Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

Overview of CheckThat! 2020: Automatic Identification and Verification of Claims in Social Media

(Article begins on next page)

14 January 2025

Overview of CheckThat! 2020: Automatic Identification and Verification of Claims in Social Media

Alberto Barrón-Cedeño[1], Tamer Elsayed[2], Preslav Nakov[3],
Giovanni Da San Martino[3], Maram Hasanain[2], Reem Suwaileh[2],
Fatima Haouari[2], Nikolay Babulkov[4], Bayan Hamdan[5],
Alex Nikolov[4], Shaden Shaar[3], and Zien Sheikh Ali[2]

[1] DIT, Università di Bologna, Forlì, Italy
`a.barron@unibo.it`
[2] Computer Science and Engineering Department, Qatar University, Doha, Qatar
`{telsayed, maram.hasanain, rs081123, 200159617, zs1407404}@qu.edu.qa`
[3] Qatar Computing Research Institute, HBKU, Doha, Qatar
`{pnakov, gmartino, sshaar}@hbku.edu.qa`
[4] FMI, Sofia University "St Kliment Ohridski", Bulgaria
`{nbabulkov, alexnickolow}@gmail.com`
[5] Independent Researcher
`bayan.hamdan995@gmail.com`

Abstract. We present an overview of the third edition of the `CheckThat!` Lab at CLEF 2020. The lab featured five tasks in two different languages: English and Arabic. The first four tasks compose the full pipeline of claim verification in social media: Task 1 on check-worthiness estimation, Task 2 on retrieving previously fact-checked claims, Task 3 on evidence retrieval, and Task 4 on claim verification. The lab is completed with Task 5 on check-worthiness estimation in political debates and speeches. A total of 67 teams registered to participate in the lab (up from 47 at CLEF 2019), and 23 of them actually submitted runs (compared to 14 at CLEF 2019). Most teams used deep neural networks based on BERT, LSTMs, or CNNs, and achieved sizable improvements over the baselines on all tasks. Here we describe the tasks setup, the evaluation results, and a summary of the approaches used by the participants, and we discuss some lessons learned. Last but not least, we release to the research community all datasets from the lab as well as the evaluation scripts, which should enable further research in the important tasks of check-worthiness estimation and automatic claim verification.

1 Introduction

The `CheckThat!` lab[1] was run for the third time in the framework of CLEF 2020. The purpose of the 2020 edition was to foster the development of technology that would enable the (semi-)automatic verification of claims posted in social media, in particular *Twitter*.[2] We turn our attention to Twitter because information posted on that platform is not checked by an authoritative entity before publication and such information tends to disseminate very quickly.[3] Moreover, social media posts lack context due to their short length and conversational nature; thus, identifying a claim's context is sometimes key for enabling effective fact-checking [13].

The full identification and verification pipeline is displayed in Figure 1. The four tasks are defined as follows:

Task 1 Check-worthiness estimation for tweets. Predict which tweet from a stream of tweets on a topic should be prioritized for fact-checking.

Task 2 Verified claim retrieval: Given a check-worthy tweet, and a set of claims previously checked, determine whether the claim in the tweet has been fact-checked already.

Task 3 Evidence retrieval. Given a check-worthy claim in a tweet on a specific topic and a set of text snippets extracted from potentially-relevant Web pages, return a ranked list of evidence snippets for the claim.

Task 4 Claim verification. Given a check-worthy claim in a tweet and a set of potentially-relevant Web pages, estimate the veracity of the claim.

Task 5 complements the lab. It is as Task 1, but on political debates ad speeches rather than on tweets: given a debate segmented into sentences, together with speaker information, prioritize sentences for fact-checking.

Figure 1 shows how the different tasks relate to each other. The first step is to detect tweets that contain check-worthy claims (Task 1; also, Task 5, which is on debates and speeches). The next step is to check whether a target check-worthy claim has been previously fact-checked (Task 2). If not, then there is a need for fact-checking, which involves supporting evidence retrieval (Task 3), followed by actual fact-checking based on that evidence (Task 4). Tasks 1, 3, and 4 were run for Arabic, while Tasks 1, 2 and 5 were offered for English.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 describes the tasks that were run in Arabic (Tasks 1, 3 and 4). Section 4 presents the tasks that were run in English (Tasks 1, 2, and 5). Note that Sections 3 and 4 are not exhaustive; the reader should refer to [27] and [46], respectively, for further details. Finally, Section 5 concludes with final remarks.

---

[1] https://sites.google.com/view/clef2020-checkthat/

[2] The 2018 edition [41] focused on the identification and verification of claims in political debates. Beside political debates, the 2019 edition [15,16] also focused on isolated claims in conjunction with a closed set of Web documents to retrieve evidence from.

[3] Recently, Twitter started flagging some tweets that violate its policy.
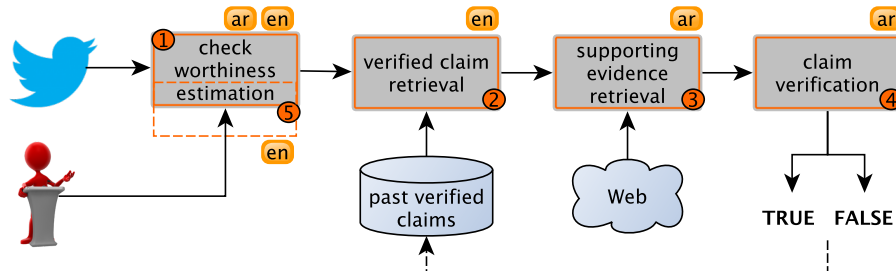
Fig. 1: The `CheckThat!` claim verification pipeline. Our tasks cover all four steps of the pipeline in Arabic or English. Tasks 1–4 focus on Twitter, while task 5 is run on political debates and speeches.

2 Related Work

Both the information retrieval and the natural language processing communities have invested significant efforts in the development of systems to deal with disinformation, misinformation, factuality, and credibility. There has been work on checking the factuality/credibility of a claim, of a news article, or of an information source [5,6,7,33,36,40,44,54]. Claims can come from different sources, but special attention has been paid to those originating in social media [22,39,47,53]. Check-worthiness estimation is still a relatively under-explored problem, and has been previously addressed primarily in political debates and speeches [19,29,30,31,51], and only recently in social media [1]. Similarly, severely under-explored is the task of detecting previously fact-checked claims [45].

This is the third edition of the `CheckThat!` lab, and it represents a clear evolution from the tasks that were featured in the previous two editions. Figure 2 shows the evolution of the `CheckThat!` tasks over these three years. The lab started in 2018 with only two tasks: check-worthiness estimation and factuality (fact-checking), with focus on political debates, speeches, and claims. In that first edition, the English language was leading and the Arabic datasets were produced by translation (manual or automatic with post-editing). The `CheckThat!` 2019 lab offered a continuity in the check-worthiness task. The Arabic task —still under the factuality umbrella— started to unfold into four subtasks in order to boost the development of models specialized in each of the stages of verification, from the ranking of relevant Websites to the final claim verification. Regarding data, transcriptions of press conferences were added, as well as news to be used to verify the claims. In 2020, we unfolded the claim verification pipeline into four differentiated tasks. Regarding data, all tasks in 2020 turned to micro-blogging with focus on Twitter, with the exception of legacy task 5 on check-worthiness, which focused on political debates and speeches.

