Empirical Study on the Effectiveness of DistilBERT Fine-tuning on IMDb Sentiment Classification Outperforming CNN/LSTM

Shengkai Yuan

College of Engineering, University of California, Davis, CA 95616, United States

skyuan@ucdavis.edu

Abstract. Sentiment analysis of user reviews is a core Natural Language Processing (NLP) task with practical uses in real scenarios like recommendation. However, the traditional approach like training neural networks models such as CNN, RNN, and LSTM, has faced challenges in improving recognition accuracy. With the development of Language Models (LM) with their self-attention mechanism for contextual understanding, this research wants to see if Language Models exceed Neural Networks on this task. This study conducts a controlled comparison on the IMDb Large Movie Review Dataset (50K reviews) for binary sentiment classification of long movie reviews. The evaluation of four models is based on the dataset that has been cleaned with max sequence length 256, and a stratified 8:1:1 train/validation/test split with multi-seeds. On the IMDb test set, TextCNN performs with an accuracy of 0.873, LSTM reaches 0.862, while DistilBERT achieves 0.911, consistently outperforming strong CNN/LSTM baselines by about 4%. In addition, an accuracy and latency trade-off is observed: DistilBERT offers the best quality with moderate runtime, while TextCNN/LSTM deliver lower latency. Overall, results confirm that a compact pretrained model with fine-tuning provides clear quality gains on long, nuanced reviews, while traditional approaches remain attractive when speed and simplicity are the priority.

Keywords: IMDb Sentiment Classification, fine-tuning, deep learning.

1. Introduction

Sentiment analysis of user reviews is a core Natural Language Processing (NLP) task with practical uses in real life, such as recommendation, reputation management, and product decisions. Among public benchmarks, the IMDb Large Movie Review Dataset, 50K reviews with a balanced 25k/25k split [1], is widely used for binary sentiment classification of long movie reviews. Movie reviews often contain linguistic phenomena, such as negation, sarcasm, or irony. The peculiarity in linguistics in combination with the polarity of movie reviews, making robust modeling non-trivial. There are a few surveys and studies that emphasize these challenges. For example, sarcasm, can shift the intended polarity of a review, as in the statement "The film looked brilliant—yet I walked out after twenty minutes." where the positive phrase "The film looked brilliant" is contradicted by the negative action of walking out mid-way, thus signaling a sarcastic tone and ultimately a negative sentiment [2].

In the early days of text classification, researchers unusually wielded convolutional and recurrent neural networks. They were fast to train, easy to deploy, and delivered solid results without needing much fine-tuning. In particular, simple convolutional models that aggregate local n-gram features via max pooling (often called "TextCNN"), first introduced by Kim in Convolutional Neural Networks for Sentence Classification, established strong baselines for sentence and document classification with minimal feature engineering [3]. While Recurrent Neural Network (RNN) based models, including Long short-term memory (LSTM), are good at handling sequences and avoiding vanishing gradients, they tend to compress long reviews into a single fixed-length vector. This often makes it hard for them to fully capture deeper meanings, especially when important signals are spread out (e.g. a negation early in the sentence or a contrasting point introduced later). Beyond modeling limits, Dodge et al. (2019) highlight that many NLP experiments lack reproducibility. Currently, published results regarding traditional NLP studies vary widely because preprocessing steps, tokenization lengths, dataset splits, random seeds, and hyperparameter budgets are rarely controlled in a consistent manner [4]. The field of NLP requires an innovative, unified paradigm on model training.

The introduction of transformer architecture completely changed how NLP works. The pretrained transformer models, particularly its encoder structure, learn deep bidirectional contextual representations from large amounts of raw texts (unlabeled corpora). With the post-training techniques like fine-tuning, pretrained models are able to adapt to specific tasks such as sentiment analysis and question answering. Among the transformer open-source models, BERT has been proven to have strong performance marks on a range of text classification benchmarks [5]. Under the scenario where computing resources are limited, the DistilBERT model introduced by Sanh et al. (2019) applies knowledge distillation. This creates a 40% smaller and 60% faster model, but remains about 97% of BERT's language understanding capabilities [6]. Such a small model is well-suited for the binary sentiment analysis on movie reviews.

