BUM at CheckThat! 2022: A Composite Deep Learning Approach to Fake News Detection using Evidence Retrieval

David La Barbera[1,†], Kevin Roitero[1,†], Joel Mackenzie[2,†], Damiano Spina[3], Gianluca Demartini[2] and Stefano Mizzaro[1]

[1]*University of Udine, Via Delle Scienze 206, Udine, 33100, Italy*
[2]*The University of Queensland, St Lucia QLD 4072, Australia*
[3]*RMIT University, 124 La Trobe St, Melbourne VIC 3000, Australia*

Abstract

We detail a deep learning approach based on the transformer architecture for performing fake news detection. The proposed approach is composed of a deep learning network which receives as input the claim to be verified, a series of predictions made by other models, and supporting evidence in the form of ranked passages. We validate our approach participating as the Brisbane–Udine–Melbourne (BUM) Team in the CLEF2022-CheckThat! Lab (Task 3: Fake News Detection), where we achieve an F1-score of 0.275, ranking 10th out of 25 participants.[1]

Keywords

fake news, fact-checking, deep learning, information retrieval

1. Introduction

The increasing popularity of the internet, particularly social networks, has been accompanied by the spread of fake news. Due to the huge amount of data that people produce and share every day, specialized human fact-checkers struggle to keep up with manually annotating and validating such data. Therefore, researchers and practitioners are investing significant resources to develop automated approaches to support fact-checkers by identifying fake news in a fast and reliable way.

To address this important issue, the CLEF-2022 CheckThat! Lab [1], as done in the previous edition [2], has a task to develop systems that, given an article described by its text and title, are able to determine whether the main claim made in the article is true, partially true, false,

[1]https://docs.google.com/spreadsheets/d/1VE3jOUx-TziO8wZPHE9VS0YRQ_txnIUm/- [Accessed: 21 June 2022].

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

[†]Equal contribution.

labarbera.david@spes.uniud.it (D. La Barbera); kevin.roitero@uniud.it (K. Roitero); joel.mackenzie@uq.edu.au (J. Mackenzie); damiano.spina@rmit.edu.au (D. Spina); demartini@acm.org (G. Demartini); mizzaro@uniud.it (S. Mizzaro)

0000-0002-8215-5502 (D. La Barbera); 0000-0002-9191-3280 (K. Roitero); 0000-0001-7992-4633 (J. Mackenzie); 0000-0001-9913-433X (D. Spina); 0000-0002-7311-3693 (G. Demartini); 0000-0002-2852-168X (S. Mizzaro)

or other (e.g., claims in dispute). To support this task (Task 3, English), the CheckThat! Lab released a database made up to 1,300 unique statements and their labels.

In this paper we present our approach to the aforementioned task as follows. In Section 2, we briefly describe the purpose of the challenge. In Section 3, we detail supplemental data that we have employed to expand the available dataset. Then, in Section 4, we describe our approach to the task: we build a deep learning pipeline relying on (i) a BERT model trained on external additional data; (ii) a T5 transformer to perform entailment for each statement with the top evidence found with information retrieval models; and (iii) we use those additional data combined together to form a novel dataset to predict the final score. Finally, Section 5 concludes the report.

2. Task 3: Fake News Detection

The goal of *Task 3: English fake news detection* is to label the truthfulness of English news articles based on four truthfulness levels defined as follows:

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. The article contains partially true and partially false information but cannot be considered 100% true;

True: This rating indicates that the primary elements of the main claim are demonstrably true;

Other: An article that cannot be categorized as true, false, or partially false due to a lack of evidence about its claims. This category includes articles in dispute and unproven articles.

The available training dataset contains 1,264 different articles with the respective title, body text, and truthfulness labels.

3. Expanding the Dataset

In this Section we describe the additional data that we use to train our models, and the retrieval techniques used to find adequate evidence for the original training set described above.

3.1. Additional Data

To train part of our models we rely on additional data suggested by Shahi et al. [3]; we manually annotate the data to map the ground truth scale to the same scale used in this challenge. We adapt the following datasets, keeping only the columns related to the title, text, and ground truth of the claim:

- We use the *Fake News Detection Challenge KDD 2020* dataset[1] which is made by 4,987 unique *true* or *false* statements;

[1] https://www.kaggle.com/c/fakenewskdd2020/overview [Accessed: 21 June 2022].

