additional Under-Sampling Methods

Experiments with under-sampling methods continued after submission to the CheckThat! Lab 2024 competition, and many training runs were performed when the test file with labels was already available. In contrast to previously reported results, this experiment reports the F1 score (Positive Class) on both *dev_test* and *test* datasets.

The application of filtering by quality and correctness did not yield improvements when applied as the first step of the processing pipeline before random under-sampling. In this phase of the experiment, the processing order was changed: all minority (positive) class examples were included in all training runs, and only the majority (negative) class examples were filtered out based on various conditions (referred to as *RUS*, *QUS*, *DUS*, *HUS*, and *CUS*, see Section 5.2.4).

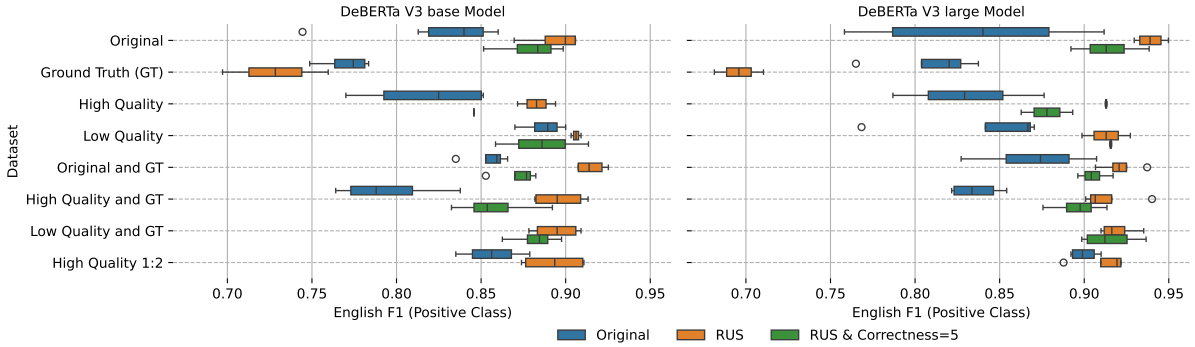

Figure 7: Results of Cross-Lingual Transfer experiments - F1 score (Positive Class) for English *train* dataset tested on *dev_test* dataset.

The distribution of the results in this experiment varied from the previous one (6.3) due to changes in hyper-parameter values; nevertheless, similar patterns emerged.

For Arabic, models trained on datasets with random under-sampling outperformed models trained on the original dataset when the F1 score was measured against *dev_test*. This was not true when measured against *test*. Random under-sampling (*RUS*) performed slightly worse than the *Original* dataset; however, the use of correctness improved results on average. The differences, however, were insignificant (0.001 to 0.002 difference between *Original* mean and *CUS* mean F1 score). Complete results are presented in Table 7 in the appendix.

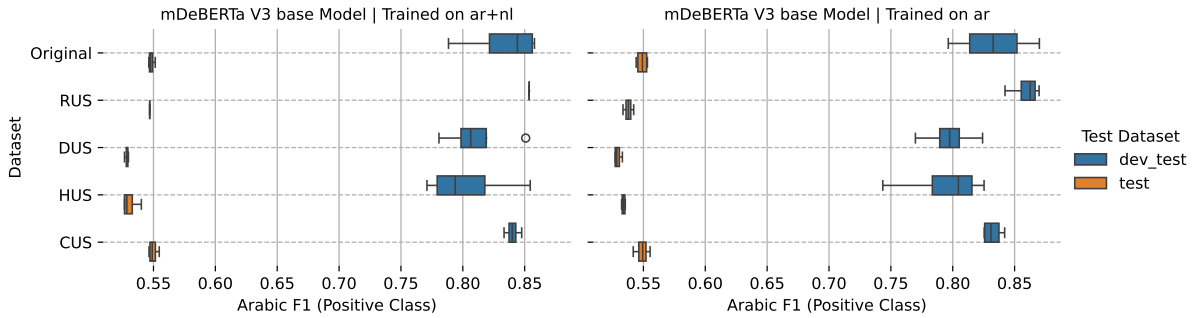

Figure 8: Results of under-sampling experiments - F1 score (Positive Class) for Arabic *dev_test* dataset.

