WatClaimCheck: A new Dataset for Claim Entailment and Inference

Kashif Khan
Computer Science
University of Waterloo
Ontario, Canada
k2@iwrk.com

Ruizhe Wang
Computer Science
University of Waterloo
Ontario, Canada
r322wang@uwaterloo.ca

Pascal Poupart
Computer Science
University of Waterloo
Vector Institute
Ontario, Canada
ppoupart@uwaterloo.ca

Abstract

We contribute a new dataset[1] for the task of automated fact checking and an evaluation of state of the art algorithms. The dataset includes claims (from speeches, interviews, social media and news articles), review articles published by professional fact checkers and premise articles used by those professional fact checkers to support their review and verify the veracity of the claims. An important challenge in the use of premise articles is the identification of relevant passages that will help to infer the veracity of a claim. We show that transferring a dense passage retrieval model trained with review articles improves the retrieval quality of passages in premise articles. We report results for the prediction of claim veracity by inference from premise articles.

1 Introduction

The rise of social media has led to a democratization of news, but it has also amplified issues related to fake news and misinformation. To that effect, many fact checking organizations (e.g., Politifact, Snopes, AFP Fact Check, Alt News, FactCheck.org, Africa Check, etc.) have emerged around the globe. They investigate debatable claims made by authorities, politicians, celebrities and the public. For each claim, they publish a review article with links to sources that support a verdict (e.g., true, partly true/false, false) about the veracity of the claim. Those reviews debunk false claims and mitigate the spread of misinformation. We consider a key NLP challenge in the context of automated fact checking: claim inference from premise articles. Note that determining the veracity of a claim without additional information is nearly impossible since claims are selected by professional fact checkers in part because their veracity is far from obvious and also because of their degree of controversy. To that effect, professional fact checkers invest a fair amount of time to research each claim by finding relevant sources and publishing a review article that explains their verdict of the claim. Hence there is a natural entailment problem, whereby anyone who reads a review article should be able to arrive at the same verdict as the professional fact checker regarding the claim. Unlike many entailment tasks that consist of short text (e.g., pairs of utterances) that may be artificially generated or extracted, this is a natural and challenging entailment task that involves an entire document (review article) with an utterance (claim) that requires a certain degree of reading comprehension. We note that this entailment problem has been tackled in some previous work (Augenstein et al., 2019; Shu et al., 2018; Nakov et al., 2021) and although it is a challenging NLP problem, it does not correspond to the problem that professional fact checkers need to solve.

In this paper, we focus on the harder problem of claim inference from premise articles. This is part of the challenge that professional fact checkers face. They find premise articles that contain relevant facts and then infer the veracity of the claim based on those facts. Unlike many existing inference tasks where it is sufficient to use one or a few facts in a few sentences (Storks et al., 2019; Schlegel et al., 2020), information from a set of premise articles must be distilled and combined in non trivial ways to infer the veracity of a claim.

We assembled a dataset of 33,697 claims made between December 1996 and July 2021 with associated review articles, premise articles and claim verdicts. Many other datasets for claim verification are listed in Table 1. However, most of them do not include premise articles needed for the inference task described above. We note two exceptions: PubHealth (Kotonya and Toni, 2020b), which is restricted to health claims and UKP

[1]Code and form to request the dataset are available at https://github.com/nxii/WatClaimCheck. The WatClaimCheck dataset is available upon request for non-commercial research purposes only under the fair dealing exception of the Canada Copyright Act.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 1293 - 1304

Snopes (Hanselowski et al., 2019), which is restricted to claims investigated by one fact checking organization (Snopes). In contrast, WatClaimCheck includes claims investigated by 8 fact checking organizations on any topic.

Since there are several premise articles for a given claim and each premise article may be long, a simple two-stage approach to identify relevant passages would consist of a lightweight retrieval technique in a first stage, followed by a heavyweight inference technique applied to those passages. When the first stage fails to retrieve key passages, then the inferred verdict will be negatively affected regardless of how good the second stage is. To that effect, several supervised dense passage retrieval techniques have been proposed for question-answering (Karpukhin et al., 2020; Qu et al., 2021; Ren et al., 2021). Unfortunately, we cannot directly apply those techniques since we do not have labels for the relevant passages. Instead, we show how to use the review articles to train a supervised dense retrieval technique that is then transferred to premise articles. The contributions of the paper can be summarized as follows:

- New dataset of claims with review and premise articles for claim inference in automated fact checking;
- Novel use of review articles to transfer a dense retrieval technique to premise articles;
- Experiments establishing the state of the art for claim inference.

The paper is organized as follows. Sect. 2 reviews previous work related to automated fact checking and claim verification. Sect. 3 describes the new dataset and summarizes the differences with previous datasets for claim verification. Sect. 4 describes a two-stage process to i) extract evidence sentences from premise articles and ii) infer the veracity of claims. This section also explains how to transfer a dense passage retrieval technique trained with review articles to premise articles. Sect. 5 reports the results for the claim veracity inference task. Finally, Sect. 6 concludes and discusses possible future work.

