The Role of Context in Detecting Previously Fact-Checked Claims

Shaden Shaar[1], **Firoj Alam**[1], **Giovanni Da San Martino**[2], **Preslav Nakov**[1]
[1]Qatar Computing Research Institute, HBKU, Doha, Qatar
[2]University of Padova, Italy
{sshaar, falam, pnakov}@hbku.edu.qa
dasan@math.unipd.it

Abstract

Recent years have seen the proliferation of disinformation and fake news online. Traditional approaches to mitigate these issues is to use manual or automatic fact-checking. Recently, another approach has emerged: checking whether the input claim has previously been fact-checked, which can be done automatically, and thus fast, while also offering credibility and explainability, thanks to the human fact-checking and explanations in the associated fact-checking article. Here, we focus on claims made in a political debate and we study the impact of modeling the context of the claim: both on the source side, i.e., in the debate, as well as on the target side, i.e., in the fact-checking explanation document. We do this by modeling the local context, the global context, as well as by means of co-reference resolution, and multi-hop reasoning over the sentences of the document describing the fact-checked claim. The experimental results show that each of these represents a valuable information source, but that modeling the source-side context is most important, and can yield 10+ points of absolute improvement over a state-of-the-art model.

1 Introduction

The fight against dis/mis-information has become an urgent social and political matter. Online media have been widely used not only for social good, but also to mislead entire communities. Many fact-checking organizations, such as FactCheck.org,[1] Snopes,[2] PolitiFact,[3] and FullFact,[4] as well as some broader international initiatives such as the *Credibility Coalition*[5] and *Eufactcheck*,[6] have emerged to address the problem (Stencel, 2019).

[1]http://www.factcheck.org/
[2]http://www.snopes.com/fact-check/
[3]http://www.politifact.com/
[4]http://fullfact.org/
[5]https://credibilitycoalition.org/
[6]https://eufactcheck.eu/

There have also been efforts to develop automatic systems to detect such content (Vo and Lee, 2018; Shu et al., 2017; Thorne and Vlachos, 2018; Li et al., 2016; Lazer et al., 2018; Vosoughi et al., 2018a; Nguyen et al., 2020), including the development of datasets (Augenstein et al., 2019), systems (Chernyavskiy et al., 2021b), and evaluation campaigns (Barrón-Cedeño et al., 2020; Nakov et al., 2021b,c; Shaar et al., 2021a; Nakov et al., 2022b).

An important issue with automatic systems is that journalists and fact-checkers often question their credibility for reasons such as (perceived) insufficient accuracy given the state of present technology, but also due to the lack of explanation about how the system has made its decision. On the other hand, manual fact-checking is time-consuming and does not scale. Yet, time is precious: it has been reported in the literature that *fake news* travels faster than real news (Vosoughi et al., 2018b), and that 50% of the spread of some very viral false claims has happened within the first ten minutes after they got published (Zaman et al., 2014). Such findings show the importance of real-time fake news detection, which can enable a timely intervention.

As both manual and automatic systems have their limitations, there have been proposals for human-in-the-loop settings, aiming to bring the best of both worlds. In order to enable such an approach, one question that arises is how to facilitate fact-checkers and journalists with automated systems (Nakov et al., 2021a). An immediate problem is to know whether a given input claim has been previously fact-checked by a reputable fact-checking organization. This would give the journalist a credible reference and could save her significant amount of time, as manually fact-checking a single non-trivial claim may take from 1-2 days to 1-2 weeks. While earlier studies have suggested that such a mechanism should be part of an end-to-end automated system, there has been limited work in this direction (Shaar et al., 2020a; Vo and Lee, 2020).

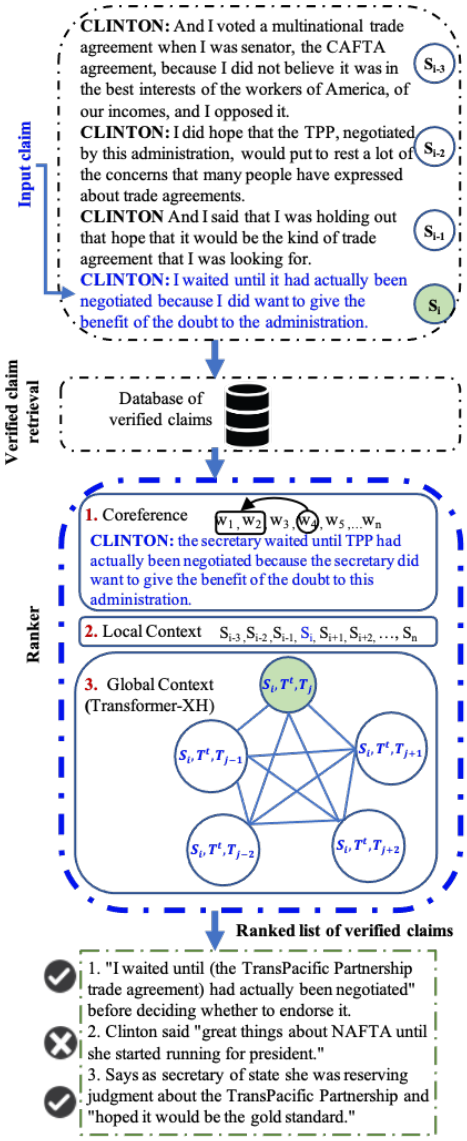

Figure 1: A pipeline of retrieving and ranking previously fact-checked claims. S_i is the claim (source), T^t is the title of the target, T_j is a sentence from the target.

At the time of COVID-19, there are a number of false claims and conspiracy theories spreading online, e.g., about Bill Gates and his chips in the COVID-19 vaccine, about garlic water as a cure, etc. Many such claims have already been debunked, but this does not stop them, as they keep being repeated, potentially in a slightly different form but with the same meaning. Thus, it is important to recognize such variations quickly, and possibly to post a reply in social media with a link to a fact-checking article. Similarly, in a scenario where a politician is being interviewed or is taking part in a debate, a quick check against a collection of previously fact-checked claims would make it possible to put him/her on the spot in real time.

However, the problem in a real-time scenario is that, unlike written text, interviews, debates, and speeches are more spontaneous, and the claims that are being made are often not clearly formulated in a single sentence. This is illustrated in Figure 1, where we can see a fragment from a Democratic debate as part of the 2016 US Presidential election, where Hillary Clinton said: "*I waited until it had actually been negotiated because I did want to give the benefit of the doubt to the administration.*" Understanding this claim requires pronominal co-reference resolution (e.g., what does *it* refer to, is it *CAFTA* or is it *TPP*, as both are mentioned in the previous sentences?), more general co-reference (e.g., that the administration being discussed is the *Obama* administration), as well as general understanding of the conversation so far, and possibly general world knowledge about US politics at the time of the debate (e.g., that Hillary Clinton was Secretary of State when TPP was being discussed).

