NLP for Fact-Checking and Claim Assessment

A Language Model based approach

Othman El houfi
MSc Data Science & Machine Learning
CY Cergy Paris University, France
othmanelhoufi@gmail.com

Dimitris Kotzinos
University Professor
CY Cergy Paris University, France
dimitrios.kotzinos@cyu.fr

***Abstract*—As false information and fake news are propagating throughout the internet and social networks, the need of fact-checking operations becomes necessary in order to maintain a truthful digital environment where general information can be reliably exploited whether in politics, finance or other domains. The need of this online claim assessment comes from the fact that fake news and false information can have a big negative impact on politics, economy (2016 USA Elections) and public health (COVID-19).**
A number of solutions have been proposed to deal with this problem and limit the spread of false information, both manual and automatic. Of course the manual approaches done on websites such as *PolitiFact.com*, *FactCheck.org* and *Snopes.com* don't construct a viable solution for the long term as the speed and scale of information propagation increase exponentially rendering this manual fact-checking operation where human fact-checkers can't scale up at the same rate limited and incapable of solving the problem.
Here, we present our contribution in this regard: an automated solution for fact-checking using FEVER dataset as a source of truth and a state of the art language models used today for NLP tasks (BERT, RoBERTa, XLNet...) in order to classify a given claim as *Supports*, *Refutes* or *Not enough information (NEI)*. We successfully prove that fine-tuning a LM with the correct settings can achieve an accuracy of 62% and F1-score of 61% which is better than the majority of fact-checking methods that exists today.

***Index Terms*—Natural Language Processing, Language Model, Wikipedia, Fine-tuning, Zero-shot Learning, Text processing, Natural Language Inferencing, Fact-Checking, Fake-news.**

I. Introduction

From a social and psychological perspective, humans have been proven irrational and vulnerable when differentiating between truth and false news (typical accuracy ranges between 55% and 58%) [1], thus fake news obtain public trust relatively easier than truthful news because individuals tend to trust fake news after constant exposure (*Validity effect*), or if it confirms their pre-existing beliefs (*Confirmation bias*), or simply due to the obligation of participating socially and proving a social identity (*Peer pressure*). The social sciences are still trying to comprehend the biological motivations that makes fake news more appealing to humans.

On the other hand, the growth of social media platforms resulted in a huge acceleration of news spreading whether true or false. As of Aug. 2017, 67% [1] of Americans get their news from social media. These platforms even give the user the right to share, forward, vote and participate to online discussions. All of this made the problem of fake news spreading more and more dangerous, our economies for example, are not robust to the spread of falsity, false rumors have affected stock prices and the motivations for large-scale investments, as we witnessed after a false tweet claimed that Barack Obama was injured in an explosion which caused $130 billion drop in stock value [2]. Another recent example is related to public health where rumors about COVID-19 vaccines and drug companies influenced people in their decision on getting vaccinated.

That being said, is there a way to monitor the spread of fake news through social media? Or more specifically, how can we differentiate between fake news and truthful news, and at what level of confidence can we do that?

From a computer engineering perspective, various approaches were examined:

- **Knowledge-based Fake News Detection [3]:** a method aims to assess news authenticity by comparing the knowledge extracted from to-be verified news content with known facts, also called fact-checking.
- **Style-based Fake News Detection [4]:** focuses on the style of writing, i.e. the form of text rather than its meaning.
- **Propagation-based Fake News Detection [5]:** a principled way to characterize and understand hierarchical propagation network features. We perform a statistical comparative analysis over these features, including micro-level and macro-level, of fake news and true news.
- **Credibility-based Fake News Detection [6]:** the information about authors of news articles can indicate news credibility and help detect fake news.

In this paper we will focus on a modern approach that utilizes Language Models (LMs) for fact-checking. The goal is not to implement an algorithm that scans social networks for real time fake news detection, but rather we will design a model that can assess with a degree of confidence the truthfulness or falseness of a claim given by a user as an input by exploiting LMs that were already trained on Wikipedia, and fine-tune each LM for a downstream task in order to solve this classification problem.