2 Related Work

There is an important line of work that focuses on claim verification (Kotonya and Toni, 2020a; Guo et al., 2022). This includes techniques that predict the veracity of a claim based on the text of the claim only (Rashkin et al., 2017), linguistic features (Popat et al., 2017), meta information about the claimant (e.g., name, job, party affiliation, veracity history) (Wang, 2017), review articles (Augenstein et al., 2019; Shu et al., 2018; Nakov et al., 2021), relevant articles returned by a search engine (Popat et al., 2018; Augenstein et al., 2019; Mishra and Setty, 2019) as well as premise articles (Aly et al., 2021; Kotonya and Toni, 2020b). There is an important distinction between articles returned by a search engine and premise articles. The techniques that use a search engine to find articles related to a claim query the search engine *after* a fact checking website has published a review article and therefore end up retrieving articles that include the review article as well as other articles that summarize and/or discuss the verdict of the fact checking website. Hence they are tackling an entailment problem. In contrast, the premise articles that we consider are the source articles used by a fact checker *before* publishing a review article. Those articles contain relevant facts, but not a summary or discussion of the review article since they are published before the review article and in fact serve as premises for the review article.

Closely related to claim verification is the problem of fake news detection. In this problem, the credibility of an entire news article is evaluated. The credibility can be estimated based on linguistic and textual features (Conroy et al., 2015; Reis et al., 2019; Li et al., 2019), discourse level structure (Karimi and Tang, 2019), network analysis (Conroy et al., 2015), knowledge graphs (Cui et al., 2020), inter-user behaviour dynamics (Gangireddy et al., 2020) or a combination of multiple modalities (Wang et al., 2020). Some techniques reorder the articles returned by a search engine based on their degree of credibility (Olteanu et al., 2013; Beyluniogl, 2020). An important task that can help the detection of fake news is stance detection (Borges et al., 2019; Jwa et al., 2019), i.e., does the content of an article agree or disagree with the title of the article? The following surveys summarize advances in fake news detection: (Kumar and Shah, 2018; Bondielli and Marcelloni, 2019).

3 Fact Checking Dataset

3.1 Data Collection

We collect claims, along with a review article, premise articles, and metadata from the follow-

ing eight fact checking services: Politifact, Snopes, AFP Fact Check, Alt News, FactCheck.org, Africa Check, USA Today, and Full Fact. We utilize Google's fact check tool APIs[2] to collect the claims' metadata for all fact checking services except Politifact and Snopes. The claims' metadata collected from Google's fact check tool APIs include the claim review article URL, which is used to retrieve the claim review article. The claim review articles published by some of the fact checking services provide the premise article URLs in a separate section while others provide the URLs as inline links in the review article body. We parse the article body, retrieving the premise article URLs used in the review article to justify the claim veracity. Finally, the premise URLs are used to retrieve the premise articles. We try to directly retrieve the article where possible, but also use archive.org's API in case a premise article is no longer available online. We follow the same general procedure for data collection from Politifact and Snopes except that we directly crawl the respective websites instead of using Google's fact check tool APIs for collecting claims and associated metadata.

We perform some basic cleanup to the collected data before inclusion in the dataset. This includes removing articles behind paywalls, removing claims with less than two premise articles, and removing non-textual premise sources. We obtain premise article text from their HTML pages by loading the HTML files into a text based web browser (Links browser) and then dumping the web page text into a text file. This allows us to bypass the CSS styling and JavaScript code included in the HTML pages and obtain only the text displayed to end users. Admittedly, this does not eliminate auxiliary text such as navigation links, footer text, recommended links, etc. The premise articles include the source document of the claim when available as well as evidence articles used by fact checkers. We map the numerous claim veracity labels used by the fact checking websites into three broad labels: True, Partially True/False, and False.

3.2 WatClaimCheck Dataset

The contributed dataset contains a total of 33,721 claims. We split those claims into the following three sets: training set containing 26,976 claims, validation set containing 3,372 claims, and test set containing 3,373 claims. For each claim in the dataset, we provide the following data: *ID, Claimant, Claim, Claim Date, Reviewer Name, Reviewer Site, Review Article URL, Review Article Date, Review Article, Rating, Original Rating, Premise Articles* and *Premise Article URLs*. Here *Original Rating* refers to the rating assigned by a fact checking organization and *Rating* corresponds to our mapping of the original rating to true, partly true/false and false (see the dataset for the precise mapping). We provide the extracted text files for the review and premise articles.

[2]https://toolbox.google.com/factcheck/apis

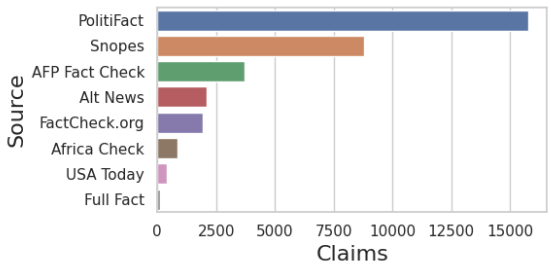

Figure 1: Number of claims from each source

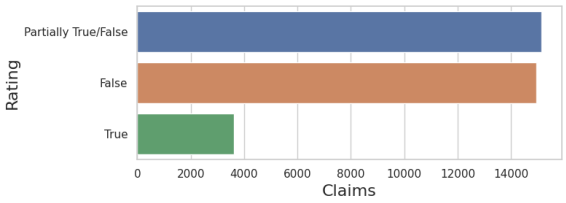

Figure 2: Dataset Claim Rating Counts

Fig. 1 shows the number of claims per fact checking services. Fig. 2 shows the claim rating distribution. Claims in the Partially True/False and False categories significantly outnumber the claims in the True category. In reality, the number of true claims is much larger than the number of partially true/false and false claims, but fact checking services focus on debunking controversial claims and therefore the majority of the claims they investigate are false or partially true/false. This imbalance poses an important challenge.

