TOBB ETU at CheckThat! 2021: Data Engineering for Detecting Check-Worthy Claims

Muhammed Said Zengin[1], Yavuz Selim Kartal[1] and Mucahid Kutlu[1]

[1]TOBB University of Economics and Technology, Ankara, Turkey

Abstract

In this paper, we present our participation in CLEF 2021 CheckThat! Lab's Task 1 on check-worthiness estimation in tweets. We explore how to fine-tune transformer models effectively by changing the train set. The methods we explore include language-specific training, weak supervision, data augmentation by machine translation, undersampling, and cross-lingual training. As our primary model submitted for official results, we fine-tune language-specific BERT-based models using cleaned tweets for each language. Our models ranked 1^{st} in Spanish and Turkish datasets. However, our rank in Arabic, Bulgarian, and English datasets is 6^{th}, 4^{th}, and 10^{th}, respectively.

Keywords

Check Worthiness, Fact Checking, Data Engineering

1. Introduction

Social media platforms provide a suitable environment to easily communicate with other people. Therefore, many people enjoy the freedom of speech on these platforms by sharing any message they want. However, the very same platforms can be also used to spread misinformation, which has a huge negative impact on society such as massive stock price changes[1], vaccine hesitancy[2], and using dangerous chemicals for medical treatment[3].

Many journalists combat against spread of misinformation by investigating veracity of claims and sharing their findings with the public via fact-checking websites such as Snopes[4] and PolitiFact[5]. While these fact-checking websites are vital in the combat against misinformation, the problem continues to exist because false news spread faster than true news [1], and fact-checking is an extremely time-consuming process [2]. Therefore, we urgently need systems assisting fact-checkers in the combat against misinformation.

Building systems that automatically detect veracity of claims is the ultimate goal for fact-checking studies. However, building a "perfect" fact checking system will not prevent spread of misinformation and its negative outcomes, if people continue to share everything they see on Internet. Therefore, we believe that we also need systems that warn social media users when

CLEF 2021 – Conference and Labs of the Evaluation Forum, September 21–24, 2021, Bucharest, Romania

[1]www.forbes.com/sites/kenrapoza/2017/02/26/can-fake-news-impact-the-stock-market

[2]www.washingtonpost.com/news/wonk/wp/2014/10/13/the-inevitable-rise-of-ebola-conspiracy-theories

[3]www.forbes.com/sites/nicholasreimann/2020/08/24/some-americans-are-tragically-still-drinking-bleach-as-a-coronavirus-cure/?sh=2af98fe86748

[4]https://www.snopes.com

[5]https://www.politifact.com

they share a check-worthy claim to allow them to re-think sharing it. In addition, these systems can help human fact-checkers to detect claims to be fact-checked, allowing them to spend their precious efforts for the most important claims. Therefore, many researchers focused on how to detect check-worthy claims [3, 4]. CLEF has been organizing CheckThat! Labs which cover detecting check-worthy tasks since 2018 [5, 6, 7].

In this work, we explain our participation in Task 1A on Check-Worthiness Estimation in Tweets [8] of CLEF 2021 CheckThat! Lab [9]. The task covers tweet datasets in five different languages including Arabic (AR), Bulgarian (BG), English (EN), Spanish (SP), and Turkish (TR). We investigate several methods to effectively fine-tune transformer models for each language. In particular, we explore five different methods: 1) fine-tuning a language-specific transformer model for each language, 2) balancing label distribution by undersampling the majority label, 3) data augmentation using machine translation, 4) weak supervision, and 5) cross-lingual training.

In our experiments on the development set, we observe that fine-tuning a language-specific model with cleaned tweets yield the best results for all languages. Therefore, we pick it as our primary model and submit its output for the official ranking. We are ranked first in Turkish and Spanish datasets. However, our rank for Arabic, Bulgarian, and English is 6^{th}, 4^{th}, and 10^{th}, respectively. Our experiments on the test data also show that our undersampling method for Arabic yields higher results than our primary model. In addition, using original tweets instead of cleaned ones yields much higher performance than our submitted results for Bulgarian and English datasets.

The organization of the paper is as follows. We discuss related work in Section 2. Section 3 explains our methods. We present experimental results in Section 4 and conclude in Section 5.

