Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models

Yanzhao Zhang* Mingxin Li* Dingkun Long* Xin Zhang*
Huan Lin Baosong Yang Pengjun Xie An Yang
Dayiheng Liu Junyang Lin Fei Huang Jingren Zhou
Tongyi Lab Alibaba Group

https://huggingface.co/Qwen
https://modelscope.cn/organization/qwen
https://github.com/QwenLM/Qwen3-Embedding

Abstract

In this work, we introduce the Qwen3 Embedding series, a significant advancement over its predecessor, the GTE-Qwen series, in text embedding and reranking capabilities, built upon the Qwen3 foundation models. Leveraging the Qwen3 LLMs' robust capabilities in multilingual text understanding and generation, our innovative multi-stage training pipeline combines large-scale unsupervised pre-training with supervised fine-tuning on high-quality datasets. Effective model merging strategies further ensure the robustness and adaptability of the Qwen3 Embedding series. During the training process, the Qwen3 LLMs serve not only as backbone models but also play a crucial role in synthesizing high-quality, rich, and diverse training data across multiple domains and languages, thus enhancing the training pipeline. The Qwen3 Embedding series offers a spectrum of model sizes (0.6B, 4B, 8B) for both embedding and reranking tasks, addressing diverse deployment scenarios where users can optimize for either efficiency or effectiveness. Empirical evaluations demonstrate that the Qwen3 Embedding series achieves state-of-the-art results across diverse benchmarks. Notably, it excels on the multilingual evaluation benchmark MTEB for text embedding, as well as in various retrieval tasks, including code retrieval, cross-lingual retrieval and multilingual retrieval. To facilitate reproducibility and promote community-driven research and development, the Qwen3 Embedding models are publicly available under the Apache 2.0 license.

1 Introduction

Text embedding and reranking are fundamental components in numerous natural language processing and information retrieval applications, including web search, question answering, recommendation systems, and beyond (Karpukhin et al., 2020; Huang et al., 2020; Zhao et al., 2023; 2024). High-quality embeddings enable models to capture semantic relationships between texts, while effective reranking mechanisms ensure that the most relevant results are prioritized. Recently, emerging application paradigms such as Retrieval-Augmented Generation (RAG) and agent systems, driven by the advancement of large language models (e.g., Qwen3 (Yang et al., 2025), GPT-4o (Hurst et al., 2024)), have introduced new requirements and challenges for text embedding and reranking, both in terms of model training paradigms and application scenarios. Despite significant advancements, training embedding and reranking models that perform well in scalability, contextual understanding, and alignment with specific downstream tasks remains challenging.

The emergence of large language models (LLMs) has significantly advanced the development of text embedding and reranking models. Prior to the introduction of LLMs, the predominant approach

* Equal contribution

Qwen3 Embedding：通过基础模型推进文本嵌入与重排序

张燕昭* 李明鑫* 龙定坤* 张欣* 林欢 杨宝松 谢鹏君 杨安 刘大一衡 林俊阳 黄飞 周靖人 通义实验室 阿里巴巴集团

https://huggingface.co/Qwen
https://modelscope.cn/organization/qwen
https://github.com/QwenLM/Qwen3-Embedding

摘要

在本项工作中，我们推出了 Qwen3 嵌入系列，这是在 Qwen3 基础模型之上构建的文本嵌入和重排序能力的重大进步，相较于前代 GTE-Qwen 系列。借助 Qwen3 大语言模型在多语言文本理解和生成方面的强大能力，我们创新的多阶段训练流程结合了大规模无监督预训练与高质量数据集上的监督微调。有效的模型合并策略进一步确保了 Qwen3 嵌入系列的鲁棒性和适应性。在训练过程中，Qwen3 大语言模型不仅作为骨干模型，还在跨多个领域和语言的高质量、丰富且多样训练数据的合成中发挥关键作用，从而增强训练流程。Qwen3 嵌入系列为嵌入和重排序任务提供多种模型尺寸（0.6B、4B、8B），满足不同部署场景需求，用户可根据效率或效果进行优化。实证评估表明，Qwen3 嵌入系列在多样化基准测试中实现了最先进的结果。值得注意的是，它在多语言评估基准 MTEB 的文本嵌入任务中表现出色，并在代码检索、跨语言检索和多语言检索等各类检索任务中取得优异成绩。为促进可复现性并推动社区驱动的研究与开发，Qwen3 嵌入模型在 Apache 2.0 许可下公开提供。

1 简介

文本嵌入和重排序是众多自然语言处理和信息检索应用中的基本组成部分，包括网络搜索、问答系统、推荐系统等（Karpukhin 等，2020；Huang 等，2020；Zhao 等，2023；2024）。高质量的嵌入使模型能够捕捉文本之间的语义关系，而有效的重排序机制则确保最相关的结果被优先排序。最近，随着大型语言模型（如 Qwen3（Yang 等，2025）、GPT-4o（Hurst 等，2024））的发展，检索增强生成（RAG）和代理系统等新兴应用范式对文本嵌入和重排序提出了新的要求和挑战，无论是在模型训练范式还是应用场景方面。尽管取得了显著进展，训练在可扩展性、上下文理解和与特定下游任务对齐方面表现良好的嵌入和重排序模型仍然具有挑战性。

大语言模型（LLMs）的出现显著推动了文本嵌入和重排序模型的发展。在引入LLMs之前，主流方法

* Equal contribution

involved using encoder-only pretrained language models like BERT as the foundational model for training (Reimers & Gurevych, 2019). The richer world knowledge, text understanding, and reasoning abilities inherent in LLMs have led to further enhancements in models trained on these architectures. Additionally, there has been considerable research facilitating the integration of LLMs into processes such as training data synthesis and quality data filtering (Wang et al., 2024; Lee et al., 2024; 2025b). The fundamental characteristics of LLMs have also inspired the introduction of new training paradigms. For instance, during the embedding model training process, incorporating differentiated tasks across aspects such as instruction type, domain, and language has yielded improved performance in downstream tasks (Su et al., 2023). Similarly, for reranking model training, advancements have been realized through both zero-shot methods based on user prompts and approaches combining supervised fine-tuning (Ma et al., 2023; Pradeep et al., 2023; Zhang et al., 2024a; Zhuang et al., 2024).

In this work, we introduce the Qwen3 Embedding series models, which are constructed on top of the Qwen3 foundation models. The Qwen3 foundation has simultaneously released base and instruct model versions, and we exploit the robust multilingual text understanding and generation capabilities of these models to fully realize their potential in training embedding and reranking models. To train the embedding models, we implement a multi-stage training pipeline that involves large-scale unsupervised pre-training followed by supervised fine tuning on high-quality datasets. We also employ model merging with various model checkpoints to enhance robustness and generalization. The Qwen3 instruct model allows for efficient synthesis of a vast, high-quality, multilingual, and multi-task text relevance dataset. This synthetic data is utilized in the initial unsupervised training stage, while a subset of high-quality, small-scale data is selected for the second stage of supervised training. For the reranking models, we adopt a two-stage training scheme in a similar manner, consisting of high-quality supervised fine tuning and a model merging stage. Based on different sizes of the Qwen3 backbone models (including 0.6B, 4B, and 8B), we ultimately trained three text embedding models and three text reranking models. To facilitate their application in downstream tasks, the Qwen3 Embedding series supports several practical features, such as flexible dimension representation for embedding models and customizable instructions for both embedding and reranking models.

We evaluate the Qwen3 Embedding series across a comprehensive set of benchmarks spanning multiple tasks and domains. Experimental results demonstrate that our embedding and reranking models achieve state-of-the-art performance, performing competitively against leading proprietary models in several retrieval tasks. For example, the flagship model Qwen3-8B-Embedding attains a score of 70.58 on the MTEB Multilingual benchmark (Enevoldsen et al., 2025) and 80.68 on the MTEB Code benchmark (Enevoldsen et al., 2025), surpassing the previous state-of-the-art proprietary embedding model, Gemini-Embedding (Lee et al., 2025b). Moreover, our reranking model delivers competitive results across a range of retrieval tasks. The Qwen3-Reranker-0.6B model exceeds previously top-performing models in numerous retrieval tasks, while the larger Qwen3-Reranker-8B model demonstrates even superior performance, improving ranking results by 3.0 points over the 0.6B model across multiple tasks. Furthermore, we include a constructive ablation study to elucidate the key factors contributing to the superior performance of the Qwen3 Embedding series, providing insights into its effectiveness.

In the following sections, we describe the design of the model architecture, detail the training procedures, present the experimental results for both the embedding and reranking models of the Qwen3 Embedding Series, and conclude this technical report by summarizing the key findings and outlining potential directions for future research.

2 Model Architecture

The core idea behind embedding and reranking models is to evaluate relevance in a task-aware manner. Given a query q and a document d, embedding and reranking models assess their relevance based on a similarity criterion defined by instruction I. To enable the models for task-aware relevance estimation, training data is often organized as $\{I_i, q_i, d_i^+, d_{i,1}^-, \cdots, d_{i,n}^-\}$, where d_i^+ represents a

涉及使用仅编码器的预训练语言模型（如BERT）作为训练的基础模型（Reimers & Gurevych, 2019）。LLM所固有的更丰富的世界知识、文本理解和推理能力，使得基于这些架构训练的模型性能进一步提升。此外，已有大量研究促进将LLM整合到训练数据合成和高质量数据过滤等流程中（Wang等，2024；Lee等，2024；2025b）。LLM的基本特性还激发了新训练范式的引入。例如，在嵌入模型训练过程中，通过在指令类型、领域和语言等方面进行差异化的任务设计，可提升下游任务性能（Su等，2023）。同样，在重排序模型训练中，通过基于用户提示的零样本方法以及结合监督微调的策略，已取得显著进展（Ma等，2023；Pradeep等，2023；Zhang等，2024a；Zhuang等，2024）。

在本工作中，我们介绍了基于Qwen3基础模型构建的Qwen3 Embedding系列模型。Qwen3基础模型同时发布了基础版和指令版，我们利用这些模型强大的多语言文本理解和生成能力，充分挖掘其在嵌入模型和重排序模型训练中的潜力。为训练嵌入模型，我们实现了一个多阶段训练流程，包括大规模无监督预训练，随后在高质量数据集上进行监督微调。我们还通过合并不同模型检查点来增强模型的鲁棒性和泛化能力。Qwen3指令模型可高效生成大规模、高质量、多语言、多任务的文本相关性数据集，该合成数据用于初始的无监督训练阶段，同时选取其中一小部分高质量、小规模数据用于第二阶段的监督训练。对于重排序模型，我们采用类似的两阶段训练方案，包括高质量监督微调和模型合并阶段。基于不同规模的Qwen3主干模型（包括0.6B、4B和8B），我们最终训练了三个文本嵌入模型和三个文本重排序模型。为便于其在下游任务中的应用，Qwen3 Embedding系列支持多种实用功能，例如嵌入模型的灵活维度表示以及嵌入和重排序模型的可定制指令。