Below, we present a brief overview of the tasks and of the most successful approaches in the 2019 and the 2018 editions of the lab. We refer the reader to [16] and to [41] for more detailed overviews of these earlier editions.
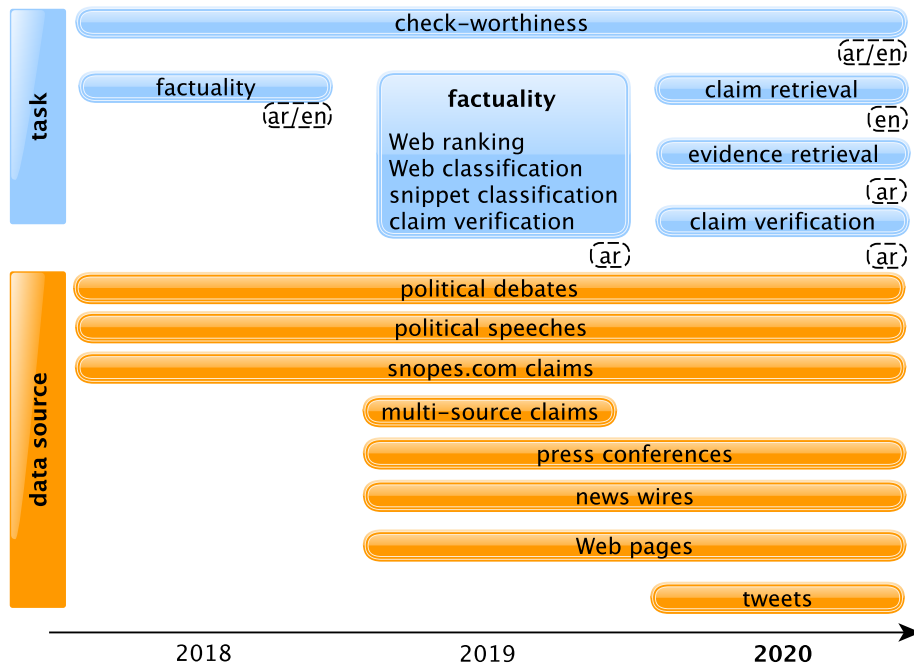
Fig. 2: The evolution of the tasks at the `CheckThat!` lab over its three editions. Top: the unfolding of the tasks that compose a full fact-checking pipeline. Bottom: the source texts and the genres included in the datasets used by these tasks.

2.1 `CheckThat!` 2019

The 2019 edition of the `CheckThat!` lab featured two tasks [16]:

Task 1$_{2019}$. Given a political debate, an interview, or a speech, transcribed and segmented into sentences, rank the sentences by the priority with which they should be fact-checked.

The most successful approaches used by the participating teams relied on neural networks for the classification of the instances instances. For example, Hansen et al. [24] learned domain-specific word embeddings and syntactic dependencies and applied an LSTM classifier. They pre-trained the network with previous Trump and Clinton debates, supervised weakly with the ClaimBuster system. Some efforts were carried out in order to consider context. Favano et al. [17] trained a feed-forward neural network, including the two previous sentences as a context. While many approaches relied on embedding representations, feature engineering was also popular [18]. We refer the interested reader to [4] for further details.

Task 2$_{2019}$. Given a claim and a set of potentially-relevant Web pages, identify which of the pages (and passages thereof) are useful for assisting a human in fact-checking the claim. Finally, determine the factuality of the claim.

The systems for evidence passage identification followed two approaches. BERT was trained and used to predict whether an input passage is useful to fact-check a claim [17]. Other participating systems used classifiers (e.g., SVM) with a variety of features including the similarity between the claim and a passage, bag of words, and named entities [25]. As for predicting claim veracity, the most effective approach used a textual entailment model. The input was represented using word embeddings and external data was also used in training [20]. See [28] for further details.

Note that Task 5 of the 2020 edition of the `CheckThat!` lab is a follow-up of Task 1$_{2019}$, while Task 1 of the 2020 edition is a reformulation that focuses on tweets. In contrast, Task 2$_{2019}$ was decomposed into two tasks in 2020: Tasks 3 and 4.

2.2 `CheckThat!` 2018

The 2018 edition featured two tasks:

Task 1$_{2018}$ was identical to Task 1$_{2019}$.

The most successful approaches used either a multilayer perceptron or an SVM. Zuo et al. [55] enriched the dataset by producing *pseudo-speeches* as a concatenation of all interventions by a debater. They used averaged word embeddings and bag of words as representations. Hansen et al. [23] represented the entries with embeddings, part of speech tags, and syntactic dependencies, and used a GRU neural network with attention as a learning model. More details can be found in the task overview paper [3].

Task 2$_{2018}$. Given a check-worthy claim in the form of a (transcribed) sentence, determine whether the claim is likely to be true, half-true, or false.

The best way to address this task was to retrieve relevant information from the Web, followed by a comparison against the claim in order to assess its factuality. After retrieving such *evidence*, it is fed into the supervised model, together with the claim in order to assess its veracity. In the case of Hansen et al. [23], they fed the claim and the most similar Web-retrieved text to convolutional neural networks and SVMs. Meanwhile, Ghanem et al. [21] computed features, such as the similarity between the claim and the Web text, and the Alexa rank for the website. Once again, this year a similar procedure had to be carried out, but this time explicitly decomposed into tasks 3 and 4. We refer the interested reader to [8] for further details.

3 Overview of the Arabic Tasks

In order to enable research on Arabic claim verification, we ran three tasks from the verification pipeline (see Figure 1) over Arabic tweets. These tasks are check-worthiness on tweets (Task 1), evidence retrieval (Task 3), and claim verification (Task 4). They attracted nine teams. Below, we describe the evaluation dataset created to support each of these tasks. We also present a summary of the approaches used by the participating systems, and we discuss the evaluation results. Further details can be found in our extended overview paper [27].

3.1 Task 1_{ar}. Check-Worthiness on Tweets

Since check-worthiness estimation for tweets in general, and for *Arabic* tweets in particular, is a relatively new task, we constructed a new dataset specifically designed for training and evaluating systems for this task. We identified the need for a "context" that affects check-worthiness of tweets and we used "topics" to represent that context. Given a topic, we define a check-worthy tweet as a tweet that is relevant to the topic, contains one main claim that can be fact-checked by consulting reliable sources, and is important enough to be worthy of verification. More on the annotation criteria is presented later in this section.

Dataset To construct the dataset for this task, we first manually created 15 topics over the period of several months. The topics were selected based on trending topics at the time among Arab social media users. Each topic was represented using a title and a description. Some example topic titles include: "Coronavirus in the Arab World", "Sudan and normalization", and "Deal of the century". Additionally, we augmented the topic by a set of keywords, hashtags and usernames to track in Twitter. Once we created a topic, we immediately crawled a 1-week stream using the constructed search terms, where we searched Twitter (via Twitter search API) using each term by the end of each day. We limited the search to original Arabic tweets (i.e., we excluded retweets). We de-duplicated the tweets and we dropped tweets matching our qualification filter that excludes tweets containing terms from a blacklist of explicit terms and tweets that contain more than four hashtags or more than two URLs. Afterwards, we ranked the tweets by popularity (defined by the sum of their retweets and likes) and selected the top 500 to be annotated.

