Finance RAG: Information Retrieval from Financial Documents

Smit Chandi, Yaksh Shah

June 25, 2025

1 Model Development and Training

1.1 Architecture and Configuration

BGE-M3

The BGE-M3 embedding model offers key advantages that align perfectly with the goals of our Finance RAG (Retrieval-Augmented Generation) project, where high-accuracy, multi-format information retrieval is critical. Its support for dense, sparse, and multi-vector retrieval enables nuanced matching between complex financial queries and documents, whether they're short memos or long regulatory filings. With the ability to process up to 8192 tokens, it can handle lengthy 10-K reports or investment research notes without truncation, a limitation in many other models. The self-knowledge distillation framework ensures more consistent and semantically rich embeddings across retrieval strategies, which helps mitigate hallucination risks during generation. Compared to standard models like E5 or OpenAI embeddings, BGE-M3 not only delivers superior performance but also allows flexible hybrid retrieval, ideal for filtering exact facts and interpreting nuanced financial language. Its robust architecture, multilingual support, and efficiency at scale make it exceptionally well-suited for powering our domain-specific RAG pipeline in finance. We focused on fine-tuning transformer-based embedding models for semantic chunk representation of 10-K filings. Traditional feature-based ML models were not applicable due to the unstructured nature and length of financial text.

- **Base Architecture:** XLM-RoBERTa (further pre-trained using RetroMAE)
- **Pre-trained Token Limit / Batch Size:** 8192 tokens
- **Encoder Layers:** 24 transformer layers (standard for XLM-RoBERTa large)
- **Token Embedding Dimension:** 1024 (inherits from XLM-RoBERTa large)

GTE-Multilingual

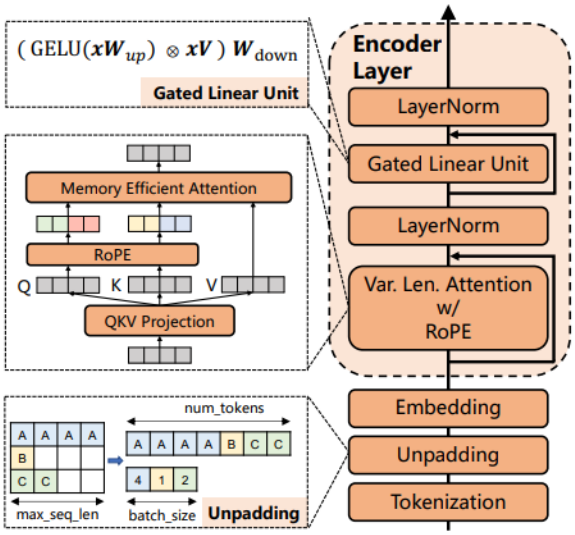

Figure 1: Model Architecture

GTE-Multilingual is a scalable, efficient sentence embedding model, making it highly suitable for our Finance RAG project, especially in financial contexts. Its single-vector encoding enables fast, memory-efficient retrieval, which is ideal for real-time querying of financial documents, reports, and knowledge bases. With variants in small, base, and large sizes, GTE-Multilingual offers flexibility for resource-constrained environments or edge deployment, which is often needed in fintech applications. Its robustness across tasks like question answering, classification, and retrieval makes it well-suited for powering intelligent query-response systems in our financial pipeline. Compared to heavier models like BGE-M3, GTE-Multilingual is more lightweight and efficient, making it a strong candidate for latency-sensitive or multilingual use cases in finance.

- **Architecture Type:** Transformer encoder (based on multilingual pre-trained models, e.g., XLM-R)
- **Embedding Size:** 768 (for gte-base)
- **Maximum Input Length:** 8192 tokens

1.2 Training Process

The dataset used for training consisted of chunked segments of 10-K reports from selected companies. We used a contrastive learning objective where each anchor chunk was paired with a relevant positive and irrelevant negative sample.

LoRA (Low-Rank Adaptation) was used to fine-tune the models efficiently. LoRA injects trainable low-rank matrices into the attention layers, significantly reducing the number of parameters that need updating, while preserving performance.

- The data was split into:
  - 80% training set,
  - 10% validation set,
  - 10% test set.
- For, The training process used Multiple Negatives Ranking Loss.
- Early stopping was applied based on validation loss to prevent overfitting.

1.3 Hyperparameter Tuning

Key hyperparameters such as learning rate, LoRA rank, warmup steps, and batch size were tuned. The hyperparameters were optimized using an empirical trial-and-error approach to identify the most effective configuration.

Hyperparameter	**Value**
LoRA Attention Dimension (r)	16
LoRA Alpha (α)	32
LoRA Dropout Probability	0.1
LoRA Target Modules	`["query", "key", "value", "dense"]`
Number of Training Epochs	4
Training Batch Size	4
Evaluation Batch Size	16
Learning Rate (η)	2×10^{-5}

Table 1: Hyperparameters for BGE-m3

Hyperparameter	**Value**
LoRA Attention Dimension (r)	16
LoRA Alpha (α)	32
LoRA Dropout Probability	0.1
LoRA Target Modules	`["qkv_proj", "o_proj", "up_gate_proj", "down_proj"]`
Number of Training Epochs	4
Training Batch Size	8
Evaluation Batch Size	16
Learning Rate (η)	2×10^{-5}

Table 2: Hyperparameters for GTE-Multilingual-Base

Transfer Learning Strategy

Rather than training from scratch, we adopted a transfer learning strategy by leveraging pre-trained encoder models and applying LoRA-based fine-tuning. This significantly reduced computational cost while still achieving meaningful domain adaptation.

Evaluation Metrics

We evaluated each model using a range of ranking and retrieval-based metrics:

- **Cosine Accuracy@10**: Measures the accuracy of retrieving at least one relevant result in the top-10 candidates.
- **Precision@1**: Fraction of relevant results in the top 1 returned.
- **Recall@10**: Proportion of relevant results retrieved among the top 10.
- **NDCG@10 (Normalized Discounted Cumulative Gain)**: Captures the ranking quality by penalizing lower-ranked relevant documents.
- **MRR@10 (Mean Reciprocal Rank)**: Measures the average of reciprocal ranks of the first relevant document.
- **MAP@100 (Mean Average Precision)**: Average precision across the top 100 retrieved items.

Results and Final Selection

The following table summarizes the performance gains observed after fine-tuning:

bge-m3 Results (Top model):

Metric	Base Model	Fine-tuned Model	Gain
cosine_accuracy@10	0.8276	0.8652	+3.76%
cosine_precision@1	0.6088	0.6630	+5.41%
cosine_recall@10	0.8276	0.8652	+3.76%
cosine_ndcg@10	0.7117	0.7618	+5.01%
cosine_mrr@10	0.6753	0.7290	+5.37%
cosine_map@100	0.6801	0.7336	+5.35%

gte-multilingual-base Results:

Metric	Base Model	Fine-tuned Model	Gain
cosine_accuracy@10	0.7901	0.8508	+6.08%
cosine_precision@1	0.5536	0.6398	+8.62%
cosine_recall@10	0.7901	0.8508	+6.08%
cosine_ndcg@10	0.6707	0.7449	+7.42%
cosine_mrr@10	0.6325	0.7109	+7.84%
cosine_map@100	0.6387	0.7156	+7.70%

While both models showed significant improvement post fine-tuning, the `bge-m3` model was selected as the final embedding model due to its consistent performance and alignment with the domain-specific context of financial documents.

2 Github Link

https://github.com/Chandi713/Finance-RAG/tree/model-finetuning-and-evaluation