3.3 Comparison with existing Datasets

We compare our proposed dataset with other publicly available fact checking related datasets in Table 1. We can broadly classify the fact checking datasets into two different categories: (1) veracity detection datasets based only on claim text and some metadata, but without supporting evidence documents and (2) datasets that provide claim text along with supporting evidence and/or context doc-

uments. The datasets that provide some evidence or context documents can be further subcategorized: (1) datasets that provide social media posts and comments related to the claim (Mitra and Gilbert, 2015; Nakamura et al., 2020; Shu et al., 2018), (2) datasets that retrieve supporting evidence for the claims by performing a web search using queries obtained from lexical and semantic features of the claim text (Baly et al., 2018; Augenstein et al., 2019; Gupta and Srikumar, 2021), (3) datasets that provide Wikipedia pages as supporting evidence (Thorne et al., 2018; Fan et al., 2020; Aly et al., 2021), and (4) datasets that include premise articles used by professional fact checkers (Hanselowski et al., 2019; Kotonya and Toni, 2020b). Our proposed dataset provides the documents cited by the professional fact checkers in the claim review article to justify their claim rating. This reflects the real world task of automated veracity detection more truthfully due to the availability of the premise articles cited by the professional fact checkers in claim review articles. Although, social media posts and comments can sometimes be helpful in claim veracity detection they are rarely treated as authoritative sources of information. Using a web search to retrieve evidence documents *after* a fact checking service has verified a claim is problematic since multiple news agencies often publish articles referencing the original fact checking review article. Top-k web search results typically contain those articles which may indirectly leak the veracity label.

4 Models

We develop a two-stage system to perform evidence based veracity detection. The first stage selects relevant sentence level evidence from the premise articles associated with a claim and the second stage performs claim veracity inference using the claim text and selected evidence sentences. For the first stage, we evaluate two different approaches. The first approach is term frequency inverse document frequency (TF-IDF), which is typically used by fact checking methods for sentence based retrieval (Aly et al., 2021). For the second approach, we propose a novel way to adapt dense passage retrieval techniques using the review articles for evidence sentence selection. In our experiments, the aforementioned dense passage retrieval technique outperforms TF-IDF text retrieval and leads to overall system performance improvements. The second stage consists of training deep learning models to perform claim veracity inference using the claim text and selected evidence. We utilize multiple deep learning models to perform claim veracity inference ranging from basic bi-directional recurrent networks to state of the art transformers.

4.1 Problem Formulation

We represent a claim containing l tokens as $C_n = \{c_1, c_2, \ldots, c_l\}$, where $n \in [1, N]$ and N is the size of the dataset. Each claim is associated with multiple premise articles, we represent the k-th premise article associated with the n-th claim containing m sentences as $A_{n,k} = \{s^P_{n,1}, s^P_{n,2}, \ldots, s^P_{n,m}\}$ where $s_{n,i}$ represents the i-th sentence. Similarly, we represent the review article associated with the claim C_n containing m sentences by $R_n = \{s^R_{n,1}, s^R_{n,2}, \ldots, s^R_{n,m}\}$. For a given claim C_n, we represent its ground truth veracity label by $\mathbf{y_n}$. We cast the problem as a textual inference problem. Given a claim C_n and a set of associated premise articles $\mathbf{A}$, our goal is to predict the ground truth veracity $\mathbf{y_n}$ of the claim.

4.2 Stage-1: Evidence Sentence Extraction

A key step performed by professional fact checkers is examining the premise articles associated with a claim and extracting useful evidence from them to establish claim veracity. Our first stage seeks to perform a similar task. Each claim in our dataset has multiple associated premise articles with each article containing a large amount of text. Our goal in the first stage is to rank the evidence available in the associated premise articles at the sentence level and extract the ones which are most useful and impactful for veracity detection in the second stage. Our experiments show that an improvement in this stage directly contributes to an overall improvement in the veracity detection performance.

4.2.1 TF-IDF

We measure TF-IDF similarity between the claim text and the premise article sentences to rank the sentence level evidence. Top ranked sentences are used in the second stage to perform veracity detection. This approach is similar to the one used by Thorne et al. (2018) to extract evidence sentences from Wikipedia articles for fact checking.

4.2.2 Dense Passage Retrieval

We propose a novel way of adapting the dense passage retrieval method proposed by Karpukhin et al. (2020) for open domain question answering to the

Dataset	Claims	Labels	Review Article	Evidence	Sources
(Bachenko et al., 2008)	275	2	N/A	None	Criminal Reports
(Mihalcea and Strapparava, 2009)	600	2	N/A	None	Crowdsourced authors
(Vlachos and Riedel, 2014)	220	5	No	None	Fact Checking Websites
(Mitra and Gilbert, 2015) (CREDBank)	1,049	5	N/A	None	Twitter
(Popat et al., 2016)	5,013	2	No	Search Results	Snopes, Wikipedia
(Zubiaga et al., 2016) (PHEME)	4,842	3	N/A	None	Twitter
(Wang, 2017) (LIAR)	12,836	6	No	Claimant Metadata	Politifact
(Pérez-Rosas et al., 2018) (FakeNewsAMT)	980	2	N/A	None	News sites & Crowdsourcing
(Thorne et al., 2018) (FEVER)	185,445	3	N/A	Wikipedia Page	Wikipedia
(Shu et al., 2018) (FakeNewsNet)	23,921	2	Yes	Social Media Metadata	Politifact, GossipCop
(Baly et al., 2018)	422	2	No	Search Results	verify-sẏcom, araſeuters.com
(Gorrell et al., 2019) (RumorEval)	446	3	N/A	None	Twitter, Reddit
(Augenstein et al., 2019) (MultiFC)	36,534	2,40	Yes	Search Results	Fact Checking Websites
(Hanselowski et al., 2019) (UKP Snopes)	6,422	5	Yes	Premise Articles	Snopes Website
(Nakamura et al., 2020) (Fakeddit)	1,063,106	2,3,6	N/A	Social Media Metadata	Reddit
(Kotonya and Toni, 2020b) (PubHealth)	11,832	4	Yes	Premise Articles	Fact checking and News Websites
(Fan et al., 2020) (QABriefs)	6,897	2	N/A	Wikipedia Passages	Fact Checking Websites
(Nakov et al., 2021) (CT-FAN-21)	1,254	4	Yes	None	Fact Checking Websites
(Gupta and Srikumar, 2021) (X-Fact)	31,189	7	Yes	Search Results, Metadata	Fact Checking websites
(Aly et al., 2021) (FEVEROUS)	87,026	3	N/A	Wikipedia Page	Wikipedia
Ours (WatClaimCheck)	33,697	3,86	Yes	Premise Articles	Fact Checking Websites

