An Investigation of Language Model Interpretability via Sentence Editing

UNDERGRADUATE RESEARCH THESIS
Presented in Fulfillment of the Requirements for
Honors Research Distinction in Computer Science Engineering
College of Engineering, Computer Science Department
The Ohio State University

By
Samuel R. Stevens
B.S. Computer Science & Engineering 2021
The Ohio State University
April 2021

Committee:

Dr. Yu Su
Assistant Professor
Dept. of Computer Science
Honors Thesis Research Advisor

Dr. Huan Sun
Assistant Professor
Dept. of Computer Science
Honors Thesis Research Committee
Abstract

Pre-trained language models (PLMs) like BERT are being used for almost all language-related tasks, but interpreting their behavior still remains a significant challenge and many important questions remain largely unanswered. For example, how does domain-specific pre-training change the dynamics within a model? Is task-specific fine-tuning necessary for model interpretability? Which interpretability techniques best correlate with human rationales? In this work, we re-purpose a sentence editing dataset, where high-quality human rationales can be automatically extracted and compared with model rationales, as a new testbed for interpretability. This enables us to conduct a systematic investigation of the aforementioned open questions regarding PLMs’ interpretability and generate new insights. The dataset and code will be released to facilitate future research on interpretability.
Acknowledgements

Thank you to Dr. Yu Su for his mentorship and support through my project. Without his guidance, I would not have discovered my passion for research.

Thank you to Dr. Huan Sun for agreeing to serve as a member of my Honors Research Distinction committee.
Table of Contents

Abstract ii
Acknowledgements iii
List of Tables v
List of Figures v
1 Introduction 1
2 Related Work 3
3 Proposed Task 4
  3.1 Human Rationales 4
  3.2 Model Rationales 5
  3.3 Evaluation 6
4 Experiments 7
  4.1 Pre-training Procedure 7
  4.2 Attention Weights vs. Input Gradients 9
  4.3 Transformer Layer 9
5 Conclusion 11
A Training Details 21
List of Tables

1. Performance on original AESW task ........................................... 7
2. Special AESW tokens ................................................................. 21
3. Hyperparameter bounds during fine-tuning ................................. 22

List of Figures

1. Two examples from AESW .......................................................... 2
2. Comparison of model interpretability ......................................... 8
3. Interpretability for each transformer layer .................................. 10
Chapter 1. Introduction

Pre-trained language models (PLMs) [1, 2, 3] are pervasively used in language-related tasks, but interpreting their predictions is notoriously difficult because of their parameters’ complex inter-dependencies. Given a specific prediction, we want to know why a model made that decision, both to further improve performance and to use the model in high-stakes scenarios such as healthcare or bank loan approvals, where explanation is important. This has motivated efforts in extracting model explanations, typically in the form of rationales, i.e., subsets of the original input that support a decision [4]. Attention heatmaps [5] and gradient-based saliency maps [6] are common extraction methods.

There have been efforts on developing datasets for interpretability research, for example, the recent ERASER benchmark [7]. However, the majority of ERASER tasks use human rationales highlighted by a different annotator after the original labeling process. Such rationales are not necessarily faithful; a rationale highlighted by the second annotator may not have been used by the first annotator while labeling.

Our first contribution is the realization that the faithful human rationales can be automatically extracted from AESW (Automatic Evaluation of Scientific Writing; [8]), a dataset for sentence editing, and thus the dataset can be re-purposed as a testbed for interpretability research. See Figure 1 for examples.

With our new task, we investigate multiple factors in PLM interpretability, comparing (1) pre-training procedures, (2) attention weight- and input gradient-based methods of extracting model rationales, and (3) transformer layer interpretability. While previous work [9, 10] has shown that attention weights are not always faithful,
The algorithm described in the previous sections has several advantages.

However, we must note that we still have no means of deciding which documents out of $\text{MATH}_1$ and $\text{MATH}_2$, respectively.

Figure 1: Two “need edit” examples from AESW in the original data format and a human-readable format. The first example (a) has a spelling error “descripted” and the second (b) is edited for concision.

we find that they correlate better with human rationales than gradient-based methods. We also find that domain-specific pre-training leads to increases in interpretability in early layers, evidence that PLMs attend strongly to novel patterns with a strong influence on loss.
Chapter 2. Related Work

Human rationales (as defined by [4]) are subsets of input highlighted by annotators as evidence to support a decision. The same annotator labeling an example might also highlight their rationale [11, 12]. In other cases, rationales are collected for an existing dataset by different annotators [13, 14, 15]. As previously stated, such rationales may not be faithful. Rationale length can vary from sub-sentence spans [16] to multiple sentences [17].

