CEUR-WS.org/Vol-3180/paper-41.pdf

RUB-DFL at CheckThat! 2022: Transformer Models and Linguistic Features for Identifying Relevant Claims

Zehra Melce Hüsünbeyi[1], Oliver Deck[1] and Tatjana Scheffler[1]

[1]*Digital Forensic Linguistics, Ruhr-Universität-Bochum, Universitätsstraße 150, 44801 Bochum, Germany*

Abstract

We describe our system for the CLEF 2022 CheckThat! Lab Task 1 Subtasks A,B,C on check-worthiness estimation, verifiable factual claims detection, and harmful tweet detection in both English and Turkish. We used transformer-based models as well as an ELMo-based attention network. We experimented with data pre-processing, data augmentation and adding linguistic features. The official evaluation ranked our system 1st and 2nd for the Turkish data while we achieved average results for the English data.

Keywords

claim identification, check-worthiness, English, Turkish, linguistic features, Twitter

1. Introduction

The CheckThat! lab at CLEF [1, 2] aims at providing automated solutions that facilitate or support fake news detection and related subtasks. Automated systems can provide the basis for human fact checkers and may take over some of the more tedious tasks in dealing with an ever increasing number of online disinformation. This paper gives an overview of team RUB-DFL's system for Task 1: Identifying Relevant Claims in Tweets [3]. Fact checking should only be applied to claims (and not e.g. opinions or predictions about the future), so identifying claims and an assessment of their relevance can be used to prioritize which claims to check.

Our team participated in three of the four subtasks, namely check-worthiness estimation, claim detection, and harmful tweet detection, for both the English and Turkish data sets. We conducted experiments with transformer-based models, data augmentation and linguistic features, as well as ELMo embeddings and attention networks. Our system reached 1st place for claim identification and check-worthiness estimation in Turkish and average results on English data. For harmful tweet detection, we placed 9th on the English data and 2nd on the Turkish data.

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

melce.husunbeyi@rub.de (Z. M. Hüsünbeyi); oliver.deck@rub.de (O. Deck); tatjana.scheffler@rub.de (T. Scheffler)

https://tscheffler.github.io/ (T. Scheffler)

© 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)

2. Related Work

Disinformation detection has received significant attention in NLP in recent years. Many systems, data sets and challenges take a holistic approach of so-called fake news detection [4, 5, 6]. However, the CheckThat! lab has a different aim: Many of the challenges in the previous years, as well as in 2022, have been looking at smaller, more manageable subtasks of disinformation detection such as check-worthiness identification and detecting previously fact-checked claims [7, 8]. In this manner, automated systems can play out their strengths in pattern detection while receiving oversight from human fact checkers. Real-world NLP systems can thus provide e.g. a list of check-worthy claims which can be used a starting point for journalistic investigation.

Similar datasets to this challenge can be found in ClaimBuster [9] and ClaimsKG [10], as well as in previous years' CheckThat! labs [11, 12, 13]. While there was no task on claim identification in the 2021 CheckThat! challenge, the winning systems on check-worthiness estimation and detection of previously fact-checked claims in English Tweets were teams NLP&IR@UNED [14] and team Aschern [15], respectively. Both used BERT models and team Aschern additionally used TF-IDF and the re-ranking LambdaMART model. More information on all the participating systems and the approaches they employed can be found in the official overview papers published by the task organizers [7, 8].

3. Data and Pre-processing

The data for all three subtasks tackled by our team consisted of between 2891 and 4542 tweets. Tweets were provided with binary labels corresponding to either check-worthiness (subtask 1a), containing a claim (1b) or containing hateful speech (1c). Since there is considerable overlap between the datasets – e.g. every check-worthy claim in subtask 1a is automatically a positive example for a claim in subtask 1b – it is not particularly helpful to combine the datasets to gain a larger basis for training models. However, for subtask 1a, we could utilize last year's CheckThat! data, which we did in section 5.

Simple data pre-processing steps were taken into consideration for the experiments in section 4.2. These include: changing all text to lower case, removing all URLs, twitter mentions, punctuation that does is not part of an emoticon, and removing all remaining characters that are not letters, numbers, white space, or #.