II. Related works

A. Language model based approach [7] [8]

A paper entitled *"Language Models as Fact Checkers?"* published by a team from *FacebookAI* and *Hong Kong University of Science and Technology*, provides an example of a fact-checking model using zero-shot LM that outperforms a random baseline LM using the FEVER dataset[9].

The goal of fact-checking as mentioned previously, and relatively to this paper, is to validate the truthfulness of a given claim. Each claim is assigned to one of these labels: *Supports*, *Refutes* or *Not enough information (NEI)* to verify.

This paper describes the difference between Traditional Pipeline fact-checking models and their zero-shot fact-checking LM:

- **Traditional pipeline:** this type of models access knowledge within an external knowledge base like Wikipedia in order to validate a claim. It involves information retrieval modules such as document retrieval and sentence retrieval.
- **Zero-shot LM pipeline:** it replaces both the external knowledge base and the information retrieval modules with a pre-trained language model.

They used the publicly available 24-layer BERT-Large as LM, which was pre-trained on Wikipedia in 2018. After fine-tuning the model they achieved 57% in accuracy and 57% in F1-macro score which was better than the baseline BERT model (without fine-tuning) that achieved 49% in accuracy and 44% in F1-macro score.

B. Perplexity based approach [10] [11]

In March 2021, *Nayeon Lee*, *Yejin Bang*, *Andrea Madotto*, *Madian Khabsa*, and *Pascale Fung* published a paper called *Towards Few-Shot Fact-Checking via Perplexity* where they propose a new approach of the powerful transfer learning ability of a language model via a perplexity score. Using a method called *few-shot learning*, they designed a model that outperforms major class baseline models by more than 10% on the F1-Macro metric score.
In this paper the goal is to determine the veracity of a claim given some evidence, for this they define a claim, evidence pair. The label *Supported* is assigned when relevant evidence exists that supports the claim, and *Unsupported* label for the opposite case.
Unsupported claims on average have higher perplexity than *Supported* claims. For example, *Supported* claim "Washing hands prevents the spread of diseases" has a perplexity value of 96.74, whereas the *Unsupported* claim "All dogs speak English fluently" has a much higher perplexity value of 328.23.
The datasets used in this experiment are: Covid19-Scientific, Covid19-Social, and FEVER. As for the perplexity based experiment they used one unidirectional LM and one masked LM:

- PPL_{GPT2-B} : a single-parameter classifier based on perplexity from GPT2-base [12] (unidirectional LM)
- PPL_{BERT-B} : a single-parameter classifier based on perplexity from BERT-base [13] (Masked LM)

They took into consideration the accuracy and the F1-Macro metrics for the evaluation. Because the datasets are unbalanced, they mainly consider the F1-Macro score over accuracy as an overall evaluation. The perplexity-based classifiers, especially PPL_{GPT2-B}, outperform all Major Class baselines across all tasks in all settings. For instance, PPL_{GPT2-B} achieved accuracy of 67.48% and F1-macro score of 64.70% on FEVER dataset.
On the other hand the classification was limited to two labels (*Supported* and *Unsupported*) which does not solve the entire classification problem in the FEVER dataset that provides three labels (*Supports*, *Refutes* or *Not enough information (NEI)*).

III. Proposed method

Most of the fact-checking algorithms today involving knowledge-based verification uses a traditional pipeline that puts in place a module for retrieving articles from an external source, another module for retrieving relevant sentences from each article and a last module for natural language inferencing (NLI) to classify a claim.

