Probing vs Fine-Tuning: Benchmarking BERT on News Classification

Jack Parry-Wingfield

April 2025

Abstract

In this project, we investigated the performance of the bert-base-uncased model on the AG News classification task, comparing classical probing approaches against end-to-end fine-tuning. For probing, using the whole dataset (all 120,000 examples) we evaluated multiple sentence-level embedding strategies (`[CLS]` token, mean pooling, and last token) using Logistic Regression and K-Nearest Neighbors as downstream classifiers. Mean-pooled embeddings paired with Logistic Regression yielded the best probing results, achieving a test accuracy of 91.13%. We then fine-tuned all layers of BERT using a 60,000-sample subset and achieved a significantly higher test accuracy of 94.51%. This 3.38 percentage point gain demonstrates the power of task-specific adaptation through gradient-based learning. We also conducted qualitative analysis using attention visualizations and found that BERT focuses more effectively on semantically informative tokens in correct predictions, while attending more diffusely in incorrect ones. Observing our findings below, we see that while probing can offer strong baselines, full fine-tuning is necessary to unlock BERT's full classification performance.

1 Introduction

Transformer-based language models like BERT [**devlin2018bert**] have revolutionized natural language processing (NLP) through pretraining on large corpora followed by task-specific fine-tuning. BERT learns contextual representations by leveraging a masked language modeling objective, making it highly effective for a wide range of downstream tasks such as question answering, sentence classification, and named entity recognition [**rogers2020primer**]. Previous research has shown that BERT encodes a wide array of syntactic and semantic information [**hewitt2019structural**, **jawahar2019does**], leading to strong transfer performance even when the model is frozen and used only as a feature extractor.

Despite this, the relative performance of probing versus full fine-tuning remains a subject of empirical investigation, especially under varying data regimes and model configurations [**peters2019tune**]. In this project, we conduct a comparative study of probing and fine-tuning using the AG News dataset [**zhang2015character**], a four-class topic classification benchmark. We explore how well frozen BERT embeddings perform when used with traditional classifiers such as Logistic Regression and K-Nearest Neighbors, and contrast that against fully fine-tuned BERT models.

Our experiments reveal that mean-pooled embeddings combined with Logistic Regression yield strong performance (91.13% accuracy), but full fine-tuning of BERT leads to significantly higher test accuracy (94.51%). This performance gap illustrates the value of end-to-end optimization, though probing remains a viable alternative in low-resource settings. Attention visualizations further demonstrate how the model's focus shifts based on prediction correctness, offering interpretability into BERT's decision-making process.

We also investigate the trade-offs between these two strategies in terms of performance and computational cost. While fine-tuning requires more GPU memory and training time, it offers the opportunity for task-specific adaptation. To better interpret model behavior, we analyze attention maps from the fine-tuned model using `bertviz` [**vig2019bertviz**]. Through these experiments, we aim to clarify when and why end-to-end fine-tuning leads to measurable performance gains, and to what extent probing can serve as a viable alternative.

2 Datasets

The AG News dataset [**zhang2015character**] is a standard benchmark for topic classification, originally curated from over 1 million news articles collected by the ComeToMyHead academic news search engine. These articles span more than 2,000 news sources and cover a broad range of categories. The dataset was processed and released as a four-class classification benchmark by Zhang et al. (2015), containing 120,000 training and 7,600 test samples split evenly across four categories: World, Sports, Business, and Sci/Tech.

Each example in the dataset consists of a short news title and description. For probing experiments, we used the full training set of 120,000 examples to extract sentence-level representations. For fine-tuning, we sampled a subset of 60,000 examples, with 20% used for validation.