We also considered two very simple approaches to data augmentation: Adding additional data from last year's challenge, as mentioned above, as well as adding linguistic inquiry and word counts (LIWC) [16]. For this step, we tokenized the tweet text and looked up each token in the LIWC dictionary; a word list categorized by psycholinguistic and cognitive dimensions, such as NegativeEmotion, Pronoun, or Health. For each token found in the LIWC dictionary, the corresponding LIWC category was simply appended to the tweet text. The hope was to push the classifier to pay greater attention to these psycholinguistic features, instead of relying simply on the given text.

4. Experiments

4.1. Transformer-based models

Transformer-based models like BERT [17] have significantly improved the performance on a wide range of NLP problems such as claim detection related tasks which are typically framed as text classification problems. The BERT architecture follows masked language model (MLM) and next sentence prediction (NSP) procedures. This structure allows the model to learn the relationship between masked words and bidirectionally incoming text, to predict whether a second sentence follows a first, and to examine sentence relationships in an advanced way. After the release of BERT, other transformer-based pretrained language models have employed similar approaches while refining aspects like model size, training speed and efficiency, multilingual embeddings and more.

After analysing recent studies on benchmark datasets [18] for text classification tasks, we decided to experiment with autoencoding pretrained language models (PLMs) which mostly outperform autoregressive PLMs (e.g., OpenGBT) and earlier contextualized language models (e.g., CNN and RNN based models). The following PLMs were chosen by considering criteria such as domain compatibility, latency and capacity constraints: *BERTweet* [19] because it fits the target domain of the task; *XLM-R* [20] to experiment with multilingual embedding spaces; *ConvBERT* [21] and *ELECTRA* [22] as more computationally efficient models. For the Turkish data, we also used the multilingual XLM-R model, but switched to the Turkish variants of the other models: *BERTurk*[1], *ConvBERTurk*[2] and the Turkish *ELECTRA*[3] model.

Despite the significance of hyperparameter tuning, the growing parameter space and lack of memory limit the tuning process to the chosen hyperparameters. We tuned hyperparameters along with controlled experiments and used a fixed seed value used to ensure consistency. For all experiments, weighted-average F1 scores are presented, considering the size of each class and their contribution to the f-score. We used the rich and publicly available AI repository Huggingface[4] for the PLMs.

English As seen in Table 1, all four models lead to comparable results, although ConvBERT was slightly ahead for subtask 1a (check-worthiness of tweets) with an f-score of 0.839, while BERTweet achieved the highest f-score on subtasks 1b (verifiable factual claims detection) at 0.814 and 1c (harmful tweet detection) at 0.895. However, all f-scores, with the exception of XLM-R in subtask 1a, were within 0.03 points of each other. Such close results, combined with different systems winning different, though related, tasks on very similar data prohibit identifying a clearly superior approach. Further experimentation is needed to explore relevant factors for the success of a particular model.

Turkish For the Turkish data, BERTurk provided the highest f-score for subtask 1a at 0.813, the multilingual model XLM-R achieved the highest f-score on subtask 1b at 0.768 and ConvBERTurk

[1] https://huggingface.co/dbmdz/bert-base-turkish-cased

[2] https://huggingface.co/dbmdz/convbert-base-turkish-cased

[3] https://huggingface.co/dbmdz/electra-base-turkish-cased-discriminator

[4] https://huggingface.co/

Table 1
Transformer-based models without data pre-processing (English).

English		accuracy	precision	recall	f-score
Check-worthiness of tweets (EN)	BERTweet	0.824	0.823	0.824	0.823
	XLM-R	0.807	0.788	0.807	0.790
	ConvBERT	**0.838**	0.841	0.838	**0.839**
	ELECTRA	0.826	0.815	0.826	0.818
Verifiable factual claims detection (EN)	BERTweet	**0.816**	0.814	0.816	**0.814**
	XLM-R	0.809	0.807	0.809	0.806
	ConvBERT	0.810	0.810	0.810	0.804
	ELECTRA	0.828	0.826	0.828	0.826
Harmful tweet detection (EN)	BERTweet	0.907	0.889	0.907	**0.895**
	XLM-R	**0.910**	0.828	0.910	0.867
	ConvBERT	0.903	0.885	0.903	0.892
	ELECTRA	0.910	0.828	0.910	0.867

Table 2
Transformer-based models without data pre-processing (Turkish).

