A Performance Evaluation of BERT-Based Models for Text Classification Tasks

Author

Etimad Fadel

Department of Computer Science, King Abdulaziz University, Jeddah, Saudi Arabia

DOI: *https://doi.org/10.21590/v5i4.01*

Abstract

BERT (Bidirectional Encoder Representations from Transformers) has revolutionized natural language processing with its ability to capture contextual information through deep bidirectional representations. This paper evaluates BERT and its variants (DistilBERT and RoBERTa) on a suite of text classification tasks including sentiment analysis, topic classification, and spam detection. Datasets include IMDB reviews, AG News, and Enron spam email corpus. Models are fine-tuned with task-specific heads and compared to LSTM and CNN baselines. BERT outperforms all baselines, achieving 94.2% accuracy on IMDB and 96.8% on AG News. RoBERTa slightly surpasses BERT in most tasks but requires more training time and memory. DistilBERT offers competitive performance with 40% fewer parameters, making it suitable for edge deployments. We examine hyperparameter sensitivities, training stability, and inference latency across models. Results indicate that BERT's pretraining depth allows for greater generalization across diverse tasks with minimal tuning. However, resource requirements remain high, particularly in low-latency environments. This study affirms the dominance of transformer-based models in text classification while providing comparative insights into their trade-offs. Our analysis informs practitioners choosing between accuracy, speed, and computational cost when deploying BERT-like models in real-world NLP applications.

Keywords: *zero-day exploits, sandboxing, threat intelligence fusion, IOC matching, behavior analysis, ransomware detection, APT, malware analysis, AlienVault OTX, hybrid detection*

1. Introduction

The introduction of transformer-based models, particularly BERT (Bidirectional Encoder Representations from Transformers), has dramatically reshaped the landscape of natural language processing (NLP). By leveraging self-attention and deep bidirectional encoding, BERT has achieved state-of-the-art results in a wide range of language understanding tasks. Its architecture allows for contextual representations of words based on both left and right contexts, a significant improvement over traditional models like LSTMs and CNNs that process sequences in a fixed direction or use shallow encodings.

Despite BERT's impressive generalization capabilities, challenges remain in balancing accuracy, computational efficiency, and deployment feasibility—particularly for real-time or resource-constrained environments. Variants like RoBERTa, which employs optimized pretraining techniques, and DistilBERT, a distilled version of BERT, aim to address these trade-offs by altering training regimens or reducing model size while preserving performance.

This paper evaluates the effectiveness of BERT and its derivatives in text classification tasks including sentiment analysis, topic categorization, and spam detection. We benchmark their performance on IMDB, AG News, and Enron spam datasets, comparing them against traditional LSTM and CNN baselines. Beyond accuracy, we assess training stability, hyperparameter sensitivity, and inference latency, all of which are critical in real-world NLP applications. Our goal is to inform practitioners and researchers about the practical considerations of deploying BERT-based models depending on task requirements and system constraints.

2. Hypothesis

This study is driven by the following hypotheses:

- **H1:** BERT-based models (BERT, RoBERTa, DistilBERT) outperform LSTM and CNN baselines on standard text classification tasks.
- **H2:** RoBERTa achieves slightly higher accuracy than BERT due to improved pretraining but incurs a higher computational cost.
- **H3:** DistilBERT provides a favorable trade-off between performance and efficiency, making it suitable for edge or mobile deployments.
- **H4:** Hyperparameter tuning and fine-tuning epochs significantly influence performance, especially for RoBERTa due to its sensitivity to learning rate and batch size.
- **H5:** Transformer-based models require more inference time per sample, posing challenges for real-time classification use cases.

These hypotheses are tested across multiple datasets and task domains to ensure generalizability

3. Experimental Setup

3.1 Datasets

We selected three representative datasets for classification:

- IMDB Movie Reviews: Sentiment analysis (positive vs. negative), 50,000 samples.
- AG News: Topic classification (4 classes), 120,000 samples.
- Enron Spam Dataset: Spam detection, 33,000 emails (split 60/40 train/test).

All datasets were tokenized using the HuggingFace Transformers tokenizer for consistency and padded to a maximum sequence length of 256 tokens.