The annotation process was performed in two steps; we first identified the tweets that are relevant to the topic and contain factual claims, then identified the check-worthy tweets among those relevant tweets.

We first recruited one annotator to annotate each tweet for its relevance to the target topic. In this step, we labeled each tweet as one of three categories:

- Non-relevant tweet for the target topic.
- Relevant tweet but *with no factual claims*, such as tweets expressing opinions about the topic, references, or speculations about the future, etc.
- Relevant tweet that contains a factual claim that can be fact-checked by consulting reliable sources.

Table 1: Summary of the best approaches for Task 1 Arabic for the participating teams. Shown is information about the learning models (including Transformers), about the main representations, whether the participants used external data, and whether they used machine translation to be able to use additional data from English (MT).

Team		Models					Distrib.					Represent.						Other	
		BERT	Bi–LSTM	NN	SVM	SGD	Laser	FastText	GloVe	word2vec	PCA	One-hot	Morphology	Syntax	Sentiment	Dependencies	NER	External data	MT
Accenture	[52]	●																●	●
bigIR	[26]	●																●	
Check_square	[14]			●						●	●	●			●				
DamascusTeam	[32]		●									●						●	
EvolutionTeam	[50]														●		●	●	
NLP&IR@UNED	[37]			●						●									
TOBB ETU	[34]	●							●										
WSSC_UPF	–				●	●	●					●	●					●	

Relevant tweets with factual claims were then labelled for check-worthiness. Two annotators initially annotated the relevant tweets. A third *expert* annotator performed disagreement resolution whenever needed. Due to the subjective nature of check-worthiness, we chose to represent the check-worthiness criteria by several questions, to help annotators think about different aspects of check-worthiness. Annotators were asked to answer the following three questions for each tweet (using a scale of 1-5):

- Do you think the claim in the tweet is of interest to the public?
- To what extent do you think the claim can negatively affect the reputation of an entity, country, etc.?
- Do you think journalists will be interested in covering the spread of the claim or the information discussed by the claim?

Once an annotator answers the above questions, she/he is required to answer the following fourth question considering all the ratings given previously: "Do you think the claim in the tweet is check-worthy?". This question is a yes/no question, and the resulting answer is the label we use to represent check-worthiness in this dataset.

For the final set, all tweets but those labelled as check-worthy were considered not check-worthy. Given 500 tweets annotated for each of the fifteen topics, the annotated set contained 2,062 check-worthy claims (27.5%). Three topics constituted the training set and the remaining twelve topics were used to later evaluate the participating systems.

Table 2: Performance of the best run per team for Arabic Task 1.

RunID	$P@10$	$P@20$	$P@30$	**MAP**
Accenture-AraBERT	0.7167	0.6875	0.7000	0.6232
TOBB-ETU-AF	0.7000	0.6625	0.6444	0.5816
bigIR-bert	0.6417	0.6333	0.6417	0.5511
Check_square-w2vposRun2	0.6083	0.6000	0.5778	0.4949
DamascusTeam-Run03	0.5833	0.5750	0.5472	0.4539
NLP&IR@UNED-run4	0.6083	0.5625	0.5333	0.4614
baseline2	0.3500	0.3625	0.3472	0.3149
baseline1	0.3250	0.3333	0.3417	0.3244
EvolutionTeam-Run1	0.2500	0.2667	0.2833	0.2675
WSSC_UPF-RF01	0.1917	0.1667	0.2028	0.2542

Overview of the approaches Eight teams participated in this task submitting a total of 28 runs. Table 1 shows an overview of the approaches. The most successful runs adopted fine-tuning existing pre-trained models, namely AraBERT and multilingual BERT models. Other approaches relied on pre-trained models such as Glove, Word2vec, and Language-Agnostic SEntence Representations (LASER) to obtain embeddings for the tweets, which were fed either to neural network models or to traditional machine learning models such as SVMs. In addition to text representations, some teams included other features to their models, namely morphological and syntactic features, part-of-speech (POS) tags, named entities, and sentiment features.

Evaluation We treated Task 1 as a ranking problem where we expected check-worthy tweets to be ranked at the top. We evaluated the runs using precision at k ($P@k$) and Mean Average Precision (MAP). We considered $P@30$ as the official measure, as we anticipated the user would check maximum of 30 claims per week. We also developed two simple baselines: *baseline 1*, which ranks tweets in descending order based on their popularity score (sum of likes and retweets a tweet has received) and *baseline 2*, which ranks tweets in reverse chronological order, i.e. most-recent first. Table 2 shows the performance of the best run per team in addition to the two baselines, ranked by the official measure. We can see that most teams managed to improve over the two baselines by a large margin.

3.2 Task 3_{ar}. Evidence Retrieval

Dataset For this task, we needed a set of claims and a set of potentially-relevant Web pages from which evidence snippets will be extracted by a system.

We first collected the set of Web pages using the topics we developed for Task 1. While developing the topics, we represented each one by a set of search phrases. We used these phrases in Google Web search daily as we crawled tweets for the topic. By the end of a week, we collected a set of Web pages that was ready to be used for constructing a dataset to evaluate evidence retrieval systems.

As for the set of claims, we draw a random sample from the check-worthy tweets identified for each topic for Task 1. Since data from Task 2, Subtask C in the last year's edition of the lab could be used for training [28], we only released test claims and Web pages from the twelve test topics used in Task 1. The dataset for this task contains a total of 200 claims and 14,742 corresponding Web pages.

Since we seek a controlled method to allow systems to return snippets, which in turn would allow us to label a consistent set of potential evidence snippets, we automatically pre-split these pages into snippets that we eventually released per page. To extract snippets from the Web pages, we first de-duplicated the crawled Web pages using the page URL. Then, we extracted the textual content from the HTML document for each page after removing any markup and scripts. Finally, we detected Arabic text and split it into snippets, where full-stops, question marks, or exclamation marks delimit the snippets. Overall, we extracted 169,902 snippets from the Web pages.

Due to the large number of snippets collected for the claims, annotating all pairs of claims and snippets was not feasible given the limited time. Therefore, we followed a *pooling* method; we annotate pooled evidence snippets returned from submitted runs by the participating systems. Since the official evaluation measure for the task was set to be $P@10$, we first extracted the top 10 evidence snippets returned by each run for each claim. We then created a pool of unique snippets per claim (considering both snippet IDs and content for de-duplication). Finally, a single annotator annotated each snippet for a claim. The annotators were asked to decide whether a snippet contains evidence useful to verify the given claim. An evidence can be statistics, quotes, facts extracted from verified sources, etc.

Overall, we annotated 3,380 snippets. After label propagation, we had 3,720 annotated snippets of which only 95 are evidence snippets. Our annotation volume was limited due to the very small number of runs participating in the task (2 runs submitted by one team).