Moreover, previous research has shown that it is beneficial to match the input claim not only against the canonical verified claim that fact-checkers worked with, but against the entire article that they wrote explaining why the claim was ultimately judged to be true/false (Shaar et al., 2020a; Vo and Lee, 2020). This is because, in the fact-checking article, the claim is likely to be paraphrased in different ways, and there could also be background information and related terms, which can facilitate claim matching, and thus improve recall. This means that we need to make use of the global contextual information contained within the full text of the fact-checking article or at least the sentences next to the claim, i.e., the local context. Similarly, for the FEVER fact-checking task, which asks to fact-check against Wikipedia, it has been shown that multi-hop reasoning (Transformer-XH) over the sentences of the target article can help (Zhao et al., 2019), an observation that was further confirmed in the context of fact-checking political claims (Ostrowski et al., 2021). Transformer-XH uses a novel attention mechanism that naturally "hops" across the connected text sequences in addition to attending over tokens within each sequence. As claims and reasoning about them are manifested across documents, this hop-based attention mechanism constructs global contextualized representation to provide better joint multi-evidence reasoning. In the present work, we rely on Transformer-XH to extract and use global contextual information.

Based on the above considerations, we propose a framework that focuses on modeling co-reference, local context (representation from neighboring sentences; see Section 4.2.2), and global context (representation from Transformer-XH; see Section 4.2.3), both on the source and on the target side, while also using multi-hop reasoning over the target side.

Our contributions can be summarized as follows:

- We perform careful manual analysis to understand what makes detecting previously fact-checked claims a hard problem, and we categorize the claims by type. We release these annotations to enable further research.
- Unlike previous work, we focus on modeling the context both on the source side and on the target side, both locally and globally, using co-reference resolution and reasoning with Transformer-XH, which yields sizable improvements over state-of-the-art models of over ten MAP points absolute.
- We propose a realistic and challenging, time-sensitive and document-aware, data split compared to previous work, which we also release.[7]

The rest of the paper is organized as follows. Section 2 provides a brief overview of previous work. Section 3 introduces the dataset development process. Section 4 presents the experiments. Section 5 discusses the evaluation results. Finally, Section 6 concludes with lessons learned and points to possible directions for future work.

2 Related Work

Below, we describe three relevant lines of research: on detecting previously fact-checked claims, on semantic matching and ranking, and on context modeling for factuality.

2.1 Previously Fact-Checked Claims

While there is a surge in research on automatic fact-checking, fully automatic systems suffer from credibility issues, e.g., in the eyes of journalists, and manual checking is still the norm. Thus, it is important to reduce that manual effort by detecting when a claim has already been fact-checked.

[7] https://github.com/firojalam/Detecting-Previously-Fact-Checked-Claims.git

A recent survey has identified the task of detecting previously fact-checked claims as one of the most important ways in which automation can assist human fact-checkers (Nakov et al., 2021a). The task was recognized as an important element of the typical sequence of fact-checking steps (Vlachos and Riedel, 2014): (*i*) extracting statements that are to be fact-checked, (*ii*) constructing appropriate questions, (*iii*) obtaining the pieces of evidence from relevant sources, and (*iv*) reaching a verdict using that evidence. Hassan et al. (2017) also mentioned the task as an important component of their end-to-end fact-checking pipeline, but did not evaluate it as a component on its own right.

Recently, Shaar et al. (2020a) gave a formulation of the task of detecting previously fact-checked claims, and proposed a learning-to-rank approach combining BM25 retrieval with BERT-based semantic matching. They further developed two specialized datasets: (a) on political debates and speeches, using fact-checked claims from PolitiFact, and (b) on tweets, using claims from Snopes.

The CLEF 2020-2022 CheckThat! lab (Barrón-Cedeño et al., 2020; Hasanain et al., 2020; Shaar et al., 2020b; Nakov et al., 2021b,c; Shaar et al., 2021b; Nakov et al., 2022b,c,d) extended these datasets with additional data in English and Arabic, adding more data each year. The best systems (Pritzkau, 2021; Mihaylova et al., 2021; Chernyavskiy et al., 2021a) used a combination of BM25 retrieval, semantic similarity using sentence embeddings (Reimers and Gurevych, 2019), and reranking. Bouziane et al. (2020) further used external data from fact-checking datasets (Wang, 2017; Thorne et al., 2018; Wadden et al., 2020).

Chernyavskiy et al. (2022) fine-tuning BERT using batch-softmax contrastive loss as an alternative to mean squared error and triplet loss, and demonstrated sizable performance gains for a number of sentence scoring tasks, including detecting previously fact-checked claims.

Another recent work by Sheng et al. (2021) highlighted the importance of using lexical, semantic, and pattern-based information and proposed a reranker based on memory-enhanced transformers for claim matching.

Vo and Lee (2020) proposed a multi-modal text+image neural ranking model for detecting previously fact-checked claims about images.

However, none of the above work modeled the context of the input claim, which is our focus here.

2.2 Semantic Matching and Ranking

Here, we focus on the textual formulation of the problem, as defined by Shaar et al. (2020a): given an input claim, we want to detect potentially matching previously fact-checked claims and to rank them accordingly. A related research area is semantic matching and ranking, as matching some *InputClaim–VerClaim* pairs might require sentence embeddings, natural language inference, and coreference resolution. An example of such a difficult pair is shown in Table 1, line 607. Recent relevant work has used neural approaches. Nie et al. (2019) proposed a semantic matching method that combines document retrieval, sentence selection, and claim verification neural models to extract claims and to verify them. Thorne et al. (2018) proposed a simple model, where pieces of evidence are concatenated together and then fed into a Natural Language Inference (NLI) model. Yoneda et al. (2018) used a four-stage approach that combines document and sentence retrieval with NLI. Hanselowski et al. (2018) used a BiLSTM-based enhanced sequential inference model (Chen et al., 2017) to rank candidate facts and to classify a new claim based on the selected facts. Several studies used model combination (i.e., document retrieval, sentence retrieval, and NLI to classify the retrieved sentences) with joint learning (Yoneda et al., 2018; Hidey and Diab, 2018; Luken et al., 2018).

2.3 Context Modeling for Factuality

Previous work has shown that modeling the context can help for predicting the check-worthiness of claims in political debates, e.g., the interaction between the debaters, and the reaction of the moderator and of the public to what was said (Gencheva et al., 2017; Atanasova et al., 2019b; Vasileva et al., 2019). The CLEF 2018-2022 CheckThat! lab had a shared task on this (Atanasova et al., 2018, 2019a; Shaar et al., 2020b, 2021c; Nakov et al., 2022a).

