

药店销售预测系统用户手册

目录

1. 系统介绍
2. 系统架构
3. 系统安装与配置
4. 功能模块说明
 - 首页概览
 - 数据管理
 - 模型训练
 - 预测分析
 - 模型管理
5. 命令行操作指南
6. 常见问题解答
7. 技术支持

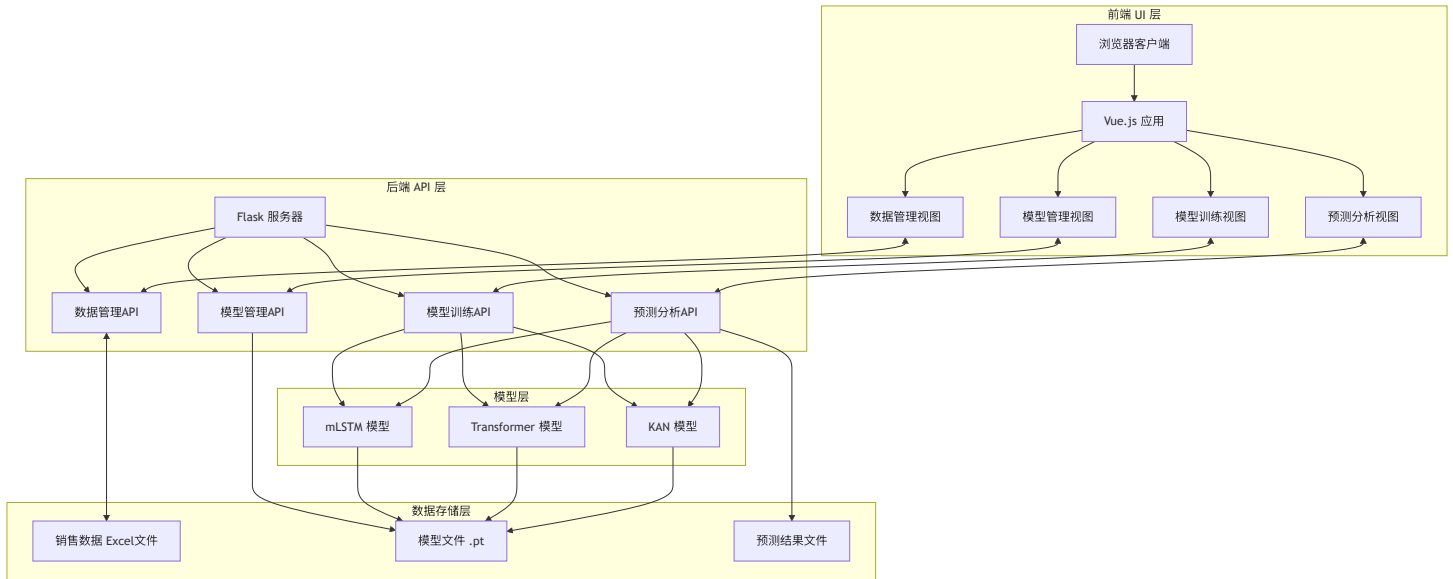
系统介绍

药店销售预测系统是一款基于人工智能的药品销售预测工具，通过深度学习算法分析历史销售数据，为药店提供精准的销售预测服务，帮助药店优化库存管理，提高经营效率。

系统采用前后端分离的架构，前端基于Vue.js和Element Plus构建现代化的用户界面，后端使用Flask提供RESTful API服务，支持多种预测模型，包括mLSTM、Transformer和KAN（Kolmogorov-Arnold Network）。



系统架构



系统由以下几部分组成：

1. **前端界面**：基于Vue.js和Element Plus构建的用户交互界面
2. **后端API**：基于Flask的RESTful API服务
3. **预测模型**：包含mLSTM、Transformer和KAN三种深度学习模型
4. **数据存储**：使用文件系统存储模型和预测结果

系统安装与配置

前端部署

1. 确保已安装Node.js环境（推荐v16.0.0以上版本）
2. 进入UI目录： `cd UI`
3. 安装依赖： `npm install`
4. 开发模式运行： `npm run dev`
5. 构建生产版本： `npm run build`

后端部署

1. 确保已安装Python环境（推荐Python 3.10以上版本）
2. 安装依赖： `pip install -r requirements.txt`
3. 启动API服务： `python api.py`

