International Journal of Computer (IJC)
ISSN 2307-4523 (Print & Online)
https://ijcjournal.org/index.php/InternationalJournalOfComputer/index

Performance Benchmarking of Traditional Machine Learning and Transformer Models for Multi-Class Text Classification

Omar El Khatib[a]*, Nabeel Alkhatib[b]

[a, b]*Math. and Computer Science Dept., Loyola University New Orleans, New Orleans, 70118 USA*
[a]*Email: oelkhat@loyno.edu*
[b]*Email: nalkhati@my.loyno.edu*

Abstract

Text classification is a fundamental task in natural language processing (NLP), widely applied in areas such as spam detection, sentiment analysis, and text categorization. This study presents a comparative analysis of three distinct machine learning paradigms—traditional machine learning algorithms (like Random Forest, XGBoost, support vector machine and Naive Bayes), a custom-built transformer architecture, and transfer learning or pre-trained transformer models (BERT, DistilBERT, RoBERTa, ELECTRA)—on the multi-class news classification dataset. While traditional models provided competitive baselines with up to 90.47% accuracy, modern transformer architecture surpassed them, achieving 91% accuracy when trained from scratch. The highest performance was observed with transfer learning using pre-trained models, where RoBERTa achieved 94.54% accuracy, DistillBERT achieved 94.32% accuracy, BERT achieved 94.07% accuracy and ELECTRA achieved 93.66%. These findings highlight the significance of contextual embeddings and large-scale pretraining in advancing text classification performance.

Keywords: NLP Multi-Class Classification; Transformer; NLP Transfer Learning; Text Classification.

1. Introduction

Text classification is a fundamental task in natural language processing (NLP) with broad applications ranging from sentiment analysis and spam detection to information retrieval and text categorization. Accurate news classification, in particular, plays a critical role in organizing digital content, enabling personalized news feeds, and filtering misinformation.

Received: *5/15/2025*
Accepted: *7/1/2025*
Published: *7/14/2025*

---

* *Corresponding author.*

Traditional machine learning (ML) models such as Random Forest, XGBoost, Support Vector Machine and Naïve Bayes, when combined with handcrafted features like bag-of-words or TF-IDF, have proven computationally efficient and interpretable. However, these models often struggle to capture long-range dependencies and semantic nuances in textual data.

Recent advances in deep learning, particularly transformer-based architecture, have significantly improved the ability of models to understand contextual meaning through mechanisms such as self-attention. Furthermore, transfer learning via pre-trained language models like BERT, DistilBERT, RoBERTa, ELECTRA … etc have enabled state-of-the-art performance on a wide range of NLP tasks with minimal labeled data.

This study conducts an empirical investigation to compare the effectiveness of traditional machine learning classifiers and transformer-based models—both trained from scratch and fine-tuned from pre-trained checkpoints—on the AG News dataset [1]. The objective is to evaluate the relative strengths and weaknesses of these approaches in terms of accuracy, training time, and resource efficiency. A unified experimental framework for comparing traditional machine learning, scratch-trained transformers, and transfer learned transformers on a common benchmark.

The remainder of this paper is organized as follows: Section 2 presents a literature review of traditional and transformer-based models. Section 3 describes methodology and experimental setup. Section 4 provides evaluation results and visualizations. Section 5 offers a discussion of the findings, and Section 6 concludes the paper with future research directions.

2. Literature Review

Text classification has evolved significantly over the past two decades, transitioning from traditional statistical models to neural networks [2] and, more recently, transformer-based architectures [3]. This section provides an overview of the relevant literature, organized into four parts: traditional machine learning approaches, word embeddings and early neural networks, transformer models, and the application of transfer learning through models such as BERT and DistilBERT.

2.1. Traditional Machine Learning for Text Classification

Traditional machine learning methods have long served as baseline techniques for text classification due to their computational efficiency, interpretability, and simplicity. These models typically rely on sparse feature representations such as bag-of-words (BoW), term frequency (TF), or term frequency–inverse document frequency (TF-IDF) vectors.

In [4], the effectiveness of Support Vector Machine for text classification was established. [5] demonstrated the surprising effectiveness of Naive Bayes classifiers in text classification, particularly when trained on sparse features. Ensemble-based models such as Random Forests [6] and XGBoost [7] later emerged, improving classification performance through ensemble learning and regularization techniques.