For Dutch, we did not observe a systematic improvement from under-sampling. Similar to the experiment on training the English model on the Ground Truth portion of data (similar in size to the Dutch dataset, approximately 1,000 examples, see Section 5.2.4), any further reduction lowered the F1 score. Complete results are presented in Table 8 in the appendix.

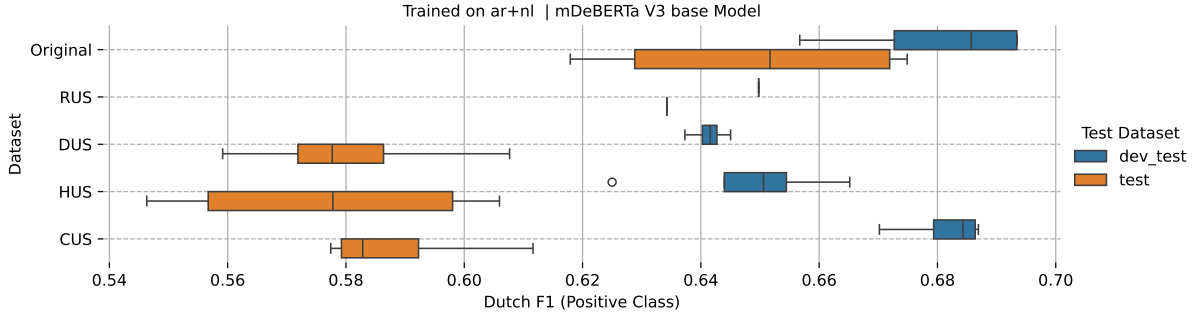

Figure 9: Results of under-sampling experiments - F1 score (Positive Class) for Dutch *dev_test* dataset.

For English, models trained on datasets with random under-sampling outperformed models trained on the original dataset when comparing both *dev_test* and *test* F1 scores. An even higher increase in F1 scores was observed when under-sampling was performed based on annotation quality criteria (*QUS*). The highest maximum F1 score with DeBERTa V3 base was 0.942, with a mean of 0.9 (a 0.03 and 0.035 increase versus the *Original* baseline). This contrasts with the quality-based filtering experiment results. Unfortunately, the *DUS*, *HUS*, and *CUS* methods generated mostly inferior results (see Figure 10). Complete results are presented in Table 9 in the appendix.

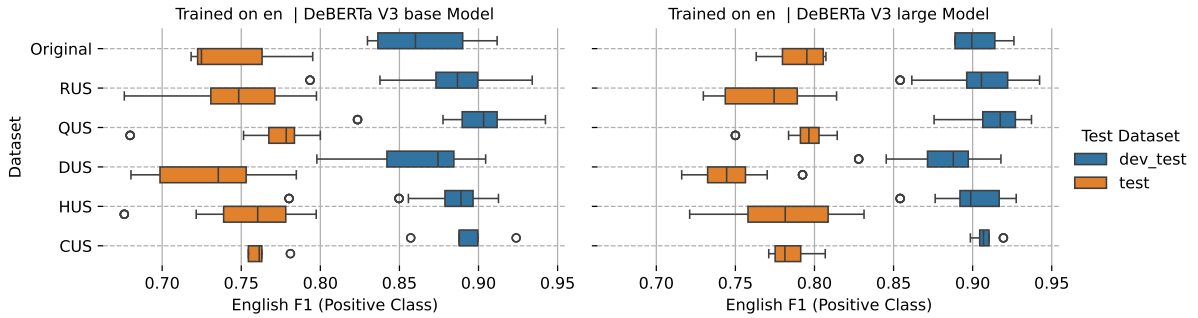

Figure 10: Results of under-sampling experiments - F1 score (Positive Class) for English *dev_test* dataset.

