A Comparative Robustness Study of Lightweight and Transformer-based Models for Short Text Classification

Ziqi Gu Shengyang Tao Yuyang Ding Alissa Wu

Abstract

Real-world text classification must handle typos, case variations, and other noise that benchmark evaluations typically ignore. We systematically compare the robustness of three architectural families—TF-IDF with Logistic Regression, BERT/DistilBERT encoders, and the decoder-only MiniMind—under identical synthetic perturbations (character swaps, drops, case flips, and emoji insertion) on the AG News dataset. Our key finding is that bidirectional attention provides substantial robustness advantages: BERT degrades only 1.79% under noise versus 10.1% for MiniMind trained on clean data. Noise-augmented fine-tuning closes this gap for MiniMind while simultaneously improving clean accuracy—from 70.7% to 76.2%—suggesting that noise exposure acts as effective regularization. For deployment, DistilBERT emerges as the strongest choice overall, retaining 99.3% of BERT's accuracy at 1.88× the inference speed with equivalent noise resilience, while TF-IDF remains viable for latency-critical applications requiring sub-millisecond response.

1 Introduction

Short text classification is a fundamental NLP task with applications in news categorization, sentiment analysis, and spam detection. While transformer-based models achieve strong benchmark performance, real-world text often contains typos, case variations, and other noise that can degrade accuracy. Prior evaluations largely neglect these noisy conditions, leaving open questions about how architectural choices affect robustness.

This paper compares three modeling approaches under both clean and noisy conditions: TF-IDF with Logistic Regression, BERT/DistilBERT encoders (Devlin et al., 2019; Sanh et al., 2019), and the MiniMind decoder-only model (Gong, 2024). We introduce synthetic perturbations (character swaps, drops, case flips, emoji insertion) to the AG News dataset (Zhang et al., 2015) and evaluate each model's degradation.

Our contributions are: (1) a controlled comparison of traditional, encoder, and decoder architectures under identical noise conditions; (2) evidence that noise-augmented training improves robustness across all model families; and (3) practical recommendations for model selection based on accuracy-robustness-speed tradeoffs.

2 Related Work

Feature-based text classification using TF-IDF representations remains a strong baseline due to its interpretability and efficiency (Das et al., 2021). However, these methods cannot capture contextual word relationships.

The Transformer architecture (Vaswani et al., 2017) enabled significant advances through self-attention mechanisms. BERT (Devlin et al., 2019) introduced bidirectional pretraining via masked language modeling, achieving state-of-the-art results on many NLP benchmarks. DistilBERT (Sanh et al., 2019) showed that knowledge distillation can compress BERT to 40% fewer parameters while retaining 97% of GLUE performance.

Decoder-only models like GPT use causal attention, restricting each token to attending only to previous positions. While designed for generation, these architectures can be adapted for classification. MiniMind (Gong, 2024) provides a lightweight decoder implementation suitable for studying smaller-scale models.

Prior work has shown neural models are sensitive to character-level perturbations, with synthetic noise augmentation proposed as mitigation. However, systematic comparisons across encoder, decoder, and traditional architectures remain limited. The question of whether robustness stems from architectural properties (bidirectional vs. causal attention) or training strategy (noise exposure) is a gap this work directly addresses.

3 Methodology

3.1 Task

We address the task of short text topic classification under both clean and noisy input conditions. Given a news headline, the model must predict one of four topic categories: World, Sports, Business, or Science/Technology.

3.2 Dataset

We use the AG News topic classification dataset (Zhang et al., 2015), which contains four balanced classes (World, Sports, Business, and Science/Technology). The official release provides 120,000 training examples (30,000 per class) and 7,600 test examples (1,900 per class). Each example includes a title and description.

Following our experimental protocol, we create a validation set by holding out 7,600 examples from the original training split, leaving 112,400 examples for training. We use the standard 7,600-example test split for final evaluation. We generate both clean and noisy versions of each split (train/validation/test) using the perturbation process described below.

- **Character drops**: Random characters deleted with probability 0.1 per character
- **Character swaps**: Adjacent characters transposed with probability 0.05 per pair
- **Case flips**: Character case inverted with probability 0.1 per alphabetic character
- **Emoji insertion**: Random emoji inserted with probability 0.02 per word boundary

These perturbations simulate common typing errors and informal text patterns found in real-world user-generated content. Table 1 illustrates the transformation on a sample input.

