Spurious Correlation Detection in Natural Language Processing

Metehan Berker metehanberker99@g.ucla.edu

Nathan Huey njhuey45@g.ucla.edu

Charley Sanchez sanchez98@g.ucla.edu

Abstract

NLP models can struggle to achieve robustness and generalization due to spurious correlations learned from the data. While there exists literature which extensively focuses on identifying and mitigating these correlations, there has been limited effort in quantifying their extent. The existing methods to quantify spurious correlations often treat models as a black box and do not analyze token level contributions. In our work, we employ DecompX, a state-of-the-art interpretability framework, to evaluate the extent of which spurious correlations are affecting predictions. Through fine-grained analysis, we quantify the models' attention to spurious features and their robustness against counterfactual examples, which offers new insights into their behavior and susceptibility to biases.

Code:github.com/metehanberker1/pretrained-encoders-spurious-learning-analysis

1 Introduction

Spurious correlations in NLP models are predictions based on superficial or irrelevant pattern, due to bias of training data or behavior of the model such as the model overfitting to co occurring terms or syntactic shortcuts. Such correlations negatively impact robustness, fairness, and generalization, resulting in unreliable real world performance. These correlations are critical to the understanding and quantification of, in order to build NLP systems that are more robust and interpretable.

However, detecting and quantifying spurious correlations is a challenging task. So far, most research has been focused on mitigation techniques such as data augmentation, and measuring the effects of these correlations at a granular level has received less attention. Naive approaches, treating models as black boxes, can not analyze internal token level contributions, nor dependencies in intermediate representations, and therefore provide little actionable insights for reducing biases.

Current solutions to this problem, including error analysis and attention attribution, have their own shortcomings. Attention based methods are unable to establish token level causality, and black box evaluations are not interpretable. Precise attributions are provided by gradient based approaches, which are computationally intensive and difficult for complex models. In our approach, we use DecompX for finegrained analysis, which creates decomposed token representations and passes them through the model without mixing layers. As a result, DecompX can directly trace out token contributions to predictions, while providing a solid basis to quantify spurious correlations in encoder models.

This work evaluates how encoder models are susceptible to spurious correlations with DecompX. We test datasets with annotated spans that influence results, measure token level contributions, and compare results across different model sizes and architectures. Our method is detailed but currently restricted to classification tasks, and would need to be adapted for more general NLP applications. While it does not directly evaluate the importance given to spurious correlations, it studies the model behavior on influential spans and a foundation for constructing bias resilient NLP systems.

2 Related Work

Existing work in NLP models has explored spurious correlations from biases in input features, label ambiguities, and uneven data distributions, which dramatically impact robustness and generalization. Data augmentation, balancing, and refined training objectives strategies have also been designed by researchers to get models to detect and learn important patterns from the data. Moreover, these correlations have been quantified by analyzing input output patterns to assess model reliance on irrelevant features, which can serve as a basis for improving fairness and robustness. Additionally, interpretability tools such as attention based and gradient based methods provide a more fine grained understanding of token level contributions and model behavior and decision making.

2.1 Sources of Spurious Correlations in Textual Data

Model robustness is highly sensitive to input-level perturbations, including syntactic changes, domain-specific term dependencies, and keyword frequency patterns. For example, just word swaps and punctuation changes can cause predictions to fail, revealing that the model is relying on surface features [1]. Moreover, models can generalize poorly by memorizing named entity dependencies[2] or by failing to generalize to broader tasks when trained on domain specific terms [3]. Furthermore, keyword frequency patterns skew predictions by forcing models to emphasize highly correlated terms at the expense of other, less prominent but relevant features in noisy datasets [4].

Such input level issues often combine with label and class ambiguity issues, making accurate predictions even more difficult. Models often fail to differentiate similar classes when features overlap between them and instead rely on shallow heuristics instead of genuine understanding [5]. To address this, one method uses joint label word embeddings to enable models to incorporate label semantics into attention mechanisms and reduce the impact of label ambiguity as well as spurious correlations [6].

These issues are built upon, and biases in data distribution and label frequency imbalances further exacerbate the model's reliance on spurious correlations. Predictions are skewed away from minority labels and dominated by frequent labels, making outcomes less fair [7]. In addition, sequential example order reinforces superficial cues as models attend to patterns that are less task specific. Researchers have shown that by strategically rearranging and highlighting challenging examples, models can better focus on meaningful patterns, and less on spurious correlations [8].

2.2 Methods to Avoid Spurious Correlations

Data augmentation and balancing methods effectively reduce spurious correlations by modifying datasets to improve model robustness. Data augmentation by counterfactual examples changes some phrases in examples while maintaining their labels, forcing models to concentrate on essential features instead of superficial patterns [9]. Concept level rebalancing also injects counterfactual examples to mitigate imbalances in concept label associations and thus mitigate spurious patterns in learned representations [10]. Data generation techniques provide high-quality examples that meet the label constraints but do not contain features that can lead to shortcuts [11]. Combined, these strategies alter the data properties and reduce the model's exposure to confounding factors.

These methods are complemented by training objectives and innovative labeling strategies that address spurious correlation during learning. Regularization techniques penalize effects learned in a model from deviating from true associations (i.e. enforcing that models internalize meaningful relationships), thereby

discouraging reliance on spurious features [12]. Secondly, label encoding strategies like soft label encoding transforms labels into probabilistic distributions to guide models to better generalization and less shortcut [13]. Moreover, feature-label correlation analysis detects and penalizes spurious patterns during training and forces models to attend to meaningful associations rather than spurious ones [14]. These approaches improve the training process to be robust, and reduce the presence of vulnerabilities to spurious correlations.

2.3 Quantifying Approaches in Current Literature

In order to increase robustness and fairness of NLP models, current literature has tried to quantify spurious correlations and biases in NLP models. In one study, it was measured how much models rely on irrelevant features by labelling texts with concepts and comparing accuracy between subsets of texts with and without these concepts, and show that data rebalancing can reduce such biases [10]. In another study, causal inference by masking was applied and each feature's necessity and sufficiency for prediction accuracy were evaluated in order to improve fairness and robustness across subgroups [15]. However, these approaches primarily regard models as black boxes by considering only input output behaviors, motivating the need for finer grained, feature level approaches to effectively counteract spurious correlations.

2.4 Existing Interpretability Methods

A number of interpretability methods aim to understand token importance in transformer models via attention mechanisms. Raw attention weights in **Attention Attribution** [16] are used to identify key tokens, but these weights do not reflect true causal contributions. **Attention Rollout and Flow** [17] aggregate attention hierarchies across layers to capture token interactions. Rollout recursively multiplies attention matrices for cumulative importance, and Flow treats attention as network capacities to compute maximum flow paths from input to output. However, these techniques lack the ability to accurately trace token level influences in deeper layers.

Other approaches completely bypass attention, using gradients, relevance propagation or game theoretic methods to provide more robust explanations. Token importance is calculated by **Integrated Gradients** [18] by integrating gradients from a baseline to the input, while ensuring sensitivity and consistency. **Layer Wise Relevance Propagation** [19] propagates output relevance back through layers, preserving attribution accuracy. Similarly, **SHAP** [20] computes Shapley values in a game theory approach to determine token contributions by looking at all possible input subsets. These methods build on attention based methods with stronger theoretical grounding and causal interpretability.

DecompX [21] is a significant step forward in that it decomposes token embeddings and propagates them layer by layer through the model. This way of doing it avoids token mixing and keeps individual contributions intact through the entire process. In contrast to existing methods, DecompX integrates all encoder components including the nonlinear feed-forward networks and the classification head to match explanations with the model's computational design. DecompX is shown to consistently outperform existing gradient based and vector based approaches on various datasets according to the standard faithfulness evaluations.

3 Problem Formulation

The project aims to quantify the extent to which pretrained language encoders rely on misaligned attributes in predictions. Addressing spurious correlations and biases in NLP models is critical to ensuring their safety, fairness, and scalability. This research establishes an evaluation framework to assess how well different encoder sizes and architectures prioritize relevant data attributes. The methodology involves defining metrics to evaluate performance on clean and augmented datasets, enabling a comprehensive assessment of robustness and alignment with human-annotated spans. The study seeks to advance transparency in model decisions and identify areas for improving the interpretability and reliability of NLP systems.