6.5. Result Submission

The following set-ups were used for result submission:

- For Arabic, we submitted results generated by the mDeBERTa V3 base model, trained on a randomly under-sampled and concatenated dataset comprising Arabic, Dutch, English, and Spanish training data.
- For Dutch, we submitted results generated by the mDeBERTa V3 base model, trained on a randomly under-sampled and concatenated dataset comprising Arabic and Dutch training data.
- For English, we submitted results generated by the DeBERTa V3 large model. The preparation of the training dataset included concatenation of the train and dev datasets, followed by a split in an 8:2 ratio and subsequent under-sampling. The annotation quality features derived from the ClaimBuster dataset were not used for training the model chosen for submission.

7. Conclusions and Future Work

Application of cross-lingual transfer learning allowed us to achieve a 0.557 F1 score for Arabic, securing second place on the leaderboard. Conversely, for Dutch, the method achieved a 0.590 F1 score, placing only seventh in the competition. For English, we submitted predictions generated with a monolingual model trained on a randomly under-sampled dataset and achieved an F1 score of 0.796, earning second place on the leaderboard.

The results of the conducted experiments shed light on the research questions.

RQ1. What was the contribution to the final score of specific features of the dataset used to create the best-performing method in the 2023 CheckThat! Lab Task 1b?

The best results achieved in the CheckThat! Lab 2023 for English, using a ClaimBuster 1:2 dataset, can be attributed to addressing the class imbalance problem rather than purely the quality of annotation.

RQ2. How effective are multilingual pre-trained language models compared to monolingual models?

We demonstrated the efficacy of multilingual models in classification tasks. The results were comparable to or better than those of dedicated monolingual models, even when fine-tuned on a single-language training dataset.

RQ3. How can cross-lingual transfer be leveraged to improve check-worthiness detection using training data in multiple languages?

In the case of the Arabic and Dutch subtasks, training on concatenated multilingual datasets led to superior results. The English dataset, on its own, was sufficient to train the best model.

RQ4. Is it possible to outperform random under-sampling with methods informed by annotation quality or training dynamics?

Although the removal of lower-quality examples did not contribute to improvements in the F1 score, the inclusion of the annotation quality feature in the under-sampling process has the potential to outperform random under-sampling. An important limitation of application of annotation-quality under-sampling comes from availability of quality measure. An alternative was proposed based on model training dynamics. Three methods for enhancing under-sampling with measures calculated from model training dynamics did not outperform random under-sampling.

Despite the failure of the training dynamics measures proposed in this paper, we believe that future work should investigate other possibilities for defining measures to support the identification of mislabeled examples to inform dataset balancing methods.

Acknowledgments

The research is supported by the project "OpenFact – artificial intelligence tools for verification of veracity of information sources and fake news detection" (INFOSTRATEG-I/0035/2021-00), granted within the INFOSTRATEG I program of the National Center for Research and Development, under the topic: Verifying information sources and detecting fake news.

References

[1] S. Swayamdipta, R. Schwartz, N. Lourie, Y. Wang, H. Hajishirzi, N. A. Smith, Y. Choi, Dataset cartography: Mapping and diagnosing datasets with training dynamics, arXiv preprint arXiv:2009.10795 (2020).

[2] P. Nakov, A. Barrón-Cedeño, G. Da San Martino, F. Alam, M. Kutlu, W. Zaghouani, C. Li, S. Shaar, H. Mubarak, A. Nikolov, Overview of the clef-2022 checkthat! lab task 1 on identifying relevant claims in tweets, CLEF '2022, Bologna, Italy, 2022.

[3] A. Barrón-Cedeño, F. Alam, A. Galassi, G. Da San Martino, P. Nakov, T. Elsayed, D. Azizov, T. Caselli, G. S. Cheema, F. Haouari, M. Hasanain, M. Kutlu, C. Li, F. Ruggeri, J. M. Struß, W. Zaghouani, Overview of the clef–2023 checkthat! lab on checkworthiness, subjectivity, political bias, factuality, and authority of news articles and their source, in: A. Arampatzis, E. Kanoulas, T. Tsikrika, S. Vrochidis, A. Giachanou, D. Li, M. Aliannejadi, M. Vlachos, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction, Springer Nature Switzerland, Cham, 2023, pp. 251–275.

