Groningen Team A at SemEval-2024 Task 8: Human/Machine Authorship Attribution Using a Combination of Probabilistic and Linguistic Features

Huseyin Alecakir **Puja Chakraborty** **Pontus Henningsson** **Matthijs van Hofslot** **Alon Scheuer**

Faculty of Arts
University of Groningen

Abstract

The emergence of generative language models has put in place the necessity of building models to discern between machine-generated and human-generated text. In this paper, we present our participation in subtasks A and B of the SemEval 2024 Task 8 shared task, which revolves around this problem. Our approach primarily centers on feature-based systems, where a diverse array of features pertinent to the text's linguistic attributes is extracted. Alongside those, we incorporate token-level probabilistic features which are fed into a Bidirectional Long Short-Term Memory (BiLSTM) model. Both resulting feature arrays are concatenated and fed into our final prediction model. Our method under-performed compared to the baseline, despite the fact that previous attempts by others have successfully used linguistic features for the purpose of discerning machine-generated text. We conclude that our examined subset of linguistically motivated features alongside probabilistic features was not able to contribute almost any performance at all to a hybrid classifier of human and machine texts. Our codebase is publicly available on GitHub.[1]

1 Introduction

Large language models capable of generating human-like text have become quite ubiquitous very quickly. There are now many such models which are commonly used to generate text across different domains and in different languages. With their increasing availability and capabilities, it has subsequently become necessary to find ways to distinguish machine-generated text from that which is produced by humans. Humans alone are not able to detect machine-generated text consistently, not even experts in this task (Guo et al., 2023), and current commercial solutions fall short (Chaka, 2023). It is natural then that this problem has seen wide discussion and participation over several domains and languages, including the creation of datasets and proposal of different feature sets and model types (Shamardina et al. 2022; Wang et al. 2024b to name a few), but it is still far from being solved. This leads us to the SemEval-2024 Shared Task 8 that this paper is concerned with "Multidomain, Multimodal and Multilingual Machine-Generated Text Detection" (Wang et al., 2024a). This task is about distinguishing human-written text from machine-generated text in multiple different domains, modalities and across different languages. The languages included in the task are: Arabic, Bulgarian, Chinese, English, Indonesian, Russian and Urdu. The domains are varied and range from Wikipedia pages to arXiv research papers to Reddit posts.

1.1 Research Question

The task of discerning machine-generated texts can be approached using classical feature-based methods or using recent neural methods. The inclusion of multiple domains, languages, and underlying models adds complexity to the problem, but also demands a more universal solution. We therefore find it important not only to strive for high accuracy, but also for explainability and universality based on linguistic concepts. Our research question thus read as the following:

- How well does the linguistically motivated probabilistic model perform for machine-generated text detection and model authorship attribution?

To answer this question, our main strategy uses a combined linear model with document-level features alongside token-level features which have been processed by a BiLSTM, resulting in a method which combines probability-based features with low-level and high-level linguistic features. Our method is inspired by Przybyła et al. (2023), which

[1] https://github.com/rug-1-at-semeval24-task8/code

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 1926–1932

used a similar model structure for the AuTexTification shared task (Sarvazyan et al., 2023), achieving results that were close behind an LLM-based model. In our method, however, we employ a linear perceptron instead of a random forest classifier to combine the document-level and processed token-level features, in an attempt to enhance the model's performance and learning. A wide range of features is employed, with our system utilizing stylometric features, entity coherence features, information-theoretic features as well as complementary features such as TF-IDF features for word-level unigrams. An LSTM model proved to attain notably high accuracy in our baseline system, which led to us combining our extracted features with a BiLSTM model. The overview of our system is presented in Figure 1. Our system performs poorly in general and relative to other teams, where we rank at the bottom ten for all tasks that we participated in.

While our primary emphasis remained on feature-based models, we developed a separate model to explore potential performance variations compared to the feature-based approach. In this independent model, we employed a basic LSTM architecture with BERT (Devlin et al., 2018) serving as the embedding layer to acquire sentence embeddings. However, the inclusion of the embedding layer introduced computational overhead, resulting in prolonged processing times. Consequently, we were only able to obtain results for Task B on the test dataset using this architecture.

