# Detecting Multilingual COVID-19 Misinformation on Social Media via Contextualized Embeddings

**Subhadarshi Panda**
Hunter College
City University of New York
spanda@gc.cuny.edu

**Sarah Ita Levitan**
Hunter College
City University of New York
sarah.levitan@hunter.cuny.edu

## Abstract

We present machine learning classifiers to automatically identify COVID-19 misinformation on social media in three languages: English, Bulgarian, and Arabic. We compared 4 multitask learning models for this task and found that a model trained with English BERT achieves the best results for English, and multilingual BERT achieves the best results for Bulgarian and Arabic. We experimented with zero shot, few shot, and target-only conditions to evaluate the impact of target-language training data on classifier performance, and to understand the capabilities of different models to generalize across languages in detecting misinformation online. This work was performed as a submission to the shared task, NLP4IF 2021: Fighting the COVID-19 Infodemic. Our best models achieved the second best evaluation test results for Bulgarian and Arabic among all the participating teams and obtained competitive scores for English.

## 1 Introduction

Automatic detection of misinformation online is a crucial problem that has become increasingly necessary in recent years. Misinformation is shared frequently online, especially via social media platforms which generally do not filter content based on veracity. The ongoing COVID-19 pandemic has highlighted the importance of creating tools to automatically identify misinformation online and to help stop the spread of deceptive messages. During a global health crisis, misinformation can cause tremendous damage. A recent study polled Americans about beliefs in COVID-19 misinformation, and found that many conspiracy theories were believed by a substantial percentage of participants. For example, 20% of participants believed that the pandemic is a ruse "to install tracking devices inside our bodies." Such conspiracy theories, when shared widely online, could influence people to make choices based on those beliefs which can jeopardize their own health and safety as well as others around them.

There has been recent work in the NLP community aimed at identifying general misinformation on social media (Shu et al., 2017; Mitra et al., 2017) and particularly COVID-19 misinformation (Hossain et al., 2020). Most of this prior work has focused on data in English. In this paper we address the problem of cross-lingual identification of COVID-19 misinformation. There is a severe data shortage of high quality datasets that are labeled for misinformation in multiple languages. Because of this, we need to develop models of deception and misinformation that can leverage large amounts of training data in a source language, such as English, and generalize to new target languages.

Some prior work on misinformation and deception detection has been applied to a cross-cultural setting (Pérez-Rosas and Mihalcea, 2014) and recently to a cross-lingual setting (Capuozzo et al., 2020b,a). Whereas previous approaches have focused on single task models, in this work we train four different multitask models and demonstrate their performance in cross-lingual settings for identifying misinformation in social media. Two of these models are based on BERT (Devlin et al., 2019). We show that even with no training data in the target language, the multilingual BERT based model can obtain 0.685 F1 in English, 0.81 F1 in Bulgarian and 0.672 F1 in Arabic.

## 2 Data

| Language → | English | Bulgarian | Arabic |
|---|---|---|---|
| Train | 451 (869) | 3000 | 198 (2536) |
| Dev | 53 | 350 | 20 (520) |
| Test | 418 | 357 | 1000 |
| Total | 922 (1340) | 3707 | 1218 (4056) |

Table 1: Data sizes for the three languages. The numbers within parentheses denote the sizes after adding additional data that was released for the shared task.

