

A Paragraph-level Multi-task Learning Model for Scientific Fact-Verification

Xiangci Li,¹ * Gully Burns,² Nanyun Peng³

¹ University of Texas at Dallas

² Chan Zuckerberg Initiative

³ University of California Los Angeles

lixiangci8@gmail.com, gully.burns@chanzuckerburg.com, violetpeng@cs.ucla.edu

Abstract

Even for domain experts, it is a non-trivial task to verify a scientific claim by providing supporting or refuting evidence rationales. The situation worsens as misinformation is proliferated on social media or news websites, manually or programmatically, at every moment. As a result, an automatic fact-verification tool becomes crucial for combating the spread of misinformation. In this work, we propose a novel, paragraph-level, multi-task learning model for the SCIFACT task by directly computing a sequence of contextualized sentence embeddings from a BERT model and jointly training the model on rationale selection and stance prediction.

1 Introduction

Many seemingly convincing rumors such as “Most humans only use 10 percent of their brain” are widely spread, but ordinary people are not able to rigorously verify them by searching for scientific literature. In fact, it is not a trivial task to verify a scientific claim by providing supporting or refuting evidence rationales, even for domain experts. The situation worsens as misinformation is proliferated on social media or news websites, manually or programmatically, at every moment. As a result, an automatic fact-verification tool becomes more and more crucial for combating the spread of misinformation.

The existing fact-verification tasks usually consist of three sub-tasks: document retrieval, rationale sentence extraction, and fact-verification. However, due to the nature of scientific literature that requires domain knowledge, it is challenging to collect a large scale scientific fact-verification dataset, and further, to perform fact-verification under a low-resource setting with limited training data. Wadden et al. (2020) collected a scientific claim-verification dataset, SCIFACT, and proposed a scientific claim-verification task: given a scientific claim, find evidence sentences that support or refute the claim in a corpus of scientific paper abstracts. Wadden et al. (2020) also proposed a simple, pipeline-based, sentence-level model, VERISCI, as a baseline solution based on DeYoung et al. (2019).

VERISCI is a pipeline model that runs modules for abstract retrieval, rationale sentence selection, and stance prediction sequentially, and thus the error generated from an upstream module may propagate to the downstream modules. To overcome this drawback, we hypothesize that a module jointly optimized on multiple sub-tasks may mitigate the error-propagation problem to improve the overall performance. In addition, we observe that a complete set of rationale sentences usually contains multiple inter-related sentences from the same paragraph. Therefore, we propose a novel, paragraph-level, multi-task learning model for the SCIFACT task.

In this work, we employ *compact paragraph encoding*, a novel strategy of computing sentence representations using BERT-family models. We directly feed an entire paragraph as a single sequence to BERT, so that the encoded sentence representations are already contextualized on the neighbor sentences by taking advantage of the attention mechanisms in BERT. In addition, we jointly train the modules for rationale selection and stance prediction as multi-task learning (Caruana 1997) by leveraging the confidence score of rationale selection as the attention weight of the stance prediction module. Furthermore, we compare two methods of transfer learning that mitigate the low-resource issue: pre-training and domain adaptation (Peng and Dredze 2017). Our experiments show that:

- The *compact paragraph encoding* method is beneficial over separately computing sentence embeddings.
- With negative sampling, the joint training of rationale selection and stance prediction is beneficial over the pipeline solution.

2 SCIFACT Task Formulation

Given a scientific claim c and a corpus of scientific paper abstracts A , the SCIFACT (Wadden et al. 2020) task retrieves all abstracts $\hat{E}(c)$ that either SUPPORTS or REFUTES c . Specifically, the *stance prediction* (a.k.a. *label prediction*) task classifies each abstract $a \in A$ into $y(c, a) \in \{\text{SUPPORT}, \text{REFUTES}, \text{NOINFO}\}$ with respect to each claim c ; the *rationale selection* (a.k.a. *sentence selection*) task retrieves all rationale sentences $\hat{S}(c, a) = \{\hat{s}_1(c, a), \dots, \hat{s}_l(c, a)\}$ of each a that SUPPORTS or REFUTES c . The performance of both tasks

* Work performed at Information Sciences Institute, Viterbi School of Engineering, University of Southern California
Copyright © 2021, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

are evaluated with $F1$ measure at both abstract-level and sentence-level, as defined by Wadden et al. (2020), where $\{\text{SUPPORTS}, \text{REFUTES}\}$ are considered as the positive labels and NOINFO is the negative label for *stance prediction*.

