

TOBB ETU at CheckThat! 2022: Detecting Attention-Worthy and Harmful Tweets and Check-Worthy Claims

Ahmet Bahadir Eyuboglu, Mustafa Bora Arslan, Ekrem Sonmezer and Mucahid Kutlu

TOBB University of Economics and Technology, Ankara, Turkey

Abstract

In this paper, we present our participation in CLEF 2022 CheckThat! Lab's Task 1 on detecting check-worthy and verifiable claims and attention-worthy and harmful tweets. We participated in all subtasks of Task1 for Arabic, Bulgarian, Dutch, English, and Turkish datasets. We investigate the impact of fine-tuning various transformer models and how to increase training data size using machine translation. We also use feed-forward networks with the Manifold Mixup regularization for the respective tasks. We are ranked first in detecting factual claims in Arabic and harmful tweets in Dutch. In addition, we are ranked second in detecting check-worthy claims in Arabic and Bulgarian.

Keywords

Fact-Checking, Check-worthiness, Attention-worthy tweets, Harmful tweets, Factual Claims

1. Introduction

Social media platforms became one of the main information resource for people by enabling their users to easily share messages and follow others. While these platforms are extremely important to help people share their thoughts and make their voice heard, they can be also used in a very negative way by spreading misinformation and/or hateful messages which will negatively impact individuals and societies. We have especially observed this dark side of social media platforms during COVID-19 pandemic. For instance, misinformation and conspiracy theories about vaccines increased hesitation towards being vaccinated [1]. Furthermore, the messages spread on social media platforms might impact public opinion on a particular issue and mobilize people, forcing government entities to take action. For instance, government entities of several countries had to regularly share information about vaccines to reduce the vaccine hesitation (e.g., [2]).


In this paper, we explain our participation in Task 1 [3] of the CLEF Check That! 2022 Lab [4, 5]. Task 1 covers four subtasks including 1) check-worthy claim detection (Subtask 1A), verifiable factual claim detection (Subtask 1B), harmful tweet detection (Subtask 1C), and attention-worthy tweet detection (Subtask 1D). Subtask 1A covers six languages including Arabic, Bulgarian, Dutch, English, Spanish, and Turkish while the other subtasks cover all the

CLEF 2022: Conference and Labs of the Evaluation Forum, September 5–8, 2022, Bologna, Italy

✉ ahmetbahadireyuboglu@gmail.com (A. B. Eyuboglu); mustafaboraarslan@outlook.com (M. B. Arslan); sonmezerekrem@outlook.com (E. Sonmezer); m.kutlu@etu.edu.tr (M. Kutlu)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

mentioned languages except Spanish. We participated in all subtasks for Arabic, Bulgarian, Dutch, English, and Turkish languages¹, yielding 20 submissions in total.

In the development phase of the shared task, we explored three different research directions including i) fine-tuning various pre-trained transformer models, ii) increasing the training data for fine-tuning transformer models, and iii) applying the Manifold Mixup regularization technique [6] for the subtasks we participated. In particular, we investigated 9, 3, 5, 13, and 3 different pre-trained transformer models for subtask 1A in Arabic, Bulgarian, Dutch, English, and Turkish, respectively. In addition, we explored increasing training data by back-translation and machine-translating datasets in other languages for subtask 1C. Next, we compared the Manifold Mixup approach, fine-tuning transformer models, and data augmentation by back-translation in all four subtasks to select models for our official submissions.

In our experiments with the development dataset, we find that the type of the transformer model causes dramatic changes in the performance, suggesting that researchers should select the models carefully. In addition, our findings about the impact of artificially increasing the data are mixed. In particular, we observe that increasing training data usually has a negative impact in Bulgarian and Turkish datasets in subtask 1C while using additional data for English and Dutch datasets improves the performance.

In the official ranking, we achieved mixed results. Considering tasks with at least three participants, we are ranked first in 1B-Arabic and second in 1A-Arabic and 1A-Bulgarian. We share our implementation for the Manifold Mixup method² for reproducibility of our results.

2. Approaches

We explore three different approaches for all subtasks including fine-tuning various transformer models, increasing dataset size via machine translation, and the Manifold Mixup regularization. In this section we explain each of them in detail.

2.1. Fine Tuning Various Transformer Models

Prior works show remarkable success of transformer models in various text classification tasks [7]. Furthermore, the best-performing systems in previous check-worthy claim detection tasks of Check That! Lab [8] usually exploited various transformer models [9, 10]. However, Kartal and Kutlu [11] show that the performance of models varies dramatically across different transformer models. Therefore, in this approach, we explore several language-specific transformer models pre-trained with different datasets.