4 Proposed Method

To study spurious correlations in encoder models, we evaluate BERT and RoBERTa models of different sizes on sentiment analysis tasks. We measure the importance given to annotated spans compared to the whole sentence using DecompX, and define a ratio metric to quantify the emphasis on spans over other tokens. We first test this metric on consistent training and testing splits to see how well it aligns with human annotations. Then, we train the model on one span and evaluate its robustness to both, using a dataset with two conflicting spans that lead to opposite test results. In this approach, we analyze the model's focus on meaningful features and its resilience to changes in critical spans.

4.1 Datasets

The first dataset we chose to fine-tune our models on is Heegyu's toxic-spans dataset [22]. This dataset includes human-annotated sentences with a binary toxicity classification in addition to relevant toxic keys highlighted.

Probability	**String**	**Text of Post**	**Toxic**
{(86, 92): 0.66, (8, 13): 0.66}	{'stupid': 0.66, 'clown': 0.66}	Another clown in favour of more tax in this country. Blows my mind people can be this stupid.	1

Table 1: Toxic spans example sentences. Probability contains start and end position of toxic spans, with the probability they are toxic based on human labelling. String contains the toxic spans themselves with associated probabilities again. Text of post contains raw text of the sentences. Toxic column is the overall toxicity label, 1 if probability >0.5, 0 otherwise.

The second dataset is ACMI Lab's counterfactually augmented data [23]. This dataset contains sentence pairs with specific sentiment-changing words altered to flip the overall sentiment of the sentence as a whole.

Sentence	**Sentiment**
Long, boring, blasphemous. Never have I been so glad to see ending credits roll.	0
Long, fascinating, soulful. Never have I been so sad to see ending credits roll.	1

Table 2: CAD example sentences. Sentence contains the raw text input. Sentiment column indicates 0 for a negative statement and 1 for a positive statement.

4.2 Models

To perform supervised learning on each dataset, BERT and RoBERTa models of various sizes were chosen. The BERT models include: BERT-Tiny [24], BERT-Mini [25], BERT-Small [26], BERT-Medium [27], BERT-Base [28], and BERT-Large [29]. The RoBERTa models include: RoBERTa-Medium [30],

RoBERTa-Base [31], and RoBERTa-Large [32]. Further details about our model training strategies, model evaluation evaluation metrics and results, and our hyperparameter optimization approach can be found at the appendix.

5 Experiments/Results

For our setup, we finetuned BERT and RoBERTa models of different sizes on the Toxic Spans dataset and the original sentences segment of the CAD dataset, with the details of model performances can be seen in the appendix. We evaluated model alignment with human annotated spans and robustness to data changes using the DecompX framework. We first evaluated how well the models prioritized annotated spans in the Toxic Spans dataset, indicating how closely they followed human annotations. Second, we applied our metric to the CAD test set to examine adaptability, by comparing importance scores for original and augmented sentences where modified spans reversed sentiment. After our experiments, three main conclusions were made. Importance ratio scores correlated with model accuracies, RoBERTa models had wider spreads of importances given to tokens, and models have shown consistent ratios for original and augmented spans which can be attributed to a degree of robustness in the trained models.

5.1 Correlation of Importance Scores with Model Accuracies

The results of the Token Importance Ratio across various BERT models are presented in the chart below. The trend suggests that models which assign higher importance to specific token spans are more effective at identifying toxic spans in sentences. This correlation has two main explanations.

The first can be described as the **Token Focus Hypothesis**. Models with higher importance ratios tend to focus more on key spans of text that contribute to toxicity classification, thus aligning better with human annotations.

The second reason for the correlation can be contributed to **Over-generalization in Larger Models**. Larger models like BERT-Large may spread token importance across entire sentences, potentially reducing the ability to pinpoint specific toxic spans.

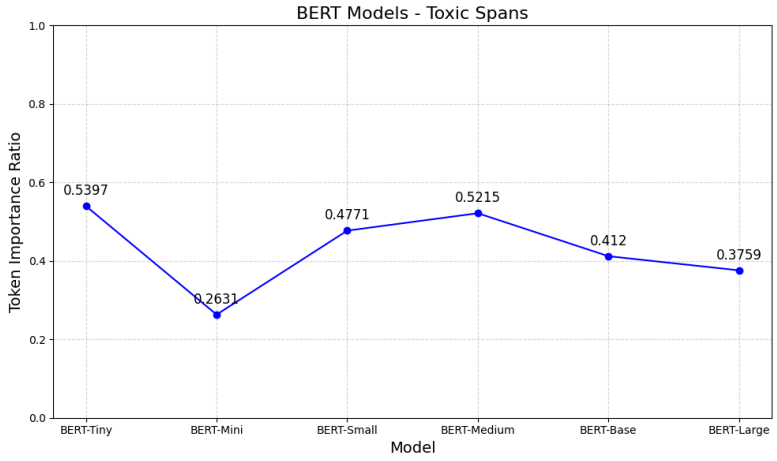


Figure 1: The Token Importance Ratio for predictions on the Toxic-spans dataset.

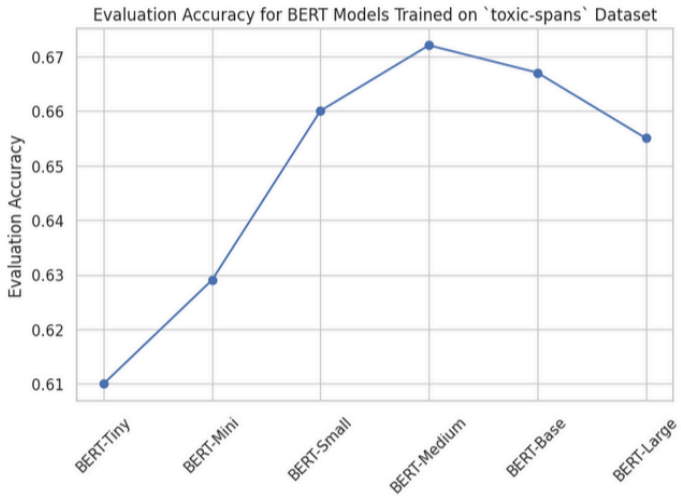


Figure 2: The evaluation accuracy for BERT models in predicting toxicity of text in the toxic_spans dataset.

The observed correlation between token importance ratios and model accuracy highlights the importance of token-level interpretability in toxicity detection tasks. Models that distribute importance more precisely onto relevant spans demonstrate superior performance. This reinforces the value of attention analysis tools like DecompX in identifying the alignment between human-annotated spans and model predictions. The outlier in the data, BERT-Tiny's low accuracy, can be attributed to the extremely small size of the model when compared to common NLP models.

5.2 Token Importance Spread in Roberta

From the annotated toxic spans we have the following dictionary associating the highest importance tokens with their impact on the classification: {'Japanese': 0.333, 'Canadian': 0.667, 'flawed logic': 0.333}

As seen in the visual, the RoBERTa-Medium model does not focus its importance solely on these spans. Instead, token importance is distributed broadly across multiple tokens. For instance, tokens like "Japanese," "Canadian," and "flawed logic" are annotated as significant, yet the model assigns relatively small, scattered importance across other non-toxic tokens.

By spreading token importance too broadly across the sentence, the model becomes less effective at pinpointing toxic spans. This lack of focus results in lower importance ratios and poorer span detection performance compared to BERT models. RoBERTa models may rely more on holistic sentence-level representations, diluting the contribution of individual spans.