We used the Hugging Face datasets library to load the dataset using the default train/test splits:

```
from datasets import load_dataset
train_dataset = load_dataset('ag_news', split='train')
test_dataset = load_dataset('ag_news', split='test')
```

All text inputs were preprocessed using the BERT tokenizer with a maximum sequence length of 128 tokens. Token-level embeddings were obtained using bert-base-uncased, and no manual feature engineering was required, as BERT's internal representations and attention mechanisms handled contextual encoding. Exploratory analysis revealed balanced class distributions and strong lexical signals across categories which serves to justify its use as a robust benchmark for both probing and fine-tuning approaches.

3 Benchmark

Our experiments use the bert-base-uncased architecture, which consists of 12 transformer layers, each with 12 attention heads and a hidden size of 768. For probing (Experiment 3.1), we extracted three types of sentence embeddings: the [CLS] token output, the mean of all token embeddings (excluding padding), and the final token embedding. These were used as features for both Logistic Regression (max_iter = 1000) and K-Nearest Neighbors (K $\in$ [1, 20]). We selected the best K based on validation accuracy.

For fine-tuning (Experiment 3.2), we used the BertForSequenceClassification model and trained for 3 epochs using AdamW with a learning rate of 2e-5 and linear scheduling. A batch size of 16 was used for training and 32 for validation/testing. The model was evaluated on the AG News test set after selecting the checkpoint with highest validation accuracy. Training was conducted on a Colab A100 GPU.

We also performed additional analysis to evaluate the impact of Logistic Regression's max_iter and training subset size. Finally, in Experiment 3.4, we used the bertviz library [**vig2019bertviz**] to examine attention matrices from Layer 8, Head 3 of the fine-tuned model for four cases: correct positive, correct negative, incorrect positive, and incorrect negative predictions.

4 Results

4.1 Experiment 3.1 - Probing

We began by extracting sentence-level embeddings from the frozen bert-base-uncased model. Three strategies were explored for deriving fixed-length representations of each news article:

- The [CLS] token from the final hidden layer,
- The mean of all token embeddings (excluding padding), and
- The last token embedding.

These representations were then used as features for two classical classifiers: K-Nearest Neighbors (KNN) and Logistic Regression (LogReg). For KNN, the optimal number of neighbors K was selected based on validation accuracy. All probing experiments were conducted using the full AG News training set (120,000 examples), with performance reported on the official test set.

[Core Findings.] Mean pooling consistently outperformed both [CLS] and last-token strategies, yielding the highest test accuracy across both KNN (91.93%) and LR (91.13%), and in terms of validation accuracy, we see a similar trend (see figure (1)), where the K value also appears to have a big impact, as validation accuracy skyrockets for the first few K values. Naturally, we used a validation set to choose the best K (see table (1)). Logistic Regression generally outperformed KNN across all embedding types, although the best KNN + mean setup closely matched LR's performance. Interestingly, the [CLS] token—often used as a default in many BERT pipelines—underperformed compared to mean pooling, suggesting limited expressivity for sentence-level classification.

Table 1: Probing results using frozen BERT embeddings on AG News.

Embedding Strategy	LogReg Accuracy	LogReg F1	Best K (KNN)	KNN Accuracy
[CLS]	0.9037	0.9036	17	0.8889
Mean	**0.9113**	**0.9113**	**13**	**0.9193**
Last	0.8997	0.8996	7	0.8954

[**Additional Experiments.**] To further analyze probing behavior, we conducted two additional experiments:

1. Effect of max_iter on Logistic Regression (using [CLS] embeddings): Test accuracy improved significantly from max_iter = 10 to 50, after which it plateaued. This suggests convergence can be reached efficiently with relatively low iteration counts. The plot showing this is illustrated in figure (2).
2. Effect of Training Subset Size (using [CLS] embeddings): Both LogReg and KNN achieved rapid gains in low-data regimes, with LogReg exceeding 85% accuracy using only 10K examples. KNN showed a more gradual upward trend, highlighting logistic regression's data efficiency when paired with BERT embeddings. From the results (depicted in figure (3)), it is clear that a larger subset implies better test accuracy, with LogReg even surpassing 90 percent test accuracy for 120,000 samples. This gets even higher when we use Mean embeddings with 120,000 samples, looking back our results in table 1 above.