| Language | Split | Q1 | Q2 | Q3 | Q4 | Q5 | Q6 | Q7 |
|---|---|---|---|---|---|---|---|---|
| English | Train | 569 Y / 300 N | 39 Y / 460 N | 510 Y / 51 N | 156 Y / 409 N | 185 Y / 384 N | 138 Y / 729 N | 229 Y / 634 N |
| | Dev | 27 Y / 26 N | 4 Y / 20 N | 22 Y / 5 N | 11 Y / 16 N | 12 Y / 15 N | 6 Y / 47 N | 8 Y / 45 N |
| Bulgarian | Train | 1933 Y / 1067 N | 64 Y / 1897 N | 1910 Y / 55 N | 181 Y / 1770 N | 392 Y / 1557 N | 316 Y / 2680 N | 300 Y / 2655 N |
| | Dev | 315 Y / 35 N | 5 Y / 316 N | 308 Y / 12 N | 25 Y / 288 N | 62 Y / 254 N | 62 Y / 288 N | 69 Y / 275 N |
| Arabic | Train | 1926 Y / 610 N | 376 Y / 1545 N | 1895 Y / 22 N | 351 Y / 1566 N | 936 Y / 990 N | 459 Y / 2075 N | 2208 Y / 328 N |
| | Dev | 225 Y / 295 N | 12 Y / 210 N | 221 Y / 4 N | 23 Y / 201 N | 107 Y / 118 N | 41 Y / 478 N | 379 Y / 141 N |

Table 2: Distribution of the labels (Yes/No) in the training and dev sets for different languages. The numbers shown are after considering the additional data for train and dev.

We used the tweet data provided for the Fighting the COVID-19 Infodemic shared task (Shaar et al., 2021).[1] The data was created by answering 7 questions about COVID-19 for each tweet (Shaar et al., 2021). Questions include: Q1 – Does the tweet contain a verifiable factual claim? Q2 – Does the tweet appear to contain false information? Each question has a Yes/No (binary) annotation. However, the answers to Q2, Q3, Q4 and Q5 are all "nan" if the answer to Q1 is No. The data includes tweets in three languages: English, Bulgarian and Arabic. An example of an English tweet from the dataset along with its 7 labels is shown below.

> Tweet: *Anyone else notice that COVID-19 seemed to pop up almost immediately after impeachment failed?*
> Labels: *Q1 Yes, Q2 Yes, Q3 Yes, Q4 Yes, Q5 No, Q6 Yes, Q7 No*

The training, development and test data sizes for each of the three languages are shown in Table 1 and the distribution of the Yes/No labels are shown in Table 2.

## 3 Methodology

The task is to predict 7 properties of a tweet about COVID-19 misinformation based on the corresponding 7 questions mentioned in Section 2. We use four multitask learning models for this task as described below.

**Logistic regression** The logistic regression model is a linear model where the output is passed through 7 different linear layers for each prediction. The input to logistic regression model is word embeddings for a given sequence of words. The embedding layer is initialized randomly. We represent the sequence of words as the sum of the token level embeddings for a given sentence. The loss is computed as the sum of the cross entropy loss for each of the 7 tasks. This logistic regression model is a simple approach and provides a baseline to compare other more complex models with.

**Transformer encoder** The logistic regression model ignores the word order in the input sentence and handles the input as a bag of words. To consider word order, models such as LSTMs (Hochreiter and Schmidhuber, 1997) and more recently attention based networks called transformers (Vaswani et al., 2017) have been shown to be effective. A transformer encoder model is an attention based model that uses positional embeddings to incorporate word order. We add a `[CLS]` token in the beginning of each sentence. The classification is done based on the `[CLS]` token's representation. For our multitask objective, the `[CLS]` token's representation is passed through 7 different linear layers separately to produce the logits corresponding to the 7 tasks. As in the case of the logistic regression model, the loss is computed as the sum of the cross-entropy loss for each task.

**English BERT** BERT (Devlin et al., 2019) is a large language model trained on a gigantic amount of text data from BookCorpus (Zhu et al., 2015) and Wikipedia. Pre-trained BERT has been shown to be effective in a wide range of NLP tasks (Devlin et al., 2019) by fine-tuning on data for a specific task. For our multitask objective, we use the `[CLS]` token's representation of BERT and pass it through 7 separate linear layers to produce the logits corresponding to the 7 tasks. The loss is computed as the sum of the cross entropy loss for each task.