- We use part of the public *Fakenews Classification Datasets*[2] on Kaggle, such as the *Emergent Phase2 2018*, consisting of 90 statements, where we map the statements with ground truth set as *Unverified* to *other*; the *Fake* dataset, consisting of 23,481 *false* statements; the *True* dataset, consisting of 21,417 *true* statements; and the *Snopes* dataset, consisting of 4,745 statements for which we map the original *mostly true, mostly false, misattributed, miscaptioned, and mixture* labels to partially false, the *scam and legend* labels as false, and leave the remaining labels unchanged. All other data was unused, since it was not clear how to properly adapt it for this challenge.
- From the public Kaggle repository FakeNewsNet [4, 5, 6],[3] we use the *BuzzFeed* dataset, which consists of 91 *false* statements and 91 *true* statements.
- We use 120 statements from the *PolitiFact* dataset, originally set by Wang [7], which we have used in prior work [8, 9, 10]. We adapt the ground truth by setting the *pants on fire* statements to *false*, the *barely true, half true, and mostly true* statements to *partially false*, and leaving the others unchanged. From that same work by Roitero et al. [8], Soprano et al. [9], La Barbera et al. [10], we use 60 statements from the *ABC* dataset, mapping the ground truth from *negative* to *false*, *in between* to *partially false*, and *positive* to *true*.
- We use the *FEVER* dataset [11], consisting of 145,449 unique statements, by mapping the ground truth from *supports* to *true*, *refutes* to *false*, and setting the remaining statements as *other*.
- Finally, we also use the data from Jiang et al. [12], consisting of 18,171 statements, in which we map the *supported* ground truth to *true*, and set the others to *false*.

Combining these individual collections results in a novel dataset made up of 213,715 additional statements with the same ground truth scale as the one made available for the challenge and described in Section 2.

3.2. Evidence Retrieval

To further enrich the data available for the classification task, we employed Wikipedia as a source of evidence. In particular, we used the WikiExtractor[4] tool to extract documents from Wikimedia's XML dump of the English segment of Wikipedia.[5] After extracting the raw documents, we also created a passage-level representation of the data. Since we had both document-level and passage-level representations, we created two separate indexes with the Lucene-based Anserini system [13]. In total, the document index consisted of 6.5 million documents, and the passage index contained 49.2 million passages; both indexes were built from around 15 GiB of raw Wikipedia text data. We used the title of each article as a query, and retrieved the top-k documents/passages using a simple bag-of-words BM25 model [14],

[2] https://www.kaggle.com/datasets/liberoliber/onion-notonion-datasets [Accessed: 21 June 2022].
[3] https://www.kaggle.com/datasets/mdepak/fakenewsnet [Accessed: 21 June 2022].
[4] https://github.com/attardi/wikiextractor [Accessed: 21 June 2022].
[5] Dump from April 2, 2022. https://dumps.wikimedia.org/enwiki/latest/enwiki-latest-pages-articles.xml.bz2 [Accessed: 21 June 2022].

Table 1
Two examples of passage-based evidence retrieval. The top example shows successful retrieval, and the bottom example shows a failure.

Title: Australian Authorities: Arsonists to Blame for Bushfires - NOT Climate Change **Rating:** False
Passage 1 Social media accounts, including Donald Trump Jr's Twitter account, circulated the false claim that 183 people had been arrested for arson during the Australian fire crisis. In 2021, the Australian Press Council determined the news report that 183 arsonists had been arrested "was not misleading". These 183 people were subject to legal action, but only 24 for "deliberately-lit bushfires". An opinion piece for "The Conversation" website stated "In the first week of 2020, hashtag #ArsonEmergency became the focal point of a new online narrative surrounding the bushfire crisis. The message: the cause is arson, not climate change. Police and bushfire services (and some journalists) have contradicted this claim [...] We have observed both troll and bot accounts spouting disinformation regarding the bushfires on Twitter". The article also argued that a disinformation was underway to downplay the role of climate change in causing the fires. The vice.com website wrote "Research conducted by the Queensland University of Technology showed that Twitter accounts with the characteristics of bots or trolls were spreading disinformation about the responsibility of arsonists and Greens". "The Guardian" accused News Corp of furthering arson disinformation.
Title: Energy secretary warns of £500m 'electric shock' after Brexit **Rating:** False
Passage 1 A regenerative shock absorber is a type of shock absorber that converts parasitic intermittent linear motion and vibration into useful energy, such as electricity. Conventional shock absorbers simply dissipate this energy as heat.