In this study, we investigate four widely adopted traditional classifiers:

- **Random Forest (RF)** is an ensemble learning method that builds multiple decision trees using different subsets of data and features and combines their outputs through majority voting [6]. Random Forests are robust to overfitting and offer interpretability through feature importance scores. However, they lack the ability to capture word order or contextual relationships, making them less effective on tasks requiring deep semantic understanding.
- **Extreme Gradient Boosting (XGBoost)** is a tree-based ensemble algorithm that improves upon traditional boosting methods by incorporating regularization, second-order optimization, and parallelization [7]. XGBoost is known for its efficiency and strong performance on structured data, and it has been successfully applied to text classification when features are carefully engineered. While powerful, XGBoost still relies on fixed, sparse representations of text (e.g., TF-IDF), and does not inherently capture the sequential nature of language, which limits its performance in comparison to models that use contextual embeddings.
- **Naive Bayes (NB)** is a probabilistic classifier based on Bayes' theorem, with the simplifying assumption that features are conditionally independent given the class label. The Multinomial Naive Bayes variant is particularly suited for document classification tasks, where word frequency plays a critical role [8]. Naive Bayes is computationally efficient and performs well on high-dimensional sparse text data. Despite its strong performance on some datasets, its independence assumption limits its ability to model more complex linguistic patterns.
- **Support Vector Machines (SVM)** are supervised learning models that aim to find the optimal hyperplane that separates data points of different classes with the maximum margin [4]. Linear SVMs are particularly effective for text data due to the typically linearly separable nature of TF-IDF vectors in high-dimensional space. They offer strong generalization capabilities and are less prone to overfitting compared to decision trees. Kernel SVMs, while theoretically more powerful, are rarely used in practice for large text datasets due to their computational cost.

Despite their strengths, these models struggle with capturing long-range dependencies, syntactic structure, and semantic context. Table 1 gives a comparison summary of the four traditional machine learning algorithms: Random Forest, XGBoost, Naïve Bayes and Support Vector Machine.

Table 1: Comparison of Traditional Machine Learning Models

Model	Handles Feature Interaction	Capture Word Order	Interpretability	Scalability
Naïve Bayes	No	No	High	High
Random Forest	Yes (non-linear)	No	Medium	Medium
XGBoost	Yes (boosted)	No	Low	High
Support Vector Machine	Yes (linear and non-linear)	No	Medium	Medium

2.2 Word Embedding and Neural Networks

The limitations of sparse vector representations in capturing semantic meaning led to the development of distributed word embeddings such as Word2Vec [9] and GloVe [10]. These static embeddings represent words in dense vector spaces, enabling models to understand semantic similarity.

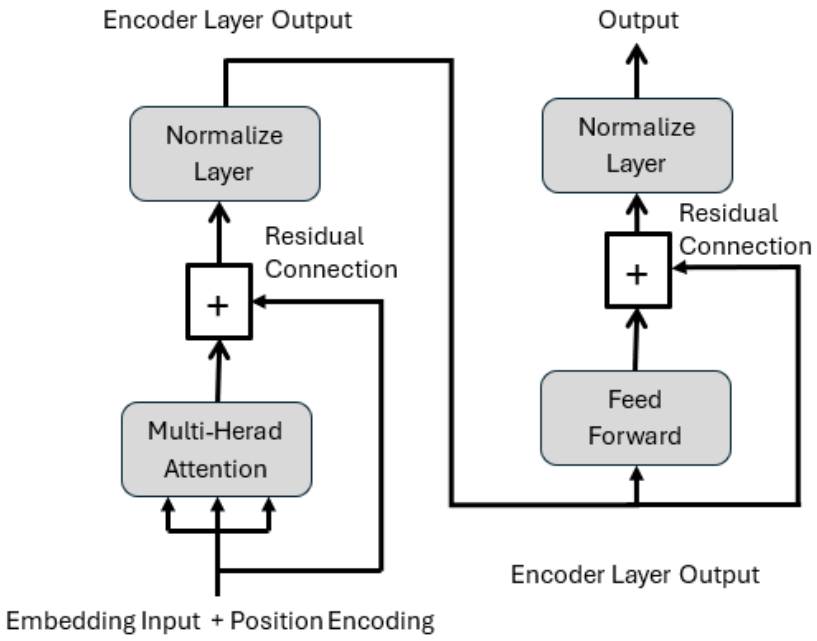
When combined with neural architectures such as feed-forward networks, convolutional neural networks (CNNs) [11], and recurrent neural networks (RNNs), classification performance improved. However, static embeddings are context-independent, and neural models like RNNs and LSTMs [12] still struggled with long-range dependencies and training inefficiencies.

2.3 Transformer Models

The transformer architecture, introduced by Vaswani and his colleagues (2017) [3], revolutionized NLP by modeling long-range dependencies via multi-head self-attention, eliminating the need for recurrence. Transformer models achieve parallelization and scalability, making them suitable for large-scale pretraining on corpora such as Wikipedia or BookCorpus. It consists of two primary components: the encoder and the decoder. The encoder processes the input sequence and generates a contextualized representation using multi-head self-attention and feedforward layers. The decoder generates output tokens one at a time, attending to both the previously generated tokens (via masked self-attention) and the encoder's output (via cross-attention).

This encoder-decoder framework is highly effective in sequence-to-sequence tasks, such as machine translation, summarization, and dialogue generation. However, for text classification, only the transformer encoder is typically used, as the task requires understanding and representing the input rather than generating new sequences.

Each encoder block in a transformer contains of embedding input to encode the input tokens, positional encodings that is added to input embeddings to retain word order, multi-head self-attention to build attention weights over all positions in a sequence, feedforward Networks to processes the classify the output of the attention mechanism, residual Connections and Layer Normalization that aid in convergence and training stability.



