INFO 7375 - Prompt Engineering for Generative AI

Assignment 3: Fine-Tuning a Large Language Model - by Ranjithnath Karunanidhi (002317425)

Technical Report

ABSTRACT

This project implements automated medical specialty classification across 40 specialties using fine-tuned DistilBERT with LoRA (Low-Rank Adaptation). The system addresses the healthcare challenge of routing clinical transcription notes to appropriate specialists. Using parameter-efficient fine-tuning, we achieved 56.62% test accuracy with 98.65% parameter reduction (from 67.9M to 916K trainable parameters).
Here are the key findings,

- LoRA requires 10-15x higher learning rates than standard fine-tuning
- Text length is the primary error factor
- Class imbalance significantly impacts rare specialty performance.

The model demonstrates production feasibility with 29ms inference latency and 57 documents/second throughput.

INTRODUCTION

Problem Statement

Healthcare institutions process thousands of unstructured clinical transcriptions daily that require routing to appropriate medical specialists. Current manual classification processes take 48 to 72 hours of delays for administrative review with 15-20% misrouting rates leading to patient safety risks. Significant administrative costs also go up to $10-15 per document.
This project develops an automated classification system using fine-tuned large language models to address these operational inefficiencies.

Objectives

1. Fine-tune a pre-trained transformer model for 40-class medical specialty classification
2. Implement parameter-efficient fine-tuning techniques (LoRA/PEFT)
3. Achieve competitive accuracy while handling severe class imbalance (238:1 ratio)
4. Conduct comprehensive evaluation against multiple baselines
5. Identify performance limitations and propose concrete improvements

Significance

This work demonstrates practical application of advanced fine-tuning techniques to real-world healthcare challenges, focusing to build deployable solutions on production that balance accuracy with computational efficiency.

METHODOLOGY AND APPROACH

Dataset Preparation

Dataset: galileo-ai/medical_transcription_40
Source: Medical transcription notes from HuggingFace Hub
Classes: 40 medical specialties
Size: 4,999 total samples (4,499 train, 500 test)

Preprocessing:

- Remove duplicates
- Text cleaning (lowercase, remove special characters, whitespace normalization)
- Minimum text length filtering (20 characters)

Final dataset: 4,457 samples (3,966 train, 491 test)

Data Splitting:

Using *train_test_split* from Scikit-learn,

- Training: 75.6% (3,371 samples)
- Validation: 13.3% (595 samples)
- Test: 11.0% (491 samples)
- Method: Stratified sampling to maintain class distribution

Tokenization:

Tokenizer: DistilBERT WordPiece
MAX_LENGTH: 512 tokens
Coverage: 37.1% of samples fit completely, 62.9% truncated
Rationale: Standard BERT limit, specialty indicators appear in opening sections

Model Selection

Selected Model: DistilBERT (*distilbert/distilbert-base-uncased*)

Reasons to choose the model:

Architectural fit: Native sequence classification support with 768-dimensional hidden state
Computational efficiency: 66M parameters (60% faster than BERT-base)
Task suitability: Proven performance on multi-class classification tasks
Production feasibility: Low inference latency suitable for real-time routing

Alternatives Considered:

BERT-large: Rejected due to 3-4x slower training limiting hyperparameter search
BioBERT: Medical pre-training advantage outweighed by reduced iteration speed
GPT-2: Designed for generation, inefficient for classification tasks

Parameter-Efficient Fine-Tuning (LoRA)

Technique: Low-Rank Adaptation (LoRA)

Configuration:

- Task type: Sequence classification (SEQ_CLS)
- LoRA rank (r): 16
- Alpha: 32
- Target modules: q_lin, v_lin (attention projection layers)
- Dropout: 0.1

Results:

- Trainable parameters: 916,264 (1.35% of total)
- Parameter reduction: 98.65%

- Training speedup: ~10x faster than full fine-tuning

Rationale: LoRA enables rapid experimentation while maintaining competitive performance, critical for production deployment where model updates must be fast and storage efficient.