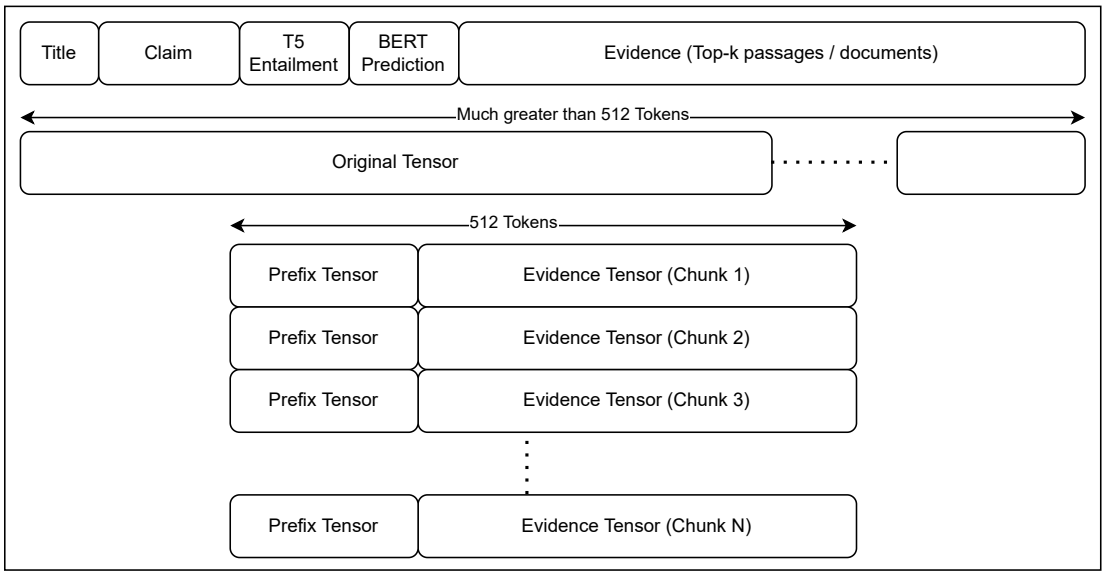
returning the full text of those top-k items for further processing. Table 1 shows two examples of passage-level evidence retrieval.

4. Proposed Approach

The rationale behind our approach is to use information retrieval techniques to provide evidence for each claim, and to use some models to perform preliminary operations on the data. This allows us to enrich the available dataset with further information which we hypothesize will increase the performance of the final classification. To this end, we build an informative textual string representation that can be fed to a transformer-based (i.e., BERT) model, consisting of:

- The title of the claim;
- The text of the claim;
- The prediction and the confidence of a BERT model trained on the external datasets;
- A T5 transformer used to perform entailment with the top ranking passage that was retrieved to find a justification for each claim using the Wikipedia index; and

Figure 1: Schema of the approach used to train the model with the combined evidence.



- The top-10 passages retrieved according to the retrieval strategy detailed above.

The idea is to use the "textification" approach, which has been successfully applied to tasks related to the medical domain, in particular to automatic encoding of diagnostic texts [15, 16, 17], as well as in human mobility forecasting [18].

All of the aforementioned information is combined for each statement to build up the final classification for the task. In the following subsection, we will describe each component of our composite model. We rely on PyTorch and Hugging Face to implement, fine-tune, and deploy our models.

4.1. BERT Classification Relying on External Data

The first component we use is a BERT model [19] to predict truthfulness of given statements. We use the additional data detailed in Section 3.1 to fine-tune a `bert-base-uncased` model[6] for three epochs on a classification task, thus using Cross-Entropy as loss function.

For all the models that we have used for our fake news detection pipeline, we rely on 10% of the statements from the original training set as a test set to validate the model performances. After the fine-tuning phase, our model achieves an accuracy of 0.779 and a macro-F1 score of 0.645.

At this point, we use this model to perform predictions on the original training set for the challenge. For each claim we than use the model prediction and its confidence to ensemble a string with the following format: "`class PREDICTED CLASS with confidence MODEL PREDICTION CONFIDENCE`".