2 Related Work

Due to the similarities in the architecture and training of different text-generation machines, generated text may possess universal characteristics that distinguish it from text written by humans. Guo et al. (2023) set up a series of human evaluation and linguistics analyses to understand the characteristic features and patterns, where a study by Mitrović et al. (2023) looked at the differences in human vs AI-generated text. The studies found that humans tend to have much more diverse and expressive vocabulary, and often tend to diverge from the topic more than ChatGPT does (Guo et al., 2023; Mitrović et al., 2023). This idea is supported by Gehrmann et al. (2019), who performed a probabilistic analysis of the vocabulary in human- and machine-generated texts and found that generation models tend to have a relatively limited and predictable vocabulary. Some work focuses on stylometric features, as these may be productive in discerning the original author of a text (Li et al., 2014; Pearl and Steyvers, 2012). Wang et al. (2024b) show that models based on such feature sets perform strongly within the domain, but the choice of training dataset may have a notable effect on performance.

Feature-based detectors work fairly well for simple binary classifications in a single domain, but tend to fall short when attempting more complex problems which consist of additional styles and sources of texts (Wang et al., 2024b), where shorter texts can have a negative impact on performance Shamardina et al. (2022). Conversely, language models may prove to be the optimal tool for detecting machine-generated text. Recent attempts mostly use (Ro)BERT(a)-based models (Devlin et al., 2018; Liu et al., 2019) that are pre-trained for language understanding, and fine-tune them using datasets of human- and machine-generated text (Zellers et al., 2019; Shamardina et al., 2022; Guo et al., 2023). These models are then able to detect authorship with varying levels of success. Much of the focus in this area has been on developing useful datasets for fine-tuning and finding optimal models and methods of fine-tuning.

An LSTM, as introduced in Hochreiter and Schmidhuber (1997), is a version of a RNN (recurrent neural network) that utilizes *long term short memory* to deal with issues present in regular RNNs caused by larger gap lengths, which can be especially relevant in NLP tasks such as ours. LSTMs have been used with success to perform authorship attribution (Deibel and Löfflad, 2021; Gupta et al., 2019) which suggests they may be useful in distinguishing human and machine authors as well.

3 Shared Task Set Up

The SemEval-2024 shared task 8 revolved around distinguishing human-written texts and machine-generated texts. It was divided into multiple subtasks. The goal of subtask A was to perform binary classification on a given text to determine whether it is human-written or machine-generated. The monolingual track of this subtask only included text in English, whereas the multilingual track included text in English, Russian, Chinese, Arabic, Urdu, Indonesian and Bulgarian. Subtask B focused on multi-way machine-generated text, where the goal of the task was to determine whether a

given text is written by a human or generated by a machine, and if generated by a machine – which specific language model was it that generated the text?

For all subtasks, we used the datasets provided by the task organizers. These datasets are an extension of the M4 dataset from Wang et al. (2024b). The datasets include texts from multiple domains, such as Reddit discussions, Wikipedia pages and arXiv papers to name a few, as well as multiple languages as stated above. In addition, the dataset for subtask B contains machine-generated texts from multiple models. For more information about the shared task, see Wang et al. (2024a).

4 System Overview

The basic components of our design consist of both document-level and token-level features. Document-level features (detailed in Section 4.1) are extracted directly from the text, and the output features of document-level features are concatenated for further use in an MLP for classification. Token-level features, *i.e.,* the measure of predictability, are the probability of the input text according to a large language model, are fed into a BiLSTM network which converts sequences into a fixed-length representation by concatenating both directions; the details of token-level features are outlined in 4.2. Document-level and token-level features are concatenated and then passed to an MLP for classification. This design remains consistent for both subtask A and subtask B, differing only in the dimensionality of the MLP output representation, which requires adjustments to the number of output classes.

4.1 Document-level features

4.1.1 Perplexity feature

Perplexity serves as a crucial measure of a language model's predictive capability regarding word sequences. Essentially, it gauges the level of surprise a language model experiences when encountering a new sequence of words. A lower perplexity score indicates that the language model excels in predicting the next word in a sequence. It's shown in previous studies that generally the text perplexity generated by large language models (e.g. ChatGPT) is lower than that human written text (Liao et al., 2023). Numerous prior studies have either directly evaluated the efficacy of perplexity in discerning machine-generated data or incorporated it into their models (Liao et al., 2023; Mindner et al., 2023).

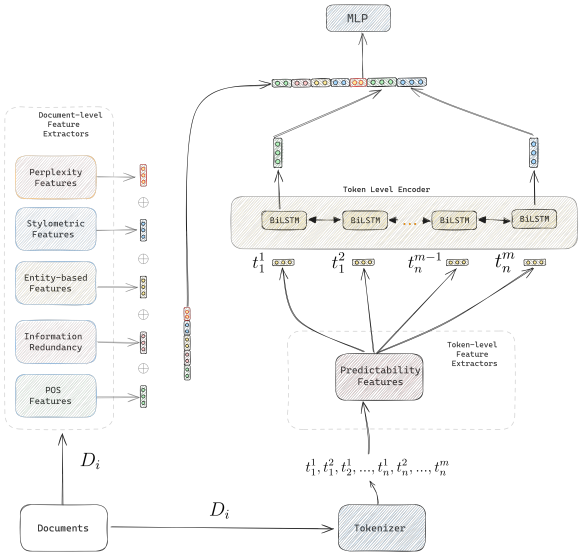

Figure 1: System architecture.