Training Configuration

Hyperparameter Search Strategy:

- Method: Manual grid search
- Parameters tuned: Learning rate, batch size, epochs, weight decay
- Evaluation metric: F1-macro (handles class imbalance)
- Key insight: LoRA requires 10-15x higher learning rates than standard fine-tuning

1. **Config 1 (Baseline LoRA):**
   - Learning rate: 1e-4
   - Batch size: 16
   - Epochs: 5
   - Weight decay: 0.01
   - Results: 56.13% val acc, 0.2368 F1-macro
2. **Config 2 (Higher LR): (BEST)**
   - Learning rate: 3e-4
   - Batch size: 16
   - Epochs: 5
   - Weight decay: 0.01
   - Results: 65.88% val acc, 0.537 F1-macro
3. **Config 3 (Conservative):**
   - Learning rate: 5e-5
   - Batch size: 32
   - Epochs: 5
   - Weight decay: 0.1
   - Results: 37.98% val acc, 0.057 F1-macro

Findings: Learning rate of 3e-4 optimal for LoRA, representing 15x increase over standard BERT fine-tuning (2e-5)

Training Details:

- Framework: HuggingFace Transformers Trainer API
- Callbacks: EarlyStopping (patience=2-3), ModelCheckpoint
- Optimization: AdamW with warmup (100 steps)
- Mixed precision: FP16
- Training time: 3.4 minutes per configuration

RESULTS AND ANALYSIS

Overall Performance

Test Set Performance (Best Model - Config 2):

- Accuracy: 56.62%
- F1-macro: 0.4331
- F1-weighted: 0.5559
- Precision: 0.5695
- Recall: 0.5662

Classes Evaluated: 37 out of 40 (excluding zero test samples)

Comparison with Baselines

Multiple Baseline Comparison:

Approach	Accuracy	F1-macro	Description
Random	2.5%	0.025	Random guessing baseline
Rule-based	34.62%	0.15	Keyword matching approach
Pre-trained	1.22%	0.0007	DistilBERT without fine-tuning
Fine-tuned (LoRA)	**56.62%**	**0.4331**	**Best approach with LoRA**

Improvement Analysis:

- vs Random: 22.6x improvement (demonstrates learning)
- vs Rule-based: 1.64x improvement (ML beats hand-crafted rules)
- vs Pre-trained: 46x improvement (confirms need for domain-specific fine-tuning)

Significance: Achieved competitive performance on challenging 40-class problem while using only 1.35% trainable parameters.

Per-Class Performance

Top Performing Specialties:

- Orthopedic: 79.22% confidence on test case
- Cardiovascular/Pulmonary: 64.94% confidence on cardiac case
- Surgery: Strong performance across surgical procedures

Challenging Specialties:

- Rare classes (<10 test samples): high error rates
- Classes with severe imbalance show degraded performance

Analysis: Performance strongly correlates with class frequency (correlation: -0.325 b/w training samples and error rate).

LoRA Efficiency Analysis

Parameter Efficiency:

- Standard fine-tuning: 67,900,496 trainable parameters
- LoRA fine-tuning: 916,264 trainable parameters
- **Reduction: 98.65%**

Training Efficiency:

- Config 2 training time: 198.1 seconds (~3.3 minutes)
- Estimated full fine-tuning: ~35-40 minutes
- **Speedup: ~10x faster**

Inference Efficiency:

- Single prediction: 29ms latency
- Batch processing: 57 documents/second
- GPU memory: ~300 MB

Production Impact: Small parameter footprint enables rapid model updates and easy deployment.

Hyperparameter Analysis

Key Finding: Learning Rate Sensitivity in LoRA

Standard fine-tuning typically uses LR=2e-5. Our experiments revealed:

- LR=5e-5: 37.98% accuracy (too low for LoRA)
- LR=1e-4: 56.13% accuracy (moderate)
- **LR=3e-4: 65.88% accuracy (optimal)** - *15x higher than standard*

Insight: LoRA's small parameter set (1.35% of model) requires stronger learning signal. This represents a novel finding for LoRA hyperparameter optimization in medical text classification.

ERROR ANALYSIS

Error Patterns Identified

Pattern 1: Text Length Impact

- Misclassified samples: Mean length 4,062 characters
- Correct predictions: Mean length 2,371 characters
- Finding: Longer texts (>512 tokens) have 71% higher error rates
- Cause: Truncation at 512 tokens loses diagnostic information in document tails

Pattern 2: Clinically Similar Specialty Confusion

- Top confusion: Orthopedic to Surgery (14 errors)
- Cardiovascular to Surgery (11 errors)
- Insight: Confusions are clinically meaningful (orthopedic procedures ARE surgical)
- Cause: Model struggles with fine-grained distinctions within medical domains