2. Related Work

One of the first check-worthy claim detection models is ClaimBuster [2]. It uses many features including part-of-speech (POS) tags, named entities, sentiment, and TF-IDF representations of claims. Patwari et al. [10] use topics from the presidential debates between 1976 and 2016, POS tuples, entity history, and bag-of-words as features. Gencheva et al. [11] propose a neural network model with a long list of sentence level and contextual features including sentiment, named entities, word embeddings, topics, contradictions, and others. Jaradat et al. [12] extend the model of Gencheva et al. for Arabic by using similar features. Vasileva et al. [13] propose a multi-task learning model to detect whether a claim is fact-checked by reputable fact-checking organizations.

CLEF has been organizing CheckThat! Labs (CTL) since 2018. Seven teams participated in the first organization, CTL'18, which covers English and Arabic claims [5]. They investigated various models like recurrent neural network (RNN) [14], multilayer perceptron [15], random forest (RF) [16], k-nearest neighbor (kNN) [17] and gradient boosting [18] with different sets of features such as bag-of-words [15], character n-gram [17], POS tags [14, 18, 15], verbal forms [15], named entities [18, 15], syntactic dependencies [15, 14], and word embeddings [14, 18, 15]. Prise de Fer team [15] achieved the best MAP scores using bag-of-words, POS tags, named entities, verbal forms, negations, sentiment, clauses, syntactic dependency, and word embeddings with SVM-Multilayer perceptron learning on the English dataset. On the Arabic dataset, BigIR

team [18] outperformed the others using POS tags, named entities, sentiment, topics, and word embeddings, as features.

In CTL'19, 11 team participated in the check-worthiness task which has been organized for only English. Participants of the task used many learning models such as LSTM, SVM, naive bayes, and logistic regression (LR) with many features including readability of sentences and their context [6]. Copenhagen team [19] achieved the best overall MAP score using syntactic dependency and word embeddings with weakly supervised LSTM model.

In CTL'20, two tasks, Task 1 and Task 5, have been organized for check-worthiness [7]. While Task 1 covers tweets in Arabic and English, Task 5 covers English debates. Participants of Task 1 used BERT [20, 21, 22, 23, 24], RoBERTa [21, 25] BiLSTM [26, 27], CNN [24], RF [28], LR [20], and SVM [23] models with various features such as FastText [20, 28], Glove [27], PCA [23], TF-IDF [28], POS tags [20, 23], and named entities [23]. Accenture team [21] achieved the best MAP score in both datasets using BERT model for Arabic and RoBERTa model for English. Participants of Task 5 used BERT [20], LR and LSTM models with TF-IDF, word embedding and POS tag features [29]. Team NLPIR01 is ranked first using LSTM model with word embeddings. They also explore different sampling methods but report that they do not improve the performance.

3. Proposed Methods

In our work, we investigated how to train transformer models effectively. The methods we use include language-specific training (Section 3.1), balancing label distribution (Section 3.2), data augmentation using machine translation (Section 3.2), weak supervision (Section 3.4), and cross-lingual training (Section 3.5). Now we explain our methods in detail.

3.1. Language-Specific Training

Prior work showed that fine-tuning transformer models pre-trained for a single language yields great performance in various NLP tasks, outperforming several state-of-the-art models [30]. Therefore, in this method, we use a language-specific transformer model for each language. In particular, we use BERTurk[6], AraBERT [31], BETO[7], and BERT base model [30] for Turkish, Arabic, Spanish, and English, respectively. For Bulgarian, we use a pre-trained RoBERTa model[8]. We fine-tune each model with the respective training data. In this method, we explore two different approaches: 1) language-specific models fine-tuned using the original tweets ($LSM_{original\_tweets}$), and 2) language-specific models fine-tuned using cleaned tweets ($LSM_{cleaned\_tweets}$) in which we remove all mentions and URLs from tweets.

3.2. Balancing Label Distribution

In a random sample of tweets, it is less likely to encounter check-worthy claims, yielding imbalanced data distribution. We also observe this situation in the datasets shared for Task 1A

[6]https://huggingface.co/dbmdz/bert-base-turkish-cased
[7]https://huggingface.co/dccuchile/bert-base-spanish-wwm-cased
[8]https://huggingface.co/iarfmoose/roberta-base-bulgarian

(See Table 2). The ratio of check-worthy tweets in the train set is 22%, 13%, 35%, 8%, and 38% for Arabic, Bulgarian, English, Spanish, and Turkish, respectively.