In this study, we employ the XLM-RoBERTa (Conneau et al., 2019) language model to compute the perplexity score for each document. Consequently, each document is represented by 1 perplexity feature.

4.1.2 TF-IDF

We use a text vectorizer to extract term frequency–inverse document frequency (TF-IDF) features based on word-level unigrams. The vocabulary and feature-set for each dataset (A monolingual, A multilingual, B) are calculated separately.

4.1.3 Simple stylometric features

We calculate a small subset of stylometric features. These include: average sentence length by word count; punctuation count, normalized by total number of tokens; number of capitalized words, normalized by total number of words; and the distribution of Part-of-Speech tags in the texts. We make use of the Stanza package (Qi et al., 2020) to perform tokenization, sentence segmentation, and PoS tagging.

4.1.4 Information redundancy

Information redundancy in text may be expressed as lexical or topical repetition. Recent comparisons suggest that machine-generated text is prone to this kind of repetition to some degree (Holtzman et al., 2019), possibly over-repeating words in the output compared to human text (Dou et al., 2021). To calculate information redundancy, we follow the method outlined by Fröhling and Zubiaga (2021).

4.1.5 Entity-based coherence

The inclusion of this feature is based on a hypothesis that human-written text and machine-generated text differ in their use of references to entities throughout the text (Fröhling and Zubiaga, 2021). We extract coherence features using a conventional method which relies on transitions of mention types between sentences (Lapata et al., 2005). An illustration of this process can be found in Figure 2. Due to the limitations of the current co-reference resolution availability, this feature was only used in the monolingual track of subtask A and in subtask B, as these only contained samples in English.

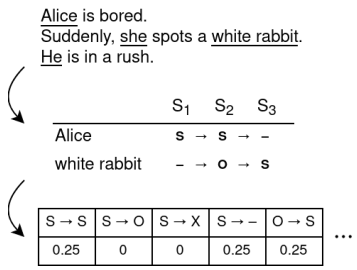

Figure 2: Entity-based coherence illustration.

4.2 Token-level features

4.2.1 Predictability feature

The predictability measurement method, the approach presented by (Przybyła et al., 2023), assesses the likelihood of token sequences using generative language models, distinguishing between machine-generated and human-authored text. Key components of the predictability measurement include:

- *Log-probability of the observed token* t_i^*:

  $\log p(i, t_i^*)$

  This feature measures the likelihood of the observed token given the model's predictions at a specific position in the sequence.

- *Log-probability of the most likely token* w_j *from dictionary* D:

  $\max_{j \in D} \log p(i, w_j)$

  This feature calculates the maximum log probability among all tokens in the model's dictionary, indicating the confidence of the model's top prediction at a particular position.

- *The entropy of the token probability distribution*:

  $-\sum_{j \in D} p(i, w_j) \log p(i, w_j)$

  This feature quantifies the uncertainty of choosing the next token according to the model at a given position.

The XLM-RoBERTa (Conneau et al., 2019) language model is utilized for both Subtasks A and B. Since we only employ the language model, each token is represented by 4 predictability features for all languages, and the maximum sequence length is limited to 128 tokens. The method employs a bidirectional LSTM to distinguish patterns from the sequence of features, without relying on averaging or aggregation functions.

4.3 BERT-LSTM model

Though we mainly focused on feature based system, we have worked on building a simple LSTM model independently as well. For this model, we have used BERT to get sentence embedding, as BERT provide different embedding for the same words based on their context in the sentence. After getting the sentence embedding, we fed it into an LSTM layer, which contains 128 hidden nodes. Subsequently, we have added a linear layer on top of the LSTM layer as the final output layer. The number of output nodes was related to the task it was assigned. For Task B, we have used 6 nodes in the output layer, as there are 6 possible classes. The overview of this system architecture is displayed in Figure 3.

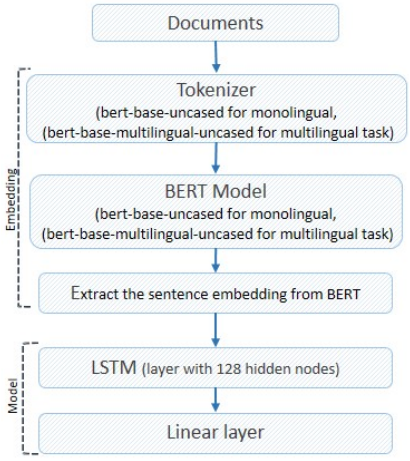

Figure 3: BERT-LSTM system overview.