Condition	**Text**
Clean	eBay goes phishing the popular online auction site rolls out a new approach in tackling account hackers cut bait
Noisy	ebby goes phisimg the popular onlin aubtion sis rolls out a new appobc in tackling account hackers ct bait

Table 1: Example of clean versus noisy input text after perturbation.

4 Experiment

4.1 TF-IDF + Logistic Regression

In our experiments, we include a traditional NLP baseline built with TF–IDF features (Das et al., 2021) and a multinomial Logistic Regression classifier. This model represents a lightweight and interpretable approach to short-text classification, providing a useful comparison point for evaluating how non-neural linear models behave under clean and noisy input settings.

Unlike pretrained Transformer-based models such as BERT or MiniMind, the TF–IDF representation does not capture contextual token interactions. Instead, each document is encoded as a sparse high-dimensional vector whose values reflect term importance across the corpus. Logistic Regression then learns a linear decision boundary over these features. Because this pipeline contains no pretrained parameters and relies solely on corpus-derived statistics, it serves as a clear and controlled baseline for understanding the performance gap between classical machine-learning methods and modern deep architectures.

The configuration used in our implementation is summarized in Table 2. All components follow the standard scikit-learn implementation and were not modified in our study.

Component	**Configuration**
Vectorizer	TF–IDF uni/bi-gram (1, 2) max_features = 75,000 min_df = 2 sublinear TF strip accents
Classifier	Logistic Regression (LBFGS) max_iter = 600
C selection	3-fold CV on train set candidate $C \in [0.1, 20]$ choose best C on dev using $0.55 \cdot \mathrm{F1}_{\text{noisy}} + 0.45 \cdot \mathrm{F1}_{\text{clean}}$

Table 2: TF-IDF + Logistic Regression configuration

4.1.1 Experiment Design

For the TF–IDF baseline, all experiments are conducted using a fixed TF–IDF representation fitted on the clean training texts. This feature space is reused for all subsequent training and evaluation, and the classifier is always trained on the clean training subset. We intentionally train the TF–IDF + Logistic Regression baseline only on clean data, treating it as a deployment-style model trained on curated text. This reflects common production settings for classical NLP pipelines and provides a conservative robustness baseline against which noise-aware neural models can be compared.

To select the Logistic Regression regularization strength C, we adopt a two-stage procedure. First, we perform 3-fold cross-validation over a log-spaced grid in $[0.1, 20]$ using macro-F1 as the scoring metric. Second, each candidate C is further evaluated by training a classifier on the full clean training data and computing its performance on both clean and noisy validation features. A robustness-oriented score,

$$0.55 \cdot \mathrm{F1}_{\text{noisy}} + 0.45 \cdot \mathrm{F1}_{\text{clean}},$$

is used to identify the final hyperparameter.

With the selected C, we train the final model on the complete clean training set. We report accuracy and macro-F1 on clean and noisy test features. To analyze the resulting error patterns, row-normalized confusion matrices are computed and saved for the four AG News classes. The trained TF–IDF vectorizer and classifier are exported for reproducibility.

4.2 BERT and DistilBERT

4.2.1 Model Architectures

To investigate the impact of contextualized representations on robustness, we employ BERT (Devlin et al., 2019) as our ”Teacher” model. Specifically, we utilize the bert-base-uncased architecture (110M parameters). Unlike TF-IDF, BERT's bidirectional attention mechanism allows it to leverage sentence-level context, theoretically enabling it to correct or ignore character-level perturbations during classification.

To address the high computational cost of BERT, we also evaluate DistilBERT (Sanh et al., 2019) as a ”Student” model. DistilBERT applies knowledge distillation to reduce the network depth from 12 layers to 6 layers, resulting in approximately 40% fewer parameters. We hypothesize that this distilled architecture can retain the robustness benefits of the Transformer architecture while significantly reducing inference latency.

4.2.2 Experiment Design

Both models were fine-tuned on the clean training set ensuring a fair comparison baseline. We employed a consistent training protocol of 3 epochs for both models. For the post-training, we evaluated their performance on both clean and noisy test sets. To quantify efficiency, we measured the inference time per sample on a standard CPU environment. The comparative results are presented in Table 4 and 5.

