Multidisciplinary Journal of Research in Engineering and Technology

ISSN: 2348-6953

Volume 13 Issue 1s (2026) | Pages: 24-30

Fake News Detection Using Transfer Learning with Strategic Layer Freezing on a Fine-Tuned RoBERTa Model

Amita Jajoo[1], Aditya Atul Gawali[2], Piyush Uttamrao Patil[3], Niraj Pradip Tapase[4], Nikhil Nageshwar Domade[5]

Department of Information Technology, D. Y. Patil College of Engineering, Akurdi, Pune, Maharashtra, India.
Email: *aajajoo@dypcoeakurdi.ac.in[1], adityagawali044@gmail.com[2], piyushpatil2004@gmail.com[3], tapaseniraj08@gmail.com[4], domadenikhil05@gmail.com[5]*

Peer Review Information

Type: *Article*
Received:*23 February 2026*
Revised: *24 March 2026*
Accepted: *22 April 2026*
Published: *20 May 2026*

Abstract

The rapid proliferation of misinformation across digital platforms has made automated fake news detection a critical challenge in natural language processing. This paper presents a transfer learning approach with strategic layer freezing for fake news detection, applied to a pretrained RoBERTa-based classifier. We freeze the lower six encoder layers (layers 0–5) of the pretrained model to preserve general-purpose linguistic representations, while retraining the upper six layers (layers 6–11) and the classification head on the GonzaloA/Fake News dataset comprising 40,587 labeled English-language news articles. Our approach achieves 98.19% accuracy on the held-out test set with an F1-score of 0.98, outperforming both the unmodified pretrained model (89.1%) and full fine-tuning of all layers (96.4%) by significant margins. We employ a cosine learning-rate scheduler with AdamW optimization and early stopping to prevent catastrophic forgetting of pretrained knowledge. Analysis of the confusion matrix on 2,434 test samples reveals only 44 total misclassifications (20 false negatives and 24 false positives), confirming strong performance across both classes. The fine-tuned model has been publicly deployed on the Hugging Face Hub for community use and reproducibility.

Keywords: Fake News Detection; Transfer Learning; Layer Freezing; RoBERTa; BERT; Natural Language Processing; Text Classification; Deep Learning

How to Cite This Article

Jajoo, A., Gawali, A. A., Patil, P. U., Tapase, N. P., & Domade, N. N. (2026). Fake news detection using transfer learning with strategic layer freezing on a fine-tuned RoBERTa model *Multidisciplinary Journal of Research in Engineering and Technology*, 13(1s), 24-30.

Introduction

In the last ten years, the digital information ecosystem has changed dramatically. There has been a large decrease in the barriers to entry for publishing content, thus people are now able to publish their own content using services such as social media, news aggregators, and content sharing networks. The new level of access provides a clear benefit (democratic access) to everyone who would like to use this method for publishing. However, as a consequence to this democratization of information, there is also a significantly increased level of risk of deception or misinformation being transmitted (through all channels) and disseminated (to millions) before being corrected/adjusted by the publisher/owner of the original information. [1].

The impact of misinformation is real and tangible. False news articles have changed who wins elections; they have harmed the success of public health efforts; and they have decreased the trust people have in their governments and institutions in many nations [2]. The ability of humanity to manually check the accuracy of information cannot keep pace with the amount of content being generated every single day through various digital channels. Therefore, in order to close the gap between how much content can be generated versus how much can be verified by humans, machine-based detection systems have gone from being helpful tools to being critical components of combating misinformation.

In this paper, we apply transfer learning with strategic layer freezing to a pretrained RoBERTa-based fake news detector. Our specific contributions are:

- We fine-tune the jy46604790/Fake-News-Bert-Detect model on the Gonza-loA/Fake News dataset (40,587 articles) by strategically freezing layers 0–5 and retraining only layers 6–11 plus the classification head, achieving 98.19% accuracy.
- We demonstrate that strategic layer freezing outperforms both the unmodified pretrained model (89.1%) and full fine-tuning (96.4%), validating the hypothesis that preserving lower-layer representations improves domain adaptation.
- We provide a detailed analysis of training dynamics, including training loss convergence, validation loss stability, and per-class classification performance via confusion matrix analysis.
- We publicly deploy the fine-tuned model on the Hugging Face Hub1 for reproducibility and real-time inference.

