Automatic Detection of Entity-Manipulated Text Using Factual Knowledge

Ganesh Jawahar[†,‡] **Muhammad Abdul-Mageed**[†] **Laks V. S. Lakshmanan**[†,‡]
[†]Deep Learning & Natural Language Processing Group, [‡]Data Management & Mining Group
The University of British Columbia
ganeshjwhr@gmail.com,{laks,amuham01}@cs.ubc.ca

Abstract

In this work, we focus on the problem of distinguishing a human written news article from a news article that is created by manipulating entities in a human written news article (e.g., replacing entities with factually incorrect entities). Such manipulated articles can mislead the reader by posing as a human written news article. We propose a neural network based detector that detects manipulated news articles by reasoning about the facts mentioned in the article. Our proposed detector exploits factual knowledge via graph convolutional neural network along with the textual information in the news article. We also create challenging datasets for this task by considering various strategies to generate the new replacement entity (e.g., entity generation from GPT-2). In all the settings, our proposed model either matches or outperforms the state-of-the-art detector in terms of accuracy. Our code and data are available at https://github.com/UBC-NLP/manipulated_entity_detection.

1 Introduction

A type of fake news that has received little attention in the research community is manipulated text. Manipulated text is typically created by manipulating a human written news article minimally (e.g., replacing every occurrence of a particular entity, 'Obama' in a news article with another American politician entity). Current fake news detectors that exploit stylometric signals from the text (e.g., choice of specific words to express false statements) are clearly insufficient for distinguishing manipulated text from human written text (Zhou et al., 2019; Schuster et al., 2020) as the style underlying the manipulated text is virtually identical to human writing style. In this work, we focus on this problem of distinguishing manipulated news articles from human written news articles.

Human written text
PubNub, a startup that develops the infrastructure to power key features in real-time applications (...) has raised $23 million in a series D round of funding from **Hewlett Packard Enterprise (HPE)**, **Relay Ventures**, **Sapphire Ventures**, **Scale Venture Partners**, **Cisco Investments**, **Bosch**, and **Ericsson**.
Manipulated text using GPT-2
PubNub, a startup that develops the infrastructure to power key features in real-time applications (...) has raised $23 million in a series D round of funding from **Hewlett Packard Enterprise (HPE)**, **Samsung**, **Sapphire Ventures**, **Scale Venture Partners**, **Cisco Investments**, **Bosch**, and **Ericsson**.

Table 1: Example human written and manipulated text. Named entities of organization type are shown in **green**. Manipulated entities are shown in **orange**.

We consider a particular type of text manipulation — entity perturbation (Zhou et al., 2019), where a manipulated news article is created by modifying a fixed number of entities in a human written news article (e.g., replacing them with entities generated from a text generative model). E.g., in Table 1, to mislead humans, the entity 'Relay Ventures' can be replaced by 'Samsung' (a candidate replacement entity generated by the generative pre-training-2 model (GPT-2) (Radford et al., 2019)), which is locally consistent as some of the other companies in the original text are also into device manufacturing.

To distinguish a manipulated news article from the original human written news article, we propose a neural network based detector that jointly utilizes the textual information along with the the factual knowledge explicitly by building entity-relation graphs which capture the relationship between different entities present in the news article. The factual knowledge is encoded by a graph convolutional neural network (Kipf and Welling, 2017) that captures the interactions between different entities and relations, which we hypothesize, carries discriminatory signals for the manipulated text detection task.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 2: Short Papers, pages 86 - 93

Our major contributions include: (i) a detector that exploits factual knowledge to overcome the limitations of relying only on stylometric signals, (ii) an approach to generate challenging manipulated news article dataset using GPT-2, and (iii) a collection of challenging datasets by considering various strategies to generate the replacement entity.

2 Background and Related Work

The manipulated text detection task is related to diverse research areas such as fake news detection, natural language understanding, and knowledge bases.