2.2. Increasing Training Data via Machine Translation

Training data has enormous impact on the performance of resultant models. Prior work on detecting check-worthy claim detection investigated several ways to increase the training data size such as back-translation [9], weak supervision [12], and utilizing datasets in other languages with multi-lingual models [11]. In this approach, we explore increasing training data size by

¹We could not participate for Spanish due to a technical problem we encountered during development.

²<https://github.com/Carnegie/manifold-mixup-text-classification>

two different methods including 1) utilizing datasets in other languages by machine-translating them into the respective language, and 2) paraphrasing the training data via back-translation and using them as additional labeled data.

In the first method, we exploit datasets in several languages provided by the Check That! Lab organizers this year. In particular, in order to develop a model for a specific language, L_O , we first select a training dataset provided for another language and machine-translate its tweets to the language L_O using Google Translate. Subsequently, we fine-tune a language-specific transformer model using the original data and machine-translated data together. In subtask 1C, we machine translate only tweets labeled as harmful to reduce the imbalance in label distribution while increasing the training data size.

In our back-translation method, we first translate the original text to another language using Google Translate. Subsequently, we translate the resultant text back to the original language. This method is likely to create slightly different texts than the original ones with a same or similar meaning. Assuming that the change in the texts will not affect their label, we combine the original data with the back-translated data and fine-tune a language specific transformer model.

2.3. Language Specific BERT with Manifold Mixup

Many of the annotations in the shared task are subjective. For instance, whether a tweet requires attention of government entities might depend on how much the annotators want governments to intervene their life. Similarly, prior work on check-worthiness points out the subjective nature of the task (e.g., [11, 13]) In order to focus on this problem, we apply the Manifold Mixup regularization proposed by Verma et al. [6]. In particular, the Manifold Mixup trains neural networks on linear combinations of hidden representations of training examples, yielding flattened class-representations and smoother decision boundaries. Verma et al. [6] demonstrate that their approach yields more robust solutions in image classification. In our work, we use BERT embeddings to represent tweets and then train a four-layer feed-forward network with the Manifold Mixup method.

In subtask 1-D, we apply a different approach than the other tasks due to its severely imbalanced label distribution. In particular, there are nine labels in subtask 1-D, but eight of them are about why a particular tweet is attention-worthy. In addition, the majority of the tweets have “not attention-worthy” label. Therefore, we first binarize labels by merging variants of attention-worthy labels into a single one, yielding only two labels: 1) attention-worthy and 2) not-attention-worthy. Subsequently, we under-sample negative class with the 1/5 ratio and train our Manifold Mixup model. Next, we build another model using eight labels for attention-worthy tweets. If a tweet is classified as attention-worthy, we use the second model to predict why it is attention-worthy. Otherwise, we do not use the second model and label it as “not attention-worthy”. Note that we do not apply this two-step approach for other subtasks because they are already binary classification tasks.

3. Experiments

We first present statistics about the datasets and explain implementation details and our experimental setup in Section 3.1. Next, we explain how we selected our submissions in Section 3.2. Finally, we present the results of our submissions in Section 3.3.

3.1. Experimental Setup

3.1.1. Implementation

In order to fine-tune and configure transformer models, we use PyTorch v.1.9.0³ and Tensorflow⁴ libraries. We import transformer models used in our experiments from Huggingface⁵. In addition, we use Google’s SentencePiece library for machine translation⁶. We set the batch size to 32 in all our experiments with fine-tuned transformer models. In experiments on increasing dataset size using machine translation, we train the models for 5 epochs.

We implemented the Manifold Mixup [6] method from scratch using PyTorch v.1.9.0, and set epoch and the batch size to 5 and 2, respectively. We use the following transformer models for each language: AraBERT.v02 [14] for Arabic, RoBERTa-base-bulgarian⁷ for Bulgarian, RobBERT [15] for Dutch, the uncased version of BERT-base⁸ for English, and DistilBERTTurk⁹ for Turkish.

3.1.2. Evaluation Metrics

We use the official metric for each subtask to evaluate and compare our methods. In particular, we use F_1 score of positive class in subtasks 1A and 1C, accuracy in subtask 1B, and weighted F_1 in subtask 1D.

3.1.3. Datasets

The shared task organizers provide train, development, test development, and test datasets for each language and subtask. The number of tweets for each label in train, development, test development, and test datasets in subtasks 1A, 1B, 1C, and 1D are presented in **Table 1, 2, 3, and 4**, respectively.