**Multilingual BERT** Multilingual BERT (m-BERT) (Devlin et al., 2019) is a single large language model pre-trained from monolingual corpora in 104 languages. It has been shown to have strong cross-lingual generalization ability (K et al., 2020). m-BERT also manages to transfer knowledge between languages that have very little or no lexical overlap (Pires et al., 2019). For our multitask objective, similar to BERT we use the `[CLS]` token's representation of BERT and pass it through 7 sepa-

[1] `https://gitlab.com/NLP4IF/nlp4if-2021`

| Trg. Lang. | Setup | Src. Lang.: Bulgarian | | | | Src. Lang.: Arabic | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Log. Reg. | Transf. Enc. | BERT | m-BERT | Log. Reg. | Transf. Enc. | BERT | m-BERT |
| English | zero | 0.523 | 0.569 | 0.488 | **0.685** | 0.436 | 0.517 | 0.594 | **0.683** |
| | few50 | 0.601 | 0.578 | 0.643 | **0.672** | 0.537 | 0.553 | 0.63 | **0.659** |
| | few100 | 0.607 | 0.619 | 0.621 | **0.659** | 0.622 | 0.609 | 0.639 | **0.663** |
| | few150 | 0.635 | 0.61 | 0.632 | **0.655** | 0.627 | 0.61 | **0.729** | 0.665 |
| | few200 | 0.674 | 0.635 | **0.713** | 0.696 | 0.661 | 0.686 | **0.722** | 0.68 |
| | full | 0.713 | 0.67 | **0.729** | 0.7 | 0.715 | 0.68 | **0.745** | 0.712 |
| | trg | 0.698 | 0.686 | **0.745** | 0.722 | 0.698 | 0.686 | **0.745** | 0.722 |

| Trg. Lang. | Setup | Src. Lang.: English | | | | Src. Lang.: Arabic | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Log. Reg. | Transf. Enc. | BERT | m-BERT | Log. Reg. | Transf. Enc. | BERT | m-BERT |
| Bulgarian | zero | 0.37 | 0.804 | 0.803 | **0.81** | 0.558 | 0.805 | 0.803 | **0.808** |
| | few50 | 0.776 | 0.8 | 0.81 | **0.819** | 0.794 | 0.804 | 0.811 | **0.815** |
| | few100 | 0.781 | 0.81 | 0.816 | **0.823** | 0.799 | 0.808 | 0.818 | **0.821** |
| | few150 | 0.796 | 0.819 | 0.819 | **0.821** | 0.8 | 0.816 | 0.82 | **0.821** |
| | few200 | 0.807 | 0.82 | 0.816 | **0.825** | 0.8 | 0.811 | **0.822** | 0.82 |
| | full | 0.812 | 0.815 | 0.822 | **0.834** | 0.821 | 0.812 | 0.822 | **0.836** |
| | trg | 0.814 | 0.81 | 0.822 | **0.843** | 0.814 | 0.81 | 0.822 | **0.843** |

| Trg. Lang. | Setup | Src. Lang.: English | | | | Src. Lang.: Bulgarian | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Log. Reg. | Transf. Enc. | BERT | m-BERT | Log. Reg. | Transf. Enc. | BERT | m-BERT |
| Arabic | zero | 0.422 | 0.585 | 0.599 | **0.672** | 0.547 | 0.615 | 0.558 | **0.638** |
| | few50 | 0.727 | 0.69 | 0.675 | **0.775** | 0.676 | 0.647 | 0.657 | **0.76** |
| | few100 | 0.743 | 0.686 | 0.692 | **0.824** | 0.698 | 0.734 | 0.662 | **0.753** |
| | few150 | 0.718 | 0.689 | 0.698 | **0.791** | 0.726 | 0.688 | 0.652 | **0.775** |
| | full | 0.747 | 0.74 | 0.712 | **0.787** | 0.708 | 0.716 | 0.679 | **0.764** |
| | trg | 0.649 | 0.684 | 0.735 | **0.738** | 0.649 | 0.684 | 0.735 | **0.738** |

Table 3: Cross-lingual (source language → target language) results (F1 score) on the development set. *fewx* setup denotes that only $x$ samples in the target language are used for training.

rate linear layers to produce the logits corresponding to the 7 tasks. The loss is computed as the sum of the cross entropy loss for each task.