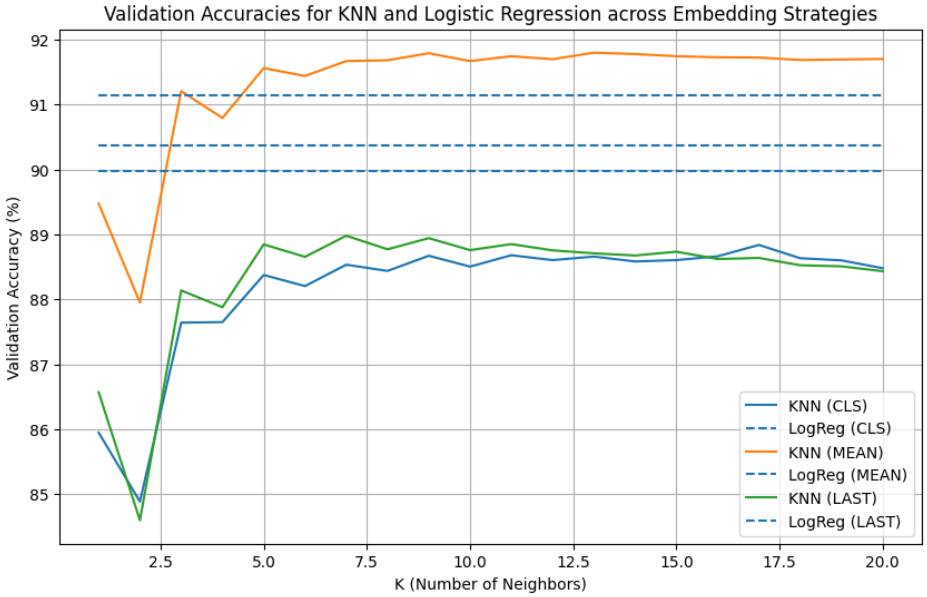

Figure 1: Validation accuracies for KNN (solid lines) and Logistic Regression (dashed lines) across different sentence embedding strategies. Each KNN curve corresponds to $K \in [1, 20]$.

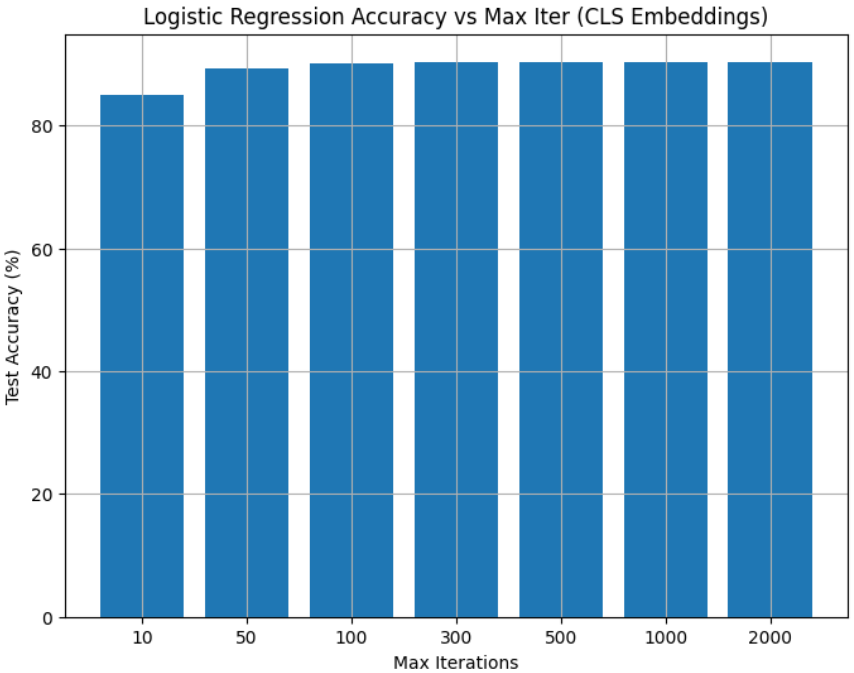

Figure 2: Logistic Regression test accuracy as a function of `max_iter` using [CLS] embeddings.