4.3 MiniMind

We evaluate MiniMind (Gong, 2024), a lightweight decoder-only Transformer inspired by GPT-style architectures, as a representative causal language model for short-text classification.

Unlike encoder-based models such as BERT (Devlin et al., 2019) or DistilBERT (Sanh et al., 2019), which employ bidirectional self-attention, MiniMind uses masked (causal) self-attention, restricting each token to attend only to previous positions. Although this design is intended primarily for text generation, it can be adapted to classification by casting the task as label generation.

We initialize MiniMind using publicly released pretrained weights and fine-tune it on the AG News dataset (Zhang et al., 2015) using supervised fine-tuning (SFT) under a causal language modeling objective. Each training example is formatted as a chat-style prompt containing the news text, followed by a target response consisting of the corresponding class label. Training minimizes the standard next-token cross-entropy loss, with a loss mask applied so that gradients are computed only over the label tokens. No additional classification head or architectural modifications are introduced. At evaluation time, the model generates a short response given the prompt, and the predicted class is obtained from the first generated label token (0–3).

The complete model architecture is illustrated in Figure 1, and the detailed configuration is provided in Table 3. These details are part of the official implementation and were not modified in our study.

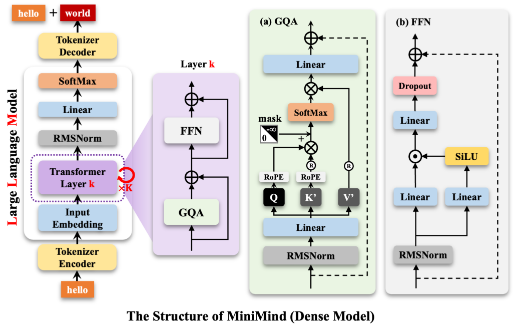

Figure 1: The architecture of MiniMind (Gong, 2024).

MiniMind-2 Parameter	Value
Total parameters	104M
Vocabulary size	6,400
RoPE θ	10^6
Number of layers	16
Hidden size (d_{model})	768
KV heads (MQA)	2
Query heads	8

Table 3: MiniMind-2 model configuration.

4.3.1 Experiment Design

We fine-tune MiniMind using a 3-epoch clean training stage to match the number of fine-tuning epochs used for BERT, enabling an epoch-matched comparison of model performance and robustness across architectures.

The baseline checkpoint (C3) is obtained after 3 epochs of supervised fine-tuning on the clean training set. To assess the impact of noise-aware training on robustness and generalization, we further continue fine-tuning C3 on the noisy training data, producing two additional checkpoints: N1 (1 epoch of noisy fine-tuning) and N3 (3 epochs of noisy fine-tuning).

4.4 Evaluation

We report classification accuracy and macro-F1 on both clean and noisy test sets. Robustness is quantified as accuracy drop: $\Delta = \text{Acc}_{\text{clean}} - \text{Acc}_{\text{noisy}}$. Lower drop indicates greater robustness. We also measure per-sample inference latency on CPU to assess computational efficiency across model families.

5 Results

5.1 TF-IDF + Logistic Regression

Table 4 summarizes the performance of the TF–IDF + Logistic Regression baseline on clean and noisy evaluation conditions. Across both validation and test sets, the model achieves strong accuracy and macro-F1 on clean inputs (approximately 0.92), while exhibiting a consistent performance degradation under noisy inputs (approximately 0.89). The roughly three-point drop is stable across both evaluation splits, indicating that this baseline is sensitive to perturbations but remains relatively robust given its purely lexical representation.

	Accuracy	Macro-F1
Validation (clean)	0.919	0.918
Validation (noisy)	0.890	0.889
Test (clean)	0.920	0.920
Test (noisy)	0.889	0.889

Table 4: Performance of TF–IDF + Logistic Regression under clean and noisy conditions.

The confusion matrices further reveal how noise affects class-level predictions. Under clean conditions, the model performs well across all four AG News categories, achieving more than 88% row-normalized accuracy for every class. The most accurately classified category is *Sports* (97.6%), while *Business* shows slightly lower performance (88.9%).

