Assignment 3: End-to-End HuggingFace Model Training & Docker Deployment

Zenith (M25CSA032)
Department of Computer Science
MLOps Course

February 22, 2026

Abstract

This report presents an end-to-end MLOps pipeline for fine-tuning `distilbert-base-cased` on the UCSD Goodreads dataset for 8-genre book review classification. The model achieves 55.4% accuracy (weighted F1: 0.547), a ∼4.4× improvement over the random baseline of 12.5%. The pipeline includes training, evaluation with visualizations, HuggingFace Hub deployment, and Docker containerization. Local and Hub evaluations produce identical results, validating deployment integrity.

Submission Links:

- **HuggingFace Model:** `https://huggingface.co/Zenith754/goodreads-bert-classifier`
- **GitHub Repository:** `https://github.com/zzethh/MLOps-Zenith-M25CSA032`

1 Introduction

In this project, we implement a complete MLOps pipeline: (1) stream book reviews from the UCSD Goodreads dataset (8 genres, 8K samples), (2) fine-tune a DistilBERT model for genre classification, (3) evaluate with metrics and visualizations, (4) push the trained model to HuggingFace Hub, (5) re-evaluate from Hub to verify deployment integrity, and (6) containerize with Docker. Target genres: Children, Comics & Graphic, Fantasy & Paranormal, History & Biography, Mystery/Thriller/Crime, Poetry, Romance, and Young Adult.

2 Model Selection

We chose `distilbert-base-cased` for the following reasons:

- **Knowledge Distillation:** DistilBERT retains 97% of BERT's capabilities while being 60% faster and 40% smaller (66M vs 110M parameters) [1].
- **Cased Variant:** The cased version preserves capitalization, which is significant for literary critique where proper nouns (character names, book titles) carry meaning.
- **Compatibility:** Natively supports `AutoModelForSequenceClassification` for our 8-class task.
- **Efficiency:** Fine-tunes on a single GPU in under 4 minutes.

3 Training Summary

Dataset: UCSD Goodreads Book Graph [2] — 10K reviews streamed per genre, 1K sampled, split 800/200 per genre = 6,400 train + 1,600 test.

Table 1: Training Configuration & Results

Parameter	**Value**	**Result**	**Value**
Epochs	3	Final Eval Loss	1.324
Train Batch Size	16	Final Accuracy	55.4%
Eval Batch Size	16	Weighted F1	0.547
Learning Rate	2×10^{-5}	Train Runtime	199.1s
Weight Decay	0.01	Throughput	96.5 samples/s
Max Length	512	Total Steps	150

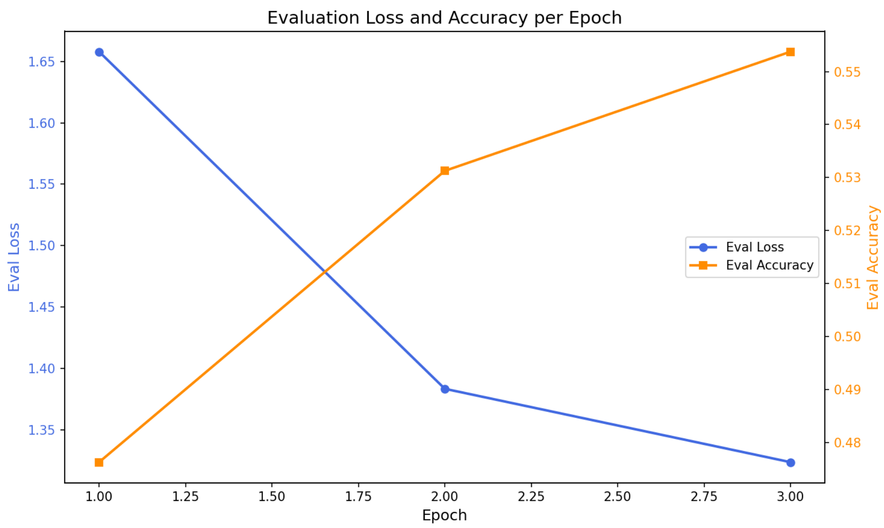


Figure 1: Evaluation loss and accuracy per epoch. Loss steadily decreases while accuracy improves from 47.6% to 55.4% over 3 epochs.

4 Evaluation Results

4.1 Overall Metrics — Local vs. Hub

After pushing the trained model to HuggingFace Hub (`Zenith754/goodreads-bert-classifier`), we re-evaluated by pulling it from the Hub. Table 2 confirms identical metrics, validating deployment integrity.

Table 2: Local vs. Hub Model Evaluation (zero difference confirms correct deployment)

Metric	**Local**	**Hub**	**Diff**
Accuracy	0.5538	0.5538	0.0000
Precision (weighted)	0.5455	0.5455	0.0000
Recall (weighted)	0.5538	0.5538	0.0000
F1-Score (weighted)	0.5471	0.5471	0.0000
Loss	1.3236	1.3236	0.0000