The CLEF-2018 CheckThat! lab featured a shared task on fact-checking a claim in the context of a political debate (Barrón-Cedeño et al., 2018; Nakov et al., 2018), and SemEval-2019 had a shared task on fact-checking in community question answering forums (Mihaylova et al., 2019).

Liu et al. (2020) proposed a kernel graph attention network to model evidence as a context for fact verification. Similarly, Zhou et al. (2019) used a fully connected evidence graph with multi-evidence information sources for fact verification.

Zhong et al. (2020) used different pre-trained Transformer models and a graph-based approach, i.e., graph convolutional network and graph attention network, for fact verification. Zhao et al. (2019) introduced extra hop attention to incorporate contextual information, while maintaining the Transformer capabilities, thus making it possible to learn a global representation of the different pieces of evidence and to jointly reason over the evidence graph. One of the limitations of their approach was the need for human-labeled evidence in relation to the input claims in existing fact-verification datasets. Ostrowski et al. (2021) developing a dataset of annotated pieces of evidence associated with input claims and used multihop attention to make a prediction about the factuality of a claim.

Unlike the above work, here we target a different task: detecting previously fact-checked claims as opposed to check-worthiness prediction or fact-checking a claim. Moreover, while the above work was limited to the target context, here we also model the source context, which turns out to be much more important.

3 Dataset

Here, we focus on the problem of detecting previously fact-checked claims, using the task formulation and an adaptation of data from (Shaar et al., 2020a). They had two datasets: one on matching tweets against Snopes claims, and another one on matching claims in the context of a political debate to PolitiFact claims. Here, we focus on the latter,[8] and we perform a close analysis of the claims and what makes them easy/hard to match.

We experimented with their PolitiFact dataset, which targets claims related to US politics. After a US political debate, speech, or interview, fact-checking journalists from PolitiFact would select few claims made in the event and would verify them either from scratch or by linking them to a previously fact-checked claim. Each fact-checked claim has an associated article stating its degree of factuality along with an explanation of how the fact-checkers arrived at their verdict. The dataset has two parts: (*i*) verified claims {normalized *VerClaim*, article *title*, and article *text*}, (*ii*) transcripts of the political events (e.g., debates). Shaar et al. (2020a) annotated the data by linking sentences from the transcript (*InputClaim*) to one or more verified claims (out of 16,636 claims in PolitiFact).

[8] github.com/sshaar/That-is-a-Known-Lie

Line No.	Type		Input Claim	Verified Claim
255	*clean*	D. Trump:	*Hillary Clinton wanted the wall.*	Says Hillary Clinton "wanted the wall."
695	*part-of*	C. Wallas:	*And since then, as we all know, nine women have come forward and have said that you either groped them or kissed them without their consent.*	The stories from women saying he groped or forced himself on them "largely have been debunked."
		⋮		
699	*part-of*	D. Trump:	*Well, first of all, those stories have been largely debunked.*	The stories from women saying he groped or forced himself on them "largely have been debunked."
688	*clean-hard*	D. Trump:	*She gave us ISIS as sure as you are sitting there.*	Hillary Clinton invented ISIS with her stupid policies. She is responsible for ISIS.
605		D. Trump:	*Now she wants to sign TransPacific Partnership.*	
		⋮		
607	*context-dep*	D. Trump:	*She lied when she said she didn't call it the gold standard in one of the debates.*	Says Hillary Clinton called the TransPacific Partnership "the gold standard. You called it the gold standard of trade deals. You said its the finest deal youve ever seen."

Table 1: Fragment from the 3rd US Presidential debate in 2016 showing the *verified claims* chosen by PolitiFact and the fine-grained category of the pair. Most input sentences have no *verified claim*, e.g., see line 605.

To further analyze the dataset, we looked at the *InputClaim–VerClaim* pairs, and we manually categorized them into one of the following categories:

1. ***clean* :** A *clean* pair is a self-contained *InputClaim* with a *VerClaim* that directly verifies it (see line 255 in Table 1 as an example).

2. ***clean-hard*:** A *clean-hard* pair is a self-contained *InputClaim* with a *VerClaim* that indirectly verifies it (see line 688 in Table 1).

3. ***part-of*:** A *part-of* pair's *InputClaim* is not self-contained and requires the addition of other sentences from the transcript to fully form a single claim.

4. ***context-dep*:** A *context-dep* pair is similar to *clean* and *clean-hard*, but the *InputClaim* is not self-contained and needs co-reference.

The above categories include all types of pairs we have seen. Moreover, since the dataset is constructed from speeches, debates, and interviews, the structure of the *InputClaim–VerClaim* pairs differs. For example, in debates, we see more *part-of* examples, as there are multiple question–answer claim pairs, as well as back-and-forth arguments splitting the claims into multiple sentences.

The annotations were performed by three annotators who are experts in fact-checking (and co-authors of this paper), using the above definitions for the categories. We consolidated their annotations using majority voting, and they had a consolidation discussion for cases with no majority. The Fleiss Kappa inter-annotator agreement was 0.5, which corresponds to moderate agreement, which is reasonable for such a complex annotation task.

Table 1 shows examples of *InputClaim–VerClaim* pairs that illustrate the four categories. We can see that the task goes beyond simple textual similarity and natural language inference, as the examples in lines 607 and 695–699 show. Moreover, matching *context-dep* pairs (lines 605–607) requires understanding the *InputClaim*'s local context, while matching *clean-hard* pairs (line 688) requires analysis of the overall global context of the *VerClaim*.

Finally, we should note while annotating the data into the above four categories, we found out that a small number of *InputClaim–VerClaim* pairs in (Shaar et al., 2020a) were false matches (which happened, as they did the matching automatically, without manually double-checking every single example). We removed these pairs, and thus our reported number of pairs is slightly lower than theirs.

InputClaim–VerClaim pairs	695	
– *clean*	291	42%
– *clean-hard*	210	30%
– *part-of*	68	10%
– *context-dep*	126	18%
Total # of verified claims (to match against)	16,636	

Table 2: **Statistics about our dataset:** total number of *InputClaim–VerClaim* pairs and of *VerClaims* in PolitiFact to match an *InputClaim* against.

Split	**MAP**
Debate-Level – Chrono	**0.429**
Debate-Level – Semi-chrono	0.539
Debate-Level – Random	0.590
Sentence-Level – Random (Shaar et al., 2020a)	0.602

Table 3: MAP scores of the reranker models when using four different splits representing different scenarios. We use *Debate-Level – Chrono* for our experiments.

Table 2 shows statistics about the distribution of the four categories of claims in our dataset. We can see that *clean* and *clean-hard* are the most frequent categories, while *part-of* is the least frequent one.