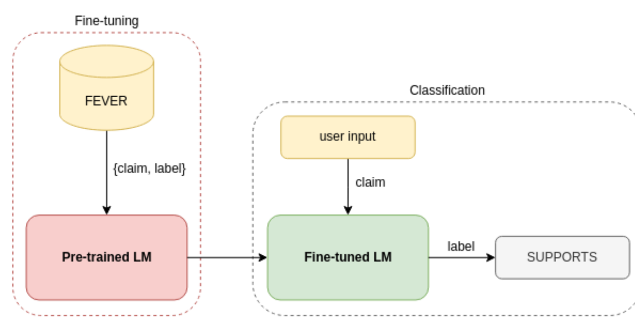

Fig. 1. Proposed method pipeline.

In this paper we present a method that is fully reliant on the powerfulness of today's best LMs. As illustrated in Figure 1, we start by fine-tuning each model for the downstream task that is claim classification using the FEVER dataset, then each model is employed to assess the validity of new input claims. This approach takes into consideration only an internal knowledge source (FEVER) for fine-tuning, that is for the learning phase, which makes the prediction phase knowledge-free rather than utilizing external knowledge sources for retrieving articles and sentences.

It is also important to mention we only use LMs for classifying claims and not for generating evidence. We leave generating evidences with language models for future work.

A. FEVER Dataset

FEVER (Fact Extraction and VERification) consists of 185,445 claims generated by altering sentences extracted from Wikipedia and subsequently verified without knowledge of the sentence they were derived from. The claims are classified as Supported, Refuted or NotEnoughInfo. For the first two classes, the annotators also recorded the sentence(s) forming the necessary evidence for their judgment[9].

Claim: The Rodney King riots took place in the most populous county in the USA.

[wiki/Los_Angeles_Riots]
The 1992 Los Angeles riots, also known as the Rodney King riots were a series of riots, lootings, arsons, and civil disturbances that occurred in Los Angeles County, California in April and May 1992.

[wiki/Los_Angeles_County]
Los Angeles County, officially the County of Los Angeles, is the most populous county in the USA.

Verdict: Supported

Fig. 2. Manually verified claim requiring evidence from multiple Wikipedia pages.

Since our mission is to classify claims to *SUPPORTS*, *REFUTES* or *NEI* and not generating evidence, we omit the evidences information in the dataset. At that point we map each label to an integer: $\{$*SUPPORTS* : 1, *REFUTES* : 0, *NEI* : 2$\}$ and that is all we do as far as data pre-preprocessing goes.

ID	**Claim**	**Label**
79044	The Apple Store first opened in 2001.	1
117129	Adventure Time won an Oscar.	0
55061	Yamaha Corporation produces hardware.	2

TABLE I
EXAMPLES OF FEVER CLAIMS AND LABELS.

Finally, we split the dataset to training, validation and testing sets that we can use for LM fine-tuning and testing:

Split	SUPPORTS	REFUTES	NEI	**Total**
Train	80,035	29,775	35,639	145,449
Val	3,333	3,333	3,333	9,999
Test	3,333	3,333	3,333	9,999

TABLE II
DATASET SPLIT SIZES FOR SUPPORTS, REFUTES AND NOTENOUGHINFO (NEI) CLASSES.

B. Language Models

The year 2018 has been an inflection point for NLP as Google introduced a LM called BERT (Bidirectional Encoder Representations from Transformers)[13]. This model was described as state-of-the-art model that solves the most difficult tasks in NLP, it is also used today in Google's search engine for text completion and translation. BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task specific architecture modifications. From there many LMs were introduced that uses the same architecture as BERT but with small changes such as number of parameters and data on which the model was pre-trained.

