**1. prepare\_data.py**

**功能说明**：数据预处理脚本。用于从原始对话数据（data/raw\_data.jsonl）中提取有效的问答对，进行清洗（如繁体转简体、长度过滤），并按 8:1:1 划分为训练集、验证集和测试集，分别保存为 data/train.jsonl、data/val.jsonl、data/test.jsonl。**主要流程**：

* 读取原始数据，提取 user-assistant 问答对，清洗内容。
* 按比例划分数据集。
* 转换为 InputExample 格式，便于后续模型训练。
* 输出各数据集样本数，并保存为 jsonl 文件。

**2. gen\_hard\_negatives\_by\_api.py**

**功能说明**：通过调用大模型 API，为每个问答对自动生成“难负样本”（hard negatives），用于增强检索模型的评测难度。支持断点续跑和速率控制。**主要流程**：

* 读取测试集问答对，检查已生成缓存，避免重复。
* 构造 prompt，调用 API 生成 hard negatives。
* 速率限制（RPM、TPM）和并发控制，保证调用稳定。
* 解析 API 返回内容，提取 hard negatives。
* 结果缓存与最终合并输出到 data/test\_hard\_negatives.jsonl。

**3. train\_bi\_encoder.py**

**功能说明**：训练双塔（Bi-Encoder）模型，用于问答匹配或语义检索任务。**主要流程**：

* 加载训练集和验证集样本。
* 初始化 SentenceTransformer 模型。
* 构建 DataLoader，定义损失函数（MultipleNegativesRankingLoss）。
* 构建 IR 评估器（InformationRetrievalEvaluator），用于训练过程中的效果评估。
* 设置训练参数（学习率、batch size、epoch、权重衰减等）。
* 启动训练，保存最优模型到指定路径。
* 支持 TensorBoard 日志记录。

**4. test\_bi\_encoder.py**

**功能说明**：评测微调前后 Bi-Encoder 模型的检索性能，并输出对比结果。**主要流程**：

* 读取测试集（支持 easy/hard 两种模式）。
* 加载微调前后模型，分别计算问题和答案的 embedding。
* 计算相似度，统计 Recall@k、MRR@k、平均排名等指标。
* 结果保存为 result\_compare.json 和 csv 文件，便于后续分析。

**5. test\_api\_embedding.py**

**功能说明**：调用外部 API（如 OpenAI、GLM 等）获取 embedding，评测不同 API embedding 的检索效果，并与本地模型对比。**主要流程**：

* 读取测试集，支持 easy/hard 模式。
* 支持多种 API embedding 模型，自动缓存 embedding。
* 评测各 API embedding 的检索指标（Recall@k、MRR@k、平均排名）。
* 结果保存为 json 和 csv，便于横向对比。

**6. test\_bi\_encoder\_with\_rerank.py**

**功能说明**：评测“Bi-Encoder 检索 + CrossEncoder 重排序”方案的效果。**主要流程**：

* 先用 Bi-Encoder 检索 top50，再用 CrossEncoder 对 top50 进行 rerank。
* 统计正样本在 rerank 后的排名，计算 Recall@k、MRR@k 等指标。
* 结果保存为 json 和 csv 文件。

**7. test\_bm25.py**

**功能说明**：基于 BM25 算法的检索基线评测。**主要流程**：

* 读取测试集，分词（优先用 jieba）。
* 用 BM25Okapi 构建索引，计算每个问题的检索排名。
* 统计 Recall@k、MRR@k、平均排名等指标。
* 结果保存为 json 和 csv 文件。

**8. test\_bm25\_bi\_encoder\_hybrid.py**

**功能说明**：评测 BM25 与 Bi-Encoder 加权融合的检索效果。**主要流程**：

* BM25 和 Bi-Encoder 分别计算分数，归一化后加权融合。
* 检索 top50，统计正样本排名，计算各项指标。
* 结果保存为 json 和 csv 文件。

**9. test\_bm25\_bi\_encoder\_hybrid\_with\_rerank.py**

**功能说明**：评测“BM25+Bi-Encoder 融合检索 + CrossEncoder 重排序”方案。**主要流程**：

* BM25+Bi-Encoder 融合分数，检索 top50。
* CrossEncoder 对 top50 进行 rerank。
* 统计正样本排名，输出各项指标。
* 结果保存为 json 和 csv 文件。

**10. test\_bm25\_bi\_encoder\_hybrid\_with\_rewrite.py**

**功能说明**：评测“查询改写 + BM25+Bi-Encoder 融合检索”方案。**主要流程**：

* 先用大模型 API 对原始 query 进行改写（如提取关键词拼接）。
* BM25 用改写后的 query 检索，Bi-Encoder 仍用原始 query。
* 分数加权融合，检索 top50，统计正样本排名。
* 结果保存为 json 和 csv 文件。