We further observed that Shaar et al. (2020a) dealt with each *InputClaim* independently, i.e., at the sentence level. This is problematic because for *part-of* claims we could end up splitting them and putting them in different sets: one in training, and one in testing. Moreover, splitting the dataset in this way means that the examples for a given topic can split between training and test, and thus information can leak, e.g., a claim can be repeated. Therefore, we considered new splits for the data:

- *Debate-Level Chrono*: We split the data chronologically. We use the first 50 debates for training, and the last 20 for testing. Specifically, we have 554 pairs for training, and 141 pairs for testing. This is a more realistic scenario, where we would only have access to earlier debates, and we can use them to make decisions about claims made in future debates. The complexity of this setting is also reflected in the MAP score as shown in Table 3. We see that this score is lower than the best model in previous work (last row). This is because this setting is complex as we use a model trained on debates and speeches from 2012-2018, and we test on debates from 2019. Across those different time frames, different politicians discuss different topics.

- *Debate-Level Semi-Chrono*: We split the data per year, e.g., for year 2018, we divide the transcripts into train and test with 80/20 splits, and then we train and evaluate using the same reranking model. In Table 3, we can see an improvement with this setting compared to the *Debate-Level Chrono* setting. This might be because the same politicians discuss the same/similar issues throughout the same year.

- *Debate-Level Random*: We randomly choose 80% of the debates for training and the remaining ones for testing. This is a comparatively easier setting as the data is randomly distributed in training and testing. This is also reflected in the results in Table 3. The reason could be that politicians repeat themselves a lot, especially in two consecutive political events, and the random split can lead to having two similar debates/speeches in two splits.

- *Sentence Level Random*: This is the setting in (Shaar et al., 2020a), where *sentences* from the debates are randomly divided into train and test in a proportion of 80:20. This is the most unrealistic split.

In our experiments, we chose to use the most realistic, but also the hardest setup: *Debate-Level Chrono*. As a result, our MAP score, when experimenting with the state-of-the-art model of (Shaar et al., 2020a), decreases from 0.602 to 0.429.

4 Experimental Setup

Below, we first introduce the experimental setup for our baseline, and then we describe our proposed model that takes the context of the input claim into account, both on the source and on the target side.

4.1 Baseline

From our analysis of the dataset (described in Section 3), we conclude that (*i*) we need to resolve the references in the *InputClaim*, (*ii*) to capture the local context of the *InputClaim*, and (*iii*) to encapsulate the global context of the *VerClaim*.

For our baseline, we use the setup of the state-of-the-art model of Shaar et al. (2020a). We use the claim as a query against the full text of the documents using BM25. We then train a reranker on the top-100 BM25 results using rankSVM (Herbrich et al., 1999) with an RBF kernel.

The reranker uses nine similarity measures that compare the *InputClaim* to the *VerClaim*, as well as the respective reciprocal ranks. In particular, we compute the BM25 score for *InputClaim* vs. *VerClaim*, *title*, *text*, *VerClaim+title+text*. We also compute the cosine using sentence-BERT embeddings for *InputClaim* vs. *VerClaim*, *title*, and the top-4 sentences from *text*. Using these scores, we create a vector representation of the *InputClaim–VerClaim* pair with dimensionality $\mathbb{R}^{18}$. We then scale the vectors of all *InputClaim–VerClaim* pairs in $[-1; 1]$ and we train a rankSVM with default values of the hyper-parameters: $KernelDegree = 3$, $\gamma = 1/num\_features$, and $\epsilon = 0.001$.

4.2 Proposed Model

As shown in Figure 1, our model uses co-reference resolution on the source and on the target side, the local context (i.e., the neighboring sentences), and the global context (using Transformer-XH) as discussed below. It is still a pairwise reranker, but with a richer context representation.

4.2.1 Co-reference Resolution

We manually inspected the training transcripts and the associated verified claims, and we realized that there were many co-reference dependencies, resolving which could potentially help to obtain more representative textual and contextual similarity scores. As for the verified claims, we noticed that not all *VerClaim*s were self-contained, and that some understanding of the context was needed[9] of the article's *text* that explains the verdict provided by the PolitiFact journalists. Therefore, our hypothesis was that resolving such co-references could improve the downstream matching scores.

For the same reason, we also performed co-reference resolution on the PolitiFact articles when they were used to compute the BM25 scores.

We experimented with various co-reference resolution tools including **NeuralCoref**,[10] **e2e-coref**,[11] and **SpanBERT**,[12] and we found that **NeuralCoref** was best on the input transcripts, while **e2e-coref** was best on the articles about the target *VerClaim*s. Hence, in the rest of our experiments below, we show results using **NeuralCoref** for the source side, and using **e2e-coref** for the target side.

[9] For example, who is speaking or what is being discussed.

[10] github.com/huggingface/neuralcoref

[11] github.com/kentonl/e2e-coref

[12] github.com/facebookresearch/SpanBERT

We resolved the co-reference in the *InputClaim* by performing co-reference resolution on the entire input transcript (as was suggested in the literature); we will refer to this as ***src-coref***. As for the verified claims, we aimed to resolve the co-references both in the *VerClaim* and in the *text* of the PolitiFact articles. We also aimed to ensure that the dependencies from the *text* can be used for the *VerClaim*. Therefore, we concatenated both the text and *VerClaim* (in the same order), and we applied the co-reference model on the concatenated text. We chose this order of concatenation because the published *text* reserves the last paragraph to rephrase the *VerClaim* and to provide a summary of the justification; hence, there is a higher probability to resolve the co-references correctly.

4.2.2 Local Context

Resolving the pronominal co-references allows us to obtain the correct objects and the names the *InputClaim* refers to. However, in the process of analyzing the dataset, we noticed that different *VerClaim*s, although having similar structure, could talk about different things, depending on the article text and also on the surrounding context. Therefore, it is important to understand the context of an *InputClaim*. In particular, we achieve this by performing a feature-level concatenation of the neighboring sentences in the transcript, i.e., we take the eighteen features ($\mathbb{R}^{18}$, as discussed in Section 4.1 above) for the neighboring sentences, and we concatenate them to the similarity score for the *InputClaim*. We then use the resulting representation as a feature vector to be fed into our reranker. For example, if we take three sentences before the *InputClaim* and one sentence after it, we denote this as ***FC***(3, 1).

Let S_i be our *InputClaim*, which is the i'th sentence in the transcript. We compute the similarity measures and the reciprocal rank (as described in Section 4.1) to obtain the vector representation $S_{i,v}$ for S_i. With $k = 3$ previous and $l = 1$ following neighbouring sentences our final feature vector is

$$FC(k = 3, l = 1) = S_{i-3,v} +\!\!+ S_{i-2,v} +\!\!+ S_{i-1,v} +\!\!+ S_{i,v} +\!\!+ S_{i+1,v} \quad (1)$$

where $+\!\!+$ represents concatenation.