Overview of the Approaches One team, EvolutionTeam, submitted two runs for this task [50]. They used machine learning models with two different types of features in each of the runs. In one run, they exploited the similarity feature by computing the cosine similarity between the claim and the snippets to rank them accordingly. They also explored the effectiveness of using linguistic features to rank snippets for usefulness in the second run for which they reported use of external data.

Evaluation This task is modeled as a ranking problem where the system is expected to return evidence at the top of the list of returned snippets. In order to evaluate the submitted runs, we computed $P@k$ at different cutoff ($k = 1$, 5, 10). The official measure was $P@10$. The team's best-performing run achieved an average $P@10$ of 0.0456 over the claims.

3.3 Task 4_{ar}. Claim Verification

Starting with the same 200 claims used in Task 3, one expert fact-checker verified each claim's veracity. We limited the annotation categories to two, true and false, excluding partially-true claims. A true claim is a claim that is supported by a reliable source that confirms the authenticity of the information published in the tweet. A false claim can be a claim that mentions information contradicting that in a reliable source or has been explicitly refuted by a reliable source.

Dataset The claims in the tweets were annotated considering two main factors; the content of the tweet (claim) and the date of the tweet publication. For the annotation, we considered supporting or refuting information that was reported before, on, or few days after the time of the claim. We consulted several reliable sources to verify the claims. The sources that were used differed according to the topic of the claim. For example, for health-related claims, we consulted refereed studies or articles published in reliable medical journals or websites such as APA.[4]

Out of the initial 200 claims, we ended up with 165 claims for which we managed to find a definite label. Six claims among these 165 were found to be False. Since data from Task 2, Subtask D in the last year's edition of the lab can be used for training [28], the final set of 165 annotated claims was used to evaluate the submitted runs.

Evaluation For this task, there were a total of two runs submitted by the same team, EvolutionTeam. The models relied on linguistic features, and they used external data in one of the runs. We treated the task as a classification problem and we used typical evaluation measures for such tasks in the case of class imbalance: Precision, Recall, and F_1 score. The latter was the official evaluation measure. The best-performing run achieved a macro-averaged F_1 score of 0.5524.

4 Overview of the English Tasks

This year we proposed three of the tasks of the verification pipeline in English: check-worthiness estimation over tweets, verified claim retrieval, and check-worthiness estimation in political debates and speeches (cf. Figure 1). A total of 18 teams participated in the English tasks.

4.1 Task 1_{en}. Check-Worthiness on Tweets

Task 1 (English) *Given a topic and a stream of potentially-related tweets, rank the tweets according to their check-worthiness for the topic.*

Previous work on check-worthiness focused primarily on political debates and speeches, while here we focus on tweets instead.

---

[4] https://www.apa.org/

Dataset We focused on a single topic, namely *COVID-19*, and we collected tweets that matched one of the following keywords and hashtags: *#covid19, #CoronavirusOutbreak, #Coronavirus, #Corona, #CoronaAlert, #CoronaOutbreak, Corona*, and *covid-19*. We ran all the data collection in March 2020, and we selected the most retweeted tweets for manual annotation.

For the annotation, we considered a number of factors. These include tweet popularity in terms of retweets, which is already taken into account as part of the data collection process. We further asked the annotators to answer the following five questions:[5]

- **Q1: Does the tweet contain a verifiable factual claim?** This is an objective question. Positive examples include[6] tweets that state a definition, mention a quantity in the present or the past, make a verifiable prediction about the future, reference laws, procedures, and rules of operation, discuss images or videos, and state correlation or causation, among others.
- **Q2: To what extent does the tweet appear to contain false information?** This question asks for a subjective judgment; it does not ask for annotating the actual factuality of the claim in the tweet, but rather whether the claim appears to be false.
- **Q3: Will the tweet have an effect on or be of interest to the general public?** This question asks for an objective judgment. Generally, claims that contain information related to potential cures, updates on number of cases, on measures taken by governments, or discussing rumors and spreading conspiracy theories should be of general public interest.
- **Q4: To what extent is the tweet harmful to the society, person(s), company(s) or product(s)?** This question also asks for an objective judgment: to identify tweets that can negatively affect society as a whole, but also specific person(s), company(s), product(s).
- **Q5: Do you think that a professional fact-checker should verify the claim in the tweet?** This question asks for a subjective judgment. Yet, its answer should be informed by the answer to questions Q2, Q3 and Q4, as a check-worthy factual claim is probably one that is likely to be false, is of public interest, and/or appears to be harmful.

For the purpose of the task, we consider as worth fact-checking the tweets that received a positive answer both to Q1 and to Q5; if there was a negative answer for either Q1 or Q5, the tweet was considered not worth fact-checking. The answers to Q2, Q3, and Q4 were not considered directly, but they helped the annotators make a better decision for Q5.

The annotations were performed by 2–5 annotators independently, and then consolidated after a discussion for the cases of disagreement. The annotation setup was part of a broader COVID-19 annotation initiative; see [1] for more details about the annotation instructions and setup.

---

[5] We used the following MicroMappers setup for the annotations:
http://micromappers.qcri.org/project/covid19-tweet-labelling/

[6] This is influenced by [35].

Table 3: **Task 1, English:** Statistics about the tweets in the dataset.

Partition	Total	Check-worthy
Train	672	231
Dev	150	59
Test	140	60

Table 3 shows statistics about the data, which is split into training, development, and testing. We can see that the data is fairly balanced with the check-worthy claims making 34-43% of the examples across the datasets.

Table 4: **Task 1, English:** Summary of the approaches used in the primary system submissions. Shown is which systems used transformers, learning models, distributional features, standard features, and other.

Team		Transf			Models						Distrib.				Features					Other
		BERT	RoBERTa	Huggingface	BiLSTM	CNN	Rnd forest	Linear reg	Logistic reg	SVM	FastText	GloVe	PCA	Topic models	tf-idf	Dependencies	POS	NEs	Ext. data	Graph relations
Accenture	[52]	●																		
BustingMisinformation	–					●			●		●			●	●					
Check_square	[14]	●							●			●				●	●	●		
Factify	–	●																	●	
NLP&IR@UNED	[37]					●								●						●
QMUL-SDS	[2]	●				●														
Team_Alex	[42]		●																	
TheUofSheffield	[38]						●				●				●					
TOBB ETU	[34]	●								●							●			
UAICS	–	●		●																
SSN_NLP	–		●																	
ZHAW	–						●										●	●		

Evaluation This is a ranking task, where a tweet has to be ranked according to its check-worthiness. Therefore, we consider mean average precision (MAP) as the official evaluation measure, which we complement with reciprocal rank (RR), R-precision (R-P), and P@k for $k \in \{1, 3, 5, 10, 20, 30\}$. The data and the evaluation scripts are available online.[7]

---

[7] https://github.com/sshaar/clef2020-factchecking-task1/

Table 5: **Task 1, English:** Evaluation results for the primary submissions.

