根据您提供的文件，我将对 `pages/service` 文件夹下的 TypeScript 文件进行分析。

### 文件：`generateQA.ts`

- \*\*目的\*\*：生成问答对（QA）并处理相关的队列逻辑。

- \*\*依赖\*\*：

- `MongoDatasetTraining`：用于操作训练数据集合。

- `TrainingModeEnum`：训练模式的枚举。

- `getAIApi`：获取 AI 接口的函数。

- 其他日志、验证、字符串处理等工具函数。

- \*\*关键函数\*\*：

- `generateQA`：主函数，用于处理 QA 生成的队列。它首先检查队列长度，获取待处理的训练数据，验证团队的 AI 积分，调用 AI 模型生成 QA，并将结果推入训练队列。

- `formatSplitText`：辅助函数，用于格式化 AI 返回的答案文本，以符合 QA 对的格式。

### 文件：`generateVector.ts`

- \*\*目的\*\*：生成向量并处理相关的队列逻辑。

- \*\*依赖\*\*：

- `insertData2Dataset`：将数据插入数据集的函数。

- `MongoDatasetTraining` 和 `MongoDatasetData`：操作训练数据和数据集数据的集合。

- `TrainingModeEnum`：训练模式的枚举。

- 其他日志、验证、向量数据操作等工具函数。

- \*\*关键函数\*\*：

- `generateVector`：主函数，用于处理向量生成的队列。它获取待处理的训练数据，验证团队的 AI 积分，插入数据到数据集，更新向量存储，并处理错误和重试逻辑。

- `rebuildData` 和 `insertData`：辅助函数，用于重建数据或插入新数据。

### 文件：`utils.ts` (第三个文件)

- \*\*目的\*\*：提供工具函数，用于检查团队的 AI 积分并处理相关逻辑。

- \*\*依赖\*\*：

- `TeamErrEnum`：团队错误代码的枚举。

- `checkTeamAIPoints`：检查团队 AI 积分的函数。

- `sendOneInform`：发送通知的 API。

- `lockTrainingDataByTeamId`：锁定团队训练数据的函数。

- \*\*关键函数\*\*：

- `checkTeamAiPointsAndLock`：异步函数，用于检查团队的 AI 积分是否足够，并在不足时发送通知，锁定训练数据。

### 综合分析

这些文件主要涉及问答对生成、向量生成和团队 AI 积分验证的逻辑。它们通过与 MongoDB 集合交互，调用 AI 接口，处理队列管理，并在积分不足时发送通知和锁定数据。这些功能对于维护系统的稳定性和确保资源合理分配至关重要。

如果您需要更详细的分析或对特定部分有疑问，请告知。