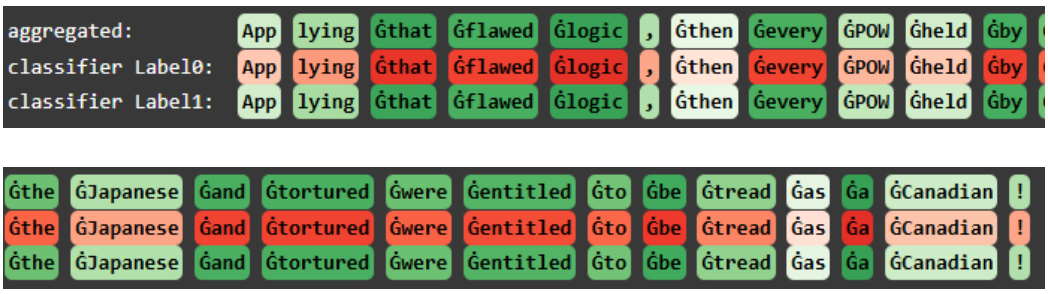


Figure 3: Token important visualized for an example text.

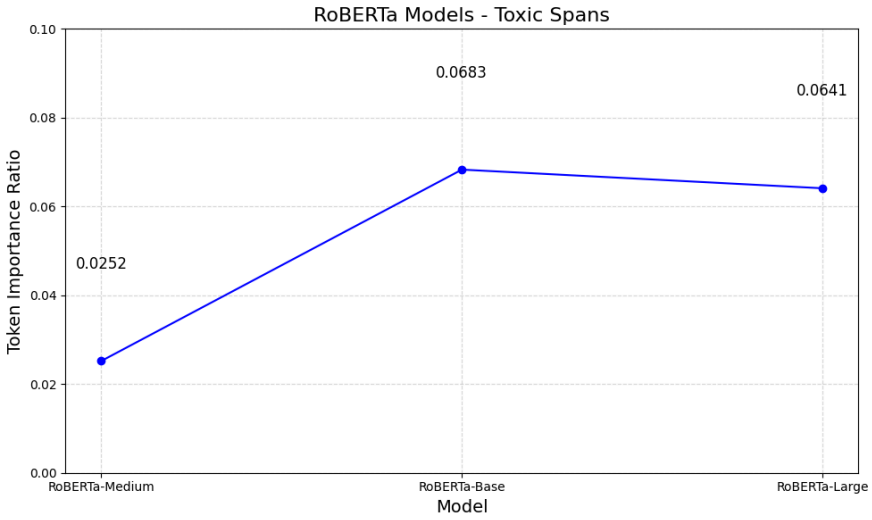


Figure 4: The evaluation accuracy for RoBERTa models in predicting toxicity of text in the toxic_spans dataset.

5.3 Robustness Against Spurious Correlations

The token importance ratios for original and augmented sentences are extremely similar across all BERT models. For example, BERT-Tiny shows a ratio of 0.1353 (Original) vs. 0.1343 (Augmented), while BERT-Large maintains ratios of 0.1637 (Original) vs. 0.1609 (Augmented). The small deviations in these values highlights the robustness of the models.

This ability to perform well even when tested on counterfactually augmented data suggests that training BERT models with well-tuned parameters enables them to adapt to variations in sentence sentiment while maintaining similar performance. It shows that the models can generalize beyond the specific sentences and patterns seen during training, a critical aspect of a model for any real world application.

Among the BERT variants, BERT-Medium and BERT-Large exhibit the highest token importance for both original and augmented sentence lists, leading to higher importance ratios. This suggests that larger BERT models are better at capturing token-level importance, even when sentence modifications are introduced.

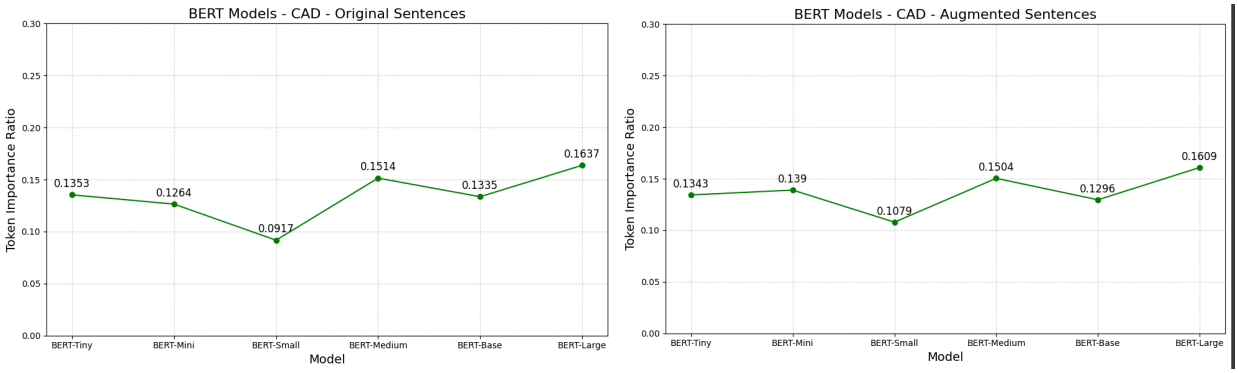


Figure 5: The evaluation accuracy for RoBERTa models in predicting toxicity of text in the toxic_spans dataset

Conclusion

This study used DecompX to detect the effects of spurious correlations on pretrained encoder-based NLP models. By conducting token-level analyses on BERT and RoBERTa models of various sizes, we quantified token importances. This led to evaluations of model robustness on human-annotated spans and counterfactually augmented examples. Our findings provide valuable insights into how spurious correlations manifest in model predictions and how well models align with human reasoning.

Our first key finding is that token importance correlates with model accuracy. We observed that models assigning higher importance to annotated spans demonstrated better alignment with ground truth labels. The higher importance resulted in improved accuracy on toxicity detection tasks. This suggests that token-level focus on meaningful spans is a strong predictor of model performance.

The second important conclusion is that RoBERTa models exhibit larger spread in the token importance distribution compared to similar-sized BERT models. This means that they distribute attention more diversely across tokens. This behavior may enhance robustness by reducing overreliance on individual spurious features and helps avoid overgeneralization.

Finally, we discovered a significant consistency across counterfactual examples. When evaluating models on the CAD dataset, we found that importance ratios remained relatively stable between original and sentiment-flipped spans. This consistency indicates robustness to CAD, as the models continued to emphasize the correct influential tokens even under sentence augmentation.

Our results demonstrate that pretrained encoder models can effectively prioritize relevant spans while maintaining a level of robustness to counterfactual variations. However, the presence of spurious correlations persists, especially in models where importance distributions are concentrated on superficial patterns. By quantifying these effects, our work highlights the need for interpretability-driven evaluations and bias mitigation strategies in NLP systems.

Future Work

Creating new datasets with high-quality span annotations, particularly in underrepresented domains, can significantly expand the applicability of the DecompX framework. Incorporating support for additional encoder architectures, such as DistilBERT and ALBERT, would enable broader evaluations of model behavior across diverse models. Applying the framework to datasets from varied domains and languages would provide deeper insights into domain-specific spurious correlations. In addition, dynamical training approaches such as reinforcement learning, or curriculum learning could make the models attend to important features and indifference to spurious correlations.

References

[1] G. Gauthier-Melancon, O. Marquez Ayala, L. Brin, C. Tyler, F. Branchaud-Charron, J. Marinier, K. Grande, and D. Le, “Azimuth: Systematic error analysis for text classification,”. [Online]. Available: https://aclanthology.org/2022.emnlp-demos.30.

[2] Y. Wang, et al., “Should we rely on entity mentions for relation extraction? Debiasing relation extraction with counterfactual analysis,” *arXiv preprint arXiv:2205.03784*, 2022. [Online]. Available: https://arxiv.org/abs/2205.03784

[3] Y. Wahba, N. Madhavji, and J. Steinbacher, “Attention Is Not Always What You Need: Towards Efficient Classification of Domain-Specific Text: Case-Study: IT Support Tickets,” in *Science and Information Conference*, Cham, Switzerland: Springer Nature, 2023.

