High-Performance Semantic Retrieval for Duplicate Question Detection on Quora via Fine-tuned Sentence-BERT

Submission Details and Technical Implementation

Yanzhi Ding
Student ID: 21163438
University of Waterloo
y48ding@uwaterloo.ca

August 1, 2025

1 Student Information

- **Name**: Yanzhi Ding
- **Student ID**: 21163438

2 Final Performance Results

2.1 Semantic Retrieval Performance

Metric	**Value**
Final Score (MRR)	**0.6401**
Recall@1	**0.5065**
Recall@5	**0.8226**
Recall@10	**0.9115**
Knowledge Base Size	297,750 questions
Test Queries	44,663 queries

Table 1: Primary Performance Metrics for Duplicate Question Detection

2.2 Baseline Comparison Results

Metric	**TF-IDF Baseline**	**Our Model**	**Improvement**
MRR	0.4482	**0.6401**	+42.8%
Recall@1	0.3156	**0.5065**	+60.5%
Recall@5	0.6319	**0.8226**	+30.2%
Recall@10	0.7428	**0.9115**	+22.7%

Table 2: Performance Comparison Against Traditional Baseline

3 Team Composition and Contributions

3.1 Team Members

This project was completed as an individual effort by:

- **Yanzhi Ding** (Student ID: 21163438)

3.2 Individual Contributions

As this was an individual project, all work was completed by Yanzhi Ding, including:

Data Analysis and Preprocessing: Conducted comprehensive analysis of the Quora Question Pairs dataset containing 537,000+ unique questions. Implemented sophisticated triplet construction pipeline generating 72,810 training triplets with anchor-positive-negative question combinations for metric learning.

Model Development and Implementation: Designed and implemented a complete Bi-Encoder architecture using `sentence-transformers/all-MiniLM-L6-v2` as the base model. Fine-tuned the model using Triplet Loss function with margin-based optimization, achieving significant domain adaptation for duplicate question detection.

Scalable Retrieval System: Integrated FAISS library for high-performance vector search, implementing `IndexFlatIP` for exact similarity search over 297,750 question embeddings. Developed comprehensive offline indexing and online query processing pipeline.

Comprehensive Evaluation Framework: Implemented robust evaluation using standard Information Retrieval metrics including MRR, Recall@k, Precision@k, NDCG@k, and MAP@k. Conducted detailed error analysis categorizing failure cases into semantic differences, world knowledge gaps, and syntactic complexity issues.

Technical Documentation: Authored complete technical report following ACL conference format, created all performance visualizations, and documented experimental procedures with full reproducibility guidelines including environment setup and dependency management.

4 Technical Implementation Summary

4.1 Model Architecture and Training

- **Base Model**: `sentence-transformers/all-MiniLM-L6-v2` (6-layer distilled BERT)
- **Output Dimensionality**: 384-dimensional sentence embeddings
- **Fine-tuning Strategy**: Triplet Loss with margin α for metric learning
- **Training Data**: 72,810 carefully constructed triplets from Quora dataset
- **Optimization**: AdamW optimizer with learning rate 2×10^{-5}
- **Training Duration**: 1 full epoch with average loss of 3.7671

4.2 Retrieval System Implementation

- **Vector Index**: FAISS `IndexFlatIP` for exact inner product search
- **Search Method**: Cosine similarity on normalized embeddings
- **Database Size**: 297,750 indexed question vectors
- **Query Processing**: Real-time encoding and top-k retrieval
- **Scalability**: Sub-second query response time with GPU acceleration

4.3 Evaluation and Analysis

- **Test Set**: 44,663 held-out queries for comprehensive evaluation
- **Primary Metrics**: Mean Reciprocal Rank (MRR) = 0.6401
- **Coverage Analysis**: 91.15% of queries find correct duplicates in top-10 results
- **Baseline Comparison**: 42.8% improvement over TF-IDF baseline in MRR
- **Error Analysis**: Systematic categorization of failure modes and limitations

5 Code Repository and Reproducibility

- **GitHub Repository**: ://github.com/Jonathan398/Quora-Duplicate-Question-Detection-with-Semantic-Search.git
- **Implementation Files**:
  - `QuoraDataPreparator.py`: Data processing and triplet generation
  - `BertModel.py`: Model training and fine-tuning implementation
  - `QA.py`: Evaluation and question answering system
- **Environment Setup**: Complete `requirements.txt` with pinned dependencies
- **Documentation**: Comprehensive README with installation and usage instructions

6 Performance Verification

This document certifies that all performance metrics and technical details listed above correspond to the final implementation of the semantic retrieval system for duplicate question detection on the Quora dataset. The results demonstrate significant improvements over traditional baseline methods and provide a scalable solution for large-scale duplicate detection tasks.

Key Achievements:

- **50.65%** of queries return correct duplicate as top result
- **82.26%** of queries find correct duplicate within top-5 results
- **91.15%** of queries find correct duplicate within top-10 results
- **42.8%** relative improvement over TF-IDF baseline

Submitted as part of NLP/Text Analytics Course Project
University of Waterloo
August 2025