Team	MAP	RR	R-P	P@1	P@3	P@5	P@10	P@20	P@30
Accenture	**0.806**	**1.000**	**0.717**	**1.000**	**1.000**	**1.000**	**1.000**	**0.950**	**0.740**
Team_Alex	0.803	**1.000**	0.650	**1.000**	**1.000**	**1.000**	**1.000**	**0.950**	**0.740**
Check_square	0.722	**1.000**	0.667	**1.000**	0.667	0.800	0.800	0.800	0.700
QMUL-SDS	0.714	**1.000**	0.633	**1.000**	**1.000**	**1.000**	0.900	0.800	0.640
TOBB ETU	0.706	**1.000**	0.600	**1.000**	**1.000**	**1.000**	0.900	0.800	0.660
SSN_NLP	0.674	**1.000**	0.600	**1.000**	**1.000**	0.800	0.800	0.800	0.620
Factify	0.656	0.500	0.683	0.000	0.333	0.600	0.700	0.750	0.700
BustingMisinformation	0.617	**1.000**	0.583	**1.000**	**1.000**	0.800	0.700	0.600	0.600
NLP&IR@UNED	0.607	**1.000**	0.567	**1.000**	**1.000**	**1.000**	0.700	0.600	0.580
Baseline (n-gram)	0.579	1.000	0.500	1.000	0.667	0.800	0.800	0.700	0.600
ZHAW	0.505	0.333	0.533	0.000	0.333	0.400	0.600	0.500	0.520
UAICS	0.495	**1.000**	0.467	**1.000**	0.333	0.400	0.600	0.600	0.460
TheUofSheffield	0.475	0.250	0.533	0.000	0.000	0.400	0.200	0.350	0.480

Overview of the approaches A total of 12 teams took part in Task 1. The submitted models range from state-of-the-art Transformers such as BERT and RoBERTa to more traditional machine learning models such as SVM and Logistic Regression. Table 4 shows a summary of the approaches used by the primary submissions of the participating teams. The highest overall score was achieved using a RoBERTa model.

The top-ranked team **Accenture** used RoBERTa with mean pooling and dropout.

The second-best **Team_Alex** trained a logistic regression classifier using as features the RoBERTaś cross-validation predictions on the data and metadata from the provided JSON file as features.

Team **Check_square** used BERT embeddings along with syntactic features with SVM/PCA and ensembles.

Team **QMUL-SDS** fine-tuned the uncased COVID-Twitter-BERT architecture, which was pre-trained on COVID-19 Twitter stream data.

Team **TOBB ETU** used BERT and word embeddings as features in a logistic regression model, adding POS tags and important hand-crafted word features.

Team **SSN_NLP** also used a RoBERTa classifier.

Team **Factify** submitted a BERT-based classifier.

Team **BustingMisinformation** used an SVM with TF-IDF features and GloVe embeddings, along with topic modelling using NMF.

Team **NLP&IR@UNED** trained a bidirectional LSTM on top of GloVe embeddings. They increased the number of inputs with a graph generated from the additional information provided for each tweet.

Team **ZHAW** used a logistic regression with POS tags and named entities along with additional features about the location of posting, its time, etc.

Team **UAICS** submitted predictions from a fine-tuned custom BERT large model.

Table 6: **Task 2, English:** example input. A subset of verified claims ordered by relevance with respect to the input claim according to our baseline system.

input tweet:	A big scandal at @ABC News. They got caught using really gruesome FAKE footage of the Turks bombing in Syria. A real disgrace. Tomorrow they will ask softball questions to Sleepy Joe Biden's son, Hunter, like why did Ukraine & China pay you millions when you knew nothing? Payoff? — Donald J. Trump (@realDonaldTrump) October 15, 2019
verified claims:	**(1)** ABC News mistakenly aired a video from a Kentucky gun range during its coverage of Turkey's attack on northern Syria in October 2019. **(2)** In a speech to U.S. military personnel, President Trump said if soldiers were real patriots, they wouldn't take a pay raise. **(3)** Former President Barack Obama tweeted: "Ask Ukraine if they found my birth certificate."

Team **TheUofSheffield** trained a custom 4-gram FastText model. Their pre-processing includes lowercasing, lemmatization, as well as URL, emoji, stop words, and punctuation removal.

Table 5 shows the performance of the primary submissions to Task 1 in English. We can see that Accenture and Team_Alex achieved very high scores on all evaluation measures and outperformed the remaining teams by a wide margin, e.g., by about eight points absolute in terms of MAP. We can further see that most systems managed to outperform an n-gram baseline by a very sizeable margin.

4.2   Task 2_{en}. Verified Claim Retrieval

Task 2 (English) *Given a check-worthy input claim and a set of verified claims, rank those verified claims, so that the claims that can help verify the input claim, or a sub-claim in it, are ranked above any claim that is not helpful to verify the input claim.*

Unlike the other tasks of the `CheckThat!`lab, Task 2 is a new one. Table 6 shows an example of a tweet by Donald Trump claiming that a video footage about Syria aired by BBC is fake (input claim), and it further shows some already verified claims ranked by their relevance with respect to the input claim.

Note that the input claim and the most relevant verified claim, while expressing the same concept, are phrased quite differently. A good system for ranking the verified claims might greatly reduce the time that a fact-checkers or a journalist would need to check whether a given input claim has already been fact-checked.

Each input claim was retrieved from the fact-checking website Snopes,[8] which dedicates an article to assessing the truthfulness of each claim they have analyzed. In that article, there might be listed different tweets that contain (a

---

[8] www.snopes.com

Table 7: **Task 2, English:** summary of the approaches used by the primary system submissions. We report which systems used search engines scores, scoring functions (supervised or not), representations (other than Transformers), and the removal of tokens. We further indicate whether external data was used.

Team		Engine		Scoring							Repr.			Removal					
		Terrier	ElasticSearch	LambdaMART	BERT	RoBERTa	Unspecified Transf.	KD search	SVM	Cosine	tf-idf	BM25	Term dependencies	URL removal	Emoji removal	Time removal	Username removal	Hashtag removal	External data
Buster.AI	[9]					●													●
Check_square	[14]						●	●											
elec-dlnlp	–				●						●								
iit	–				●				●					●	●	●			
TheUofSheffield	[38]									●	●	●		●			●	●	
trueman	–					●													
UB_ET	[49]	●			●						●	●	●						
UNIPI-NLE	[43]		●		●					●									

paraphrase of) the target claim. Together with the title of the article page and the rating of the claim, as assigned by Snopes, we collect all those tweets and we use them as input claims. Then, the task is, given such a tweet, to find the corresponding claim. The set of target claims consists of the claims that correspond to the tweets we collected, augmented with all Snopes claims collected by ClaimsKG [48]. Note that we have just one list of verified claims, which is used for matching by all input tweets.

Our data consists of 1,197 input tweets, which we split into training (800 input tweets), development (197 tweets), and test set (200 tweets). These input tweets are to be matched against a set of 10,375 verified claims.

Overview of the approaches A total of eight teams participated in Task 2. A variety of scoring functions have been tested, based on supervised learning such as BERT and its variants and SVM, to unsupervised approaches such as simple cosine similarity and scores produced by Terrier and Elastic Search. Two teams focused also on data cleaning by removing URLs, hashtags, usernames and emojis from the tweets. Table 7 shows a summary of the approaches used by the primary submissions of the participating teams.

The winning team, **Buster.AI**, cleaned the tweets from non-readable input and used a pre-trained and fine-tuned version of RoBERTa to build their system.

