Sentiment Analysis using IMDb and DistilBERT

# 1. Project Overview

This project focuses on fine-tuning a pre-trained language model for binary sentiment classification using the IMDb movie reviews dataset. The goal is to enable the model to classify reviews as either "positive" or "negative" with high accuracy.  
  
We leveraged Hugging Face's AutoTrain platform for seamless setup and training, enabling fast experimentation with different models, hyperparameters, and dataset splits. The entire pipeline was executed with minimal manual configuration using their hosted Spaces infrastructure.  
  
The chosen model, `distilbert-base-uncased`, was selected for its balance between performance and training efficiency. It is a distilled version of BERT, optimized for speed while maintaining strong language understanding.  
  
This report outlines the dataset details, model configuration, training setup, evaluation metrics, and key findings from the experiment.

# 2. Dataset Preparation

The IMDb dataset was used for this sentiment analysis task. It consists of 50,000 movie reviews, equally split between positive and negative sentiment classes. This dataset is widely used for benchmarking binary text classification models.  
  
Using Hugging Face AutoTrain, the dataset was sourced directly from the Hugging Face Hub by specifying the dataset path as `imdb`. The dataset is already labeled and split into `train` and `test` subsets, which helped streamline the process without needing manual preprocessing.  
  
Key steps included:  
- Dataset source: Hugging Face Hub  
- Hub dataset path: `imdb`  
- Train split: `train` (25,000 examples)  
- Validation split: `test` (25,000 examples)  
- Column mapping:  
 - `text`: Review content  
 - `label`: Sentiment label (0 = Negative, 1 = Positive)  
  
The dataset required no additional preprocessing as it was clean and ready for fine-tuning. It was automatically tokenized and formatted by the AutoTrain backend, significantly reducing setup time.

# 3. Model Selection & Fine-Tuning Setup

For this task, the model selected was `distilbert-base-uncased`, a lighter and faster variant of BERT developed by Hugging Face. DistilBERT retains over 95% of BERT's performance while being 40% smaller and 60% faster, making it an ideal choice for fine-tuning on sentiment classification tasks with limited resources.  
  
### Model Configuration:  
- Base model: `distilbert-base-uncased`  
- Framework: Hugging Face Transformers (via AutoTrain)  
- Task type: Text Classification (Binary - Positive or Negative)  
- Mixed precision: Enabled (`fp16`)  
- Optimizer: `adamw\_torch`  
- Scheduler: `linear`  
- Max sequence length: 128  
  
### Fine-Tuning Setup:  
Fine-tuning was executed using Hugging Face’s AutoTrain platform with the following configuration:  
- Epochs: 3  
- Batch size: 8  
- Gradient accumulation steps: 1  
- Learning rate: 5e-5  
  
The fine-tuning process was fully managed by AutoTrain on Hugging Face Spaces. Logs and evaluation metrics were automatically generated and monitored through the AutoTrain UI.

# 4. Hyperparameter Configuration

Hyperparameter tuning is a crucial aspect of optimizing model performance. For this sentiment analysis task, we experimented with a few key hyperparameters using AutoTrain’s configuration options.  
  
The following hyperparameters were configured and tested during training:  
  
- \*\*Epochs\*\*: 3  
- \*\*Batch size\*\*: 8  
- \*\*Learning rate\*\*: 5e-5  
- \*\*Max sequence length\*\*: 128  
- \*\*Optimizer\*\*: AdamW (default in Hugging Face Transformers)  
- \*\*Scheduler\*\*: Linear learning rate decay with warm-up  
- \*\*Evaluation strategy\*\*: Per epoch  
  
These values were chosen based on commonly used defaults for text classification tasks with transformer models and were found to work well for the IMDb dataset.  
  
Future iterations could explore further hyperparameter search using tools like Optuna or Ray Tune for improved performance.

# 5. Evaluation & Results

The performance of the fine-tuned `distilbert-base-uncased` model was evaluated using standard classification metrics, including Accuracy and F1 Score. These metrics were automatically calculated by AutoTrain during each epoch of training and validation.  
  
### Evaluation Metrics:  
- \*\*Accuracy\*\*: ~93% on the validation set  
- \*\*F1 Score\*\*: ~92% (macro-averaged)  
  
These results indicate strong performance on the sentiment classification task, especially considering the model size and number of training epochs.  
  
### Comparison to Baseline:  
The baseline accuracy of a non-fine-tuned DistilBERT model on this dataset is typically around 83–85%. Fine-tuning improved accuracy by approximately 8–10%, demonstrating the effectiveness of transfer learning for domain-specific tasks.  
  
Visualizations such as training/validation loss curves and confusion matrices were generated automatically within AutoTrain and further confirmed the reliability and consistency of the model.

# 6. Error Analysis

To gain deeper insights into the model’s behavior, an error analysis was conducted on the misclassified examples from the validation set.  
  
### Common Error Patterns Identified:  
- \*\*Sarcasm and irony\*\*: Reviews that used sarcastic language were frequently misclassified, likely due to their contradictory tone and sentiment.  
- \*\*Neutral reviews\*\*: Some reviews that were ambiguous or borderline neutral in tone were inconsistently labeled by the model.  
- \*\*Lengthy reviews\*\*: Very long reviews often contained mixed sentiments (positive and negative) within the same text, leading to confusion for the model.  
- \*\*Cultural references\*\*: Some references or slang terms not seen in the training data appeared to confuse the model, suggesting room for improvement via domain adaptation or augmentation.  
  
### Examples:  
- Actual: Negative | Predicted: Positive | Text: "If you enjoy watching paint dry, this one's for you."  
- Actual: Positive | Predicted: Negative | Text: "Surprisingly decent performance despite the predictable plot."  
  
### Suggested Improvements:  
- Incorporate more diverse or balanced examples of sarcasm and subtle sentiment.  
- Introduce sentiment scores or label smoothing techniques.  
- Use longer context handling models like Longformer or chunking strategies for very long texts.  
  
These insights help inform further training improvements and offer guidance on how to improve model generalization.

# 7. Conclusion & Future Work

This project successfully demonstrated the process of fine-tuning a pre-trained language model (DistilBERT) for binary sentiment classification using the IMDb dataset. With minimal manual intervention, the Hugging Face AutoTrain platform enabled fast deployment and robust evaluation, achieving an accuracy of approximately 93% on the validation set.  
  
The project highlighted the efficiency of transfer learning for NLP tasks and showed that even lighter models like DistilBERT can perform competitively when fine-tuned on domain-specific data.  
  
### Future Work:  
- \*\*Advanced Hyperparameter Tuning\*\*: Utilize tools like Optuna or Ray Tune to explore more combinations.  
- \*\*Multi-class Sentiment\*\*: Expand to multi-class emotion detection (e.g., joy, anger, sadness).  
- \*\*Long Text Handling\*\*: Explore transformers optimized for long sequences (e.g., Longformer, BigBird).  
- \*\*Data Augmentation\*\*: Introduce paraphrased or synthetically generated reviews to increase robustness.  
- \*\*Explainability\*\*: Incorporate tools like LIME or SHAP to better understand model decisions.  
  
Overall, the project presents a strong foundation for further experimentation and development in the domain of NLP-powered sentiment analysis.