Turkish		accuracy	precision	recall	f-score
Check-worthiness of tweets (TR)	BERTurk	0.820	0.808	0.820	**0.813**
	XLM-R	**0.833**	0.803	0.833	0.805
	ConvBERTurk	0.830	0.800	0.830	0.806
	ELECTRA	0.827	0.795	0.827	0.801
Verifiable factual claims detection (TR)	BERTurk	**0.782**	0.777	0.782	0.772
	XLM-R	0.777	0.771	0.777	**0.768**
	ConvBERTurk	0.762	0.755	0.762	0.756
	ELECTRA	0.770	0.764	0.770	0.764
Harmful tweet detection (TR)	BERTurk	0.781	0.773	0.782	0.776
	XLM-R	0.736	0.675	0.736	0.630
	ConvBERTurk	**0.788**	0.777	0.788	**0.781**
	ELECTRA	0.760	0.743	0.761	0.748

took the lead in task 1c at 0.781 f-score, see Table 2. Again, the close field – only XLM-R in subtask 1c deviated by more than 0.03 f-score from any of the other systems – provided little insight into which system would perform best in general.

4.2. Transformer-based models with pre-processed data

To investigate the merits of data pre-processing described in section 3, we ran the same systems again on the simpler, cleaner data. With fewer confusing factors such as Twitter mentions and punctuation, the transformer-based models presumably encountered fewer situations they had not seen in training. As can be seen in Table 3 and Table 4 we therefore achieved slightly higher f-scores.

English ConvBERT increased in subtask 1a from 0.839 to 0.843, overtaking BERTweet (previously 0.814) with 0.817 in subtask 1b and achieving even numbers with BERTweet (previously 0.895) in subtask 1c where both increased to 0.906 f-score with the pre-processed data. XLM-R and BERTweet were not strongly affected by the pre-processing (BERTweet performance shows a slight decrease for subtask 1a and XLM-R for subtask 1b). ELECTRA, however, exhibited lower scores for all subtasks, leading to the assumption that it managed to pick up on signals that were removed by pre-processing.

Turkish Simple pre-processing lead to small increases in all three subtasks for the Turkish data as well: In subtask 1a, the top system BERTurk increased from 0.813 to 0.822 f-score, for subtask 1b, BERTurk overtook XLM-R (which decreased from 0.768 to 0.740) with an f-score of 0.788. In subtask 1c, ConvBERTurk increase from 0.781 to 0.781. All increases are fairly small and some systems even decreased in performance. However, since the best models for each task showed improvements, it seems that pre-processing also helps with the agglutinative structure of the Turkish language.

Table 3
Transformer-based models with data pre-processing (English).

English		accuracy	precision	recall	f-score
Check-worthiness of tweets (EN)	BERTweet	0.819	0.821	0.819	0.820
	XLM-R	0.820	0.784	0.820	0.793
	ConvBERT	**0.845**	0.841	0.845	**0.843**
	ELECTRA	0.775	0.601	0.775	0.677
Verifiable factual claims detection (EN)	BERTweet	0.819	0.817	0.819	0.816
	XLM-R	0.799	0.797	0.799	0.795
	ConvBERT	**0.820**	0.818	0.820	**0.817**
	ELECTRA	0.787	0.784	0.787	0.784
Harmful tweet detection (EN)	BERTweet	**0.914**	0.902	0.914	**0.906**
	XLM-R	0.910	0.828	0,910	0,867
	ConvBERT	0.909	0.903	0.909	**0.906**
	ELECTRA	0.910	0.828	0.910	0.867

Table 4
Transformer-based models with data pre-processing (Turkish).

		accuracy	precision	recall	f-score
Check-worthiness of tweets (TR)	BERTurk	**0.828**	0.818	0.827	**0.822**
	XLM-R	0.833	0.798	0.833	0.792
	ConvBERTurk	0.805	0.776	0.805	0.787
	ELECTRA	0.817	0.780	0.817	0.789
Verifiable factual claims detection (TR)	BERTurk	**0.794**	0.789	0.794	**0.788**
	XLM-R	0.755	0.747	0.755	0.740
	ConvBERTurk	0.767	0.760	0.767	0.760
	ELECTRA	0.721	0.710	0.721	0.711
Harmful tweet detection (TR)	BERTurk	0.774	0.759	0.774	0.762
	XLM-R	0.736	0.542	0.736	0.625
	ConvBERTurk	**0.783**	0.781	0.783	**0.782**
	ELECTRA	0.773	0.769	0.773	0.771