Team **UNIPI-NLE** performed two cascade fine-tuning of a sentence-BERT model. Initially, they fine-tuned on the task of predicting the cosine similarity for

Table 8: **Task 2, English:** performance for the primary submissions and for an Elastic Search (ES) baseline.

Team	MAP				Precision			RR		
	@1	@3	@5	–	@1	@3	@5	@1	@3	@5
Buster.AI	**0.897**	**0.926**	**0.929**	**0.929**	**0.895**	**0.320**	**0.195**	**0.895**	**0.923**	**0.927**
UNIPI-NLE	0.877	0.907	0.912	0.913	0.875	0.315	0.193	0.875	0.904	0.909
UB_ET	0.818	0.862	0.864	0.867	0.815	0.307	0.186	0.815	0.859	0.862
NLP&IR@UNED	0.807	0.851	0.856	0.861	0.805	0.300	0.185	0.805	0.848	0.854
TheUofSheffield	0.807	0.807	0.807	0.807	0.805	0.270	0.162	0.805	0.805	0.805
trueman	0.743	0.768	0.773	0.782	0.740	0.267	0.164	0.740	0.766	0.771
elec-dlnlp	0.723	0.749	0.760	0.767	0.720	0.262	0.166	0.720	0.747	0.757
Check_square	0.652	0.690	0.695	0.706	0.650	0.247	0.152	0.650	0.688	0.692
baseline (ES)	0.470	0.601	0.609	0.619	0.472	0.249	0.156	0.472	0.603	0.611
iit	0.263	0.293	0.298	0.311	0.260	0.112	0.071	0.260	0.291	0.295

tweet–claim. For each tweet, they trained on 20 random negative verified claims and the gold verified claim. The second fine-tuning step fine-tuned the model as a classification task for which sentence-BERT has to output 1 if the pair is correct, and 0 otherwise. They selected randomly two negative examples and used them with the gold to fine-tune the model. Before inference, they pruned the verified claim list, top-2500 using Elastic Search and simple word matching techniques.

Team **UB_ET** trained their model on a limited number of tweet–claim pairs per tweet. They retrieved the top-1000 tweet–claim pairs for each tweet using the DPH information retrieval weighing model and computed several query-related features and then built a LambdaMart model on top of them.

Team **NLP&IR@UNED** used the Universal Sentence Encoder to obtain embeddings for the tweets and for the verified claims. They then trained a feed-forward neural network using the cosine similarity between a tweet and a verified claim, and statistics about the use of words from different parts of speech.

Team **TheUniversityofSheffield** pre-processed the input tweets in order to eliminate hashtags, and then trained a Linear SVM using as features TF.IDF-weighted cosine similarity and BM25 matching scores between the tweets and the verified claims.

Teams **trueman** and **elec-dlnlp** prepared the input tweets to eliminate hashtags and then used Transformer-based similarity along with Elastic Search scores.

Team **Check_square** fine-tuned sentence-BERT with mined triplets and KD-search.

Team **iit** used cosine similarity using a pre-trained BERT model between the embeddings of the tweet and of the verified claim.

Evaluation The official evaluation measure for Task 2 is MAP@k for $k = 5$. However, we further report MAP for $k \in \{1, 3, 10, 20\}$, overall MAP, R-Precision,

Average Precision, Reciprocal Rank, and Precision@k. Table 8 shows the evaluation results in terms of some of the performance measures for the primary submissions to Task 2. We can see that the winner Buster.AI and the second-best UNIPI-NLE are well ahead of the remaining teams by several points absolute on all evaluation measures. We can further see that most systems managed to outperform an Elastic Search (ES) baseline by a huge margin. The data and the evaluation scripts are available online.[9]

4.3   Task 5$_{en}$. Check-Worthiness on Debates

Task 5 is a legacy task that has evolved from the first edition of `CheckThat!` In each edition, more data from more diverse sources have been added, always with focus on politics. The task focuses on mimicking the selection strategy that fact-checkers, e.g., in PolitiFact, use to select the sentences and the claims to fact-check. The task is defined as follows:

Task 5 (English) *Given a transcript, rank the sentences in the transcript according to the priority to fact-check them.*

We used *PolitiFact* as the main fact-checking source. On *PolitiFact*, often after a major political event such as a public debate or a speech by a government official, a journalist would go through the transcript of the event and would select few claims that would then be fact-checked. These claims would then be discussed in an article about the debate, published on the same site. We collected all such articles, we further obtained the official transcripts of the event from ABCNews, Washington Post, CSPAN, etc. Since sometimes the claims published in the articles are paraphrased, we double-checked and we manually matched them to the transcripts.

We collected a total of 70 transcripts, and we annotated them based on overview articles from *PolitiFact*. The transcripts belonged to one of four types of political events: Debates, Speeches, Interviews, and Town-halls. We used 50 transcripts for training and 20 for testing. We used the older transcripts for training and the more recent ones for testing. Table 9 shows the total number of sentences of the transcripts and the number of sentences that were fact-checked by *PolitiFact*.

Overview of the approaches Three teams participated in this task submitting a total of eight runs. Each of the teams used different text embedding models for the transcripts. The best results were obtained using GloVe's embeddings.

Team **NLP&IR@UNED** used 6B-100D GloVe embeddings as an input to a bidirectional LSTM. They further tried sampling techniques but without success.

Team **UAICS** used the TF.IDF representations using sentences unigrams. They then trained different binary classifiers, such as Logistic regression, Decision Trees, and Naïve Bayes, and they found the latter to perform best.

---

[9] https://github.com/sshaar/clef2020-factchecking-task2/

Table 9: **Task 5, English:** total number of sentences and number of sentences containing claims that are worth fact-checking —organized by type.

Type	Partition	Transcripts	Sentences	Check-worthy
Debates	Train	18	25,688	254
	Test	7	11,218	56
Speeches	Train	18	7,402	163
	Test	8	7,759	50
Interviews	Train	11	7,044	62
	Test	4	2,220	23
Town-halls	Train	3	2,642	8
	Test	1	317	7
Total	**Train**	**50**	**42,776**	**487**
	Test	**20**	**21,514**	**136**

Table 10: Performance of the primary submissions to Task 5 English.

Team	MAP	RR	R-P	P@1	P@3	P@5	P@10	P@20	P@30
NLP&IR@UNED	**0.087**	**0.277**	**0.093**	**0.150**	**0.117**	**0.130**	**0.095**	**0.073**	**0.039**
Baseline (n-gram)	0.053	0.151	0.053	0.050	0.033	0.040	0.055	0.043	0.038
UAICS	0.052	0.225	0.053	**0.150**	0.100	0.070	0.050	0.038	0.027
TOBB ETU P	0.018	0.033	0.014	0.000	0.017	0.020	0.010	0.010	0.006

Team **TOBB ETU** tried fine-tuning BERT and modeling the task as a classification task, but ultimately used part-of-speech (POS) tags with logistic regression and a handcrafted word list from the dataset as their official submission.