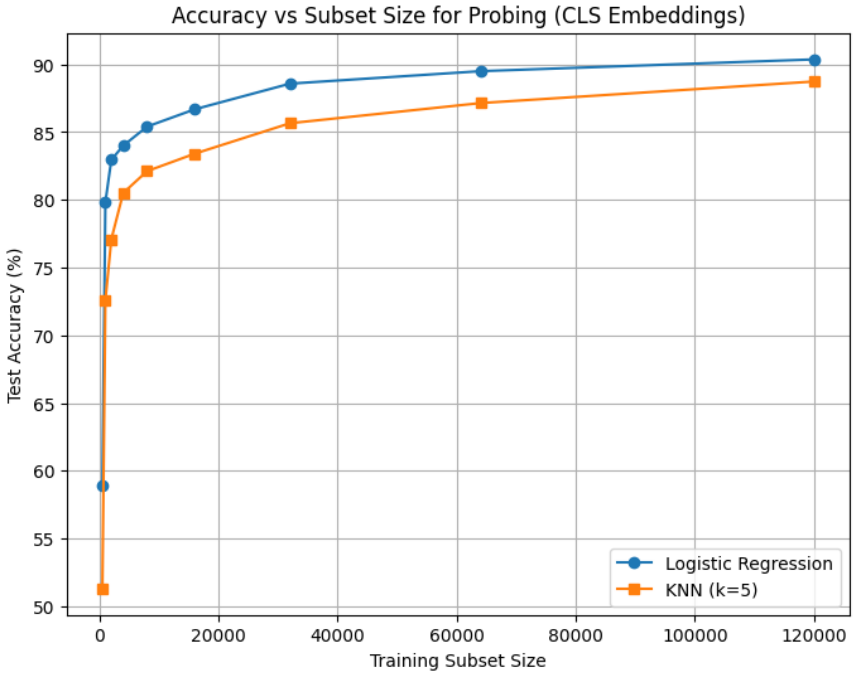

Figure 3: Test accuracy for Logistic Regression and KNN ($k = 5$) with [CLS] embeddings across varying training subset sizes.

These findings suggest that high performance can be achieved via probing alone, particularly when using mean-pooled embeddings. Among all configurations, mean embeddings combined with logistic regression achieved the highest test accuracy of 91.13%, outperforming both [CLS] and last-token strategies across classifiers. This setup provides a robust baseline for text classification without requiring costly end-to-end fine-tuning. However, this performance comes at the expense of computational resources: training on the full dataset (120,000 examples) significantly increases runtime and memory consumption. In our experiments, the use of an A100 GPU on Google Colab was necessary to process the complete training set efficiently.

4.2 Experiment 3.2 - Fine Tuning

To evaluate the performance of end-to-end training, we fine-tuned all parameters of the pre-trained bert-base-uncased model on the AG News classification task. A subset of 60,000 training examples was used for training, with 20% reserved for validation. The model was fine-tuned over 3 epochs using the AdamW optimizer with a learning rate of 2e-5 and a batch size of 16.

[Training Dynamics.] During training, we observed consistent improvements in both training loss and validation accuracy across epochs. The training loss decreased from 0.2194 to 0.0717 over 3 epochs, while validation accuracy peaked at 94.70%, with a corresponding validation F1 score of 94.70%. Final model selection was based on validation accuracy.

Table 2: Training and validation performance across 3 epochs of BERT fine-tuning (trained on 60K subset). Final test accuracy and F1 are also reported.