3.2 Models

We evaluated the following architectures:

- **BERT-base (uncased):** 12 layers, 768 hidden units, 110M parameters.
- **RoBERTa-base:** Optimized BERT with longer training, dynamic masking.
- **DistilBERT:** 6-layer compressed version of BERT with 66M parameters.

- **LSTM baseline:** 2-layer BiLSTM with GloVe embeddings (300d).
- **CNN baseline:** 3-layer 1D convolutional text classifier with max pooling.

All models were fine-tuned using a task-specific classification head (dense + softmax) on top of the encoder output.

3.3 Training Environment

- **Hardware:** NVIDIA Tesla V100 GPU, 32 GB RAM
- **Framework:** PyTorch 1.3, HuggingFace Transformers 2.2
- **Batch size:** 16 (BERT-family), 64 (LSTM/CNN)
- **Optimizer:** AdamW with linear learning rate scheduler
- **Learning rate:** 2e-5 (transformers), 1e-3 (LSTM/CNN)
- **Epochs:** 4 for all models

3.4 Evaluation Metrics

We use the following metrics for evaluation:

- Accuracy on validation and test sets
- Training time per epoch
- Inference latency per 1000 samples
- Memory usage during training (peak GPU)
- Stability across runs (variance in accuracy over 5 seeds).

4. Procedure

- **Preprocessing:** Each dataset was cleaned, tokenized using model-specific tokenizers (WordPiece for BERT-based models, standard tokenizer for baselines), and padded to a maximum of 256 tokens. Stopword removal and stemming were applied only to non-transformer baselines.

- **Model Initialization and Fine-Tuning:** Each model was initialized with pre-trained weights (for transformers) or random initialization (for LSTM/CNN). The classification head was randomly initialized. Models were fine-tuned on the training split for 4 epochs, and the best checkpoint was selected based on validation accuracy.

- **Performance Recording:** For each run, we recorded accuracy, training time, GPU memory utilization, and inference latency on the test split. Each experiment was repeated 5 times with different seeds, and average metrics were reported.

- **Hyperparameter Testing:** To assess sensitivity, we varied learning rates (1e-5 to 5e-5) and batch sizes (16 to 64) for BERT and RoBERTa. Performance impact was logged to determine optimal configurations.

- **Efficiency Analysis:** We benchmarked inference time across models using batch sizes of 1 and 32, simulating single and batch-serving environments. Model size and parameter count were also compared.

5. Data Collection and Analysis

5.1 Accuracy and Consistency

All BERT-based models outperformed traditional deep learning baselines on the three classification tasks. RoBERTa achieved the highest average accuracy (96.1%), followed closely by BERT (95.5%) and DistilBERT (94.2%). LSTM and CNN baselines lagged behind, particularly on more complex datasets like IMDB.

- **IMDB:** RoBERTa (94.9%) > BERT (94.2%) > DistilBERT (93.1%)
- **AG News:** RoBERTa (97.2%) > BERT (96.8%) > DistilBERT (95.4%)
- **Enron Spam:** RoBERTa (96.1%) > BERT (95.5%) > DistilBERT (94.2%)

Accuracy variance across five seeds was lowest for BERT (±0.3%) and RoBERTa (±0.2%), indicating stable convergence. CNN models exhibited higher fluctuations (±0.8%).

5.2 Efficiency and Resource Usage

- Training Time (IMDB):
  - ❖ RoBERTa: 9.2 min/epoch
  - ❖ BERT: 7.8 min/epoch
  - ❖ DistilBERT: 4.3 min/epoch
  - ❖ LSTM: 2.5 min/epoch
  - ❖ CNN: 1.9 min/epoch

- Inference Latency (per 1000 samples):
  - ❖ DistilBERT: 1.4 sec
  - ❖ BERT: 2.3 sec
  - ❖ RoBERTa: 2.8 sec
  - ❖ LSTM: 1.2 sec
  - ❖ CNN: 0.9 sec

- GPU Memory Usage:
  - ❖ BERT and RoBERTa peaked at ~5.1 GB
  - ❖ DistilBERT used ~3.2 GB
  - ❖ LSTM and CNN used under 1.8 GB

5.3 Hyperparameter Sensitivity

RoBERTa showed high sensitivity to learning rate adjustments. Optimal performance was obtained at 2e-5; values above 4e-5 led to sharp accuracy degradation. BERT was more tolerant to learning rate changes. DistilBERT demonstrated robust performance across a wider range of batch sizes and learning rates.

