CoulterOzler at CheckThat! 2022: Detecting fake news with transformers

Kadir Bulut Ozler, Riah Coulter

University of Arizona

Abstract

In the age of the internet, people interact with each other more often than ever. Almost everybody with internet access has an affiliation with a social media website. With this popularity, spreading of misinformation has inevitably become a huge problem of the current age. In recent years, 2016 US Presidential Election brought the attention to fake news. With the Coronavirus Pandemic misinformation became an increasingly popular area to research in academia. To be a part of the research on detecting misinformation in the internet, we participated in task 3: Fake News Detection of the Checkthat! Lab at CLEF2022. In this paper, we show the details of our system consisting of data collection, transformer based pre-trained models and extensive preprocessing methods. We achieved an F1-score (macro) of 0.328 against a top score of 0.339 on the official test set.

Keywords

Fake News, Multi-class Classification, Transformers, Fine-tuning, Misinformation

1. Introduction

In the information age, internet became the main source of news on what is happening in the world. Individual access to the internet became easy and affordable which gave people a massive freedom to obtain and share information online. Although there have been major benefits of this freedom, it often came with a cost that is called misinformation. Misinformation is seen in variety of forms [1]. It can be a Facebook post with fake content, a tweet from a fake profile of a credible source, a news article that has a manipulative narrative or a misleading title that tells a different story in the article.

In recent years, misinformation became a significant research area in natural language processing. Some of the past studies focused on rumor detection [2, 3, 4, 5, 6], fake news detection [7, 8, 9, 10, 11, 12], spam detection [13, 14, 15, 16, 17] and bot detection [18, 19, 20, 21]. There have been several shared tasks related to misinformation detection. Recent SemEval tasks [22, 23, 24] aimed to question stance and veracity of given texts and categorize them to pre-defined classes. MediaEval [25] focused on misinformation regarding to Coronavirus Pandemic and 5G.

Task 3 of the Checkthat! Lab at CLEF2022 [26, 27, 28] is another shared task that focuses on fake news detection. The task's goal is to determine if the claim of the article belongs to following categories [29] : true, partially true, false, or other (label descriptions can be found

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

kbozler@email.arizona.edu (K. B. Ozler); riahcoulter@email.arizona.edu (R. Coulter)

Table 1
Task's label descriptions

Label	Description
False	The main claim made in an article is untrue.
Partially False	The main claim of an article is a mixture of true and false information.
True	The primary elements of the main claim are demonstrably true.
Other	The claim is open to discussion regarding its misinformation status.

in table 1). The task has 2 sub-tasks: mono-lingual in English and cross-lingual for English and German where the training set is in English and the test set is in German. This year, we participated in mono-lingual sub-task. In the previous version of the lab [30], the sub-task 3A [31] is very similar to the mono-lingual sub-task of this year's task 3 of the shared task. For that sub-task 3A, there have been many different approaches from the participants. [32] employed several transformer based [33] models and got their best results with Albert [34]. [35] used an ensemble of Roberta [36] and Longformer [37]. [38] showed that gradient boosting with extensive preprocessing performed better than widely popular deep learning architectures such as LSTMs [39] and BERT [40].

In the following sections, we show the features of the datasets we used in this work, our methods, experiments, results, error analysis and our conclusion.

2. Dataset Analysis

There are multiple datasets with mixed domains that focus on fake news detection. In this section, we show features of the used datasets in our work. In table 2, you can find the label count for all the datasets below. Label counts are calculated after dropping NaN values, bad lines and duplicates except for official test set. In table 3, you can find the final distribution of labels in the training set, dev set and test set that have been used in our work.

Table 2
Label details of each dataset

Dataset	# of samples	False	Partially False	True	Other
politifact	21898	4699	14311	2888	-
true-fake	44689	23478	-	21211	-
fakenewskdd2020	1068	434	-	634	-
official-training-set	1183	571	312	206	94
official-test-set	612	315	56	210	31

2.1. politifact

Introduced in [41], this dataset consists of fact-checking articles from politifact.com. It includes article title and article text. The available labels are true, false, partially false. The exact version

Table 3
Label details of final train-dev-test distribution

Dataset	# of samples	False	Partially False	True	Other
training set	46855	19870	10032	16891	62
dev set	396	191	104	69	32
test set	612	315	56	210	31

we used in our work can be found at Kaggle. We randomly chose 15k of the samples and included them in our training set.