3 Approach

We formulate the SCIFACT task (Wadden et al. 2020) as a sentence-level sequence-tagging problem. We first apply an *abstract retrieval* module to filter out negative candidate abstracts that do not contain sufficient information with respect to each given claim. Then we propose a novel model for joint *rationale selection* and *stance prediction* using multi-task learning (Caruana 1997).

3.1 Abstract Retrieval

In contrast to the TF-IDF similarity used by Wadden et al. (2020), we leverage BioSentVec (Chen, Peng, and Lu 2019) embedding, which is the biomedical version of Sent2Vec (Pagliardini, Gupta, and Jaggi 2018), for a fast and scalable sentence-level similarity computation. We first compute the BioSentVec (Chen, Peng, and Lu 2019) embedding of each abstract in the corpus by treating the concatenation of each title and abstract as a single sentence. Then for each given claim, we compute the cosine similarities of the claim embedding against the pre-computed abstract embeddings, and choose the top $k_{\text{retrieval}}$ similar abstracts as the candidate abstracts for the next module.

3.2 Joint Rationale Selection and Stance Prediction Model

Compact Paragraph Encoding A major usage of BERT-family models (Devlin et al. 2018; Liu et al. 2019) for sentence-level sequence tagging computes each sentence embedding in a paragraph with batches. Since each batch is independent, such method leaves the contextualization of the sentences to the subsequent modules. Instead, we propose a novel method of encoding paragraphs by directly feeding the concatenation of the claim c and the whole paragraph P to a BERT model $BERT$ as a single sequence Seq . By separating each sentence s using the BERT model’s $[SEP]$ token, we fully leverage the multi-head attention (Vaswani et al. 2017) within the BERT model to compute the contextualized word representations h_{Seq} with respect to the claim sentence and the whole paragraph.

$$\begin{aligned} c &= [cw_1, cw_2, \dots, cw_n] \\ s_i &= [w_1, w_2, \dots, w_m] \\ P &= [s_1, s_2, \dots, s_l] \\ Seq &= [c[SEP]s_1[SEP]s_2[SEP]\dots[SEP]s_l] \quad (1) \\ h_{Seq} &= BERT(Seq) \in R^{\text{len}(Seq) \times d_{BERT}} \\ h_{Seq} &= [h_{CLS}, h_{cw_1}, \dots, h_{cw_n}, \\ &\quad h_{SEP}, h_{w_1}, \dots, h_{w_m}, h_{SEP}, \dots] \end{aligned}$$

Sentence Representations via Word-level Attention Next, we apply a weighted sum to the contextualized word

representations of each sentence h_{sent} to compute the sentence representations h_s . The weights are obtained by applying a self-attention $SelfAttn_{word}$ with a two-layer multi-layer perceptron on the word representations in the scope of each sentence, as separated by the $[SEP]$ tokens.

$$h_s = SelfAttn_{word}([h_{SEP}, h_{w1}, \dots, h_{wm}]) \in R^d \quad (2)$$

Dynamic Rationale Representations We use a two-layer multi-layer perceptron $MLP_{rationale}$ to compute the rationale score and use the *softmax* function to compute the probability of each candidate sentence being a rationale sentence p^r or not p^{not_r} with respect to the claim sentence c . Then we only feed rationale sentences r into the next stance prediction module.

$$\begin{aligned} p_i^{not_r}, p_i^r &= softmax(MLP_{rationale}(h_{s_i})) \in (0, 1) \\ h_{r_i} &\leftarrow h_{s_i} \text{ if } p_i^{not_r} < p_i^r \end{aligned} \quad (3)$$

Stance Prediction We use two variants for *stance prediction*: a simple sentence-level attention and the Kernel Graph Attention Network (KGAT) (Liu et al. 2020).

