# Fine-Tuning BERT for Sentiment Analysis of Movie Reviews

**Note on Model Execution:**
Due to computational constraints, model training may take several minutes to complete. The results reported in the accompanying report were obtained from a completed training run using the same code, dataset, and configuration presented in this notebook.

# 1. Introduction

Sentiment analysis is a common problem in natural language processing (NLP) that involves identifying whether a piece of text expresses a positive or negative opinion. It is widely used in practice, for example in analysing customer feedback, product reviews, and social media posts. Earlier approaches to sentiment analysis typically relied on traditional machine learning methods, using manually designed features such as bag-of-words or TF–IDF. While these methods can work reasonably well, they often struggle to capture context and deeper meaning in text.

More recently, Large Language Models (LLMs) have led to major improvements in NLP tasks. One of the most influential models is **BERT** (Bidirectional Encoder Representations from Transformers). Unlike earlier models, BERT processes text in both directions at the same time, allowing it to learn contextual representations of words. This makes it particularly suitable for sentiment analysis, where meaning often depends on surrounding words rather than individual terms.

The aim of this project is to fine-tune a pre-trained BERT model for binary sentiment classification using movie reviews from the IMDb dataset. To better understand the benefits of using an LLM, the performance of the fine-tuned BERT model is compared with a simpler baseline model based on TF–IDF features and Logistic Regression (Geron, 2019).

# 2. Dataset

The **IMDb movie review dataset** was used for this project. This dataset is a standard benchmark for sentiment analysis and contains a large number of movie reviews labelled as either positive or negative.

The dataset consists of 50,000 reviews in total, evenly split into 25,000 reviews for training and 25,000 reviews for testing. Each review is assigned a binary sentiment label, where 0 represents negative sentiment and 1 represents positive sentiment.

The reviews are relatively long and often include detailed opinions, which makes the dataset suitable for evaluating models that can capture long-range dependencies in text. The dataset was accessed using the Hugging Face datasets library, which also helps ensure that the experiment can be reproduced (Geron, 2019).

# 3. Methodology

### 3.1 Baseline Model

To provide a reference point, a traditional machine learning baseline was implemented. This approach uses **TF–IDF** to convert text into numerical feature vectors, followed by a **Logistic Regression** classifier. TF–IDF captures how important a word is within a document relative to the whole dataset but does not take word order or context into account (Géron, 2019).

The baseline model was trained on the training set and evaluated on the test set. Although simple, this method is widely used and provides a useful comparison when assessing more complex models such as BERT.

### 3.2 Preprocessing and Tokenization

For the BERT-based approach, the **bert-base-uncased** tokenizer was used. Each review was tokenized and then padded or truncated to a maximum length of 256 tokens. This ensures that all input sequences have the same length and can be processed efficiently by the model.

The IMDb dataset stores sentiment labels under the key `label`. During training and evaluation, this was mapped to the `labels` argument expected by the **BertForSequenceClassification** model.

### 3.3 Model Architecture and Training

The model used for fine-tuning was **BertForSequenceClassification**, which adds a simple classification layer on top of the BERT encoder. The model outputs predictions for the two sentiment classes.

The training setup was kept relatively simple:

- Optimizer: AdamW
- Learning rate: $2 \times 10^{-5}$
- Batch size: 16
- Number of epochs: 2

The loss function used was cross-entropy loss, which is handled internally by the model. Fine-tuning updates all model parameters so that the pre-trained representations can adapt to the sentiment analysis task.

### 3.4 Evaluation Metrics

Both the baseline and BERT models were evaluated using accuracy, precision, recall, and F1-score. These metrics provide a more complete picture of performance than accuracy alone, particularly when evaluating classification models.

# 4. Results and Discussion

The results show that the fine-tuned BERT model performs better than the baseline Logistic Regression model.

**Table 1: Comparison of Model Performance**

| Model | Accuracy | F1-score |
|---|---|---|
| Logistic Regression (TF–IDF) | ~0.88 | ~0.88 |
| Fine-tuned BERT | ~0.92–0.94 | ~0.92–0.94 |

The baseline model achieves reasonably strong performance, which suggests that simple text representations can still be effective for sentiment analysis. However, the BERT model consistently achieves higher accuracy and F1-score. This improvement is likely due to BERT's ability to take context into account and model complex sentence structures, such as negation or mixed opinions within a review.

That said, there are some limitations. Fine-tuning BERT is computationally more expensive than training a traditional model, and training was limited to only two epochs due to resource constraints. In addition, only a small number of hyperparameters were explored.

# 5. Conclusion and Future Work

This project explored the use of a fine-tuned BERT model for sentiment analysis on movie reviews. The results clearly show that BERT outperforms a traditional TF–IDF and Logistic Regression baseline, demonstrating the advantages of using transformer-based models for text classification tasks.

Future work could involve experimenting with different transformer models, such as **RoBERTa** or **DistilBERT**, as well as performing more extensive hyperparameter tuning. Applying the same approach to other datasets or sentiment-related tasks could also provide further insight. Overall, this project highlights the effectiveness of Large Language Models for sentiment analysis while also illustrating some of the practical trade-offs involved..

# References


- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). *BERT: Pre-training of deep bidirectional transformers for language understanding*. arXiv:1810.04805. https://arxiv.org/abs/1810.04805
- Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics. https://aclanthology.org/P11-1015.pdf
- Geron, A. (2019) Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. 2nd Edition, O'Reilly Media, Inc., Sebastopol. https://www.scirp.org/reference/referencespapers?referenceid=3265407


**Appendix**

**Colab link:** https://colab.research.google.com/drive/1Vg9zsf30HTzl6HUS-TdBsreaK9stFG9Z#scrollTo=08zyJXTMUV9V

**Github:** https://github.com/daminimukhi/Research-Methods-In-Data-Science