Although some studies have shifted toward pretrained models like DistilBERT, it is still unclear how much they perform beyond trained CNN and LSTM baselines with a higher efficiency. This is especially true in experiment and prototyping settings, where it's important to control all variables like preprocessing, data splits, sequence length, and training steps. Without such a fair comparison, it is hard for scientists and scholars to decide whether the extra complexity of pretrained models is worth it. To address this, this study asks a simple question: under a consistent and reproducible setup, how much better does DistilBERT perform compared to traditional CNN, RNN, and LSTM models in terms of accuracy, F1 score, and training efficiency? With these challenges in mind, this study conducts a controlled comparison and previews the key finding: a compact pretrained model (DistilBERT) consistently outperforms strong CNN, RNN and LSTM baselines on IMDb, while introducing a clear accuracy–latency trade-off. Section 2 details the experimental setup; results and analyses follow in Section 4.

2. Method

2.1. Dataset preparation

The IMDb Large Movie Review Dataset includes 50,000 English movie reviews, with a positive or negative label on each review, and with a balanced distribution. This study will evaluate four models (RNN, LSTM, TextCNN, and fine-tuned DistilBERT) on the IMDb 50K dataset using a stratified 8:1:1 train/validation/test split. Stratified sampling preserves the 50/50 class ratio in train (40k), validation (5k), and test (5k) subsets. This study takes the multi-seeds setting between 42 to 44 for a more accurate result for all split operations, vocabulary construction, and weight initialization (where applicable) to maximize reproducibility.

This study treats this as a binary sentence classification task. To avoid leakage from metadata or HTML markup embedded in the raw dataset, this paper standardizes the input through a lightweight normalization pipeline. The pipeline removes HTML artifacts and URLs and lowercases text. After HTML tag removal (including
), this article filters non-alphabetic characters except spaces, collapses multiple spaces, and trim leading or trailing spaces. Any review that becomes empty after cleaning is skipped during minibatch collation but retained as a zero-length mask so indices remain stable.

Regarding the neural network baseline models (RNN, LSTM, and TextCNN), This study constructs a word-level vocabulary from the split of training (about 40k most frequent tokens), and reserves special symbols for <PAD> and <UNK>. Reviews are tokenized by whitespace, mapped to integer ids, and truncated to length of 256. This ensures the identical sequence length and comparable input statistics across baselines. For DistilBERT, tokenization is handled by the model's WordPiece tokenizer with max_length=256, truncation=True, and padding="max_length" so that the encoded inputs align with the same length budget.

2.2. DistilBERT

DistilBERT, a distilling model trained by Sanh et al. (2019), is a smaller, general-purpose version of BERT [6]. With the design of a smaller "student" transformer model, DistillBERT was trained by

knowledge distillation during the pre-training stage on large unlabeled text. The masked-LM loss, including both distillation loss matching "teacher"'s model (BERT) output distribution and a cosine loss aligning the hidden states. This article utilizes this lighter and faster encoder which contains most of BERT's ability on language understanding to fine-tune on IMDb sentiment classification.

This study initializes the public distilbert-base-uncased checkpoint and incorporates a single linear layer to deal with two labels (negative/positive). Every review is processed with the model's default tokenizer, which converts it to lower case, and is clipped to 256 tokens to keep the input length consistent with the baselines. DistilBERT is fine-tuned under the normal setup of AdamW scheduling along with the mixed-precision option where available. Generally, for the training stage of this study, this study applies three epochs along with a small learning rate (2e-5) and a light weight decay (0.01). An initial short warm-up period is offered at the beginning of training. For training, the batch size is set to 16, while it is 32 for testing. With the presence of a GPU, the automatic mixed-precision reduces memory use and accelerates the training process without making any changes in the model's logic. This study performs the evaluation for the validation set after each epoch and retains the passed F1 best one for the testing and also for the demo service.