- **Simple Attention.** We apply another weighted summation on the predicted rationale sentence representations h_r to compute the whole paragraph’s rationale representation, where the attention weights are obtained by applying another self-attention $SelfAttn_{sentence}$ on the rationale sentence representations h_r . Finally, we apply another two-layer multi-layer perceptron MLP_{stance} and the *softmax* function to compute the probability of the paragraph serving the role of $\{\text{SUPPORTS}, \text{REFUTES}, \text{NOINFO}\}$ with respect to the claim c .

$$\begin{aligned} h_r &= SelfAttn_{sentence}([h_{r1}, h_{r2}, \dots, h_{rl}]) \in R^d \\ p^{stance} &= softmax(MLP_{stance}(h_r)) \in (0, 1)^3 \end{aligned} \quad (4)$$

- **Kernel Graph Attention Network.** Liu et al. (2020) proposed KGAT as a *stance prediction* module for their pipeline solution on the FEVER (Thorne et al. 2018) task. In addition to the Graph Attention Network (Veličković et al. 2017), which applies attention mechanisms on each word pair and sentence pair in the input paragraph, KGAT applies a kernel pooling mechanism (Xiong et al. 2017) to extract better features for *stance prediction*. We integrate KGAT (Liu et al. 2020) into our multi-task learning model for *stance prediction* on SCIFACT (Wadden et al. 2020). The KGAT module $KGAT$ takes the word representation of the claim h_c and the predicted rationale sentence representations h_R as inputs, and outputs the probability of the paragraph serving the role of $\{\text{SUPPORTS}, \text{REFUTES}, \text{NOINFO}\}$ with respect to the claim c .

$$\begin{aligned} h_c &= [h_{CLS}, h_{cw_1}, \dots, h_{cw_n}] \\ h_{Ri} &= [h_{SEP}, h_{rw_1}, \dots, h_{rw_m}] \text{ where } p_i^{not_r} < p_i^r \\ h_R &= [h_{R1}, h_{R2}, \dots, h_{Rl}] \\ p^{stance} &= KGAT(h_c, h_R) \in (0, 1)^3 \end{aligned} \quad (5)$$

3.3 Model Training

Multi-task Learning We train our model on *rationale selection* and *stance prediction* using multi-task learning approach (Caruana 1997). We use cross-entropy loss as the training objective for both tasks. We introduce a coefficient γ to adjust the proportion of two loss values $L_{rationale}$ and L_{stance} in the joint loss L .

$$L = \gamma L_{rationale} + L_{stance} \quad (6)$$

Scheduled Sampling Because the *stance prediction* module takes the predicted rationale sentences as the input, errors in rationale selection may propagate to the *stance prediction* module, especially during the early stage of training. To mitigate this issue, we apply scheduled sampling (Bengio et al. 2015), which starts by feeding the *ground truth* rationale sentences to the stance prediction module, and gradually increasing the proportion of the predicted rationale sentences, until eventually all input sentences are the predicted rationale sentences. We use a *sin* function to compute the probability of sampling predicted rationale sentences p_{sample} as a function of the progress of the training:

$$\begin{aligned} progress &= \frac{current_epoch - 1}{total_epoch - 1} \\ p_{sample} &= \sin\left(\frac{\pi}{2} \times progress\right) \end{aligned} \quad (7)$$

Negative Sampling and Down-sampling Although the *abstract retrieval* module filters out the majority of the negative candidate abstracts, the false-positive rate is still inevitably high, in order to ensure the retrieval of most of the positive abstracts. As a result, the input to the joint prediction model is highly biased towards negative samples. Therefore, in addition to the positive samples from the SCIFACT dataset (Wadden et al. 2020), we perform negative sampling (Mikolov et al. 2013) to sample the top k_{train} similar negative abstracts using our *abstract retrieval* module as an augmented dataset for training and validation to increase the downstream model’s tolerance to false positive abstracts. Furthermore, in order to increase the diversity of the dataset, we augment the dataset by down-sampling sentences within each paragraph.

FEVER Pre-training As Wadden et al. (2020) proposed, due to the similar task structure of FEVER (Thorne et al. 2018) and SCIFACT (Wadden et al. 2020), we first pre-train our model on the FEVER dataset, then fine-tune on the SCIFACT dataset by partially re-initializing the *rationale selection* and *stance prediction* attention modules.

Domain Adaptation Instead of pre-training, we also explore domain adaptation (Peng and Dredze 2017) from FEVER (Thorne et al. 2018) to SCIFACT (Wadden et al. 2020). We use shared representations for the *compact paragraph encoding* and word-level attention, while using domain-specific representations for the *rationale selection* and *stance prediction* modules.