Fake news detection. Research on Fake news detection typically deals with challenges such as understanding the news content (Schuster et al., 2020), claim verification (Thorne and Vlachos, 2018), verifying the credibility of the source (Castillo et al., 2011), and exploiting fake news propagation patterns (Vosoughi et al., 2018). Our work is primarily focused on detecting fake news in the form of manipulated text, by understanding the news content. In the traditional problem setting, both fake and real news is assumed to be written by a human (Shu et al., 2017; Oshikawa et al., 2020). Since humans tend to make stylistic choices (e.g., choosing some specific language for writing false statements), the fake news detector can perform reasonably on the task by picking up on these stylometric signals. One can also create fake news by manipulating a human written news article minimally. Such manipulations include: entity perturbation (e.g., '12 people were injured in the shooting' to '24 people were killed in the shooting') (Zhou et al., 2019), subject-object exchange (e.g., 'A gangster was shot by the police' to 'A policeman was shot by the gangster') (Zhou et al., 2019), and adding/deleting negations (e.g., 'Trump doesn't like Obamacare' to 'Trump likes Obamacare') (Schuster et al., 2020). These manipulations do not typically affect the style and hence stylometric signals alone cannot help in building accurate manipulated text detection models (Zhou et al., 2019; Schuster et al., 2020).

Natural language understanding. Pre-trained language models such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) achieve strong performance in diverse NLP tasks. Specifically, RoBERTa is the state-of-the-art detector when fine-tuned for detection of synthetic text (Solaiman et al., 2019; Jawahar et al., 2020). These models can also capture implicit world knowledge (e.g., Paris is the capital of France) that occurs frequently in the text (Petroni et al., 2019). However, it is insufficient for solving our task (Schuster et al., 2020), as it is limited to frequent patterns.

Knowledge bases (KBs). Knowledge bases (e.g., YAGO (Tanon et al., 2020)) containing typically a collection of facts (e.g., subject-relation-object triples), provide specialized knowledge for downstream NLP tasks (e.g., question answering (Banerjee and Baral, 2020)). One can integrate such symbolic knowledge into pre-trained language models during pre-training (Zhang et al., 2019) and finetuning (Liu et al. (2020); Zhong et al. (2020), which we follow in this work).

3 Manipulated Text Creation

In this work, we focus on a particular type of manipulation — entity perturbation (Zhou et al., 2019), where all occurrences of a fixed number of randomly picked entities from a human written news article are replaced with different replacement entities. We replace named entities of three types: person, organization and location (recognized using spaCy's named entity recognizer (NER) (Honnibal et al., 2020)). We ensure the replacement (new) entity belongs to the same type as the original (old) entity. We create challenging manipulated text datasets by considering various strategies to identify the new replacement entity: random most frequent entity (pick randomly from among the top 5000 entities), random least frequent entity (pick randomly from the bottom 5000 entities), and entity generated by GPT-2. Sample manipulated entities obtained from different replacement strategies are shown in Table 2.

Entity replacement strategy		
Random least	**Random most**	**GPT-2 generated**
Inverkeithing High School	Tribune	U.S.
Mark Forman	East Jerusalem	Canada
Netgear	Englishman	Microsoft
Bangalore North	Jason Aldean	Donald Trump
Mackintosh	UFA	BBC

Table 2: Sample manipulated entities

GPT-2 generated entity replacement. Strategies that randomly identify the replacement entity ignore the context provided by the news article. For example, in news portion (1), a random replacement entity for 'Relay Ventures' can be 'Salesforce'. However, it is likely locally inconsistent as 'Salesforce' is not into device manufacturing unlike

many other co-occurring companies in the original text. We propose a novel approach that makes use of the state-of-the-art text generative model GPT-2 to pick replacement entities that are locally consistent. Revisiting the news portion (1), let the randomly selected entity to be replaced be 'Relay Ventures'. We treat the fragment of text from the beginning of the article up to the tokens before the first occurrence of the target entity ('Relay Ventures') as the prompt. We provide this prompt to GPT-2, which can then generate the next few tokens. We call the generated token sequence a candidate replacement entity if the sequence starts with an entity (e.g., 'Samsung') of same type as the target entity ('Relay Ventures') and has no string overlap with the target entity. If the constraints are not met, we ask GPT-2 to create the generated sequence again up to a maximum of 10 attempts. The candidate replacement entity thus obtained will be used to replace all occurrences of the target entity. For the news portion (1), the candidate replacement entity generated by GPT-2 is 'Samsung', which is locally consistent: similar to other companies in the original text, Samsung manufactures devices.

4 Manipulated Text Detection

The goal of this work is to build a detector that distinguishes manipulated news article from human written news article with high accuracy. In prior work, Zhou et al. (2019) conclude that the manipulated article can possibly be detected by checking the facts underlying the article with knowledge bases and Schuster et al. (2020) show that humans can identify the manipulated text well when they are allowed to consult external sources (e.g., internet). Building on these findings, we hypothesize that *factual knowledge underlying the news article can provide discriminatory signals for manipulated text detection.* To this end, we embody the RoBERTa detector with explicit factual knowledge so that the detector can reason about facts present in the news article, whose details we discuss next.