我们在涵盖多个任务和领域的全面基准测试集上评估了Qwen3 Embedding系列模型。实验结果表明，我们的嵌入和重排序模型实现了最先进的性能，在多个检索任务中能与领先的专有模型展开竞争。例如，旗舰模型Qwen3-8B-Embedding在MTEB Multilingual基准（Enevoldsen等，2025）上获得70.58分，在MTEB Code基准（Enevoldsen等，2025）上获得80.68分，超越了此前最先进的专有嵌入模型Gemini-Embedding（Lee等，2025b）。此外，我们的重排序模型在多种检索任务中表现出竞争力。Qwen3-Reranker-0.6B模型在众多检索任务中超越了此前表现最佳的模型，而更大的Qwen3-Reranker-8B模型展现出更优的性能，在多个任务中相比0.6B模型提升了3.0个排名分数。此外，我们还进行了建设性的消融研究，阐明了Qwen3 Embedding系列卓越性能的关键因素，为其有效性提供了深入见解。

在以下章节中，我们描述了模型架构的设计，详细说明了训练过程，展示了Qwen3 Embedding Series中嵌入模型和重排序模型的实验结果，并通过总结关键发现和概述未来研究的潜在方向来结束本技术报告。

2 模型架构

嵌入和重排序模型的核心思想是以任务感知的方式评估相关性。给定一个查询q和一个文档d，嵌入和重排序模型会根据由指令I定义的相似性标准来评估它们的相关性。为了使模型能够进行任务感知的相关性估计，训练数据通常组织为$\{I, q_i, d_i^+, d_{i,1}^-, \cdots, d_{i,n}^-\}$，其中$d_i^+$表示一个

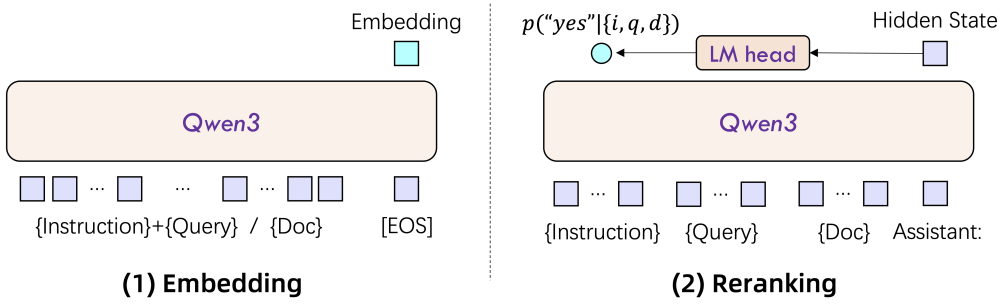

Figure 1: Model architecture of Qwen3-Embedding (left) and Qwen3-Reranker (right).

positive (relevant) document for query q_i, and $d_{i,j}^-$ are negative (irrelevant) documents. Training the model on diverse text pairs broadens its applicability to a range of downstream tasks, including retrieval, semantic textual similarity, classification, and clustering.

Architecture The Qwen3 embedding and reranking models are built on the dense version of Qwen3 foundation models and are available in three sizes: 0.6B, 4B, and 8B parameters. We initialize these models using the Qwen3 foundation models to leverage their capabilities in text modeling and instruction following. The model layers, hidden size, and context length for each model configuration are detailed in Table 1.

Embedding Models For text embeddings, we utilize LLMs with causal attention, appending an `[EOS]` token at the end of the input sequence. The final embedding is derived from the hidden state of the last layer corresponding to this `[EOS]` token.

To ensure embeddings follow instructions during downstream tasks, we concatenate the instruction and the query into a single input context, while leaving the document unchanged before processing with LLMs. The input format for queries is as follows:

```
{Instruction} {Query}<|endoftext|>
```

Reranking Models To more accurately evaluate text similarity, we employ LLMs for point-wise reranking within a single context. Similar to the embedding model, to enable instruction-following capability, we include the instruction in the input context. We use the LLM chat template and frame the similarity assessment task as a binary classification problem. The input to LLMs adheres to the template shown below:

```
<|im_start|>system
Judge whether the Document meets the requirements based on the Query and the
↪  Instruct provided. Note that the answer can only be "yes" or
↪  "no".<|im_end|>
<|im_start|>user
<Instruct>: {Instruction}
<Query>: {Query}
<Document>: {Document}<|im_end|>
<|im_start|>assistant
<think>\n\n</think>\n\n
```

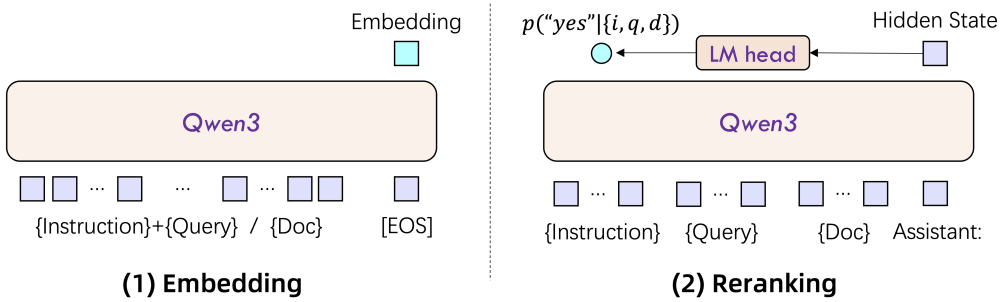

图1：Qwen3-Embedding（左）和Qwen3-Reranker（右）的模型架构。

q_i 是查询的相关（正面）文档，而 $d_{i,j}^-$ 是无关（负面）文档。在多样化的文本对上训练模型拓宽了其在一系列下游任务中的适用性，包括检索、语义文本相似性、分类和聚类。

架构 Qwen3的嵌入和重排序模型基于Qwen3基础模型的密集版本构建，提供三种规模：0.6B、4B和8B参数。我们使用Qwen3基础模型对这些模型进行初始化，以利用其在文本建模和指令遵循方面的能力。每种模型配置的模型层数、隐藏层大小和上下文长度详见表1。

文本嵌入模型 对于文本嵌入，我们利用具有因果注意机制的大语言模型 (LLMs)，在输入序列的末尾附加一个 `[EOS]` 标记。最终的嵌入是从与该 `[EOS]` 标记对应的最后一层的隐藏状态中得出的。

为了确保嵌入在下游任务中遵循指令，我们在使用LLMs处理之前，将指令和查询连接成一个输入上下文，同时保持文档不变。查询的输入格式如下：

```
{Instruction} {Query}<|endoftext|>
```

重排序模型 为了更准确地评估文本相似度，我们采用大语言模型（LLMs）进行单个上下文内的逐点重排序。类似于嵌入模型，为了启用遵循指令的能力，我们在输入上下文中包含指令。我们使用LLM的聊天模板，并将相似度评估任务构造成二分类问题。输入LLM的内容遵循以下模板：

```
<|im_start|>system
Judge whether the Document meets the requirements based on the Query and the
↪  Instruct provided. Note that the answer can only be "yes" or
↪  "no".<|im_end|>
<|im_start|>user
<Instruct>: {Instruction}
<Query>: {Query}
<Document>: {Document}<|im_end|>
<|im_start|>assistant
<think>\n\n</think>\n\n
```

Model Type	Models	Size	Layers	Sequence Length	Embedding Dimension	MRL Support	Instruction Aware
Text Embedding	Qwen3-Embedding-0.6B	0.6B	28	32K	1024	Yes	Yes
	Qwen3-Embedding-4B	4B	36	32K	2560	Yes	Yes
	Qwen3-Embedding-8B	8B	36	32K	4096	Yes	Yes
Text Reranking	Qwen3-Reranker-0.6B	0.6B	28	32K	-	-	Yes
	Qwen3-Reranker-4B	4B	36	32K	-	-	Yes
	Qwen3-Reranker-8B	8B	36	32K	-	-	Yes

Table 1: Model architecture of Qwen3 Embedding models. "MRL Support" indicates whether the embedding model supports custom dimensions for the final embedding. "Instruction Aware" notes whether the embedding or reranker model supports customizing the input instruction according to different tasks.

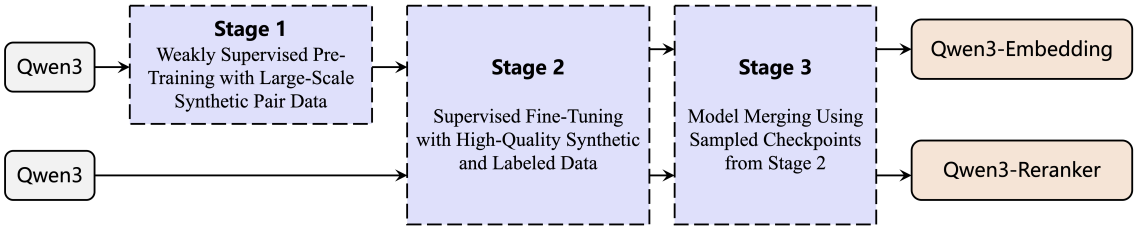

Figure 2: Training pipeline of Qwen3 Embedding and Reranking models.

To calculate the relevance score based on the given input, we assess the likelihood of the next token being "yes" or "no." This is expressed mathematically as follows:

$$\text{score}(q,d) = \frac{e^{P(\text{yes}|I,q,d)}}{e^{P(\text{yes}|I,q,d)} + e^{P(\text{no}|I,q,d)}}$$

3 Models Training

In this section, we describe the multi-stage training pipeline adopted and present the key elements of this training recipe, including training objective, training data synthesis, and filtering of high-quality training data.