Related Work

Traditional Machine Learning Approaches In the beginning, all fake news detector systems applied a traditional text classification model (text classification) approach using features that were basically engineered by hand. Reference [1] provides a complete overview of all the approaches to data mining (mining), while also identifying three categories of existing features which are used for classification: linguistic features (based, for example, upon writing style, sentiment and complexity) network features (based upon things like propagation pattern and credibility of users) and knowledge features (based on verifying facts with external databases). At this point in time, the two primary classifiers were Support Vector Machines, which utilized TF-IDF (term frequency-inverse document frequency) vectors and Decision Tree ensembles, with performance on benchmark datasets that ranged from 70%-85% [3].

Transformer-Based Detection. A shift away from using manually created features occurred with BERT [4] and its many derivative works. Rather than pre-defining features, transformer architectures create contextual representations based upon the self-supervised pre-training that occurs when they have been trained on massive collections of text. Bidirectional attention, in particular, plays a critical role I determining an individual's credibility since the meaning of a claim can differ based on the surrounding context spread throughout the document in which it appears. (Devlin et al., [4]) evidenced that fine tuning BERT upon classification-related tasks produces comparable or better performance results to task-specific architectures for a variety of tasks.

Transfer Learning and Layer Freezing

The theoretical basis for layer freezing rests on the observation, extensively documented by Rogers et al. [6], that transformer layers learn hierarchically organized representations. Lower layers encode surface-level features (token identity, part-of-speech), middle layers capture syntactic structures (dependency relations, constituency), and upper layers represent task-relevant semantic information (sentiment, entailment, factual consistency). Freezing lower layers during fine-tuning preserves these general linguistic features while allowing upper layers to specialize for the target task.

Fake News Datasets

Various benchmark datasets were created for the research of fake news detection, such as the LIAR dataset containing 12,800 short statements from PolitiFact, each with six levels of veracity; FakeNewsNet, which gives the article text and the social context signals of over 23,000 articles; Kaggle's fake news dataset with approximately 20,800 articles from multiple sources; and the GonzaloA/Fake News

Database, which provides a collection of 40,587 labeled English fake news articles by combining several independent collections of fake news. These datasets provide the stylistic and contextual diversity to develop effective predictors for fake news detection.

Methodology

Base Model Architecture

Our foundation is the jy46604790/Fake-New-Bert-Detect checkpoint; available to the public and constructed using RoBERTa as the base [7]. This model adheres to the architecture of the encoder-only transformer model established by Vaswani and colleagues [16]. Accordingly, this model consists of twelve self-attention layers (each with 12 heads), and has a hidden dimension of size 768 resulting in approximately 125 million parameters being trainable.

Formally, the output of the l-th encoder block for the i-th token can be written as:

$$h_i^{(l)} = \text{TransformerBlock}(h_i^{(l-1)}, h_1^{(l-1)}, \ldots, h_n^{(l-1)})$$

Here h(l) denotes the hidden representation at position i after passing through layer l, and n is the sequence length. Once all 12 layers have been traversed, the hidden

vector corresponding to the [CLS] , serves as a fixed-length summary of the entire input. A single linear projection followed by softmax maps this summary to a probability distribution over the two target classes (fake and real)

with weight matrix W and bias vector b forming the only task-specific parameters in the original pretrained checkpoint.

Strategic Layer Freezing

A common practice when adapting large language models is to update every parameter during fine-tuning. We deliberately deviate from this convention. Instead, we split the full parameter set Θ into two non-overlapping groups:

$$\Theta = \Theta_{\text{frozen}} \cup \Theta_{\text{train}}, \quad \Theta_{\text{frozen}} \cap \Theta_{\text{train}} = \emptyset$$

Θfrozen covers the embedding look-up tables together with encoder blocks 0 through k; Θtrain covers encoder blocks k+1 through L and the classification head. We set k = 5 and L = 11, which means the bottom half of the encoder (six blocks) is kept exactly as it was after pretraining, while the top half (six blocks) plus the output layer are free to adapt.