2.3. Hyperparameter configuration

Classical baselines are trained under exactly the same conditions, then they have the same vocabulary and embedding setup, they are trained with identical optimizers, and early-stopping rules; hence models can be compared without considering preprocessing or training budgets. For all three baselines, this study creates a word vocabulary using the training split and applies 128-dimensional embeddings. The RNN baseline is implemented by using a single bidirectional recurrent layer with 128 hidden units plus a linear classifier as the last layer. The LSTM baseline has the same structure but replaces the recurrent layer with a bidirectional LSTM. The TextCNN baseline uses three convolution windows (widths 3, 4, and 5) with 100 filters each, performs global max-pooling, and concatenates the vector to a linear classifier. This article applies these baselines AdamW (learning rate 2e-3), a batch size of 128, gradient clipping at 1.0 for six epochs. The checkpoint, which achieves the highest validation F1, is selected before running on the test set. DistilBERT implements the fine-tuning strategies set forth in Section 2.2. The loss function for all models is binary cross-entropy with logits.

This study reports Accuracy and F1 on the held-out test set, track all runs with MLflow, save best checkpoints, and expose a minimal FastAPI endpoint for inference. For completeness, experiments are run with Python 3.10, PyTorch and the Hugging Face Transformers library, and this study fixes the random seed at 42 to 44 with multiple trainings across libraries to make results precise.

3. Results and discussion

3.1. The performance of various models

3.1.1 Traditional baselines model performance

Table 1 provides the results of model's performance with traditional approaches. On the IMDb test set, TextCNN attains Acc = 0.873 ± 0.001 and F1 = 0.877 ± 0.001, while Bi-LSTM reaches 0.862 ± 0.013 and 0.866±0.011, and the simple RNN lags far behind at 0.501 ± 0.003 and 0.634 ± 0.004. Under the unified protocol setting of this study (max length 256, identical cleaning and splits), TextCNN emerges as the strongest classical baseline, with LSTM close behind.

Table 1. RNN, LSTM, and TextCNN's performance

model	best_val_acc_mean	best_val_acc_std	best_val_f1_mean	best_val_f1_std	test_acc_mean	test_acc_std	test_f1_mean	test_f1_std
CNN	0.877	0.002	0.879	0.003	0.873	0.001	0.877	0.001
LSTM	0.868	0.010	0.869	0.010	0.862	0.013	0.866	0.011
RNN	0.512	0.001	0.635	0.001	0.501	0.003	0.634	0.004

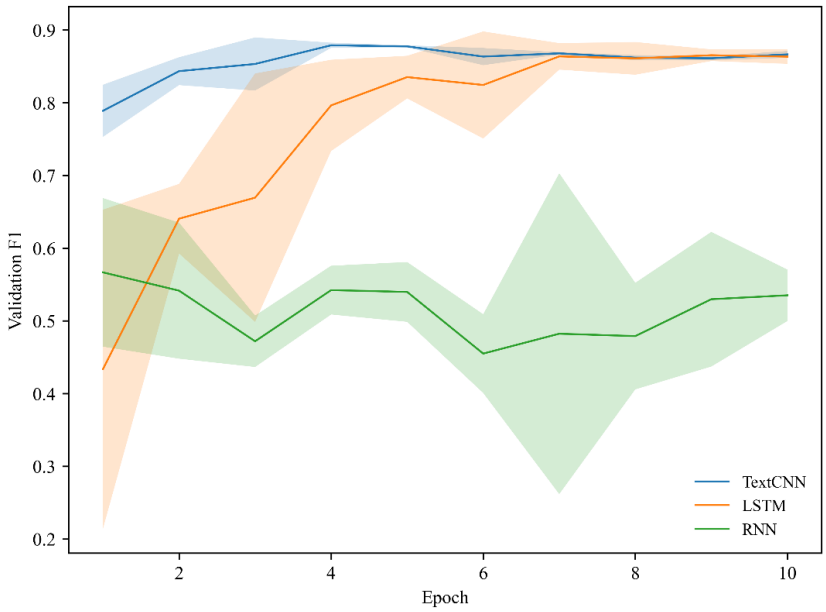

Figure 1. Traditional baselines model performance (Picture credit: Original)

Fig. 1 demonstrates the performance of various models originated from the traditional approach. From the graph, TextCNN with its fitness to text performs well at the start of the training, maintaining its F1 score above 0.8 to the end. LSTM begins with a low accuracy, but it grows gradually to 0.8 F1 score, matching its strength on sequential and contextual understanding. However, RNN performs worst, which means that it is not suitable for this kind of task.

