FakeDTML at CheckThat! 2023: Identifying Check-Worthiness of Tweets and Debate Snippets

Notebook for the CheckThat! Lab at CLEF 2023

Abdullah Al Mamun Sardar[1], Md. Ziaul Karim[2], Krishno Dey[2] and Md. Arid Hasan[2]

[1] *Jahangirnagar University, Dhaka, Bangladesh*
[2] *Daffodil International University, Dhaka, Bangladesh*

Abstract

There is a wealth of knowledge available online. Some are trustworthy, while others are deceptive and phony. The need to identify such false information arises from the danger it poses to society at a mass. Nowadays, there is a significant need for information that requires fact-checking. As a result, we need a layer preceding fact-checking, where it can be determined whether a claim is check-worthy. This will streamline the automated fact-checking process by filtering out a lot of unnecessary data that is nonetheless necessary. We carried out such a study as part of CLEF 2023 CheckThat! Lab (CTL) task 1B, where we were provided with a dataset of tweets and debate snippets and were asked to conduct an experiment to verify whether a particular news tweet/debate snippet is check worthy. The dataset contains 3 languages (English, Arabic, Spanish). We used several machine learning and deep learning algorithms in our experiments. Among them, XLM-RoBERTa which outperformed other algorithms for English and Arabic but for Spanish we found that Logistic Regression can outperform other models.

Keywords

Check-worthiness, Fact-Checking, Check-worthy claim detection, XLM-RoBERTa, PassiveAggressive Classifier

1. Introduction

In the digital age, where information is easily accessible and rapidly disseminated, the proliferation of misinformation poses a significant challenge to society. Now we read more news online than any time before. As a consequence of this phenomenon, misleading claims, false narratives, and fabricated facts can easily spread through online platforms, leading to widespread confusion and a distortion of public understanding. To combat this issue, NLP researchers developed different strategies such as satire detection [20], automatic fact checking [18, 19], clickbait detection [27], harmful and toxic comment detection [12] [28] and check-worthy claim detection [3] [5]. Among these variant strategies, fact-checking has emerged as a crucial mechanism for verifying the validity of claims and promoting the dissemination of reliable information. Recently a lot of investigation has been done on automatic fact-checking [18] [19] [21], where researchers tried to come up with machine learning and deep learning based language modeling to identify whether a claim is fake or real. However, the amount of information which needs to be processed is huge. To filter out unnecessary claims which are not check-worthy, researchers set a prior step in the fact-checking process, where a claim should be checked first

[1]CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece
EMAIL: 3182abdullahalmamun1@juniv.edu (A. A. M. Sardar); ziaul35-1883@diu.edu.bd (Md. Z. Karim); krishno.cse@diu.edu.bd (K. Dey); arid.cse0325.c@diu.edu.bd (Md. A. Hasan)
ORCID: 0000-0002-0192-6986 (A. A. M. Sardar); 0000-0001-9989-0493 (Md. Z. Karim); 0000-0002-3194-5483 (K. Dey); 0000-0001-7916-614X (Md. A. Hasan)

CEUR Workshop Proceedings (CEUR-WS.org)

to identify whether it is check-worthy or not. *Wright, D., & Augenstein, I* [3] have visualized two claims on three different domains each, one which is check-worthy and the other which is not. They have shown that the second claim (for each domain) is not worthy of verification particularly when it comes to fact-checking. In this study we have investigated the check-worthiness of claims under the CTL 2023 subtask 1B [31]. We were given a dataset of 3 languages and then we performed a binary classification task to find whether a claim is check-worthy or not.

The rest of the paper is structured as follows: In section 2, we review the related work covering fact-checking and check-worthiness studies. Our adopted approach to conduct our experiments is presented in detail in Section 3. And then in section 4, we will compare the results of different algorithms along with the analysis of further experiments which we have conducted after the task submission. The rest two sections are conclusion and reference.