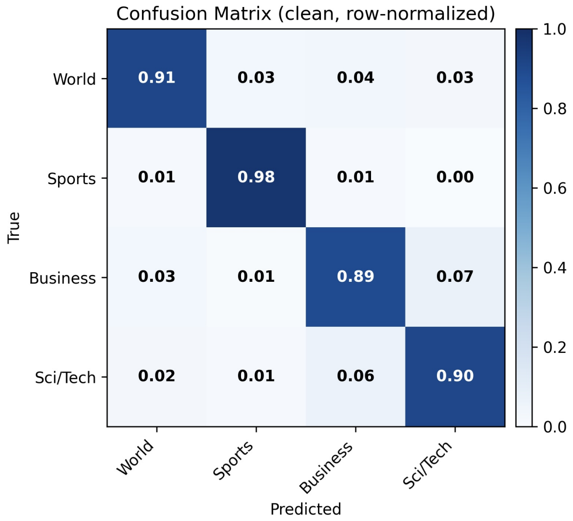

Figure 2: Row-normalized confusion matrix for the clean test set.

Under noisy inputs, we observe a clear degradation in class separation. World and Sci/Tech exhibit small but notable increases in off-diagonal confusions, and Business drops from 88.9% to 84.5%. Nonetheless, the overall class structure remains stable, suggesting that the noise-injection process does not fundamentally alter the discriminative patterns learned by the model, but rather introduces local ambiguity that affects lexical matching.

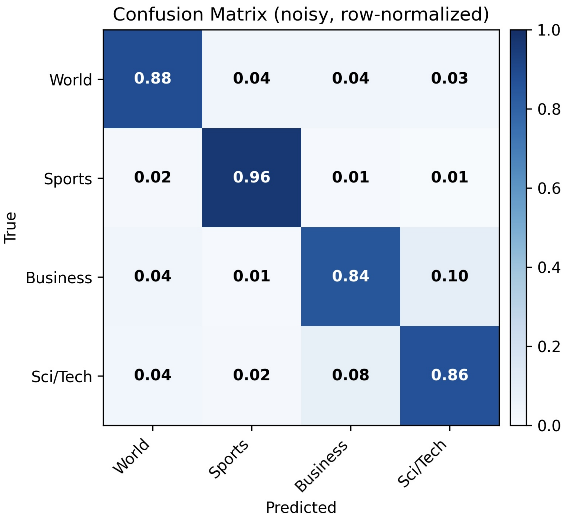

Figure 3: Row-normalized confusion matrix for the noisy test set.

These observations also validate the effectiveness of the robustness-aware hyperparameter selection strategy. The chosen regularization strength C maintains balanced performance across clean and noisy conditions, preventing the model from overfitting to clean lexical cues while preserving general discriminative capacity.

Overall, despite being a non-contextual and non-pretrained baseline, the TF–IDF + Logistic Regression model demonstrates competitive accuracy and moderate robustness, making it a suitable reference point for evaluating the benefits of more advanced neural architectures in subsequent sections.

5.2 BERT and DistilBERT

The classification performance and robustness results are summarized in Table 5. **BERT** establishes the highest performance standard among all evaluated models, achieving a clean accuracy of **94.13%**, which significantly outperforms the TF-IDF baseline (92.01%).

To visually understand the model's behavior, Figure 4 shows the confusion matrix for BERT on the clean test set. As observed, the diagonal elements are highly dominant, indicating precise classification across all categories.

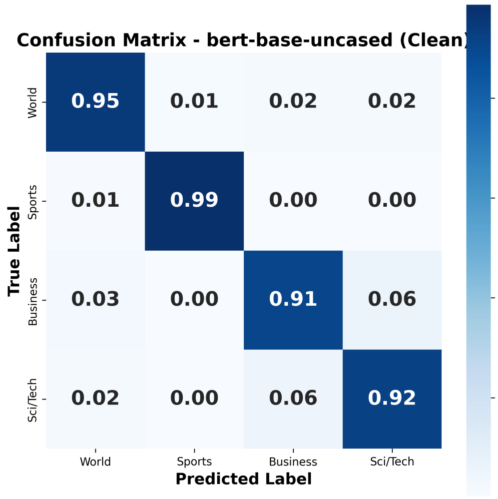

Figure 4: Row-normalized confusion matrix of BERT on clean data.