Parameter	Explored	Used
$k_{retrieval}$	3 ~ 100	30
k_{FEVER}	1 ~ 15	5
k_{train}	0 ~ 50	12
γ	0.1 ~ 10	6
drop out	0 ~ 0.6	0
learning rate	$1 \times 10^{-6} \sim 1 \times 10^{-4}$	5×10^{-6}
BERT learning rate	$1 \times 10^{-6} \sim 1 \times 10^{-4}$	1×10^{-5}
batch size	1, 2	1

Table 1: Hyper-parameters explored and used.

$k_{retrieval}$	TF-IDF			BioSentVec		
	P	R	F_1	P	R	F_1
3	16.2	69.9	26.3	15.6	67.0	25.3
10	5.83	83.6	10.9	5.86	84.2	11.0
100	0.67	96.7	1.33	0.68	98.1	1.35
150	0.45	96.7	0.90	0.46	98.1	0.92

Table 2: Abstract retrieval performance on dev set in %.

3.4 Implementation Details

BERT Encoding. We follow Wadden et al. (2020) in using Roberta-large (Liu et al. 2019) as our BERT-family model.

Dummy Rationale Sentence. We dynamically feed only the predicted rationale sentence representations to the *stance prediction* module. To address the special case when an abstract contains no rationale sentences, we append a fixed dummy sentence (e.g. “@”) whose rationale label is always 0 at the beginning of each of the paragraph. When the *stance prediction* module has no actual rationale sentence to take as input, we feed it with the representation of the dummy sentence and expect the module to predict NOINFO.

Post Processing. To prevent inconsistency between the outputs of rationale selection and stance prediction, we enforce the predicted stance to be NOINFO if no rationale sentence is proposed.

Hyper-parameters. Table 1 lists the hyper-parameters used for training the Joint-Paragraph model in Table 4¹, where k_{FEVER} refers to the number of negative samples retrieved from FEVER (Thorne et al. 2018) for model pre-training.

4 Experiments

4.1 SCIFACT Dataset

SCIFACT (Wadden et al. 2020) is a small dataset, whose corpus contains 5183 abstracts. There are 1409 claims, including 809 in the training set, 300 in the development set and 300 in the test set.

¹<https://github.com/jacklxc/ParagraphJointModel>

Models	Sentence-level						Abstract-level					
	Selection-Only			Selection+Label			Label-Only			Label+Rationale		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
VERISCI	77.7	56.3	65.3	69.8	50.5	58.6	89.6	66.0	76.0	84.4	62.2	71.6
Paragraph-Pipeline	74.9	67.7	71.2	65.6	59.3	62.3	81.6	72.2	76.6	76.8	67.9	72.1
Paragraph-Joint	75.9	62.8	68.8	67.7	56.0	61.3	83.3	76.6	79.8	76.6	70.3	73.3
Paragraph-Joint KGAT	75.5	68.3	71.7	66.2	59.8	62.8	81.8	75.1	78.3	76.5	70.3	73.3
VERT5ERINI*	83.5	72.1	77.4	78.2	67.5	72.4	92.7	79.0	85.3	88.8	75.6	81.7

Table 3: Model performance on dev set oracle abstracts in %. The model with * is only for reference.

Models	Sentence-level						Abstract-level					
	Selection-Only			Selection+Label			Label-Only			Label+Rationale		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
VERISCI	54.3	43.4	48.3	48.5	38.8	43.1	56.4	48.3	52.1	54.2	46.4	50.0
Paragraph-Pipeline	71.2	51.4	59.7	62.1	44.8	52.1	77.6	54.5	64.0	72.8	51.2	60.1
Paragraph-Joint	74.2	57.4	64.7	63.3	48.9	55.2	71.4	59.8	65.1	65.7	55.0	59.9
Paragraph-Joint SciFact-only	69.3	50.0	58.1	59.8	43.2	50.2	69.9	52.1	59.7	64.7	48.3	55.3
Paragraph-Joint TF-IDF	72.5	55.7	63.1	62.2	47.8	54.1	70.6	59.8	64.7	65.5	55.5	60.1
Paragraph-Joint DA	69.4	56.6	62.3	60.4	49.2	54.2	67.4	57.4	62.0	63.5	54.1	58.4
Paragraph-Joint KGAT	68.8	56.6	62.1	60.1	49.5	54.3	68.2	62.7	65.3	61.5	56.5	58.9
Paragraph-Joint KGAT DA	70.2	55.5	62.0	61.9	48.9	54.7	70.9	59.3	64.6	66.3	55.5	60.4
VERT5ERINI (BM25)*	67.7	53.8	60.0	63.9	50.8	56.6	70.9	61.7	66.0	67.0	58.4	62.4
VERT5ERINI (T5)*	64.8	57.4	60.9	60.8	53.8	57.1	65.1	65.1	65.1	61.7	61.7	61.7

Table 4: Model performance on dev set with abstract-retrieval in %. The models with * are only for reference.