Epoch	Train Loss	Val Accuracy	Val F1 Score
1	0.2194	0.9434	0.9436
2	0.1233	**0.9470**	**0.9470**
3	0.0717	0.9468	0.9468
Test (final model)	–	**0.9451**	**0.9451**

[Test Performance.] The fine-tuned model achieved a test accuracy of **94.51%** and a test F1 score of **94.51%**, as depicted in table (2) above. This outperforms the best probing configuration (mean embeddings + logistic regression), which achieved 91.13% test accuracy (see Table (1)). The performance gain underscores the benefit of full gradient-based adaptation over frozen feature extraction.

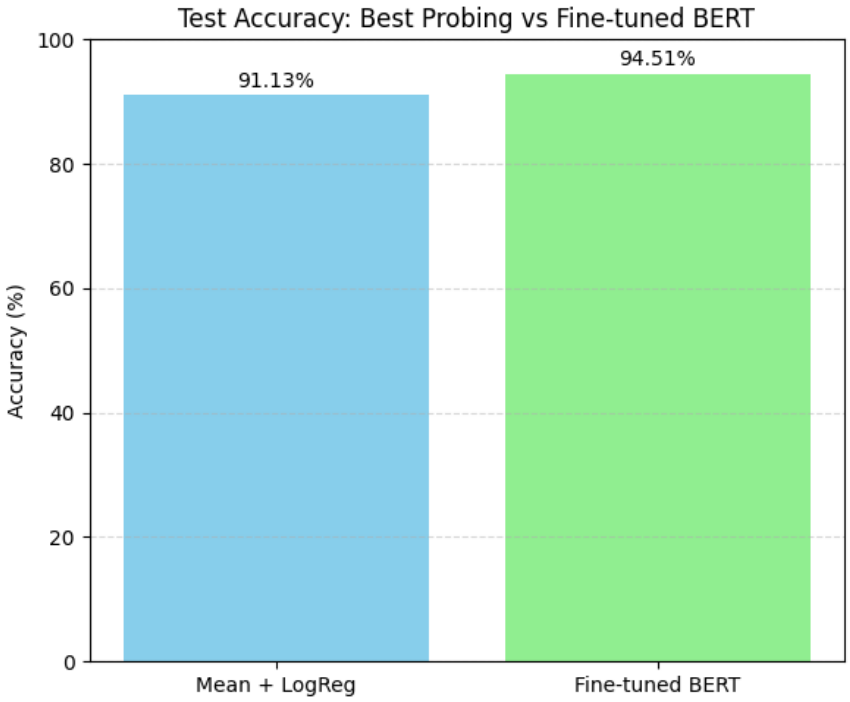

Figure 4: Comparison of test accuracy between the best probing configuration and fine-tuned BERT.

Fine-tuning yields a $\approx 3.4\%$ absolute improvement in accuracy over the best probing setup (see figure (4)). While more computationally expensive, this result demonstrates that BERT's performance can be further unlocked through task-specific parameter updates.

4.3 Experiment 3.3 - Reporting Classification Performances

We compare the classification performance of traditional probing-based methods and end-to-end fine-tuning on the AG News test set. The best probing configuration—mean-pooled embeddings with logistic regression—achieved a test accuracy of 91.13%, whereas fine-tuning the entire BERT model led to a significantly higher test accuracy of 94.51%.

This improvement of approximately 3.38 percentage points highlights the power of task-specific adaptation. While probing utilizes fixed representations extracted from the frozen BERT encoder, fine-tuning updates all layers of the model to directly optimize classification performance. This allows BERT to better align its internal features with the AG News domain, resulting in stronger performance.

However, this gain comes at a computational cost. Fine-tuning required significantly more GPU time and memory—our experiments were run on an A100 GPU to efficiently train on 60,000 examples. In contrast, probing methods are much more lightweight and can be executed quickly with minimal hardware. Thus, probing remains a viable alternative when compute resources are limited, especially given its strong baseline performance.