Note that after the concatenation, the resulting dimensionality of the feature vector for ***FC(3, 1)*** is $18 \times (3 + 1 + 1) = 90$.

Line No.	Model	Overall	*clean*	*clean-hard*	*part-of*	*context-dep*
1	Baseline	0.429	0.661	0.365	0.161	0.375
	Source-Side Experiments: Co-reference Resolution, Local Context					
2	***FC***(3, 1)	0.513	0.690	**0.485**	0.305	0.448
3	src-coref	0.479	0.667	0.408	0.286	0.429
4	src-coref + ***FC***(3, 1)	**0.532**	0.695	0.452	**0.385**	**0.485**
	Target-Side Experiments: Co-reference Resolution, Global Context					
5	*Transformer-XH*	0.468	0.680	0.441	0.226	0.384
6	tgt-coref	0.443	0.673	0.422	0.182	0.339
7	tgt-coref + *Transformer-XH*	0.458	0.702	0.444	0.161	0.357
	Source+Target-Side Experiments: Co-reference Resolution, Local Context, Global Context					
8	src-coref + tgt-coref	0.487	0.672	0.440	0.291	0.411
9	All	0.517	**0.749**	0.389	0.321	0.464

Table 4: MAP scores of the reranker models on the test set using the *Debate-Level Chrono* split.

4.2.3 Global Context

The similarity scores that leverage the local context in the textual content of the *InputClaim* and the *VerClaim* are obtained using (*i*) BM25, and (*ii*) the cosine similarity between the Sentence-BERT embeddings of the *InputClaim* vs. the top-4 sentences of the *VerClaim*. This might miss relevant information further away from the *InputClaim* in the input document and further away from the *VerClaim* in the document accompanying the *VerClaim*. We refer to such scattered information as the **global context**. To capture it, we use Transformer-XH (Zhao et al., 2019), which is pretrained on the FEVER (Fact Extraction and VERification) dataset to predict whether a given input claim is supported/refuted by a set of target sentences (from Wikipedia), represented as a graph, or there is no enough information. We used the model from (Zhao et al., 2019). For a given *InputClaim*, we generate a graph for each of the top-100 *VerClaim*s retrieved using BM25 and the normalized claim, the *title*, and the top-3 sentences from the *text* as nodes. Using the ***Transformer-XH*** model on the graph, we obtain three additional scores that correspond to the posterior probability that *VerClaim* supports or refutes the *InputClaim*, or there is no enough information.

4.3 Hyper-Parameter Values

For the baseline, we use the best values of the hyper-parameters as found in (Shaar et al., 2020a). For our context-aware models, we select the values of the hyper-parameters by splitting the training dataset into train-train (debates from 2012-2017) and train-dev (debates from 2018), then we train on the former, and we test on the latter.

4.4 Evaluation Measures

As we have a ranking task, we use mean average precision (MAP) for evaluation. It is a suitable measure as some *InputClaim*s are paired with more than one *VerClaim*. This is why we opted for not using mean reciprocal rank (MRR), which would only pay attention to the highest-ranked match.

5 Results

Below, we described the results for our source-side and target-side context modeling experiments.

5.1 Source-Side Experiments

For the source side experiments, we used co-reference resolution on transcripts and variations of the local context by varying k and l in Eq. 1.

When we inspected the transcripts, we found that co-references tended to be resolved by a few sentences before the *InputClaim*; therefore, we tried ***FC***(1, 1), ***FC***(3, 1), ***FC***(3, 3), and ***FC***(5, 1). We obtained the best results on cross-validation using ***FC***(3, 1), which we use below. As shown in Table 4, the local context (line 2) improves over the baseline (line 1) by eight MAP points absolute.

We experimented using co-reference resolution with **NeuralCoref**. This yielded a sizable improvement over the baseline as shown in line 3 in Table 4, especially for *part-of* and *context-dep* pairs, as they have many co-references, which can make it hard for the model the understand the *InputClaim*. After combining the two methods, i.e., *src-coref* and ***FC***(3,1) (see line 4), we achieved the highest MAP score of 0.532. We always see an improvement for the *clean* category as the resolved *InputClaim* can match the article text better.

5.2 Target-Side Experiments

For the target-side experiments, we tried using co-reference resolution (on the source and on the target side) for the *VerClaim* and the fact-checking article, as well as modeling the global context with *Transformer-XH*. Compared to the baseline, we see on line 5 of Table 4 a sizable improvement from 0.365 to 0.441 MAP points for *clean-hard*.

This is expected as the pair does not exhibit much semantic similarity, and we need to build our own understanding of the *text* of the *VerClaim* in order to capture the contextual similarity in the pair. We also experimented with co-reference resolution on the *VerClaim* and the *text* of the *VerClaim* and also see some improvement. Combining *tgt-coref* and *Transformer-XH* (line 7) improved the performance over *tgt-coref* alone, but it is worse than *Transformer-XH* alone. The combination outperforms other target-side experiments for *clean*.

5.3 Source-Side & Target-Side Experiments

Eventually, we experimented with modeling the context both on the source and on the target side. Line 8 in Table 4 shows the evaluation results when we use co-reference resolution both on the source and on the target side. We can see that this yields a higher overall MAP score of 0.487, compared to using source-side (MAP of 0.479; line 3) or target-side context only (MAP of 0.443; line 6). Moreover, co-reference resolution on both sides helps for *clean-hard* and *part-of* (compared to using co-reference on one side only) as they require better local and global context, respectively.

We further tried putting it all together, and the result is shown in line 9.[13] While this yielded better results for *clean*, it was slightly worse compared to the source-side context modeling combination in line 4. This is probably due to the source-side context models being generally stronger than the target-side ones (compare lines 2–3 to lines 5–6).

We can conclude that modeling the context on the source side is much more important than on the target side. This is expected for political debates, which are conversational in nature. In contrast, the target side is a well-written journalistic article, where sentences are much more self-contained. Thus, features from the source side (i.e., from the debate) are more useful as can be seen in Table 4.

[13]Note that in this result we did not use target-side co-reference, as adding it yielded somewhat worse results. It seems to interact badly with Transformer-XH, which can also be seen by comparing lines 5 and 7.

5.4 Discussion

As mentioned above, our baseline is a reimplementation of the best system of Shaar et al. (2020a), and our context modeling extensions add additional components on top of it. Note, however, that our experimental results are not directly comparable to their published ones, as we use a more realistic and also a much harder setup, where the data is split by entire debates and also chronologically, following the *Debate-Level Chrono* data split, as we discussed in Section 3, i.e., training on the data from 2012 to 2018 and testing on 2019 (while they split all debates into sentences and randomly distribute them to training/testing). However, we do have comparison to their approach, as we ran their model on our data split, which is our baseline, as shown on line 1 of Table 4.