4.3. Transformer-based models with data augmentation

In a first simple step, we focused on augmenting the data of subtask 1a (check-worthiness estimation) with additional data from last year's CheckThat! challenge; subtasks 1b and 1c were different from last year. We first collected the tweets from the 2021 challenge, removed all duplicates and negative examples (to balance the dataset more towards the positive, i.e. check-worthy class). This left us with an additional 875 English and 237 Turkish tweets, which we added to the training data.

As can be seen in Table 5, results varied based on the language of the data. For English, we saw f-score increases from 0.839 to 0.854 (without pre-processing) and 0.843 to 0.853 for the ConvBERT system, indicating that the additional data contained textual markers that could be picked up by the transformer model. For Turkish, on the other hand, the performance of the winning system BERTweet decreased from 0.813 to 0.805 without pre-processing. For the pre-processed data, while BERTweet achieved 0.822 f-score on the non-augmented data, its performance dropped to 0.797 when trained on the augmented data, reaching second place behind ConvBERTurk with 0.805 f-score. It therefore seems that the Turkish systems may have picked up specific markers of the 2022 data before that better solved the development set, but may not have been actual markers of check-worthiness, leading to a reduced performance when trained on additional data from 2021. Further discussion of the challenges of the Turkish dataset can be found in the Error Analysis section below.

Our second approach to data augmentation was adding LIWC categories to tweets. This was only possible for the English data, since we had no access to the Turkish version of LIWC. Table 6 shows the best performing systems for each subtask on this augmented data. As can

Table 5
Transformer-based models, data augmentation with additional positive samples and pre-processing.

		accuracy	precision	recall	f-score
Check-worthiness of tweets (EN) without data pre-processing	BERTweet	0.824	0.823	0.824	0.823
	XLM-R	0.787	0.787	0.787	0.787
	ConvBERT	**0.855**	0.853	0.855	**0.854**
	ELECTRA	0.815	0.807	0.815	0.810
Check-worthiness of tweets (EN) with data pre-processing	BERTweet	0.819	0.821	0.819	0.820
	XLM-R	0.786	0.776	0.786	0.780
	ConvBERT	**0.854**	0.852	0.854	**0.853**
	ELECTRA	0.845	0.835	0.845	0.835
Check-worthiness of tweets (TR) without data pre-processing	BERTurk	0.803	0.807	0.803	**0.805**
	XLM-R	0.785	0.749	0.785	0.763
	ConvBERTurk	0.802	0.788	0.802	0.794
	ELECTRA	**0.814**	0.795	0.814	0.802
Check-worthiness of tweets (TR) with data pre-processing	BERTurk	0.797	0.797	0.797	0.797
	XLM-R	0.789	0.754	0.789	0.768
	ConvBERTurk	0.804	0.805	0.805	**0.805**
	ELECTRA	**0.806**	0.784	0.806	0.793

be seen, there was no increase in performance when compared to training on either the raw or pre-processed data. Highest f-scores were 0.821 as opposed to 0.843 for subtask 1a, 0.771 as opposed to 0.817 for subtask 1b, and 0.895 as opposed to 0.906 for subtask 1c.

One explanation is that transformer-based models are trained on natural text and artificially appended LIWC categories are not something the model has seen in training. Such features may be more helpful when integrated in an ensemble model where one part picks up on the LIWC features and can then be combined with the transformers' output. Due to time constraints, we must leave this experiment for future work.

Table 6
Best-performing transformer-based models, data augmentation with LIWC categories.

		accuracy	precision	recall	f-score
Check-worthiness of tweets (EN)	ConvBERT	0.826	0.818	0.826	0.821
Verifiable factual claims detection (EN)	ELECTRA	0.774	0.770	0.774	0.771
Harmful tweet detection (EN)	BERTweet	0.905	0.889	0.905	0.895

4.4. Transformer-based models with additional linguistic features

For this approach, we first calculated nine basic linguistic features as a baseline: word count, character count, punctuation count, emoji count, contains emoji, contains non-Twitter URL, number of LIWC categories, text complexity, and sentiment. For Turkish, only the first six features were calculated. The features were concatenated with our transformer-based models to see if adding simple linguistic markers would lead to improvements.