2.2. true-fake

This combined dataset consists of 2 seperate datasets, each includes article titles and article texts. In true dataset, all samples are considered as true, in fake dataset all samples are considered as fake. We randomly chose 30k of the samples and included them in our training set. The exact version we used in our work can be found at here and here. Unfortunately we could not find the original source that introduced these datasets.

2.3. fakenewskdd2020

This dataset has only article texts and their labels. Fake label is defined as "potentially unreliable". We randomly chose 1k of the samples and included them in our training set. The dataset can be found from Kaggle and it was provided by Kai Shu to the competition organizers.

2.4. official-training-set

This is the official dataset that is released by task organizers privately for the participants [42]. It contains article text, article title, and all 4 labels. 2/3 of it has been used in our training set, and 1/3 of it has been used in our dev set during model development phase. It was built by following steps in [43].

2.5. official-test-set

This is official test set released by task organizers. It contains article id, article text, and all 4 labels. It can be found at Zenodo. We calculated our final scores based on our predictions on this dataset.

3. Methods

In this section, we give details about the methods we employed for the task. They consist of text preprocessing and fine-tuning pre-trained language models.

3.1. Preprocessing

- **Concatenating title and content when available**: In some datasets, there exists article title column that contains the title of the articles. In this case, we merged title and article content in one sequence of text.
- **Converting to lower case**: It is usually unhelpful to keep the characters in both lowercase and uppercase form.
- **Removing stop words**: Stop words are usually the most frequently occurring words in natural language and they do not contribute much to the meaning. It's a widely used practice to remove them from text before feeding the sequences to the model.
- **Removing punctuation**: As stop words, punctuation marks are usually unnecessary to keep.
- **Standardizing certain words**: In order to make the text as clear as possible, we used some pre-defined tokens to replace urls, email addresses, phone numbers, names, numbers, digits and currency characters.
- **Lemmatizing**: In order to simplify the words we used lemmatizing over stemming to avoid creating words that are not in the dictionary or lost their meaning.
- **Shortening**: We shortened the sequences to 500 tokens and 4000 tokens in order to fit them into the models, depending on the model's capacity.

We used the Natural Language Toolkit (NLTK) [44] for lemmatizing, name standardization, and stop words removal, unicodedata[1] for punctuation removal, and the clean-text project[2] for converting to lower case and standardizing urls, email addresses, phone numbers, numbers, digits and currency characters.

3.2. Fine-tuning LMs

Transformer based pre-trained language models have become significantly popular in recent years due to the fact that they led to state of the art improvements in many natural language processing tasks [40]. They also do not require in domain training from scratch which would need more data and more GPU time. Therefore, we decided to go with fine-tuning a pre-trained language model to develop our in domain model for this shared task. We used Huggingface's transformers [45] during this stage. The code repository has been shared[3]. We explored multiple LMs, training/eval batch size, number of epochs and learning rate. We chose distilbert [46] because it is a smaller, yet promising model. We also chose longformer anticipating that if the model is fed long sequences (which is normal for articles), the predictions could be more accurate. Our hyperparameter space can be found in table 4.

4. Experiments and Results

In the model development stage, we used our custom split of training and dev sets that are explained in section 2. We used 4 32GB Nvidia V100 GPUs. After the initial experiments on dev

[1] https://docs.python.org/3/library/unicodedata.html
[2] https://pypi.org/project/clean-text/
[3] https://github.com/kbulutozler/clef2022-checkthat-task3

Table 4
Hyperparameter space

model	distilbert-base-uncased, allenai/longformer-base-4096
# of epochs	4, 12, 16, 20, 32
learning rate	2e-05, 5e-05
batch size	2, 4, 16, 64

set, we decided to choose 2e-05 as learning rate, 64 as batch size to focus more on number of epochs for the rest of the experiments. For the longformer model, we had to reduce the batch size to 2 due to GPU limitations. The results obtained in the development stage can be found in table 5 with metrics micro f1 and macro f1.