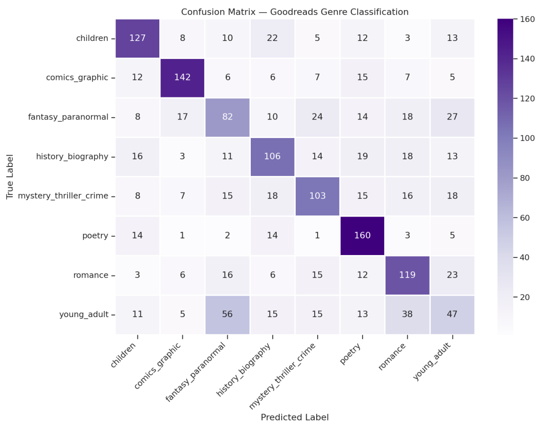
4.2 Per-Class Performance

Table 3: Per-Class Classification Report

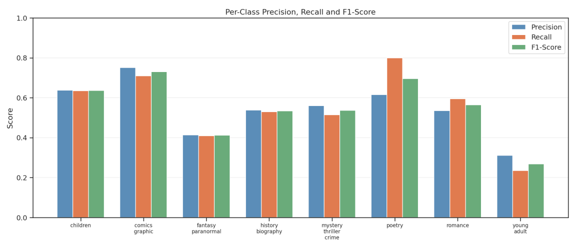
Genre	Precision	Recall	F1	N
Children	0.64	0.64	0.64	200
Comics & Graphic	0.75	0.71	0.73	200
Fantasy & Paranormal	0.41	0.41	0.41	200
History & Biography	0.54	0.53	0.53	200
Mystery/Thriller/Crime	0.56	0.52	0.54	200
Poetry	0.62	0.80	0.70	200
Romance	0.54	0.59	0.56	200
Young Adult	0.31	0.23	0.27	200
Macro Avg	0.55	0.55	0.55	1600

Comics & Graphic (F1: 0.73) and Poetry (F1: 0.70) perform best due to distinctive vocabulary. Young Adult (0.27) and Fantasy & Paranormal (0.41) are the weakest due to vocabulary overlap with other fiction genres.

4.3 Visualizations

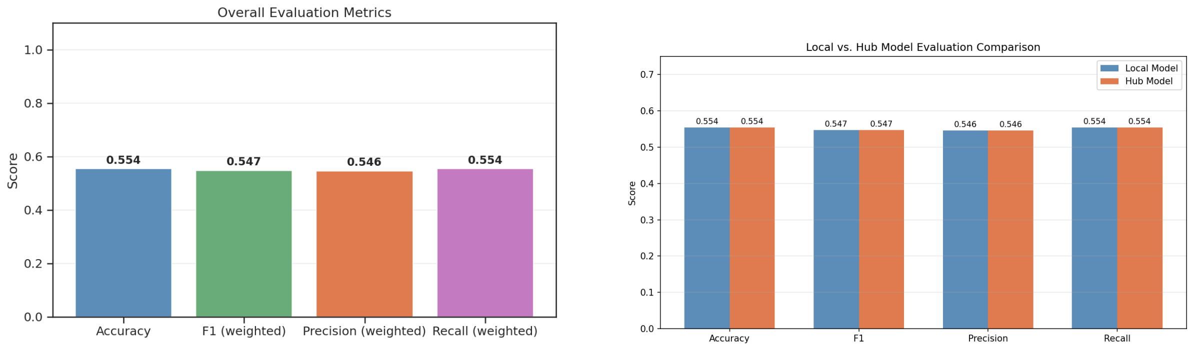


(a) Confusion Matrix



(b) Per-Class Precision, Recall, F1

Figure 2: Classification analysis visualizations.



(a) Overall Evaluation Metrics

(b) Local vs. Hub Comparison

Figure 3: Overall metrics and deployment verification.

5 Docker Deployment

Development Image (`Dockerfile`): Based on `python:3.9-slim-bullseye`. Bundles the trained model locally for standalone evaluation.

Production Image (`Dockerfile.eval`): Downloads model from HuggingFace Hub on startup. Configured via `HF_TOKEN` and `HF_REPO` environment variables.

```
# Development (local model)
docker build -t mlops-assignment .
docker run --rm -v $(pwd)/results:/app/results mlops-assignment

# Production (from HuggingFace Hub)
docker build -f Dockerfile.eval -t mlops-eval .
docker run --rm -e HF_TOKEN=<token> \
    -e HF_REPO=Zenith754/goodreads-bert-classifier \
    -v $(pwd)/results:/app/results mlops-eval
```

6 Challenges

1. **GPU Access in Docker:** Missing NVIDIA Docker toolkit caused silent fallback to CPU. Resolved by training natively with GPU and constraining Docker to evaluation-only.

2. **Multiclass Metric Averaging:** Default `evaluate` package uses binary averaging. Required updating to weighted averaging for the 8-genre task.

3. **Dataset Streaming:** Downloading large gzip files live requires careful memory management. Used iterative streaming with configurable limits.

4. **Genre Overlap:** Young Adult and Fantasy share vocabulary with other fiction genres, limiting per-class performance.

5. **Reproducibility:** Fixed random seed (42) across data sampling to ensure consistent train/test splits.

7 Conclusion

We successfully implemented an end-to-end MLOps pipeline achieving 55.4% accuracy on 8-genre classification ($\sim$4.4$\times$ over random baseline). Identical local/hub results validate deployment integrity. The project is containerized with Docker and publicly available on HuggingFace Hub.

References

[1] V. Sanh, L. Debut, J. Chaumond, T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *arXiv:1910.01108*, 2019.

[2] M. Wan, J. McAuley, "Item recommendation on monotonic behavior chains," *Proc. ACM RecSys*, 2018.