Comparison	BERT October 11, 2018	RoBERTa July 26, 2019	DistilBERT October 2, 2019	ALBERT September 26, 2019
Parameters	**Base:** 110M **Large:** 340M	**Base:** 125 **Large:** 355	**Base:** 66	**Base:** 12M **Large:** 18M
Layers / Hidden Dimensions / Self-Attention Heads	**Base:** 12 / 768 / 12 **Large:** 24 / 1024 / 16	**Base:** 12 / 768 / 12 **Large:** 24 / 1024 / 16	**Base:** 6 / 768 / 12	**Base:** 12 / 768 / 12 **Large:** 24 / 1024 / 16
Training Time	**Base:** 8 x V100 x 12d **Large:** 280 x V100 x 1d	1024 x V100 x 1 day (4-5x more than BERT)	**Base:** 8 x V100 x 3.5d (4 times less than BERT)	[not given] **Large:** 1.7x faster
Performance	Outperforming SOTA in Oct 2018	88.5 on GLUE	97% of BERT-base's performance on GLUE	89.4 on GLUE
Pre-Training Data	BooksCorpus + English Wikipedia = 16 GB	BERT + CCNews + OpenWebText + Stories = 160 GB	BooksCorpus + English Wikipedia = 16 GB	BooksCorpus + English Wikipedia = 16 GB
Method	Bidirectional Transformer, MLM & NSP	BERT without NSP, Using Dynamic Masking	BERT Distillation	BERT with reduced parameters & SOP (not NSP)

Fig. 3. Comparison of BERT, RoBERTa, DistilBERT, and ALBERT. [1]

In this paper we will classify claims by fine-tuning the following LMs:

- BERT-base-uncased [13]
- RoBERTa-base [14]
- DistilBERT-base-uncased [15]
- XLNET-base-cased [16]
- ALBERT-base-v2 [17]
- BigBird-RoBERTa-base [18]

All of these LMs were trained on Wikipedia dataset containing cleaned articles. The datasets are built from the Wikipedia dump [2] with one split per language. Each example contains the content of one full Wikipedia article with cleaning to strip markdown and unwanted sections (references, etc.). So in one way, all these LMs were trained on facts from Wikipedia, for example if we give the following input *"Paris is the capital of [MASK]."* to BERT-base model, the output will be *"Paris is the capital of France."* with a probability of 0.951.
Therefore the conjecture behind our method is that: by fine-tuning LMs that were pre-trained on Wikipedia, and by exploiting the already stored knowledge within these LMs, we can create a self-knowledge-independent fact-checking classifier.

IV. EXPERIMENT PROTOCOL AND RESULTS

A. Experiment Setup

As mentioned before, we conduct our experiment on the FEVER dataset using the splits in Table II. As for LMs

[1] https://humboldt-wi.github.io/blog/research/information_systems_1920/uncertainty_identification_transformers/
[2] https://dumps.wikimedia.org/

fine-tuning we used one GPU, the NVIDIA RTX6000P-8C with 24Go of GDDR6 memory and a peak single precision floating point performance of 14,9 Tflops.

For all models we chose the following hyperparameters:

- Tokenizer max sequence length: 128
- Output layer size: 3 (classes: 0, 1 and 2)
- Activation function: GeLU
- Learning rate: 3e-5
- Optimization: Adam with linear decay
- Loss function: Cross-Entropy
- Epochs: 3
- Training batch size: 20
- Validation batch size: 20

These hyperparameters may not be optimal, but they were chosen after many different repeated experiments. Optimizing the models is left for future work.

B. Evaluation Metric

Most of the fact-checking methods that use FEVER dataset employ FEVER scoring[9], a metric that considers classification accuracy and evidence recall, but since we don't tackle the evidence problem in our approach, we rely on accuracy, recall, precision and F1-score of the model classification as metrics for evaluation.
In addition, we track other aspects of each LM such as training time, model size, and memory usage during the fine-tuning step in order to make an overall evaluation.

C. Results & Discussion

The results of the six models are reported in Table III. We can observe that our approach yields better results than FacebookAI's model and most of the traditional methods that involves external information retrieval modules.

Specifically the fine-tuned *RoBERTa-base* model surpasses the fine-tuned *BERT-large* model created by FacebookAI in every metric. For instance our model achieved an accuracy score of 62% and a macro precision score of 64% which is an improvement of 5% and 7% respectively. Not only that, it is also worth mentioning that *RoBERTa-base* has only 125 million parameters and 12 encoding layers in comparison to *BERT-large* that has 340 million parameters and 24 encoding layers, thus rendering our model more efficient to train, to store and to implement.