[6] see https://huggingface.co/bert-base-uncased [Accessed: 21 June 2022].

4.2. T5 Transformer Entailment

The second component that we use to enrich the available information for each claim is a pre-trained T5 transformer which is used to perform entailment. In particular, we rely on the `t5-base`[7] model which has been already fine-tuned for the entailment task on multiple datasets (see Raffel et al. [20, Appendix]). After different attempts we decided to perform entailment by relying on the Recognizing Textual Entailment (RTE) task prefix and modality (we refer to the original paper [20] for an in depth explanation of each available modality, an example of the one used in this paper can be found in Appendix C.5 of that work), which we found to perform best. We choose to not rely on other T5 modalities other than RTE, since RTE was the most effective. We leave an in-depth study of the different T5 entailment modalities for fact-checking purposes to future work. Thus, to check if the claim is a logical consequence of the best evidence provided, we use the textual representation of the highly ranked passage from the retrieved evidence described in Section 3.2 and perform entailment for each claim in the test set. Finally, we store the model prediction results as evidence for the final classification.

4.2.1. BERT Classification Aggregation

We now describe the model that we develop to perform the final classification. Relying on the information available for each statement, along with the additional information that we have computed using both BERT and T5, we build a string for each claim containing its title, the claim, the T5 entailment prediction, and the BERT string described in Section 4.1. Moreover, to provide the model with the top-10 most relevant retrieved passages for each claim and overcome the maximum length for a given input for BERT (i.e. 512 tokens), we use the information listed above for each statement as a prefix. We then concatenate part of the evidence (i.e. part of the top-10 passages, from rank one to rank ten) until the size of 512 tokens is reached. We repeat this process until we have paired the result string with all of the evidence passages. This process is detailed in Figure 1. We use the model to compute its predictions for each string. Thus, we aggregate those predictions over the same statement (identified not only by an ID, but also from the prefix) using a majority vote (i.e., the mode function) to obtain the final prediction from our pipeline.

We decided on applying the aforementioned model as our submission to the challenge after trying several alternatives to padding and splitting on instances of 512 tokens. In fact, we did experiment with models implementing more efficient attention such as the Longformer [21] and Reformer [22] models, but they performed worse than the padded BERT model and exhibited unstable losses during training.

4.3. System Performance

The performance of our composite model is summarized in Table 2, where the first row shows the overall results of the system, and the subsequent rows show the results computed per ground truth label. The metric used to rank the systems for the challenge, the F1-score, was 0.275 for our model, leading us to rank 10^{th} overall for the task of fake news detection in the CLEF-2022

[7] see https://huggingface.co/t5-base [Accessed: 21 June 2022].

Table 2
Overall and per-label metric scores of the proposed system under the test set for the challenge. Submissions were ranked using the F1-score.

	Accuracy	Precision	Recall	F1-score
overall	.472	.301	.299	.275
false	–	.624	.781	.694
other	–	.060	.065	.062
partially false	–	.104	.214	.140
true	–	.414	.138	.207

CheckThat! Lab. Investigating the model performance by taking into account the measures for each label, we can see that we have much greater scores for the *false* label with respect to all of the other labels. This result is probably due to bias from the large amount of *false* statements in the training dataset. Thus, given more equally distributed training data, we believe that our system performance could improve significantly.

5. Conclusions

In this paper, we present our approach for the task of fake news detection for the CLEF-2022 CheckThat! Lab. We proposed a deep learning pipeline strongly relying on retrieval techniques to augment the training data, and textify the data for use within a BERT model. Our best run achieved the 10^{th} place on the English fake news classification dataset.

We believe that our solution is suitable for further improvements that could enhance the quality of the overall predictions. First of all, we aim to improve data quality, since we found that some claims were noisy. Also, we aim to include more statements from other datasets in the additional meta-dataset that we have built. We also aim to improve the quality of our evidence retrieval, by using models beyond bag-of-words, and by using other sources of information beyond Wikipedia. Further investigation on the fine-tuning of the T5 Transformer and alternative modalities or combinations thereof may further improve performance. Finally we also aim to test further alternatives with respect to those described and tested in Section 4.2.1, such as different textification and combination of models and methodologies such as those explored by Xue et al. [18] and Radford et al. [23].

In the future we will continue to improve this approach, as we believe that our system can contribute to the fight against misinformation by leveraging the complementary roles of evidence retrieval and the power of large language models.