Table 1: Proposed WatClaimCheck dataset compared with other existing fact checking datasets. The "Labels" column indicates the number of veracity labels. For some datasets, there are several labeling schemes with different numbers of labels separated by commas. For instance, WatClaimCheck includes the original set of 86 fine grained labels from the fact checking organizations as well as a remapping into 3 coarse labels only.

task of retrieving evidence sentences from premise articles. Karpukhin et al.'s method uses a dual encoder architecture. Each encoder is implemented using BERT (Devlin et al., 2018). The question encoder E_Q and the passage encoder E_P embed question q and passage p into d-dimensional vectors. The similarity between the question and passage is defined as the dot product of their vectors:

$$\text{sim}(q, p) = E_Q(q)^T E_P(p) \quad (1)$$

The model is then trained to learn embeddings such that the similarity score between relevant question-passage pairs will be higher than irrelevant ones.

We adapt this method for our first stage by taking advantage of the fact that the review article published by fact checking websites (along with a claim) typically contains key evidence taken from the premise articles. The evidence is usually paraphrased in order to form a coherent argument in support of the claim veracity verdict.

To train the dense passage retrieval model for stage-1, we use the claims and the associated review articles in the training set of our dataset. We form positive pairs using the claim and the sentences from the associated review article. The negative pairs are formed using that same claim and sentences from review articles associated with other claims. This corresponds to the "gold" negative sampling technique in (Karpukhin et al., 2020).

Let $D = \{\langle C_i, s_{i,j}^{R+}, s_{i,1}^{R-}, s_{i,2}^{R-}, \ldots, s_{i,n-1}^{R-} \rangle_{j=1}^{|R_i|}\}_{i=1}^{N}$ be the training data containing $\sum_{i=1}^{N} |R_i|$ instances where N is the number of claims in the training set, $|R_i|$ is the number of sentences in the review article associated with the i-th claim. Each instance is made up of a claim C_i with one positive sentence from the associated review article $s_{i,j}^{R+}$ and $n-1$ randomly chosen negative sentences $s_{i,k}^{R-}$. These negative sentences are positive sentences for other claims within the same batch. We train the model by optimizing the negative log likelihood of the positive sentences:

$$L(C_i, s_{i,j}^{R+}, s_{i,1}^{R-}, s_{i,2}^{R-}, \ldots, s_{i,n-1}^{R-}) = -\log \frac{e^{\text{sim}(C_i, s_{i,j}^{R+})}}{e^{\text{sim}(C_i, s_{i,j}^{R+})} + \sum_{k=1}^{n-1} e^{\text{sim}(C_i, s_{i,k}^{R-})}} \quad (2)$$

For model evaluation, we use the top-k recall rate for retrieving the review article sentences corresponding to the claims in the validation and test set using the similarity score. The review article sentences are retrieved from the corpus formed by all the sentences from every review article in the corresponding set.

After training, we use the encoders to encode the claim text and the sentences of the associated premise articles. We compute the similarity score using the dot product between the encoded claim vector and the premise article sentences. We use the top scoring sentences as evidence sentences in the next stage to perform claim veracity inference.

4.3 Stage-2: Claim Veracity Inference

In this section, we describe how several popular sequence models are used to classify a claim as

true, partly true/false or false based on the text of the claim, the claimant and the evidence sentences extracted in stage 1.

4.3.1 Bi-LSTM and Bi-GRU

We first consider bi-directional long short term memory (Bi-LSTM) networks and bi-directional gated recurrent units (Bi-GRUs). The evidence sentences of each premise article are concatenated with the claim and claimant, and then encoded by a Bi-LSTM or Bi-GRU into a latent vector. For N premise articles, the resulting N vectors are then averaged and passed through a softmax layer with 3 outputs corresponding to the predicted probabilities of true, partly true/false and false.

4.3.2 HAN

Instead of concatenating the evidence sentences of each premise article into a long sequence, we can also use hierarchical attention networks (HANs) (Yang et al., 2016; Mishra and Setty, 2019) to compute sentence level embeddings that are then combined into article level embeddings. A HAN is used to embed each premise article with the claim as follows. Each sentence (claimant with claim text or each evidence sentence of the premise article) is embedded as a sequence of hidden vectors (one per word) by a bi-directional recurrent network (Bi-LSTM or Bi-GRU). Then, a word-level attention layer computes a sentence level embedding. Next, those embeddings are fed to another bi-directional recurrent network (Bi-LSTM or Bi-GRU) that computes a sequence of hidden vectors (one per sentence) and a sentence level attention layer computes an embedding for the document-claim pair. Finally, the embeddings of the document-claim pairs are averaged and passed through a softmax over the labels true, partly true/false and false.