4.2 Abstract Retrieval Performance

Table 2 compares the performance of *abstract retrieval* modules using using TF-IDF and BioSentVec (Chen, Peng, and Lu 2019). As Table 2 indicates, the overall difference between these two methods is small. Wadden et al. (2020) chose $k_{retrieval} = 3$ to maximize the F_1 score of the abstract retrieval module, while we choose a larger $k_{retrieval}$ to pursue a larger recall score, in order to retrieve more positive abstracts for the down-stream models.

4.3 Baseline Models

VERISCI. Along with the SCIFACT task and dataset, Wadden et al. (2020) proposed VERISCI, a sentence-level, pipeline-based solution. After retrieving the top similar abstracts for each claim with TF-IDF vectorization method, they applied a sentence-level “BERT to BERT” model DeYoung et al. (2019) to extract rationales, sentence by sentence, with a BERT model, and they predict the stance with another BERT model using the concatenation of the extracted rationale sentences. Wadden et al. (2020) used Roberta-large (Liu et al. 2019) as their BERT model and pre-trained their *stance prediction* module on the FEVER dataset (Thorne et al. 2018).

VERT5ERINI. Very recently, Pradeep et al. (2020) proposed a strong model VERT5ERINI, based on T5 (Raffel et al. 2019). They applied T5 for all three steps of the SCIFACT task in a sentence-level, pipeline fashion. Because

of the known significant performance gap between Roberta-large (Liu et al. 2019) that we use and T5 (Raffel et al. 2019; Pradeep et al. 2020), we only use VERT5ERINI as a reference (marked with *).

4.4 Model Performances and Ablation Studies

We experiment on the oracle task, which performs *rationale selection* and *stance prediction* given the oracle abstracts (Table 3), and the open task, which performs the full task of *abstract retrieval*, *rationale selection*, and *stance prediction* (Table 4). We tune our models based on the sentence-level, final development set performance (Selection+Label). The test labels are not released by Wadden et al. (2020). Unless explicitly stated, all models are pre-trained on FEVER (Thorne et al. 2018).

Paragraph-level Model vs. Sentence-level Model. We compare our paragraph-level pipeline model against VERISCI (Wadden et al. 2020), which is a sentence-level solution on the oracle task. As Table 3 shows, our paragraph-level pipeline model (Paragraph-Pipeline) outperforms VERISCI, particularly on *rationale selection*. This suggests the benefit of computing the contextualized sentence representations using the *compact paragraph encoding* over individual sentence representations.

Joint Model vs. Pipeline Model. Although our joint model does not show benefits over the pipeline model

on the oracle task (Table 3), the benefit emerges on the open task. Along with negative sampling, which greatly increases the tolerance of models to false positive abstracts, the Paragraph-Joint model shows its benefit over the Paragraph-Pipeline model. The small difference between the Paragraph-Joint model and the same model except with TF-IDF *abstract retrieval* (Paragraph-Joint TF-IDF) shows that the performance improvement is mainly attributed to the joint training, instead of replacing TF-IDF similarity with BioSentVec embedding similarity in *abstract retrieval*.

Pre-training vs. Domain Adaptation. We also compare two methods of transfer learning from FEVER (Thorne et al. 2018) to SCIFACT (Wadden et al. 2020). Table 4 shows that the effect of pre-training (Paragraph-Joint) or domain adaptation (Peng and Dredze 2017) (Paragraph-Joint DA) is similar. Both of them are effective as transfer learning, as they significantly outperform the same model that is only trained on SCIFACT (Paragraph-Joint SCIFACT-only).

KGAT vs. Simple Attention as *Stance Prediction* Module. We expected a significant performance improvement by applying the strong stance prediction model KGAT (Liu et al. 2020), but the actual improvement is limited. This is likely due to the strong regularization of KGAT that under-fits the training data.