The improvements in classification are presumably due to the fact that *RoBERTa* is pre-trained on more data than *BERT* as shown in Figure 3. On the other hand, our fine-tuned *BERT-base* model also achieved better results than FacebookAI's model which may be explained by the difference of hyperparameters choice.

Similarly, the fine-tuned *RoBERTa-base* model achieved more accuracy score for label classification than most of the traditional pipelines. Therefore it puts our approach on the top-10 best fact-checking models that were published by FEVER community[19] (without taking into consideration the evidence generation metric). It is also safe to say that the *DREAM* fact-checking model is certainly superior than all our LMs as it achieved an accuracy score of 77% which is 15% higher. This is a proof that there is still much room for subsequent research and improvements.

Same as FacebookAI results[7], and upon examining the results of our fine-tuned LMs closely, we also find that all our LMs struggle immensely with the NEI category (lowest F1 scores) indicating that our current approach might also need specific modules to better tackle that category.

Furthermore, the difference in classification performance between our LMs is not big considering the fact that each LM was pre-trained differently. The only model that performed poorly is *ALBERT-base-v2*, it achieved only 53% accuracy score and 52% macro F1 score. This can be explained by the reduced number of the model's parameters (12 million vs. *BERT*'s 110 million).

We can also see the differences between each LM during the training and the evaluation as illustrated in Figure 4 and Figure 5.

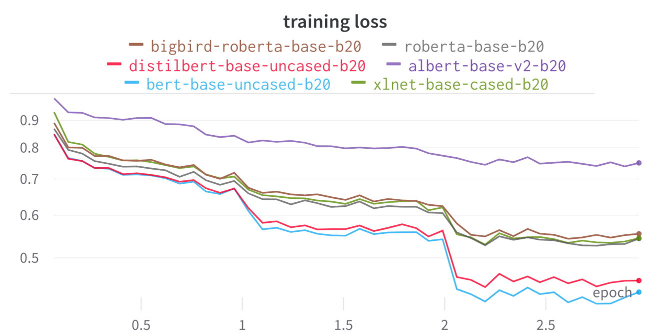

Fig. 4. The Cross-Entropy Loss of each LM during training.

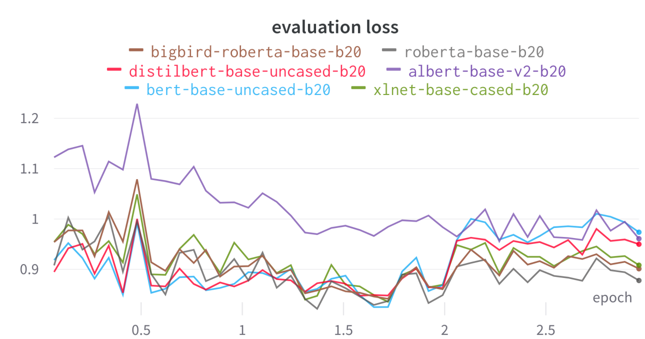

Fig. 5. The Cross-Entropy Loss of each LM during evaluation.