Imbalanced label distribution can negatively affect the learning process for models. In order to make the dataset fully balanced, we can oversample check-worthy claims or undersample not-check-worthy claims. Yasser et al. [18] report that oversampling do not improve their model's performance on check-worthy claim detection. Therefore, we investigate undersampling approach by setting the check-worthy claim ratio same for all languages, to make a fair comparison across languages. However, having a fully balanced dataset (i.e., the same amount of tweets for each label) would cause removing many tweets. Therefore, we make the check-worthy claim ratio 30% of the train set by undersampling. In particular, we undersample not-check-worthy claims in Arabic, Bulgarian, and Spanish datasets. Check-worthy claim ratio for Turkish and English datasets is higher than 30%. Therefore, we undersample check-worthy claims in these datasets. While undersampling the positive class might not be effective, we do it to make a fair comparison across languages. Subsequently, we remove mentions and URLs and fine-tune language-specific models for each language, as mentioned before.

3.3. Data Augmentation Using Machine Translation

The amount of labeled data has a significant impact on the trained models. However, labeling is a costly and time consuming process. Therefore, in order to increase labeled data size automatically, we exploit machine translation methods. In particular, for each language, we translate tweets labeled as check-worthy in the other languages using Google Translate. After translation, we remove mentions and URLs and fine-tune language-specific models for each language. This method also reduces the imbalanced label distribution problem. For instance, the ratio of check-worthy claims for Spanish dataset increases to 50.8% by this method.

3.4. Weak Supervision

Another way to increase the labeled data size is weak supervision [32]. Therefore, we use the following weak-supervision method. We first rank words based on their frequency in each dataset, and manually select 10 words, which are related to the topic of the respective datasets, among the most frequent 100 words. **Table 1** shows these keywords. Then we crawl 500 tweets tracking each of these words separately using Twint[9] tool, yielding 5000 tweets in total for each language. Subsequently, we label these collected tweets using XLM-R [33] model which is fine-tuned using cleaned tweets of the respective train data. Finally, we remove URLs and hashtags from the tweets and fine-tune our language-specific transformer models for each language using the training data and tweets labeled by our XLM-R model.

3.5. Cross Lingual Training

Multilingual transformer models enable training models with labeled data in a particular language and use the trained model for another language. Therefore, they have great potential for further advancements in NLP, especially for low-resource languages. In this method, we

[9]https://pypi.org/project/twint

Table 1
Selected Words to Collect Additional Tweets

Arabic	Bulgarian	English	Spanish	Turkish
كورونا	България	covid19	españa	yüzde
النسويات	случаи	virus	gobierno	milyar
بفيروس	заразени	people	millones	türkiye
النسوية	кризата	cases	sánchez	dolar
الشعب	пандемията	health	personas	istanbul
الصحة	ваксина	testing	euros	belediye
إصابة	разпространението	confirmed	gobierno	ticaret
عاجل	мерките	coronavirus	madrid	seçim
وزارة	европа	hospital	política	ülkeler
التطبيع	денонощие	patients	contra	enflasyon

explore whether cross-lingual training is effective to detect check-worthy claims. In particular, for each language pair, we combine their train set and fine-tune mBERT [30] model using the combined dataset. Subsequently, the fined-tuned model can be used for all five languages.

4. Experiments

In this section, we first explain our experimental setup (Section 4.1). Then we present our results on the development and test sets. (Section 4.2)

4.1. Experimental Setup

Dataset. The datasets shared by the shared-task organizers are divided into train, development, and test sets for each language. **Table 2** shows data and label distribution for each dataset. In our experiments with the development set, we use the train sets of languages (and additional data we get with our methods explained in Section 3) to fine-tune models. In our experiments with the test set, we also add the development set of each language to their train data to fine-tune models.

Table 2
Data and Label Distribution for Each Language.

Language	Topic	Train		Development		Test	
		CW	Not CW	CW	Not CW	CW	Not CW
Arabic	miscellaneous	763	2676	265	396	242	358
Bulgarian	covid-19	392	2608	62	288	76	281
English	covid-19	290	532	60	80	19	331
Spanish	politics	200	2295	109	1138	120	1128
Turkish	miscellaneous	729	1170	146	242	183	830

Table 3
Model Names

Language	Model Name	Batch Size	Epoch
Arabic	AraBERT	6	1
Bulgarian	RoBERTa Base for Bulgarian	6	3
English	BERT Base	3	3
Multilingual	mBERT	6	1
Spanish	BETO	6	1
Turkish	BERTurk	6	3

Implementation. We use ktrain[10] library to fine-tune our models. In order to set parameters of each model, we conducted (not reported) experiments on the development set with various configurations and picked the best performing one for each model. In particular, the parameters of each model we use are shown in **Table 3**. We set learning rate to 5e-5 for all models.