3.1 Training Objective

Before introducing our training pipeline, we first outline the optimized loss functions used for the embedding and reranking models during the training process. For the embedding model, we utilize an improved contrastive loss based on the InfoNCE framework (Oord et al., 2018). Given a batch of N training instances, the loss is defined as:

$$L_{\text{embedding}} = -\frac{1}{N}\sum_{i}^{N} \log \frac{e^{(s(q_i,d_i^+)/\tau)}}{Z_i}, \tag{1}$$

where $s(\cdot,\cdot)$ is a similarity function (we use cosine similarity), τ is a temperature parameter, and Z_i is the normalization factor that aggregates the similarity scores of the positive pair against various negative pairs:

$$Z_i = e^{(s(q_i,d_i^+)/\tau)} + \sum_{k}^{K} m_{ik} e^{(s(q_i,d_{i,k}^-)/\tau)} + \sum_{j\neq i} m_{ij} e^{(s(q_i,q_j)/\tau)} + \sum_{j\neq i} m_{ij} e^{(s(d_i^+,d_j)/\tau)} + \sum_{j\neq i} m_{ij} e^{(s(q_i,d_j)/\tau)}$$

Model Type	Models	Size	Layers	Sequence Length	Embedding Dimension	MRL Support	Instruction Aware
Text Embedding	Qwen3-Embedding-0.6B	0.6B	28	32K	1024	Yes	Yes
	Qwen3-Embedding-4B	4B	36	32K	2560	Yes	Yes
	Qwen3-Embedding-8B	8B	36	32K	4096	Yes	Yes
Text Reranking	Qwen3-Reranker-0.6B	0.6B	28	32K	-	-	Yes
	Qwen3-Reranker-4B	4B	36	32K	-	-	Yes
	Qwen3-Reranker-8B	8B	36	32K	-	-	Yes

表1：Qwen3嵌入模型的模型架构。“MRL支持”表示嵌入模型是否支持为最终嵌入自定义维度。“指令感知”说明嵌入或重排序模型是否支持根据不同的任务自定义输入指令。

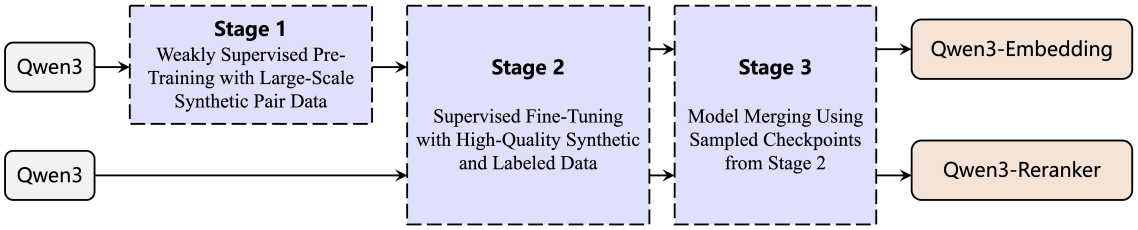

图2：Qwen3 嵌入和重排序模型的训练流程。

为了根据给定的输入计算相关性分数，我们评估下一个标记是 "yes" 或 "no" 的可能性。这在数学上表示如下：

$$\text{score}(q,d) = \frac{e^{P(\text{yes}|I,q,d)}}{e^{P(\text{yes}|I,q,d)} + e^{P(\text{no}|I,q,d)}}$$

3模型训练

在本节中，我们描述了采用的多阶段训练流程，并介绍了该训练方案的关键要素，包括训练目标、训练数据合成以及高质量训练数据的筛选。

3.1 训练目标

在介绍我们的训练流程之前，我们首先概述在训练过程中用于嵌入模型和重排序模型的优化损失函数。对于嵌入模型，我们利用基于InfoNCE框架（Oord等，2018）的改进对比损失函数。给定一个包含N个训练实例的批次，损失函数定义为：

$$L_{\text{embedding}} = -\frac{1}{N}\sum_{i}^{N}\log\frac{e^{(s(q_i,d_i^+)/\tau)}}{Z_i}, \tag{1}$$

其中 $s(\cdot、\cdot)$ 是相似度函数（我们使用余弦相似度），τ 是温度参数，Z_i 是标准化因子，用于聚合正样本对与各个负样本对的相似度得分：

$$Z_i = e^{(s(q_i,d_i^+)/\tau)} + \sum_{k}^{K} m_{ik} e^{(s(q_i,d_{i,k}^-)/\tau)} + \sum_{j\neq i} m_{ij} e^{(s(q_i,q_j)/\tau)} + \sum_{j\neq i} m_{ij} e^{(s(d_i^+,d_j)/\tau)} + \sum_{j\neq i} m_{ij} e^{(s(q_i,d_j)/\tau)}$$

where these terms represent similarities with: (1) the positive document d_i^+, (2) K hard negatives $d_{i,k}^-$, (3) other in-batch queries q_j, (4) other in-batch documents d_j compared against the positive document d_i^+. (5) other in-batch documents d_j compared against the query q_i. The mask factor m_{ij} is designed to mitigate the impact of false negatives and is defined as:

$$m_{ij} = \begin{cases} 0 & \text{if } s_{ij} > s(q_i, d_i^+) + 0.1 \text{ or } d_j == d_i^+, \\ 1 & \text{otherwise,} \end{cases}$$

among which s_{ij} is the corresponding score of q_i, d_j or q_i, q_j.

For the reranking model, we optimize the Supervised Fine-Tuning (SFT) loss defined as:

$$L_{\text{reranking}} = -\log p(l|\mathcal{P}(q,d)), \tag{2}$$

where $p(\cdot|*)$ denotes the probability assigned by LLM. The label l is "yes" for positive documents and "no" for negatives. This loss function encourages the model to assign higher probabilities to correct labels, thereby improving the ranking performance.

3.2 Multi-stage Training

The multi-stage training approach is a common practice for training text embedding models (Li et al., 2023; Wang et al., 2022; Chen et al., 2024). This strategy typically begins with initial training on large-scale, semi-supervised data that includes noise, followed by fine-tuning using smaller, high-quality supervised datasets. This two-step process enhances the performance and generalization capabilities of embedding models. Large-scale weakly supervised training data contribute significantly to the model's generalization, while fine-tuning with high-quality data in subsequent stages further improves model performance. Both stages of training for embedding models utilize the optimization objective defined in Equation 1, whereas the reranking model training employs the loss function defined in Equation 2 as the optimization target.

Building upon the existing multi-stage training framework, the Qwen3 Embedding series introduces the following key innovations:

- Large-Scale Synthetic Data-Driven Weak Supervision Training: Unlike previous works (e.g., GTE, E5, BGE models), where weakly supervised training data are primarily collected from open-source communities such as Q&A forums or academic papers, we propose leveraging the text understanding and generation capabilities of foundation models to synthesize pair data directly. This approach allows for arbitrary definition of various dimensions of the desired pair data, such as task, language, length, and difficulty within the synthesis prompts. Compared to data collection from open-domain sources, foundation model-driven data synthesis offers greater controllability, enabling precise management of the quality and diversity of the generated data, particularly in low-resource scenarios and languages.
- High-Quality Synthetic Data Utilization in Supervised Fine Tuning: Due to the exceptional performance of the Qwen3 Foundation model, the synthesized data is of notably high quality. Therefore, in the second stage of supervised training, selective incorporation of this high-quality synthetic data further enhances the overall model performance and generalization capabilities.
- Model Merging: Inspired by previous work (Li et al., 2024), after completing the supervised fine-tuning, we applied a model merging technique based on spherical linear interpolation (slerp). This technique involves merging multiple model checkpoints saved during the fine-tuning process. This step aims to boost the model's robustness and generalization performance across various data distributions.

It is important to note that the reranking model's training process does not include a first-stage weakly supervised training phase.

这些项表示与以下内容的相似性：(1) 正样本文档 d_i^+，(2) K 硬负样本 $d_{i,k}^-$，(3) 其他同批次查询 q_j，(4) 与正样本文档 d_i^+ 相比的其他同批次文档 d_j。(5) 与查询 q_i 相比的其他同批次文档 d_j。掩码因子 m_{ij} 的设计目的是减轻假负样本的影响，其定义为：

$$m_{ij} = \begin{cases} 0 & \text{if } s_{ij} > s(q_i, d_i^+) + 0.1 \text{ or } d_j == d_i^+, \\ 1 & \text{otherwise,} \end{cases}$$

其中 s_{ij} 是 q_i、d_j 或 q_i、q_j 的对应分数。

对于重排序模式 l，我们优化监督微调（SFT）损失函数定义 Translated Text:

$$L_{\text{reranking}} = -\log p(l|\mathcal{P}(q,d)), \tag{2}$$

其中 $p(\cdot|*)$ 表示由LLM分配的概率。标签 l 对于正样本文档为“是”，对于负样本文档为“否”。该损失函数促使模型为正确的标签分配更高的概率，从而提高排序性能。

3.2 多阶段训练

多阶段训练方法是训练文本嵌入模型的常见做法（李等人，2023年；王等人，2022年；陈等人，2024年）。该策略通常首先在包含噪声的大规模半监督数据上进行初始训练，随后使用较小规模的高质量监督数据集进行微调。这种两步流程提升了嵌入模型的性能和泛化能力。大规模弱监督训练数据对模型的泛化能力有显著贡献，而后续阶段使用高质量数据进行微调则进一步提升了模型性能。嵌入模型的两个训练阶段均采用公式1中定义的优化目标，而重排序模型训练则采用公式2中定义的损失函数作为优化目标。

在现有多阶段训练框架的基础上，Qwen3 Embedding 系列引入了以下关键创新：

- 大规模合成数据驱动的弱监督训练：与以往工作（例如，GTE、E5、BGE模型）不同，这些工作中的弱监督训练数据主要从问答论坛或学术论文等开源社区收集，我们提出利用基础模型的文本理解和生成能力直接合成配对数据。这种方法允许在合成提示中任意定义所需配对数据的各种维度，例如任务、语言、长度和难度。与从开放域来源收集数据相比，基础模型驱动的数据合成具有更高的可控性，能够精确管理生成数据的质量和多样性，特别是在低资源场景和语言中。

- 监督微调中的高质量合成数据利用：由于Qwen3基础模型的卓越性能，合成数据具有显著的高质量特性。因此，在监督训练的第二阶段，选择性地引入这种高质量合成数据进一步提升了模型的整体性能和泛化能力。

- 模型合并：受先前工作（Li 等，2024）的启发，在完成监督微调后，我们应用了一种基于球面线性插值（slerp）的模型合并技术。该技术涉及合并微调过程中保存的多个模型检查点。此步骤旨在提升模型在不同数据分布下的鲁棒性和泛化性能。

需要注意的是，重排序模型的训练过程不包括第一阶段的弱监督训练阶段。

3.3 Synthetic Dataset

To create a robust synthetic dataset for training models on various similarity tasks, we generate diverse text pairs spanning categories such as retrieval, bitext mining, classification, and semantic textual similarity (STS). The quality of these synthetic data pairs is ensured by utilizing the Qwen3-32B model as the foundational model for data synthesis. We have designed a diverse prompting strategy to improve the variety and authenticity of the generated data. For instance, in the text retrieval task, we synthesize data using the multilingual pre-training corpus from Qwen3. During the data synthesis process, specific roles are assigned to each document to simulate potential users querying that document. This injection of user perspectives enhances the diversity and realism of the synthetic queries. Specifically, we utilize a retrieval model to identify the top five role candidates for each document from a role library and present these documents along with their role candidates to the prompt. This guides the model in outputting the most suitable role configuration for query generation. Moreover, the prompt incorporates various dimensions such as query type (e.g., keyword, factual, summary, judgment), query length, difficulty, and language. This multidimensional approach ensures the quality and diversity of the synthetic data.