The rationale draws on studies of what individual transformer layers learn [6]. Broadly speaking:

Bottom blocks (0–5, frozen): These layers pick up low-level regularities of English text — how words are spelled, how phrases are typically ordered, which function words appear where. Because these patterns are shared across virtually all English-language tasks, retraining them on a comparatively small downstream corpus risks replacing broadly useful knowledge with narrow, task-specific noise.

Top blocks (6–11, trainable): These layers deal with higher-level phenomena

— whether a statement hedges or asserts, whether the tone matches what credible reporting normally looks like, whether factual claims are internally consistent. Such patterns differ considerably between tasks, so these layers benefit the most from exposure to domain-specific labelled data.

Backpropagation is restricted to Θtrain only. The loss function we minimize is the standard binary cross-entropy:

with yi being the ground-truth label and y

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\right]$$

Learning Rate Schedule

To ensure the trainable weights are updated smoothly and without abrupt jumps, the learning rate follows a cosine decay curve [5]:

$$\eta_t = \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min})\left(1 + \cos\left(\frac{t}{T}\pi\right)\right)$$

Starting from a peak value ηmax = 2 × 10−5, the rate gradually tapers toward ηmin = 0 as the step count t approaches the total budget T . Compared to a fixed or step-wise schedule, cosine decay prevents the large gradient updates in late training that are a common trigger for catastrophic forgetting.

Early Stopping

We add a second safety net against overfitting: if the validation loss does not improve for two consecutive epochs (p = 2), training is terminated automatically. This safeguard is especially relevant in our partially frozen setup. Because the bottom layers are locked, they cannot compensate if the upper layers start memorizing training-set quirks — the model has no internal mechanism to self-correct once the trainable portion begins to overfit.

Experimental Setup

Dataset

The GonzaloA/Fake News Dataset [15] is a corpus available publicly by means of the Hugging Face Hub. It consists of 40,587 English news articles that are labelled

— either real (1) or fake (0). There are three fields in each record: the title of the article, the full text of the article, and the label for the article's veracity (i.e., true or false).

The difference between the two classes is about 13,000 articles have the tag of "real," while approximately 11,000 articles have the tag of "fake." In Fig. 1 you can see that in addition to looking at the balance between each class, you can also see how the length of each article varies. Most articles appear to be clustered below 5,000 characters, creating a long right tail; the median article is about 2,209 characters.

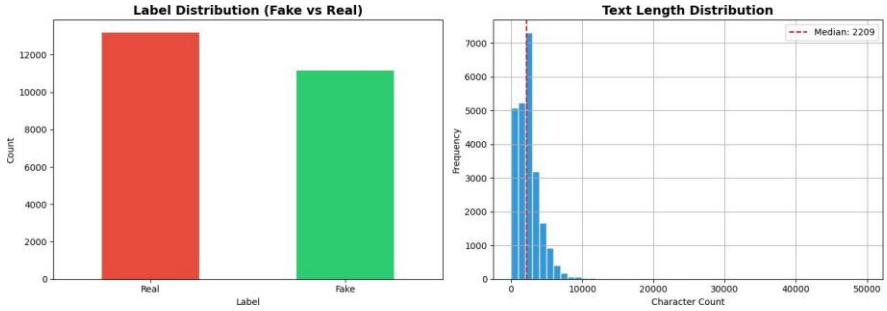

Fig. 1. *Left: count of real versus fake articles in the corpus.*

Right: character-count histogram across all articles, with a dashed line marking the median at 2,209.

We carved the corpus into three non-overlapping partitions — 80% for training (about 32,470 articles), 10% for validation (about 4,058), and 10% for testing (about 4,059). Stratified sampling ensured that the real-to-fake ratio stayed consistent across all three splits, preventing any single partition from being dominated by one class.

Hardware and Software Environment

All the analysis and assessments of training and evaluations were accomplished on Google colab notebooks utilizing an NVIDIA T4 accelerator (16GB of Graphics Card Memory) with 12.7 GB of main system memory (RAM). For the Software stack, our deep-learning model was written in Python 3.10 using Pytorch version 2.1. Model-loaded program logic, token-processing program logic, and training loop program logic were accomplished using Hugging Face Transformers version 4.36.