More importantly, the transformer architecture demonstrates superior robustness against character-level perturbations. While the TF-IDF baseline suffers a performance drop of 3.04% on noisy data, BERT degrades by only **1.79%** (92.34% noisy accuracy). This suggests that the self-attention mechanism effectively captures semantic context, allowing the model to be resilient to local noise.

Figure 5 illustrates the error distribution under noisy conditions. Although there is a slight increase in off-diagonal elements (particularly between Business and Sci/Tech), the overall structure remains remarkably stable compared to the clean baseline.

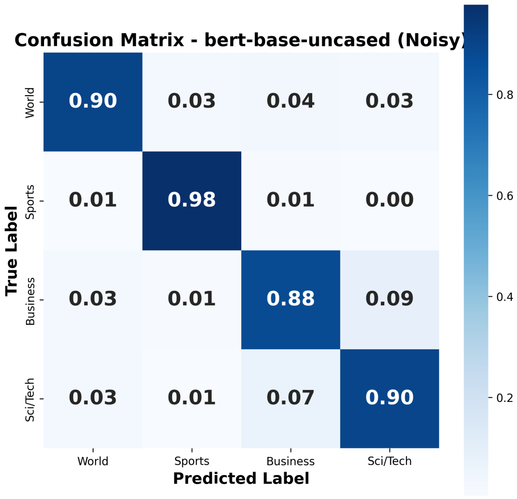

Figure 5: Row-normalized confusion matrix of BERT on noisy data.

Despite BERT's superior accuracy, its computational cost is substantial. As shown in Table 6, the original BERT model requires an average of 34.03 ms to process a single sample. In contrast, **DistilBERT** validates the effectiveness of knowledge distillation, reducing inference latency to **18.07 ms** (1.88x speedup) while retaining **99.3%** of the teacher's clean accuracy (93.49%).

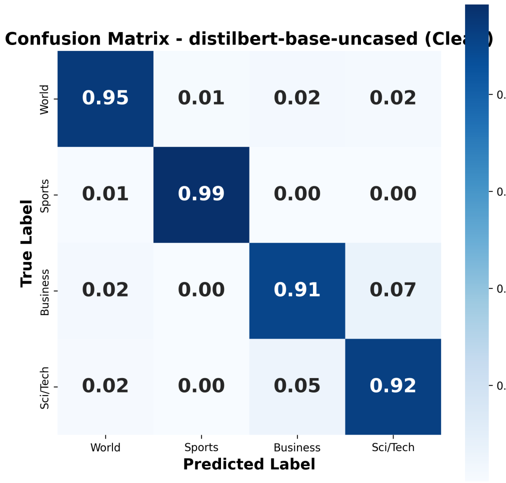

Figure 6: Row-normalized confusion matrix of DistilBERT on clean data.

Regarding robustness, DistilBERT shows a nearly identical profile to its teacher, with a performance drop of only 1.81%. As shown in Figure 7, the student model replicates the specific error patterns of BERT, such as the persistent confusion cluster between Business and Sci/Tech. This confirms that the distillation process effectively transfers both the capabilities and the semantic decision boundaries of the teacher model.

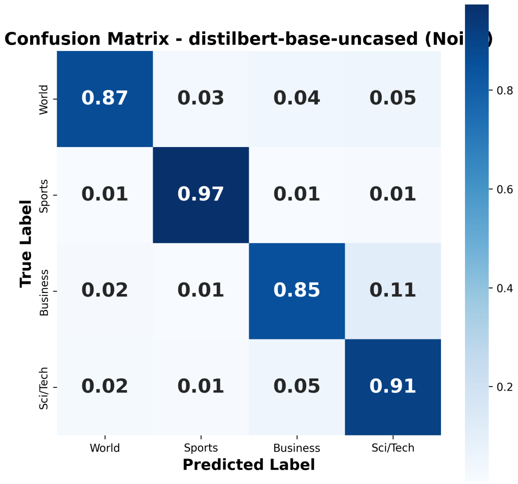

Figure 7: Row-normalized confusion matrix of DistilBERT on noisy data.