3.1.2 DistilBERT

Table 2. DistilBERT' performance

	best_val_acc_mean	best_val_acc_std	best_val_f1_mean	best_val_f1_std	test_acc_mean	test_acc_std	test_f1_mean	test_f1_std
DistilBERT	0.913	0.001	0.913	0.003	0.911	0.003	0.911	0.004

Table 2 provides the results of DistilBERT's performance on the task of IMDb reviews classification. It achieves an average of 0.913 with 0.001 standard deviations on the accuracy on validation set with multi-trials, and an average of 0.913 with 0.003 standard deviations.

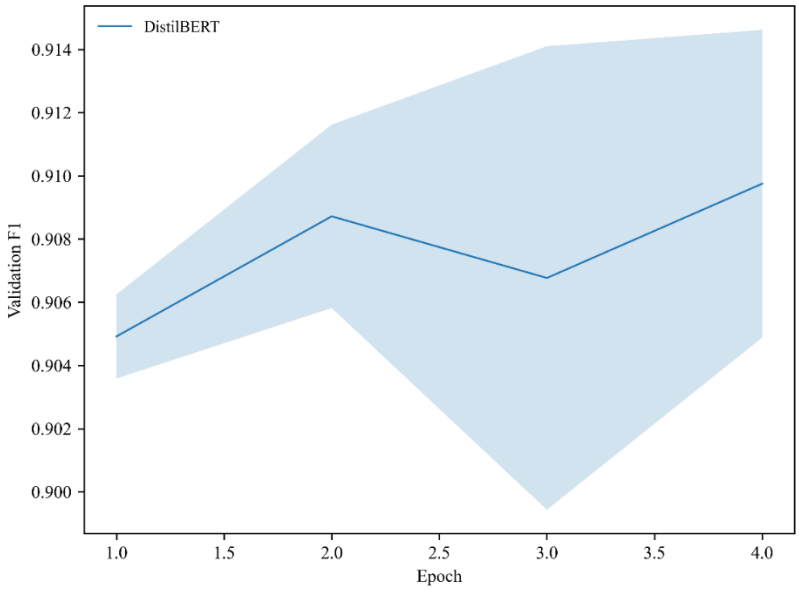

Figure 2. DistilBERT' performance (Picture credit: Original)

Fig. 2 illustrates that DistilBERT has a steady increase in its performance, where its Validation F1 score increases from an average of 0.9045 on the first epoch to an average 0.911 on the last epoch.

3.1.3 Comparisons

Table 3. Comparisons of models' performancee

Model	Accuracy	F1	Params	Latency-1(ms)	Latency-32(ms)
CNN	0.873±0.001	0.877±0.001	3994502	0.9	0.0
LSTM	0.862±0.013	0.866±0.011	4104706	1.5	0.0
RNN	0.501±0.003	0.634±0.004	3906562	0.6	0.0
DistillBERT	0.911±0.003	0.911±0.004	66955010	6.3	1.6

Table 3 provides the main results of different approaches on the IMDb test set under a unified protocol (seeded runs, max length 256). DistilBERT achieves the best accuracy and F1, exceeding the classical baseline models such as TextCNN and LSTM by around 4% to 5% on average.