5 Experimental Setup

We initially divided the training data into two subsets: training and development. We utilized the

development set for testing purposes during the model development phase. However, following the availability of the test data, we incorporated the entire training set for training and the development set for validation. We then evaluated the model's performance on the raw documents in the publicly available test set. For preprocessing, we employed the XLM-RoBERTa tokenizer for features related to perplexity and predictability. Additionally, we utilized the Stanza (Qi et al., 2020) tokenization pipeline for features such as stylometric analysis, entity coherence, information redundancy, and part-of-speech (POS) tagging. Our model was implemented using PyTorch. We employed accuracy as the official evaluation measure.

6 Results

6.1 Feature-based Model

Our numerical results on the test dataset are displayed fully in Table 1. Overall, our system did not perform very well in general or according to official metrics. Our monolingual subtask A model did not learn to differentiate between human and machine texts, and predicted all test examples to be machine-generated. Our subtask B model suffered a similar fate, predicting all test examples to be written by ChatGPT. Our multilingual subtask A was our only model which was able to distinguish between examples to some extent. However, this model also had an extreme bias towards the "machine" label.

Table 1: Overview of our results on each of the subtasks. Values represent accuracy of predictions made on the test set of each subtask.

Subtask	Task Baseline	Our Result	Our Ranking
A Mono.	.884	.525	128/137
A Multi.	.808	.512	61/68
B	.746	.166	74/77

In an attempt to further understand the lack of learning by our models, we examined the raw features produced by our feature extractors on the test set examples. Interestingly, we find that some features did actually differ notably in value for human and machine texts. We calculate means and standard deviations for the raw features on human and machine texts separately, and compare the results using the Cohen's d effect-size metric. Some notable results are shown in Table 3. As expected, the information loss, represented as the norm of the difference between the original document matrix and the truncated matrix, was higher in human texts than in machine texts in the test set, suggesting that the machine texts had more information redundancy, *i.e.,* repetition of information. We observe some interesting findings regarding PoS distribution in the texts, such as higher presence of pronouns, auxiliary verbs, and particles in human texts versus higher presence of nouns, adjectives, and adpositions in machine texts.

Table 2: Confusion matrix for predictions made by our subtask A multilingual model, comparing predicted labels with gold labels.

Subtask A Multi.		Predicted	
		Human	Machine
Gold	Human	406	16259
	Machine	460	17147

Note: Confusion matrices for other subtasks are redundant, as our models only predicted a single label for each of them ('Machine' for subtask A monolingual, and 'ChatGPT' for subtask B).

Table 3: Notable features with effect sizes > 0.3 as calculated on examples from the monolingual A test set. Positive values denote that these features were higher in human texts than in machine texts.

Feature	Effect size (d) (Human − Machine)
Frequency of pronouns	1.59
Frequency of auxiliary verbs	1.49
Frequency of particles	0.8
Frequency of adverbs	0.58
Frequency of verbs	0.54
$\|A - A_{\text{trunc}}\|$ (Information loss)	0.31
$\min(A_{\text{trunc}})$ (Info. redundancy)	-0.51
Frequency of adpositions	-0.75
Punctuation count	-1.03
Frequency of adjectives	-1.31
Frequency of nouns	-1.62

6.2 BERT-LSTM model

We employed BERT (Devlin et al., 2018) to obtain sentence embeddings, a process that significantly increased the computational complexity of our BERT-LSTM system. The model ended up pre-

dicting all the labels on the test set as the same. As a result this system get an 16.67% accuracy on the task B which is not better than a random selection. Due to the time constraint, we could not manage to experiment with Task A.

7 Conclusion

Our overall conclusion is that our examined subset of linguistically motivated features alongside probabilistic features was not able to contribute almost any performance at all to a classifier of human and machine texts. While some features did differ in value between human and machine texts, these differences did not translate into a learning advantage for a hybrid model. Our findings underscore the nuanced challenges inherent in developing robust detection mechanisms for machine-generated text, emphasizing the need for further exploration and refinement of feature engineering strategies to effectively address this evolving domain.

8 Acknowledgments

This submission has been carried out as part of the 2023-2024 edition of the master course Shared Task Information Science (LIX026M05) at the University of Groningen, taught by Lukas Edman and Antonio Toral. We want to humbly express our sincerest gratitude to both Antonio Toral Ruiz and Lukas Edman for their help and support during this task.