4.3.3 Transformer

We finetune a RoBERTa-base (Liu et al., 2019) model to perform claim veracity inference using the claim and the evidence sentences. We concatenate the claim text, the name of the claimant, and the evidence sentences extracted for that particular claim in the first stage to build a training data instance. The input sequence is encoded using the RoBERTa-base model and passed through a dense linear layer followed by a softmax to obtain the predicted claim veracity label distribution. We use the cross entropy loss function to train the model.

5 Experiments

We evaluate the two-stage process and the algorithms described in the previous section on the claim inference problem with our new dataset.

5.1 Stage-1 Results

In order to reduce the computational resources and memory requirements, we implement the encoders in the dense passage retrieval model using DistilRoBERTa (Dis). We use a batch size of 64 and the **in-batch negatives** technique as described in (Karpukhin et al., 2020).

We evaluate the stage-1 methods by comparing their performance using the top-k recall rate metric. The claim text is used to retrieve the ground truth review article sentences from the corpus containing all the sentences of all the review articles in the test set. The test contains a total of $114,290$ sentences and $3,373$ claims. We report the top-k recall rate for $k = 10, 25, 50, 100$ in Table 2. The results clearly show that the DPR (dense passage retrieval) method outperforms the method based on TF-IDF.

Top-k Recall	TF-IDF	DPR (DistilRoberta)
Top-10 Recall	18%	26%
Top-25 Recall	25%	38%
Top-50 Recall	30%	46%
Top-100 Recall	35%	54%

Table 2: Top-K recall performance for stage-1 methods

5.2 Stage-2 Results

To evaluate whether the inference models in stage-2 can do better with the inclusion of additional evidence sentences, we perform the experiments in stage-2 in two settings: Pooled and Averaged.

Pooled: In this setting, for each claim we pool all the sentences from every associated premise article and rank them using the similarity score. The evidence sentences are concatenated in the descending order of their similarity score. Afterwards, the claim text and evidence sentences are concatenated. The resulting text is then truncated to the maximum sequence length capability of the transformer model being used to perform claim veracity inference. For each claim, we get exactly one data instance.

Averaged: This refers to the setting where we generate one data instance per claim and associated premise article. So, if a claim has m premise articles, we get m data instances. For each premise

article associated with a claim, we score the sentences from that article and extract the top scoring sentences to form a data instance. We concatenate the evidence sentences in the descending order of their similarity score. The evidence sentences are then concatenated to the claim text and truncated to the maximum sequence length capability of the transformer model being used to perform claim veracity inference. During training, each data instance for a claim is used independently, but during inference, we compute the average of the claim veracity prediction distributions of the data instances associated with a single claim. We show in our reported results that the inclusion of additional evidence in the form of m data instances per claim (instead of 1 data instance for the pooled setting) does improve the performance when the retrieval method of stage 1 is not very effective.

We use macro F1 as the evaluation metric. We report the results in Table 3. We report all the hyper parameters used in our experiments in the appendix. The best performance when doing the claim veracity inference is obtained by using the DPR model in the first stage and the RoBERTa-base model in the second stage. We also report results for claim entailment from the review articles as an upper bound on the accuracy that could be achieved for claim inference based on the premise articles.

5.2.1 Prequential Results

We note that the traditional experimental setup of dividing a dataset at random into train, validation and test does not reflect the streaming nature of claims. When new topics arise (i.e., election, covid-19), the nature of the claims and the premise articles changes. Randomly splitting the dataset into train/validation/test ensures that all claim topics are well represented across the train/validation/test splits, which would not be the case in practice. In reality, when a new topic arises, the test split may have new types of claims that are not well represented in the train/validation splits. To evaluate the effect of this distribution shift over time, we performed a prequential evaluation (Bifet et al., 2015). More precisely, we divide the dataset into subsets corresponding to periods of 6 months. We repeatedly evaluate the performance for each 6-month period by treating the claims in that period as the test set and the claims in previous periods as the train/validation sets. This corresponds to a realistic setting where a claim verification algorithm may be re-trained every 6 months on the data seen so far to predict the veracity of the claims for the next 6 months. Naturally, the time period between each re-training iteration may be shorter than 6 months in practice. We chose 6 months simply to ensure that the size of the test set would be large enough to obtain reliable results.

Fig. 3 shows the number of claims investigated in each 6-month period in our dataset. We note two peaks. The first one in 2016 corresponds to a sudden surge of claims investigated by some fact checking websites regarding India politics. The second peak in 2020 corresponds to the 2020 US presidential election and the start of the covid-19 pandemic. Fig. 4 shows the macro F1 results achieved by the top 4 algorithms with DPR evidence in each 6-month period. We note that the prequential results are significantly lower than the results in the DPR column of Table 3. This drop of accuracy is precisely due to the distribution shift of claims that naturally occurs over time. We also note a trend whereby the accuracy increases as time passes by. This is explained by the fact that more data is available for training in later time periods. We strongly recommend that future algorithms be evaluated in prequential mode since this evaluation setup is more realistic.