Pattern 3: Class Imbalance Impact

- Correlation: -0.325 between training samples and error rate
- Classes with <10 samples: 100% error rate
- Classes with >50 samples: 25-35% error rate
- Cause: Insufficient examples for rare specialties

Pattern 4: Administrative vs Clinical Confusion

- SOAP/Chart notes confusing with clinical specialties
- Cause: Administrative document types appear across multiple specialties

Specific Error Examples

Example 1: Cardiovascular to Consult (Text: 3,213 chars) - Long document, diagnostic info likely truncated
Example 2: Urology to Surgery (Text: 2,650 chars) - Contains "preoperative diagnosis" - strong surgery signal
Example 3: Hospice to Consult (Text: 4,897 chars) - Very long text, severe truncation, consult keywords present

LIMITATIONS AND FUTURE IMPROVEMENTS

Current Limitations

1. Text Truncation (Primary Limitation)

- 62.9% of samples truncated at 512 tokens
- Critical information loss in longer documents

- Impact: Estimated 10-15% accuracy loss

2. Class Imbalance

- 238:1 ratio between most and least common classes
- Rare specialties (< 10 samples) show 100% error rates
- Impact: F1-macro significantly lower than accuracy

3. Model Capacity

- DistilBERT lacks medical domain pre-training
- General vocabulary may miss specialized medical terminology
- Impact: Estimated 3-5% accuracy gap vs medical-specific models

4. Evaluation Limitations

- 4 specialties absent from test set (insufficient samples)
- Cannot assess generalization to unseen specialties

Proposed Improvements

High Priority (Immediate Impact):

1. Address Text Length

- Implement Longformer (4,096 token capacity)
- Or hierarchical classification (chunk → aggregate)
- Expected: +10-15% accuracy

2. Handle Class Imbalance

- Class-weighted loss function
- Data augmentation for minority classes
- Expected: +10-15% on rare classes

Medium Priority:

3. Hierarchical Classification

- Stage 1: Broad domain (Surgical/Medical/Diagnostic)
- Stage 2: Specific specialty within domain
- Expected: +5-8% accuracy, reduced confusion

4. Medical Domain Model

- Try BioBERT (PubMed/PMC pre-training)
- Expected: +3-5% from specialized vocabulary

Low Priority:

5. Extended Training

- Increase epochs with learning rate scheduling
- Expected: +2-4% from better convergence

Combined Potential: 65-75% accuracy achievable with Phase 1+2 improvements.

Deployment Considerations

For Production Implementation:

- Confidence thresholding: high-confidence (>70%) - low-confidence for human review
- A/B testing: 90% automated, 10% manual validation
- Continuous learning: Monthly retraining with corrected predictions
- Monitoring: Track accuracy by specialty, alert on performance degradation

CONCLUSION

This project successfully demonstrates,

1. Implementation of advanced ML techniques: LoRA achieved 98.65% parameter reduction while maintaining competitive performance (56.62% accuracy on challenging 40-class problem)
2. Novel findings: Discovered LoRA requires 10-15x higher learning rates than standard fine-tuning (LR=3e-4 optimal vs typical 2e-5)
3. Real-world applicability: System achieves production-feasible latency (29ms) and throughput (57 docs/sec) suitable for clinical deployment
4. Rigorous evaluation: Comprehensive comparison against 3 baselines (random, rule-based, pre-trained) demonstrating 22.6x, 1.64x, and 46x improvements respectively
5. Actionable insights: Identified text truncation and class imbalance as primary bottlenecks with concrete mitigation strategies

The project achieves its core objective of building a production-deployable medical specialty classifier while advancing understanding of parameter-efficient fine-tuning techniques in healthcare NLP applications.

REFERENCES

[1] Hu, E. J., et al. (2021). "LoRA: Low-Rank Adaptation of Large Language Models." arXiv:2106.09685.

[2] Sanh, V., et al. (2019). "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter." arXiv:1910.01108.

[3] Devlin, J., et al. (2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." NAACL-HLT.

[4] Lee, J., et al. (2020). "BioBERT: a pre-trained biomedical language representation model for biomedical text mining." Bioinformatics, 36(4), 1234-1240.

[5] HuggingFace Transformers Documentation. https://huggingface.co/docs/transformers/

[6] HuggingFace PEFT Documentation. https://huggingface.co/docs/peft/

[7] Dataset: galileo-ai/medical_transcription_40. https://huggingface.co/datasets/galileo-ai/medical_transcription_40