Finally, we create a total of approximately 150 million pairs of multi-task weak supervision training data. Our experiments reveal that the embedding model trained with these synthetic data performs exceptionally well in downstream evaluations, particularly surpassing many previously supervised models in the MTEB Multilingual benchmarks. This motivates us to filter the synthetic data to identify high-quality pairs for inclusion in a second stage of supervised training. We employ a simple cosine similarity calculation to select data pairs, retaining those with a cosine similarity greater than 0.7 from randomly sampled data. Ultimately, approximately 12 million high-quality supervised training data pairs are selected for further training.

Model	**Size**	**Mean (Task)**	**Mean (Type)**	Bitext Mining	Class-ification	Clus-tering	Inst. Retrieval	Multilabel Class.	Pair Class.	Rerank	Retrieval	STS
Selected Open-Source Models												
NV-Embed-v2	7B	56.29	49.58	57.84	57.29	40.80	1.04	18.63	78.94	63.82	56.72	71.10
GritLM-7B	7B	60.92	53.74	70.53	61.83	49.75	3.45	22.77	79.94	63.78	58.31	73.33
BGE-M3	0.6B	59.56	52.18	79.11	60.35	40.88	-3.11	20.1	80.76	62.79	54.60	74.12
multilingual-e5-large-instruct	0.6B	63.22	55.08	80.13	64.94	50.75	-0.40	22.91	80.86	62.61	57.12	76.81
gte-Qwen2-1.5B-instruct	1.5B	59.45	52.69	62.51	58.32	52.05	0.74	24.02	81.58	62.58	60.78	71.61
gte-Qwen2-7b-Instruct	7B	62.51	55.93	73.92	61.55	52.77	4.94	25.48	85.13	65.55	60.08	73.98
Commercial APIs												
text-embedding-3-large	-	58.93	51.41	62.17	60.27	46.89	-2.68	22.03	79.17	63.89	59.27	71.68
Cohere-embed-multilingual-v3.0	-	61.12	53.23	70.50	62.95	46.89	-1.89	22.74	79.88	64.07	59.16	74.80
Gemini Embedding	-	68.37	59.59	79.28	71.82	54.59	5.18	**29.16**	83.63	65.58	67.71	79.40
Qwen3 Embedding Models												
Qwen3-Embedding-0.6B	0.6B	64.33	56.00	72.22	66.83	52.33	5.09	24.59	80.83	61.41	64.64	76.17
Qwen3-Embedding-4B	4B	69.45	60.86	79.36	72.33	57.15	**11.56**	26.77	85.05	65.08	69.60	80.86
Qwen3-Embedding-8B	8B	**70.58**	**61.69**	**80.89**	**74.00**	**57.65**	10.06	28.66	**86.40**	**65.63**	**70.88**	**81.08**

Table 2: Performance on MTEB Multilingual (Enevoldsen et al., 2025). For compared models, the scores are retrieved from MTEB online leaderboard on June 4th, 2025.

4 Evaluation

We conduct comprehensive and fair evaluations across multiple benchmarks to assess the capabilities of Qwen3 Embedding models.

4.1 Settings

For the text embedding models, we utilize the Massive Multilingual Text Embedding Benchmark (MMTEB) (Enevoldsen et al., 2025) for evaluation. MMTEB is a large-scale, community-driven expansion of MTEB (Muennighoff et al., 2023), covering over 500 quality-controlled evaluation tasks

3.3 合成数据集

为了创建一个用于训练各种相似性任务模型的鲁棒合成数据集，我们生成涵盖检索、双语对齐挖掘、分类和语义文本相似性（STS）等类别的多样化文本对。这些合成数据对的质量通过使用Qwen3-32B模型作为数据合成的基础模型来确保。我们设计了一种多样化的提示策略，以提高生成数据的多样性和真实性。例如，在文本检索任务中，我们利用Qwen3的多语言预训练语料库合成数据。在数据合成过程中，为每个文档分配特定角色，以模拟潜在用户对该文档的查询。这种用户视角的注入增强了合成查询的多样性和真实性。具体而言，我们利用检索模型从角色库中为每个文档识别前五个角色候选，并将这些文档及其角色候选呈现给提示。这引导模型输出最适合查询生成的角色配置。此外，提示融入了查询类型（例如，关键词、事实性、摘要、判断）、查询长度、难度和语言等多个维度。这种多维方法确保了合成数据的质量和多样性。

最终，我们总共创建了大约1.5亿对多任务弱监督训练数据。我们的实验表明，使用这些合成数据训练的嵌入模型在下游评估中表现出色，特别是在MTEB多语言基准测试中超越了许多先前的监督模型。这促使我们过滤合成数据，以识别高质量的数据对用于监督训练的第二阶段。我们采用简单的余弦相似度计算来选择数据对，从随机采样的数据中保留余弦相似度大于0.7的数据对。最终，约有1200万对高质量监督训练数据被选中用于进一步训练。

Model	Size	Mean (Task)	Mean (Type)	Bitext Mining	Class-ification	Clus-tering	Inst. Retrieval	Multilabel Class.	Pair Class.	Rerank	Retrieval	STS
Selected Open-Source Models												
NV-Embed-v2	7B	56.29	49.58	57.84	57.29	40.80	1.04	18.63	78.94	63.82	56.72	71.10
GritLM-7B	7B	60.92	53.74	70.53	61.83	49.75	3.45	22.77	79.94	63.78	58.31	73.33
BGE-M3	0.6B	59.56	52.18	79.11	60.35	40.88	-3.11	20.1	80.76	62.79	54.60	74.12
multilingual-e5-large-instruct	0.6B	63.22	55.08	80.13	64.94	50.75	-0.40	22.91	80.86	62.61	57.12	76.81
gte-Qwen2-1.5B-instruct	1.5B	59.45	52.69	62.51	58.32	52.05	0.74	24.02	81.58	62.58	60.78	71.61
gte-Qwen2-7b-Instruct	7B	62.51	55.93	73.92	61.55	52.77	4.94	25.48	85.13	65.55	60.08	73.98
Commercial APIs												
text-embedding-3-large	-	58.93	51.41	62.17	60.27	46.89	-2.68	22.03	79.17	63.89	59.27	71.68
Cohere-embed-multilingual-v3.0	-	61.12	53.23	70.50	62.95	46.89	-1.89	22.74	79.88	64.07	59.16	74.80
Gemini Embedding	-	68.37	59.59	79.28	71.82	54.59	5.18	**29.16**	83.63	65.58	67.71	79.40
Qwen3 Embedding Models												
Qwen3-Embedding-0.6B	0.6B	64.33	56.00	72.22	66.83	52.33	5.09	24.59	80.83	61.41	64.64	76.17
Qwen3-Embedding-4B	4B	69.45	60.86	79.36	72.33	57.15	**11.56**	26.77	85.05	65.08	69.60	80.86
Qwen3-Embedding-8B	8B	**70.58**	**61.69**	**80.89**	**74.00**	**57.65**	10.06	28.66	**86.40**	**65.63**	**70.88**	**81.08**

表2：在MTEB多语言上的表现（Enevoldsen等，2025）。对于比较的模型，分数是从MTEB在线排行榜于2025年6月4日获取的。

4 评估

我们在多个基准测试中进行全面且公正的评估，以评估Qwen3嵌入模型的能力。

4.1 设置

对于文本嵌入模型，我们利用大规模多语言文本嵌入基准测试（MMTEB）（Enevoldsen 等，2025）进行评估。MMTEB 是 MTEB（Muennighoff 等，2023）的大规模、社区驱动的扩展，涵盖了超过500个质量控制的评估任务。

Model	Size	Dim	MTEB (Eng, v2)		CMTEB		MTEB (Code)
			Mean (Task)	Mean (Type)	Mean (Task)	Mean (Type)	
Selected Open-Source Models							
NV-Embed-v2	7B	4096	69.81	65.00	63.0	62.0	-
GritLM-7B	7B	4096	67.07	63.22	-	-	73.6[α]
multilingual-e5-large-instruct	0.6B	1024	65.53	61.21	-	-	65.0[α]
gte-Qwen2-1.5b-instruct	1.5B	1536	67.20	63.26	67.12	67.79	-
gte-Qwen2-7b-instruct	7B	3584	70.72	65.77	71.62	72.19	56.41[γ]
Commercial APIs							
text-embedding-3-large	-	3072	66.43	62.15	-	-	58.95[γ]
cohere-embed-multilingual-v3.0	-	1024	66.01	61.43	-	-	51.94[γ]
Gemini Embedding	-	3072	73.30	67.67	-	-	74.66[γ]
Qwen3 Embedding Models							
Qwen3-Embedding-0.6B	0.6B	1024	70.70	64.88	66.33	67.44	75.41
Qwen3-Embedding-4B	4B	2560	74.60	68.09	72.26	73.50	80.06
Qwen3-Embedding-8B	8B	4096	**75.22**	**68.70**	**73.83**	**75.00**	**80.68**

Table 3: Performance on MTEB Engilish, MTEB Chinese, MTEB Code. [α]Taken from (Enevoldsen et al., 2025). [γ]Taken from (Lee et al., 2025b). For other compared models, the scores are retrieved from MTEB online leaderboard on June 4th, 2025.

across more than 250 languages. In addition to classic text tasks such as as a variety of retrieval, classification, and semantic textual similarity, MMTEB includes a diverse set of challenging and novel tasks, such as instruction following, long-document retrieval, and code retrieval, representing the largest multilingual collection of evaluation tasks for embedding models to date. Our MMTEB evaluations encompass 216 individual evaluation tasks, consisting of 131 tasks for MTEB (Multilingual) (Enevoldsen et al., 2025), 41 tasks for MTEB (English, v2) (Muennighoff et al., 2023), 32 tasks for CMTEB (Xiao et al., 2024), and 12 code retrieval tasks for MTEB (Code) (Enevoldsen et al., 2025).

Moreover, we select a series of text retrieval tasks to assess the text reranking capabilities of our models. We explore three types of retrieval tasks: (1) Basic Relevance Retrieval, categorized into English, Chinese, and Multilingual, evaluated on MTEB (Muennighoff et al., 2023), CMTEB (Xiao et al., 2024), MMTEB (Enevoldsen et al., 2025), and MLDR (Chen et al., 2024), respectively; (2) Code Retrieval, evaluated on MTEB-Code (Enevoldsen et al., 2025), which comprises only code-related retrieval data.; and (3) Complex Instruction Retrieval, evaluated on FollowIR (Weller et al., 2024).

Compared Methods We compare our models with the most prominent open-source text embedding models and commercial API services. The open-source models include the GTE (Li et al., 2023; Zhang et al., 2024b), E5 (Wang et al., 2022), and BGE (Xiao et al., 2024) series, as well as NV-Embed-v2 (Lee et al., 2025a), GritLM-7B Muennighoff et al. (2025). The commercial APIs evaluated are text-embedding-3-large from OpenAI, Gemini-embedding from Google, and Cohere-embed-multilingual-v3.0. For reranking, we compare with the rerankers of jina[1], mGTE (Zhang et al., 2024b) and BGE-m3 (Chen et al., 2024).