6 Conclusion

This paper introduces a new dataset for automated fact checking. WatClaimCheck includes premise articles used by professional fact checkers and therefore corresponds more closely to the task of claim veracity inference in automated fact checking. An important challenge is the extraction of relevant facts from the premise articles since it is not generally possible to apply heavyweight models on the entire content of all premise articles. To that effect, we described how to train the encoders of a dense passage retrieval technique with the review articles and then transfer the resulting retrieval technique to the premise articles. This increased the overall performance of the claim verification algorithms. We also performed a prequential evaluation that highlighted an important distribution shift that caused a significant drop in accuracy for all algorithms. We strongly recommend that future algorithms be evaluated in prequential mode. In fact, an important direction for future research would be to design algorithms based on transfer learning or domain generalization that can cope better with this distributional shift. We also note that the tech-

Algorithm	Review	Evidence(TF-IDF)	Evidence(DPR)
Bi-GRU	0.779±0.009	0.418±0.010	0.453±0.009
Bi-LSTM	0.777±0.008	0.421±0.011	0.454±0.010
HAN-Bi-GRU	**0.821±0.007**	0.445±0.010	0.471±0.009
HAN-Bi-LSTM	0.818±0.007	0.444±0.008	0.471±0.011
Roberta-base (pooled)	0.741±0.005	0.541±0.017	**0.580±0.009**
Roberta-base (averaged)	0.741±0.005	**0.563±0.010**	0.565±0.009

