以下是扩充后的进度汇报内容，细化了系统框架、当前进展和实施计划，便于展示系统开发的完整性和团队协作的推进情况：

**AI Agent调用小模型预测股票收益率——进度汇报**

**1. 系统总体框架**

**业务目标**

我们致力于开发一个基于深度学习和Java Web的股票收益率预测系统，满足以下业务目标：

1. **数据采集与处理**：
   * 采集多支股票的历史量价数据。
   * 使用深度学习模型预测指定股票未来3天的收益率。
2. **数据库构建与更新**：
   * 利用Tushare或Ricequant等数据源，构建一个实时更新的股票量价信息数据库。
3. **用户交互**：
   * 构建B/S模式信息系统。
   * 用户通过指令与大模型交互，完成数据查询、预测等任务。
   * 垂直领域任务（如收益率预测）由本地模型完成，结果通过前端反馈给用户。

**系统模块划分及交互**

**1. 前端**

* **页面设计**：
  + 使用课程提供的模板设计用户界面，确保系统美观、易用。
  + 提供动态交互功能，例如数据可视化、任务指令输入框。
* **大模型API和Agent接入**：
  + 与OpenAI、通义千问等大模型API集成，解析用户指令。
  + 动态调用本地深度学习模型完成垂直任务。

**2. 中端**

* **服务逻辑**：
  + 使用Spring Boot框架搭建服务端。
  + 负责任务分发，包括解析用户指令、调用数据库或本地模型。
  + 实现RESTful API，为前端提供统一的接口支持。

**3. 后端**

* **数据库**：
  + 数据源：使用Tushare采集10支高校持股的股票交易数据。
  + 构建MySQL数据库，存储历史量价数据和实时更新信息。
  + 提供查询接口，支持大模型任务和用户查询。
* **Agent模型**：
  + 模型选择：计划使用Transformer或LSTM建模，因其适合处理时间序列特征。
  + 数据：基于采集的历史交易数据进行模型训练和微调。
  + 功能：支持收益率预测任务，作为系统的垂直领域工具。

**2. 当前进展汇报**

**已完成的任务**

1. **系统设计**：
   * 完成系统模块划分，明确前中后端的功能与交互方式。
   * 选定技术栈，包括Spring Boot、MySQL、深度学习框架（PyTorch/TensorFlow）和大模型API。
2. **数据采集与处理**：
   * 已采集10支高校持股的股票过去一年的历史量价数据。
   * 初步完成数据预处理，包括缺失值填充、特征工程（如移动平均线、收益率计算）。
3. **数据库搭建**：
   * 完成MySQL数据库的初始化，创建了包含股票基本信息和交易数据的表结构。
   * 使用Python编写了数据更新脚本，可定期从Tushare拉取新数据并更新数据库。
4. **模型选择与初步实验**：
   * 已实现LSTM模型的初步训练代码，验证了其在样本数据上的表现。
   * 探索了Transformer模型在时间序列数据上的应用，实验结果表明其具备较强的特征捕捉能力。

**正在进行的工作**

1. **前端开发**：
   * 集成课程模板，设计用户界面布局。
   * 开发交互组件，如指令输入框和数据可视化面板。
2. **Agent模块开发**：
   * 编写Agent逻辑，支持基于大模型API的用户指令解析。
   * 搭建本地模型调用框架，将垂直任务结果反馈给用户。
3. **Spring Boot服务搭建**：
   * 开发中端RESTful API，连接前端和后端。
   * 实现数据查询和模型调用的接口。

**遇到的问题**

1. **数据实时性**：
   * 数据源（Tushare）的接口速率限制导致数据更新效率较低，正在尝试优化更新频率。
2. **模型训练难点**：
   * Transformer模型对小样本时间序列数据敏感，当前训练结果不够稳定。
3. **前后端交互**：
   * 大模型解析的指令多样性较高，可能需要进一步规范用户输入格式。

**3. 实施进度安排**

**时间表**

| **阶段** | **时间范围** | **任务内容** |
| --- | --- | --- |
| 第一阶段 | 已完成 | 系统设计、数据采集、数据库搭建和模型选择。 |
| 第二阶段 | 1周内 | 完成前端页面布局，Agent模块初步开发。 |
| 第三阶段 | 2周内 | 完成Spring Boot服务开发，数据库与前端整合。 |
| 第四阶段 | 3周内 | 深度学习模型的优化训练及本地部署。 |
| 第五阶段 | 4周内 | 集成大模型API，完成整体测试和功能演示。 |

**分工安排**

1. **数据处理**：
   * 负责：数据预处理、实时更新脚本的优化。
   * 成员：数据分析负责人。
2. **前端开发**：
   * 负责：完成UI设计，确保功能组件与后台的接口对接。
   * 成员：前端开发负责人。
3. **后端与模型**：
   * 负责：模型训练、服务接口设计与实现。
   * 成员：后端与算法负责人。
4. **集成与测试**：
   * 负责：大模型API与本地服务的集成，系统整体测试。
   * 成员：集成测试负责人。

**4. 总结**

系统开发已完成整体框架设计和关键模块的搭建工作，接下来将重点优化前端交互、Agent逻辑及模型性能。我们团队将按计划推进各项任务，确保系统在预定时间内完成并展示所有功能。

这样的扩充包含明确的框架、进展和未来计划，可用于开发进度的正式汇报。