4.4 Experiment 3.4 - Attention Matrix Visualization

To further investigate the behavior of the fine-tuned BERT model, we used the `bertviz` library [**vig2019bertviz**] to visualize the attention patterns between the `[CLS]` token and input tokens. We focused on attention head 3 from layer 8, and analyzed four types of test examples:

- Correctly predicted positive example
- Correctly predicted negative example
- Incorrectly predicted positive example
- Incorrectly predicted negative example

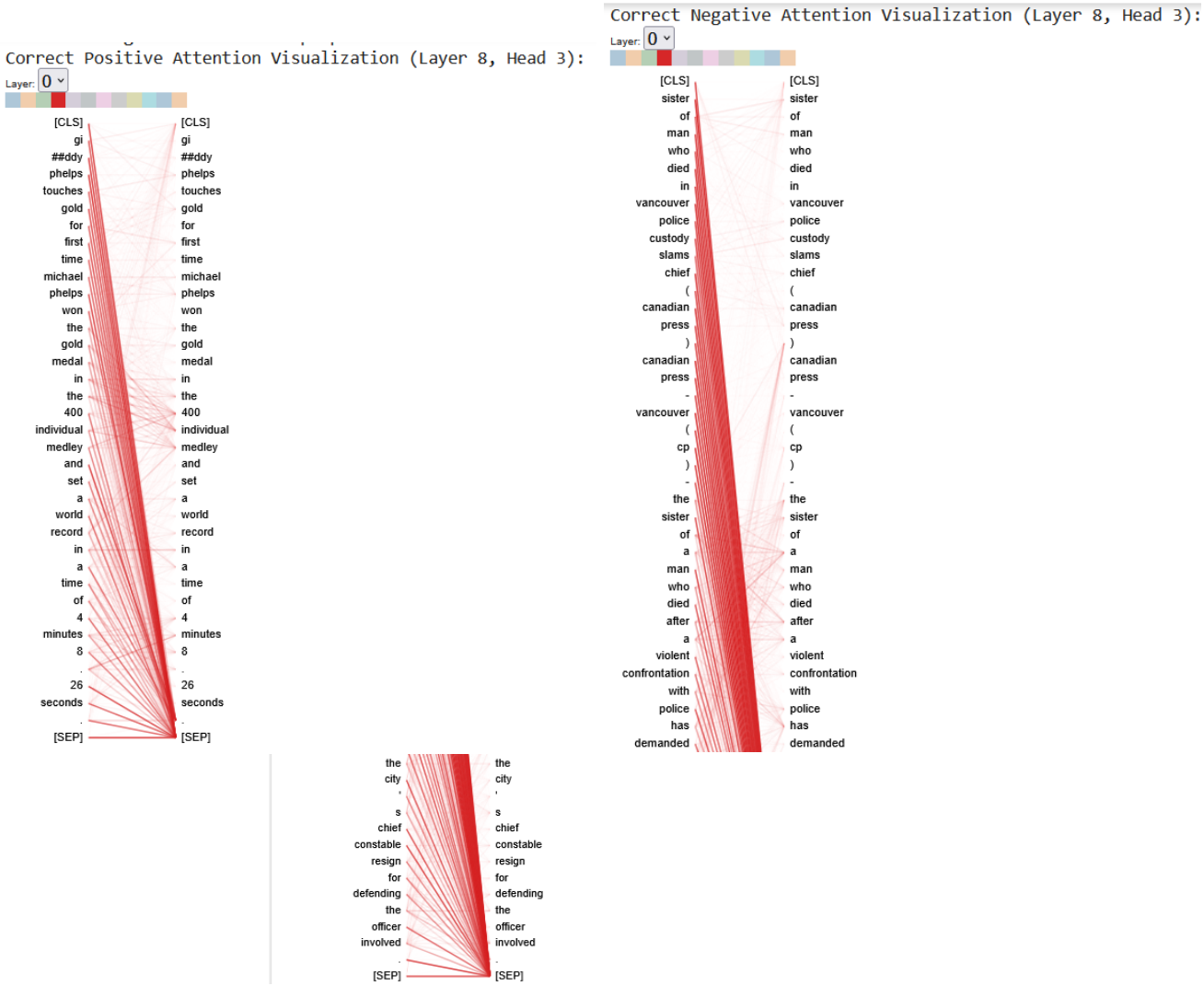

Figure 5: Attention patterns for correctly predicted examples. Top: Positive (left) and Negative (right). Bottom: A continuation of the Negative example (it was too large to fit in one output cell)