4.2 Main Results

Embedding In Table 2, we present the evaluation results on MMTEB (Enevoldsen et al., 2025), which comprehensively covers a wide range of embedding tasks across multiple languages. Our Qwen3-Embedding-4B/8B models achieve the best performance, and our smallest model, Qwen3-Embedding-0.6B, only lags behind the best-performing baseline method (Gemini-Embedding), despite having only 0.6B parameters. In Table 3, we present the evaluation results on MTEB (English, v2) (Muennighoff et al., 2023), CMTEB (Xiao et al., 2024), and MTEB (Code) (Enevoldsen et al., 2025). The scores reflect similar trends as MMTEB, with our Qwen3-Embedding-4B/8B models

[1] https://hf.co/jinaai/jina-reranker-v2-base-multilingual

Model	Size	Dim	MTEB (Eng, v2)		CMTEB		MTEB (Code)
			Mean (Task)	Mean (Type)	Mean (Task)	Mean (Type)	
Selected Open-Source Models							
NV-Embed-v2	7B	4096	69.81	65.00	63.0	62.0	-
GritLM-7B	7B	4096	67.07	63.22	-	-	73.6[α]
multilingual-e5-large-instruct	0.6B	1024	65.53	61.21	-	-	65.0[α]
gte-Qwen2-1.5b-instruct	1.5B	1536	67.20	63.26	67.12	67.79	-
gte-Qwen2-7b-instruct	7B	3584	70.72	65.77	71.62	72.19	56.41[γ]
Commercial APIs							
text-embedding-3-large	-	3072	66.43	62.15	-	-	58.95[γ]
cohere-embed-multilingual-v3.0	-	1024	66.01	61.43	-	-	51.94[γ]
Gemini Embedding	-	3072	73.30	67.67	-	-	74.66[γ]
Qwen3 Embedding Models							
Qwen3-Embedding-0.6B	0.6B	1024	70.70	64.88	66.33	67.44	75.41
Qwen3-Embedding-4B	4B	2560	74.60	68.09	72.26	73.50	80.06
Qwen3-Embedding-8B	8B	4096	**75.22**	**68.70**	**73.83**	**75.00**	**80.68**

表3：MTEB英语、MTEB中文、MTEB代码的性能。[α]引自(Enevoldsen等，2025年)。[γ]引自(Lee等，2025b)。对于其他对比模型，分数来自2025年6月4日MTEB在线排行榜。

涵盖250多种语言。除了各种检索、分类和语义文本相似性等经典文本任务之外，MMTEB还包括一系列具有挑战性和新颖性的任务，如指令遵循、长文档检索和代码检索，代表了迄今为止针对嵌入模型的最大多语言评估任务集合。我们的MMTEB评估包含216个独立的评估任务，包括MTEB（多语言）（Enevoldsen等，2025）的131个任务，MTEB（英语，v2）（Muennighoff等，2023）的41个任务，CMTEB（Xiao等，2024）的32个任务，以及MTEB（代码）（Enevoldsen等，2025）的12个代码检索任务。

此外，我们选择了一系列文本检索任务来评估模型的文本重排序能力。我们探索了三类检索任务：(1) 基本相关性检索，分为英文、中文和多语言，分别在 MTEB (Muennighoff 等, 2023)、CMTEB (Xiao 等, 2024)、MMTEB (Enevoldsen 等, 2025) 和 MLDR (Chen 等, 2024) 上进行评估；(2) 代码检索，在 MTEB-Code (Enevoldsen 等, 2025) 上进行评估，该数据集仅包含与代码相关的检索数据；以及 (3) 复杂指令检索，在 FollowIR (Weller 等, 2024) 上进行评估。

对比方法 我们将我们的模型与最突出的开源文本嵌入模型和商业API服务进行比较。开源模型包括GTE（Li等，2023；Zhang等，2024b）、E5（Wang等，2022）和BGE（Xiao等，2024）系列，以及NV-Embed-v2（Lee等，2025a）、GritLM-7B（Muennighoff等，2025）。评估的商业API包括来自OpenAI的text-embedding-3-large、来自Google的Gemini-embedding以及Cohere-embed-multilingual-v3.0。在重排序方面，我们与jina[1]、mGTE（Zhang等，2024b）和BGE-m3（Chen等，2024）的重排序器进行比较。

4.2 主要结果

在表2中，我们展示了在MMTEB（Enevoldsen等，2025）上的评估结果，该基准全面覆盖了多语言环境下的多种嵌入任务。我们的Qwen3-Embedding-4B/8B模型取得了最佳性能，而我们最小的模型Qwen3-Embedding-0.6B仅以0.6B参数落后于表现最佳的基线方法（Gemini-Embedding）。在表3中，我们展示了在MTEB（英语，v2）（Muennighoff等，2023）、CMTEB（Xiao等，2024）和MTEB（Code）（Enevoldsen等，2025）上的评估结果。分数反映了与MMTEB相似的趋势，我们的Qwen3-Embedding-4B/8B模型

[1] https://hf.co/jinaai/jina-reranker-v2-base-multilingual

Model	Param	Basic Relevance Retrieval					
		MTEB-R	CMTEB-R	MMTEB-R	MLDR	MTEB-Code	FollowIR
Qwen3-Embedding-0.6B	0.6B	61.82	71.02	64.64	50.26	75.41	5.09
Jina-multilingual-reranker-v2-base	0.3B	58.22	63.37	63.73	39.66	58.98	-0.68
gte-multilingual-reranker-base	0.3B	59.51	74.08	59.44	66.33	54.18	-1.64
BGE-reranker-v2-m3	0.6B	57.03	72.16	58.36	59.51	41.38	-0.01
Qwen3-Reranker-0.6B	0.6B	65.80	71.31	66.36	67.28	73.42	5.41
Qwen3-Reranker-4B	4B	**69.76**	75.94	72.74	69.97	81.20	**14.84**
Qwen3-Reranker-8B	8B	69.02	**77.45**	**72.94**	**70.19**	**81.22**	8.05

Table 4: Evaluation results for reranking models. We use the retrieval subsets of MTEB(eng, v2), MTEB(cmn, v1) and MMTEB, which are MTEB-R, CMTEB-R and MMTEM-R. The rest are all retrieval tasks. All scores are our runs based on the retrieval top-100 results from the first row.

Model	MMTEB	MTEB (Eng, v2)	CMTEB	MTEB (Code, v1)
Qwen3-Embedding-0.6B w/ only synthetic data	58.49	60.63	59.78	66.79
Qwen3-Embedding-0.6B w/o synthetic data	61.21	65.59	63.37	74.58
Qwen3-Embedding-0.6B w/o model merge	62.56	68.18	64.76	74.89
Qwen3-Embedding-0.6B	64.33	70.70	66.33	75.41

Table 5: Performance (mean task) on MMTEB, MTEB(eng, v2), CMTEB and MTEB(code, v1) for Qwen3-Embedding-0.6B model with different training setting.

consistently outperforming others. Notably, the Qwen3-Embedding-0.6B model ranks just behind the Gemini-Embedding, while being competitive with the gte-Qwen2-7B-instruct.

Reranking In Table 4, we present the evaluation results on various reranking tasks (§4.1). We utilize the Qwen3-Embedding-0.6B model to retrieve the top-100 candidates and then apply different reranking models for further refinement. This approach ensures a fair evaluation of the reranking models. Our results indicate that all three Qwen3-Reranker models enhance performance compared to the embedding model and surpass all baseline reranking methods, with Qwen3-Reranker-8B achieving the highest performance across most tasks.

4.3 Analysis

To further analyze and explore the key elements of the Qwen3 Embedding model training framework, we conduct an analysis from the following dimensions:

Effectiveness of Large-Scale Weakly Supervised Pre-Training We first analyze the effectiveness of the large-scale weak supervised training stage for the embedding models. As shown in Table 5, the Qwen3-Embedding-0.6B model trained solely on synthetic data (without subsequent training stages, as indicated in the first row) achieves reasonable and strong performance compared to the final Qwen3-Embedding-0.6B model (as shown in the last row). If we further remove the weak supervised training stage (i.e., without synthetic data training, as seen in the second row), the final performance shows a clear decline. This indicates that the large-scale weak supervised training stage is crucial for achieving superior performance.

Effectiveness of Model Merging Next, we compare the performance differences arising from the model merging stage. As shown in Table 5, the model trained without model merging techniques (the third row, which uses data sampling to balance various tasks) performs considerably worse than the final Qwen3-Embedding-0.6B model (which employs model merging, as shown in the last row). This indicates that the model merging stage is also critical for developing strong models.

Model	Param	Basic Relevance Retrieval					
		MTEB-R	CMTEB-R	MMTEB-R	MLDR	MTEB-Code	FollowIR
Qwen3-Embedding-0.6B	0.6B	61.82	71.02	64.64	50.26	75.41	5.09
Jina-multilingual-reranker-v2-base	0.3B	58.22	63.37	63.73	39.66	58.98	-0.68
gte-multilingual-reranker-base	0.3B	59.51	74.08	59.44	66.33	54.18	-1.64
BGE-reranker-v2-m3	0.6B	57.03	72.16	58.36	59.51	41.38	-0.01
Qwen3-Reranker-0.6B	0.6B	65.80	71.31	66.36	67.28	73.42	5.41
Qwen3-Reranker-4B	4B	**69.76**	75.94	72.74	69.97	81.20	**14.84**
Qwen3-Reranker-8B	8B	69.02	**77.45**	**72.94**	**70.19**	**81.22**	8.05

表4：重排序模型的评估结果。我们使用了MTEB(eng, v2)、MTEB(cmn, v1)和MMTEB的检索子集，分别为MTEB-R、CMTEB-R和MMTEM-R。其余均为检索任务。所有分数均基于第一行的检索前100结果进行计算。

Model	MMTEB	MTEB (Eng, v2)	CMTEB	MTEB (Code, v1)
Qwen3-Embedding-0.6B w/ only synthetic data	58.49	60.63	59.78	66.79
Qwen3-Embedding-0.6B w/o synthetic data	61.21	65.59	63.37	74.58
Qwen3-Embedding-0.6B w/o model merge	62.56	68.18	64.76	74.89
Qwen3-Embedding-0.6B	64.33	70.70	66.33	75.41

表5：Qwen3-Embedding-0.6B模型在不同训练设置下，MMTEB、MTEB(eng, v2)、CMTEB和MTEB(code, v1)上的性能（平均任务）

始终表现优于其他模型。值得注意的是，Qwen3-Embedding-0.6B模型排名紧随Gemini-Embedding之后，同时与gte-Qwen2-7B-instruct具有竞争力。