6 Conclusion and Future Work

We have presented our work on the important but under-studied problem of detecting previously fact-checked claims in political debates and speeches. We studied the impact of modeling the context: both on the source side, i.e., in the debate, as well as on the target side, i.e., in the fact-checking document that explains how human fact-checkers have arrived at their decision about the factuality of the claim. In particular, we modeled the local context, the global context, and we further used co-reference resolution and multi-hop reasoning over the target text using Transformer-XH. The experimental results have shown that each of these components represents a valuable information source, but modeling the source-side context is more important, and can yield 10+ points of absolute improvement over a context-free state-of-the-art baseline.

In future work, we want to try other multi-hop reasoning frameworks for context modeling. We also plan to experiment with other kinds of conversations, e.g., in community forums and in social media, including for other languages.

Acknowledgments

This research is part of the Tanbih mega-project, developed at the Qatar Computing Research Institute, HBKU, which aims to limit the impact of "fake news," propaganda, and media bias by making users aware of what they are reading, thus promoting media literacy and critical thinking.

Ethics and Broader Impact

Biases We note that there might be some biases in the data we use, as well as in some manual judgments for claim matching. There could be also biases in the data selection and the fact-checking process of the human fact-checkers, which are beyond our control. Finally, there are known biases in the large-scale pre-trained transformer models that we experiment with.

Intended Use and Misuse Potential Our models can make it possible to put politicians on the spot in real time, e.g., during an interview or a political debate, by providing journalists with tools to do trustable fact-checking in real time. They can also save a lot of time to fact-checkers for unnecessary double-checking something that was already fact-checked. However, these models could also be misused by malicious actors. We, therefore, ask researchers to exercise caution.

Environmental Impact We would also like to warn that the use of large-scale Transformers requires a lot of computations and the use of GPUs/TPUs for training, which contributes to global warming (Strubell et al., 2019). This is a bit less of an issue in our case, as we do not train such models from scratch; rather, we fine-tune them on relatively small datasets. Moreover, running on a CPU for inference, once the model has been fine-tuned, is perfectly feasible, and CPUs contribute much less to global warming.

References

Pepa Atanasova, Lluís Màrquez, Alberto Barrón-Cedeño, Tamer Elsayed, Reem Suwaileh, Wajdi Zaghouani, Spas Kyuchukov, Giovanni Da San Martino, and Preslav Nakov. 2018. Overview of the CLEF-2018 CheckThat! lab on automatic identification and verification of political claims, task 1: Check-worthiness. In *CLEF 2018 Working Notes. Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum*, CEUR Workshop Proceedings, Avignon, France.

Pepa Atanasova, Preslav Nakov, Georgi Karadzhov, Mitra Mohtarami, and Giovanni Da San Martino. 2019a. Overview of the CLEF-2019 CheckThat! Lab on Automatic Identification and Verification of Claims. Task 1: Check-Worthiness. In *CLEF 2019 Working Notes*, Lugano, Switzerland.

Pepa Atanasova, Preslav Nakov, Lluís Màrquez, Alberto Barrón-Cedeño, Georgi Karadzhov, Tsvetomila Mihaylova, Mitra Mohtarami, and James Glass. 2019b. Automatic fact-checking using context and discourse information. *Journal of Data and Information Quality (JDIQ)*, 11(3):1–27.

Isabelle Augenstein, Christina Lioma, Dongsheng Wang, Lucas Chaves Lima, Casper Hansen, Christian Hansen, and Jakob Grue Simonsen. 2019. MultiFC: A real-world multi-domain dataset for evidence-based fact checking of claims. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, EMNLP-IJCNLP '19', pages 4685–4697, Hong Kong, China.

Alberto Barrón-Cedeño, Tamer Elsayed, Preslav Nakov, Giovanni Da San Martino, Maram Hasanain, Reem Suwaileh, Fatima Haouari, Nikolay Babulkov, Bayan Hamdan, Alex Nikolov, Shaden Shaar, and Zien Sheikh Ali. 2020. Overview of CheckThat! 2020: Automatic identification and verification of claims in social media. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction Proceedings of the Eleventh International Conference of the CLEF Association (CLEF 2020)*, LNCS (12260).

Alberto Barrón-Cedeño, Tamer Elsayed, Preslav Nakov, Giovanni Da San Martino, Maram Hasanain, Reem Suwaileh, and Fatima Haouari. 2020. CheckThat! at CLEF 2020: Enabling the automatic identification and verification of claims in social media. In *Proceedings of the 42nd European Conference on Information Retrieval*, ECIR '20, pages 499–507, Lisbon, Portugal.

Alberto Barrón-Cedeño, Tamer Elsayed, Reem Suwaileh, Lluis Marquez, Pepa Atanasova, Wajdi Zaghouani, Spas Kyuchukov, Giovanni Da San Martino, and Preslav Nakov. 2018. Overview of the CLEF-2018 CheckThat! lab on automatic identification and verification of political claims. Task 2: Factuality. In *CLEF 2018 Working Notes. Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum*, CEUR Workshop Proceedings, Avignon, France.

Mostafa Bouziane, Hugo Perrin, Aurèlien Cluzeau, Julien Mardas, and Amine Sadeq. 2020. Buster.AI at CheckThat! 2020: Insights and recommendations to improve fact-checking. In *Working Notes of CLEF 2020—Conference and Labs of the Evaluation Forum*, CLEF '2020, Thessaloniki, Greece.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, ACL '17, pages 1657–1668, Vancouver, Canada.

Anton Chernyavskiy, Dmitry Ilvovsky, Pavel Kalinin, and Preslav Nakov. 2022. Batch-softmax contrastive loss for pairwise sentence scoring tasks. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT '22, Seattle, Washington, USA.

Anton Chernyavskiy, Dmitry Ilvovsky, and Preslav Nakov. 2021a. Aschern at CLEF CheckThat! 2021: Lambda-calculus of fact-checked claims. In *CLEF 2021 Working Notes. Working Notes of CLEF 2021–Conference and Labs of the Evaluation Forum*, Bucharest, Romania (online).

Anton Chernyavskiy, Dmitry Ilvovsky, and Preslav Nakov. 2021b. WhatTheWikiFact: Fact-checking claims against Wikipedia. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*, CIKM '21, pages 4690–4695.

Pepa Gencheva, Preslav Nakov, Lluís Màrquez, Alberto Barrón-Cedeño, and Ivan Koychev. 2017. 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 '17, pages 267–276, Varna, Bulgaria.

Andreas Hanselowski, Hao Zhang, Zile Li, Daniil Sorokin, Benjamin Schiller, Claudia Schulz, and Iryna Gurevych. 2018. UKP-Athene: Multi-sentence textual entailment for claim verification. In *Proceedings of the First Workshop on Fact Extraction and VERification*, FEVER '18, pages 103–108, Brussels, Belgium.