### 3.1 Post-processing

The multitask learning models produce outputs which may not satisfy the required constraint that if the prediction for Q1 is No, then the predictions for Q2, Q3, Q4 and Q5 should all be "nan". To make sure that this constraint is satisfied, we apply post-processing to the output. The post-processing involves overwriting the prediction for Q2, Q3, Q4, Q5 to “nan” if the prediction for Q1 is No. The post-processing does not have any impact if the prediction to Q1 is Yes.

## 4 Experiments

We performed cross lingual experiments in different setups as outlined below. We are interested in gauging the cross-lingual ability of prediction models when trained on data from one language (source language) and tested on data from another language (target language). We compare 4 experimental conditions, varying the amount of source language and target language data that is used for training. The four conditions are described below.

**Zero shot** In this condition, we only used the source language training data to train the model. The model does not see any training data in the target language and we only evaluated the model on the target language dev set. The advantage of the zero shot setup is that it enables us to evaluate the prediction models in conditions when no training data is available in the target language.

**Few shot** In this setup, we considered all the source language training data combined with $x$ training samples from the target language. We set $x$ to 50, 100, 150 and 200. We select $x$ samples from the complete target language training data uniformly at random to simulate the few shot setup. For Arabic, the number of training data samples is 198 (without using additional data, see Table 1). So we set $x$ to 50, 100, and 150. The advantage of the few shot setup is that it enables us to gauge the performance when only a handful of training samples are available in the target language.

**Full shot** In this setup, we used all the available training data from the source and target languages for training the model. This setup is useful to see the impact of the source language training data when combined to the target language training data.

**Target** In this setup, we used only the target language training data for training. This setup enables us to evaluate the models when there is availability of training data in the target language, and compare monolingual with cross-lingual classification.

### 4.1 Training

We implemented the models, training and evaluation pipelines using PyTorch (Paszke et al., 2019).[2] For the logistic regression and the transformer en-

[2] The code is available in https://github.com/subhadarship/nlp4if-2021.

coder models, we first tokenized the tweets and normalized the emojis, urls and usernames.[3] We tuned the hidden layer size {128, 256, 512} and the maximum vocabulary size {8k, 16k, 32k} by considering only the most frequent set of tokens. We set the dropout to 0.1 and used the Adam optimizer setting the learning rate to 0.0005. For the transformer encoder model, we used 3 encoder layers and 8 heads.

For the BERT based models, we used the Transformers library (Wolf et al., 2020) and loaded the `bert-base-uncased` model for English BERT and `bert-base-multilingual-cased` model for m-BERT, and also the corresponding tokenizers. We tuned the hidden layer size {128, 256, 512} of the added linear layer and the learning rate {0.0005, 0.005, 0.05}. The optimizer used was Adam. We also experimented with two variants: training the BERT pre-trained weights or freezing them. We found that freezing them resulted in better results overall. For training all the models, we used early stopping, that is, we stopped training when the dev F1 score does not improve for 10 consecutive epochs.

## 5 Results

| Lang. | Model | Dev F1 | | Test F1 |
|---|---|---|---|---|
| | | No add. data | Add. data | |
| En | BERT | **0.745** | 0.729 | 0.736 |
| Bg | m-BERT | **0.843** | - | 0.817 |
| Ar | m-BERT | 0.556 | **0.688** | 0.741 |

Table 4: Best scores on the dev set and final score on the test set. Best scores were obtained when trained in the target setup. For Arabic the dev set used for evaluation contains the additional data also (see Table 1).