As shown in Figure 5, the attention for correct predictions tends to focus on highly informative tokens. In the positive case (a sports-related headline), the model attends strongly to tokens like “phelps”, “gold”, and “medal” — clear indicators of a sports article. Similarly, in the negative example, the model captures key contextual phrases like “died in vancouver police custody”, which are likely representative of a world news topic.

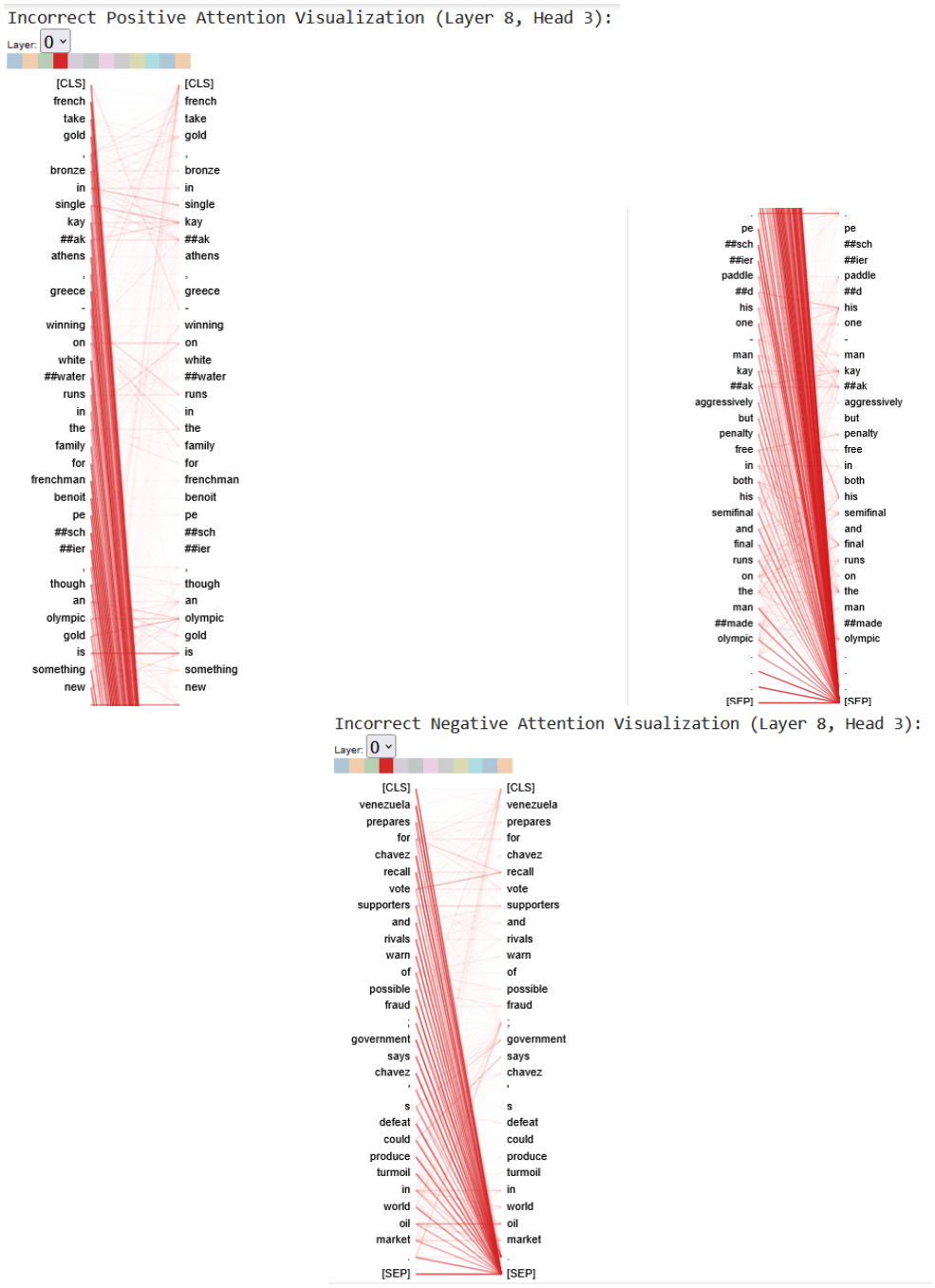

Figure 6: Attention patterns for incorrectly predicted examples. Top: positive (Olympic kayaking event misclassified). Bottom: negative (Venezuela recall vote article misclassified).

In contrast, the incorrect examples in Figure 6 display more diffuse or misaligned attention. The misclassified positive example contains relevant terms like "olympic" and "gold", but also includes many fragmented or ambiguous tokens (e.g., subword pieces like "##sch", "##ier"), which may have disrupted the model's decision boundary. The incorrect negative prediction exhibits strong attention to political terms such as "chavez" and "government", yet may have lacked sufficient disambiguation from other world news articles, leading to a misclassification.

We can see that these visualizations highlight the interpretability benefits of attention-based architectures and offer qualitative evidence that attention can serve as a useful diagnostic tool for understanding BERT's decision-making process.

5 Discussion & Conclusion

Our study provides a comprehensive comparison between probing and end-to-end fine-tuning of BERT for news topic classification. We found that mean-pooled frozen embeddings combined with a simple logistic regression classifier achieve competitive

accuracy of 91.13%, confirming that much of BERT's linguistic knowledge is accessible even without updating its parameters. However, fine-tuning the entire model leads to a notable performance boost, with a final test accuracy of 94.51%, indicating the added benefit of task-specific adaptation.