Maram Hasanain, Fatima Haouari, Reem Suwaileh, Zien Sheikh Ali, Bayan Hamdan, Tamer Elsayed, Alberto Barrón-Cedeño, Giovanni Da San Martino, and Preslav Nakov. 2020. Overview of CheckThat! 2020 Arabic: Automatic identification and verification of claims in social media. In *Working Notes of CLEF 2020—Conference and Labs of the Evaluation Forum*, CLEF '2020, Thessaloniki, Greece.

Naeemul Hassan, Gensheng Zhang, Fatma Arslan, Josue Caraballo, Damian Jimenez, Siddhant Gawsane, Shohedul Hasan, Minumol Joseph, Aaditya Kulkarni, Anil Kumar Nayak, Vikas Sable, Chengkai Li, and Mark Tremayne. 2017. ClaimBuster: The first-ever end-to-end fact-checking system. *Proc. VLDB Endow.*, 10(12):1945–1948.

Ralf Herbrich, Thore Graepel, and Klaus Obermayer. 1999. Support vector learning for ordinal regression. In *Proceedings of the 1999 Ninth International Conference on Artificial Neural Networks*, ICANN '99, pages 97–102, Edinburgh, UK.

Christopher Hidey and Mona Diab. 2018. Team SWEEPer: Joint sentence extraction and fact checking with pointer networks. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 150–155, Brussels, Belgium.

David M.J. Lazer, Matthew A. Baum, Yochai Benkler, Adam J. Berinsky, Kelly M. Greenhill, Filippo Menczer, Miriam J. Metzger, Brendan Nyhan, Gordon Pennycook, David Rothschild, Michael Schudson, Steven A. Sloman, Cass R. Sunstein, Emily A. Thorson, Duncan J. Watts, and Jonathan L. Zittrain. 2018. The science of fake news. *Science*, 359(6380):1094–1096.

Yaliang Li, Jing Gao, Chuishi Meng, Qi Li, Lu Su, Bo Zhao, Wei Fan, and Jiawei Han. 2016. A survey on truth discovery. *ACM SIGKDD Explorations Newsletter*, 17(2):1–16.

Zhenghao Liu, Chenyan Xiong, Maosong Sun, and Zhiyuan Liu. 2020. Fine-grained fact verification with kernel graph attention network. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7342–7351, Online.

Jackson Luken, Nanjiang Jiang, and Marie-Catherine de Marneffe. 2018. QED: A fact verification system for the FEVER shared task. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 156–160, Brussels, Belgium.

Simona Mihaylova, Iva Borisova, Dzhovani Chemishanov, Preslav Hadzhitsanev, Momchil Hardalov, and Preslav Nakov. 2021. DIPS at CheckThat! 2021: Verified claim retrieval. In *CLEF 2021 Working Notes. Working Notes of CLEF 2021–Conference and Labs of the Evaluation Forum*, Bucharest, Romania (online).

Tsvetomila Mihaylova, Georgi Karadzhov, Pepa Atanasova, Ramy Baly, Mitra Mohtarami, and Preslav Nakov. 2019. SemEval-2019 task 8: Fact checking in community question answering forums. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 860–869, Minneapolis, Minnesota, USA.

Preslav Nakov, Alberto Barrón-Cedeño, Giovanni Da San Martino, Firoj Alam, Rubén Míguez, Tommaso Caselli, Mucahid Kutlu, Wajdi Zaghouani, Chengkai Li, Shaden Shaar, Gautam Kishore Shahi, Hamdy Mubarak, Alex Nikolov, Yavuz Selim Kartal, and Javier Beltrán. 2022a. Overview of the CLEF-2022 CheckThat! lab task 1 on identifying relevant claims in tweets. In *Working Notes of CLEF 2022—Conference and Labs of the Evaluation Forum*, CLEF '2022, Bologna, Italy.

Preslav Nakov, Alberto Barrón-Cedeño, Giovanni Da San Martino, Firoj Alam, Julia Maria Struß, Thomas Mandl, Rubén Míguez, Tommaso Caselli, Mucahid Kutlu, Wajdi Zaghouani, Chengkai Li, Shaden Shaar, Gautam Kishore Shahi, Hamdy Mubarak, Alex Nikolov, Nikolay Babulkov, Yavuz Selim Kartal, and Javier Beltrán. 2022b. The CLEF-2022 CheckThat! lab on fighting the COVID-19 infodemic and fake news detection. In *Proceedings of the 44th European Conference on IR Research: Advances in Information Retrieval*, ECIR '22, pages 416–428, Stavanger, Norway.

Preslav Nakov, Alberto Barrón-Cedeño, Giovanni Da San Martino, Firoj Alam, Julia Maria Struß, Thomas Mandl, Rubén Míguez, Tommaso Caselli, Mucahid Kutlu, Wajdi Zaghouani,

Chengkai Li, Shaden Shaar, Gautam Kishore Shahi, Hamdy Mubarak, Alex Nikolov, Nikolay Babulkov, Yavuz Selim Kartal, and Javier Beltrán. 2022c. Overview of the CLEF-2022 CheckThat! lab on fighting the COVID-19 infodemic and fake news detection. In *Proceedings of the 13th International Conference of the CLEF Association: Information Access Evaluation meets Multilinguality, Multimodality, and Visualization*, CLEF '2022, Bologna, Italy.

Preslav Nakov, Alberto Barrón-Cedeño, Tamer Elsayed, Reem Suwaileh, Lluís Màrquez, Wajdi Zaghouani, Pepa Gencheva, Spas Kyuchukov, and Giovanni Da San Martino. 2018. 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.

Preslav Nakov, David Corney, Maram Hasanain, Firoj Alam, Tamer Elsayed, Alberto Barrón-Cedeño, Paolo Papotti, Shaden Shaar, and Giovanni Da San Martino. 2021a. Automated fact-checking for assisting human fact-checkers. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence*, IJCAI '21, pages 4551–4558.

Preslav Nakov, Giovanni Da San Martino, Firoj Alam, Shaden Shaar, Hamdy Mubarak, and Nikolay Babulkov. 2022d. Overview of the CLEF-2022 CheckThat! lab task 2 on detecting previously fact-checked claims. In *Working Notes of CLEF 2022—Conference and Labs of the Evaluation Forum*, CLEF '2022, Bologna, Italy.

Preslav Nakov, Giovanni Da San Martino, Tamer Elsayed, Alberto Barrón-Cedeno, Rubén Míguez, Shaden Shaar, Firoj Alam, Fatima Haouari, Maram Hasanain, Nikolay Babulkov, et al. 2021b. The CLEF-2021 CheckThat! lab on detecting check-worthy claims, previously fact-checked claims, and fake news. In *Proceedings of the European Conference on Information Retrieval*, ECIR '21, pages 639–649, Lucca, Italy. Springer.