2. Related Work

Assessing the check worthiness of tweets, or the need for fact-checking, is crucial in filtering out misinformation. Recently, the field of Natural Language Processing (NLP) and its researchers have witnessed a surge in the popularity of automated fact-checking systems [6]. These systems have been developed using pre-trained language models specific to a particular language [2], incorporating datasets from local fact-checking organizations. An alternative approach suggested a lookup method that makes decisions based on the labeling of a given sample [3]. This approach further integrates transfer learning from Wikipedia's citation needed detection by employing a unified approach along with BERT [29] and PUC (Positive Unlabelled Conversion). In certain cases, to assess the credibility of information, it becomes necessary to comprehend the context at both word and sentence levels [4]. To address this, a layered approach combining RNN Encoders (LSTM/biLSTM units), a custom check-worthiness classifier, and handcrafted claim rank features was utilized. A different study using a combination of transformer-based and traditional models was explored to enhance computational efficiency and overall performance [5]. In the context of fact-checking on social media and low-resource languages, CheckthaT5, a sequence-to-sequence model based on mT5, was introduced as a solution [6]. The current state-of-the-art fact-checking models have limitations in terms of low-resource languages such as Spanish [1]. In the table below we have given a very brief overview of some previous works on check-worthiness detection of claims.

Table 1
Summary of some published works on check-worthiness detection

Ref	Year	Contribution	Dataset	Models
[8]	2021	Improved performance through contextual embedding augmentation on training dataset.	CT-CWT-21	BERT, RoBERTa-based
[9]	2022	Utilizing a specialized ensemble architecture, it combines the strengths of ten diverse transformer-based models. These models have been pre-trained on Twitter data, enabling them to generate raw predictions with precision and accuracy.	CT-CWT-22	Twitter-domain adapted version of RoBERTa, TweetEva , BERTweet
[10]	2022	Fine Tuning Various Transformer Models. Increasing Training Data via Machine Translation. Language Specific BERT with Manifold Mixup.	CT-CWT-22	AraBERT v0.2-Twitter, Bert-base-bg-cased, RobBERT

[11]	2022	Used XLNet embedding techniques in the proposed language model for its autoregressive and autoencoding properties and SVM classifier for tweet classification.	CT-CWT-22	XLNet, SVM
[12]	2022	The approach outperforms the official baseline by 8%. To improve the model performance and counteract class imbalance they set up class weights that correspond to a manual rescaling weight assigned to each class	CT-CWT-22	GCN, ELECTRA
[13]	2022	Applies augmentation techniques like back translation to increase the number of data. Uses large pretrained models and finetunes a few of them to get satisfactory results.	CT-CWT-22	DistilBERT, BERT and RoBERTa
[14]	2022	Demonstrated the performance of gated recurrent units for each of the subtasks AraBERT and BERT base Arabic were trained and fine-tuned. Despite the small sized annotated data, the model achieved satisfactory results.	CT-CWT-22	AraBERT, ARBERT, MARBERT, Arabic base BERT
[15]	2021	Presents machine learning classifiers for news claim and topic classification, achieving F1 scores of 38.92% and 78.96% respectively, and discusses the dataset augmentation findings regarding the ineffectiveness of alternative word insertion for fake news classification.	CT-CWT-21	LR, MLP, SVM, RF
[16]	2019	Proposes a multi-task deep-learning approach for estimating the check-worthiness of claims in political debates, demonstrating the benefits of learning from multiple fact-checking sources and achieving state-of-the-art results.	CW-USPD-2016	Neural Multi-task Learning Model (novel)
[17]	2021	The research paper proposes an approach for check-worthiness estimation using RoBERTa, fine-tuning a pre-trained language representation model on the classification task with annotated data.	CT-CWT-21	RoBERTa

3. Experiment Setup

3.1. Data

In this study, we used the dataset released by CLEF CheckThat! organizers. The dataset consists of a total of 3 languages (English, Spanish & Arabic). The dataset contains a unique id, a text snippet from a tweet or a debate/speech transcription which will be classified and a class label column where two

class labels (Yes & No) tells whether any claim is check-worthy or not. Distribution of the dataset for 3 different languages is presented in Table 2:

Table 2
CLEF dataset distribution for English, Spanish and Arabic

Language	Label	Train + Dev	Test
English	YES	5,651	108
	NO	17,882	210
Spanish	YES	3,211	509
	NO	11,737	4,491
Arabic	YES	2,654	377
	NO	5,772	123

3.2. Preprocessing

For data cleaning and preprocessing we have used traditional NLP techniques. First, we perform URLs and unnecessary character removal steps by following the approach discussed in [32]. Then we performed the removal of punctuations and null valued rows. Finally, along with the removal of stopwords, we removed hashtag signs and usernames.