[4] M. Sawiński, K. Węcel, E. Księżniak, M. Stróżyna, W. Lewoniewski, P. Stolarski, W. Abramowicz, Openfact at checkthat! 2023: Head-to-head gpt vs. bert - a comparative study of transformers language models for the detection of check-worthy claims, in: Working Notes of CLEF 2023 - Conference and Labs of the Evaluation Forum, CLEF '2023, Thessaloniki, Greece, 2023.

[5] R. Frick, I. Vogel, I. Nunes Grieser, Fraunhofer sit at checkthat! 2022: semi-supervised ensemble classification for detecting check-worthy tweets, in: Working Notes of CLEF 2022–Conference and Labs of the Evaluation Forum, CLEF '2022, Bologna, Italy, 2022.

[6] A. Aziz, M. Hossain, A. Chy, Csecu-dsg at checkthat! 2023: transformer-based fusion approach for multimodal and multigenre check-worthiness, Working Notes of CLEF (2023).

[7] S. Tran, P. Rodrigues, B. Strauss, E. Williams, Accenture at checkthat! 2023: Identifying claims with societal impact using nlp data augmentation, Working Notes of CLEF (2023).

[8] H. T. Sadouk, F. Sebbak, H. E. Zekiri, Es-vrai at checkthat! 2023: Analyzing checkworthiness in multimodal and multigenre (2023).

[9] P. Tarannum, M. A. Hasan, F. Alam, S. R. H. Noori, Z-index at checkthat! 2023: Unimodal and multimodal checkworthiness classification, Working Notes of CLEF (2023).
[10] A. Modzelewski, W. Sosnowski, A. Wierzbicki, Dshacker at checkthat! 2023: Check-worthiness in multigenre and multilingual content with gpt-3.5 data augmentation, Working Notes of CLEF (2023).
[11] A. Jiang, A. Zubiaga, Cross-lingual offensive language detection: A systematic review of datasets, transfer approaches and challenges, 2024. arXiv:2401.09244.
[12] R. Frick, I. Vogel, J. Choi, Fraunhofer sit at checkthat! 2023: enhancing the detection of multimodal and multigenre check-worthiness using optical character recognition and model souping., in: Working Notes of CLEF 2023 - Conference and Labs of the Evaluation Forum, CLEF '2023, Thessaloniki, Greece, 2023.

Appendices

A. Cross-Lingual Transfer Learning

Table 3
Results for the Arabic Test Dataset measured with the F1 score (Positive Class) for models trained with full datasets (*Original*), Random Under-Sampling (*RUS*), and Random Under-Sampling with a combined and new split of *train* and *dev* (*RUS & new split*).

	Sampling	Original		RUS		RUS & new split	
Arabic – F1 (Positive Class)		max	mean	max	mean	max	mean
Model	Dataset						
mDeBERTa V3 base	ar	0.85	0.833	0.899	0.886	**0.894**	0.877
	en	0.769	0.769	0.881	0.881	0.842	0.842
	ar+nl	-	-	0.885	0.873	0.886	0.876
	ar+nl(x3)	-	-	0.852	0.846	0.871	0.846
	ar+en	0.839	0.839	-	-	-	-
	ar+es+nl	0.767	0.767	0.875	0.858	0.823	0.818
	ar+es+nl(x3)	-	-	0.892	0.889	0.846	0.834
	ar+en+es+nl	0.783	0.783	0.899	0.885	0.891	**0.885**
	ar+en+es+nl(x5)	-	-	**0.901**	**0.894**	0.89	0.879
XLM-RoBERTa base	ar	**0.863**	0.785	0.885	0.876	0.885	0.869
	en	0.771	0.771	0.775	0.775	0.873	0.873
	ar+nl	0.76	0.76	0.885	0.871	0.882	0.862
	ar+nl(x3)	-	-	0.85	0.821	0.872	0.822
	ar+en	0.843	**0.843**	-	-	-	-
	ar+es+nl	0.689	0.689	0.891	0.882	0.847	0.827
	ar+es+nl(x3)	-	-	0.889	0.858	0.808	0.791
	ar+en+es+nl	0.766	0.766	0.883	0.86	0.847	0.844
	ar+en+es+nl(x5)	-	-	0.867	0.837	0.857	0.787
CAMeLBERT MSA	ar	0.761	0.741	0.864	0.852	0.873	0.841

Table 4
Results for the Dutch Test Dataset measured with the F1 score (Positive Class) for models trained with full datasets (*Original*), Random Under-Sampling (*RUS*), and Random Under-Sampling with a combined and new split of *train* and *dev* (*RUS & new split*).