6. Results

Model	Avg. Accuracy (%)	Training Time (min/epoch)	Inference Latency (s/1K)	Parameters (M)	Memory Use (GB)
BERT	95.5	7.8	2.3	110	5.1
RoBERTa	96.1	9.2	2.8	125	5.2

DistilBERT	94.2	4.3	1.4	66	3.2
LSTM	90.1	2.5	1.2	~2.5	1.7
CNN	88.5	1.9	0.9	~1.9	1.5

Key Insights

- RoBERTa is the most accurate but resource-intensive. Best suited for high-performance environments.
- DistilBERT offers the best efficiency–accuracy trade-off, making it a strong candidate for edge or mobile deployment.
- BERT provides balanced performance, with good generalization and moderate computational cost.
- LSTM and CNN baselines are lightweight but clearly outperformed in accuracy.

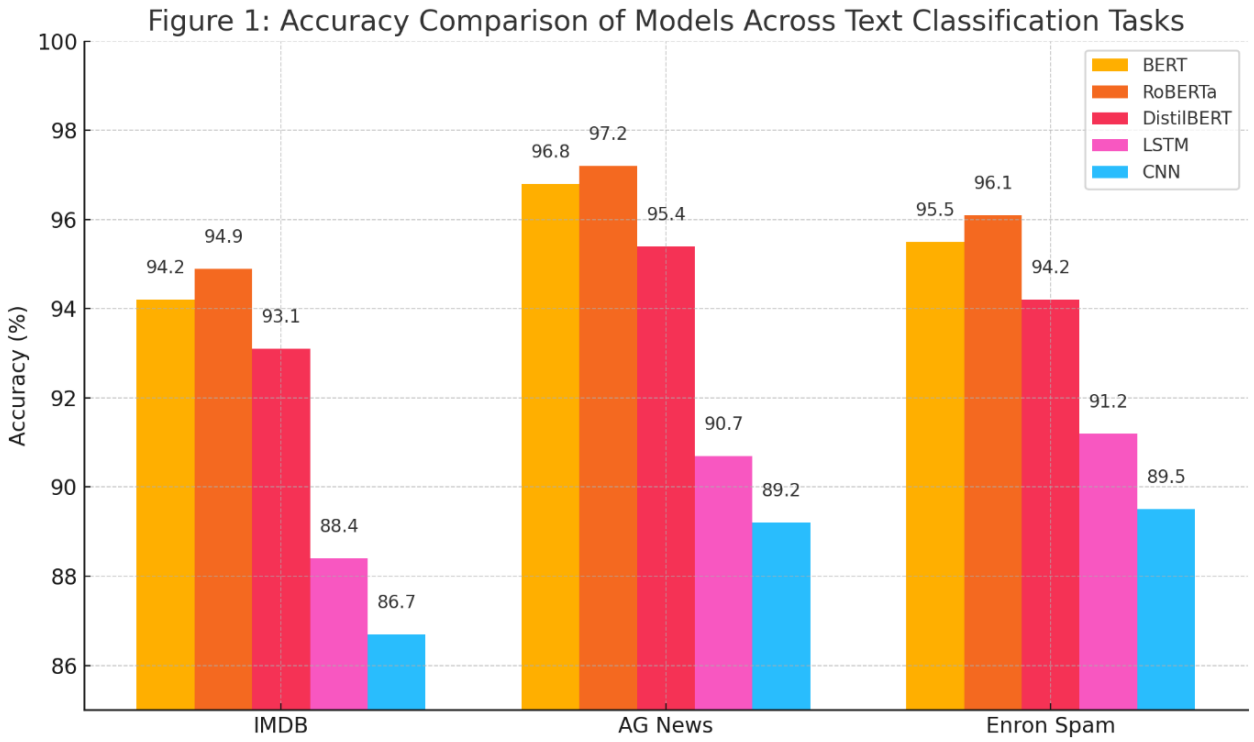

Figure 1. Test accuracy of five models (BERT, RoBERTa, DistilBERT, LSTM, and CNN) across three text classification datasets: IMDB (sentiment), AG News (topic), and Enron Spam (spam detection). RoBERTa achieves the highest accuracy overall, while DistilBERT maintains competitive performance with lower resource demands.