Fine-tuned model	Label	prec	recall	f1	accuracy	macro prec	macro recall	macro f1
BERT-base-uncased	SUPPORTS	0.55	0.78	0.64	**0.62**	0.63	**0.62**	**0.61**
	REFUTES	0.75	0.59	0.66				
	NEI	0.61	0.47	0.53				
ALBERT-base-v2	SUPPORTS	0.46	0.81	0.59	0.53	0.58	0.53	0.52
	REFUTES	0.77	0.46	0.58				
	NEI	0.50	0.33	0.40				
DistilBERT-base-uncased	SUPPORTS	0.54	0.78	0.64	0.61	0.63	0.61	**0.61**
	REFUTES	0.75	0.58	0.65				
	NEI	0.60	0.47	0.53				
RoBERTa-base	SUPPORTS	0.54	0.81	0.65	**0.62**	**0.64**	**0.62**	**0.61**
	REFUTES	0.75	0.59	0.66				
	NEI	0.63	0.45	0.53				
BigBird-RoBERTa-base	SUPPORTS	0.53	0.81	0.64	0.61	**0.64**	0.61	0.60
	REFUTES	0.75	0.58	0.66				
	NEI	0.63	0.44	0.52				
XLNET-base-cased	SUPPORTS	0.53	0.81	0.64	0.61	0.63	0.61	0.60
	REFUTES	0.74	0.59	0.65				
	NEI	0.63	0.43	0.51				
Related work	**Label**	**prec**	**recall**	**f1**	**accuracy**	**macro prec**	**macro recall**	**macro f1**
BERT-large [7]	SUPPORTS	0.54	0.67	0.59	0.57	0.57	0.57	0.57
	REFUTES	0.62	0.55	0.58				
	NEI	0.57	0.49	0.53				
FEVER Baseline [19]	-	-	-	-	0.49	-	-	-
Ohio State University [19]	-	-	-	-	0.50	-	-	-
Columbia NLP [19]	-	-	-	-	0.58	-	-	-
Papelo [19]	-	-	-	-	0.61	-	-	-
UNC-NLP [19]	-	-	-	-	0.68	-	-	-
DREAM [20]	-	-	-	-	**0.77**	-	-	-

TABLE III
CLASSIFICATION METRICS FOR EACH FINE-TUNED LM USING OUR APPROACH VS. BERT-LARGE FINE-TUNED BY FACEBOOKAI TEAM VS. OTHER MODELS BASED ON KNOWLEDGE GRAPHS AND/OR TRADITIONAL PIPELINES THAT USES FEVER DATASET (WE TAKE INTO CONSIDERATION ONLY THE ACCURACY OF LABEL CLASSIFICATION AND NOT THE FEVER SCORING SYSTEM).

As expected, *RoBERTa-base* model reached the lowest Cross-Entropy loss of 0.878 during evaluation while *ALBERT-base-v2* sustained a higher loss of 0.961 during evaluation (also 0.75 during training). It is even more explicit in Figure 6 and Figure 7 that *ALBERT-base-v2* had the lowest Matthews correlation coefficient score (mcc) as well as accuracy score during evaluation.

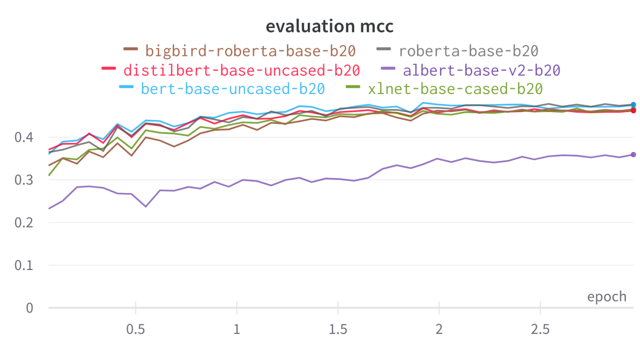

Fig. 6. The Matthews correlation coefficient score of each LM during evaluation.

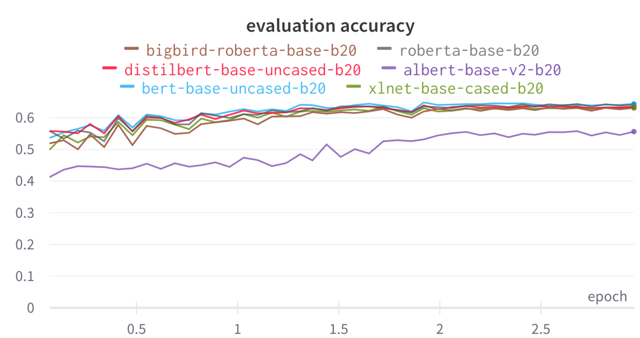

Fig. 7. The accuracy score of each LM during evaluation.