Preslav Nakov, Da San Martino Giovanni, Tamer Elsayed, Alberto Barrón-Cedeño, Rubén Míguez, Shaden Shaar, Firoj Alam, Fatima Haouari, Maram Hasanain, Watheq Mansour, Bayan Hamdan, Zien Sheikh Ali, Nikolay Babulkov, Alex Nikolov, Gautam Kishore Shahi, Julia Maria Struß, Thomas Mandl, Mucahid Kutlu, and Yavuz Selim Kartal. 2021c. Overview of the CLEF-2021 CheckThat! lab on detecting check-worthy claims, previously fact-checked claims, and fake news. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Twelfth International Conference of the CLEF Association*, LNCS (12880).

Van-Hoang Nguyen, Kazunari Sugiyama, Preslav Nakov, and Min-Yen Kan. 2020. FANG: Leveraging social context for fake news detection using graph representation. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, CIKM '20, pages 1165–1174.

Yixin Nie, Haonan Chen, and Mohit Bansal. 2019. Combining fact extraction and verification with neural semantic matching networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33 of *AAAI '19*, pages 6859–6866, Honolulu, Hawaii, USA.

Wojciech Ostrowski, Arnav Arora, Pepa Atanasova, and Isabelle Augenstein. 2021. Multi-hop fact checking of political claims. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, IJCAI '21, pages 3892–3898, Motreal, Canada.

Albert Pritzkau. 2021. NLytics at CheckThat! 2021: Check-worthiness estimation as a regression problem on transformers. In *CLEF 2021 Working Notes. Working Notes of CLEF 2021–Conference and Labs of the Evaluation Forum*, Bucharest, Romania (online).

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China.

Shaden Shaar, Firoj Alam, Giovanni Da San Martino, Alex Nikolov, Wajdi Zaghouani, Preslav Nakov, and Anna Feldman. 2021a. Findings of the NLP4IF-2021 shared tasks on fighting the COVID-19 infodemic and censorship detection. In *Proceedings of the Fourth Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda*, pages 82–92, Online.

Shaden Shaar, Nikolay Babulkov, Giovanni Da San Martino, and Preslav Nakov. 2020a. That is a known lie: Detecting previously fact-checked claims. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3607–3618, Online.

Shaden Shaar, Fatima Haouari, Watheq Mansour, Maram Hasanain, Nikolay Babulkov, Firoj Alam, Giovanni Da San Martino, Tamer Elsayed, and Preslav Nakov. 2021b. Overview of the CLEF-2021 CheckThat! lab task 2 on detecting previously fact-checked claims in tweets and political debates. In *Working Notes of CLEF 2021—Conference and Labs of the Evaluation Forum*, CLEF '2021, Bucharest, Romania (online).

Shaden Shaar, Maram Hasanain, Bayan Hamdan, Zien Sheikh Ali, Fatima Haouari, Alex Nikolov, Mucahid Kutlu, Yavuz Selim Kartal, Firoj Alam, Giovanni Da San Martino, Alberto Barrón-Cedeño, Rubén Míguez, Tamer Elsayed, and Preslav Nakov. 2021c. 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).

Shaden Shaar, Alex Nikolov, Nikolay Babulkov, Firoj Alam, Alberto Barrón-Cedeño, Tamer Elsayed, Maram Hasanain, Reem Suwaileh, Fatima Haouari, Giovanni Da San Martino, and Preslav Nakov. 2020b. Overview of CheckThat! 2020 English: Automatic identification and verification of claims in social media. In *Working Notes of CLEF 2020—Conference and Labs of the Evaluation Forum*, CLEF '2020, Thessaloniki, Greece.

Qiang Sheng, Juan Cao, Xueyao Zhang, Xirong Li, and Lei Zhong. 2021. Article reranking by memory-enhanced key sentence matching for detecting previously fact-checked claims. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, ACL-IJCNLP '21, pages 5468–5481, Online.

Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 19(1):22–36.

Mark Stencel. 2019. Number of fact-checking outlets surges to 188 in more than 60 countries. *Duke Reporters' LAB*, pages 12–17.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650, Florence, Italy.

James Thorne and Andreas Vlachos. 2018. Automated fact checking: Task formulations, methods and future directions. In *Proceedings of the 27th International Conference on Computational Linguistics*, COLING '18, pages 3346–3359, Santa Fe, New Mexico, USA.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL '18, pages 809–819, New Orleans, Louisiana.

Slavena Vasileva, Pepa Atanasova, Lluís Màrquez, Alberto Barrón-Cedeño, and Preslav Nakov. 2019. 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*, RANLP '19, pages 1229–1239, Varna, Bulgaria.

Andreas Vlachos and Sebastian Riedel. 2014. Fact checking: Task definition and dataset construction. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, pages 18–22, Baltimore, Maryland, USA.

Nguyen Vo and Kyumin Lee. 2018. The rise of guardians: Fact-checking URL recommendation to combat fake news. In *Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, pages 275–284, Ann Arbor, Michigan, USA.

Nguyen Vo and Kyumin Lee. 2020. Where are the facts? Searching for fact-checked information to alleviate the spread of fake news. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, EMNLP '20, pages 7717–7731, Online.

Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018a. The spread of true and false news online. *Science*, 359(6380):1146–1151.

Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018b. The spread of true and false news online. *Science*, 359(6380):1146–1151.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online.

William Yang Wang. 2017. "Liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, ACL '17, pages 422–426, Vancouver, Canada.

Takuma Yoneda, Jeff Mitchell, Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. UCL machine reading group: Four factor framework for fact finding (HexaF). In *Proceedings of the First Workshop on Fact Extraction and VERification*, FEVER '18, pages 97–102, Brussels, Belgium.

Tauhid Zaman, Emily B Fox, Eric T Bradlow, et al. 2014. A bayesian approach for predicting the popularity of tweets. *Annals of Applied Statistics*, 8(3):1583–1611.

Chen Zhao, Chenyan Xiong, Corby Rosset, Xia Song, Paul Bennett, and Saurabh Tiwary. 2019. Transformer-XH: Multi-evidence reasoning with extra hop attention. In *Proceedings of the International Conference on Learning Representations*, ICLR '19, New Orleans, Louisiana, USA.

Wanjun Zhong, Jingjing Xu, Duyu Tang, Zenan Xu, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. 2020. Reasoning over semantic-level graph for fact checking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, ACL '20, pages 6170–6180, Online.

Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2019. GEAR: Graph-based evidence aggregating and reasoning for fact verification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, ACL '19, pages 892–901, Florence, Italy.