Test-set Performance on the SCIFACT Leaderboard By the time this paper is updated, our Paragraph-Joint model trained on the combination of SCIFACT training set and development set achieved the first place on the SCIFACT leaderboard². We obtain test sentence-level F_1 score (Selection+Label) of 60.9% and test abstract-level F_1 score (Label+Rationale) of 67.2%.

5 Related Work

Fact-verification has been widely studied. There are many datasets available on various domains (Vlachos and Riedel 2014; Ferreira and Vlachos 2016; Popat et al. 2017; Wang 2017; Derczynski et al. 2017; Popat et al. 2017; Atanasova 2018; Baly et al. 2018; Chen et al. 2019; Hanselowski et al. 2019), among which the most influential one is FEVER shared task (Thorne et al. 2018), which aims to develop systems to check the veracity of human-generated claims by extracting evidences from Wikipedia. Most existing systems (Nie, Chen, and Bansal 2019) leverages a three-step pipeline approach by building modules for each of the step: document retrieval, rationale selection and fact verification. Many of them focus on the claim verification step (Zhou et al. 2019; Liu et al. 2020), such as KGAT (Liu et al. 2020), one of the top models on FEVER leaderboard. On the other hand, there are some attempts on jointly optimizing rationale selection and stance prediction. TwoWingOS (Yin and Roth 2018) leverages attentive CNN (Yin and Schütze 2018) to inter-wire two modules,

while Hidey et al. (2020) used a single pointer network (Vinyals, Fortunato, and Jaitly 2015) for both sub-tasks. We propose another variation that directly links two modules by a dynamic attention mechanism.

Because SCIFACT (Wadden et al. 2020) is a scientific version of FEVER (Thorne et al. 2018), systems designed for FEVER can be applied to SCIFACT in principle. However, as a fact-verification task in scientific domain, SCIFACT task has inherited the common issue of lacking sufficient data, which can be mitigated with transfer learning by leveraging language models and introducing external dataset. The base-line model by Wadden et al. (2020) leverages Roberta-large (Liu et al. 2019) fine-tuned on FEVER dataset (Thorne et al. 2018), while VERT5ERINI (Pradeep et al. 2020) leverages T5 (Raffel et al. 2019) and fine-tuned on MS MARCO dataset (Bajaj et al. 2016). In this work, in addition to fine-tuning Roberta-large on FEVER, we also explore domain adaptation (Peng and Dredze 2017) to mitigate the low resource issue.

6 Conclusion

In this work, we propose a novel paragraph-level multi-task learning model for SCIFACT task. Experiments show that (1) The compact paragraph encoding method is beneficial over separately computing sentence embeddings. (2) With negative sampling, the joint training of *rationale selection* and *stance prediction* is beneficial over the pipeline solution.

Acknowledgement

We thank the anonymous reviewers for their useful comments, and Dr. Jessica Ouyang for her feedback. This work is supported by a National Institutes of Health (NIH) R01 grant (LM012592). The views and conclusions of this paper are those of the authors and do not reflect the official policy or position of NIH.

²<https://scifact.apps.allenai.org/leaderboard>, as of January 23, 2021.