Even if the other LM reached the same accuracy and mcc as *RoBERTa-base* during evaluation or a lower loss during training they did not perform as well in the test set which was demonstrated in Table III.

Finally, it is also important to mention that training time differs from one LM to another (Figure 8) as the number of parameters are not similar, for instance *XLNET-base-cased* model spent 1h56m for training while *RoBERTa-base* took 1h25m and achieved better results. The idea that we are trying to prove is we don't need a bigger model in order to achieve greater results for a downstream task like text classification.

Fig. 8. Training time of each LM.

V. Conclusion & Future Work

In this paper, we explored the capabilities of language models to be fine-tuned and utilized for a downstream task that is fact-checking claims. We have successfully proven the effectiveness of pre-trained language models as an independent source of knowledge rather than implementing modules for external information retrieval adopted by traditional approaches. Our experiment conclusively yields results that surpasses most of the existing fact-checking methods both traditional and LM-based. Nevertheless, our approach does not beat state-of-art traditional models leaving us with more paths to explore in order to produce a reliable fact-checking engine.

In time to come, we plan to investigate solutions to deal with NEI category where our approach struggles. In addition, we will attempt to combine other models with our approach like credibility-based or style-based models, we will also implement evidence generation alongside claim classification in order to provide the user with reliable information.

VI. Acknowledgements

We would like to thank Massinissa Yebka for providing us with the computational power that made our calculations possible in a short time-frame.

References

[1] Xinyi Zhou, Reza Zafarani, Kai Shu, and Huan Liu. Fake news: Fundamental theories, detection strategies and challenges. In *Proceedings of the twelfth ACM international conference on web search and data mining*, pages 836–837, 2019.

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

[3] Anton Chernyavskiy, Dmitry Ilvovsky, and Preslav Nakov. Whatthewikifact: Fact-checking claims against wikipedia. *arXiv preprint arXiv:2105.00826*, 2021.

[4] Piotr Przybyla. Capturing the style of fake news. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[5] Kai Shu, Deepak Mahudeswaran, Suhang Wang, and Huan Liu. Hierarchical propagation networks for fake news detection: Investigation and exploitation. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 14, pages 626–637, 2020.

[6] Niraj Sitaula, Chilukuri K Mohan, Jennifer Grygiel, Xinyi Zhou, and Reza Zafarani. Credibility-based fake news detection. In *Disinformation, Misinformation, and Fake News in Social Media*, pages 163–182. Springer, 2020.

[7] Nayeon Lee, Belinda Z Li, Sinong Wang, Wen-tau Yih, Hao Ma, and Madian Khabsa. Language models as fact checkers? *arXiv preprint arXiv:2006.04102*, 2020.

[8] Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*, 2019.

[9] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. Fever: a large-scale dataset for fact extraction and verification. *arXiv preprint arXiv:1803.05355*, 2018.

[10] Nayeon Lee, Yejin Bang, Andrea Madotto, Madian Khabsa, and Pascale Fung. Towards few-shot fact-checking via perplexity. *arXiv preprint arXiv:2103.09535*, 2021.

[11] Nayeon Lee, Yejin Bang, Andrea Madotto, and Pascale Fung. Misinformation has high perplexity. *arXiv preprint arXiv:2006.04666*, 2020.

[12] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[14] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[15] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[16] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.

[17] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019.

[18] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems*, 33:17283–17297, 2020.

[19] James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. The fact extraction and verification (fever) shared task. *arXiv preprint arXiv:1811.10971*, 2018.

[20] Wanjun Zhong, Jingjing Xu, Duyu Tang, Zenan Xu, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. Reasoning over semantic-level graph for fact checking. *arXiv preprint arXiv:1909.03745*, 2019.