Model rationales can be produced as an explicit training objective [13] or extracted as a post-hoc explanation. Post-hoc methods typically assign token-level importance scores: attention weights are often used in attention-based models [18], gradient-based explanations are typical for differentiable models [19, 20], and LIME is a model-agnostic method [21]. We follow work that uses BERT’s attention directly [22, 23] to extract rationales.

A model rationale is evaluated on faithfulness (if it is actually used to make a decision) and plausibility (if it is easily understood by humans). Faithfulness can be measured by perturbing inputs marked as evidence and measuring change in outputs [9, 10]. Plausibility can be measured through user studies, wherein users are given a model rationale and asked either to predict the model’s decision [24] or to rate rationale understandability [25, 26, 27, 28]. Rationale plausibility can also be measured by similarity to human rationales [7], but this requires faithful human rationales. We use similarity to evaluate rationale plausibility because we gather faithful human rationales.
Chapter 3. Proposed Task

We propose re-purposing the AESW classification task for measuring interpretability. We filter AESW for examples from which we can automatically extract human rationales, then evaluate model rationales on their similarity to human rationales.

Chapter 3.1. Human Rationales

Human rationales are substrings used as evidence for a decision \[^4\]. Faithful and sufficient (enough evidence to justify a decision) human rationales can be used as gold labels for evaluating model rationale plausibility.

The original AESW task is to predict if a sentence from a scientific paper needs editing. Daudaravicius et al. label spans of a sentence before and after editing as deleted (\(<\text{del}>\)) or inserted (\(<\text{ins}>\)) and provide 1.1M training, 140K validation and 140K testing examples.

We exploit the data’s format to automatically extract faithful and sufficient human rationales. Deleted text (text between \(<\text{del}>\) tags) is always a faithful rationale (if it was somehow edited, the sentence would be error-free). Deleted text alone is not always a sufficient rationale to justify “need edit”; sometimes text must be added before a sentence is acceptable.

To find edits where deleted text is always a sufficient rationale, we use two criteria:

1. A misspelled word is corrected (spelling error)
2. Text is only deleted, not added (deleted text)
Spelling errors are always a sufficient rationale to justify editing a sentence (see Figure 1a). In edits with no insertions, removing the deleted text would form an error-free sentence, so the deleted text is sufficient explanation to justify editing (see Figure 1b). We extract faithful and sufficient human rationales for 1,321 spelling error edits and 6,741 deleted text edits from the validation examples.

Chapter 3.2. Model Rationales

Model rationales are substrings provided by the model as evidence for a decision. Given a model, an example $x_i$ and a prediction $y_i$, we extract two model rationales.

First, we use attention maps [5, 23] to rank word relevance. We find the total attention weight from the initial [CLS] token to each token $t$ across $H$ attention heads. Then we add those totals together for each token $t$ in a word $w$:

$$\text{score}(w) = \sum_{t \in w} \sum_{h=1}^{H} \text{Attn}_h ([\text{CLS}] \to t)$$  \hspace{1cm} (1)

We also use gradient-based saliency maps (specifically gradient $\times$ input; [19, 20]) to rank word relevance. We calculate a saliency score for each token $t$ in $x_i$. The change in loss with respect to $t$’s input embedding $-\nabla_{e(t)} L_{\hat{y}_i}$ captures the sensitivity to token $t$. Multiplying by $e(t)$ then measures each token’s marginal impact on the model prediction [29]. Again, we compute a word score by summing over each token $t$ in word $w$:

$$\text{score}(w) = \sum_{t \in w} -\nabla_{e(t)} L_{\hat{y}_i} \cdot e(t)$$  \hspace{1cm} (2)

In contrast to attention weights, input gradients are always faithful [30].
Chapter 3.3. Evaluation

We evaluate model interpretability by the similarity of model and human rationales. We measure the Jaccard similarity between the human rationale’s words and the model’s top $n$ ranked words, where $n$ is the number of words in the human rationale. We only compare the top $n$ words to assess whether models prioritize the same words as humans.
Chapter 4. Experiments

To demonstrate the utility of the AESW task for interpretability research, we present three experiments, each with the goal of understanding factors in PLM interpretability.