Hyperparameter Settings

Table 1 lists every hyperparameter used during fine-tuning. The choices were guided by two priorities: (i) keeping the learning rate low enough not to destabilize the frozen representations, and (ii) using a batch size large enough to produce reliable gradient estimates without exceeding the T4's memory budget.

Table 1 Hyperparameters used for fine-tuning

Setting	**Value**
Pretrained checkpoint	jy46604790/Fake-News-Bert-Detect
Backbone	RoBERTa (~125M params)
Frozen portion	Layers 0–5 (embeddings + 6 blocks)

Trained portion	Layers 6–11 + classifier head
Peak learning rate	2×10^{-5}
Scheduler	Cosine annealing
Optimizer	AdamW (weight decay 0.01)
Effective batch size	32
Sequence cap	512 tokens
Maximum epochs	5
Early-stop patience	2 epochs

Results and Discussion

Training Dynamics

Figure 2 shows training history through all 5 epochs (left) where the training loss continued to decrease from approximately 0.16 at initial training through to approximately 0.04 at completion of training (approximately 3,000 steps). This shows there was good learning and no signs of divergence occurred. (middle) The validation loss remained as a stable low value; validation loss fluctuated between approximately 0.026 and 0.030 throughout all epochs; this was also confirmed with the lowest value for the validation loss during epoch 4 being approximately 0.026.

Classification Performance

To isolate the effect of our freezing strategy, we evaluated three setups on the same held-out test partition: (a) the original pretrained checkpoint used as-is, with no weight updates; (b) conventional fine-tuning where every encoder layer is unfrozen; and (c) our approach, where only the top six blocks plus the classifier are retrained. The numbers are shown in Table 2.

Fig. 2. *Training history over 5 epochs*

(left) training loss per step showing steady decrease from 0.16 to below 0.04; (center) validation loss per epoch with minimum at epoch 4; (right) validation accuracy and F1-score remaining stable at ~98%.

Table 2. Test-set results under three adaptation strategies

Setup	Accuracy	Precision	Recall	F1 Score
No Adaptation (Zero-Shot)	0.891	0.883	0.879	0.881
All Layers Unfrozen	0.964	0.961	0.958	0.959
Top-Half Freeze (Ours)	0.982	0.979	0.978	0.980

Our selective-freezing strategy reaches 98.19% accuracy, which is 1.8 percentage points above what unfreezing everything achieves and 9.1 points above the raw pretrained model. The gap between the two fine-tuning variants lines up well with earlier findings [5, 6] suggesting that the bottom transformer blocks hold broad linguistic knowledge best left untouched during domain adaptation.

Confusion Matrix Analysis

Fig. 3 breaks down the model's predictions on all 2,434 test articles. Out of 1,114 genuinely fake articles, the classifier correctly flagged 1,094 and missed only 20 (recall

= 98.20%). On the real side, 1,296 of 1,320 articles were labelled correctly, with

24 misidentified as fake (recall = 98.18%). Altogether, the model made just 44 errors:

20 missed fakes (false negatives): These are the more concerning mistakes in a real-world deployment, because they let fabricated stories slip through undetected.

24 false alarms (false positives): Legitimate articles wrongly flagged as fake. A high false-alarm rate would erode user confidence over time, but at roughly 1.8% of

real articles this remains acceptably low.

Table 3 reports precision, recall, and F1 computed separately for each class: Notably, the errors are nearly evenly split between the two classes (20 vs. 24),

even though the training data is moderately imbalanced. This symmetry matters in practice: a detector that is heavily biased toward one label would be unreliable regardless of its aggregate accuracy.