[4] Z. Zhu, et al., "Quantifying the task-specific information in text-based classifications," arXiv preprint arXiv:2110.08931, 2021. [Online]. Available: https://arxiv.org/abs/2110.08931

[5] T. Niven and H.-Y. Kao, "Probing neural network comprehension of natural language arguments," arXiv preprint arXiv:1907.07355, 2019. [Online]. Available: https://arxiv.org/abs/1907.07355

[6] G. Wang, et al., "Joint embedding of words and labels for text classification," arXiv preprint arXiv:1805.04174, 2018. [Online]. Available: https://arxiv.org/abs/1805.04174

[7] L. Song, et al., "Adaptive ranking-based sample selection for weakly supervised class-imbalanced text classification," arXiv preprint arXiv:2210.03092, 2022. [Online]. Available: https://arxiv.org/abs/2210.03092

[8] Y. Yaghoobzadeh, et al., "Increasing robustness to spurious correlations using forgettable examples," arXiv preprint arXiv:1911.03861, 2019. [Online]. Available: https://arxiv.org/abs/1911.03861

[9] A. Wang and O. Russakovsky, "Overwriting pretrained bias with finetuning data," Proc. IEEE/CVF Int. Conf. Comput. Vis., 2023.

[10] C. Cheng, et al., "Exploring the robustness of in-context learning with noisy labels," arXiv preprint arXiv:2404.18191, 2024. [Online]. Available: https://arxiv.org/abs/2404.18191.

[11] I. Sen, M. Samory, C. Wagner, and I. Augenstein, "Counterfactually augmented data and unintended bias: The case of sexism and hate speech detection," in Proc. 2022 Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol., Seattle, WA, USA, 2022, pp. 4716–4726.

[12] Y. Zhou, et al., "Explore spurious correlations at the concept level in language models for text classification," arXiv preprint arXiv:2311.08648, 2023. [Online]. Available: https://arxiv.org/abs/2311.08648.

[13] Y. Wu, et al., "Generating data to mitigate spurious correlations in natural language inference datasets," arXiv preprint arXiv:2203.12942, 2022. [Online]. Available: https://arxiv.org/abs/2203.12942.

[14] P. Bansal and A. Sharma, "Controlling learned effects to reduce spurious correlations in text classifiers," arXiv preprint arXiv:2305.16863, 2023. [Online]. Available: https://arxiv.org/abs/2305.16863.

[15] Z. He, H. Deng, H. Zhao, N. Liu, and M. Du, "Mitigating shortcuts in language models with soft label encoding," arXiv preprint arXiv:2309.09380, 2023. [Online]. Available: https://arxiv.org/abs/2309.09380.

[16] Y. Hao, L. Dong, F. Wei, and K. Xu, "Self-attention attribution: Interpreting information interactions inside transformer," arXiv preprint arXiv:2004.11207, 2021. [Online]. Available: https://arxiv.org/abs/2004.11207.

[17] S. Abnar and W. Zuidema, "Quantifying attention flow in transformers," arXiv preprint arXiv:2005.00928, 2020. [Online]. Available: https://arxiv.org/abs/2005.00928.

[18] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," arXiv preprint arXiv:1703.01365, 2017. [Online]. Available: https://arxiv.org/abs/1703.01365.

[19] A. Binder, G. Montavon, S. Bach, K.-R. Müller, and W. Samek, “Layer-wise relevance propagation for neural networks with local renormalization layers,” arXiv preprint arXiv:1604.00825, 2016. [Online]. Available: https://arxiv.org/abs/1604.00825.

[20] S. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” arXiv preprint arXiv:1705.07874, 2017. [Online]. Available: https://arxiv.org/abs/1705.07874.

[21] A. Modarressi, et al., “DecompX: Explaining transformers decisions by propagating token decomposition,” arXiv preprint arXiv:2306.02873, 2023. [Online]. Available: https://arxiv.org/abs/2306.02873

[22] H. Kim, "Heegyu/Toxic-Spans · Datasets at Hugging Face," Hugging Face. [Online]. Available: https://huggingface.co/datasets/heegyu/toxic-spans. Accessed: Nov. 22, 2024.

[23] ACMI-Lab, "ACMI-Lab/Counterfactually-Augmented-Data: Learning the Difference That Makes a Difference with Counterfactually-Augmented Data," GitHub. [Online]. Available: https://github.com/acmi-lab/counterfactually-augmented-data. Accessed: Nov. 25, 2024.

[24] "Lyeonii/Bert-Tiny · Hugging Face," Hugging Face. [Online]. Available: https://huggingface.co/lyeonii/bert-tiny. Accessed: Nov. 15, 2024.

[25] "Lyeonii/Bert-Mini · Hugging Face," Hugging Face. [Online]. Available: https://huggingface.co/lyeonii/bert-mini. Accessed: Nov. 15, 2024.

[26] "Lyeonii/Bert-Small · Hugging Face," Hugging Face. [Online]. Available: https://huggingface.co/lyeonii/bert-small. Accessed: Nov. 15, 2024.

[27] "Lyeonii/Bert-Medium · Hugging Face," Hugging Face. [Online]. Available: https://huggingface.co/lyeonii/bert-medium. Accessed: Nov. 15, 2024.

[28] "Google-Bert/Bert-Base-Uncased · Hugging Face," Hugging Face. [Online]. Available: https://huggingface.co/google-bert/bert-base-uncased. Accessed: Nov. 15, 2024.

[29] "Google-Bert/Bert-Large-Uncased · Hugging Face," Hugging Face. [Online]. Available: https://huggingface.co/google-bert/bert-large-uncased. Accessed: Nov. 15, 2024.

[30] "JackBAI/Roberta-Medium · Hugging Face," Hugging Face. [Online]. Available: https://huggingface.co/JackBAI/roberta-medium. Accessed: Nov. 17, 2024.

[31] "FacebookAI/Roberta-Base · Hugging Face," Hugging Face. [Online]. Available: https://huggingface.co/FacebookAI/roberta-base. Accessed: Nov. 18, 2024.

[32] "FacebookAI/Roberta-Large · Hugging Face," Hugging Face. [Online]. Available: https://huggingface.co/FacebookAI/roberta-large. Accessed: Nov. 18, 2024.

Appendix:

Model Training Strategies: Training was orchestrated using the Trainer API from Hugging Face. Training was conducted over three epochs. To enhance computational efficiency, mixed precision (FP16) training and gradient checkpointing were employed. Regularization was achieved through a weight decay factor of 0.01. Training arguments were configured to save model checkpoints and evaluate performance at the end of each epoch, ensuring iterative improvements and safeguarding against overfitting.

Model Evaluation: The training process involved the use of the Trainer object to manage the optimization loop and evaluate the model periodically. Evaluation metrics included accuracy, precision, recall, and F1 score, calculated using the precision_recall_fscore_support and accuracy_score methods. These metrics provided a comprehensive understanding of the model's performance, particularly on imbalanced datasets. Regular logging was implemented to monitor training progress and ensure transparency.

Hyperparameter Optimization: To identify the optimal configuration for model performance, a grid search was conducted over a range of learning rates and batch sizes. Each configuration was evaluated using the same training and validation processes, with performance measured by the F1 score. The best-performing model was selected and saved for further use. CUDA optimizations, including cache clearing, were employed during hyperparameter search to maximize computational efficiency.