3.3. Model Description

3.3.1. Traditional Machine Learning Algorithms

In our experiment, we used five traditional ML algorithms: *(i) MultinomialNB (MNB) [22] (ii) Support Vector Machine (SVM) [23] (iii) LogisticRegression (LR) [24] (iv) RandomForest Classifier (RF) [25]*. As this task is a kind of online learning, we used another algorithm named *(v)PassiveAggresive Classifier [26]*, which is popular for online learning tasks. We have trained our dataset with uni, bi and tri grams. For word embedding, we used both TF-IDF and CountVectorizer techniques. We used linear kernel for Support Vector Machine. To tune hyperparameter, we used grid search for all traditional ML models.

3.3.2. Transformer Based Algorithm

In 2018, Google released BERT [29], a pre-trained language model. Since then BERT based models are widely used for language based tasks. In our experiment we used a variant of the BERT based language model (RoBERTa) which was introduced by Facebook back in 2019 [30]. In particular, we used XLM-RoBERTa [5] in our experiment. We used a learning rate of 2e-5 to fine tune the hyperparameter. A full list of hyperparameters that we used in our experiment and their corresponding values are given in Table 3.

Table 3
Hyperparameter description for XLM-RoBERTa

Parameter Name	Corresponding Values
Maximum sequence length	128
Batch size	16
Learning rate	2e-5

Epochs	5

3.4. Methodology

In our experiment, we used both traditional machine learning algorithms and a transformer based deep learning algorithm**.** We merged the train and dev-test set to train the model. We used different n-grams for traditional machine learning algorithms. We have experimented with both TF-IDF and CountVectorizer techniques. Below in the diagram, we provided our proposed approach for the experiment.

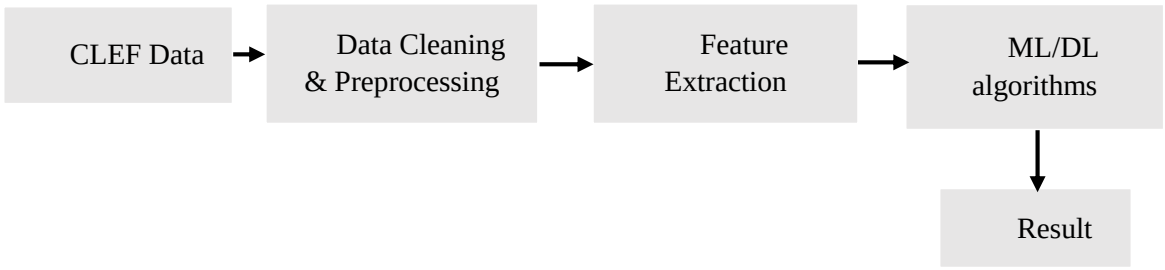

Figure 1: Our proposed approach

4. Result Analysis

4.1. Performance of Transformer and ML models

For traditional ML algorithms, we have experimented with different n-grams and vectorization techniques which described in in section 3.3.1 and we took the one which provided the best performance. Among 5 traditional ML algorithms, MultinomialNB outperformed the other four for English with bi-gram and CountVectorizer by achieving an F1 score of 80.36%, Logistic Regression outperformed the other four for Spanish with uni-gram and CountVectorizer, and for Arabic, PassiveAggressive Classifier provides the best result with uni-gram and TF-IDF vectorizer by achieving an F1 score of 57.87%. For transformer based algorithms, we used XLM-RoBERTa. We have provided a list of hyperparameters in Table 3. We used those same corresponding values to train the model for each language. Our experiment shows that the transformer based model can outperform all the traditional ML algorithms for English and Arabic but it gave poor results for Spanish. We have already seen in [1] and [5] that transformer based models did not give better results for low resource languages (e.g. Spanish). Our experiments show that Logistic Regression achieves 89.76% f1 score for Spanish while XLM-RoBERTa achieved only 51.26%. A performance comparison table for Transformer and traditional ML based models are given in Table 4.

Table 4

Performance comparison for Transformer and ML models

Language	Model	F1 Score
English	MultinomialNB	80.36
	SVC	72.81
	LogisticRegression	79.11
	PassiveAggressive Classifier	75.93
	RandomForestClassifier	67.92
	XLM-RoBERTa	**83.30**
Spanish	MultinomialNB	88.87
	SVC	89.11
	LogisticRegression	**89.76**

	PassiveAggressive Classifier	88.84
	RandomForestClassifier	87.17
	XLM-RoBERTa	51.26
	MultinomialNB	53.0
	SVC	20.05
Arabic	LogisticRegression	45.88
	PassiveAggressive Classifier	57.87
	RandomForestClassifier	20.77
	XLM-RoBERTa	**72.06**

4.2. Leaderboard Result

For the leaderboard submission we used XLM-RoBERTa for English language, for Spanish and Arabic, we used MultinomialNB. Our task on English language ranked 8th but due to late submission our task on Spanish and Arabic language did not get position number but it came right after 6th (for both Arabic and Spanish). In the following subsection we described in detail how with further experiments we have gained better results. The performance of our models for each language on the leaderboard is presented in Table 5.