Table 3: Macro F1 score averaged over 10 runs with standard deviation

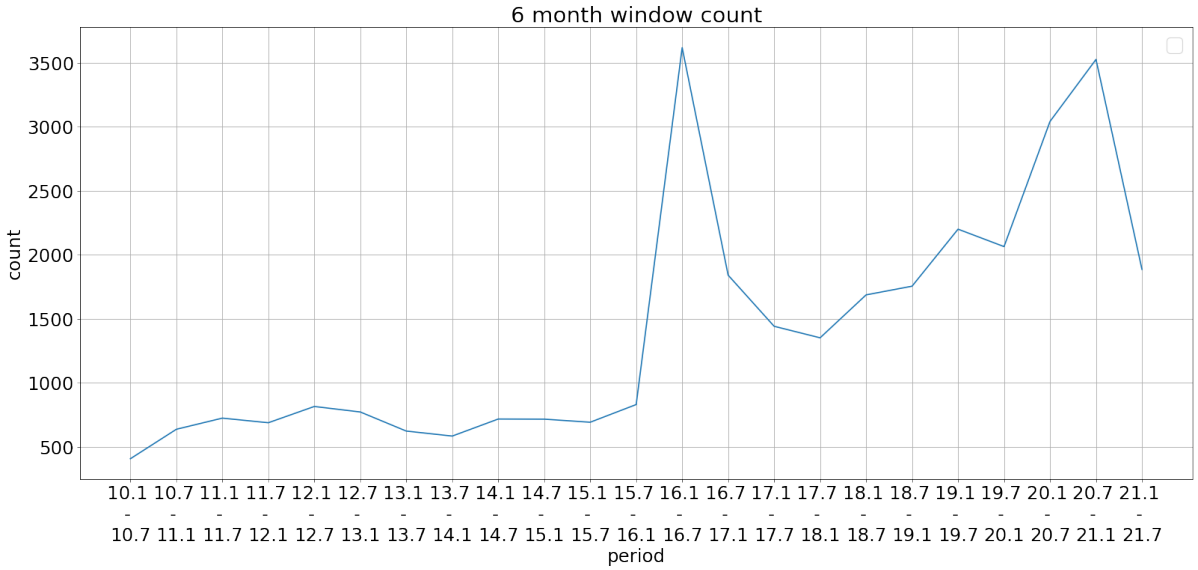

Figure 3: count of claims in each time window

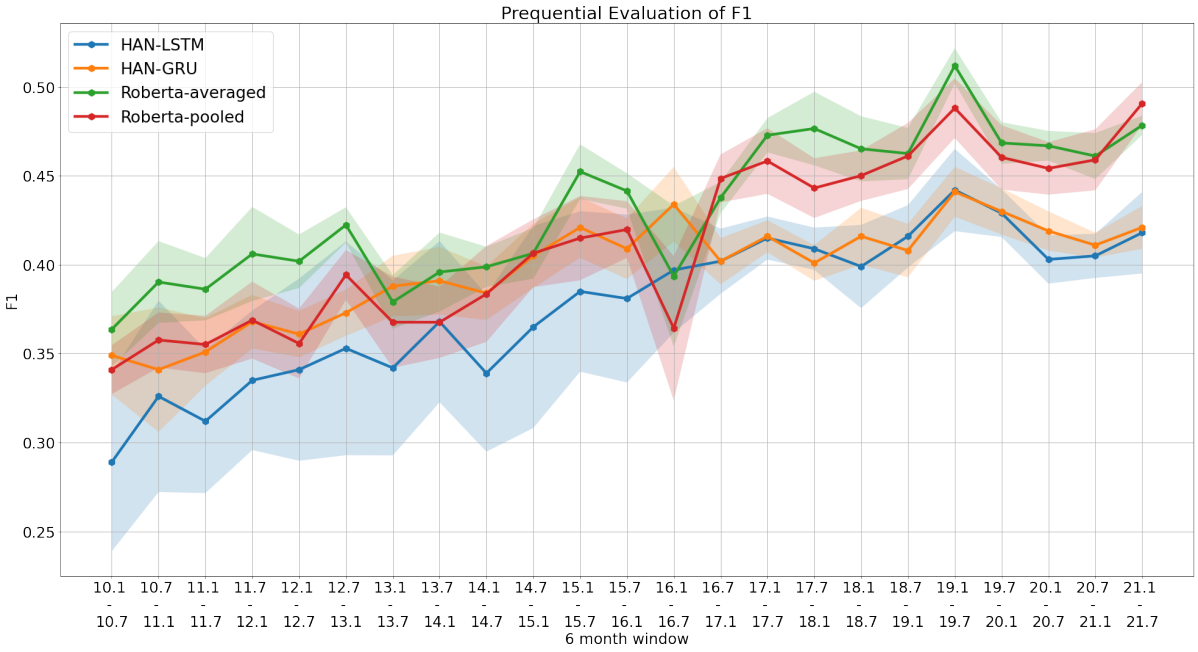

Figure 4: Prequential evaluation score based on macro F1 (average of 10 runs with standard deviation)

niques that we evaluated are black boxes and therefore it is not clear how they do inference. Hence, another direction for future research would be to develop inference techniques that are explainable in the sense that they could provide explanations to the users to justify their veracity prediction for a claim.

Acknowledgements

We thank the Schulich Foundation and the Natural Sciences and Engineering Research Council of Canada (NSERC) for funding support. Resources used in this work were provided, in part, by the Province of Ontario, the Government of Canada through CIFAR, and companies sponsoring the Vector Institute https://vectorinstitute.ai/partners/.

References

DistilRoBERTa model. https://huggingface.co/distilroberta-base. Accessed: 2021-11-01.

Rami Aly, Zhijiang Guo, Michael Sejr Schlichtkrull, James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Oana Cocarascu, and Arpit Mittal. 2021. Feverous: Fact extraction and verification over unstructured and structured information. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

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)*, pages 4685–4697, Hong Kong, China. Association for Computational Linguistics.

Joan Bachenko, Eileen Fitzpatrick, and Michael Schonwetter. 2008. Verification and implementation of language-based deception indicators in civil and criminal narratives. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 41–48, Manchester, UK. Coling 2008 Organizing Committee.

Ramy Baly, Mitra Mohtarami, James Glass, Lluís Màrquez, Alessandro Moschitti, and Preslav Nakov. 2018. Integrating stance detection and fact checking in a unified corpus. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 21–27, New Orleans, Louisiana. Association for Computational Linguistics.

Fuat Can Beylunioglu. 2020. Using a credibility classifier to improve health-related information retrieval. Master's thesis, University of Waterloo.

Albert Bifet, Gianmarco de Francisci Morales, Jesse Read, Geoff Holmes, and Bernhard Pfahringer. 2015. Efficient online evaluation of big data stream classifiers. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 59–68.

Alessandro Bondielli and Francesco Marcelloni. 2019. A survey on fake news and rumour detection techniques. *Information Sciences*, 497:38–55.

Luís Borges, Bruno Martins, and Pável Calado. 2019. Combining similarity features and deep representation learning for stance detection in the context of checking fake news. *Journal of Data and Information Quality (JDIQ)*, 11(3):1–26.

Nadia K Conroy, Victoria L Rubin, and Yimin Chen. 2015. Automatic deception detection: Methods for finding fake news. *Proceedings of the Association for Information Science and Technology*, 52(1):1–4.

Limeng Cui, Haeseung Seo, Maryam Tabar, Fenglong Ma, Suhang Wang, and Dongwon Lee. 2020. Deterrent: Knowledge guided graph attention network for detecting healthcare misinformation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 492–502.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Angela Fan, Aleksandra Piktus, Fabio Petroni, Guillaume Wenzek, Marzieh Saeidi, Andreas Vlachos, Antoine Bordes, and Sebastian Riedel. 2020. Generating fact checking briefs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7147–7161.

Siva Charan Reddy Gangireddy, Cheng Long, and Tanmoy Chakraborty. 2020. Unsupervised fake news detection: A graph-based approach. In *Proceedings of the 31st ACM Conference on Hypertext and Social Media*, pages 75–83.

Genevieve Gorrell, Elena Kochkina, Maria Liakata, Ahmet Aker, Arkaitz Zubiaga, Kalina Bontcheva, and Leon Derczynski. 2019. SemEval-2019 task 7: RumourEval, determining rumour veracity and support for rumours. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 845–854, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. 2022. A Survey on Automated Fact-Checking. *Transactions of the Association for Computational Linguistics*, 10:178–206.

Ashim Gupta and Vivek Srikumar. 2021. X-fact: A new benchmark dataset for multilingual fact checking. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 675–682, Online. Association for Computational Linguistics.

Andreas Hanselowski, Christian Stab, Claudia Schulz, Zile Li, and Iryna Gurevych. 2019. A richly annotated corpus for different tasks in automated fact-checking. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 493–503, Hong Kong, China. Association for Computational Linguistics.

Heejung Jwa, Dongsuk Oh, Kinam Park, Jang Mook Kang, and Heuiseok Lim. 2019. exBAKE: Automatic fake news detection model based on bidirectional encoder representations from transformers (BERT). *Applied Sciences*, 9(19):4062.

Hamid Karimi and Jiliang Tang. 2019. Learning hierarchical discourse-level structure for fake news detection. *arXiv preprint arXiv:1903.07389*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Neema Kotonya and Francesca Toni. 2020a. Explainable automated fact-checking: A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5430–5443, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Neema Kotonya and Francesca Toni. 2020b. Explainable automated fact-checking for public health claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7740–7754, Online. Association for Computational Linguistics.