在表4中，我们展示了在各种重排序任务上的评估结果（§4.1）。我们使用Qwen3-Embedding-0.6B模型检索前100个候选结果，然后应用不同的重排序模型进行进一步优化。这种方法确保了对重排序模型的公平评估。我们的结果表明，所有三个Qwen3-Reranker模型相比嵌入模型提升了性能，并超越了所有基线重排序方法，其中Qwen3-Reranker-8B在大多数任务中实现了最高性能。

4.3 分析

为了进一步分析和探索Qwen3嵌入模型训练框架的关键要素，我们从以下维度进行分析：

大规模弱监督预训练的有效性 我们首先分析嵌入模型中大规模弱监督训练阶段的有效性。如表5所示，仅在合成数据上训练的Qwen3-Embedding-0.6B模型（未进行后续训练阶段，见第一行）相比最终的Qwen3-Embedding-0.6B模型（见最后一行）表现出合理且强劲的性能。如果进一步移除弱监督训练阶段（即不进行合成数据训练，见第二行），最终性能会明显下降。这表明大规模弱监督训练阶段对于实现优越性能至关重要。

模型合并的有效性 接下来，我们比较模型合并阶段带来的性能差异。如表5所示，未采用模型合并技术的模型（第三行，使用数据采样平衡各项任务）的表现明显差于最终的Qwen3-Embedding-0.6B模型（最后一行，采用了模型合并技术）。这表明模型合并阶段对于开发高性能模型同样至关重要。

5 Conclusion

In this technical report, we present the Qwen3-Embedding series, a comprehensive suite of text embedding and reranking models based on the Qwen3 foundation models. These models are designed to excel in a wide range of text embedding and reranking tasks, including multilingual retrieval, code retrieval, and complex instruction following. The Qwen3-Embedding models are built upon a robust multi-stage training pipeline that combines large-scale weakly supervised pre-training on synthetic data with supervised fine-tuning and model merging on high-quality datasets. The Qwen3 LLMs play a crucial role in synthesizing diverse training data across multiple languages and tasks, thereby enhancing the models' capabilities. Our comprehensive evaluations demonstrate that the Qwen3-Embedding models achieve state-of-the-art performance across various benchmarks, including MTEB, CMTEB, MMTEB, and several retrieval benchmarks. We are pleased to open-source the Qwen3-Embedding and Qwen3-Reranker models (0.6B, 4B, and 8B), making them available for the community to use and build upon.

References

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 2318–2335, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.findings-acl.137/.

Kenneth Enevoldsen, Isaac Chung, Imene Kerboua, Márton Kardos, Ashwin Mathur, David Stap, Jay Gala, Wissam Siblini, Dominik Krzemiński, Genta Indra Winata, et al. MMTEB: Massive multilingual text embedding benchmark. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=zl3pfz4VCV.

Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. Scaling synthetic data creation with 1,000,000,000 personas. *arXiv preprint arXiv:2406.20094*, 2024.

Jui-Ting Huang, Ashish Sharma, Shuying Sun, Li Xia, David Zhang, Philip Pronin, Janani Padmanabhan, Giuseppe Ottaviano, and Linjun Yang. Embedding-based retrieval in facebook search. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2553–2561, 2020.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *EMNLP (1)*, pp. 6769–6781, 2020.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428*, 2024.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. NV-embed: Improved techniques for training LLMs as generalist embedding models. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=lgsyLSsDRe.

Jinhyuk Lee, Feiyang Chen, Sahil Dua, Daniel Cer, Madhuri Shanbhogue, Iftekhar Naim, Gustavo Hernández Ábrego, Zhe Li, Kaifeng Chen, Henrique Schechter Vera, et al. Gemini embedding: Generalizable embeddings from gemini. *arXiv preprint arXiv:2503.07891*, 2025b.

5 结论

在本技术报告中，我们介绍了基于Qwen3基础模型的Qwen3-Embedding系列，这是一套全面的文本嵌入和重排序模型。这些模型旨在在广泛的文本嵌入和重排序任务中表现出色，包括多语言检索、代码检索和复杂指令遵循。Qwen3-Embedding模型基于一个强大的多阶段训练流水线构建，该流水线结合了在合成数据上的大规模弱监督预训练，以及在高质量数据集上的监督微调和模型合并。Qwen3大语言模型在合成多种语言和任务的多样化训练数据方面发挥着关键作用，从而增强了模型的能力。我们的全面评估表明，Qwen3-Embedding模型在各种基准测试中实现了最先进的性能，包括MTEB、CMTEB、MMTEB以及多个检索基准。我们很高兴开源Qwen3-Embedding和Qwen3-Reranker模型（0.6B、4B和8B），供社区使用和进一步开发。

参考文献

陈建律、肖世涛、张培田、罗坤、连德福和刘铮。M3-embedding：通过自知识蒸馏实现多语言性、多功能性、多粒度文本嵌入。发表于
Findings of the Association for Computational Linguistics: ACL 2024，第2318–2335页，泰国曼谷，2024年8月。计算语言学协会。URL https:// aclanthology.org/2024.findings-acl.137/。肯尼斯·埃内沃尔森、伊萨克·钟、伊梅内·科尔布瓦、Márton Kardos、阿什温·马图尔、戴维·斯塔普、杰伊·加拉、威萨姆·西布利尼、多米尼克·克热米斯基、根塔·因德拉·温塔等。MMTEB：大规模多语言文本嵌入基准。发表于*The Thirteenth International Conference on Learning Representations*，2025年。URL https://openreview.net/forum?id=zl3pfz4VCV。葛涛、陈欣、王小阳、于典、米海涛和余东。通过10亿个角色扩展合成数据生成。*arXiv preprint arXiv:2406.20094*，2024年。黄瑞廷、阿什希什·沙玛、孙书颖、李霞、张大卫、菲利普·普罗宁、贾纳尼·帕德马纳巴汉、吉乌塞佩·奥塔瓦尼奥和杨林君。基于嵌入的Facebook搜索检索。发表于
Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining，第2553–2561页，2020年。阿伦·赫斯特、亚当·莱勒、亚当·P·戈彻、亚当·佩尔曼、阿迪亚·拉梅什、艾丹·克拉克、AJ·奥斯特罗、阿基拉·韦利欣达、艾伦·海斯、亚历克斯·拉德福德等。GPT-4o系统卡。*arXiv preprint arXiv:2410.21276*，2024年。弗拉基米尔·卡尔普欣、巴拉斯·奥古兹、肖恩·敏、帕特里克·SH·刘易斯、李德尔·吴、谢尔盖·埃杜诺夫、丹琪·陈和温-陶·易。开放域问答的密集段落检索。发表于*EMNLP (1)*，第6769–6781页，2020年。李灿奎、拉贾什·罗伊、徐梦瑶、乔纳森·雷曼、穆罕默德·肖埃比、布莱恩·卡坦扎罗和魏平。NV-embed：改进训练大型语言模型作为通用嵌入模型的技术。*arXiv preprint arXiv:2405.17428*，2024年。李灿奎、拉贾什·罗伊、徐梦瑶、乔纳森·雷曼、穆罕默德·肖埃比、布莱恩·卡坦扎罗和魏平。NV-embed：改进训练大型语言模型作为通用嵌入模型的技术。发表于*The Thirteenth International Conference on Learning Representations*，2025a。URL https://openreview.net/forum?id=lgsyLSsDRe。李金昱、陈飞阳、萨希尔·杜阿、丹尼尔·塞勒、马杜里·尚博戈、伊夫特哈尔·奈姆、Gustavo Hernández 阿布雷戈、李哲、陈凯峰、亨里克·谢克特·维拉等。Gemini嵌入：来自Gemini的通用嵌入。*arXiv preprint arXiv:2503.07891*，2025b。

Mingxin Li, Zhijie Nie, Yanzhao Zhang, Dingkun Long, Richong Zhang, and Pengjun Xie. Improving general text embedding model: Tackling task conflict and data imbalance through model merging. *arXiv preprint arXiv:2410.15035*, 2024.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning, 2023. URL https://arxiv.org/abs/2308.03281.

Xueguang Ma, Xinyu Zhang, Ronak Pradeep, and Jimmy Lin. Zero-shot listwise document reranking with a large language model. *arXiv preprint arXiv:2305.02156*, 2023.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. MTEB: Massive text embedding benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 2014–2037, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.eacl-main.148/.

Niklas Muennighoff, Hongjin SU, Liang Wang, Nan Yang, Furu Wei, Tao Yu, Amanpreet Singh, and Douwe Kiela. Generative representational instruction tuning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=BC4lIvfSzv.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. Rankvicuna: Zero-shot listwise document reranking with open-source large language models. *arXiv preprint arXiv:2309.15088*, 2023.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. URL https://aclanthology.org/D19-1410/.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A Smith, Luke Zettlemoyer, and Tao Yu. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 1102–1121, 2023.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training, 2022. URL https://arxiv.org/abs/2212.03533.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Improving text embeddings with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11897–11916, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.acl-long.642/.

Orion Weller, Benjamin Chang, Sean MacAvaney, Kyle Lo, Arman Cohan, Benjamin Van Durme, Dawn Lawrie, and Luca Soldaini. Followir: Evaluating and teaching information retrieval models to follow instructions. *arXiv preprint arXiv:2403.15246*, 2024.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, pp. 641–649, New York, NY, USA, 2024. Association for Computing Machinery. URL https://doi.org/10.1145/3626772.3657878.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