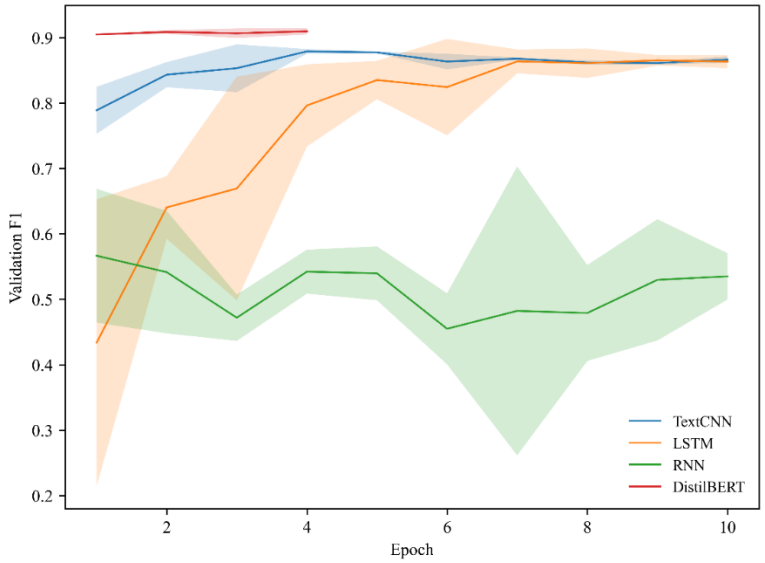

Figure 3. Comparisons of models' performance (Picture credit: Original)

Fig. 3 shows that DistilBERT maintain a horizontal straight line with around 91.1% f1 score, converging faster than traditional neural networks. TextCNN improves steadily and saturates later; LSTM lags early but catches up after several epochs. RNN remains unstable, illustrating its difficulty in modeling long-range dependencies.

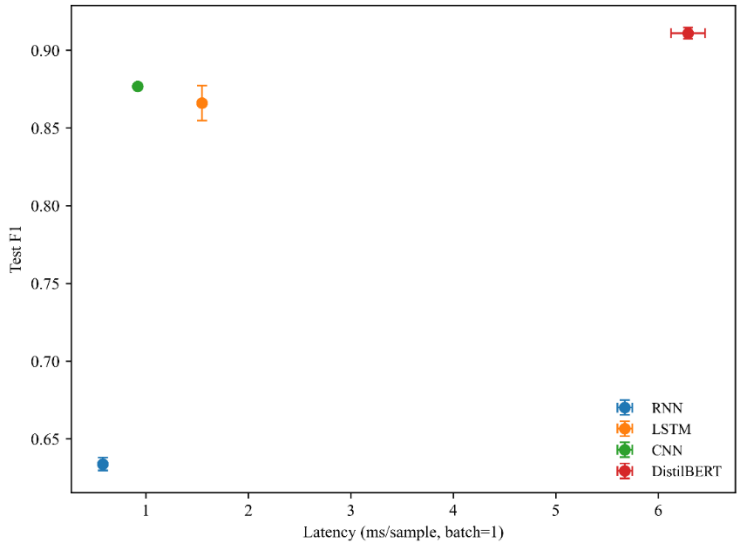

Figure 4. Accuracy-Latency Trade-off (Picture credit: Original)

Fig. 4 offers a visualization of the relationship between accuracy and latency. DistilBERT offers the best quality with moderate runtime on a single RTX 3090; however, TextCNN and LSTM deliver lower latency at the cost of a few F1 points, showing its usefulness when a strict runtime or memory limits are required.

Overall, although it is clear that DistilBERT obtains a significant lead compared to the traditional approaches, its latency and GPU computing cost are far more than the training cost of traditional neural networks. The choice of model on IMDb reviews classification or similar binary NLP tasks should be based on actual situations.

3.2. Discussions

3.2.1 Language Model superiority on text processing than Neural Networks

Traditional neural networks, such as TextCNN, rely on local n-gram convolutions and max-pooling, demonstrating limited sensitivity to long-range dependencies and long tail situation (such as contrastive relationships and scope of negation). In contrast, DistilBert is a language model based on transformer architecture. The architecture of transformer, proposed by Google researchers [7], mentions that its self-attention mechanism can directly construct global dependencies and cross-sentence relationships.

This explains why DistilBERT achieves an around 4% accuracy higher than TextCNN and LSTM on IMDb reviews classification, where long sentences and negations occur frequently. The phenomenon matches the literature review of text classification done by Minaee et al. (2021). In their study, Multiway Attention Network has achieved a 88.3% accuracy on the natural language inference task, where LSTM encoders only got a 77.6% and Tree-based CNN got a 82.1% [8].