(a) Encoder Layer (b) Feed forward predictive

Figure 1: Encoder layer and feed forward classifier

The purpose of the transformer encoder layer is to encode the meaning of the whole input sentence into rich vector and the feed forward is to classify the input tokens.

The decoder generates the output sequence (e.g., a translated sentence in French) one token at a time, attending to both previous output token and encoder outputs. The Transformer decoder components are similar to the transformer encoder components except for the addition of an extra attention mechanism that allows the decoder to attend to the encoder's output.

2.4 Transformer and Transfer Learning for Text Classification

Transformer-based models have significantly advanced the field of text classification by enabling the modeling of long-range dependencies, syntactic structure, and contextual relationships through self-attention mechanisms. In this task, the input text is tokenized and converted into dense embedding vectors, then passed through a transformer encoder to generate a fixed-length contextual representation. This contextual vector is then fed into a classification head composed of a feedforward layer followed by a SoftMax function to compute class probabilities.

The most impactful progress in transformer-based NLP has come through **transfer learning**, where models are pre-trained on large-scale unlabeled corpora using unsupervised objectives and then fine-tuned on specific downstream tasks such as text classification. This paradigm allows the transfer of linguistic knowledge to task-specific data with limited labeled examples, enhancing model performance and generalizability.

Several pre-trained transformer variants have been developed, each introducing architectural or training modifications for improved efficiency or performance:

- **BERT (Bidirectional Encoder Representations from Transformers)** [13]: BERT introduced a masked language modeling (MLM) task and next sentence prediction (NSP) objective during pretraining. It uses only the encoder component of the transformer, with [CLS] representing the entire sequence. BERT-base consists of 12 layers, 768 hidden units, and 12 attention heads. BERT-large doubles these dimensions with 24 layers and 16 heads (see Figure 2).

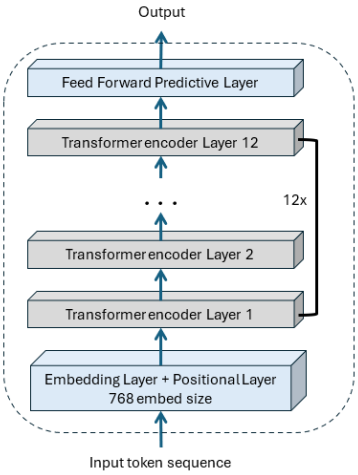


Figure 2: BERT Encoder Architecture

- **DistilBERT** [14]: A distilled version of BERT, this model retains 97% of BERT's language understanding while being 40% smaller and 60% faster. It uses 6 layers and omits the NSP objective to streamline training.
- **ELECTRA** [15]: Instead of MLM, ELECTRA introduces a replaced token detection objective, where a discriminator learns to detect whether input tokens are real or replaced. It maintains a BERT-like architecture but achieves faster and more data-efficient learning.
- **RoBERTa (Robustly Optimized BERT)** [16]: RoBERTa builds upon BERT by eliminating the NSP task, training on larger datasets with dynamic masking, and using longer training schedules. This leads to richer contextual embeddings and state-of-the-art performance on various NLP benchmarks.

During fine-tuning, these pre-trained models are adapted by adding classification heads and training them on the AG News dataset. This process significantly boosts performance compared to training from scratch or using traditional machine learning models.

Comparative benchmarking has been conducted in earlier works, such as [11], who demonstrated that CNNs with static word embeddings could outperform traditional models, and [13], who introduced BERT with remarkable results on many NLP tasks. However, these studies often evaluated models in isolation. In contrast, our work integrates traditional ML models, a custom-built transformer, and several transfer-learned transformers

within a unified framework. This enables a holistic performance evaluation across paradigms and provides practical insights for selecting suitable models based on task demands, computational constraints, and desired accuracy.

3. Methodology

This section details the experimental setup used to evaluate and compare the performance of traditional machine learning models, transformer models trained from scratch, and pre-trained transformer models on a standardized text classification task. All experiments were conducted using the **AG News** dataset from Kaggle.

3.1 Dataset Description and Preprocessing

The **AG News corpus** is a benchmark dataset for text classification that contains English news articles categorized into four classes: *World*, *Sports*, *Business*, and *Science/Technology*. Each record comprises a news title and a short description. For this study, the title and description fields were concatenated to form a single input text. The dataset size is 120,000 samples for training and 7,600 samples for testing.

To ensure consistency across models, a preprocess pipeline is performed that includes lower case text, punctuation removal, and tokenization.

Each classifier was trained and evaluated using stratified data split 80% training, 10% validation and 10% testing.

3.2 Traditional Machine Learning Models

For traditional machine learning models TF-IDF vectorization using unigrams and bigrams are performed. Model selection was conducted using grid search with five-fold stratified cross-validation, and accuracy was used as the primary evaluation metric. Random Forest is applied with 500 estimators, max depth of 60, max samples of 63%. XGBoost is applied with 200 estimators, number of classes of 4, max depth of 5, learning rate of 0.1 and evaluation metric of 'mlogloss'. Naïve Bayes is applied with Laplace smoothing of 0.1.