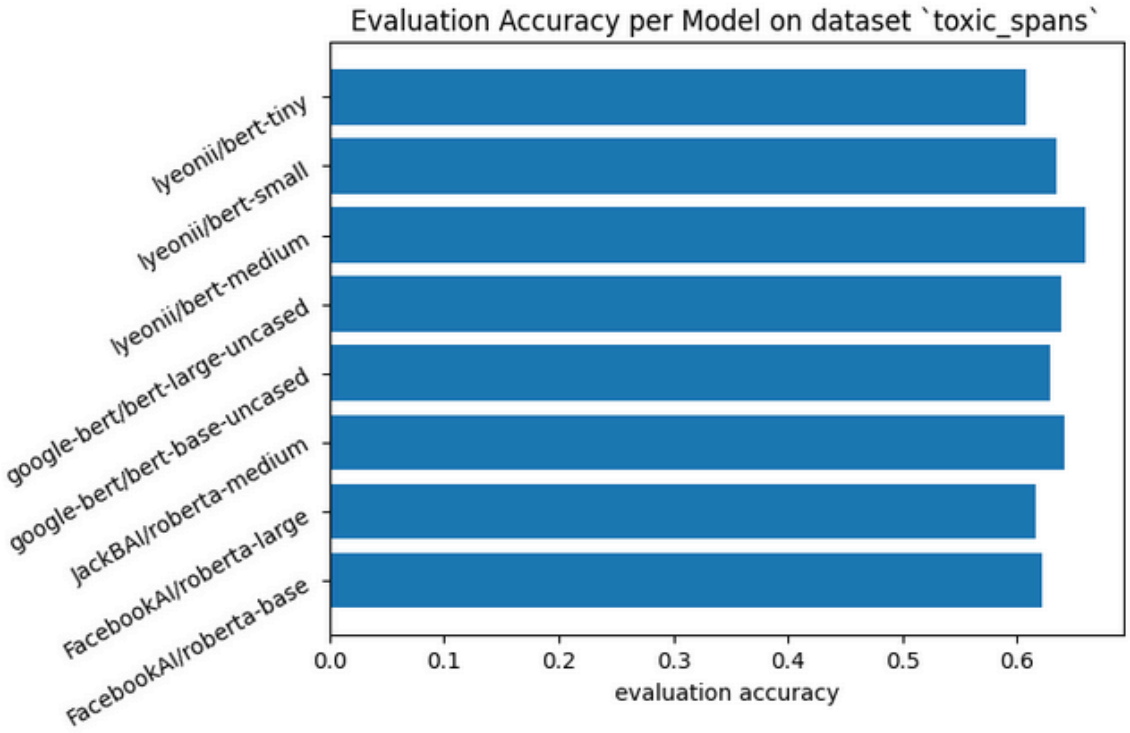


Figure 1: Evaluation accuracy for NLP models predicting the toxicity of input text.

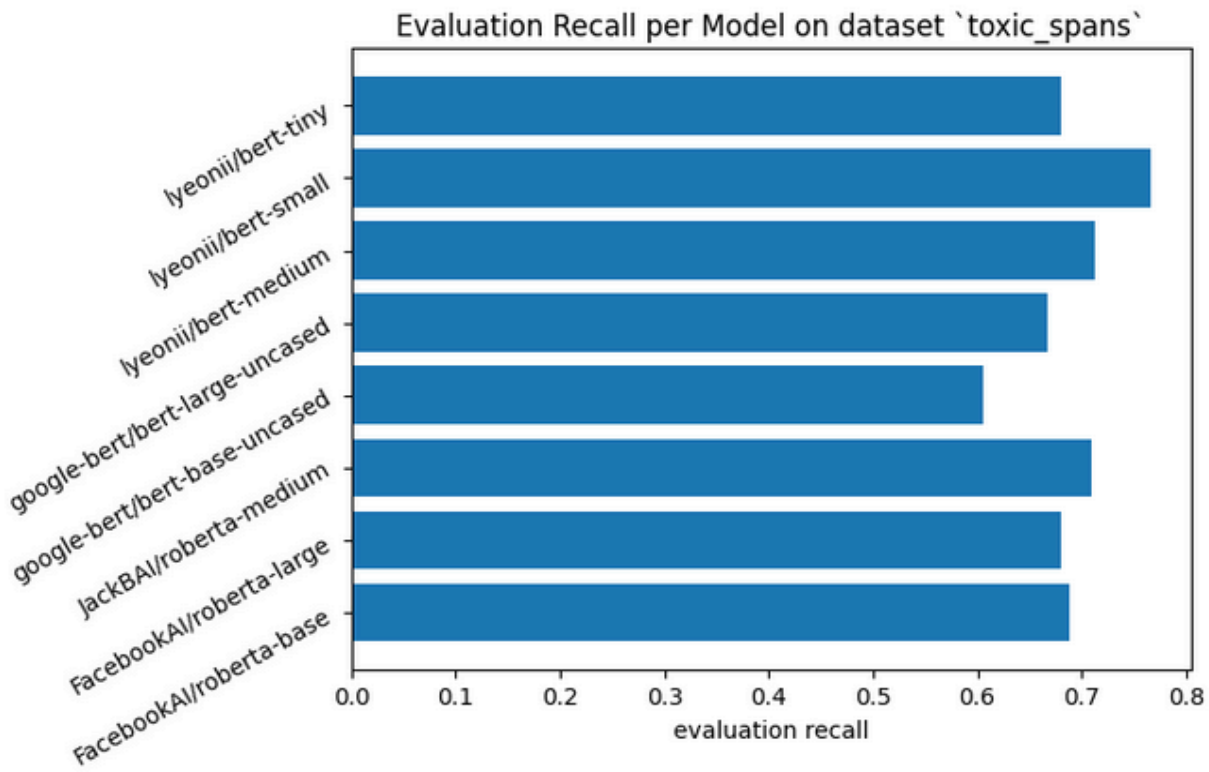


Figure 2: Evaluation recall for NLP models predicting the toxicity of input text.

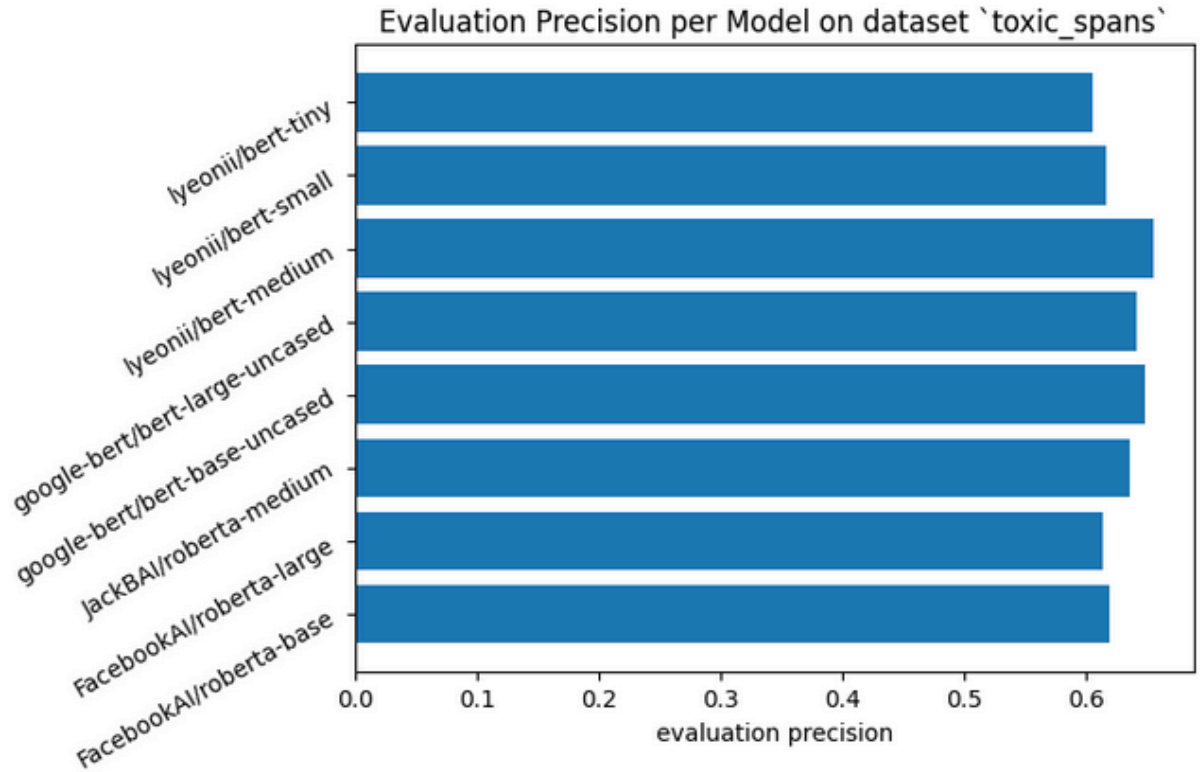
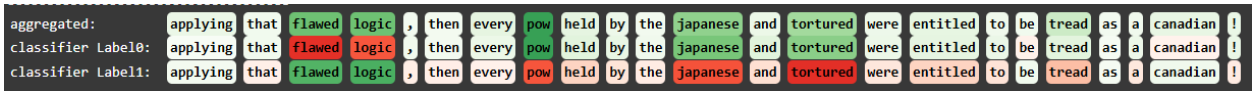
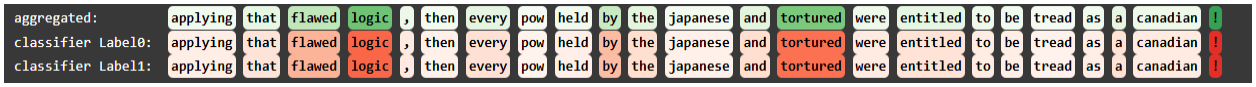


Figure 3: Evaluation precision for NLP models predicting the toxicity of input text.