Factual knowledge. For factual knowledge, we leverage a variant of YAGO 4 KB (Tanon et al., 2020) that contains only instances that have an English Wikipedia article. We then extract the facts in a given document by first identifying all the entities present in the document using spaCy's NER. For each target entity, we grab all the triples in the KB where the subject matches with the target entity at surface level. These triples can be seen as the first hop neighbors of the target entity in the KB. For a given document, the set of triples collected over all identified entities is used to build the corresponding factual graph. A node can be an entity or a relation. A directed edge is added between subject and relation, as well as relation and object. This factual graph contains rich factual information about entities present in the document, which can be exploited to reason about facts mentioned in the article for correctness.

Integrating factual knowledge with RoBERTa. Our proposed detector is an integration of the RoBERTa model with factual knowledge. This allows the detector to reason about facts mentioned in the article. To embed the factual knowledge, we employ graph convolutional networks (GCNs) (Kipf and Welling, 2017), where we stack l GCN layers and the definition of the hidden representation of each node v of the factual graph as layer $k+1$, in a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$:

$$\mathbf{h}_v^{k+1} = f\left(\frac{1}{|\mathcal{N}(v)|}\sum_{u\in\mathcal{N}(v)} \mathcal{W}^k h_u^k + b^k\right), \quad \forall v \in \mathcal{V}, \tag{1}$$

where $\mathcal{W}^k$, b^k, h_u^k, $\mathcal{N}(v)$ correspond to layer specific model weights, biases, node representation, and neighbors of v in $\mathcal{G}$ respectively. Note that h_u^1 denotes the initial node features, which can be initialized randomly or using a pre-trained entity embedding such as Wikipedia2vec (Yamada and Shindo, 2019).

Detector prediction. The factual knowledge about entities present in the article is captured in the node embeddings (h_u^l) corresponding to the last layer l of the GCN model. The textual knowledge corresponding to the document can be obtained from the last layer representation (r_{CLS}^d) of the RoBERTa model corresponding to the first token ('[CLS]', special classification token) of the RoBERTa input. We combine the factual and the textual knowledge by simply averaging all the GCN's entity embeddings and concatenating the entity average with the RoBERTa's document embedding. Thus, the unnormalized prediction probabilities ($mf(d)$) of our detector for the document d can be given by:

$$\mathbf{mf}(d) = \mathcal{W}_{mtd}\left[r_{[CLS]}^d; \sum_{e\in entities(d)} h_e^l\right] + b_{mtd}, \tag{2}$$

where $[;]$ corresponds to the concatenation operation and $\mathcal{W}_{mtd}$, b_{mtd} correspond to the affine transformation specific model parameters for manipu-

Entity replacement strategy	**Random least frequent entity replacement**			**Random most frequent entity replacement**			**GPT-2 generated entity replacement**		
Maximum no. of entity replacements	1	2	3	1	2	3	1	2	3
Manipulated Article Detection Task									
(1) Overall Accuracy									
RoBERTa	67.09	78.37	84.26	65.56	76.86	83.93	67.09	74.12	78.79
Ours (w/o Entity Identification Objective)	67.25	78.36	**84.59**	66.99*	77.98*	83.86	**67.16**	73.84	**79.11**
Ours	**68.25***	**78.99**	83.84	**67.21***	**78.26***	**84.39**	65.84	**74.80**	**79.05**
Manipulated Entity Identification Task									
(1) Overall Precision - Ours	49.99	50.02	50.08	49.94	50.00	49.83	49.49	48.52	48.71
(2) Overall Recall - Ours	38.56	55.11	65.11	48.20	50.04	47.71	45.82	46.76	45.67
(3) Overall F-Score - Ours	42.29	46.50	46.12	46.07	47.79	46.83	44.82	47.42	44.92
(4) Manipulated Entity - Precision - Ours	81.06	91.76	84.14	84.71	88.06	86.06	85.59	85.91	73.80
(5) Manipulated Entity - Recall - Ours	0.00	3.70	12.12	6.08	4.63	14.03	9.14	1.64	12.50
(6) Manipulated Entity - F-Score - Ours	0.00	7.11	21.19	11.35	8.80	24.13	16.52	3.22	21.38

Table 3: Evaluation performance (%) for different maximum number of entity replacements across different replacement strategies. **Bolded** refers to the best results for each dataset. Note that the state-of-the-art detector cannot identify manipulated entities present in the document. For the manipulated article detection task, statistically significant overall accuracy results obtained using bootstrap test with $p < 0.01$ are marked using asterisk (*).

lated text detection. The output from $mf(d)$ passes through dropout followed by ReLU layer.