Qualitative insights from attention visualizations revealed that correctly classified examples exhibited more focused attention on key tokens, while misclassifications were associated with diffuse or misaligned attention patterns. This suggests that attention heads may serve as useful indicators for model confidence and decision quality.

While fine-tuning offers higher accuracy, it also requires significantly more computational resources and training time. In contrast, probing methods are lightweight, faster to execute, and more suitable for constrained environments. These trade-offs highlight the practical relevance of probing methods in low-resource settings or during early prototyping.

Future work could explore hybrid strategies, such as partially fine-tuning select layers or leveraging adapters, to balance performance and efficiency. Additionally, further research into attention alignment and interpretability may yield better insights for trust and transparency in transformer-based models.

6 References

- Devlin, Jacob, et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." arXiv preprint arXiv:1810.04805 (2018).
- Rogers, Anna, et al. "A Primer in BERTology: What we know about how BERT works." Transactions of the Association for Computational Linguistics 8 (2020): 842–866.
- Hewitt, John, and Christopher D. Manning. "A Structural Probe for Finding Syntax in Word Representations." Proceedings of NAACL (2019).
- Jawahar, Ganesh, et al. "What does BERT learn about the structure of language?." Proceedings of ACL (2019).
- Peters, Matthew E., et al. "To Tune or Not to Tune? Adapting Pretrained Representations to Diverse Tasks." arXiv preprint arXiv:1903.05987 (2019).
- Zhang, Xiang, et al. "Character-level Convolutional Networks for Text Classification." Advances in Neural Information Processing Systems 28 (2015).
- Vig, Jesse. "A Multiscale Visualization of Attention in the Transformer Model." arXiv preprint arXiv:1906.05714 (2019).