3.3 Transformer Model Trained from Scratch

For transformer-based model from scratch, a custom tokenizer was trained jointly with the model to generate input embeddings. The sequences are truncated or padded to a maximum of 128 tokens. A special [CLS] token at the beginning of each sequence is added. The model architecture implementation is as follows: number of layers is 4, attention heads is 12, hidden size is 624, feed forward dimension is 2496. The embeddings were augmented with positional encodings to incorporate sequence information.

3.4 Transfer Learning with Pretrained Models

We employed transfer learning by fine-tuning four pre-trained transformer models on the AG News dataset. The four pretrained transformer models are BERT (base-uncased) model, DistillBERT (base-uncased) model,

ELECTRA (electra-small-discriminator) and RoBERTa (roberta-base).

3.5 Experimental Environment

Both transformer models from scratch and transfer learning are trained using AdamW with a learning rate of 1e-4, weight decay of 0.1, Learning rate schedule 'ReduceLRONPlateau' were employed to prevent overfitting and optimize convergence, Batch Size: 32, Epochs: Up to 20, with early stopping, Loss Function: Cross-entropy loss.

All models were implemented in PyTorch deep learning framework, with APIs provided by Hugging-Face Transformers library. Training was conducted in a GPU-accelerated environment to expedite experimentation and enable scalability. Figure 4 shows the steps to the training process. First the title and description are concatenated to form a single text input. After that is the preprocess steps that include case lowering, multiple whitespace removal, lemmatization, and tokenization of the text. For traditional machine learning, preprocess step of TF-IDF is applied to the tokens, then applying the Random Forest, XGBoost, Naïve Bayes and Support Vector Machine. For Transform-based models, embedding and positional encoding is applied, then either applying transformer from scratch or pretrained transformer model, after that the [CLS] token is passed through a feed forward layer with SoftMax for classification.

4. Experimental Results

This section presents the comparative results of traditional machine learning classifiers, a transformer trained from scratch, and pretrained models on the AG News dataset.

4.1 Evaluation Metrics

All models were evaluated using **accuracy, training time** and number of parameters (weights) as the primary performance metric. Additional metrics such as macro F1-score and per class F1-score were calculated to provide a more comprehensive view.

4.2 Result

This study evaluated the performance of four traditional machine learning classifiers and five transformer-based models on the AG News classification task. The primary performance metrics are summarized in Table 2. The bar chart of test accuracy for each model is shown in figure 5. In addition, the bar chart of training time is shown in figure 6. Detailed confusion matrices for each model are presented to illustrate classification behavior across the four categories: World, Sports, Business, and Technology, shown in figure 7 and 8.

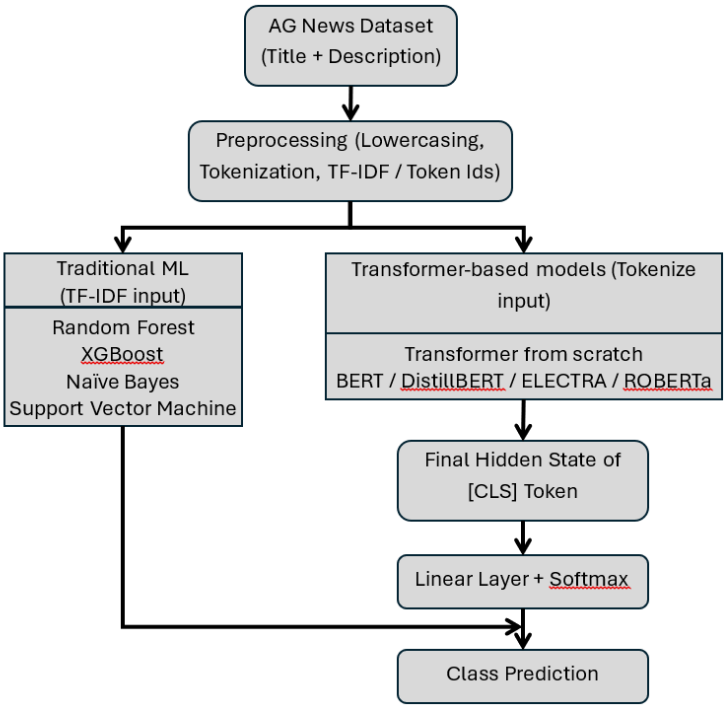


Figure 3: Overview of the architecture and training pipeline for traditional machine learning models, transformer models trained from scratch, and pretrained transformer-based models.