For the English data, we also calculated 239 additional linguistic features with the help of the `lingfeat` library[5] which was originally used for readability assessment [23]. Due to time constraints, we were not able to implement our own feature set specifically adapted for claim detection and relied on this out-of-the-box solution for English. The 239 features include semantic (e.g. Wikipedia knowledge), discourse (e.g. entity density), syntactic (e.g. part-of-speech), lexico-semantic (e.g. type token ratio), as well as shallow traditional features (e.g. average number of tokens). An overview of all features can be found in [23, p. 10672].

The transformer-based models capture different levels of semantic and syntactic knowledge by use of multi-head attention layers. By concatenating the last four layers of our best performing transformer-based model for each task, we aimed to obtain better representations. These 3072-dimensional document embeddings were processed through a fully connected layer with 1024 hidden units and the ReLU activation function. A dropout regularization with a rate of 0.2 was then performed. The resulting hidden layer was incorporated with the 9-dimensional and 239-dimensional external linguistic features separately for the English datasets and the 6-dimensional numerical features for the Turkish datasets. The concatenated vectors were passed to a fully connected layer with 128 hidden units and the ReLU activation function. Another dropout regularization with a rate of 0.1 was applied to the hidden layer and predictions were generated with a sigmoid activation function.

The results for both the baseline features and the whole range of linguistic features provided by the `lingfeat` library can be found in Table 7. Like before, only the best performing models are shown here. As can be seen, the performance was lower than our pure transformer-based models trained on pre-processed data in Tables 3 and 4. What is more, the 239 linguistic features for English lead to lower performance than the 9 simple features. Other experiments with a logistic regression classifier on the linguistic features alone provided very low numbers that barely beat a random baseline. From this we can gather that simply adding a large list of linguistic features which are not necessarily adapted to the task at hand is not helpful. Instead, the low performance of the linguistic features lead to a deterioration of the ensemble when compared to the transformer models alone. However, with more fine-tuning and by identifying linguistic features that are domain-specific, different fusion techniques could be explored in the future.

[5] https://github.com/brucewlee/lingfeat

Table 7
Best-performing transformer-based models merged with additional linguistic features.

			accuracy	precision	recall	f-score
9 basic ling. features	Check-worthiness of tweets (EN)	ConvBERT	0.777	0.753	0.777	0.684
	Verifiable factual claims detection (EN)	ELECTRA	0.816	0.816	0.816	0.816
	Harmful tweet detection (EN)	BERTweet	0.910	0.828	0.910	0.867
6 basic ling. features	Check-worthiness of tweets (TR)	BERTurk	0.835	0.806	0.835	0.809
	Verifiable factual claims detection (TR)	BERTurk	0.765	0.778	0.765	0.769
	Harmful tweet detection (TR)	ConvBERTurk	0.783	0.774	0.783	0.777
239 advanced ling. features	Check-worthiness of tweets (EN)	ConvBERT	0.775	0.714	0.775	0.680
	Verifiable factual claims detection (EN)	ELECTRA	0.653	0.760	0.653	0.539
	Harmful tweet detection (EN)	BERTweet	0.903	0.827	0.903	0.864

4.5. ELMo embeddings, attention network and linguistic features

In a final round of experiments, we moved away from the transformer architecture and evaluated the basic and advanced linguistic features in an ensemble of ELMo embeddings [see 24] in an attention network. Pre-trained ELMo embeddings were processed along with the encoder, a bidirectional RNN based model. We used a GRU rather than an LSTM model to decrease parameters and prevent overfitting given the small size of our corpus. The models were trained with a 500-dimensional bidirectional GRU token encoder. Then, an attention layer producing a sequence vector with indicative tokens received the hidden states of the encoder layer. After dropout regularization with a rate of 0.2, the attention layer's output vectors were merged with either the 9-dimensional or 239-dimensional external linguistic features separately. These concatenated vectors were then fed to a fully connected layer with a ReLU activation function. We also added dropout regularization with a rate of 0.1 to the hidden layer. Predictions were created using the sigmoid activation function.