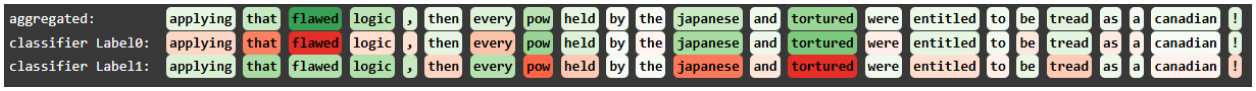
Bert tiny



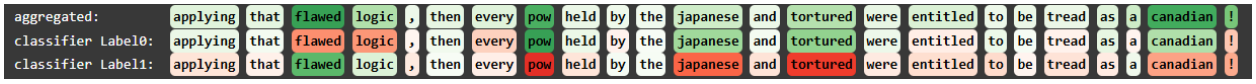
Bert mini



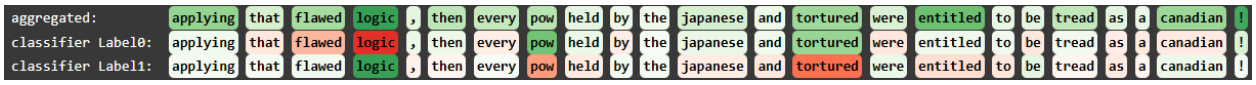
Bert small



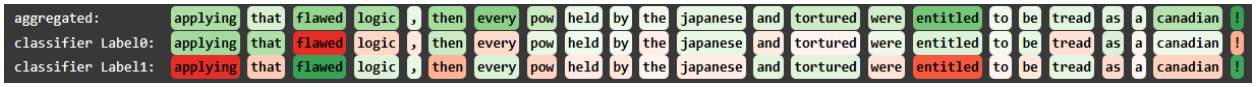
Bert medium



Bert base



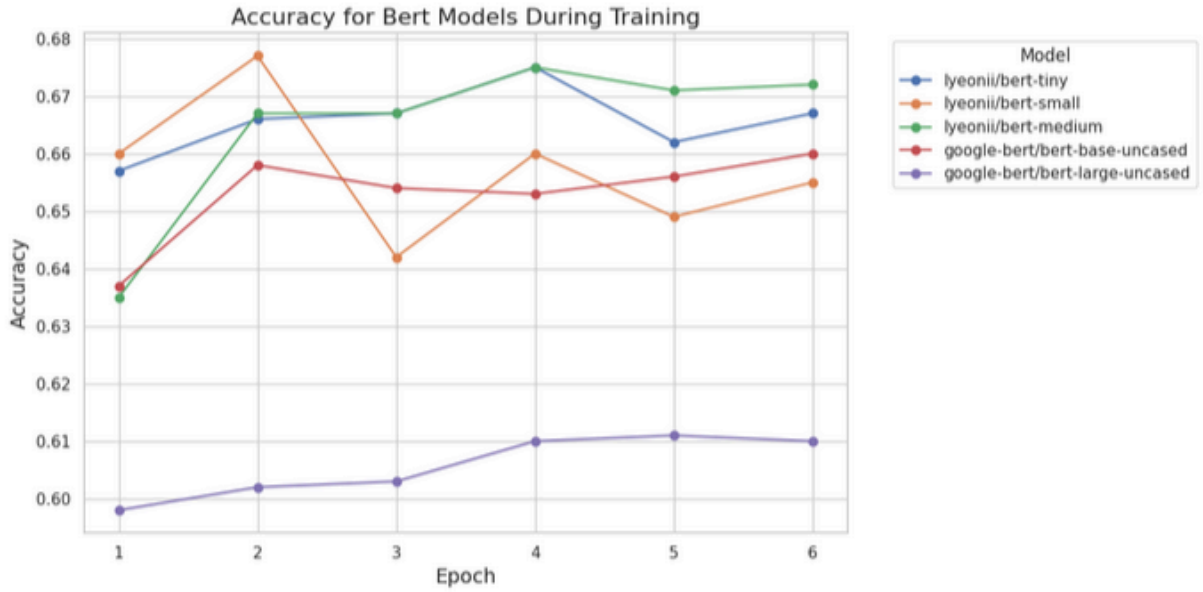
Bert large

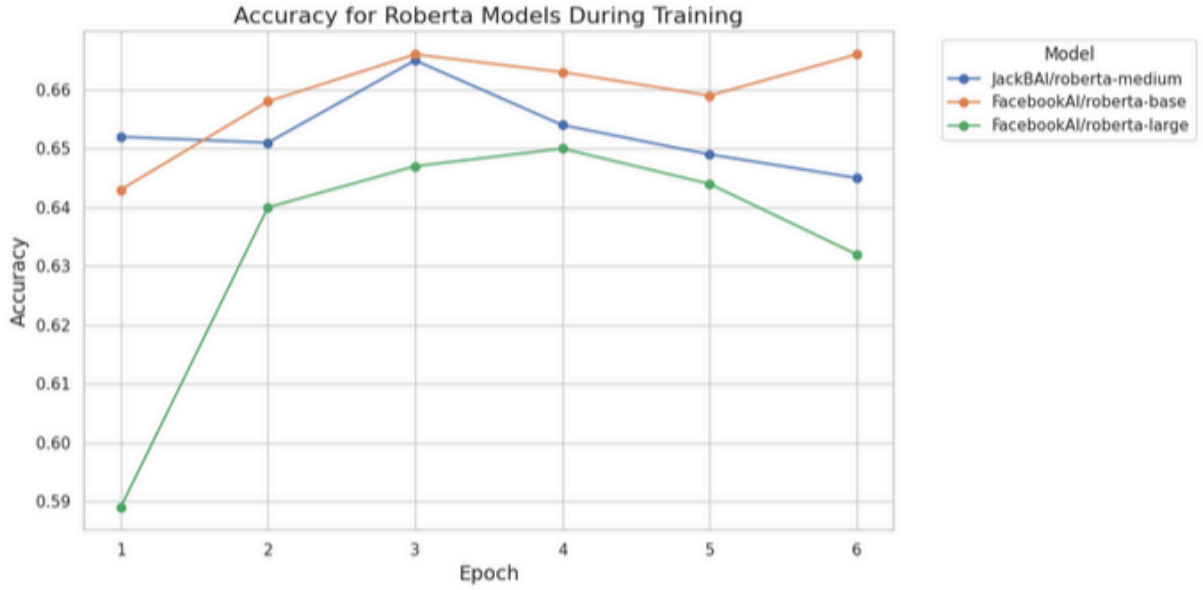


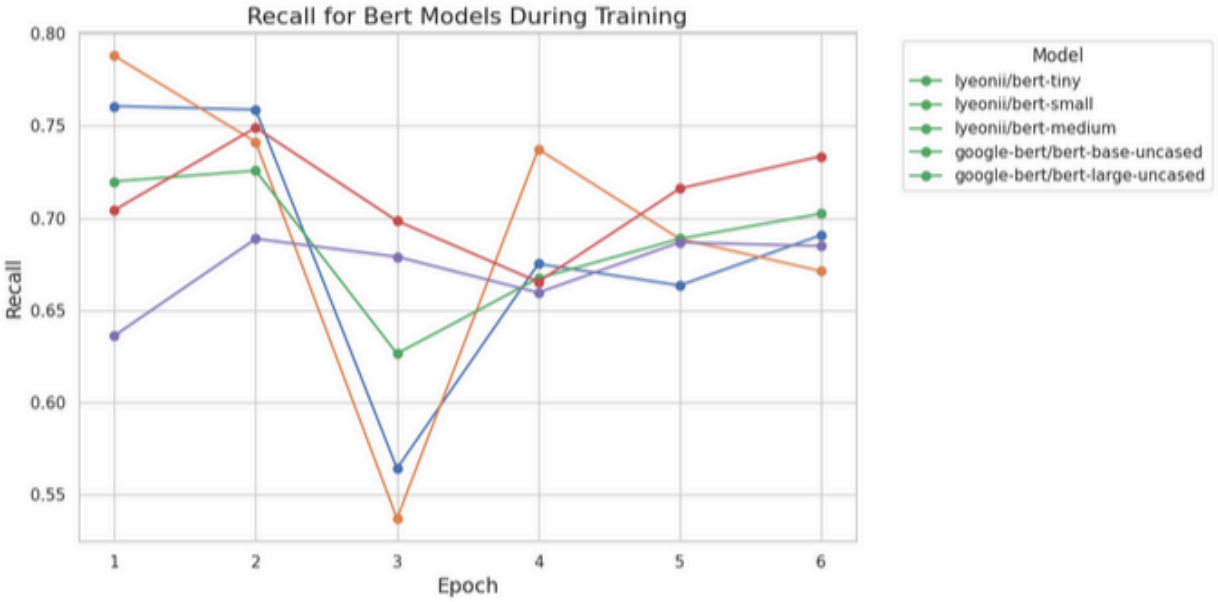
Roberta medium

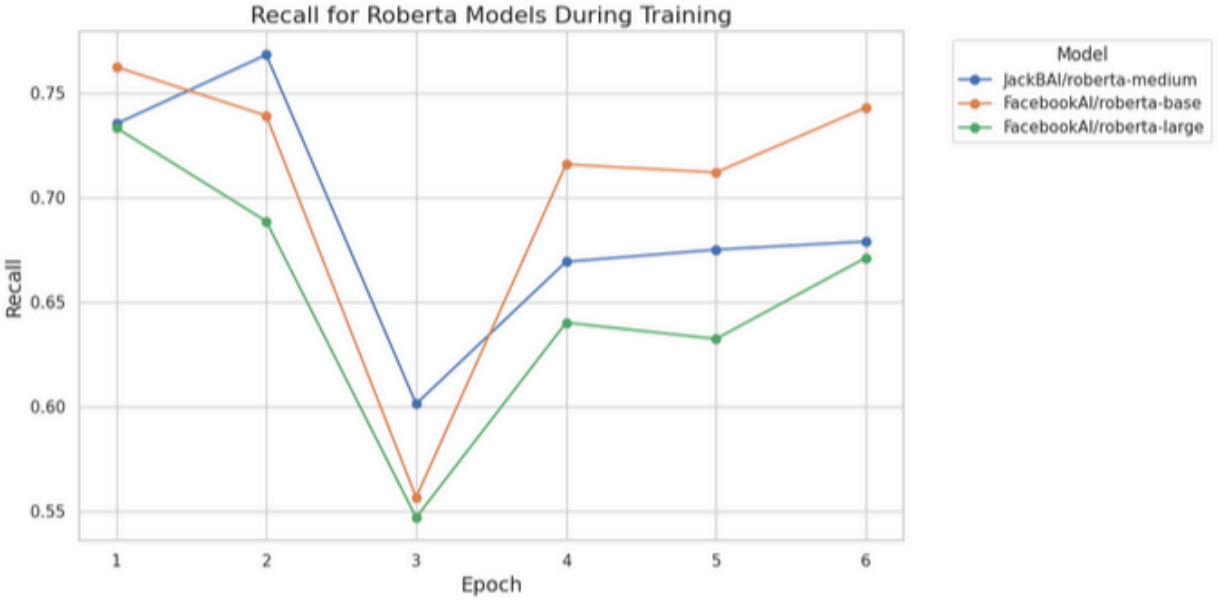
Roberta base

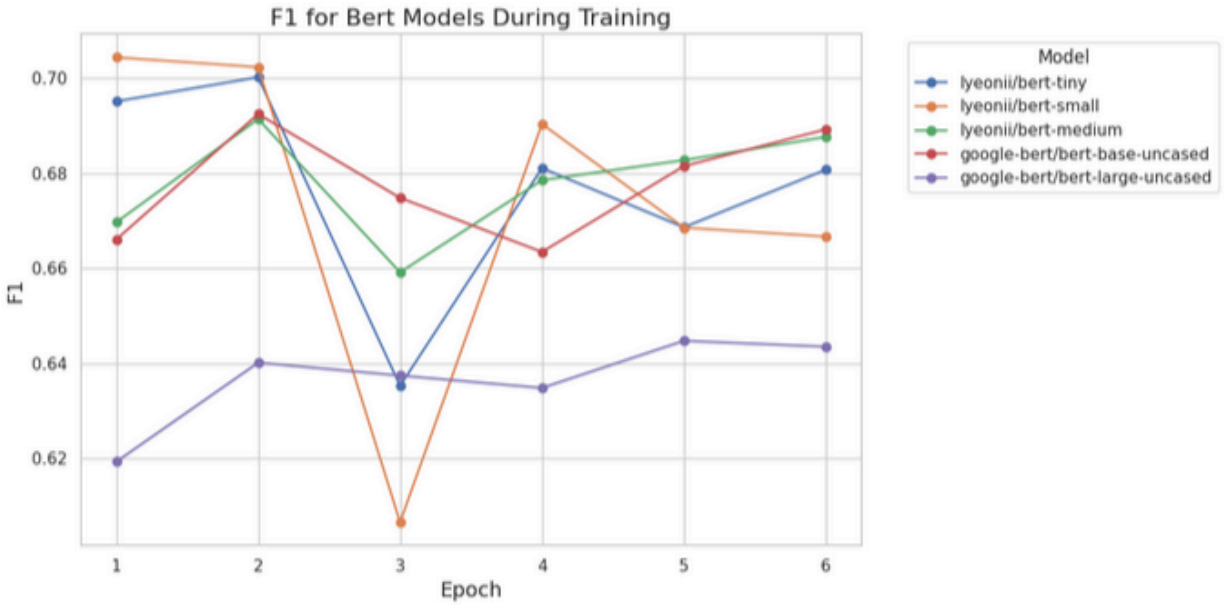
Roberta large

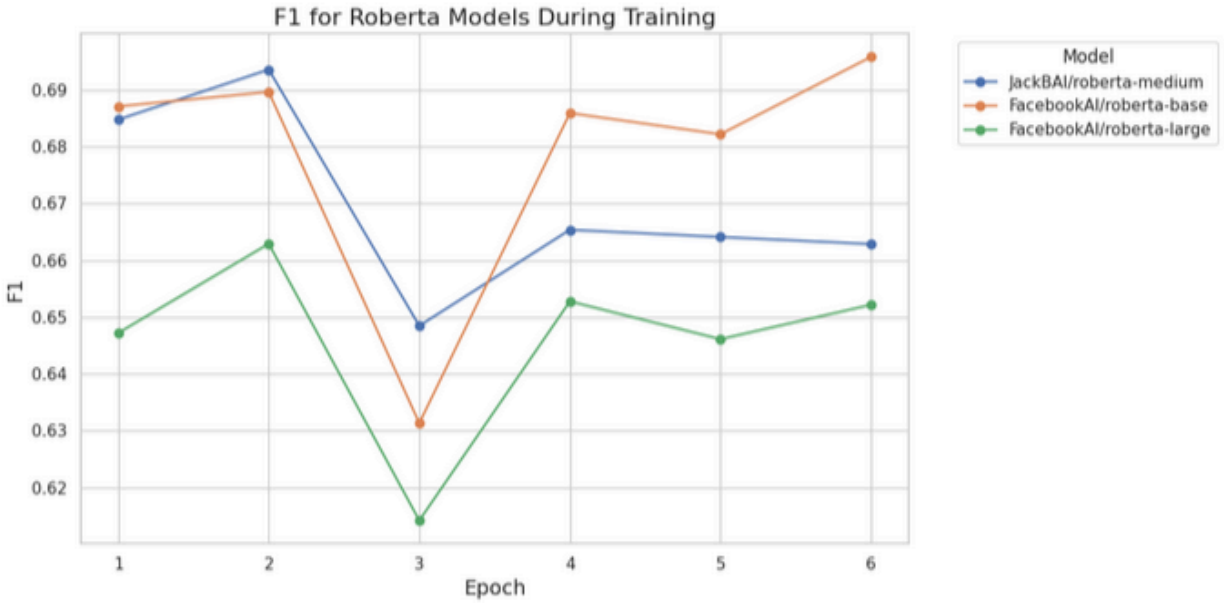

Accuracy for Bert Models During Training
Accuracy
Epoch
Model
lyeonii/bert-tiny
lyeonii/bert-small
lyeonii/bert-medium
google-bert/bert-base-uncased
