Are modern text encoders more robust to textual noise?

Cocheci Cristiana
University of Bucharest, Romania
cristiana.cocheci@gmail.com

Apostol Ilie-Daniel
University of Bucharest, Romania
apostoldaniel854@gmail.com

Abstract

This paper investigates the robustness of modern text encoders to input perturbations by comparing BERT and ModernBERT on an IMDB review sentiment classification task. We evaluated the performance of the models when faced with typos, synonym replacements, and word deletions. Our experiments demonstrate that ModernBERT consistently maintains higher accuracy on perturbed inputs compared to BERT, exhibiting enhanced robustness.

1 Introduction

BERT (1) (Bidirectional Encoder Representations from Transformers) revolutionized NLP by leveraging bidirectional context and self-attention mechanisms to capture rich linguistic representations. However, BERT was found to often struggle with noisy inputs containing typos or grammatical errors (2).

ModernBERT (3), developed by Answer.ai (2023), introduces architectural improvements in tokenization, attention mechanisms, and training objectives that may enhance robustness to textual perturbations.

Our study investigates whether ModernBERT maintains higher accuracy than BERT when tested on perturbed inputs, specifically examining:

1. Performance comparison on clean training data but noisy test inputs
2. Effects of noise-aware training on model robustness
3. Generalization capabilities across different noise levels

We evaluate both models on a classification task using perturbed text, providing both quantitative results and qualitative analyses of how these models process noisy text.

2 Implementation

2.1 Dataset and Perturbation Methods

For our experiments, we utilized the IMDB movie review dataset (4), a sentiment analysis dataset consisting of user reviews with binary sentiment labels (positive/negative). The dataset was chosen for its real-world applicability and the natural presence of linguistic variations.

We employed three distinct perturbation methods to evaluate model robustness: typo generation, synonym replacing and word dropout. Each perturbed example has 20% of the words perturbed. There are also two types of perturbed train datasets:

1. All perturbations: all reviews are perturbed with one of the three perturbations methods chosen randomly.
2. Typo perturbations: Five train datasets where 2% / 5% / 10% / 20% of reviews are perturbed

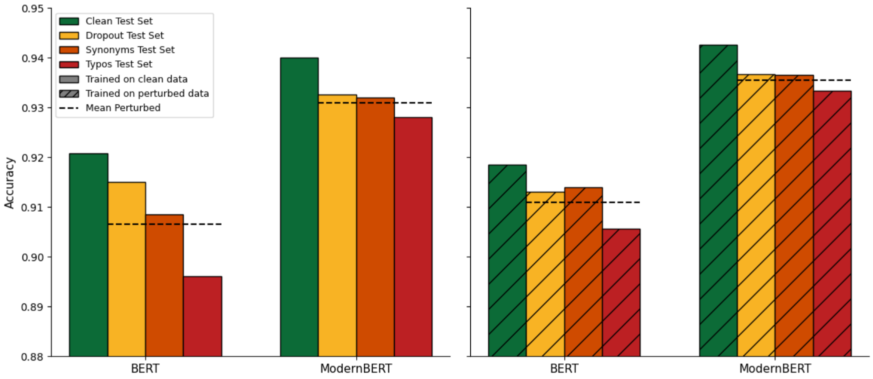


Figure 1: Performance comparison across different perturbation types for models trained on clean data vs. perturbed data

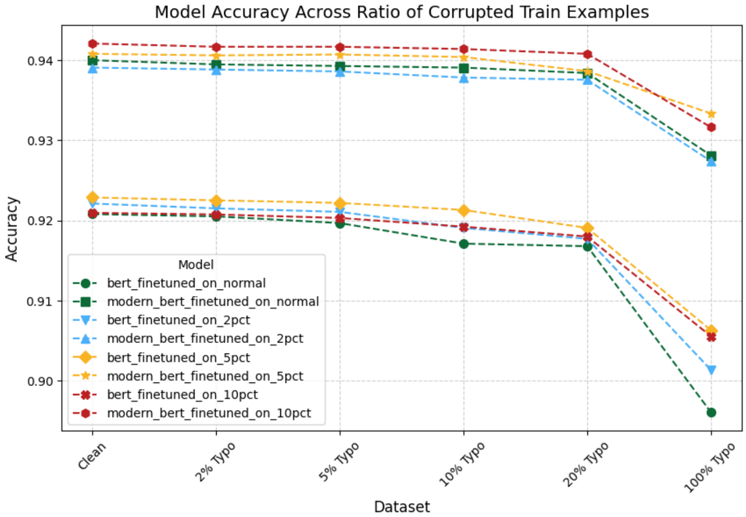


Figure 2: Performance comparison between BERT and ModernBERT on clean vs. perturbed test data

by typos and the other samples are clean.