李明新、聂志杰、张延昭、龙定坤、张日聪和谢鹏君。改进通用文本嵌入模型：通过模型合并解决任务冲突和数据不平衡。*arXiv preprint arXiv:2410.15035*, 2024。李泽涵、张欣、张延昭、龙定坤、谢鹏君和张美山。通过多阶段对比学习实现通用文本嵌入，2023。URL https://arxiv.org/abs/2308.03281。马学光、张欣宇、Ronak Pradeep和Jimmy Lin。使用大语言模型进行零样本列表级文档重排序。*arXiv preprint arXiv:2305.02156*, 2023。Niklas Muennighoff、Nouamane Tazi、Loic Magne和Nils Reimers。MTEB：大规模文本嵌入基准测试。发表于
Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, 第2014–2037页，克罗地亚杜布罗夫尼克，2023年5月。计算语言学协会。URL https://aclanthology.org/2023.eacl-main.148/。Niklas Muennighoff、苏宏金、王良、杨楠、魏福瑞、Yu Tao、Amanpreet Singh和Douwe Kiela。生成式表示指令微调。发表于
The Thirteenth International Con- ference on Learning Representations, 2025。URL https://openreview.net/forum?id=BC4lIvfSzv。Aaron van den Oord、李雅哲和Oriol Vinyals。使用对比预测编码的表示学习。*arXiv preprint arXiv:1807.03748*, 2018。Ronak Pradeep、Sahel Sharifymoghaddam和Jimmy Lin。Rankvicuna：使用开源大语言模型进行零样本列表级文档重排序。*arXiv preprint arXiv:2309.15088*, 2023。Nils Reimers和Iryna Gurevych。Sentence-BERT：使用孪生BERT网络的句子嵌入。发表于
Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 第3982–3992页，中国香港，2019年11月。计算语言学协会。URL https://aclanthology.org/D19-1410/。苏宏金、石伟佳、Jungo Kasai、王义中、胡宇石、Mari Ostendorf、Wen-tau Yih、Noah A Smith、Luke Zettlemoyer和Yu Tao。一个嵌入器，任意任务：指令微调文本嵌入。发表于*Findings of the Association for Computational Linguistics: ACL 2023*, 第1102–1121页，2023。王良、杨楠、黄小龙、焦斌星、杨林君、蒋大鑫、Rangan Majumder和魏福瑞。通过弱监督对比预训练的文本嵌入，2022。URL https://arxiv.org/abs/2212.03533。王良、杨楠、黄小龙、杨林君、Rangan Majumder和魏福瑞。使用大语言模型改进文本嵌入。发表于
Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 第11897–11916页，泰国曼谷，2024年8月。计算语言学协会。URL https://aclanthology.org/2024.acl-long.642/。Orion Weller、Benjamin Chang、Sean MacAvaney、Kyle Lo、Arman Cohan、Benjamin Van Durme、Dawn Lawrie和Luca Soldaini。Followir：评估和训练信息检索模型遵循指令。
arXiv preprint arXiv:2403.15246, 2024。肖世涛、刘铮、张沛田、Niklas Muennighoff、连德福和聂建云。C-pack：通用中文嵌入的打包资源。发表于*Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, 第641–649页，美国纽约，2024。计算机协会。URL https://doi.org/10.1145/3626772.3657878。杨安、李安丰、杨宝松、张宝臣、回斌元、郑博、于博文、高昌、黄成根、吕晨旭等。Qwen3技术报告。
arXiv preprint arXiv:2505.09388, 2025。

Longhui Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, Meishan Zhang, and Min Zhang. A two-stage adaptation of large language models for text ranking. In *Findings of the Association for Computational Linguistics ACL 2024*, pp. 11880–11891, 2024a.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, Meishan Zhang, Wenjie Li, and Min Zhang. mGTE: Generalized long-context text representation and reranking models for multilingual text retrieval. In Franck Dernoncourt, Daniel Preoţiuc-Pietro, and Anastasia Shimorina (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 1393–1412, Miami, Florida, US, November 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-industry.103. URL https://aclanthology.org/2024.emnlp-industry.103/.

Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji-Rong Wen. Dense text retrieval based on pretrained language models: A survey. *ACM Transactions on Information Systems*, 42(4):1–60, 2024.

Xiangyu Zhao, Maolin Wang, Xinjian Zhao, Jiansheng Li, Shucheng Zhou, Dawei Yin, Qing Li, Jiliang Tang, and Ruocheng Guo. Embedding in recommender systems: A survey. *arXiv preprint arXiv:2310.18608*, 2023.

Shengyao Zhuang, Honglei Zhuang, Bevan Koopman, and Guido Zuccon. A setwise approach for effective and highly efficient zero-shot ranking with large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 38–47, 2024.

张龙辉、张燕昭、龙定坤、谢鹏君、张美山和张敏。大型语言模型的两阶段适应用于文本排序。在 *Findings of the Association for Computational Linguistics ACL 2024*，第11880–11891页，2024a。

X张燕昭、丁坤龙、谢文、戴子琪、唐嘉龙、林欢、杨宝松、谢鹏君、黄飞、张美山、李文杰和张敏。mGTE：面向多语言文本检索的通用长上下文文本表示与重排序模型。弗兰克·德农库尔、丹尼尔·普雷奥伊乌-皮埃特罗和阿纳斯塔西娅·希莫里娜（编），*Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*，第1393–1412页，美国佛罗里达州迈阿密，2024年11月b。计算语言学协会。doi: 10.18653/ v1/2024.emnlp-industry.103。URL https://aclanthology.org/2024.emnlp-industry.103/。

Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji-Rong Wen. 基于预训练语言模型的密集文本检索：综述. *ACM Transactions on Information Systems*, 42(4):1–60, 2024.

Xiangyu Zhao, Maolin Wang, Xinjian Zhao, Jiansheng Li, Shucheng Zhou, Dawei Yin, Qing Li, Jiliang Tang, and Ruocheng Guo. 推荐系统中的嵌入: 一项综述。*arXiv preprint arXiv:2310.18608*, 2023.

庄盛尧，庄宏磊，Bevan Koopman，和Guido Zuccon。一种集合式方法用于在大型语言模型中实现有效且高效的零样本排序。发表于 *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 第38–47页, 2024。

A Appendix

A.1 Synthetic Data

We construct four types of synthetic data—retrieval, bitext mining, semantic textual similarity, and classification to enable the model to adapt to various similarity tasks during pre-training. To ensure both multilingual and cross-lingual diversity, the data is generated using Qwen3 32B. Below is an example of a synthetic retrieval text pair. The retrieval data is synthesized using a document-to-query approach. We collect a multilingual corpus from the pre-training corpus of the Qwen3 base model to serve as the document source. A two-stage generation pipeline is then applied, consisting of: (1) configuration and (2) query generation. In the configuration stage, we use large language models (LLMs) to determine the "Question Type", "Difficulty", and "Character" for the synthetic query. The candidate characters are retrieved from Persona Hub (Ge et al., 2024), selecting the top five most relevant to the given document. This step aims to enhance the diversity of the generated queries. The template used is as follows:

```
Given a **Passage** and **Character**, select the appropriate option from three
 ↪  fields: Character, Question_Type, Difficulty, and return the output in JSON
 ↪  format.
First, select the Character who are likely to be interested in the Passage from
 ↪  the candidates. Then select the Question_Type that the Character might ask
 ↪  about the Passage; Finally, choose the Difficulty of the possible question
 ↪  based on the Passage, the Character, and the Question_Type.
Character: Given by input **Character**

Question_Type:
- keywords: ...
- acquire_knowledge: ...
- summary: ...
- yes_or_no: ...
- background: ...

Difficulty:
- high_school: ...
- university: ...
- phd: ...

Here are some examples
<Example1> <Example2> <Example3>

Now, generate the **output** based on the **Passage** and **Character** from
 ↪  user, the **Passage** will be in {language} language and the **Character**
 ↪  will be in English.
Ensure to generate only the JSON output with content in English.

**Passage**:
{passage}
**Character**:
{character}
```

In the query generation stage, we use the configuration selected in the first stage to guide the generation of queries. Additionally, we explicitly specify the desired length and language of the generated query. The template used is as follows:

附录A

A.1 合成数据

我们构建了四种类型的合成数据——检索、双语文本挖掘、语义文本相似性以及分类，以使模型在预训练过程中能够适应各种相似性任务。为了确保多语言和跨语言的多样性，数据是使用Qwen3 32B生成的。以下是一个合成检索文本对的示例。检索数据通过文档到查询方法合成。我们从Qwen3基础模型的预训练语料库中收集多语言语料库作为文档来源。随后应用两阶段生成流程，包括：(1) 配置和(2) 查询生成。在配置阶段，我们使用大语言模型（LLMs）来确定合成查询的“问题类型”、“难度”和“角色”。候选角色从Persona Hub（Ge等，2024）中检索，选择与给定文档最相关的前五个。此步骤旨在增强生成查询的多样性。使用的模板如下：

```
Given a **Passage** and **Character**, select the appropriate option from three
 ↪  fields: Character, Question_Type, Difficulty, and return the output in JSON
 ↪  format.
First, select the Character who are likely to be interested in the Passage from
 ↪  the candidates. Then select the Question_Type that the Character might ask
 ↪  about the Passage; Finally, choose the Difficulty of the possible question
 ↪  based on the Passage, the Character, and the Question_Type.
Character: Given by input **Character**

Question_Type:
- keywords: ...
- acquire_knowledge: ...
- summary: ...
- yes_or_no: ...
- background: ...

Difficulty:
- high_school: ...
- university: ...
- phd: ...

Here are some examples
<Example1> <Example2> <Example3>

Now, generate the **output** based on the **Passage** and **Character** from
 ↪  user, the **Passage** will be in {language} language and the **Character**
 ↪  will be in English.
Ensure to generate only the JSON output with content in English.

**Passage**:
{passage}
**Character**:
{character}
```

在查询生成阶段，我们使用第一阶段中选择的配置来指导查询的生成。此外，我们明确指定了生成查询的期望长度和语言。使用的模板如下：

```
Given a **Character**, **Passage**, and **Requirement**, generate a query from
 ↪  the **Character**'s perspective that satisfies the **Requirement** and can
 ↪  be used to retrieve the **Passage**. Please return the result in JSON
 ↪  format.

Here is an example:
<example>

Now, generate the **output** based on the **Character**, **Passage** and
 ↪  **Requirement** from user, the **Passage** will be in {corpus_language}
 ↪  language, the **Character** and **Requirement** will be in English.
Ensure to generate only the JSON output, with the key in English and the value
 ↪  in {queries_language} language.

**Character**
{character}
**Passage**
{passage}
**Requirment**
- Type: {type};
- Difficulty: {difficulty};
- Length: the length of the generated sentences should be {length} words;
- Languange: the language in which the results are generated should be
 ↪  {language} language;
```

Stage	Dataset	Size
Weakly Supervised Pre-Training	Synthetic Data	$\sim$ 150M
Supervised Fine Tuning	MS MARCO, NQ, HotpotQA, NLI, Dureader, T^2-Ranking, SimCLUE, MIRACL, MLDR, Mr.TyDi, Multi-CPR, CodeSearchNet .etc + High-quality Synthetic Data	Labeled Data: $\sim$ 7M Synthetic Data: $\sim$ 12M

Table 6: Statistics of training data utilized at each stage.