For our experiments, we fine-tune three BERT-based models: BERT, RoBERTa and SciBERT.1 We add a linear classifier on top of the \([\text{CLS}]\) token representation, fine-tune each model end-to-end on the AESW training set, and use validation loss to tune hyperparameters. We do not add any interpretability-related objectives.2 As shown in Table 1, our fine-tuned models set a new state of the art on the AESW dataset, laying a strong foundation for the subsequent interpretability analyses.

Chapter 4.1. Pre-training Procedure

Does pre-training procedure affect model interpretability? Different pre-training procedures can improve BERT’s performance on down-stream tasks \([2\,31]\); we are

| Model            | Dev Set | Test Set |
|------------------|---------|----------|
|                  | Prec.   | Rec.     | F1     | Prec.   | Rec.     | F1     |
| CNN+LSTM         | -       | -        | -      | 0.544   | 0.741    | 0.628  |
| CNN              | -       | -        | -      | 0.503   | 0.779    | 0.611  |
| SVM              | -       | -        | -      | 0.448   | 0.728    | 0.555  |
| BERT\textsubscript{base} | 0.690   | 0.622    | 0.654  | 0.704   | 0.633    | 0.666  |
| RoBERTa\textsubscript{base}| 0.716   | 0.614    | 0.661  | 0.726   | 0.622    | 0.670  |
| SciBERT\textsubscript{base} | 0.705   | 0.617    | 0.658  | 0.715   | 0.627    | 0.668  |

Table 1: Performance on the original AESW sentence classification task. Dev set results are not available for models reported in Daudaravicius et al. [8].

---

1Devlin et al. [1], Liu et al. [2], and Beltagy, Lo, and Cohan [3], respectively.
2Appendix A contains more details on fine-tuning.
We are interested in how pre-training affects interpretability after fine-tuning. We compare BERT, RoBERTa (pre-trained for 5x longer than BERT with 10x times more data) and SciBERT (pre-trained on a corpus of academic papers). Different interpretability scores would mean the models rank word relevance differently.

We extract model rationales using attention weights and measure their Jaccard similarity to human rationales. We add three baselines: random word rankings, rationales from BERT with no fine-tuning and rationales from BERT fine-tuned on the CoLA task. As seen in Figure 2, RoBERTa and BERT are nearly equally interpretable despite differences in pre-training corpus size. We hypothesize that SciBERT is less interpretable because it encodes “need edit” representations in early layers, then attends to [SEP] as a no-op in later layers, as proposed in Clark et al. [22] and Kobayashi et al. [34].

---

3The Corpus of Linguistic Acceptability [32] is a task on the GLUE Benchmark [33] wherein models evaluate a sentence’s grammatical acceptability.
Chapter 4.2. Attention Weights vs. Input Gradients

Do attention weights or input gradients produce better rationales? Input gradient saliency scores are naturally faithful, while attention weights are not [9]. However, attention weights in later layers represent word relevance in context, potentially leading to more plausible rationales.

We extract rationales using the gradient×input method described in Section 3.2 and measure their similarity to human rationales. We also extract a second set of rationales using gradient×input’s magnitude to rank words (|gradient × input|).

Figure 2 shows that attention weights match human rationales better than input gradients and that the difference is more pronounced on deleted text edits. This is consistent with work showing that input gradients are better for more syntactically oriented tasks [29]. Using |gradient × input| (right-most) also shows improvements over directional gradient×input (middle-right), in contrast to Han, Wallace, and Tsvetkov [29].

Chapter 4.3. Transformer Layer

How interpretable is each transformer layer? It is widely agreed that BERT’s middle layers encode more syntactic information than other layers [35,36,37]. We hypothesize that those middle layers will form better rationales for spelling error than deleted text edits because spelling is a more syntax-oriented task.

We extract rationales from each layer’s attention weights and measure their similarity to human rationales in Figure 3. SciBERT identifies the majority of spelling errors in layers 3 and 4 and peaks at layer 10 for both edit types.

We hypothesize that PLMs attend heavily to novel patterns that have a strong
influence on loss. Spelling errors are a novel pattern to SciBERT (spelling errors are likely rare in its pre-training corpus of scientific papers) and strongly influence loss (a spelling error strongly indicates “need edit”). In contrast, RoBERTa and BERT only show an increase in interpretability in their final layers because those are the layers that change the most during fine-tuning [23].

Figure 3: Mean Jaccard similarity for each layer (using the mean strategy) for each model for spelling error and deleted text edits.\textsuperscript{4}

\textsuperscript{4}BERT shows a decline in interpretability at layer 11 in both edit types because it attends heavily to periods.
Chapter 5. Conclusion

We re-purpose the AESW task to gather faithful human rationales and investigate an array of questions regarding PLM interpretability. We find that attention produces more plausible rationales than input gradients, especially when considering direction, and that domain-specific pre-training makes earlier layers attend to more relevant words.