As shown in Table 1 and figure 5, the pre-trained transformer models achieved the highest test accuracy and F1-scores, with RoBERTa emerging as the top-performing model with a test accuracy of 94.54% and an F1-score of 0.95, closely followed by DistllBERT and BERT. Although ELECTR has a slightly lower accuracy (93.66), it demonstrated strong performance with the shortest training time among pretrained transformers. Among the traditional classifiers, the Support Vector Machine (SVC) and XGBoost achieved the best results, with 90.47% test accuracy and 90.33% test accuracy, respectively. However, as shown in Table 1 and figure 6 SVC required the longest training time by far (8746 seconds ≈ 2 hours and 25 minutes), limiting its scalability. Notably, the Naïve Bayes classifier offered an exceptionally fast training time of only 0.2 seconds, while still maintaining a

Table 2: Comparison of Model on the AG News dataset.

Model	Test Accuracy (%)	Training Time (seconds)	F1-Score	World F1	Sports F1	Business F1	Tech F1
Traditional Models							
Random Forest	85.06	87.66	85	86	91	82	81
XGBoost	90.33	79.6	90	91	95	87	88
Naïve Bayes	88.95	0.2	89	89	96	84	86
Support Vector Machine	90.47	8746	90	91	96	87	88
Transformer Models							
Transformer (scratch)	91.00	1015.95	91	91	96	88	89
BERT	94.07	3474.8	94	95	99	91	92
DistilBERT	93.32	1857.3	94	95	99	92	92
ELECTRA	93.66	1103	94	94	98	90	92
RoBERTa	95.54	4111.9	95	96	99	92	92

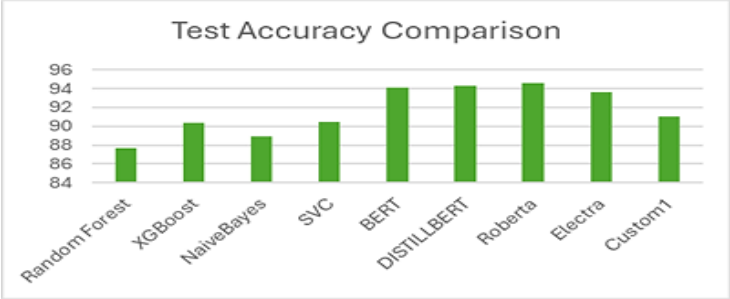


Figure 5: Test time accuracy comparison between different models

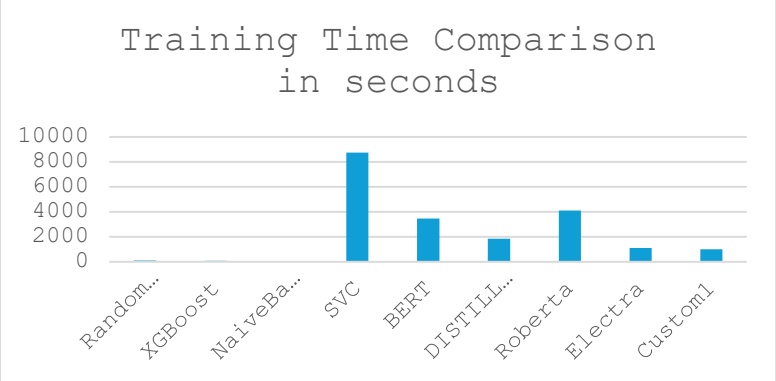


Figure 6: Training time comparison

competitive accuracy of 88.95%. Random Forest classifier is also offered a fast-training time of 28.1 seconds but achieve lower accuracy of 85.08.

Analysis of class-wise F1-scores revealed interesting patterns across different news categories:

- **Sports category**: All models performed exceptionally well, with transformer models achieving 98-99% F1-scores and traditional methods ranging from 93-96%
- **World category**: Transformer models achieved 94-96% F1-scores, while traditional methods ranged from 89-91%
- **Business and Technology categories**: These showed the most variation, with transformer models achieving 90-92% for both categories, while traditional methods ranged from 84-88%

4.3 Confusion Matrix (for BERT)

The confusion matrix for traditional machine learning is shown in figure 7 and the confusion matrix for transformer models is shown in figure 8. Most misclassifications occurred between closely related topics like Business and World.The confusion matrices reveal the specific error patterns of each model. For instance, the Random Forest classifier frequently misclassified 'Business' news as 'Technology' (253 instances). In contrast, the top-performing RoBERTa model showed significantly fewer misclassifications, with its largest error being the confusion between 'Business' and 'Technology' (101 instances), indicating a better ability to distinguish between semantically similar classes.

5. Discussion and Analysis

The results clearly demonstrate a hierarchy of performance between the model classes. The pre-trained transformer models, as a group, substantially outperformed the traditional machine learning models and the custom transformer built from scratch.

5.1. Transformer Superiority

The experimental results clearly demonstrate the superiority of transformer-based models over traditional machine learning approaches for text classification tasks. The improvement in accuracy and F1-scores in transformer models can be attributed to several key factors inherent in transformer architecture. First, the self-attention mechanism enables transformers to capture long-range dependencies and contextual relationships within text that traditional bag-of-words and TF-IDF approaches cannot effectively model. This is particularly evident in the Business and Technology categories, where semantic similarity and domain-specific terminology require nuanced understanding beyond simple keyword matching. Second, pre-trained transformer models leverage transfer learning from large-scale language modeling tasks, providing rich semantic representations that traditional methods cannot access without extensive feature engineering. The consistent high performance across all transformer variants (BERT, DistilBERT, RoBERTa, ELECTRA) validates the effectiveness of this pre-training paradigm.