The results are shown in Table 8. As with the transformer models in section 4.2, the ELMo embedding ensemble performed worse when compared to the transformer models trained on pre-processed text. For the runs with 9 linguistic features, the subtask 1a f-score was 0.803 which would take 3rd place in direct comparison with the transformer models. In subtask 1b, all transformer models beat the 0.7612 f-score of the ELMo ensemble, but in subtask 1c, the 0.881 f-score would place it in 3rd place behind the 0.906 of BERTweet and ConvBERT. When compared to combining transformer models with linguistic features, the attention network with ELMo embeddings performed much better, which may be based on the transformers picking up more relevant linguistic features in their training process inherently, while the architecture used in this chapter lends itself more easily to adding additional signals.

Again, the 239-dimensional linguistic features lead to lower performance. Since the features are not task-specific for any of the three subtasks, they may simply provide too much noisy data leading to lower performance in the systems.

Table 8
ELMo embeddings and attention network model merged with additional linguistics features

		accuracy	precision	recall	f-score
9 basic ling. features	Check-worthiness of tweets (EN)	0.814	0.800	0.814	0.803
	Verifiable factual claims detection (EN)	0.767	0.763	0.767	0.7612
	Harmful tweet detection (EN)	0.897	0.872	0.897	0.881
239 advanced ling. features	Check-worthiness of tweets (EN)	0.774	0.599	0.774	0.675
	Verifiable factual claims detection (EN)	0.370	0.137	0.370	0.200
	Harmful tweet detection (EN)	0.910	0.828	0.910	0,867

4.6. Official results on the test set

We submitted the best models in terms of f-score measure for subtasks 1a, 1b, and 1c in both English and Turkish: For subtask 1a English ConvBert with additional data, for Turkish BERTurk with data pre-processing. For subtask 1b English we chose ELECTRA, for Turkish BERTurk with data pre-processing. For subtask 1c English BERTweet with data pre-processing, and for Turkish ConvBert with data pre-processing were chosen.

Our systems reached average scores on the English data, placing 6th out of 13 teams in subtask 1a with an F1 score for the positive class of 0.525 (winning system: 0.698). In subtask 1b, we placed 6th out of 9 systems with an F1 accuracy score of 0.709 (winning system: 0.761) and for subtask 1c we placed 9th out of 11 teams with an F1 for the positive class of 0.273 (winning system: 0.397).

On the Turkish data, we placed 1st in subtask 1a (F1 positive class: 0.212) and 1b (F1 accuracy: 0.801) and 2nd in subtask 1c (F1 positive class: 0.353, winning system: 0.366). While the scores for task 1c were low across both languages, as well as in the Arabic, Bulgarian and Dutch data sets, the extremely low numbers for task 1a (check-worthiness) in Turkish are an outlier. Here, we were the only team that managed to surpass 0.2 F1 score. It seems that all systems overfit on the training and development data and were not capable of identifying actual check-worthiness markers that would translate to performing well on the test set.

5. Error Analysis

Due to the overall low evaluation scores on the test set of the check-worthiness subtask in Turkish, we analyzed some of the incorrectly predicted results of our best model. Out of a total of 67 misclassified tweets, there were 5 false negative and 62 false positive instances.

We checked the false negatives for clues to improve recall. In one example tweet, a well-known Turkish person is mentioned with a mention tag. The tweet also contains the use of the quotation sign and the last sentence ends with a question mark. It can be interpreted as containing a claim, with the author exhibiting a skeptical distance from that claim. There were also examples in which an exclamation mark was placed in two parentheses, signifying sarcastic use, and suffixes were used to compare two opposite situations. We also found cases where the masses were tried to be mobilized around a claim with the words of the address.