3.2.2 Advantages from Pre-training and Post-training: Distillation and Fine-tuning

Pre-training enables models to learn general language knowledge from large-scale unlabeled data, where fine-tuning aligns this knowledge to the specific sentiment task. DistilBERT preserves much of BERT's capacity while being faster, which aligns with its superior F1 under the controlled setting of this research. Compared to the neural networks (TextCNN, RNN, LSTM) trained by this paper using the IMDb dataset, the pre-trained BERT model possesses a larger parameter count (110 million) and was trained on a significantly larger dataset (approximately 3.3 billion tokens) (Devlin et al., 2019). These data and computational resources are clearly not available for most researchers. Even after distillation, DistilBERT maintains strong performance with a 97% performance but 40% smaller than BERT, laying a solid foundation for subsequent fine-tuning [6].

3.2.3 Trade-off between latency, cost, and accuracy

Under the same protocol, DistilBERT delivers the best quality but at a higher runtime and compute cost. In Table 3 and Fig. 4, single-sample (batch=1) latency is in the single-digit millisecond range for DistilBERT (≈6.3 ms on a single RTX 3090), while TextCNN and LSTM run notably faster at sub-millisecond to low-millisecond levels. When the batch size grows (e.g., batch=32), DistilBERT amortizes its cost and narrows the gap, but the classical baselines still remain the most responsive when strict latency or memory limits are the first priority (mobile/CPU, real-time UI, or tight service-level objectives). In short: if the goal is maximum accuracy/F1 on long reviews and a small latency budget is acceptable, DistilBERT is a good default; if the goal is minimal delay and very small footprints, a tuned TextCNN/LSTM is a practical choice.

3.2.4 Limitations and improvements

First, this study only reported rough inference latency for a single RTX 3090 card; in the future, comparisons may incorporate INT8/quantization and distillation retraining for a higher efficiency. Second, this study only provides the numerical results, lacking explainability. In the further study, the incorporation of attention and gradient-based analysis will be applied to support error discussions [9, 10].

4. Conclusion

This study provides a controlled comparison of TextCNN, RNN/LSTM, and DistilBERT on IMDb sentiment classification. Under the same preprocessing, sequence length (256), and splits, DistilBERT achieves the highest Accuracy/F1, outperforming the strongest classical baseline by abwout 3 to 4 F1 points, while TextCNN/LSTM offers lower latency an smaller computing resource. These results confirm that a small pretrained model with fine-tuning, delivers clear quality gains on long, nuanced reviews, but at a higher runtime cost. Overall, the choice of model should reflect deployment goals. DistilBERT is encouraged when accuracy is the priority and slight delay is acceptable; however, when speed and simplicity are important, traditional approaches like TextCNN/LSTM is prefered.

References

[1] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, C. Potts, Learning word vectors for sentiment analysis, Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, Oregon, USA, 142–150 (2011). http://www.aclweb.org/anthology/P11-1015.

[2] D. V. N. Devi, K. VijayBhaskar, Y. Pavan, Survey on detection of sarcasm in sentiment analysis, Journal of Engineering Research and Application 9 (10, Series II), 39–44 (2019). http://www.ijera.com/papers/Vol9_issue10/Series-2/G0910023944.pdf.

[3] Y. Kim, Convolutional neural networks for sentence classification, Proceedings of EMNLP 2014 (2014).

[4] J. Dodge, S. Gururangan, D. Card, R. Schwartz, N. A. Smith, Show your work: Improved reporting of experimental results, Proceedings of EMNLP-IJCNLP 2019 (2019).

[5] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv: 1810.04805 (2019).

[6] V. Sanh, L. Debut, J. Chaumond, T. Wolf, DistilBERT: Smaller, faster, cheaper and lighter, arXiv preprint arXiv: 1910.01108 (2019).

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in Neural Information Processing Systems (NeurIPS) 30 (2017). https://papers.nips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[8] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, J. Gao, Deep learning–based text classification: A comprehensive review, ACM Computing Surveys 54 (3), Article 62, 1–40 (2021). https://doi.org/10.1145/3439726.

[9] G. W. Lindsay, Attention in psychology, neuroscience, and machine learning, Frontiers in Computational Neuroscience 14, 29 (2020).

[10] M. Maurya, N. Yadav, A comparative analysis of gradient-based optimization methods for machine learning problems, International Conference on Data Analytics and Computing, Singapore: Springer Nature Singapore, 85–102 (2022).