	Sampling	Original		RUS		RUS & new split	
Dutch – F1 (Positive Class)		max	mean	max	mean	max	mean
Model	Dataset						
mDeBERTa V3 base	en	0.413	0.413	0.51	0.51	0.487	0.487
	nl	0.663	0.636	0.656	0.642	0.672	0.666
	ar+nl	-	-	**0.706**	**0.677**	**0.714**	**0.684**
	ar+nl(x3)	-	-	0.687	0.665	0.688	0.672
	en+nl	0.573	0.573	-	-	-	-
	ar+es+nl	0.606	0.606	0.664	0.65	0.578	0.565
	ar+es+nl(x3)	-	-	0.643	0.624	0.609	0.599
	ar+en+es+nl	0.621	0.621	0.651	0.639	0.656	0.635
	ar+en+es+nl(x5)	-	-	0.662	0.624	0.629	0.619
XLM-RoBERTa base	en	0.458	0.458	0.487	0.487	0.561	0.561
	nl	0.649	0.629	0.661	0.65	0.656	0.631
	ar+nl	0.529	0.529	0.664	0.643	0.671	0.64
	ar+nl(x3)	-	-	0.665	0.635	0.65	0.616
	en+nl	0.507	0.507	-	-	-	-
	ar+es+nl	0.545	0.545	0.62	0.61	0.624	0.597
	ar+es+nl(x3)	-	-	0.629	0.593	0.632	0.591
	ar+en+es+nl	0.561	0.561	0.636	0.601	0.655	0.628
	ar+en+es+nl(x5)	-	-	0.652	0.616	0.598	0.586
RobBERT 2023 large	nl	**0.671**	**0.65**	0.678	0.668	0.667	0.657
BERTje	nl	0.613	0.594	0.641	0.634	0.652	0.639

Table 5
Results for the English Test Dataset measured with the F1 score (Positive Class) for models trained with full datasets (*Original*), Random Under-Sampling (*RUS*), and Random Under-Sampling with a combined and new split of *train* and *dev* (*RUS & new split*).

	Sampling	Original		RUS		RUS & new split	
English – F1 (Positive Class)		max	mean	max	mean	max	mean
Model	Dataset						
mDeBERTa V3 base	en	0.724	0.724	0.825	0.825	0.899	**0.899**
	ar+nl	-	-	0.726	0.633	0.714	0.648
	ar+nl(x3)	-	-	0.608	0.58	0.627	0.591
	en+nl	0.819	0.819	-	-	-	-
	ar+en	0.861	**0.861**	-	-	-	-
	ar+es+nl	-	-	0.726	0.723	0.667	0.655
	ar+es+nl(x3)	-	-	0.754	0.712	0.647	0.621
	ar+en+es+nl	0.778	0.778	0.898	0.878	0.892	0.876
	ar+en+es+nl(x5)	-	-	0.878	0.864	0.893	0.878
XLM-RoBERTa base	en	0.811	0.811	0.777	0.777	0.873	0.873
	ar+nl	-	-	0.607	0.567	0.721	0.639
	ar+nl(x3)	-	-	0.663	0.618	0.667	0.656
	en+nl	0.743	0.743	-	-	-	-
	ar+en	0.773	0.773	-	-	-	-
	ar+es+nl	-	-	0.677	0.654	0.686	0.68
	ar+es+nl(x3)	-	-	0.732	0.641	0.686	0.657
	ar+en+es+nl	0.75	0.75	0.874	0.856	0.896	0.883
	ar+en+es+nl(x5)	-	-	0.855	0.849	0.882	0.873
DeBERTa V3 large	en	**0.888**	0.848	**0.932**	**0.899**	**0.908**	0.897

B. Filtering by Quality and Correctness

Table 6
Results for the English dataset measured with the F1 score (Positive Class) for models trained with several pre-configured datasets filtered by quality and correctness.