4.2. Experimental Results

4.2.1. Results on the Development Set

We first evaluate the performance of each model we propose in the development set in order to pick the model to be submitted as our primary model. **Table 4** shows average precision (AP) scores of our cross-lingual approach for each language pair.

Table 4
AP Score of Cross-lingual Training on the Development Test. Best score for each language is written in bold.

Languages for Training	Arabic	Bulgarian	English	Spanish	Turkish
AR + SP	0.349	0.274	0.457	0.118	0.451
BG + AR	0.512	0.194	0.481	0.077	0.372
BG + SP	0.386	0.362	0.477	0.187	0.522
EN + BG	0.252	0.152	0.536	0.140	0.511
EN + AR	**0.713**	0.241	0.564	0.156	0.505
EN + SP	0.183	0.195	0.410	0.269	0.417
TR + BG	0.433	**0.505**	0.607	0.121	0.585
TR + EN	0.532	0.264	**0.610**	0.135	**0.601**
TR + AR	0.606	0.214	0.507	0.090	0.536
TR + SP	0.300	0.189	0.495	**0.298**	0.556

For each language, we achieve the best result when one of the languages in the train set is same with the language of the development set. Interestingly, even though Turkish is linguistically the most distant language to others, using Turkish as one of the languages in the train set yields the best results for Bulgarian, English, Spain, and Turkish. This might be because the ratio of check-worthy tweets in the train set of Turkish dataset is higher than all other languages. In addition, using English as one of the languages in the train set yields the best performance

[10]https://pypi.org/project/ktrain

for three languages (i.e., Turkish, Arabic, and English). This might be because of mBERT's capability to represent English texts better than other languages.

Table 5 shows results of our other methods on the development test. $LSM_{cleaned\_tweets}$ yields the best performance for all languages. Therefore, we use it as our primary model for our participation in the shared-task. In addition, we observe that data augmentation with translation yields the lowest score in most of the cases, suggesting that check-worthiness of claims vary across nations.

Table 5
AP Score in the Development Test Using Language Specific Transformer Models. Best score for each language is written in bold.

Model Name	Arabic	Bulgarian	English	Spanish	Turkish
$LSM_{original\_tweets}$	0.714	0.466	0.601	0.489	0.694
$LSM_{cleaned\_tweets}$	**0.755**	**0.528**	**0.712**	**0.544**	**0.701**
Undersampling	**0.755**	0.489	0.526	0.444	0.573
Weak supervision	0.702	0.413	0.673	0.422	0.684
Data Augmentation w/ Translation	0.618	0.351	0.536	0.149	0.571

4.2.2. Results on Test Test

We submit our results for $LSM_{cleaned\_tweets}$ model. **Table 6** presents AP score and rank of our submitted results. We are ranked first for Turkish and Spanish languages based on AP score. However, our models do not achieve a high ranking for the other languages.

Table 6
AP Score and Rank of Our Submissions.

Language	Result	Rank
Arabic	0.575	6
Bulgarian	0.149	4
English	0.081	10
Spanish	0.537	1
Turkish	0.581	1

In order to further investigate the performance of our models, we report performance of each model on the test set in **Table 7**. For the cross-lingual training approach, we use the best performing model in the development set for each language. In particular, we report results for mBERT model trained with TR+BG tweets for Bulgarian, TR+EN tweets for English and Turkish, TR+AR tweets for Arabic, and TR+SP tweets for Spanish.

Comparing our results for the development and test sets, we observe that performance of models change dramatically across datasets. In the development set, fine-tuning language specific models with cleaned data yields the best scores in all datasets. However, in the test set, it yields the best results for only Turkish and Spanish. For Arabic, undersampling yields 0.622 AP score, outperforming the participant ranked 2^{nd} in the official results of the shared-task. In addition, we observe that removing mentions and URLs has negative impact on the

Table 7
AP Score in Test Set. Best score for each language is written in bold.

Model Name	Arabic	Bulgarian	English	Spanish	Turkish
$LSM_{original\_tweets}$	0.600	**0.548**	**0.172**	0.505	0.553
$LSM_{cleaned\_tweets}$	0.575	0.149	0.081	**0.537**	**0.581**
Undersampling	**0.622**	0.241	0.158	0.522	0.580
Weak supervision	0.546	0.217	0.156	0.489	0.535
Data Augmentation w/ Translation	0.524	0.228	0.126	0.457	0.489
Cross-Lingual Training	0.543	0.532	0.151	0.149	0.443

language-specific models for Bulgarian and English. $LSM_{original\_tweets}$ yields 0.171 AP score in the English dataset, outperforming the participant ranked 3^{rd} in the official results of the shared-task.