In the false positive samples, on the other hand, there was a large number of tweets which are difficult to classify. In our manual re-evaluation, we found sentences that could be reclassified as checkworthy claims. It can be seen that quotations which specify the source are frequently used to strengthen statements that can be very dangerous, as in the following example:

(1) *Prof. Serhat Fındık: Hindistan Covid i aşılamayı bırakıp, İvermectin'e geçerek yendi. Afrika da aynı şekilde. İvermectin çok ucuz bir ilaçtır. Küresel ilaç şirketleri ucuz ilaçları sevmezler.* 'Prof. Serhat Fındık: India defeated Covid by stopping the vaccine and switching to Ivermectin, likewise in Africa. Ivermectin is a very inexpensive drug. Global pharmaceutical companies do not like cheap drugs.'

The tweet in (1) is judged "non-checkworthy", even though in our view it does contain several checkworthy claims (mixed with opinions). The high rate of such gray area cases in the Turkish test data could partially explain the extremely low scores across all systems submitted for this task.

6. Conclusion and Future Work

We have described our system for the CLEF 2022 CheckThat! Lab Task 1. We tackled subtasks 1a, 1b and 1c on check-worthiness, claim detection, and harmful tweet detection, in both English and Turkish. We experimented with four different transformer-based architectures as well as an ELMo-based attention network ensemble. We also tried different methods of pre-processing, data augmentation and included a number of linguistic features. We placed 6^{th}, 6^{th} and 9^{th} for the English data and 1^{st}, 1^{st}, and 2^{nd} for Turkish for the three subtasks. During this trial-and-error process, we realized that transformer based models already capture more comprehensive linguistic features than those we included in the system.

In the future, we plan to investigate more adapted and task-specific linguistic features, especially since transformer models rely on large amounts of training text which are not available for the majority of the world's languages. Additionally, we will examine what features are most relevant for our problem for designing a more interpretable model.

Acknowledgments

This work was partially funded by the German Federal Ministry of Education and Research (BMBF) in the project "NoFake: AI assisted system for the crowdsourcing based detection of disinformation spread via digital platforms" (16KIS1518K).

References

[1] P. Nakov, A. Barrón-Cedeño, G. Da San Martino, F. Alam, J. M. Struß, T. Mandl, R. Míguez, T. Caselli, M. Kutlu, W. Zaghouani, 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. Nørvåg, V. Setty (Eds.), Advances in Information Retrieval, Springer International Publishing, Cham, 2022, pp. 416–428.

[2] P. Nakov, A. Barrón-Cedeño, G. Da San Martino, F. Alam, J. M. Struß, T. Mandl, R. Míguez, T. Caselli, M. Kutlu, W. Zaghouani, 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.

[3] P. Nakov, A. Barrón-Cedeño, G. Da San Martino, F. Alam, R. Míguez, T. Caselli, M. Kutlu, W. Zaghouani, 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 andd Ferro, A. Hanbury, M. Potthast (Eds.), Working Notes of CLEF 2022—Conference and Labs of the Evaluation Forum, CLEF '2022, Bologna, Italy, 2022.

[4] X. Zhou, R. Zafarani, A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities, ACM Computing Surveys 53 (2020) 1–40. doi:10.1145/3395046. arXiv:1812.00315.

[5] I. Augenstein, Towards Explainable Fact Checking, arXiv:2108.10274 [cs, stat] (2021). arXiv:2108.10274.

[6] Z. Guo, M. Schlichtkrull, A. Vlachos, A Survey on Automated Fact-Checking, Transactions of the Association for Computational Linguistics 10 (2022) 178–206. doi:10.1162/tacl_a_00454.

[7] S. Shaar, F. Haouari, W. Mansour, M. Hasanain, N. Babulkov, F. Alam, P. Nakov, Overview of the CLEF-2021 CheckThat! Lab Task 2 on Detecting Previously Fact-Checked Claims in Tweets and Political Debates, in: CEUR Workshop Proceedings, Bucharest, Romania, 2021, p. 13.

[8] S. Shaar, M. Hasanain, B. Hamdan, Z. S. Ali, F. Haouari, A. Nikolov, M. Kutlu, Y. S. Kartal, F. Alam, J. Beltrán, T. Elsayed, P. Nakov, Overview of the CLEF-2021 CheckThat! Lab Task 1 on Check-Worthiness Estimation in Tweets and Political Debates, in: CEUR Workshop Proceedings, Bucharest, Romania, 2021, p. 24.