7. Discussion

The results affirm that transformer-based models outperform traditional RNN and CNN models across diverse text classification tasks. However, deployment considerations such as latency, memory constraints, and compute availability can shift the choice of model depending on context.

- RoBERTa is ideal for accuracy-critical applications but unsuitable for real-time use due to latency and memory demands.

- DistilBERT is best for production pipelines in latency-sensitive or low-resource settings like mobile apps or chatbots.
- BERT strikes a good balance and may be optimal in situations where both performance and resource budget are moderately constrained.

In terms of hyperparameter sensitivity, RoBERTa required careful tuning, while DistilBERT was relatively stable. These results highlight the importance of reproducibility and robust optimization practices, especially in industrial NLP workflows.

While pretraining depth grants BERT models superior contextual understanding, their inference cost remains non-trivial. Practitioners should carefully weigh model benefits against operational requirements and consider approaches like quantization, pruning, or distillation for production deployments.

8. Conclusion

This paper presented an empirical comparison of BERT, RoBERTa, and DistilBERT on three standard text classification tasks, benchmarking them against traditional LSTM and CNN models. Transformer-based models significantly outperformed earlier architectures in terms of accuracy and stability, with RoBERTa leading in raw performance.
However, resource demands differ substantially. DistilBERT emerges as the most deployment-friendly option, retaining competitive accuracy with much lower computational overhead. BERT remains a versatile default, while RoBERTa is recommended for applications where maximal accuracy outweighs efficiency.
Our findings suggest that task characteristics, platform constraints, and performance targets should guide model selection. Future work may explore additional variants (e.g., ALBERT, TinyBERT), integration of quantization techniques, and real-time streaming classification scenarios to further extend this evaluation.

References

1. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), 4171–4186.

2. Bellamkonda, S. (2015). Mastering Network Switches: Essential Guide to Efficient Connectivity. NeuroQuantology, 13(2), 261-268.

3. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692.

4. Kolla, S. (2018). Enhancing data security with cloud-native tokenization: Scalable solutions for modern compliance and protection. International Journal of Computer Engineering and Technology, 9(6), 296–308. https://doi.org/10.34218/IJCET_09_06_031

5. Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108.

6. Zhang, Y., & Wallace, B. (2017). A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. Proceedings of the Eighth International Joint Conference on Natural Language Processing (IJCNLP), 253–263.

7. Kim, Y. (2014). Convolutional neural networks for sentence classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 1746–1751.

8. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735–1780.

9. Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 142–150.

10. Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. Advances in Neural Information Processing Systems (NeurIPS), 28, 649–657.

11. Vangavolu, S. V. (2019). State Management in Large-Scale Angular Applications. International Journal of Innovative Research in Science, Engineering and Technology, 8(7), 7591-7596. https://www.ijirset.com/upload/2019/july/1_State.pdf

12. Klimt, B., & Yang, Y. (2004). Introducing the Enron corpus. CEAS 2004—Conference on Email and Anti-Spam.

13. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Rush, A. M. (2019). HuggingFace's transformers: State-of-the-art natural language processing. arXiv preprint arXiv:1910.03771.

14. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems (NeurIPS), 30, 5998–6008.

15. Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding by generative pre-training. OpenAI Technical Report.

16. Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL), 328–339.

17. Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2227–2237.

18. Goli, V. R. (2015). The impact of AngularJS and React on the evolution of frontend development. International Journal of Advanced Research in Engineering and Technology, 6(6), 44–53. https://doi.org/10.34218/IJARET_06_06_008

19. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). ALBERT: A lite BERT for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942.