A.2 Detail Results

MTEB(eng, v2)	Param	Mean (Task)	Mean (Type)	Class-ification	Clus-tering	Pair Class.	Rerank	Retrieval	STS	Summ.
multilingual-e5-large-instruct	0.6B	65.53	61.21	75.54	49.89	86.24	48.74	53.47	84.72	29.89
NV-Embed-v2	7.8B	69.81	65.00	87.19	47.66	88.69	49.61	62.84	83.82	35.21
GritLM-7B	7.2B	67.07	63.22	81.25	50.82	87.29	49.59	54.95	83.03	35.65
gte-Qwen2-1.5B-instruct	1.5B	67.20	63.26	85.84	53.54	87.52	49.25	50.25	82.51	33.94
stella_en_1.5B_v5	1.5B	69.43	65.32	89.38	57.06	88.02	50.19	52.42	83.27	36.91
gte-Qwen2-7B-instruct	7.6B	70.72	65.77	88.52	58.97	85.9	50.47	58.09	82.69	35.74
gemini-embedding-exp-03-07	-	73.3	67.67	90.05	59.39	87.7	48.59	64.35	85.29	38.28
Qwen3-Embedding-0.6B	0.6B	70.70	64.88	85.76	54.05	84.37	48.18	61.83	86.57	33.43
Qwen3-Embedding-4B	4B	74.60	68.09	89.84	57.51	87.01	50.76	68.46	88.72	34.39
Qwen3-Embedding-8B	8B	75.22	68.70	90.43	58.57	87.52	51.56	69.44	88.58	34.83

Table 7: Results on MTEB(eng, v2) (Muennighoff et al., 2023). We compare models from the online leaderboard.

```
Given a **Character**, **Passage**, and **Requirement**, generate a query from
 ↪  the **Character**'s perspective that satisfies the **Requirement** and can
 ↪  be used to retrieve the **Passage**. Please return the result in JSON
 ↪  format.

Here is an example:
<example>

Now, generate the **output** based on the **Character**, **Passage** and
 ↪  **Requirement** from user, the **Passage** will be in {corpus_language}
 ↪  language, the **Character** and **Requirement** will be in English.
Ensure to generate only the JSON output, with the key in English and the value
 ↪  in {queries_language} language.

**Character**
{character}
**Passage**
{passage}
**Requirment**
- Type: {type};
- Difficulty: {difficulty};
- Length: the length of the generated sentences should be {length} words;
- Languange: the language in which the results are generated should be
 ↪  {language} language;
```

Stage	Dataset	Size
Weakly Supervised Pre-Training	Synthetic Data	$\sim$ 150M
Supervised Fine Tuning	MS MARCO, NQ, HotpotQA, NLI, Dureader, T^2-Ranking, SimCLUE, MIRACL, MLDR, Mr.TyDi, Multi-CPR, CodeSearchNet .etc + High-quality Synthetic Data	Labeled Data: $\sim$ 7M Synthetic Data: $\sim$ 12M

表6：每个阶段所使用的训练数据统计。

A.2 详细结果

MTEB(eng, v2)	Param	Mean (Task)	Mean (Type)	Class-ification	Clus-tering	Pair Class.	Rerank	Retrieval	STS	Summ.
multilingual-e5-large-instruct	0.6B	65.53	61.21	75.54	49.89	86.24	48.74	53.47	84.72	29.89
NV-Embed-v2	7.8B	69.81	65.00	87.19	47.66	88.69	49.61	62.84	83.82	35.21
GritLM-7B	7.2B	67.07	63.22	81.25	50.82	87.29	49.59	54.95	83.03	35.65
gte-Qwen2-1.5B-instruct	1.5B	67.20	63.26	85.84	53.54	87.52	49.25	50.25	82.51	33.94
stella_en_1.5B_v5	1.5B	69.43	65.32	89.38	57.06	88.02	50.19	52.42	83.27	36.91
gte-Qwen2-7B-instruct	7.6B	70.72	65.77	88.52	58.97	85.9	50.47	58.09	82.69	35.74
gemini-embedding-exp-03-07	-	73.3	67.67	90.05	59.39	87.7	48.59	64.35	85.29	38.28
Qwen3-Embedding-0.6B	0.6B	70.70	64.88	85.76	54.05	84.37	48.18	61.83	86.57	33.43
Qwen3-Embedding-4B	4B	74.60	68.09	89.84	57.51	87.01	50.76	68.46	88.72	34.39
Qwen3-Embedding-8B	8B	75.22	68.70	90.43	58.57	87.52	51.56	69.44	88.58	34.83

表7：MTEB(eng, v2)上的结果 (Muennighoff 等, 2023)。我们比较了来自在线排行榜的模型。

MTEB(cmn, v1)	**Param**	**Mean (Task)**	**Mean (Type)**	Class-ification	Clus-tering	Pair Class.	Rerank	Retrieval	STS
multilingual-e5-large-instruct	0.6B	58.08	58.24	69.80	48.23	64.52	57.45	63.65	45.81
gte-Qwen2-7B-instruct	7.6B	71.62	72.19	75.77	66.06	81.16	69.24	75.70	65.20
gte-Qwen2-1.5B-instruct	1.5B	67.12	67.79	72.53	54.61	79.5	68.21	71.86	60.05
Qwen3-Embedding-0.6B	0.6B	66.33	67.44	71.40	68.74	76.42	62.58	71.03	54.52
Qwen3-Embedding-4B	4B	72.26	73.50	75.46	77.89	83.34	66.05	77.03	61.26
Qwen3-Embedding-8B	8B	73.84	75.00	76.97	80.08	84.23	66.99	78.21	63.53

Table 8: Results on C-MTEB (Xiao et al., 2024) (MTEB(cmn, v1).

MTEB(Code, v1)	Avg.	Apps	COIR-CodeSearch-Net	Code-Edit-Search	Code-Feedback-MT	Code-Feedback-ST	Code-SearchNet-CCR	Code-SearchNet	Code-Trans-Ocean-Contest	Code-Trans-Ocean-DL	CosQA	Stack-Overflow-QA	Synthetic-Text2SQL
$BGE_{multilingual}$	62.04	22.93	68.14	60.48	60.52	76.70	73.23	83.43	86.84	32.64	27.93	92.93	58.67
NV-Embed-v2	63.74	29.72	61.85	73.96	60.27	81.72	68.82	86.61	89.14	33.40	34.82	92.36	60.90
gte-Qwen2-7B-instruct	62.17	28.39	71.79	67.06	57.66	85.15	66.24	86.96	81.83	32.17	31.26	84.34	53.22
gte-Qwen2-1.5B-instruct	61.98	28.91	71.56	59.60	49.92	81.92	72.08	91.08	79.02	32.73	32.23	90.27	54.49
BGE-M3 (Dense)	58.22	14.77	58.07	59.83	47.86	69.27	53.55	61.98	86.22	29.37	27.36	80.71	49.65
Jina-v3	58.85	28.99	67.83	57.24	59.66	78.13	54.17	85.50	77.37	30.91	35.15	90.79	41.49
Qwen3-Embedding-0.6B	75.41	75.34	84.69	64.42	90.82	86.39	91.72	91.01	86.05	31.36	36.48	89.99	76.74
Qwen3-Embedding-4B	80.06	89.18	87.93	76.49	93.21	89.51	95.59	92.34	90.99	35.04	37.98	94.32	78.21
Qwen3-Embedding-8B	80.68	91.07	89.51	76.97	93.70	89.93	96.35	92.66	93.73	32.81	38.04	94.75	78.75
Qwen3-Reranker-0.6B	73.42	69.43	85.09	72.37	83.83	78.05	94.76	88.8	84.69	33.94	36.83	93.24	62.48
Qwen3-Reranker-4B	81.20	94.25	90.91	82.53	95.25	88.54	97.58	92.48	93.66	36.78	35.14	97.11	75.06
Qwen3-Reranker-8B	81.22	94.55	91.88	84.58	95.64	88.43	95.67	92.78	90.83	34.89	37.43	97.3	73.4

Table 9: Performance on `MTEB(Code, v1)` (Enevoldsen et al., 2025). We report nDCG@10 scores.

MTEB(cmn, v1)	**Param**	**Mean (Task)**	**Mean (Type)**	Class-ification	Clus-tering	Pair Class.	Rerank	Retrieval	STS
multilingual-e5-large-instruct	0.6B	58.08	58.24	69.80	48.23	64.52	57.45	63.65	45.81
gte-Qwen2-7B-instruct	7.6B	71.62	72.19	75.77	66.06	81.16	69.24	75.70	65.20
gte-Qwen2-1.5B-instruct	1.5B	67.12	67.79	72.53	54.61	79.5	68.21	71.86	60.05
Qwen3-Embedding-0.6B	0.6B	66.33	67.44	71.40	68.74	76.42	62.58	71.03	54.52
Qwen3-Embedding-4B	4B	72.26	73.50	75.46	77.89	83.34	66.05	77.03	61.26
Qwen3-Embedding-8B	8B	73.84	75.00	76.97	80.08	84.23	66.99	78.21	63.53

表8：C-MTEB上的结果 (Xiao等, 2024) (MTEB(cmn, v1).

MTEB(Code, v1)	Avg.	Apps	COIR-CodeSearch-Net	Code-Edit-Search	Code-Feedback-MT	Code-Feedback-ST	Code-SearchNet-CCR	Code-SearchNet	Code-Trans-Ocean-Contest	Code-Trans-Ocean-DL	CosQA	Stack-Overflow-QA	Synthetic-Text2SQL
$BGE_{multilingual}$	62.04	22.93	68.14	60.48	60.52	76.70	73.23	83.43	86.84	32.64	27.93	92.93	58.67
NV-Embed-v2	63.74	29.72	61.85	73.96	60.27	81.72	68.82	86.61	89.14	33.40	34.82	92.36	60.90
gte-Qwen2-7B-instruct	62.17	28.39	71.79	67.06	57.66	85.15	66.24	86.96	81.83	32.17	31.26	84.34	53.22
gte-Qwen2-1.5B-instruct	61.98	28.91	71.56	59.60	49.92	81.92	72.08	91.08	79.02	32.73	32.23	90.27	54.49
BGE-M3 (Dense)	58.22	14.77	58.07	59.83	47.86	69.27	53.55	61.98	86.22	29.37	27.36	80.71	49.65
Jina-v3	58.85	28.99	67.83	57.24	59.66	78.13	54.17	85.50	77.37	30.91	35.15	90.79	41.49
Qwen3-Embedding-0.6B	75.41	75.34	84.69	64.42	90.82	86.39	91.72	91.01	86.05	31.36	36.48	89.99	76.74
Qwen3-Embedding-4B	80.06	89.18	87.93	76.49	93.21	89.51	95.59	92.34	90.99	35.04	37.98	94.32	78.21
Qwen3-Embedding-8B	80.68	91.07	89.51	76.97	93.70	89.93	96.35	92.66	93.73	32.81	38.04	94.75	78.75
Qwen3-Reranker-0.6B	73.42	69.43	85.09	72.37	83.83	78.05	94.76	88.8	84.69	33.94	36.83	93.24	62.48
Qwen3-Reranker-4B	81.20	94.25	90.91	82.53	95.25	88.54	97.58	92.48	93.66	36.78	35.14	97.11	75.06
Qwen3-Reranker-8B	81.22	94.55	91.88	84.58	95.64	88.43	95.67	92.78	90.83	34.89	37.43	97.3	73.4

表9：在 `MTEB(Code, v1)` (Enevoldsen 等人，2025) 上的表现。我们报告了 nDCG@10 分数。