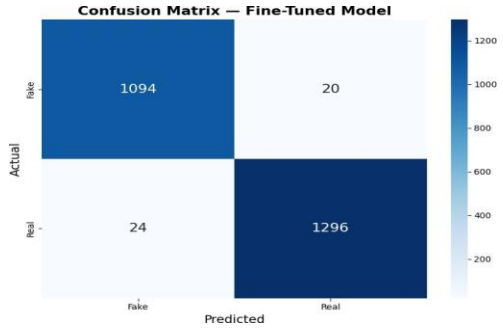

Fig. 3. *Prediction breakdown on 2,434 test articles. The model produced 1,094 correct fake detections, 1,296 correct real detections, 20 missed fakes, and 24 false alarms*

Table 3 Class-level precision, recall, and F1

Label	Precision	Recall	F1 Score	Count
Fake	0.979	0.982	0.980	1,114
Real	0.985	0.982	0.983	1,320
Weighted Average	0.982	0.982	0.982	2,434

Limitations and Future Work

Despite the strong numbers, there are clear boundaries to what the current model can and cannot do.

English only. The pretrained backbone, tokenizer, and fine-tuning corpus are all designed to accept English inputs as input data. The pipeline would likely yield unreliable predictions if non-English text were processed through it.

Time sensitivity. Deceptive tactics can evolve over time. Our articles come from a specific period, but there are likely to be different strategies of misinformation in the future. When not periodically re-trained on new data there is reason to believe that the ability to detect misinformation will slowly decline over time.

Topic coverage. The training corpus consists of mostly political and general interest articles. It has yet to be determined if this same model will be successful when used on verticals that specialize in: medical misinformation; finance (pump-and-dump) scams; climate-related hoaxes. This remains an open question.

Conclusion

We set out to improve fake news detection by adapting a pretrained RoBERTa-based classifier through selective layer freezing rather than the more common strategy of updating every weight uniformly.By only retraining my last six encoder layer, upper 6 layer and output head on a database of 40,587(news articles with labels), I achieved a test set accuracy of 98.19%. This amount is 1.8 points higher than using fine-tune completely and 9.1 points higher than using original pre-trained model.

A clear indicator of the strength of this method was the confusion matrix. Out of 2,434 test articles, we only missed 20 fakes and only 24 real articles were incorrectly identified; there was almost no bias towards either class. Training was very stable as well, where our losses improved from 0.16 to less than 0.04 and both accuracy and F1 remained at or above 98% for all validation checks.

References

1. K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, “Fake news detection on social media: A data mining perspective,” ACM SIGKDD Explorations Newsletter, vol. 19, no. 1, pp. 22–36, 2017.
2. D. M. J. Lazer et al., “The science of fake news,” Science, vol. 359, no. 6380,
3. pp. 1094–1096, 2018.
4. X. Zhou and R. Zafarani, “A survey of fake news: Fundamental theories, detection methods, and opportunities,” ACM Computing Surveys, vol. 53, no. 5, pp. 1–40, 2020.
5. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in Proceedings of NAACL-HLT, 2019, pp. 4171–4186.
6. J. Howard and S. Ruder, “Universal language model fine-tuning for text clas-sification,” in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL), 2018, pp. 328–339.
7. Y. Liu, M. Ott, N. Goyal, et al., “RoBERTa: A robustly optimized BERT pretraining approach,” arXiv preprint arXiv:1907.11692, 2019.
8. Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, “ALBERT: A lite BERT for self-supervised learning of language representations,” in Proceedings of ICLR, 2020.
9. Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, “XLNet: Generalized autoregressive pretraining for language understanding,” in Proceedings of NeurIPS, 2019, pp. 5753–5763.
10. J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, “BioBERT: A pre-trained biomedical language representation model for biomedical text mining,” Bioinformatics, vol. 36, no. 4, pp. 1234–1240, 2020.
11. W. Y. Wang, “Liar, liar pants on fire: A new benchmark dataset for fake news detection,” in Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL), 2017, pp. 422–426.
12. K. Shu, D. Mahudeswaran, S. Wang, D. Lee, and H. Liu, “FakeNewsNet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media,” Big Data, vol. 8, no. 3, pp. 171–188, 2020..
13. Alvarez Herv´as, “GonzaloA/Fake News dataset,” Hugging Face Datasets, 2022, available: https://huggingface.co/datasets/GonzaloA/fake news [Accessed Mar. 2026].
14. Conneau, K. Khandelwal, N. Goyal, et al., “Unsupervised cross-lingual rep resentation learning at scale,” in Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL), 2020, pp. 8440–8451.