For most experiments, we evaluate on the target language dev sets, since those labels are available for evaluation. We also report the final test set evaluation, which was conducted by the organizers based on our submitted predicted labels for the blind test set. We used the initial data release for most of our reported experiments (i.e. without the additional data released by the shared task organizers closer to the deadline) unless otherwise noted. Table 3 shows the results for different source-target language pairs, comparing the 4 multitask learning models in the multiple experimental conditions (zero shot, few shot, target). The results indicate that out of all the four models considered, fine-tuning multilingual BERT generalizes best across languages. Remarkably, for the target language Bulgarian, even without using any Bulgarian training data, multilingual BERT obtains 0.81 F1 score when trained on English only and 0.808 F1 score when trained on Arabic only. As we increase the target language training samples in the few shot setup, the performance increases, as one would expect. For each model, the best scores are usually obtained by using all the target language training data, either in the full shot setup or in the target-only setup. Overall, multilingual BERT achieves the best F1 scores.

We identified the best systems based on the dev set scores and predicted the test set labels using them. Table 4 shows the top performing models that we submitted for test evaluation, along with their dev and test F1 scores. In addition, the table shows a comparison between the dev F1 scores with and without the additional training data. Surprisingly, the additional English training data did not improve the English F1 score. However, using the additional Arabic training data resulted in a substantial improvement in Arabic F1 score.

## 6 Conclusion

In this paper, we described Hunter SpeechLab's submission to the shared task, NLP4IF 2021: Fighting the COVID-19 Infodemic. We explored the cross-lingual generalization ability of multitask models trained from scratch (logistic regression, transformer encoder) and pre-trained models (English BERT, m-BERT) for deception detection. We found that even without using any training samples in Bulgarian and Arabic (zero shot), m-BERT achieves impressive scores when evaluating on those languages. In some cases, using just a few training samples in the target language achieves results equal or better than using all the training data in the target language. Our best systems are based on English BERT for English and multilingual BERT for Bulgarian and Arabic. We obtained competitive evaluation test scores on all the three languages, especially Bulgarian and Arabic for which we obtained second best scores among all participating teams. In future work we will further explore the cross-lingual generalization ability of BERT based models in detecting false or deceptive information.

[3] We used the script from https://github.com/VinAIResearch/BERTweet/blob/master/TweetNormalizer.py for tokenization and normalization of tweets.

## References

Pasquale Capuozzo, Ivano Lauriola, Carlo Strapparava, Fabio Aiolli, and Giuseppe Sartori. 2020a. Automatic detection of cross-language verbal deception.

Pasquale Capuozzo, Ivano Lauriola, Carlo Strapparava, Fabio Aiolli, and Giuseppe Sartori. 2020b. DecOp: A multilingual and multi-domain corpus for detecting deception in typed text. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1423–1430, Marseille, France. European Language Resources Association.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Tamanna Hossain, Robert L Logan IV, Arjuna Ugarte, Yoshitomo Matsubara, Sean Young, and Sameer Singh. 2020. Covidlies: Detecting covid-19 misinformation on social media. In *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020*.

Karthikeyan K, Zihan Wang, Stephen Mayhew, and Dan Roth. 2020. Cross-lingual ability of multilingual bert: An empirical study. In *International Conference on Learning Representations*.

Tanushree Mitra, Graham P Wright, and Eric Gilbert. 2017. A parsimonious language model of social media credibility across disparate events. In *Proceedings of the 2017 ACM conference on computer supported cooperative work and social computing*, pages 126–145.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Verónica Pérez-Rosas and Rada Mihalcea. 2014. Cross-cultural deception detection. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 440–445, Baltimore, Maryland. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Shaden Shaar, Firoj Alam, Giovanni Da San Martino, Alex Nikolov, Wajdi Zaghouani, Preslav Nakov, and Anna Feldman. 2021. Findings of the NLP4IF-2021 shared task on fighting the COVID-19 infodemic and censorship detection. In *Proceedings of the Fourth Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda*, NLP4IF@NAACL' 21, Online. Association for Computational Linguistics.

Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter*, 19(1):22–36.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27.