Table 5
Leaderboard Results

Language	Model	F1 Score
English	XLM-RoBERTa	0.833
Spanish	MultinomialNB	0.440
Arabic	MultinomialNB	0.530

4.3. Further Experiment Analysis

Du, Mingzhe, et al. [6] performed an error analysis for their model submitted for CTL which did not perform well for the English language. They presumed that their model most probably became language agnostic as they tried several languages with the same model and that might be the reason why they got poor performance on English language. For our task at CTL, after the publication of the results on leaderboard, we saw our model performed poorly on Spanish and Arabic, gave an F1-score of 44% and 53% for Spanish and Arabic respectively. Then we performed some further investigation into why the performance was not so good when other teams came up with better results. In our later experiment, we have applied whitespace tokenization for all languages, Snowball stemmer for Spanish, Arabic Stemmer for Arabic and Porter Stemmer for English. With this set up, we then experimented with some other ML algorithms (LR, RF, SVM, PassiveAggressive Classifier). Our further experiment shows that LR outperformed all the other models (including XLM-RoBERTa) for Spanish language. For Arabic language, using PassiveAggressive Classifier we were able to increase the performance by 4.87% than the leaderboard result.

5. Conclusion

In this work, we present our participation in CLEF CTL 2023 subtask 1B to detect check-worthy claims. Our experiment shows that both uni-gram and bi-gram models can perform better with TF-IDF and CountVectorizer techniques. Although our initial models did not perform well on the leaderboard for Arabic and Spanish, we conducted some further investigations. We experimented with several

machine learning and deep learning algorithms and found that the transformer based deep learning model (XLM-RoBERTa) can outperform traditional machine learning models for English and Arabic but for Spanish language, one traditional ML model (Logistic Regression) did better than other models.

6. References

[1] Nakov, P., Barrón-Cedeño, A., Da San Martino, G., Alam, F., Kutlu, M., Zaghouani, W., ... & Nikolov, A. (2022). Overview of the CLEF-2022 CheckThat! lab task 1 on identifying relevant claims in tweets.

[2] Sheikhi, G., Touileb, S., & Khan, S. A. (2023, March). Automated Claim Detection for Fact-checking: A Case Study using Norwegian Pre-trained Language Models. In *The 24rd Nordic Conference on Computational Linguistics*.

[3] Wright, D., & Augenstein, I. (2020). Claim check-worthiness detection as positive unlabelled learning. *arXiv preprint arXiv:2003.02736*.

[4] Ren, O. T. (2019). *Detecting check-worthy claims* (Doctoral dissertation, Massachusetts Institute of Technology).

[5] Tarannum, P., Alam, F., Hasan, M. A., & Noori, S. R. H. (2022). Z-Index at CheckThat! Lab 2022: Check-Worthiness Identification on Tweet Text. *arXiv preprint arXiv:2207.07308*.

[6] Du, M., Gollapalli, S. D., & Ng, S. K. (2022). NUS-IDS at CheckThat! 2022: identifying check-worthiness of tweets using CheckthaT5. *Working Notes of CLEF*.

[7] Arslan, F., Hassan, N., Li, C., & Tremayne, M. (2020, May). A benchmark dataset of check-worthy factual claims. In *Proceedings of the International AAAI Conference on Web and Social Media* (Vol. 14, pp. 821-829).

[8] Williams, E., Rodrigues, P., & Tran, S. (2021). Accenture at CheckThat! 2021: interesting claim identification and ranking with contextually sensitive lexical training data augmentation. *arXiv preprint arXiv:2107.05684*.

[9] Buliga, N., & Raschip, M. (2022). Zorros at CheckThat! 2022: Ensemble Model for Identifying Relevant Claims in Tweets.

[10] Eyuboglu, A. B., Arslan, M. B., Sonmezer, E., & Kutlu, M. (2022). TOBB ETU at CheckThat! 2022: detecting attention-worthy and harmful tweets and check-worthy claims. *Working Notes of CLEF*.