2.2 Model Finetuning

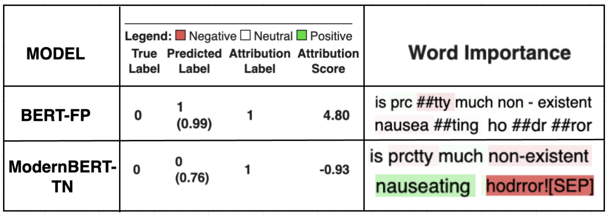
Using the transformers library from Hugging Face we have taken the "bert-base-uncased" and "answerdotai/ModernBERT-base" pretrained models and finetuned them on our custom datasets. We passed the original text through the matching pretrained tokenizers from the same library and then trained on the *train - validation* split (80%-20%) for 3 epochs.

3 Results

1. Figure 1 compares performance across perturbation types with models trained on clean (left) versus perturbed data (right). ModernBERT performs better in all scenarios, with its biggest advantage seen with typos. It also degrades less on perturbed inputs. Moreover, noise-aware training (right side) closes performance gaps for both models on perturbed data.
2. Figure 2 shows ModernBERT consistently outperforming BERT across all typo corruption levels. Training with just 2-5% noise significantly improves both models' resilience to corruption, with ModernBERT maintaining its advantage even at 100% corruption.

4 Qualitative analysis

We have chosen a negative review from the test set manipulated with typos. It was correctly classified by ModernBERT but misclassified by BERT; both models trained on clean data. For understanding the underlying decision processes, we used (5) Captum, a model interpretability library from PyTorch. This tool allows us to visualize the influence of individual words on the model's predictions, with red highlighting negative and green positive sentiment. Figure 3 illustrate key differences between the two models. Notably, ModernBERT identifies terms such as *non-existent* and *hodrorr*(typo for horror) as conveying negative sentiment, whereas BERT fails to do so. Even though some negative words such as *nauseating* have a positive attribution score for ModernBERT, they have a small overall impact on the sentence attribution score, leading to a correct classification of the review.



Legend: 🟥 Negative □ Neutral 🟩 Positive

MODEL	True Label	Predicted Label	Attribution Label	Attribution Score	Word Importance
BERT-FP	0	1 (0.99)	1	4.80	is prc ##tty much non - existent nausea ##ting ho ##dr ##ror
ModernBERT-TN	0	0 (0.76)	1	-0.93	is prctty much non-existent nauseating hodrror![SEP]

Figure 3: Modern Bert True Negative Vs Bert False Positive

5 Conclusions

ModernBERT demonstrates significantly greater robustness to various types of input perturbations compared BERT. Its performance degradation remains minimal when evaluated on datasets containing a moderate proportion of altered samples ($\leq$20%). Both models benefit from noise-aware training, with ModerBERT consistently outperforming BERT.

Our qualitative analysis reveals that ModernBERT can correctly predict examples where BERT fails. BERT sometimes fails to appropriately identify negative reviews by not being able to attribute importance to key class-defining words.

References

[1] Devlin, J., Chang, M.W., Lee, K., and Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. In Proceedings of NAACL-HLT 2019, pages 4171-4186.

[2] Si, C., Yang, Z., Cui, Y., Ma, W., Liu, T., & Wang, S. (2020). *Benchmarking Robustness of Machine Reading Comprehension Models*. arXiv preprint arXiv:2004.14004. https://arxiv.org/abs/2004.14004

[3] Answer.ai (2023). *ModernBERT: A Modern Approach to BERT Pretraining*. [Technical report]

[4] Maas, A.L., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., and Potts, C. (2011). *Learning Word Vectors for Sentiment Analysis*. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 142-150.

[5] Kokhlikyan, N., Miglani, V., Martin, M., Wang, E., Alsallakh, B., Reynolds, J., Melnikov, A., Kliushkina, N., Araya, C., Yan, S., & Reblitz-Richardson, O. (2020). *Captum: A unified and generic model interpretability library for PyTorch*. arXiv preprint arXiv:2009.07896. https://doi.org/10.48550/arXiv.2009.07896