In our experiments during the development phase, we use the train and development datasets for training and validation of the Manifold Mixup model, respectively. In our experiments for fine-tuning various transformer models and increasing dataset size via machine translation, we combine train and development sets for each case and fine-tune models accordingly. In all experiments during the development phase, we use the development test dataset for testing.

³<https://pytorch.org/>

⁴<https://www.tensorflow.org/>

⁵<https://huggingface.co/docs/transformers/index>

⁶<https://github.com/google/sentencepiece>

⁷<https://huggingface.co/iarfmoose/roberta-base-bulgarian>

⁸<https://huggingface.co/bert-base-uncased>

⁹<https://huggingface.co/dbmdz/distilbert-base-turkish-cased>

Table 1

Data & Label Distribution for Each Language in Subtask 1A.

Language	Label	Train	Dev.	Dev. Test	Test
English	not check-worthy	1675	151	445	110
	check-worthy	447	44	129	39
Bulgarian	not check-worthy	1493	141	413	73
	check-worthy	378	36	106	57
Dutch	not check-worthy	546	44	150	350
	check-worthy	377	28	102	316
Turkish	not check-worthy	1995	177	427	289
	check-worthy	422	45	84	14
Arabic	not check-worthy	1551	135	425	435
	check-worthy	962	100	266	247

Table 2

Data & Label Distribution for Each Language in Task 1B.

Language	Label	Train	Dev.	Dev. Test	Test
English	not claim	3031	276	828	102
	claim	292	31	82	149
Bulgarian	not claim	839	74	217	130
	claim	1871	177	519	199
Dutch	not claim	1021	109	282	750
	claim	929	72	252	608
Turkish	not claim	828	72	222	303
	claim	1589	150	438	209
Arabic	not claim	1118	104	305	682
	claim	2513	235	691	566

Table 3

Data & Label Distribution for Each Language in Task 1C.

Language	Label	Train	Dev.	Dev. Test	Test
English	not harmful	3031	276	828	211
	harmful	292	31	82	40
Bulgarian	not harmful	2341	209	636	314
	harmful	248	18	67	11
Dutch	not harmful	1775	165	476	1145
	harmful	171	14	55	215
Turkish	not harmful	1790	157	476	466
	harmful	627	65	174	46
Arabic	not harmful	2946	276	805	1011
	harmful	678	60	189	190

Table 4

Data & Label Distribution in Training (Tr), Development (D), Test Development (TD), and Test (T) Sets for Each Language in Subtask 1D.

Label	English				Bulgarian				Dutch				Turkish				Arabic			
	Tr	D	TD	T	Tr	D	TD	T	Tr	D	TD	T	Tr	D	TD	T	Tr	D	TD	T
not interesting	2851	267	774	202	2341	209	636	308	1545	142	405	1078	1698	151	466	429	1185	115	298	354
harmful	173	21	55	26	248	18	67	3	94	11	31	86	24	8	10	2	511	50	164	98
blame authorities	138	7	36	7	35	7	9	3	128	10	39	54	82	8	21	5	71	5	17	61
calls for action	48	3	12	4	4	1	3	1	27	5	11	22	15	1	5	4	36	6	19	53
discusses cure	42	3	15	5	56	12	11	8	5	1	2	13	38	5	14	6	1132	101	303	248
discusses action	27	1	7	4	17	2	6	3	23	1	8	42	21	1	6	11	501	42	152	250
contains advice	12	2	4	1	6	1	3	1	38	2	10	12	4	1	5	0	79	3	20	48
asks question	5	1	1	1	1	0	0	1	84	6	26	29	16	2	5	7	98	14	17	47
other	25	1	5	1	2	1	1	1	5	1	1	20	6	1	1	1	8	2	5	27

3.2. Experimental Results in the Development Phase

We participate in all subtasks of Task 1 for five languages, yielding 20 different submissions. In addition, we explore three different approaches to determine our final submissions. Therefore, in order to reduce the complexity of experiments and meet the deadlines of the shared task, we first evaluate using various transformer models and increasing training data size in subtask 1A and 1C, respectively, on the respective test development datasets. Next, based on our experiments in subtask 1A and 1C, we compare three different approaches in all subtasks to determine our submissions for the official evaluation on the test data. We note that this is not an ideal way to select systems for submission, but we take this step to meet the deadlines.

3.2.1. Impact of Transformer Model on Detecting Check-Worthy Claims

In order to observe the impact of transformer models, we identify several transformer models available on the Huggingface platform based on their monthly download scores and evaluate their performance in subtask 1A. The number of transformer models we compare is 9, 3, 5, 13, and 3 for Arabic, Bulgarian, Dutch, English, and Turkish, respectively.