Evaluation As this task was very similar to Task 1, but on a different genre, we used the same evaluation measures: MAP as the official measure, and we also report P@k for various values of k. Table 10 shows the performance of the primary submissions of the participating teams. The overall results are quite low, and only one team managed to beat the n-gram baseline. Once again, the data and the evaluation scripts are available online.[10]

5 Conclusion and Future Work

We have described the 2020 edition of the `CheckThat!` lab, intended to foster the creation of technology for the (semi-)automatic identification and verification of claims in social media. The task attracted submissions from 23 teams (up from 14 at CLEF 2019): 18 made submissions for English, and 8 for Arabic. We believe that the technology developed to address the five tasks we have proposed will

---

[10] https://github.com/sshaar/clef2020-factchecking-task5/

be useful not only as a supportive technology for investigative journalism, but also for the lay citizen, which today needs to be aware of the factuality of the information available online.

Acknowledgments

References

1. Alam, F., Shaar, S., Dalvi, F., Sajjad, H., Nikolov, A., Mubarak, H., Da San Martino, G., Abdelali, A., Durrani, N., Darwish, K., Nakov, P.: Fighting the COVID-19 infodemic: Modeling the perspective of journalists, fact-checkers, social media platforms, policy makers, and the society. ArXiv:2005.00033 (2020)
2. Alkhalifa, R., Yoong, T., Kochkina, E., Zubiaga, A., Liakata, M.: QMUL-SDS at CheckThat! 2020: Determining COVID-19 tweet check-worthiness using an enhanced CT-BERT with numeric expressions. In: Cappellato et al. [10]
3. Atanasova, P., Màrquez, L., Barrón-Cedeño, A., Elsayed, T., Suwaileh, R., Zaghouani, W., Kyuchukov, S., Da San Martino, G., Nakov, P.: Overview of the CLEF-2018 CheckThat! lab on automatic identification and verification of political claims. Task 1: Check-worthiness. In: Cappellato et al. [12]
4. Atanasova, P., Nakov, P., Karadzhov, G., Mohtarami, M., Da San Martino, G.: Overview of the CLEF-2019 CheckThat! lab on automatic identification and verification of claims. Task 1: Check-worthiness. In: Cappellato et al. [11]
5. Ba, M.L., Berti-Equille, L., Shah, K., Hammady, H.M.: VERA: A platform for veracity estimation over web data. In: Proceedings of the 25th International Conference Companion on World Wide Web. pp. 159–162. WWW '16 Companion (2016)
6. Baly, R., Karadzhov, G., An, J., Kwak, H., Dinkov, Y., Ali, A., Glass, J., Nakov, P.: What was written vs. who read it: News media profiling using text analysis and social media context. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 3364–3374. ACL '20, Seattle, WA, USA (2020)
7. Baly, R., Karadzhov, G., Saleh, A., Glass, J., Nakov, P.: Multi-task ordinal regression for jointly predicting the trustworthiness and the leading political ideology of news media. In: Proceedings of the 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 2109–2116. NAACL-HLT '19, Minneapolis, MN, USA (2019)
8. Barrón-Cedeño, A., Elsayed, T., Suwaileh, R., Màrquez, L., Atanasova, P., Zaghouani, W., Kyuchukov, S., Da San Martino, G., Nakov, P.: Overview of the CLEF-2018 CheckThat! lab on automatic identification and verification of political claims. Task 2: Factuality. In: Cappellato et al. [12]

9. Bouziane, M., Perrin, H., Cluzeau, A., Mardas, J., Sadeq, A.: Buster.AI at Check-That! 2020: Insights and recommendations to improve fact-checking. In: Cappellato et al. [10]
10. Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.): CLEF 2020 Working Notes (2020)
11. Cappellato, L., Ferro, N., Losada, D., Müller, H. (eds.): Working Notes of CLEF 2019 Conference and Labs of the Evaluation Forum. CEUR Workshop Proceedings, CEUR-WS.org (2019)
12. Cappellato, L., Ferro, N., Nie, J.Y., Soulier, L. (eds.): Working Notes of CLEF 2018–Conference and Labs of the Evaluation Forum. CEUR Workshop Proceedings, CEUR-WS.org (2018)
13. Cazalens, S., Lamarre, P., Leblay, J., Manolescu, I., Tannier, X.: A content management perspective on fact-checking. In: Proceedings of The Web Conference 2018. pp. 565–574. WWW '18 (2018)
14. Cheema, G.S., Hakimov, S., Ewerth, R.: Check_square at CheckThat! 2020: Claim detection in social media via fusion of transformer and syntactic features. In: Cappellato et al. [10]
15. Elsayed, T., Nakov, P., Barrón-Cedeño, A., Hasanain, M., Suwaileh, R., Da San Martino, G., Atanasova, P.: CheckThat! at CLEF 2019: Automatic identification and verification of claims. In: Advances in Information Retrieval. pp. 309–315. Springer International Publishing (2019)
16. Elsayed, T., Nakov, P., Barrón-Cedeño, A., Hasanain, M., Suwaileh, R., Da San Martino, G., Atanasova, P.: Overview of the CLEF-2019 CheckThat!: Automatic identification and verification of claims. In: Experimental IR Meets Multilinguality, Multimodality, and Interaction. pp. 301–321. LNCS, Springer (2019)
17. Favano, L., Carman, M., Lanzi, P.: TheEarthIsFlat's submission to CLEF'19 CheckThat! Challenge. In: Cappellato et al. [11]
18. Gasior, J., Przybyła, P.: The IPIPAN team participation in the check-worthiness task of the CLEF2019 CheckThat! Lab. In: Cappellato et al. [11]
19. Gencheva, P., Nakov, P., Màrquez, L., Barrón-Cedeño, A., Koychev, I.: A context-aware approach for detecting worth-checking claims in political debates. In: Proceedings of the International Conference Recent Advances in Natural Language Processing. pp. 267–276. RANLP'17 (2017)
20. Ghanem, B., Glavaš, G., Giachanou, A., Ponzetto, S., Rosso, P., Rangel, F.: UPV-UMA at CheckThat! lab: Verifying Arabic claims using cross lingual approach. In: Cappellato et al. [11]
21. Ghanem, B., Montes-y Gómez, M., Rangel, F., Rosso, P.: UPV-INAOE-Autoritas - Check That: Preliminary approach for checking worthiness of claims. In: Cappellato et al. [12]
22. Gupta, A., Kumaraguru, P., Castillo, C., Meier, P.: TweetCred: Real-time credibility assessment of content on Twitter. In: Proceeding of the 6th International Social Informatics Conference. pp. 228–243. SocInfo'142 (2014)
23. Hansen, C., Hansen, C., Simonsen, J., Lioma, C.: 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: Cappellato et al. [12]
24. Hansen, C., Hansen, C., Simonsen, J., Lioma, C.: Neural weakly supervised fact check-worthiness detection with contrastive sampling-based ranking loss. In: Cappellato et al. [11]
25. Haouari, F., Ali, Z., Elsayed, T.: bigIR at CLEF 2019: Automatic verification of Arabic claims over the web. In: Cappellato et al. [11]

26. Hasanain, M., Elsayed, T.: bigIR at CheckThat! 2020: Multilingual BERT for ranking Arabic tweets by check-worthiness. In: Cappellato et al. [10]