[9] N. Hassan, G. Zhang, F. Arslan, J. Caraballo, D. Jimenez, S. Gawsane, S. Hasan, M. Joseph, A. Kulkarni, A. K. Nayak, V. Sable, C. Li, M. Tremayne, ClaimBuster: The first-ever end-to-end fact-checking system, Proceedings of the VLDB Endowment 10 (2017) 1945–1948. doi:10.14778/3137765.3137815.

[10] A. Tchechmedjiev, P. Fafalios, K. Boland, M. Gasquet, M. Zloch, B. Zapilko, S. Dietze, K. Todorov, ClaimsKG: A Knowledge Graph of Fact-Checked Claims, in: C. Ghidini, O. Hartig, M. Maleshkova, V. Svátek, I. Cruz, A. Hogan, J. Song, M. Lefrançois, F. Gandon (Eds.), The Semantic Web – ISWC 2019, Lecture Notes in Computer Science, Springer International Publishing, Cham, 2019, pp. 309–324. doi:10.1007/978-3-030-30796-7_20.

[11] M. Hasanain, R. Suwaileh, T. Elsayed, A. Barron-Cedeno, P. Nakov, Overview of the CLEF-2019 CheckThat! Lab: Automatic Identification and Verification of Claims. Task 2: Evidence and Factuality, in: CEUR Workshop Proceedings, Lugano, Switzerland, 2019,

p. 15.

[12] P. Atanasova, P. Nakov, G. Karadzhov, M. Mohtarami, Overview of the CLEF-2019 Check-That! Lab: Automatic Identification and Verification of Claims. Task 1: Check-Worthiness, in: CEUR Workshop Proceedings, Lugano, Switzerland, 2019, p. 15.

[13] S. Shaar, A. Nikolov, N. Babulkov, F. Alam, A. Barron-Cedeno, T. Elsayed, M. Hasanain, R. Suwaileh, F. Haouari, Overview of CheckThat! 2020 English: Automatic Identification and Verification of Claims in Social Media, in: CEUR Workshop Proceedings, Thessaloniki, Greece, 2020, p. 24.

[14] J. R. Martinez-Rico, J. Martinez-Romo, L. Araujo, NLP\&IR@UNED at CheckThat! 2021: Check-worthiness estimation and fake news detection using transformer models, in: CEUR Workshop Proceedings, Bucharest, Romania, 2021, p. 13.

[15] A. Chernyavskiy, D. Ilvovsky, P. Nakov, Aschern at CheckThat! 2021: Lambda-Calculus of Fact-Checked Claims, in: CEUR Workshop Proceedings, Bucharest, Romania, 2021, p. 10.

[16] J. W. Pennebaker, M. E. Francis, R. J. Booth, Linguistic inquiry and word count: LIWC 2015, Pennebaker Conglomerates (2015).

[17] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, arXiv:1810.04805 [cs] (2019). arXiv:1810.04805.

[18] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, J. Gao, Deep learning–based text classification: a comprehensive review, ACM Computing Surveys (CSUR) 54 (2021) 1–40.

[19] D. Q. Nguyen, T. Vu, A. Tuan Nguyen, BERTweet: A pre-trained language model for English Tweets, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online, 2020, pp. 9–14. doi:10.18653/v1/2020.emnlp-demos.2.

[20] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised Cross-lingual Representation Learning at Scale, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 8440–8451. doi:10.18653/v1/2020.acl-main.747.

[21] Z.-H. Jiang, W. Yu, D. Zhou, Y. Chen, J. Feng, S. Yan, ConvBERT: Improving BERT with Span-based Dynamic Convolution, in: Advances in Neural Information Processing Systems, volume 33, Curran Associates, Inc., 2020, pp. 12837–12848.

[22] K. Clark, M.-T. Luong, Q. V. Le, C. D. Manning, ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators, 2020. arXiv:2003.10555.

[23] B. W. Lee, Y. S. Jang, J. Lee, Pushing on Text Readability Assessment: A Transformer Meets Handcrafted Linguistic Features, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2021, pp. 10669–10686. doi:10.18653/v1/2021.emnlp-main.834.

[24] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep Contextualized Word Representations, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New

Orleans, Louisiana, 2018, pp. 2227–2237. doi:10.18653/v1/N18-1202.