Model	Clean (%)	Noisy (%)	Drop (%)
TF-IDF (Base)	92.01	88.97	−3.04
BERT (Ours)	**94.13**	**92.34**	**−1.79**
DistilBERT (Ours)	93.49	91.68	−1.81

Table 5: Robustness evaluation on clean vs. noisy test sets.

Model	Infer Time (ms)	Speedup
TF-IDF (Base)	0.025	1361.2
BERT (Teacher)	34.03	1.0
DistilBERT (Student)	**18.07**	**1.88**

Table 6: Computational efficiency on CPU.

5.3 GPT / MiniMind

We evaluated four MiniMind checkpoints: C1 (1 clean epoch), C3 (3 clean epochs), N1 (C3 + 1 noisy epoch), and N3 (C3 + 3 noisy epochs). All checkpoints are evaluated on identical clean and noisy test sets. Table 7 summarizes the results.

Checkpoint	**Clean Acc**	**Noisy Acc**	**Drop**
C1	0.5269	0.4426	0.0844
C3	0.7066	0.6058	0.1008
N1	0.7286	0.6884	0.0401
N3	0.7621	0.7216	0.0405

Table 7: MiniMind results across different training stages. Drop indicates clean accuracy minus noisy accuracy.

Moving from C1 to C3 significantly improves both clean and noisy accuracy, reflecting better overall convergence with additional training.

However, MiniMind trained only on clean data (C3) shows greater sensitivity to noise perturbations than BERT, with a 10.1% accuracy drop on noisy input. Introducing noise-aware fine-tuning (N1, N3) substantially improves robustness, reducing the performance gap to approximately 4% while simultaneously improving clean accuracy.

To better understand MiniMind's behavior across training stages, we visualize confusion matrices for C3 and N3 on both clean and noisy test sets. These four matrices highlight how noise affects prediction patterns and how noise-aware fine-tuning improves robustness.

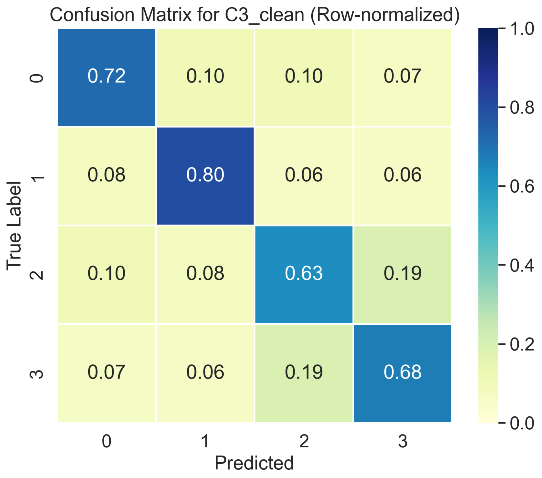

Figure 8: Confusion matrix for C3 on clean test data.

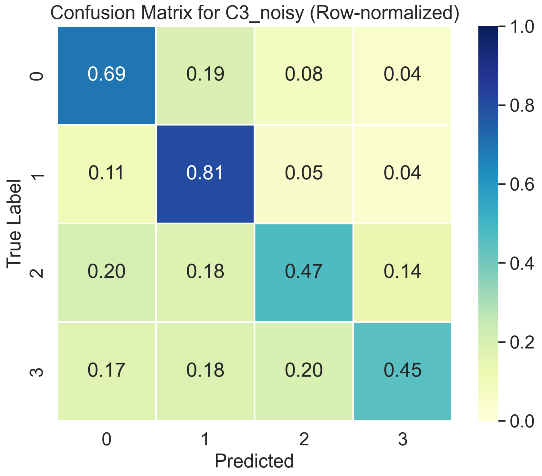

Figure 9: Confusion matrix for C3 on noisy test data.

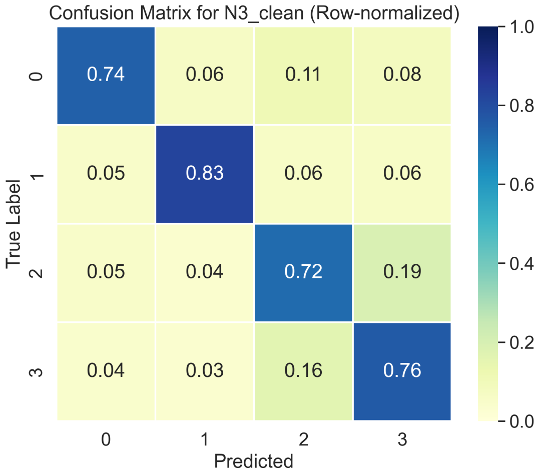

Figure 10: Confusion matrix for N3 on clean test data.