	Data preparation	Original		RUS		RUS & Correctness=5	
		max	mean	max	mean	max	mean
Model	Dataset						
DeBERTa V3 base	Original	0.86	0.828	0.906	0.894	0.899	0.879
	Ground Truth (GT)	0.784	0.77	0.76	0.728	-	-
	High Quality	0.851	0.818	0.894	0.883	0.846	0.846
	Low Quality	0.9	0.887	0.909	0.906	0.913	0.886
	Original and GT	0.866	0.855	0.925	0.915	0.882	0.872
	High Quality and GT	0.838	0.794	0.913	0.896	0.892	0.858
	Low Quality and GT	-	-	0.909	0.894	0.898	0.882
	High Quality 1:2	0.879	0.857	0.911	0.893	-	-
DeBERTa V3 large	Original	**0.912**	0.836	**0.95**	**0.939**	**0.938**	0.914
	Ground Truth (GT)	0.837	0.811	0.71	0.696	-	-
	High Quality	0.876	0.83	0.913	0.913	0.893	0.878
	Low Quality	0.87	0.843	0.927	0.913	0.916	**0.916**
	Original and GT	0.907	0.871	0.937	0.921	0.917	0.905
	High Quality and GT	0.854	0.836	0.94	0.914	0.913	0.896
	Low Quality and GT	-	-	0.935	0.919	0.937	0.915
	High Quality 1:2	0.91	**0.9**	0.922	0.912	-	-

C. Additional Under-Sampling Methods

Table 7
F1 scores (Positive Class) for Arabic using the mDeBERTa V3 base model trained with different under-sampling methods.

F1 (Positive Class)		max		mean	
	Test Dataset	*dev_test*	*test*	*dev_test*	*test*
Train Language	Dataset				
ar	Original	**0.87**	0.553	0.833	0.549
	RUS	**0.87**	0.542	**0.859**	0.538
	DUS	0.824	0.533	0.797	0.529
	HUS	0.825	0.535	0.794	0.534
	CUS	0.842	**0.555**	0.832	0.549
ar+nl	Original	0.858	0.551	0.834	0.548
	RUS	0.854	0.547	0.854	0.547
	DUS	0.851	0.53	0.811	0.528
	HUS	0.854	0.54	0.803	0.531
	CUS	0.848	**0.555**	0.84	**0.55**

Table 8
F1 scores (Positive Class) for Dutch using the mDeBERTa V3 base model trained with different under-sampling methods on the *ar+nl* dataset.

F1 (Positive Class)	max		mean	
Test Dataset	*dev_test*	*test*	*dev_test*	*test*
Dataset				
Original	**0.693**	0.675	0.573	0.532
RUS	0.65	**0.707**	0.55	**0.663**
DUS	0.663	0.608	**0.646**	0.561
HUS	0.665	0.623	0.624	0.563
CUS	0.687	0.685	0.639	0.612

Table 9
F1 scores (Positive Class) for English using a model trained with different under-sampling methods on the *en* dataset.

F1 (Positive Class)		max		mean	
	Test Dataset	*dev_test*	*test*	*dev_test*	*test*
Model	Dataset				
DeBERTa V3 base	Original	0.912	0.795	0.865	0.749
	RUS	0.934	0.798	0.884	0.749
	QUS	**0.942**	0.8	0.9	0.773
	DUS	0.905	0.785	0.865	0.731
	HUS	0.913	0.797	0.882	0.756
	CUS	0.924	0.781	0.892	0.764
DeBERTa V3 large	Original	0.926	0.807	0.903	0.79
	RUS	**0.942**	0.814	0.905	0.771
	QUS	0.937	0.814	**0.915**	**0.795**
	DUS	0.918	0.792	0.883	0.746
	HUS	0.928	**0.831**	0.899	0.779
	CUS	0.919	0.807	0.908	0.785