We present the results in **Table 5**. Our observations based on our extensive experiments are as follows. Firstly, the results for English show the importance of evaluation metric to report the performance of systems. For instance, *distilroberta-base-climate-f* has the worst recall and F_1 scores, but achieves the best accuracy. Secondly, our results suggest that the text used in pre-training has a major impact on the models' performance. For instance, *COVID-Twitter-BERT v1* achieves the best F_1 score among all English models. This should be because it is pretrained with tweets about COVID-19 while the tweets used in the shared task are also about COVID-19. Similarly, *PubMedBERT*, which is pretrained with research articles on PubMed, yields the second best results for English. However, we also observe some unexpected results in our experiments. For instance, *AraBERT.v1*, which is pre-trained on a smaller dataset compared to other variants of AraBERT (i.e., *AraBERTv0.2-Twitter*, *AraBERTv0.2*, and *AraBERTv2*), outperforms all Arabic specific models. In addition, while *DarijaBERT* is pre-trained with only texts in Moroccan

Arabic, it outperforms all other Arabic specific models except *AraBERT.v1*. Furthermore, the best performing model in the Turkish dataset is the one with the smallest vocabulary size. Therefore, our results show that it is not easy to determine a pre-trained model by just comparing models' configurations and texts used in pre-training. We think that one of the reasons for having these unexpected results is the subjective nature of the task [11].

Table 5

Results of Various Transformer Models in Detecting Check-Worthy Claims. For each language the best-performing case is shown in **bold**.

	Model	Accuracy	Precision	Recall	F_1
Arabic	AraBERT.v1 [14]	0.413	0.390	0.932	0.550
	DarijaBERT ¹⁰	0.499	0.420	0.789	0.548
	Ara_DialectBERT ¹¹	0.431	0.393	0.887	0.545
	arabert_c19 [16]	0.548	0.439	0.627	0.517
	AraBERTv0.2-Twitter [14]	0.600	0.482	0.526	0.503
	bert-base-arabic [17]	0.481	0.397	0.672	0.5
	CAMeLBERTE [18]	0.451	0.372	0.620	0.465
	bert-base-arabertv2 ¹²	0.534	0.399	0.417	0.408
	bert-base-arabertv02 ¹³	0.599	0.454	0.206	0.284
Bulg.	RoBERTa-base-bulgarian ⁷	0.776	0.451	0.443	0.447
	RoBERTa-small-bulgarian-POS ¹⁴	0.485	0.259	0.820	0.394
	bert-base-bg-cased [19]	0.784	0.448	0.245	0.317
Dutch	BERTje [20]	0.619	0.516	0.941	0.666
	RobBERT [15]	0.650	0.549	0.764	0.639
	bert-base-nl-cased ¹⁵	0.559	0.469	0.676	0.554
	bert-base-dutch-cased-finetuned-gem ¹⁶	0.638	0.582	0.382	0.461
English	COVID-Twitter-BERT v1 [21]	0.721	0.434	0.798	0.562
	PubMedBERT [22]	0.745	0.447	0.558	0.496
	BERT base model (uncased) [7]	0.634	0.343	0.689	0.458
	LEGAL-BERT [23]	0.630	0.326	0.604	0.423
	ALBERT Base v2 [24]	0.689	0.353	0.457	0.398
	Bio_ClinicalBERT [25]	0.682	0.337	0.426	0.376
	BERT base model (cased) [7]	0.224	0.224	1.0	0.366
	bert-base-uncased-contracts ¹⁷	0.740	0.405	0.333	0.365
	ALBERT Base v1 ¹⁸	0.707	0.338	0.317	0.328
	hateBERT [26]	0.770	0.476	0.232	0.312
	COVID-Twitter-BERT v2 MNLI ¹⁹	0.667	0.265	0.271	0.268
	RoBERTa base [27]	0.731	0.295	0.139	0.189
	DistilRoBERTa-base-climate-f [28]	0.783	0.631	0.093	0.162
Turkish	BERTurk uncased 32K Vocabulary ²⁰	0.760	0.333	0.385	0.357
	BERTurk uncased 128K Vocabulary ²¹	0.337	0.188	0.859	0.309
	BERTurk cased 128K Vocabulary ²²	0.562	0.203	0.526	0.293

3.2.2. Impact of Training Data in Detecting Harmful Tweets

We use *roberta-small-bulgarian*²³ for Bulgarian, *BERTje* [20] for Dutch, *BERT-base-cased* for English, and *bert-base-turkish-sentiment-cased*²⁴ for Turkish as language-specific transformer models. **Table 6** shows the performance of each model when a different dataset is machine-translated to the corresponding language and respective language-specific model is fine-tuned with the original data and the machine-translated data. In this experiment, we are not able to report results for Arabic because we run into technical challenges (e.g., insufficient memory) preventing us to obtain results. We observe that increasing training data does not always improve the performance. In particular, using the original dataset for Turkish and Bulgarian yields the highest results while the performance of models usually increase in English and Dutch datasets by utilizing more labeled samples.