Identifying manipulated entities. To enable humans to understand our detector's decision and perform further investigation, we introduce a subtask for the detector, namely identify the manipulated entities among different entities present in the document. For this subtask, we build on the entity representations output by the last layer of the GCN model. The unnormalized class prediction probabilities ($ef(v)$) for a given entity v from the article can be given by:

$$\mathbf{ef}(v) = Dropout\left(ReLU\left(\mathcal{W}_{ec}h_v^l + b_{ec}\right)\right), \quad (3)$$

where h_v^l denotes the hidden representation at last layer l for the entity v, and $\mathcal{W}_{ec}$, b_{ec} correspond to the affine transformation specific model parameters for entity classification. The overall objective function of the proposed detector can be given by:

$$\min_{\theta} \sum_{i=1}^{n} \left[\mathcal{L}(s(mf(x_i)), y_i) + \sum_{e \in entities(x_i)} \mathcal{L}(s(ef(e)), y^e) \right]. \quad (4)$$

where $\mathcal{L}$, mf, and s resp. correspond to the function that computes the negative log-probability of the correct label, detection prediction function, and softmax function. y^e denotes the entity manipulation class label, which is 1 if the entity e is manipulated, and 0 otherwise. y_i denotes the article manipulation class label, which is 1 if at least one entity in article i is manipulated, and 0 otherwise.

5 Experiments and Results

Dataset and Detector Settings. The human written news articles used in our study are taken from the RealNews dataset (Zellers et al., 2019), which contains 5000, 2000, and 8000 news articles in the training, validation, and test set respectively. We randomly pick half of the news articles in each set for human written news article category and the rest in each set for manipulation based on the chosen replacement strategy. We also create three different datasets for each replacement strategy by varying the maximum number of entities to be manipulated from 1 to 3. Detailed statistics of the proposed datasets is in A.1. The hyperparameter search space for all detectors is offered in A.2.

Hardest detection task. Table 3 presents the detection accuracy results. We observe that the most challenging dataset for the state-of-the-art detector is surprisingly from random most frequent entity replacement strategy with exactly one entity replacement. The random strategies fail to create a challenging dataset with high (e.g., 3) number of entity replacements, which indicates that the detection task becomes easier with increase in the number of locally inconsistent entities. Nevertheless, our proposed GPT-2 based entity replacement strategy keeps the detection task harder even for large number of replacements, thanks to the ability of the strategy to generate locally consistent entities mostly. Regardless of the replacement strategies, the detection performance of all the detectors increases with the increase in the number of entities that are manipulated in a document, that is, more the manipulations in a document, the easier the detection task. This result is similar to previous research which performs manipulation by adding/deleting negations in news articles (Schuster et al., 2020). A fake news propagator can thus

Entity replacement strategy	**Random least frequent entity replacement**			**Random most frequent entity replacement**			**GPT-2 generated entity replacement**		
Maximum no. of entity replacements	1	2	3	1	2	3	1	2	3
Test set size (Percent)	3,797 (47.5)	3,625 (45.3)	3,447 (43.1)	3,288 (41.1)	2,660 (33.2)	2,207 (27.6)	3,302 (41.3)	2,737 (34.2)	2,359 (29.5)
RoBERTa	48.17	68.69	77.81	45.62	66.32	74.94	51.97	**66.97**	**74.95**
Ours (w/o Entity Identification Objective)	47.20	65.19	**78.76**	51.55	68.20	**75.44**	56.27	66.68	72.11
Ours	**52.04**	**68.99**	75.98	**54.65**	**68.38**	72.81	**62.11**	66.53	71.22

Table 4: Manipulated article detection performance (%) for different maximum number of entity replacements across different replacement strategies on a subset of our test set. This text subset contains manipulated articles with all the manipulated entities absent in the knowledge base. **Bolded** refers to best results for each dataset.

manipulate exactly one entity in the news article to make the detection task harder.