Future work might expand the subset of examples for which human rationales can be automatically extracted, evaluate more complex methods to extract model rationales or include human rationales during training.
References

[1] Jacob Devlin et al. “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). Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423 URL: https://www.aclweb.org/anthology/N19-1423.

[2] Yinhan Liu et al. RoBERTa: A Robustly Optimized BERT Pretraining Approach. 2019. arXiv: 1907.11692 [cs.CL].

[3] Iz Beltagy, Kyle Lo, and Arman Cohan. “SciBERT: A Pretrained Language Model for Scientific Text”. 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). Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 3615–3620. DOI: 10.18653/v1/D19-1371 URL: https://www.aclweb.org/anthology/D19-1371.

[4] Omar Zaidan, Jason Eisner, and Christine Piatko. “Using “Annotator Rationales” to Improve Machine Learning for Text Categorization”. In: Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference. Rochester, New York: Association for Computational Linguistics,
[5] Kelvin Xu et al. “Show, Attend and Tell: Neural Image Caption Generation with Visual Attention”. In: Proceedings of the 32nd International Conference on Machine Learning. Ed. by Francis Bach and David Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, July 2015, pp. 2048–2057. URL: http://proceedings.mlr.press/v37/xuc15.html.

[6] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. 2014. arXiv: 1312.6034 [cs.CV].

[7] Jay DeYoung et al. “ERASER: A Benchmark to Evaluate Rationalized NLP Models”. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics, July 2020, pp. 4443–4458. DOI: 10.18653/v1/2020.acl-main.408. URL: https://www.aclweb.org/anthology/2020.acl-main.408.

[8] Vidas Daudaravicius et al. “A Report on the Automatic Evaluation of Scientific Writing Shared Task”. In: Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications. San Diego, CA: Association for Computational Linguistics, June 2016, pp. 53–62. DOI: 10.18653/v1/W16-0506. URL: https://www.aclweb.org/anthology/W16-0506.
[9] Sarthak Jain and Byron C. Wallace. “Attention is not Explanation”. 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). Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 3543–3556. DOI: 10.18653/v1/N19-1357. URL: https://www.aclweb.org/anthology/N19-1357.

[10] Sofia Serrano and Noah A. Smith. “Is Attention Interpretable?” In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 2931–2951. DOI: 10.18653/v1/P19-1282. URL: https://www.aclweb.org/anthology/P19-1282.

[11] Daniel Khashabi et al. “Looking Beyond the Surface: A Challenge Set for Reading Comprehension over Multiple Sentences”. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). New Orleans, Louisiana: Association for Computational Linguistics, June 2018, pp. 252–262. DOI: 10.18653/v1/N18-1023. URL: https://www.aclweb.org/anthology/N18-1023.

[12] James Thorne et al. “FEVER: a Large-scale Dataset for Fact Extraction and VERification”. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). New Orleans, Louisiana: Association for Computational Linguistics, June 2018, pp. 809–819. DOI: 10.18653/v1/N18-1074. URL: https://www.aclweb.org/anthology/N18-1074.
[13] Omar F. Zaidan, Jason Eisner, and Christine Piatko. “Machine Learning with Annotator Rationales to Reduce Annotation Cost”. In: Proceedings of the NIPS*2008 Workshop on Cost Sensitive Learning. Dec. 2008.

[14] Oana-Maria Camburu et al. “e-SNLI: Natural Language Inference with Natural Language Explanations”. In: Advances in Neural Information Processing Systems. Ed. by S. Bengio et al. Vol. 31. Curran Associates, Inc., 2018, pp. 9539–9549. URL: https://proceedings.neurips.cc/paper/2018/file/4c7a167bb329bd92580a99ce422d6fa6-Paper.pdf.

[15] Nazneen Fatema Rajani et al. “Explain Yourself! Leveraging Language Models for Commonsense Reasoning”. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 4932–4942. DOI: 10.18653/v1/P19-1487. URL: https://www.aclweb.org/anthology/P19-1487.

[16] Alon Talmor et al. “CommonsenseQA: A Question Answering Challenge Targeting Commonsense Knowledge”. 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). Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 4149–4158. DOI: 10.18653/v1/N19-1421. URL: https://www.aclweb.org/anthology/N19-1421.