Table 6

Impact of increasing training data by machine-translating another dataset in a different language in detecting harmful tweets. We report F_1 score for each case. The best result for each language is written in **bold**.

Machine-Translated Data	Bulgarian	Dutch	English	Turkish
None	0.26	0.26	0.11	0.55
Bulgarian	-	0.39	0.23	0.13
Dutch	0.23	-	0.23	0.53
English	0.21	0.39	-	0.48
Turkish	0.19	0.25	0.25	-
Arabic	0.16	0.27	0.21	0.47

The subjective nature of this task might be one of the reasons for having lower performance by using additional data from other languages. In particular, as each country is dealing with different social issues, it is likely that people living in different countries might disagree on what makes a message harmful for a society. For instance, Turkish annotators might be more sensitive to tweets about refugees compared to annotators for other languages because Turkey hosts nearly 3.8 million refugees, i.e., the largest refugee population worldwide²⁵, and thereby, misinformation about refugees might have unpleasant consequences.

¹⁰<https://huggingface.co/Kamel/DarijaBERT>

¹¹https://huggingface.co/MutazYounes/Ara_DialectBERT

¹²<https://huggingface.co/aubmindlab/bert-base-arabertv2>

¹³<https://huggingface.co/aubmindlab/bert-base-arabertv02>

¹⁴<https://huggingface.co/iarfmoose/roberta-small-bulgarian-pos>

¹⁵<https://huggingface.co/Geotrend/distilbert-base-nl-cased>

¹⁶<https://huggingface.co/GeniusVoice/bert-base-dutch-cased-finetuned-gem>

¹⁷<https://huggingface.co/nlpaeub/bert-base-uncased-contracts>

¹⁸<https://huggingface.co/albert-base-v1>

¹⁹<https://huggingface.co/digitalepidemiologylab/covid-twitter-bert-v2-mnli>

²⁰<https://huggingface.co/dbmdz/bert-base-turkish-uncased>

²¹<https://huggingface.co/dbmdz/bert-base-turkish-128k-uncased>

²²<https://huggingface.co/dbmdz/bert-base-turkish-128k-cased>

²³<https://huggingface.co/iarfmoose/roberta-small-bulgarian>

²⁴<https://huggingface.co/savasy/bert-base-turkish-sentiment-cased>

²⁵<https://www.unhcr.org/figures-at-a-glance.html>

Another method to increase the training data size is back-translation which does not deal with social differences across countries. Therefore, in our next experiment, we increase training data using various languages for back-translation. Again, we are not able to report results for Arabic due to technical challenges we encountered. In this experiment, we also use Spanish for back-translation of the Bulgarian dataset, but not the others to meet the deadlines of the lab. The results are shown in **Table 7**

Table 7

The impact of increasing train data using various languages for back-translation (BT). The best result for each language is written in **bold**.

Lang. used for BT	Bulgarian	Dutch	English	Turkish
None	0.26	0.26	0.11	0.55
Bulgarian	-	0.39	0.30	0.51
Dutch	0.26	-	0.23	0.51
English	0.26	0.35	-	0.54
Turkish	0.25	0.41	0.25	-
Arabic	-	0.36	0.27	0.49
Spanish	0.27	-	-	-

We again observe that we achieve the best result for Turkish when we use only the original dataset for training. However, back-translation improves the performance in the Dutch and English datasets. For Bulgarian, back-translation has a minimal impact. We do not observe a particular language which yields consistently higher results than others when used as the language for back-translation.