google-bert/bert-large-uncased

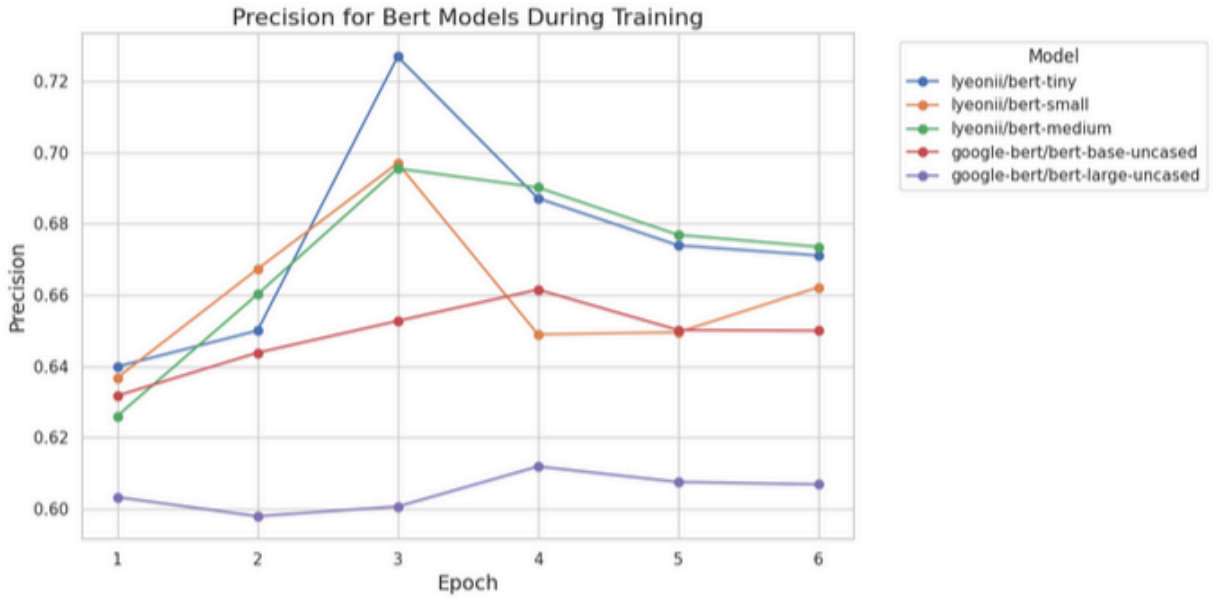

Accuracy for Roberta Models During Training
Accuracy
Epoch
Model
JackBAI/roberta-medium
FacebookAI/roberta-base
FacebookAI/roberta-large

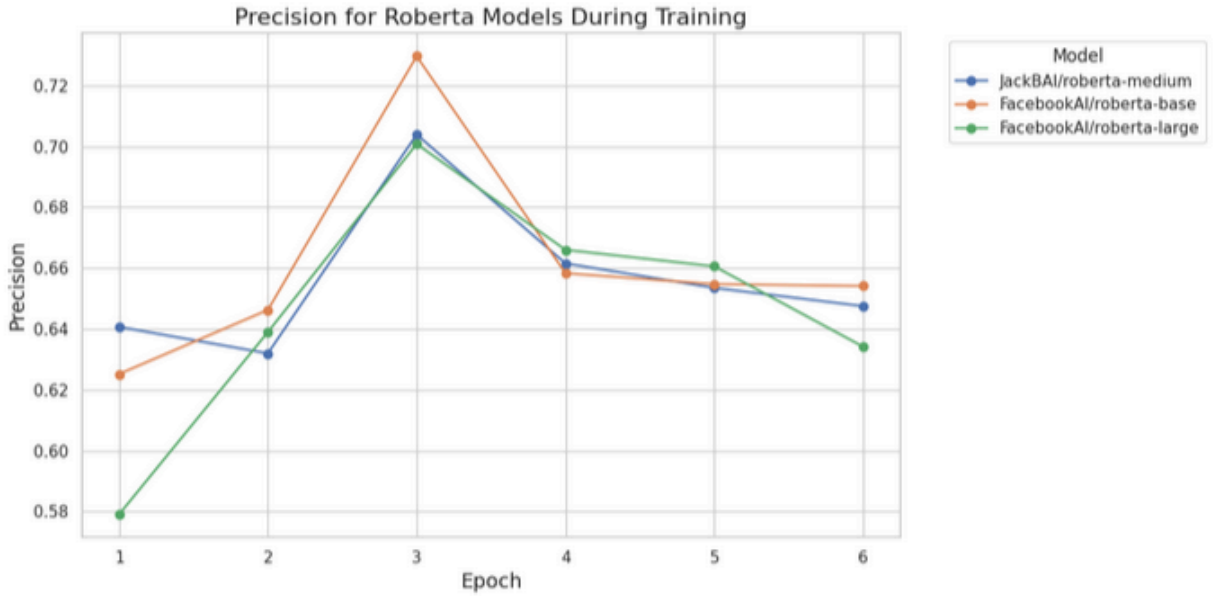

Recall for Bert Models During Training
Recall
Epoch
0.80
0.75
0.70
0.65
0.60
0.55
1
2
3
4
5
6
Model
lyeonii/bert-tiny
lyeonii/bert-small
lyeonii/bert-medium
google-bert/bert-base-uncased
google-bert/bert-large-uncased


Recall for Roberta Models During Training
Recall
Epoch
0.75
0.70
0.65
0.60
0.55
1
2
3
4
5
6
Model
JackBAI/roberta-medium
FacebookAI/roberta-base
FacebookAI/roberta-large


F1 for Bert Models During Training
F1
Epoch
0.70
0.68
0.66
0.64
0.62
1
2
3
4
5
6
Model
lyeonii/bert-tiny
lyeonii/bert-small
lyeonii/bert-medium
google-bert/bert-base-uncased
google-bert/bert-large-uncased


F1 for Roberta Models During Training
F1
Epoch
0.69
0.68
0.67
0.66
0.65
0.64
0.63
0.62
1
2
3
4
5
6
Model
JackBAI/roberta-medium
FacebookAI/roberta-base
FacebookAI/roberta-large


Precision for Bert Models During Training
Precision
Epoch
Model
lyeonii/bert-tiny
lyeonii/bert-small
lyeonii/bert-medium
google-bert/bert-base-uncased
google-bert/bert-large-uncased


Precision for Roberta Models During Training
Precision
Epoch
Model
JackBAI/roberta-medium
FacebookAI/roberta-base
FacebookAI/roberta-large