[17] Eric Lehman et al. “Inferring Which Medical Treatments Work from Reports of Clinical Trials”. In: Proceedings of the 2019 Conference of the North
[18] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. *Neural Machine Translation by Jointly Learning to Align and Translate*. 2016. arXiv: [1409.0473 [cs.CL]].

[19] Misha Denil, Alban Demiraj, and Nando de Freitas. *Extraction of Salient Sentences from Labelled Documents*. 2015. arXiv: [1412.6815 [cs.CL]].

[20] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. “Learning Important Features Through Propagating Activation Differences”. In: *Proceedings of the 34th International Conference on Machine Learning*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. International Convention Centre, Sydney, Australia: PMLR, Aug. 2017, pp. 3145–3153. URL: [http://proceedings.mlr.press/v70/shrikumar17a.html](http://proceedings.mlr.press/v70/shrikumar17a.html).

[21] Marco Ribeiro, Sameer Singh, and Carlos Guestrin. “‘Why Should I Trust You?’: Explaining the Predictions of Any Classifier”. In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*. San Diego, California: Association for Computational Linguistics, June 2016, pp. 97–101. DOI: [10.18653/v1/N16-3020] URL: [https://www.aclweb.org/anthology/N16-3020](https://www.aclweb.org/anthology/N16-3020).
[22] Kevin Clark et al. “What Does BERT Look at? An Analysis of BERT’s Attention”. In: Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. Florence, Italy: Association for Computational Linguistics, Aug. 2019, pp. 276–286. DOI: [10.18653/v1/W19-4828](https://www.aclweb.org/anthology/W19-4828).

[23] Olga Kovaleva et al. “Revealing the Dark Secrets of BERT”. 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). Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 4365–4374. DOI: [10.18653/v1/D19-1445](https://www.aclweb.org/anthology/D19-1445).

[24] Been Kim, Rajiv Khanna, and Oluwasanmi Koyejo. “Examples Are Not Enough, Learn to Criticize! Criticism for Interpretability”. In: Proceedings of the 30th International Conference on Neural Information Processing Systems. NIPS’16. Barcelona, Spain: Curran Associates Inc., 2016, pp. 2288–2296. ISBN: 9781510838819.

[25] Dong Nguyen. “Comparing Automatic and Human Evaluation of Local Explanations for Text Classification”. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). New Orleans, Louisiana: Association for Computational Linguistics, June 2018, pp. 1069–1078. DOI: [10.18653/v1/N18-1097](https://www.aclweb.org/anthology/N18-1097).
[26] Upol Ehsan et al. “Rationalization: A Neural Machine Translation Approach to Generating Natural Language Explanations”. In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*. AIES ’18. New Orleans, LA, USA: Association for Computing Machinery, 2018, pp. 81–87. ISBN: 9781450360128. DOI: 10.1145/3278721.3278736 URL: https://doi.org/10.1145/3278721.3278736.

[27] Upol Ehsan et al. “Automated Rationale Generation: A Technique for Explainable AI and Its Effects on Human Perceptions”. In: *Proceedings of the 24th International Conference on Intelligent User Interfaces*. IUI ’19. Marina del Ray, California: Association for Computing Machinery, 2019, pp. 263–274. ISBN: 9781450362726. DOI: 10.1145/3301275.3302316 URL: https://doi.org/10.1145/3301275.3302316.

[28] Julia Strout, Ye Zhang, and Raymond Mooney. “Do Human Rationales Improve Machine Explanations?” In: *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Florence, Italy: Association for Computational Linguistics, Aug. 2019, pp. 56–62. DOI: 10.18653/v1/W19-4807 URL: https://www.aclweb.org/anthology/W19-4807.

[29] Xiaochuang Han, Byron C. Wallace, and Yulia Tsvetkov. “Explaining Black Box Predictions and Unveiling Data Artifacts through Influence Functions”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, July 2020, pp. 5553–5563. DOI: 10.18653/v1/2020.acl-main.492 URL: https://www.aclweb.org/anthology/2020.acl-main.492.
[30] Jasmijn Bastings and Katja Filippova. “The elephant in the interpretability room: Why use attention as explanation when we have saliency methods?” In: Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP. Online: Association for Computational Linguistics, Nov. 2020, pp. 149–155. DOI: 10.18653/v1/2020.blackboxnlp-1.14 URL: https://www.aclweb.org/anthology/2020.blackboxnlp-1.14.

[31] Kevin Clark et al. “ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators”. In: ICLR. 2020. URL: https://openreview.net/pdf?id=r1xMH1BtvB.