Detector performance. Nevertheless, our proposed detector performs similarly to or outperforms the state-of-the-art detector on all replacement strategies across different numbers of entity replacements. This result validates our hypothesis that leveraging both factual and textual knowledge can improve detection performance, overcoming the limitations of relying only on textual knowledge. Improvements of our proposed detector on the GPT-2 generated entity manipulation task are not significantly high due to sizeable increase in manipulated entities absent in the knowledge base (e.g., $\sim$50%, see last three rows in Table 6).

Entity identification performance. Our proposed detector is equipped to identify entities that are manipulated in a news article. This task is harder due to the imbalanced nature of the task as most of the entities present in the news article are not manipulated. As shown in Table 3, our proposed detector achieves high precision ($\geq$ 70%) in identifying manipulated entities, which makes our detector appealing for applications that favor precision. The recall is very low ($<$ 15%), which indicates the difficulty of the task. We also experiment with a baseline RoBERTa model trained at the token level to identify spans of manipulated entities. However, the model seems overwhelmed by the majority class (token not part of the manipulated entity span) and predicts all the tokens to belong to the majority class. We believe there is a lot of room for improvement in this subtask.

Detecting articles with unknown manipulated entities. Table 4 shows performance of the detector on manipulated articles when all the manipulated entities are not present in the knowledge base. We observe that our proposed detector can rely on the relations corresponding to the non-manipulated entities and pretrained textual representations to outperform, or at least be on par with, the RoBERTa model.

Repl. strategy / # replacements	1	2	3
Random least frequent	93.67	95.06	95.05
Random most frequent	93.75	93.37	93.79
GPT-2 generated	95.1	93.35	94.88

Table 5: Quality gap - Human vs. Manipulated text

Quality gap between human and manipulated text. Table 5 shows how the quality of the manipulated text changes with respect to human written text across different replacement strategies, for different numbers of replacements. We utilize MAUVE (Pillutla et al., 2021), a metric to measure the closeness of machine generated text to human language based on divergence frontiers. Since the proposed manipulations touch only limited spans (i.e., entities) in the entire document, the overall quality of the manipulated text does not change much with more replacements.

6 Conclusion

We presented the first principled approach for developing a model that can detect entity-manipulated text articles. In addition to textual information, our proposed detector exploits explicit factual knowledge from a knowledge base to overcome the limitations of relying only on stylometric signals. We constructed challenging manipulated datasets by considering various entity replacement strategies, including with random selection and GPT-2 generation. On all the experimental settings, our proposed model outperforms (or is at least on par with) the baseline detector in overall detection accuracy. Our results show that manipulated text detection remains challenging. We hope that our work will trigger further research on this important but relatively understudied subfield of fake news detection.

Acknowledgements

We gratefully acknowledge support from the Natural Sciences and Engineering Research Council of Canada (NSERC; RGPIN-2018-04267), the Social Sciences and Humanities Research Council of Canada (SSHRC; 435-2018-0576), Canadian Foundation for Innovation (CFI; 37771), Compute Canada (CC),[1] UBC ARC-Sockeye,[2] and Advanced Micro Devices, Inc. (AMD). Any opinions, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NSERC, SSHRC, CFI, CC, ARC-Sockeye, or AMD. We also thank Ayushi Dalmia for proofreading and helpful discussions.

References

Pratyay Banerjee and Chitta Baral. 2020. Self-supervised knowledge triplet learning for zero-shot question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 151–162, Online. Association for Computational Linguistics.

Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. 2011. Information credibility on twitter. In *Proceedings of the 20th International Conference on World Wide Web*, WWW '11, page 675–684, New York, NY, USA. Association for Computing Machinery.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Ganesh Jawahar, Muhammad Abdul-Mageed, and Laks Lakshmanan, V.S. 2020. Automatic detection of machine generated text: A critical survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2296–2309, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020. K-BERT: Enabling language representation with knowledge graph. In *Proceedings of AAAI 2020*.

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.

Ray Oshikawa, Jing Qian, and William Yang Wang. 2020. A survey on natural language processing for fake news detection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6086–6093, Marseille, France. European Language Resources Association.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. MAUVE: Measuring the gap between neural text and human text using divergence frontiers. In *Advances in Neural Information Processing Systems*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

Tal Schuster, Roei Schuster, Darsh J. Shah, and Regina Barzilay. 2020. The Limitations of Stylometry for Detecting Machine-Generated Fake News. *Computational Linguistics*.