5. Conclusion

In this paper, we present our participation in Task 1A of CLEF 2021 CheckThat! lab. We explore several methods to fine-tune transformer models effectively by changing the train set. In particular, we investigate five different methods including language-specific training, undersampling, data augmentation with machine translation, weak supervision, and cross-lingual training. Our experiments on the development set show that fine-tuning a language specific transformer model with cleaned tweets yields the highest performance. Therefore, we submit our results using this model. We are ranked first for Turkish and Spanish datasets in the official ranking. However, the models we submitted for other languages did not achieve the same performance. Our experiments also show that data augmentation with machine translation and weak supervision generally do not yield high performance.

In the future, we plan to investigate how our methods' performance can be increased by better parameter tuning and more sophisticated weak-supervision and data augmentation methods. In addition, we think that developing subject-specific check-worthy claim detection models might be an effective solution for this problem.

References

[1] S. Vosoughi, D. Roy, S. Aral, The spread of true and false news online, Science 359 (2018) 1146–1151.

[2] 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, PVLDB 10 (2017) 1945–1948.

[3] Y. S. Kartal, M. Kutlu, B. Guvenen, Too many claims to fact-check: Prioritizing political claims based on check-worthiness, in: Proceedings of the 5th International Workshop on Mining Actionable Insights from Social Networks (MAISoN), 2020.

[4] N. Hassan, F. Arslan, C. Li, M. Tremayne, Toward automated fact-checking: Detecting check-worthy factual claims by claimbuster, in: Proceedings of the 23rd ACM SIGKDD

International Conference on Knowledge Discovery and Data Mining, ACM, 2017, pp. 1803–1812.

[5] P. Nakov, A. Barrón-Cedeño, T. Elsayed, R. Suwaileh, L. Màrquez, W. Zaghouani, P. Atanasova, S. Kyuchukov, G. Da San Martino, Overview of the clef-2018 checkthat! lab on automatic identification and verification of political claims, in: International Conference of the Cross-Language Evaluation Forum for European Languages, 2018, pp. 372–387.

[6] P. Atanasova, P. Nakov, G. Karadzhov, M. Mohtarami, G. Da San Martino, Overview of the clef-2019 checkthat! lab on automatic identification and verification of claims. task 1: Check-worthiness, in: CEUR Workshop Proceedings, 2019.

[7] A. Barrón-Cedeño, T. Elsayed, P. Nakov, G. Da San Martino, M. Hasanain, R. Suwaileh, F. Haouari, N. Babulkov, B. Hamdan, A. Nikolov, S. Shaar, Z. Ali, Overview of CheckThat! 2020 — automatic identification and verification of claims in social media, in: Proceedings of the 11th International Conference of the CLEF Association: Experimental IR Meets Multilinguality, Multimodality, and Interaction, CLEF '2020, 2020, pp. 215–236.

[8] S. Shaar, M. Hasanain, B. Hamdan, Z. S. Ali, F. Haouari, A. Nikolov, M. Kutlu, Y. S. Kartal, F. Alam, D. S. M. Giovanni, A. Barrón-Cedeño, R. Míguez, T. Elsayed, P. Nakov, Overview of the CLEF-2021 CheckThat! Lab Task 1 on Check-Worthiness Estimation in Tweets and Political Debates, in: Working Notes of CLEF 2021—Conference and Labs of the Evaluation Forum, CLEF '2021, Bucharest, Romania (online), 2021.

[9] 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.

[10] A. Patwari, D. Goldwasser, S. Bagchi, Tathya: A multi-classifier system for detecting check-worthy statements in political debates, in: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, ACM, 2017, pp. 2259–2262.

[11] P. Gencheva, P. Nakov, L. Màrquez, A. Barrón-Cedeño, I. Koychev, A context-aware approach for detecting worth-checking claims in political debates, in: Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017, 2017, pp. 267–276.

[12] I. Jaradat, P. Gencheva, A. Barrón-Cedeño, L. Màrquez, P. Nakov, Claimrank: Detecting check-worthy claims in arabic and english, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations, 2018, pp. 26–30.

[13] S. Vasileva, P. Atanasova, L. Màrquez, A. Barrón-Cedeño, P. Nakov, It takes nine to smell a rat: Neural multi-task learning for check-worthiness prediction, in: Proceedings of the International Conference on Recent Advances in Natural Language Processing, 2019.