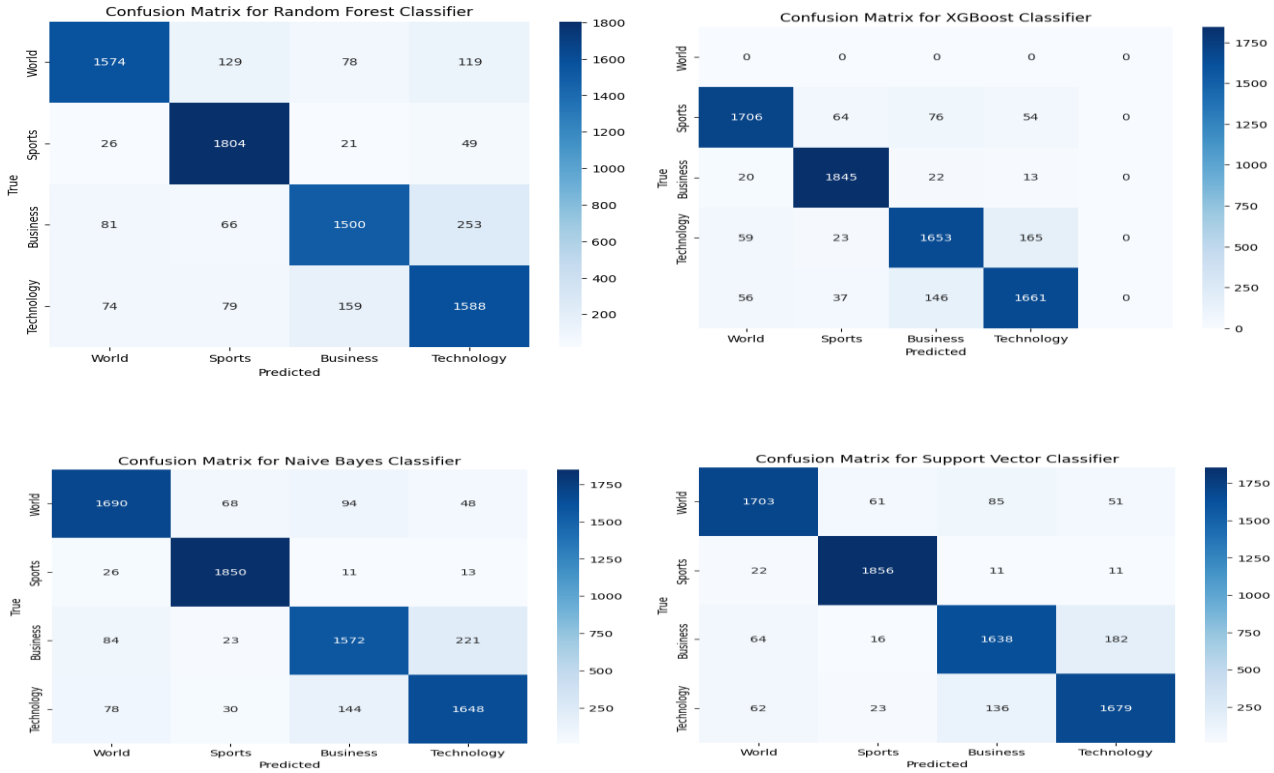


Figure 7: Confusion Matrix for each model in the traditional machine learning across all four AG News categories.

5.2. The Performance-Efficiency Trade-off

A critical finding is the stark trade-off between predictive performance and computational cost. The RoBERTa model, while the most accurate, also required one of the longest training times (4,111.9 seconds). The SVC model presented an anomaly, demanding the longest training time by a significant margin (7,956 seconds) without delivering performance comparable to the top transformers. On the other hand, Naïve Bayes stands out for its incredible efficiency, making it a viable baseline or a choice for applications where training speed is paramount. Among the high-performing transformers, DistilBERT and ELECTRA offer a compelling balance. DistilBERT, a distilled version of BERT, provides accuracy (94.32%) nearly matching RoBERTa while reducing the training time by over 50%. ELECTRA is even more efficient, achieving a strong accuracy of 93.66% in just 1,103 seconds, making it a highly practical choice for production environments.

5.3. Custom Transformer Performance

The custom transformer, while not reaching the performance levels of pre-trained models, achieved an accuracy of 91%, outperforming all traditional classifiers except for SVC and XGBoost, where it was marginally better. This demonstrates that the transformer architecture itself is powerful, but the transfer learning approach, which leverages knowledge from massive datasets, provides a decisive advantage.