References

- Atanasova, P. 2018. Lluís Marquez, Alberto Barrón-Cedeno, 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 *Working Notes of the Conference and Labs of the Evaluation Forum, CLEF*, volume 18.
- Bajaj, P.; Campos, D.; Craswell, N.; Deng, L.; Gao, J.; Liu, X.; Majumder, R.; McNamara, A.; Mitra, B.; Nguyen, T.; et al. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.
- Baly, R.; Mohtarami, M.; Glass, J.; Márquez, L.; Moschitti, A.; and Nakov, P. 2018. Integrating stance detection and fact checking in a unified corpus. *arXiv preprint arXiv:1804.08012*.
- Bengio, S.; Vinyals, O.; Jaitly, N.; and Shazeer, N. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, 1171–1179.
- Caruana, R. 1997. Multitask learning. *Machine learning* 28(1): 41–75.
- Chen, Q.; Peng, Y.; and Lu, Z. 2019. BioSentVec: creating sentence embeddings for biomedical texts. In *2019 IEEE International Conference on Healthcare Informatics (ICHI)*, 1–5. IEEE.
- Chen, S.; Khashabi, D.; Yin, W.; Callison-Burch, C.; and Roth, D. 2019. Seeing things from a different angle: Discovering diverse perspectives about claims. *arXiv preprint arXiv:1906.03538*.
- Derczynski, L.; Bontcheva, K.; Liakata, M.; Procter, R.; Hoi, G. W. S.; and Zubiaga, A. 2017. SemEval-2017 Task 8: RumourEval: Determining rumour veracity and support for rumours. *arXiv preprint arXiv:1704.05972*.
- Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- DeYoung, J.; Jain, S.; Rajani, N. F.; Lehman, E.; Xiong, C.; Socher, R.; and Wallace, B. C. 2019. Eraser: A benchmark to evaluate rationalized nlp models. *arXiv preprint arXiv:1911.03429*.
- Ferreira, W.; and Vlachos, A. 2016. Emergent: a novel data-set for stance classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*, 1163–1168.
- Hanselowski, A.; Stab, C.; Schulz, C.; Li, Z.; and Gurevych, I. 2019. A Richly Annotated Corpus for Different Tasks in Automated Fact-Checking. *arXiv preprint arXiv:1911.01214*.
- Hidey, C.; Chakrabarty, T.; Alhindi, T.; Varia, S.; Krstovski, K.; Diab, M.; and Muresan, S. 2020. DeSePtion: Dual Sequence Prediction and Adversarial Examples for Improved Fact-Checking. *arXiv preprint arXiv:2004.12864*.
- Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- Liu, Z.; Xiong, C.; Sun, M.; and Liu, Z. 2020. Fine-grained fact verification with kernel graph attention network. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7342–7351.
- Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, 3111–3119.
- Nie, Y.; Chen, H.; and Bansal, M. 2019. Combining fact extraction and verification with neural semantic matching networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 6859–6866.
- Pagliardini, M.; Gupta, P.; and Jaggi, M. 2018. Unsupervised Learning of Sentence Embeddings using Compositional n-Gram Features. In *NAACL 2018 - Conference of the North American Chapter of the Association for Computational Linguistics*.
- Peng, N.; and Dredze, M. 2017. Multi-task multi-domain representation learning for sequence tagging. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*.
- Popat, K.; Mukherjee, S.; Strötgen, J.; and Weikum, G. 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*, 1003–1012.
- Pradeep, R.; Ma, X.; Nogueira, R.; and Lin, J. 2020. Scientific Claim Verification with VERT5ERINI. *arXiv preprint arXiv:2010.11930*.
- Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.
- Thorne, J.; Vlachos, A.; Christodoulopoulos, C.; and Mittal, A. 2018. FEVER: a large-scale dataset for fact extraction and verification. *arXiv preprint arXiv:1803.05355*.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.
- Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.
- Vinyals, O.; Fortunato, M.; and Jaitly, N. 2015. Pointer networks. In *Advances in neural information processing systems*, 2692–2700.
- Vlachos, A.; and Riedel, S. 2014. Fact checking: Task definition and dataset construction. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, 18–22.
- Wadden, D.; Lo, K.; Wang, L. L.; Lin, S.; van Zuylen, M.; Cohan, A.; and Hajishirzi, H. 2020. Fact or Fiction: Verifying Scientific Claims. In *EMNLP*.
- Wang, W. Y. 2017. ”liar, liar pants on fire”: A new benchmark dataset for fake news detection. *arXiv preprint arXiv:1705.00648*.
- Xiong, C.; Dai, Z.; Callan, J.; Liu, Z.; and Power, R. 2017. End-to-end neural ad-hoc ranking with kernel pooling. In *Proceedings of the 40th International ACM SIGIR conference on research and development in information retrieval*, 55–64.
- Yin, W.; and Roth, D. 2018. Twowingos: A two-wing optimization strategy for evidential claim verification. *arXiv preprint arXiv:1808.03465*.
- Yin, W.; and Schütze, H. 2018. Attentive convolution: Equipping cnns with rnn-style attention mechanisms. *Transactions of the Association for Computational Linguistics* 6: 687–702.
- Zhou, J.; Han, X.; Yang, C.; Liu, Z.; Wang, L.; Li, C.; and Sun, M. 2019. GEAR: Graph-based evidence aggregating and reasoning for fact verification. *arXiv preprint arXiv:1908.01843*.