Srijan Kumar and Neil Shah. 2018. False information on web and social media: A survey. *arXiv preprint arXiv:1804.08559*.

Qian Li, Qingyuan Hu, Youshui Lu, Yue Yang, and Jingxian Cheng. 2019. Multi-level word features based on CNN for fake news detection in cultural communication. *Personal and Ubiquitous Computing*, pages 1–14.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Rada Mihalcea and Carlo Strapparava. 2009. The lie detector: Explorations in the automatic recognition of deceptive language. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 309–312, Suntec, Singapore. Association for Computational Linguistics.

Rahul Mishra and Vinay Setty. 2019. Sadhan: Hierarchical attention networks to learn latent aspect embeddings for fake news detection. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 197–204.

Tanushree Mitra and Eric Gilbert. 2015. Credbank: A large-scale social media corpus with associated credibility annotations. In *ICWSM*.

Kai Nakamura, Sharon Levy, and William Yang Wang. 2020. Fakeddit: A new multimodal benchmark dataset for fine-grained fake news detection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6149–6157, Marseille, France. European Language Resources Association.

Preslav Nakov, Giovanni Da San Martino, Tamer Elsayed, Alberto Barrón-Cedeño, Rubén Míguez, Shaden Shaar, Firoj Alam, Fatima Haouari, Maram Hasanain, Nikolay Babulkov, Alex Nikolov, Gautam Kishore Shahi, Julia Maria Struß, and Thomas Mandl. 2021. The clef-2021 checkthat! lab on detecting check-worthy claims, previously fact-checked claims, and fake news. In *Advances in Information Retrieval*, pages 639–649, Cham. Springer International Publishing.

Alexandra Olteanu, Stanislav Peshterliev, Xin Liu, and Karl Aberer. 2013. Web credibility: Features exploration and credibility prediction. In *European conference on information retrieval*, pages 557–568. Springer.

Verónica Pérez-Rosas, Bennett Kleinberg, Alexandra Lefevre, and Rada Mihalcea. 2018. Automatic detection of fake news. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3391–3401, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. 2016. Credibility assessment of textual claims on the web. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, page 2173–2178, New York, NY, USA. Association for Computing Machinery.

Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. 2017. Where the truth lies: Explaining the credibility of emerging claims

on the web and social media. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 1003–1012.

Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. 2018. DeClarE: Debunking fake news and false claims using evidence-aware deep learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 22–32, Brussels, Belgium. Association for Computational Linguistics.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5835–5847.

Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. 2017. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2931–2937.

Julio CS Reis, André Correia, Fabrício Murai, Adriano Veloso, and Fabrício Benevenuto. 2019. Explainable machine learning for fake news detection. In *Proceedings of the 10th ACM Conference on Web Science*, pages 17–26.

Ruiyang Ren, Shangwen Lv, Yingqi Qu, Jing Liu, Wayne Xin Zhao, Qiaoqiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021. Pair: Leveraging passage-centric similarity relation for improving dense passage retrieval. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2173–2183.

Viktor Schlegel, Goran Nenadic, and Riza Batista-Navarro. 2020. Beyond leaderboards: A survey of methods for revealing weaknesses in natural language inference data and models. *arXiv preprint arXiv:2005.14709*.

Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2018. Fakenewsnet: A data repository with news content, social context and dynamic information for studying fake news on social media. *CoRR*, abs/1809.01286.

Shane Storks, Qiaozi Gao, and Joyce Y Chai. 2019. Recent advances in natural language inference: A survey of benchmarks, resources, and approaches. *arXiv preprint arXiv:1904.01172*.

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, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

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, MD, USA. Association for Computational Linguistics.

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 (Volume 2: Short Papers)*, pages 422–426, Vancouver, Canada. Association for Computational Linguistics.

Youze Wang, Shengsheng Qian, Jun Hu, Quan Fang, and Changsheng Xu. 2020. Fake news detection via knowledge-driven multimodal graph convolutional networks. In *Proceedings of the 2020 International Conference on Multimedia Retrieval*, pages 540–547.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

Arkaitz Zubiaga, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Peter Tolmie. 2016. Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PLOS ONE*, 11(3):1–29.

A Appendix

A.1 Dataset

Since the dataset is larger than the 200 Mb limit for the supplementary material, we include a sample corresponding to the data collected from March 15 to July 1, 2021 in the supplementary material. A link to the entire dataset will be made available once the paper is accepted.

A.2 Hyperparameters

The code will be made public once the paper is accepted. Table 4 lists the hyperparameters used for the Bi-LSTM, Bi-GRU and HANs. Table 5 describes the hyperparameters of the DPR technique in stage 1 and Table 6 lists the hyperparameters of RoBERTA-base in stage 2.

Parameter	Value
Max Sentence Length	20 words per sentence
Max Sentence Count	30 sentences per claim
Embedding Dimension	100
Batch Size	64
Learning Rate	0.01
Convergence Patience	6 epochs
Optimizer	Adam

Table 4: Hyperparameters used in Bi-LSTM/GRU and HAN

Parameter	Value
Max Sequence Length	512
Batch Size	64
Learning Rate	$1e^{-5}$
Epochs	500
Optimizer	AdamW

Table 5: Hyperparameters used in stage-1 DPR model

Parameter	Value
Max Sequence Length	512
Batch Size	12
Learning Rate	$1e^{-5}$
Epochs	10
Optimizer	AdamW

Table 6: Hyperparameters used in stage-2 RoBERTA-base averaged and pooled models