Table 5
Results on custom dev set

model	# of epochs	micro f1	macro f1
longformer-base-4096	16	0.482	0.163
longformer-base-4096	32	0.487	0.177
distilbert-base-uncased	4	0.477	0.364
distilbert-base-uncased	12	0.495	0.381
distilbert-base-uncased	16	0.515	0.399
distilbert-base-uncased	20	**0.538**	**0.408**

We can sum up our findings during model development as follows:

- It can be said that using longformer did not lead to the results we anticipated.
- Longer training had diminishing returns.
- Macro f1 scores are lower than micro f1 scores because the performance on certain label(s) is significantly worse.

As seen from Table 5, best model setup is obtained by distilbert model trained for 20 epochs with batch size of 64 and learning rate of 2e-05. Furthermore, we explored how the model would perform with no additional data. We used the best model setup and trained on just official training set with no hyperparameter search. Apart from that, we explored the effect of preprocessing methods. For this, we used the same setup and trained on just official training set with no preprocessing. In table 6, we show the results we obtained on official test set in terms of accuracy, macro precision, macro recall, macro f1 metrics.

Results show that additional data hurt the performance the most. One cause might be the difference in domains of the official dataset and additional datasets. We anticipated that for sequence classification, combining multiple domains might lead the model to make better predictions as shown in [47], however we couldn't obtain parallel results following similar intuition. Moreover, training on preprocessed data led to better precision, recall and f1 scores in comparison to training on unpreprocessed data.

Table 6
Results on official test set

preprocessing	training set	accuracy	precision	recall	f1
yes	official data + additional data	0.462	0.327	0.295	0.262
yes	official data	0.451	**0.345**	**0.359**	**0.328**
no	official data	**0.464**	0.337	0.318	0.299

5. Error Analysis

In this section we present the confusion matrix on official test set for the model that was trained on official training set after preprocessing (second model in table 6) in table 7. The table shows the model managed to learn the most detecting false claims and struggled to see other labels. This can be explained by the challenging nature of the data and the dominance of "False" label in the training set.

Table 7
Confusion matrix

gold label	count	False	Partially False	True	Other
False	315	204	48	36	27
Partially False	56	20	13	17	6
True	210	72	68	49	21
Other	31	13	5	3	10

6. Conclusion and Future Scope

We participated in task 3: Fake News Detection of the Checkthat! Lab at CLEF2022 and developed models to detect and classify misinformation in the internet. We applied extensive preprocessing methods and fine-tuned several pre-trained language models with the released dataset and additional datasets. We found that being able to feed longer sequences and additional data with mixed domains did not improve performance, preprocessing and smaller model led to better predictions.

For the future work, one might explore extra pre-training an already pre-trained language model with data that has similar nature to the official training set to develop better models. In this task, class imbalance seems to be a significant issue. Therefore another direction might be exploring data augmentation methods to increase in domain data or modifying loss function to increase penalty for misprediction of samples of the underrepresented label(s).

References

[1] D. Bawden, L. Robinson, The dark side of information: overload, anxiety and other paradoxes and pathologies, Journal of Information Science 35 (2009) 180 – 191.