3.2.3. Selecting Models for Submission

In order to select the models to submit for official ranking, we compare three different approaches for each subtask and language:

- **Fine-tuning the best-performing pre-trained transformer model with the original dataset (FT-BP-TM).** We use the best-performing pre-trained transformer model in our experiments in Section 3.2.1 for all subtasks except 1D. In particular, we fine-tune *AraBERT.v1*, *RoBERTa-base-bulgarian*, *BERTje*, *COVID-Twitter-BERT v1*, and *BERTurk*, for Arabic, Bulgarian, Dutch, English, and Turkish, respectively, using the corresponding datasets.
- **Fine-tuning a transformer model with back translation (FT-TM-BT).** We use the best-performing model in our experiments in Section 3.2.2. In particular, we use Spanish, Turkish, Bulgarian, and English for back-translation to increase the size of Bulgarian, Dutch, English, and Turkish datasets, respectively. Note that the back-translation does not improve the performance in the Turkish dataset. However, the FT-BP-TM approach also uses the original dataset for fine-tuning. Therefore, in this approach, we increase the size of Turkish dataset using back-translation. In particular, we use English as the back-translation language because it yields the best results among others (See Table 7).
- **Manifold Mixup.** We use the Manifold Mixup model explained in Section 2.3.

Table 8, 9, 10, and 11, present results comparing three approaches for subtasks 1A, 1B, 1C, and 1D, respectively. Results for some cases are missing due to technical challenges we encountered and the limited time frame for submissions. In our submissions, we chose the best-performing method for each case and submitted our results accordingly.

Table 8

Development Test Results in Subtask 1A for F_1 Score for the Positive Class

Model	Arabic	Bulgarian	Dutch	English	Turkish
Manifold Mixup	0.14	0	0.58	0.48	0.22
FT-TM-BT	-	0.42	0.64	0.48	0.40
FT-BP-TM	0.47	0.47	0.57	0.55	0.40

Table 9

Development Test Results in Subtask 1B for F_1 Score for the Positive Class

Model	Arabic	Bulgarian	Dutch	English	Turkish
Manifold Mixup	0.76	0.75	0.49	0.67	0.63
FT-TM-BT	-	0.86	0.73	-	0.78
FT-BP-TM	-	0.87	0.72	0.76	0.78

Table 10

Development Test Results in Subtask 1C for F_1 Score for the Positive Class

Model	Arabic	Bulgarian	Dutch	English	Turkish
Manifold Mixup	0.64	0	0.12	0.18	0.30
FT-TM-BT	0.12	0.27	0.41	0.30	0.54
FT-BP-TM	-	0.24	0.33	0.35	0.52

Table 11

Development Test Results in Subtask 1D for Average Weighted F_1 . We do not have results for FT-BP-TM case in this experiment.

Model	Arabic	Bulgarian	Dutch	English	Turkish
Manifold Mixup	0.65	0.80	0.65	0.78	0.79
FT-TM-BT	-	0.33	0.31	-	0.28

3.3. Results of Our Submissions

Table 12 shows our results and ranking for each case we participated. We are ranked first in 1B Arabic and 1C Dutch. Focusing on subtasks with at least four participants, we are ranked second in Arabic 1A and Bulgarian 1A. We also observe that our rankings are generally higher in 1A than other subtasks.

Table 12

Results for our official submissions. Results show F_1 , accuracy, F_1 , and weighted F_1 scores for tasks 1A, 1B, 1C, and 1D, respectively (i.e., the official evaluation metrics).

Task	Language	Submitted Model	Rank	Score
1A	Arabic	FT-BP-TM	2 (out of 5)	0.495
	Bulgarian	FT-BP-TM	2 (out of 6)	0.542
	Dutch	FT-TM-BT	3 (out of 6)	0.534
	English	FT-BP-TM	4 (out of 14)	0.561
	Turkish	FT-TM-BT	3 (out of 5)	0.118
1B	Arabic	Manifold Mixup	1 (out of 4)	0.570
	Bulgarian	FT-BP-TM	2 (out of 3)	0.742
	Dutch	FT-TM-BT	2 (out of 3)	0.658
	English	FT-BP-TM	9 (out of 10)	0.641
	Turkish	FT-TM-BT	4 (out of 4)	0.729
1C	Arabic	Manifold Mixup	2 (out of 3)	0.268
	Bulgarian	FT-TM-BT	2 (out of 3)	0.054
	Dutch	FT-TM-BT	1 (out of 3)	0.147
	English	FT-BP-TM	5 (out of 12)	0.329
	Turkish	FT-TM-BT	3 (out of 5)	0.262
1D	Arabic	Manifold Mixup	2 (out of 2)	0.184
	Bulgarian	Manifold Mixup	2 (out of 3)	0.887
	Dutch	Manifold Mixup	2 (out of 3)	0.694
	English	Manifold Mixup	4 (out of 7)	0.670
	Turkish	Manifold Mixup	3 (out of 3)	0.806

4. Conclusion

In this paper, we present our participation in CLEF 2022 CheckThat! Lab’s Task 1. We participated in all four subtasks of Task1 for Arabic, Bulgarian, Dutch, English, and Turkish, yielding 20 submissions in total. We explore which transformer model yields the highest performance, the impact of increasing training data size by machine translating datasets in other languages and back-translation, and the Manifold Mixup method proposed by Verma et al. [6]. We are ranked first in subtask 1B for Arabic and in subtask 1C for Dutch. In addition, we are ranked second in subtask 1A for Arabic and Bulgarian.