References

[1] 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.

[2] 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.

[3] G. K. Shahi, J. M. Struß, T. Mandl, J. Köhler, M. Wiegand, M. Siegel, CT-FAN-22 corpus: A Multilingual dataset for Fake News Detection, 2022. doi:10.5281/zenodo.6508748.

[4] 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.

[5] K. Shu, S. Wang, H. Liu, Exploiting tri-relationship for fake news detection, arXiv preprint arXiv:1712.07709 (2017).

[6] K. Shu, D. Mahudeswaran, S. Wang, D. Lee, H. Liu, Fakenewsnet: A data repository with news content, social context and dynamic information for studying fake news on social media, arXiv preprint arXiv:1809.01286 (2018).

[7] W. Y. Wang, "liar, liar pants on fire": A new benchmark dataset for fake news detection, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 422–426. doi:10.18653/v1/P17-2067.

[8] K. Roitero, M. Soprano, S. Fan, D. Spina, S. Mizzaro, G. Demartini, Can the crowd identify misinformation objectively? The effects of judgment scale and assessor's background, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Association for Computing Machinery, New York, NY, USA, 2020, p. 439–448. doi:10.1145/3397271.3401112.

[9] M. Soprano, K. Roitero, D. La Barbera, D. Ceolin, D. Spina, S. Mizzaro, G. Demartini, The many dimensions of truthfulness: Crowdsourcing misinformation assessments on a multidimensional scale, Information Processing & Management 58 (2021) 102710. doi:10.1016/j.ipm.2021.102710.

[10] D. La Barbera, K. Roitero, D. Spina, S. Mizzaro, G. Demartini, Crowdsourcing Truthfulness: The Impact of Judgment Scale and Assessor Bias, Springer, New York, NY, USA, 2020, pp. 207–214. doi:https://doi.org/https://doi.org/10.1007/978-3-030-45442-5_26.

[11] J. Thorne, A. Vlachos, C. Christodoulopoulos, A. Mittal, FEVER: a large-scale dataset for fact extraction and VERification, 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. 809–819. doi:10.18653/v1/N18-1074.

[12] Y. Jiang, S. Bordia, Z. Zhong, C. Dognin, M. Singh, M. Bansal, HoVer: A dataset for many-hop fact extraction and claim verification, in: Findings of the Association for Computational Linguistics: EMNLP 2020, Association for Computational Linguistics, Online, 2020, pp. 3441–3460. URL: https://aclanthology.org/2020.findings-emnlp.309. doi:10.18653/v1/2020.findings-emnlp.309.

[13] P. Yang, H. Fang, J. Lin, Anserini: Reproducible ranking baselines using Lucene, Journal of Data and Information Quality 10 (2018) 1–20.

[14] S. E. Robertson, H. Zaragoza, The probabilistic relevance framework: BM25 and beyond, Foundations and Trends in Information Retrieval 3 (2009) 333–389.

[15] M. H. Popescu, K. Roitero, S. Travasci, V. Della Mea, Automatic assignment of ICD-10 codes

to diagnostic texts using transformers based techniques, in: 2021 IEEE 9th International Conference on Healthcare Informatics (ICHI), IEEE, 2021, pp. 188–192.

[16] K. Roitero, B. Portelli, M. H. Popescu, V. Della Mea, DiLBERT: Cheap embeddings for disease related medical NLP, IEEE Access 9 (2021) 159714–159723.

[17] V. Della Mea, M. H. Popescu, K. Roitero, Underlying cause of death identification from death certificates using reverse coding to text and a nlp based deep learning approach, Informatics in Medicine Unlocked 21 (2020) 100456.

[18] H. Xue, F. D. Salim, Y. Ren, C. L. Clarke, Translating human mobility forecasting through natural language generation, in: Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, 2022, pp. 1224–1233. doi:10.1145/3488560.3498387.

[19] 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), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. doi:10.18653/v1/N19-1423.

[20] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, Journal of Machine Learning Research 21 (2020) 1–67.

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

[22] N. Kitaev, L. Kaiser, A. Levskaya, Reformer: The efficient transformer, in: International Conference on Learning Representations, 2020. URL: https://openreview.net/forum?id=rkgNKkHtvB.

[23] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: International Conference on Machine Learning, PMLR, 2021, pp. 8748–8763.