[11] Kavatagi, S., Rachh, R., & Mulimani, M. (2022). VTU_BGM at CheckThat! 2022: An Autoregressive Encoding Model for Detecting Check-worthy Claims.

[12] Lomonaco, F., Donabauer, G., & Siino, M. (2022). Courage at checkthat! 2022: Harmful tweet detection using graph neural networks and electra. *Working Notes of CLEF*, *1*.

[13] Savchev, A. (2022). AI Rational at CheckThat! 2022: using transformer models for tweet classification. *Working Notes of CLEF*.

[14] Taboubi, B., Nessir, M. A. B., & Haddad, H. (2022). iCompass at CheckThat! 2022: ARBERT and AraBERT for Arabic Checkworthy Tweet Identification.

[15] Ashraf, N., Butt, S., Sidorov, G., & Gelbukh, A. F. (2021, September). CIC at CheckThat! 2021: Fake News detection Using Machine Learning And Data Augmentation. In *CLEF (Working Notes)* (pp. 446-454).

[16] Vasileva, S., Atanasova, P., Màrquez, L., Barrón-Cedeño, A., & Nakov, P. (2019). It takes nine to smell a rat: Neural multi-task learning for check-worthiness prediction. *arXiv preprint arXiv:1908.07912*.

[17] Pritzkau, A. (2021). NLytics at CheckThat! 2021: Check-Worthiness Estimation as a Regression Problem on Transformers. In *CLEF (Working Notes)* (pp. 592-602).

[18] Sardar, A. A. M., Salma, S. A., Islam, M. S., Hasan, M. A., & Bhuiyan, T. (2021). Team Sigmoid at CheckThat! 2021 Task 3a: Multiclass fake news detection with Machine Learning. In *CLEF (Working Notes)* (pp. 612-618).

[19] Zhang, J., Dong, B., & Philip, S. Y. (2020, April). Fakedetector: Effective fake news detection with deep diffusive neural network. In *2020 IEEE 36th international conference on data engineering (ICDE)* (pp. 1826-1829). IEEE.

[20] Razali, M. S., Halin, A. A., Chow, Y. W., Norowi, N. M., & Doraisamy, S. (2022). Context-Driven Satire Detection With Deep Learning. *IEEE Access*, *10*, 78780-78787.

[21] Kumar, S., Asthana, R., Upadhyay, S., Upreti, N., & Akbar, M. (2020). Fake news detection using deep learning models: A novel approach. *Transactions on Emerging Telecommunications Technologies*, *31*(2), e3767.

[22] Xu, S., Li, Y., & Wang, Z. (2017). Bayesian multinomial Naïve Bayes classifier to text classification. In *Advanced Multimedia and Ubiquitous Engineering: MUE/FutureTech 2017 11* (pp. 347-352). Springer Singapore.

[23] Rossi, F., & Villa, N. (2006). Support vector machine for functional data classification. *Neurocomputing*, *69*(7-9), 730-742.

[24] Larsen, K., Petersen, J. H., Budtz-Jørgensen, E., & Endahl, L. (2000). Interpreting parameters in the logistic regression model with random effects. *Biometrics*, *56*(3), 909-914.

[25] Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. *R news*, *2*(3), 18-22.

[26] Gupta, S., & Meel, P. (2021). Fake news detection using passive-aggressive classifier. In *Inventive Communication and Computational Technologies: Proceedings of ICICCT 2020* (pp. 155-164). Springer Singapore.

[27] Agrawal, A. (2016, October). Clickbait detection using deep learning. In *2016 2nd international conference on next generation computing technologies (NGCT)* (pp. 268-272). IEEE.

[28] Andročec, D. (2020). Machine learning methods for toxic comment classification: a systematic review. *Acta Universitatis Sapientiae, Informatica*, *12*(2), 205-216.

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

[30] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

[31] Alam, Barrón-Cedeño, Cheema, Hakimov, Hasanain, Li, Míguez, Mubarak, Shahi, Zaghouani, and Nakovet. Overview of the CLEF-2023 CheckThat! Lab Task 1 on Check-Worthiness in Multimodal and Multigenre Content. In Proceedings of the Working Notes of CLEF 2023 - Conference and Labs of the Evaluation Forum. Thessaloniki, Greece.

[32] Alam, F., Sajjad, H., Imran, M., & Ofli, F. (2021, May). CrisisBench: Benchmarking crisis-related social media datasets for humanitarian information processing. In *Proceedings of the International AAAI Conference on Web and Social Media* (Vol. 15, pp. 923-932).