Our observations based on our comprehensive experiments are as follows. Firstly, the performance of transformer models varies dramatically based on the text used for pre-training. Secondly, increasing training data does not always improve the performance. Therefore, it is important to consider biases existing in each dataset. Thirdly, we do not observe that a particular language used for back-translation yields consistently higher performance than others.

In the future, we plan to focus on the subjective nature of the tasks in this lab. In particular, we will first qualitatively analyze the datasets to better understand annotations. Subsequently, we plan to develop a model focusing on dealing with subjective annotations.

References

- [1] J. Roozenbeek, C. R. Schneider, S. Dryhurst, J. Kerr, A. L. Freeman, G. Recchia, A. M. Van Der Bles, S. Van Der Linden, Susceptibility to misinformation about covid-19 around the world, *Royal Society open science* 7 (2020) 201199.
- [2] Republic of turkey ministry of health covid-19 vaccination information platform, https://covid19asi.saglik.gov.tr/?_Dil=2, 2022. Accessed: 2022-06-22.
- [3] P. Nakov, A. Barrón-Cedeño, G. Da San Martino, F. Alam, R. Míguez, T. Caselli, M. Kutlu, W. Zaghouni, C. Li, S. Shaar, H. Mubarak, A. Nikolov, Y. S. Kartal, J. Beltrán, Overview of the CLEF-2022 CheckThat! lab task 1 on identifying relevant claims in tweets, in: N. Faggioli, Guglielmo and Ferro, A. Hanbury, M. Potthast (Eds.), *Working Notes of CLEF 2022—Conference and Labs of the Evaluation Forum, CLEF '2022, Bologna, Italy, 2022*.
- [4] P. Nakov, A. Barrón-Cedeño, G. Da San Martino, F. Alam, J. M. Struß, T. Mandl, R. Míguez, T. Caselli, M. Kutlu, W. Zaghouni, C. Li, S. Shaar, G. K. Shahi, H. Mubarak, A. Nikolov, N. Babulkov, Y. S. Kartal, J. Beltrán, The CLEF-2022 CheckThat! Lab on fighting the covid-19 infodemic and fake news detection, in: M. Hagen, S. Verberne, C. Macdonald, C. Seifert, K. Balog, K. Nørvg, V. Setty (Eds.), *Advances in Information Retrieval, Springer International Publishing, Cham, 2022*, pp. 416–428.
- [5] P. Nakov, A. Barrón-Cedeño, G. Da San Martino, F. Alam, J. M. Struß, T. Mandl, R. Míguez, T. Caselli, M. Kutlu, W. Zaghouni, C. Li, S. Shaar, G. K. Shahi, H. Mubarak, A. Nikolov, N. Babulkov, Y. S. Kartal, J. Beltrán, M. Wiegand, M. Siegel, J. Köhler, Overview of the CLEF-2022 CheckThat! lab on fighting the COVID-19 infodemic and fake news detection, in: A. Barrón-Cedeño, G. Da San Martino, M. Degli Esposti, F. Sebastiani, C. Macdonald, G. Pasi, A. Hanbury, M. Potthast, G. Faggioli, F. Nicola (Eds.), *Proceedings of the 13th International Conference of the CLEF Association: Information Access Evaluation meets Multilinguality, Multimodality, and Visualization, CLEF '2022, Bologna, Italy, 2022*.
- [6] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, Y. Bengio, Manifold mixup: Better representations by interpolating hidden states, in: *International Conference on Machine Learning, PMLR, 2019*, pp. 6438–6447.
- [7] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019*, pp. 4171–4186.
- [8] S. Shaar, M. Hasanain, B. Hamdan, Z. S. Ali, F. Haouari, A. Nikolov, M. Kutlu, Y. S. Kartal, F. Alam, G. Da San Martino, et al., Overview of the clef-2021 checkthat! lab task 1 on check-worthiness estimation in tweets and political debates, in: *CLEF (Working Notes), 2021*.
- [9] E. Williams, P. Rodrigues, S. Tran, Accenture at checkthat! 2021: Interesting claim identification and ranking with contextually sensitive lexical training data augmentation, *arXiv preprint arXiv:2107.05684 (2021)*.
- [10] M. Zengin, Y. Kartal, M. Kutlu, Tobb etu at checkthat! 2021: Data engineering for detecting check-worthy claims, in: *CEUR Workshop Proceedings, CEUR-WS, 2021*.
- [11] Y. S. Kartal, M. Kutlu, Re-think before you share: A comprehensive study on prioritizing check-worthy claims, *IEEE Transactions on Computational Social Systems (2022)*.