[2] J. Yu, J. Jiang, L. M. S. Khoo, H. L. Chieu, R. Xia, Coupled hierarchical transformer for stance-aware rumor verification in social media conversations, Association for Computational Linguistics, 2020.
[3] S. Kwon, M. Cha, K. Jung, Rumor detection over varying time windows, PloS one 12 (2017) e0168344.
[4] Q. Zhang, S. Zhang, J. Dong, J. Xiong, X. Cheng, Automatic detection of rumor on social network, in: Natural Language Processing and Chinese Computing, Springer, 2015, pp. 113–122.
[5] S. Hamidian, M. T. Diab, Rumor detection and classification for twitter data, arXiv preprint arXiv:1912.08926 (2019).
[6] T. Takahashi, N. Igata, Rumor detection on twitter, in: The 6th International Conference on Soft Computing and Intelligent Systems, and The 13th International Symposium on Advanced Intelligence Systems, IEEE, 2012, pp. 452–457.
[7] V. Pérez-Rosas, B. Kleinberg, A. Lefevre, R. Mihalcea, Automatic detection of fake news, arXiv preprint arXiv:1708.07104 (2017).
[8] J. C. Reis, A. Correia, F. Murai, A. Veloso, F. Benevenuto, Supervised learning for fake news detection, IEEE Intelligent Systems 34 (2019) 76–81.
[9] K. Shu, A. Sliva, S. Wang, J. Tang, H. Liu, Fake news detection on social media: A data mining perspective, ACM SIGKDD explorations newsletter 19 (2017) 22–36.
[10] R. K. Kaliyar, A. Goswami, P. Narang, Fakebert: Fake news detection in social media with a bert-based deep learning approach, Multimedia tools and applications 80 (2021) 11765–11788.
[11] C. Liu, X. Wu, M. Yu, G. Li, J. Jiang, W. Huang, X. Lu, A two-stage model based on bert for short fake news detection, in: International Conference on Knowledge Science, Engineering and Management, Springer, 2019, pp. 172–183.
[12] J. C. B. Cruz, J. A. Tan, C. Cheng, Localization of fake news detection via multitask transfer learning, arXiv preprint arXiv:1910.09295 (2019).
[13] A. Gupta, R. Kaushal, Improving spam detection in online social networks, in: 2015 International conference on cognitive computing and information processing (CCIP), IEEE, 2015, pp. 1–6.
[14] T. Wu, S. Liu, J. Zhang, Y. Xiang, Twitter spam detection based on deep learning, in: Proceedings of the australasian computer science week multiconference, 2017, pp. 1–8.
[15] G. Jain, M. Sharma, B. Agarwal, Spam detection on social media using semantic convolutional neural network, International Journal of Knowledge Discovery in Bioinformatics (IJKDB) 8 (2018) 12–26.
[16] G. Jain, M. Sharma, B. Agarwal, Optimizing semantic lstm for spam detection, International Journal of Information Technology 11 (2019) 239–250.
[17] J. Cao, C. Lai, A bilingual multi-type spam detection model based on m-bert, in: GLOBECOM 2020-2020 IEEE Global Communications Conference, IEEE, 2020, pp. 1–6.
[18] M. Heidari, J. H. Jones, Using bert to extract topic-independent sentiment features for social media bot detection, in: 2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), IEEE, 2020, pp. 0542–0547.
[19] S. Feng, Z. Tan, R. Li, M. Luo, Heterogeneity-aware twitter bot detection with relational graph transformers, arXiv preprint arXiv:2109.02927 (2021).

[20] D. Martín-Gutiérrez, G. Hernández-Peñaloza, A. B. Hernández, A. Lozano-Diez, F. Álvarez, A deep learning approach for robust detection of bots in twitter using transformers, IEEE Access 9 (2021) 54591–54601.

[21] M. Heidari, S. Zad, P. Hajibabaee, M. Malekzadeh, S. HekmatiAthar, O. Uzuner, J. H. Jones, Bert model for fake news detection based on social bot activities in the covid-19 pandemic, in: 2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), IEEE, 2021, pp. 0103–0109.

[22] G. Da San Martino, A. Barrón-Cedeno, H. Wachsmuth, R. Petrov, P. Nakov, Semeval-2020 task 11: Detection of propaganda techniques in news articles, in: Proceedings of the fourteenth workshop on semantic evaluation, 2020, pp. 1377–1414.

[23] T. Mihaylova, G. Karadjov, P. Atanasova, R. Baly, M. Mohtarami, P. Nakov, Semeval-2019 task 8: Fact checking in community question answering forums, arXiv preprint arXiv:1906.01727 (2019).

[24] G. Gorrell, E. Kochkina, M. Liakata, A. Aker, A. Zubiaga, K. Bontcheva, L. Derczynski, Semeval-2019 task 7: Rumoureval, determining rumour veracity and support for rumours, in: Proceedings of the 13th International Workshop on Semantic Evaluation, 2019, pp. 845–854.

[25] K. Pogorelov, D. T. Schroeder, L. Burchard, J. Moe, S. Brenner, P. Filkukova, J. Langguth, Fakenews: Corona virus and 5g conspiracy task at mediaeval 2020, in: MediaEval 2020 Workshop, 2020.