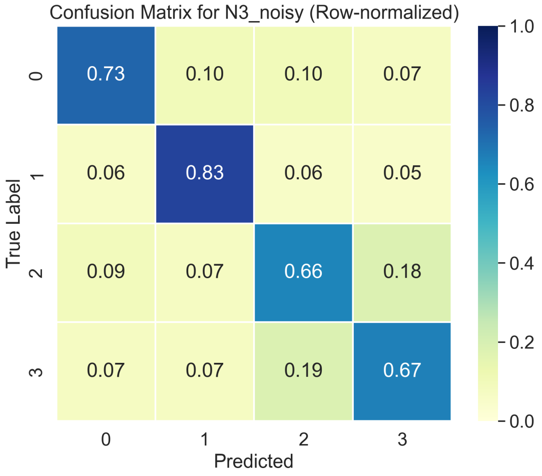

Figure 11: Confusion matrix for N3 on noisy test data.

Figure 8 and Figure 9 show the confusion matrices for the clean-only model (C3). This model performs reasonably well on clean input but exhibits noticeable confusion between classes 2 (Business) and 3 (Sci/Tech). Under noisy input, this confusion becomes substantially worse—class 3 correct prediction rate drops from 0.68 to 0.45. This confirms that decoder-only models trained only on clean data are highly sensitive to character-level perturbations.

In contrast, the noise-aware model (N3) (Figures 10 and 11) shows more stable confusion patterns across both conditions. Diagonal values strengthen for all classes and most off-diagonal confusions decrease. However, Business/Sci-Tech confusion remains strong even after noise-aware training, suggesting these categories share vocabulary and news topics that create intrinsic ambiguity beyond what noise training alone can resolve.

Checkpoint	Clean Infer (ms)	Noisy Infer (ms)	Avg Infer (ms)
C3	14.96	15.16	15.06
N3	15.19	14.96	15.08

Table 8: Per-sample inference time on clean and noisy test sets. Avg Infer denotes the mean latency across clean and noisy conditions.

Table 8 reports the inference time every sample for representative clean-only and noise-aware MiniMind checkpoints. We observe that inference time remains stable across training stages and input conditions, with all configurations averaging approximately 15 ms per sample. Noise-aware fine-tuning does not introduce measurable inference overhead, indicating that the robustness gains of N3 come without additional runtime cost.

6 Evaluation and Analysis

6.1 Overall Comparison

Across all experiments, we observe a consistent accuracy–latency tradeoff across model families. BERT achieves the highest accuracy (94.13% clean, 92.34% noisy) but incurs the largest inference cost (34 ms per sample). At the opposite extreme, the TF–IDF baseline provides sub-millisecond inference while maintaining competitive accuracy (92% clean, 89% noisy). DistilBERT offers the most favorable balance, retaining 99.3% of BERT's clean accuracy while achieving a 1.88× speedup and comparable robustness.

6.1.1 Model Selection Guidelines.

Based on these tradeoffs, we recommend:

- **Latency-critical applications**: TF–IDF + Logistic Regression achieves competitive accuracy (92% clean / 89% noisy) with sub-millisecond inference and minimal infrastructure requirements.

- **Balanced production systems**: DistilBERT offers the best efficiency-accuracy tradeoff, achieving 99.3% of BERT's performance at 1.88× the speed. This is the recommended default for most real-world deployments.

- **Accuracy-critical offline tasks**: BERT remains the gold standard when latency is not a constraint, providing the highest accuracy and strongest robustness.

- **Resource-constrained or experimental settings**: MiniMind with noise-aware fine-tuning demonstrates that smaller decoder-only models can achieve reasonable performance (76%/72%) when properly trained, though they lag behind encoder architectures on this task.

6.2 Key Findings

Our experiments yield several insights into how architectural choices and training strategies affect robustness in short-text classification.