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

Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, and Jasmine Wang. 2019. Release Strategies and the Social Impacts of Language Models. *CoRR*, abs/1908.09203.

Thomas Pellissier Tanon, Gerhard Weikum, and Fabian M. Suchanek. 2020. YAGO 4: A reasonable knowledge base. In *The Semantic Web*, volume 12123 of *Lecture Notes in Computer Science*, pages 583–596.

[1] https://www.computecanada.ca
[2] https://arc.ubc.ca/ubc-arc-sockeye

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

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

Ikuya Yamada and Hiroyuki Shindo. 2019. Neural attentive bag-of-entities model for text classification. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 563–573. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. In *Advances in Neural Information Processing Systems*, volume 32, pages 9054–9065. Curran Associates, Inc.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

Wanjun Zhong, Duyu Tang, Zenan Xu, Ruize Wang, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. 2020. Neural deepfake detection with factual structure of text. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2461–2470, Online. Association for Computational Linguistics.

Zhixuan Zhou, Huankang Guan, Meghana Moorthy Bhat, and Justin Hsu. 2019. Fake news detection via NLP is vulnerable to adversarial attacks. In *Proceedings of the 11th International Conference on Agents and Artificial Intelligence, ICAART 2019, Volume 2, Prague, Czech Republic, February 19-21, 2019*, pages 794–800.

A Appendices

A.1 Summary Statistics of Proposed Datasets.

Table 6 displays the statistics of proposed datasets.

A.2 Hyperparameter Search Space for All Detectors

Table 7 displays the search space for hyperparameters used to tune all the detectors.

Name	**Random least frequent entity replacement**			**Random most frequent entity replacement**			**GPT-2 generated entity replacement**		
Maximum no. of entity replacements	1	2	3	1	2	3	1	2	3
Dataset Size									
Train	5,000	5,000	5,000	5,000	5,000	5,000	5,000	5,000	5,000
Validation	2,000	2,000	2,000	2,000	2,000	2,000	2,000	2,000	2,000
Test	8,000	8,000	8,000	8,000	8,000	8,000	8,000	8,000	8,000
Average Length (# words)									
Train	604	604	605	603	603	603	603	613	614
Validation	595	595	596	594	594	594	607	598	599
Test	597	597	597	596	596	596	598	598	601
% Documents with Person Entities									
Train	97.92	98.00	97.96	97.74	97.84	98.00	97.22	97.60	97.82
Validation	98.65	98.65	98.85	98.55	98.65	98.50	97.80	98.00	98.30
Test	97.86	98.04	98.16	97.92	97.91	97.95	97.45	97.49	97.76
% Documents with Organization Entities									
Train	99.14	99.12	99.10	99.20	99.26	99.12	99.04	99.10	99.14
Validation	99.35	99.35	99.30	99.35	99.35	99.40	99.20	99.50	99.25
Test	99.28	99.20	99.17	99.24	99.12	99.17	99.06	99.05	99.11
% Documents with Location Entities									
Train	90.44	90.16	89.84	90.70	90.70	91.00	90.70	91.34	91.88
Validation	90.40	89.90	89.75	90.55	90.55	90.80	90.80	91.05	91.90
Test	90.69	90.28	89.91	90.83	90.64	90.66	90.95	91.05	91.62
Average % Entity Coverage by YAGO-4									
Train	9.78	9.63	9.46	9.97	10.01	10.03	10.01	10.03	10.01
Validation	9.80	9.62	9.51	9.98	10.03	10.10	9.68	10.02	10.15
Test	9.85	9.70	9.54	10.05	10.07	10.09	10.05	10.01	10.10
Avg. % Known Ents. post Manipulation									
Train	6.94	9.26	11.07	30.28	26.33	28.35	60.85	54.26	51.83
Validation	11.97	9.07	10.16	26.76	23.89	27.18	48.72	49.26	48.68
Test	7.68	8.99	9.03	26.13	27.15	25.76	48.85	52.72	51.51

Table 6: Summary statistics of proposed datasets.

Hyperparameter Name	**Hyperparameter Values**
RoBERTa model variant	Large
Minimum frequency of node (i.e., entity)	{10}
Batch size	{8}
Initial learning rate	{1e-5, 2e-5, 3e-5}
Epochs	{10}
Number of warmup steps	{10%}
Node intialization	{Wikipedia2vec}
Node embedding size	{100, 300}
Number of GCN layers	{1, 2}

Table 7: Hyperparameter search space for all detectors.