[14] C. Hansen, C. Hansen, J. Simonsen, C. Lioma, The Copenhagen team participation in the check-worthiness task of the competition of automatic identification and verification of claims in political debates of the CLEF-2018 fact checking lab, in: [34], 2018.

[15] C. Zuo, A. Karakas, R. Banerjee, A hybrid recognition system for check-worthy claims

using heuristics and supervised learning, in: [34], 2018.
[16] R. Agez, C. Bosc, C. Lespagnol, J. Mothe, N. Petitcol, IRIT at CheckThat! 2018, in: [34], 2018.
[17] B. Ghanem, M. Montes-y Gómez, F. Rangel, P. Rosso, UPV-INAOE-Autoritas - Check That: Preliminary approach for checking worthiness of claims, in: [34], 2018.
[18] K. Yasser, M. Kutlu, T. Elsayed, bigir at CLEF 2018: Detection and verification of check-worthy political claims, in: Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, 2018.
[19] C. Hansen, C. Hansen, J. Simonsen, C. Lioma, Neural weakly supervised fact check-worthiness detection with contrastive sampling-based ranking loss, CEUR Workshop Proceedings, CEUR-WS.org, 2019.
[20] Y. S. Kartal, M. Kutlu, TOBB ETU at CheckThat! 2020: Prioritizing English and Arabic claims based on check-worthiness, in: [35], 2020.
[21] E. Williams, P. Rodrigues, V. Novak, Accenture at CheckThat! 2020: If you say so: Post-hoc fact-checking of claims using transformer-based models, in: [35], 2020.
[22] M. Hasanain, T. Elsayed, bigIR at CheckThat! 2020: Multilingual BERT for ranking Arabic tweets by check-worthiness, in: [35], 2020.
[23] G. S. Cheema, S. Hakimov, R. Ewerth, Check_square at CheckThat! 2020: Claim detection in social media via fusion of transformer and syntactic features, in: [35], 2020.
[24] R. Alkhalifa, T. Yoong, E. Kochkina, A. Zubiaga, M. Liakata, QMUL-SDS at CheckThat! 2020: Determining COVID-19 tweet check-worthiness using an enhanced CT-BERT with numeric expressions, in: [35], 2020.
[25] A. Nikolov, G. Da San Martino, I. Koychev, P. Nakov, Team_Alex at CheckThat! 2020: Identifying check-worthy tweets with transformer models, in: [35], 2020.
[26] A. Hussein, A. Hussein, N. Ghneim, A. Joukhadar, DamascusTeam at CheckThat! 2020: Check worthiness on Twitter with hybrid CNN and RNN models, in: [35], 2020.
[27] J. Martinez-Rico, L. Araujo, J. Martinez-Romo, NLP&IR@UNED at CheckThat! 2020: A preliminary approach for check-worthiness and claim retrieval tasks using neural networks and graphs, in: [35], 2020.
[28] T. McDonald, Z. Dong, Y. Zhang, R. Hampson, J. Young, Q. Cao, J. Leidner, M. Stevenson, The University of Sheffield at CheckThat! 2020: Claim identification and verification on Twitter, in: [35], 2020.
[29] A. Barrón-Cedeño, T. Elsayed, P. Nakov, G. D. S. Martino, M. Hasanain, R. Suwaileh, F. Haouari, Checkthat! at clef 2020: Enabling the automatic identification and verification of claims in social media, Advances in Information Retrieval 12036 (2020) 499 – 507.
[30] 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), 2019, pp. 4171–4186.
[31] W. Antoun, F. Baly, H. Hajj, AraBERT: Transformer-based model for Arabic language understanding, in: Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection, European Language Resource Association, Marseille, France, 2020, pp. 9–15.
[32] A. Ratner, S. H. Bach, H. Ehrenberg, J. Fries, S. Wu, C. Ré, Snorkel: Rapid training data

creation with weak supervision, in: Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases, volume 11, NIH Public Access, 2017, p. 269.
[33] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, É. 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, 2020, pp. 8440–8451.
[34] L. Cappellato, N. Ferro, J.-Y. Nie, L. Soulier (Eds.), Working Notes of CLEF 2018–Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2018.
[35] L. Cappellato, C. Eickhoff, N. Ferro, A. Névéol (Eds.), CLEF 2020 Working Notes, CEUR Workshop Proceedings, CEUR-WS.org, 2020.