[32] Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. “Neural Network Acceptability Judgments”. In: Transactions of the Association for Computational Linguistics 7 (Mar. 2019), pp. 625–641. DOI: 10.1162/tacl_a_00290 URL: https://www.aclweb.org/anthology/Q19-1040.

[33] Alex Wang et al. “GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding”. In: Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. Brussels, Belgium: Association for Computational Linguistics, Nov. 2018, pp. 353–355. DOI: 10.18653/v1/W18-5446 URL: https://www.aclweb.org/anthology/W18-5446.

[34] Goro Kobayashi et al. “Attention is Not Only a Weight: Analyzing Transformers with Vector Norms”. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Online: As-
[35] John Hewitt and Christopher D. Manning. “A Structural Probe for Finding Syntax in Word Representations”. 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). Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 4129–4138. DOI: 10.18653/v1/N19-1419 URL: https://www.aclweb.org/anthology/N19-1419.

[36] Yoav Goldberg. “Assessing BERT’s Syntactic Abilities”. In: CoRR abs/1901.05287 (2019). arXiv: 1901.05287 URL: http://arxiv.org/abs/1901.05287

[37] Anna Rogers, Olga Kovaleva, and Anna Rumshisky. “A Primer in BERTology: What We Know About How BERT Works”. In: Transactions of the Association for Computational Linguistics 8 (2020), pp. 842–866. DOI: 10.1162/tacl_a_00349, eprint: https://doi.org/10.1162/tacl_a_00349 URL: https://doi.org/10.1162/tacl_a_00349

[38] Thomas Wolf et al. “HuggingFace’s Transformers: State-of-the-art Natural Language Processing”. In: ArXiv abs/1910.03771 (2019).
Appendix A. Training Details

The AESW task uses scientific papers written in LaTeX, which contains markup characters that impact sentence meaning. The original authors (Daudaravicius et al.) replace these characters with special tokens, as seen in Table 2. We add these four special tokens (_MATH_, _MATHDISP_, _CITE_ and _REF_) to the model vocabulary, fine-tuning the word representations during training.

| LaTeX Example                  | Special Token |
|-------------------------------|---------------|
| $\beta_{2}^2$                | _MATH_        |
| $$2 + 3$$                     | _MATHDISP_    |
| \cite{google2018}            | _CITE_        |
| \ref{tab:results}            | _REF_         |

Table 2: Special tokens found in the original AESW data that should not be split further into bytes/tokens.

We train all models for a maximum of 30 epochs with a patience of 5 on a single Tesla P100 GPU. All models (BERT\(^5\), SciBERT\(^6\), RoBERTA\(^7\)) are based on their HuggingFace \([38]\) implementations. We list all the key hyperparameters and tuning bounds for reproducibility in Table 3. Additionally, we will release code and instructions for reproducing results.

\(^5\)https://huggingface.co/transformers/v3.0.2/model_doc/bert.html#bertforsequenceclassification
\(^6\)https://github.com/allenai/scibert#pytorch-huggingface-models
\(^7\)https://huggingface.co/transformers/v3.0.2/model_doc/roberta.html#robertaforsequenceclassification
| Model         | Hyperparameters                        | Hyperparameter bounds |
|--------------|---------------------------------------|-----------------------|
| BERT<sub>base</sub> | learning rate: $1 \times 10^{-6}$    | learning rate: $(2 \times 10^{-7}, $ $1 \times 10^{-6}, 2 \times 10^{-5}, $ $1 \times 10^{-4})$ |
|              | batch size: 32                         |                       |
|              | model: bert-base-uncased               |                       |
|              | vocab size: 30526 (normally 30522)     |                       |
|              | learning rate: $1 \times 10^{-6}$     |                       |
| RoBERTa<sub>base</sub> | batch size: 32                        | learning rate: $(1 \times 10^{-6})$ |
|              | model: roberta-base                    |                       |
|              | vocab size: 50269 (normally 50265)     |                       |
|              | learning rate: $1 \times 10^{-6}$     |                       |
|              | batch size: 32                         |                       |
| SciBERT      | model: allenai/scibert_scivocab_uncased | learning rate: $(1 \times 10^{-6})$ |
|              | vocab size: 31094 (normally 31090)     |                       |

Table 3: Hyperparameter options for each model. Note that each model had 4 special tokens added to the vocabulary. BERT was fine-tuned first. Because of compute limitations, RoBERTa and SciBERT were both fine-tuned using the same hyperparameters as the optimal BERT configuration (learning rate of $(1 \times 10^{-6})$).