5.4. Error Analysis

Analysis of the confusion matrices indicates that the most common misclassifications across nearly all models occurred between the 'Business' and 'Technology' categories, and to a lesser extent, between 'World' and 'Business'. This suggests a significant semantic overlap in the vocabulary used in articles from these categories, posing a challenge for all classifiers. However, the advanced transformer models managed this confusion more effectively than their traditional counterparts, as evidenced by the lower number of off-diagonal errors.

5.5. Comparison with Prior Work

Several prior studies have benchmarked traditional models and transformers on text classification tasks, but few have combined all three paradigms—traditional ML, scratch-built transformers, and transfer-learned transformers—within a unified pipeline. For example, [4] focused on SVMs with TF-IDF vectors, achieving strong results on binary classification tasks. Reference [13] demonstrated the power of BERT in fine-tuning for downstream NLP tasks, but comparisons with traditional models on multi-class classification were limited. Our results reaffirm and expand upon these findings by empirically quantifying the trade-offs between accuracy, resource demands, and training time. Additionally, while studies like [14,17] introduced model improvements such as distillation and robust optimization, our study is among the few that explicitly compares all such variants side-by-side under control conditions.

6. Limitations

While the results presented in this study are robust and comprehensive within the AG News dataset context, it

has some limitations. First, the study focuses solely on the AG News dataset, which may not generalize to domains like legal or biomedical text, where vocabulary and structure differ significantly. Second, the transformer models were fine-tuned using fixed hardware and training configurations. Different batch sizes, learning rate schedules, or hardware accelerators (e.g., TPU) might lead to different results. Third, only base

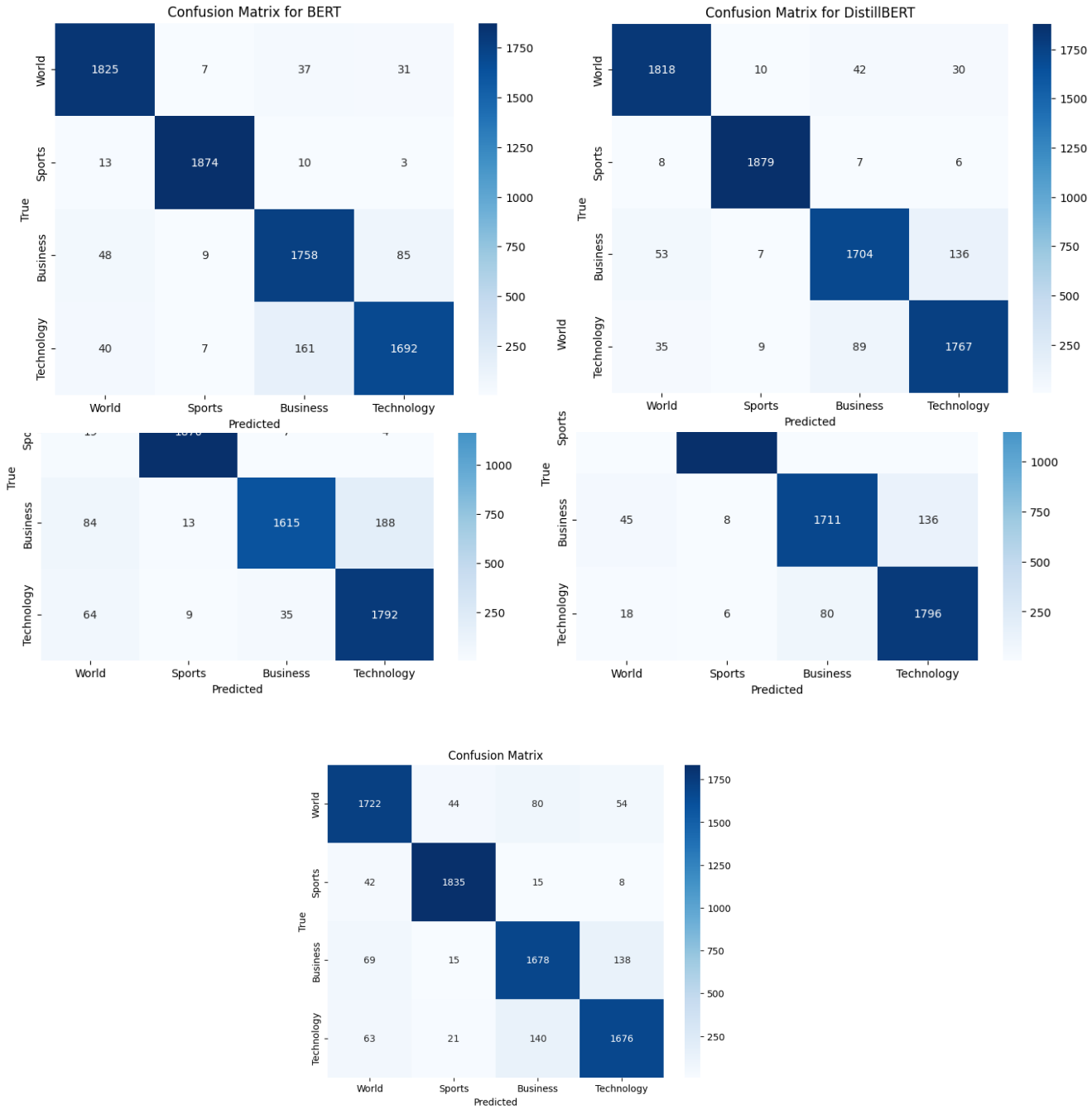


Figure 8: Confusion Matrix for transformer models across all four AG News categories.