Bidirectional context substantially improves noise robustness. Encoder-based models with bidirectional self-attention exhibit markedly greater resilience to character-level perturbations than decoder-only models trained under a causal objective. BERT experiences only a 1.79% accuracy drop under noise, compared to a 10.1% drop for MiniMind trained solely on clean data, suggesting that access to full sentence context enables effective compensation for locally corrupted tokens—an advantage unavailable to decoder-only models restricted to causal attention.

Model compression via distillation preserves robustness. Despite having approximately 40% fewer parameters, DistilBERT closely mirrors BERT's robustness profile, with nearly identical degradation under noise (1.81% vs. 1.79%). This indicates that knowledge distillation transfers not only predictive accuracy but also noise-tolerant decision boundaries, making DistilBERT a strong efficiency-oriented alternative to full-scale encoders.

Noise-aware fine-tuning improves both robustness and clean accuracy for decoder-only models. For MiniMind, continued fine-tuning on noisy data substantially reduces sensitivity to perturbations, shrinking the clean–noisy accuracy gap from 10.1% to approximately 4%. Notably, noise-aware fine-tuning also improves clean accuracy (from 70.66% to 76.21%), indicating that noise acts as effective regularization rather than merely increasing robustness at the expense of clean performance.

Lexical baselines remain surprisingly competitive under noise. Despite lacking contextual representations, the TF–IDF + Logistic Regression baseline achieves 92.01% clean accuracy with only a 3.04% drop under noise. This demonstrates that for short, topic-focused text, sparse lexical features can retain strong discriminative power, particularly in latency-constrained settings where neural models may be impractical.

6.3 Error Analysis

Confusion matrix analysis reveals consistent error patterns across models. The Business and Sci/Tech categories exhibit the highest mutual confusion rates, likely due to overlapping vocabulary in technology business news. This confusion persists even after noise-aware training, suggesting inherent semantic similarity between these categories that extends beyond surface-level perturbations.

Under noisy conditions, all models show increased off-diagonal predictions, but the degradation is not uniform across classes. Sports classification remains highly stable across all models (97.6% for TF-IDF, similar for BERT), likely because sports vocabulary is distinctive and less susceptible to ambiguity from character-level noise. World news shows moderate degradation, while Business exhibits the largest drops, potentially due to its reliance on precise numerical and entity information that noise corrupts.

7 Conclusion

We compared the robustness of TF–IDF + Logistic Regression, encoder-based Transformers (BERT, DistilBERT), and a decoder-only model (MiniMind) for short-text classification under character-level noise on AG News. Encoder architectures were the most robust: BERT achieved the highest accuracy (94.13% clean, 92.34% noisy; 1.79% drop), while DistilBERT preserved comparable robustness (1.81% drop) at a 1.88× CPU inference speedup, retaining 99.3% of BERT's clean accuracy.

TF–IDF remained competitive for latency-critical settings (92.01% clean, 88.97% noisy) with sub-millisecond inference. MiniMind trained only on clean data was substantially more noise-sensitive (10.1% drop), but noise-aware fine-tuning reduced this gap to $\approx$4% while also improving clean accuracy (70.66% $\rightarrow$ 76.21%). Overall, DistilBERT offers the best robustness–efficiency tradeoff for deployment, TF–IDF suits extreme latency constraints, and noise-aware training is important for improving model robustness.

8 Future Work

Future directions include extending our noise model to word-level perturbations and real-world corruptions (e.g., OCR errors, social media text), evaluating larger decoder-only models, and combining noise augmentation with adversarial training. Investigating model calibration under noise would also inform deployment in uncertainty-critical applications. Additionally, our inference time comparisons were conducted across different hardware; future work should benchmark all models on identical CPU environments for more interpretable efficiency comparisons.

References

Mamata Das, Selvakumar Kamalanathan, and P.J.A. Alphonse. 2021. A comparative study on TF-IDF feature weighting method and its analysis using unstructured dataset. In *Proceedings of the 5th International Conference on Computational Linguistics and Intelligent Systems (COLINS)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jingyao Gong. 2024. MiniMind: Train a tiny LLM from scratch. https://github.com/jingyaogong/minimind.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, volume 28.