- [12] C. Hansen, C. Hansen, J. G. Simonsen, C. Lioma, Neural weakly supervised fact check-worthiness detection with contrastive sampling-based ranking loss., in: CLEF (Working Notes), 2019.
- [13] Y. S. Kartal, M. Kutlu, Trclaim-19: The first collection for turkish check-worthy claim detection with annotator rationales, in: Proceedings of the 24th Conference on Computational Natural Language Learning, 2020, pp. 386–395.
- [14] W. Antoun, F. Baly, H. Hajj, Arabert: Transformer-based model for arabic language understanding, in: LREC 2020 Workshop Language Resources and Evaluation Conference 11–16 May 2020, 2020, p. 9.
- [15] P. Delobelle, T. Winters, B. Berendt, RobBERT: a Dutch RoBERTa-based Language Model, in: Findings of the Association for Computational Linguistics: EMNLP 2020, Association for Computational Linguistics, Online, 2020, pp. 3255–3265. URL: <https://www.aclweb.org/anthology/2020.findings-emnlp.292>. doi:10.18653/v1/2020.findings-emnlp.292.
- [16] M. S. H. Ameer, H. Aliane, Aracovid19-mfh: Arabic covid-19 multi-label fake news and hate speech detection dataset, 2021. arXiv:2105.03143.
- [17] A. Safaya, M. Abdullatif, D. Yuret, KUISAIL at SemEval-2020 task 12: BERT-CNN for offensive speech identification in social media, in: Proceedings of the Fourteenth Workshop on Semantic Evaluation, International Committee for Computational Linguistics, Barcelona (online), 2020, pp. 2054–2059. URL: <https://www.aclweb.org/anthology/2020.semeval-1.271>.
- [18] G. Inoue, B. Alhafni, N. Baimukan, H. Bouamor, N. Habash, The interplay of variant, size, and task type in Arabic pre-trained language models, in: Proceedings of the Sixth Arabic Natural Language Processing Workshop, Association for Computational Linguistics, Kyiv, Ukraine (Online), 2021.
- [19] A. Abdaoui, C. Pradel, G. Sigel, Load what you need: Smaller versions of multilingual bert, in: SustaiNLP / EMNLP, 2020.
- [20] W. de Vries, A. van Cranenburgh, A. Bisazza, T. Caselli, G. v. Noord, M. Nissim, BERTje: A Dutch BERT Model, arXiv:1912.09582, 2019. URL: <http://arxiv.org/abs/1912.09582>.
- [21] M. Müller, M. Salathé, P. E. Kummervold, Covid-twitter-bert: A natural language processing model to analyse covid-19 content on twitter, arXiv preprint arXiv:2005.07503 (2020).
- [22] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, 2020. arXiv:arXiv:2007.15779.
- [23] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, I. Androutsopoulos, LEGAL-BERT: The muppets straight out of law school, in: Findings of the Association for Computational Linguistics: EMNLP 2020, Association for Computational Linguistics, Online, 2020, pp. 2898–2904. doi:10.18653/v1/2020.findings-emnlp.261.
- [24] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut, ALBERT: A lite BERT for self-supervised learning of language representations, CoRR abs/1909.11942 (2019). URL: <http://arxiv.org/abs/1909.11942>. arXiv:1909.11942.
- [25] E. Alsentzer, J. R. Murphy, W. Boag, W.-H. Weng, D. Jin, T. Naumann, M. McDermott, Publicly available clinical bert embeddings, arXiv preprint arXiv:1904.03323 (2019).
- [26] T. Caselli, V. Basile, J. Mitrović, M. Granitzer, HateBERT: Retraining BERT for abusive language detection in English, in: Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021), Association for Computational Linguistics, Online, 2021, pp. 17–25.

URL: <https://aclanthology.org/2021.woah-1.3>. doi:10.18653/v1/2021.woah-1.3.

- [27] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized BERT pretraining approach, CoRR abs/1907.11692 (2019). URL: <http://arxiv.org/abs/1907.11692>. arXiv:1907.11692.
- [28] N. Webersinke, M. Kraus, J. Bingler, M. Leippold, Climatebert: A pretrained language model for climate-related text, arXiv preprint arXiv:2110.12010 (2021).