[26] 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.

[27] 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.

[28] J. Köhler, G. K. Shahi, J. M. Struß, M. Wiegand, M. Siegel, T. Mandl, M. Schütz, Overview of the CLEF-2022 CheckThat! lab task 3 on fake news detection, in: Working Notes of CLEF 2022—Conference and Labs of the Evaluation Forum, CLEF '2022, Bologna, Italy, 2022.

[29] G. K. Shahi, A. Dirkson, T. A. Majchrzak, An exploratory study of covid-19 misinformation on twitter, Online Social Networks and Media 22 (2021) 100104.

[30] P. Nakov, G. Da San Martino, T. Elsayed, A. Barrón-Cedeño, R. Míguez, S. Shaar, F. Alam, F. Haouari, M. Hasanain, W. Mansour, B. Hamdan, Z. S. Ali, N. Babulkov, A. Nikolov, G. K. Shahi, J. M. Struß, T. Mandl, M. Kutlu, Y. S. Kartal, Overview of the CLEF-2021 CheckThat! lab on detecting check-worthy claims, previously fact-checked claims, and fake news, in: Proceedings of the 12th International Conference of the CLEF Association:

Information Access Evaluation Meets Multiliguality, Multimodality, and Visualization, CLEF '2021, Bucharest, Romania (online), 2021. URL: https://link.springer.com/chapter/10.1007/978-3-030-85251-1_19.

[31] G. K. Shahi, J. M. Struß, T. Mandl, Overview of the CLEF-2021 CheckThat! lab task 3 on fake news detection, in: Working Notes of CLEF 2021—Conference and Labs of the Evaluation Forum, CLEF '2021, Bucharest, Romania (online), 2021. URL: http://ceur-ws.org/Vol-2936/paper-30.pdf.

[32] J. R. Martinez-Rico, J. Martinez-Romo, L. Araujo, L.: Nlp&ir@ uned at checkthat! 2021: check-worthiness estimation and fake news detection using transformer models, Faggioli et al.[33] (2021).

[33] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[34] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut, Albert: A lite bert for self-supervised learning of language representations, arXiv preprint arXiv:1909.11942 (2019).

[35] H. Lekshmiammal, A. K. Madasamy, Nitk _ nlp at checkthat! 2021: Ensemble transformer model for fake news classification, in: Conference and Labs Ofthe Evaluation Forum (CLEF 2021), 2021.

[36] 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, arXiv preprint arXiv:1907.11692 (2019).

[37] I. Beltagy, M. E. Peters, A. Cohan, Longformer: The long-document transformer, arXiv preprint arXiv:2004.05150 (2020).

[38] C. G. Cusmuliuc, M. A. Amarandei, I. Pelin, V. I. Cociorva, A. Iftene, Uaics at checkthat! 2021: fake news detection, Faggioli et al.[33] (2021).

[39] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (1997) 1735–1780.

[40] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[41] N. Vo, K. Lee, Where are the facts? searching for fact-checked information to alleviate the spread of fake news, arXiv preprint arXiv:2010.03159 (2020).

[42] G. K. Shahi, J. M. Struß, T. Mandl, Overview of the clef-2021 checkthat! lab task 3 on fake news detection, Working Notes of CLEF (2021).

[43] G. K. Shahi, Amused: An annotation framework of multi-modal social media data, arXiv preprint arXiv:2010.00502 (2020).

[44] S. Bird, E. Klein, E. Loper, Natural language processing with Python: analyzing text with the natural language toolkit, " O'Reilly Media, Inc.", 2009.

[45] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al., Huggingface's transformers: State-of-the-art natural language processing, arXiv preprint arXiv:1910.03771 (2019).

[46] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, arXiv preprint arXiv:1910.01108 (2019).

[47] K. B. Ozler, K. Kenski, S. Rains, Y. Shmargad, K. Coe, S. Bethard, Fine-tuning for multi-

domain and multi-label uncivil language detection, in: Proceedings of the Fourth Workshop on Online Abuse and Harms, 2020, pp. 28–33.