We thank the Center for Information Technology of the University of Groningen for their support and for providing access to the Hábrók high-performance computing cluster.

Finally, we would like to express our sincere gratitude to the Erasmus Mundus Masters Program in Language and Communication Technologies (LCT) for supporting students who participated in this project.

References

Chaka Chaka. 2023. Detecting ai content in responses generated by chatgpt, youchat, and chatsonic: The case of five ai content detection tools. *Journal of Applied Learning and Teaching*, 6(2).

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Robert Deibel and Denise Löfflad. 2021. Style change detection on real-world data using an lstm-powered attribution algorithm. In *CLEF (Working Notes)*, pages 1899–1909.

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

Yao Dou, Maxwell Forbes, Rik Koncel-Kedziorski, Noah A Smith, and Yejin Choi. 2021. Is gpt-3 text indistinguishable from human text? scarecrow: A framework for scrutinizing machine text. *arXiv preprint arXiv:2107.01294*.

Leon Fröhling and Arkaitz Zubiaga. 2021. Feature-based detection of automated language models: tackling gpt-2, gpt-3 and grover. *PeerJ Computer Science*, 7:e443.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander M Rush. 2019. Gltr: Statistical detection and visualization of generated text. *arXiv preprint arXiv:1906.04043*.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*.

Shriya TP Gupta, Jajati Keshari Sahoo, and Rajendra Kumar Roul. 2019. Authorship identification using recurrent neural networks. In *Proceedings of the 2019 3rd International Conference on Information System and Data Mining*, pages 133–137.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Mirella Lapata, Regina Barzilay, et al. 2005. Automatic evaluation of text coherence: Models and representations. In *Ijcai*, volume 5, pages 1085–1090.

Jenny S Li, John V Monaco, Li-Chiou Chen, and Charles C Tappert. 2014. Authorship authentication using short messages from social networking sites. In *2014 IEEE 11th International Conference on e-Business Engineering*, pages 314–319. IEEE.

Wenxiong Liao, Zhengliang Liu, Haixing Dai, Shaochen Xu, Zihao Wu, Yiyang Zhang, Xiaoke Huang, Dajiang Zhu, Hongmin Cai, Tianming Liu, et al. 2023. Differentiate chatgpt-generated and human-written medical texts. *arXiv preprint arXiv:2304.11567*.

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. *arXiv preprint arXiv:1907.11692*.

Lorenz Mindner, Tim Schlippe, and Kristina Schaaff. 2023. Classification of human-and ai-generated texts: Investigating features for chatgpt. In *International Conference on Artificial Intelligence in Education Technology*, pages 152–170. Springer.

Sandra Mitrović, Davide Andreoletti, and Omran Ayoub. 2023. Chatgpt or human? detect and explain. explaining decisions of machine learning model for detecting short chatgpt-generated text. *arXiv preprint arXiv:2301.13852*.

Lisa Pearl and Mark Steyvers. 2012. Detecting authorship deception: a supervised machine learning approach using author writeprints. *Literary and linguistic computing*, 27(2):183–196.

Piotr Przybyła, Nicolau Duran-Silva, and Santiago Egea-Gómez. 2023. I've seen things you machines wouldn't believe: Measuring content predictability to identify automatically-generated text. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2023). CEUR Workshop Proceedings, CEUR-WS, Jaén, Spain*.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Areg Mikael Sarvazyan, José Ángel González, Marc Franco-Salvador, Francisco Rangel, Berta Chulvi, and Paolo Rosso. 2023. Overview of autextification at iberlef 2023: Detection and attribution of machine-generated text in multiple domains. *arXiv preprint arXiv:2309.11285*.

Tatiana Shamardina, Vladislav Mikhailov, Daniil Chernianskii, Alena Fenogenova, Marat Saidov, Anastasiya Valeeva, Tatiana Shavrina, Ivan Smurov, Elena Tutubalina, and Ekaterina Artemova. 2022. Findings of the the ruatd shared task 2022 on artificial text detection in russian. *arXiv preprint arXiv:2206.01583*.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, Thomas Arnold, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024a. Semeval-2024 task 8: Multigenerator, multidomain, and multilingual black-box machine-generated text detection. In *Proceedings of the 18th International Workshop on Semantic Evaluation*, SemEval 2024, Mexico, Mexico.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Toru Sasaki, Thomas Arnold, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024b. M4: Multi-generator, multi-domain, and multi-lingual black-box machine-generated text detection. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics*, Malta.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. *CoRR*, abs/1905.12616.