27. Hasanain, M., Haouari, F., Suwaileh, R., Ali, Z., Hamdan, B., Elsayed, T., Barrón-Cedeño, A., Da San Martino, G., Nakov, P.: Overview of CheckThat! 2020 Arabic: Automatic identification and verification of claims in social media. In: Cappellato et al. [10]

28. Hasanain, M., Suwaileh, R., Elsayed, T., Barrón-Cedeño, A., Nakov, P.: Overview of the CLEF-2019 CheckThat! lab on automatic identification and verification of claims. Task 2: Evidence and factuality. In: Cappellato et al. [11]

29. Hassan, N., Li, C., Tremayne, M.: Detecting check-worthy factual claims in presidential debates. In: Proceedings of the 24th ACM International Conference on Information and Knowledge Management. pp. 1835–1838. CIKM '15 (2015)

30. Hassan, N., Tremayne, M., Arslan, F., Li, C.: Comparing automated factual claim detection against judgments of journalism organizations. In: Computation+Journalism Symposium (2016)

31. Hassan, N., Zhang, G., Arslan, F., Caraballo, J., Jimenez, D., Gawsane, S., Hasan, S., Joseph, M., Kulkarni, A., Nayak, A.K., et al.: Claimbuster: The first-ever end-to-end fact-checking system. Proceedings of the VLDB Endowment **10**(12), 1945–1948 (2017)

32. Hussein, A., Hussein, A., Ghneim, N., Joukhadar, A.: DamascusTeam at CheckThat! 2020: Check worthiness on Twitter with hybrid CNN and RNN models. In: Cappellato et al. [10]

33. Karadzhov, G., Nakov, P., Màrquez, L., Barrón-Cedeño, A., Koychev, I.: Fully automated fact checking using external sources. In: Proceedings of the International Conference Recent Advances in Natural Language Processing. pp. 344–353. RANLP' 17 (2017)

34. Kartal, Y.S., Kutlu, M.: TOBB ETU at CheckThat! 2020: Prioritizing English and Arabic claims based on check-worthiness. In: Cappellato et al. [10]

35. Konstantinovskiy, L., Price, O., Babakar, M., Zubiaga, A.: Towards automated factchecking: Developing an annotation schema and benchmark for consistent automated claim detection. arXiv:1809.08193 (2018)

36. Ma, J., Gao, W., Mitra, P., Kwon, S., Jansen, B.J., Wong, K.F., Cha, M.: Detecting rumors from microblogs with recurrent neural networks. In: Proceedings of the International Joint Conference on Artificial Intelligence. pp. 3818–3824. IJCAI '16 (2016)

37. Martinez-Rico, J., Araujo, L., Martinez-Romo, J.: NLP&IR@UNED at CheckThat! 2020: A preliminary approach for check-worthiness and claim retrieval tasks using neural networks and graphs. In: Cappellato et al. [10]

38. McDonald, T., Dong, Z., Zhang, Y., Hampson, R., Young, J., Cao, Q., Leidner, J., Stevenson, M.: The University of Sheffield at CheckThat! 2020: Claim identification and verification on Twitter. In: Cappellato et al. [10]

39. Mitra, T., Gilbert, E.: Credbank: A large-scale social media corpus with associated credibility annotations. In: Proceedings of the Ninth International AAAI Conference on Web and Social Media. pp. 258–267. ICWSM '15 (2015)

40. Mukherjee, S., Weikum, G.: Leveraging joint interactions for credibility analysis in news communities. In: Proceedings of the 24th ACM International Conference on Information and Knowledge Management. pp. 353–362. CIKM'15 (2015)

41. Nakov, P., Barrón-Cedeño, A., Elsayed, T., Suwaileh, R., Màrquez, L., Zaghouani, W., Gencheva, P., Kyuchukov, S., Da San Martino, G.: Overview of the CLEF-2018 lab on automatic identification and verification of claims in political debates.

In: Working Notes of CLEF 2018 – Conference and Labs of the Evaluation Forum. CLEF '18, Avignon, France (2018)

42. Nikolov, A., Da San Martino, G., Koychev, I., Nakov, P.: Team_Alex at CheckThat! 2020: Identifying check-worthy tweets with transformer models. In: Cappellato et al. [10]

43. Passaro, L., Bondielli, A., Lenci, A., Marcelloni, F.: UNIPI-NLE at CheckThat! 2020: Approaching fact checking from a sentence similarity perspective through the lens of transformers. In: Cappellato et al. [10]

44. Popat, K., Mukherjee, S., Strötgen, J., Weikum, G.: Credibility assessment of textual claims on the web. In: Proceedings of the 25th ACM International Conference on Information and Knowledge Management. pp. 2173–2178. CIKM '16 (2016)

45. Shaar, S., Babulkov, N., Da San Martino, G., Nakov, P.: That is a known lie: Detecting previously fact-checked claims. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 3607–3618. ACL '20 (2020)

46. Shaar, S., Nikolov, A., Babulkov, N., Alam, F., Barrón-Cedeño, A., Elsayed, T., Hasanain, M., Suwaileh, R., Haouari, F., Da San Martino, G., Nakov, P.: Overview of CheckThat! 2020 English: Automatic identification and verification of claims in social media. In: Cappellato et al. [10]

47. Shu, K., Sliva, A., Wang, S., Tang, J., Liu, H.: Fake news detection on social media: A data mining perspective. SIGKDD Explor. Newsl. **19**(1), 22–36 (2017)

48. Tchechmedjiev, A., Fafalios, P., Boland, K., Gasquet, M., Zloch, M., Zapilko, B., Dietze, S., Todorov, K.: ClaimsKG: A knowledge graph of fact-checked claims. In: Proceedings of the 18th International Semantic Web Conference. pp. 309–324. ISWC '19, Auckland, New Zealand (2019)

49. Thuma, E., Motlogelwa, N.P., Leburu-Dingalo, T., Mudongo, M.: UB_ET at CheckThat! 2020: Exploring ad hoc retrieval approaches in verified claims retrieval. In: Cappellato et al. [10]

50. Touahri, I., Mazroui, A.: EvolutionTeam at CheckThat! 2020: Integration of linguistic and sentimental features in a fake news detection approach. In: Cappellato et al. [10]

51. Vasileva, S., Atanasova, P., Màrquez, L., Barrón-Cedeño, A., Nakov, P.: 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. pp. 1229–1239. RANLP '19 (2019)

52. Williams, E., Rodrigues, P., Novak, V.: Accenture at CheckThat! 2020: If you say so: Post-hoc fact-checking of claims using transformer-based models. In: Cappellato et al. [10]

53. Zhao, Z., Resnick, P., Mei, Q.: Enquiring minds: Early detection of rumors in social media from enquiry posts. In: Proceedings of the 24th International Conference on World Wide Web. pp. 1395–1405. WWW'15 (2015)

54. Zubiaga, A., Liakata, M., Procter, R., Hoi, G.W.S., Tolmie, P.: Analysing how people orient to and spread rumours in social media by looking at conversational threads. PLoS ONE **11**(3) (2016)

55. Zuo, C., Karakas, A., Banerjee, R.: A hybrid recognition system for check-worthy claims using heuristics and supervised learning. In: Cappellato et al. [12]