versions of BERT, RoBERTa, etc., were evaluated. Larger or domain-specific variants (e.g., large BERT or BioBERT) might yield superior results. Traditional models also used a fixed feature set (TF-IDF); incorporating semantic embeddings (e.g., doc2vec) could affect outcomes. Fourth, while transformer models achieved higher

accuracy, they remain less interpretable compared to traditional classifiers like Naïve Bayes or decision trees. This could hinder deployment in sensitive or regulated environments. Finally, the reported training times for pre-trained transformers only account for fine-tuning, not pretraining. This might skew perceptions of overall cost-effectiveness if evaluated in isolation.

7. Conclusion

This comprehensive comparative study demonstrates the clear superiority of transformer-based models over traditional machine learning approaches for text classification on the AG News dataset. Transformer models achieved 4-7 percentage points higher accuracy than traditional methods, with RoBERTa leading at 94.54% accuracy and 95 F1-score. All transformer variants (BERT, DistilBERT, RoBERTa, ELECTRA) outperformed the best traditional method (SVM at 90.47% accuracy), indicating the robustness of the transformer architecture. While transformer models require significantly more training time (1,103-4,112 seconds vs 0.2-8,746 seconds for traditional methods), DistilBERT provides an optimal balance of performance and efficiency. Sports classification achieved the highest accuracy across all models, while Business and Technology categories presented the greatest challenges due to semantic overlap. The choice between traditional and transformer approaches should consider the specific requirements of accuracy, training time, and computational resources available.

The study validates the transformative impact of attention-based architectures in natural language processing and provides practical guidance for practitioners selecting appropriate models for text classification tasks.

8. Future Work

Based on the findings of this study, several avenues for future research can be pursued. First, Investigating the performance of these models on domain-specific news datasets (e.g., financial news, scientific articles, local news) would provide insights into model generalizability and the need for domain-specific fine-tuning strategies. In addition, developing ensemble methods that combine transformer models with traditional ML approaches could potentially achieve superior performance while maintaining computational efficiency. Investigating weighted voting schemes, stacking, and blending techniques represents a promising research avenue. Extending this comparative analysis to multilingual news datasets using models like mBERT, XLM-R, and language-specific transformers could provide insights into cross-lingual transfer learning effectiveness. Future research should address bias detection and mitigation in news classification systems, particularly examining how different models handle sensitive topics, cultural context, and potential discriminatory patterns in classification decisions. These future directions would significantly contribute to the advancement of text classification methodologies and their practical applications in news media analysis, content recommendation systems, and automated journalism tools.

References

[1] AG News Dataset source: https://www.kaggle.com/datasets/amanandrai/ag-news-classification-datase

[2] Juergen Schmidhuber. Deep Learning in Neural Networks: An Overview. Neural Networks, Vol 61, pp 85-117, Jan 2015. https://arxiv.org/abs/1404.7828

[3] Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). *Attention is All You Need*. In Advances in Neural Information Processing Systems (NeurIPS), 30.

[4] Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. In: Nédellec, C., Rouveirol, C. (eds) Machine Learning: ECML-98. ECML 1998. Lecture Notes in Computer Science, vol 1398. Springer, Berlin, Heidelberg. https://doi.org/10.1007/BFb0026683

[5] McCallum, A., & Nigam, K. A comparison of event models for naive bayes text classification. *AAAI Workshop*, 1998.

[6] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

[7] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD*, 785–794.

[8] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze. Introduction to Information Retrieval. Cambridge University Press, 2008. ISBN-13: 978-0521865715.

[9] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. 2013. https://arxiv.org/abs/1301.3781

[10] Jeffrey Pennington, Richard Socher, Christopher Manning. GloVe: Global Vectors for Word Representation. Proceedings Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014. https://aclanthology.org/D14-1162

[11] Kim, Y. (2014). Convolutional neural networks for sentence classification. *EMNLP*.

[12] Sepp Hochreiter, Jürgen Schmidhuber. Long Short-Term Memory. Neural Computation, Volume 9, Issue 8. Pages 1735 – 1780, 1998 https://doi.org/10.1162/neco.1997.9.8.1735

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

[14] 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.

[15] Kevin Clark, Minh-Thang Luong, Quoc V. Le, Christopher D. Manning. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. 2020. https://arxiv.org/abs/2003.10555

[16] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. 2019. https://arxiv.org/abs/1907.11692

[17] Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, Noah A. Smith. Linguistic Knowledge and Transferability of Contextual Representations, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1. 2019. https://arxiv.org/abs/1903.08855