服务器将在默认端口5000上运行。

访问前端界面

在浏览器中访问：

`http://localhost:5000/ui/`

功能模块说明

首页概览

首页提供系统的整体概况，包括产品数量、已训练模型数量、平均预测准确率等关键指标，以及最近的预测结果和活跃模型列表。

操作步骤：

1. 登录系统后，默认进入首页
2. 查看关键统计指标和最近活动

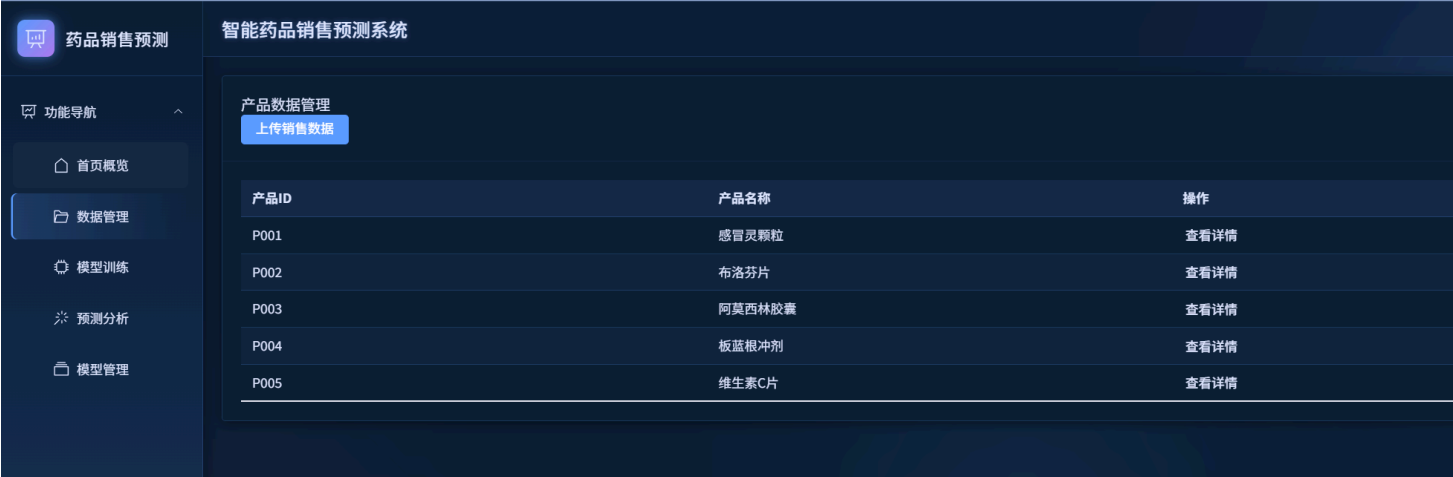


数据管理

数据管理模块允许用户上传、查看和管理药品销售数据。系统支持Excel格式的数据上传，并提供数据可视化功能。

操作步骤：

1. 点击左侧菜单的"数据管理"
2. 查看现有产品列表
3. 点击"上传销售数据"按钮上传新数据
4. 点击产品名称查看详细销售数据和趋势图



历史数据查看：



模型训练

模型训练模块允许用户选择产品和算法模型，启动训练任务，并查看训练进度和结果。

操作步骤：

1. 点击左侧菜单的"模型训练"
2. 在左侧面板选择产品、模型类型和训练参数
3. 点击"启动训练"按钮
4. 在右侧任务列表查看训练状态和结果

智能药品销售预测系统

启动模型训练

产品

请选择产品

模型类型

mLSTM

训练轮次

-

50

+

启动训练

训练任务队列

任务ID	产品ID	模型类型	状态	创建时间	详情
No Data					

预测分析

预测分析模块允许用户使用已训练的模型进行销售预测，并提供预测结果的可视化展示。

操作步骤：

- 1. 点击左侧菜单的"预测分析"
- 2. 选择产品、模型类型和预测参数
- 3. 点击"查询可用模型"按钮
- 4. 从模型列表选择一个模型，点击"执行预测"
- 5. 查看预测结果图表和数据

功能导航

首页概览

数据管理

模型训练

预测分析

模型管理

模型预测

产品

选择产品

模型类型

选择模型类型

预测起始日期

2025-06-11

预测天数

-

7

+

查询可用模型

可用模型列表

模型ID	产品	创建时间	评估指标	操作
kan kan_P003	阿莫西林胶囊	2025/6/11 13:03:33	R²:N/A RMSE:3.6451 MAE:2.8219 MAPE:N/A%	执行预测
mlstm mlstm_P002	布洛芬片	2025/6/11 10:22:29	R²:N/A RMSE:4.6271 MAE:3.6711 MAPE:N/A%	执行预测
kan kan_P002	布洛芬片	2025/6/11 09:11:39	R²:N/A RMSE:4.5373 MAE:3.5170 MAPE:N/A%	执行预测
			R²:N/A RMSE:4.5373 MAE:3.5170 MAPE:N/A%	

预测结果展示：

预测结果详情

产品名称	阿莫西林胶囊	产品ID	P003	模型类型	kan	预测时间	2025/6/11 13:38:01
------	--------	------	------	------	-----	------	--------------------

预测趋势图

阿莫西林胶囊 - kan销量预测 (从2025-06-11开始, 预测7天)

同期销量对比

阿莫西林胶囊 - 同期销量趋势对比 (7天)

数据详情

预测数据

历史数据

全部数据

平均预测销量

20.01

最高预测销量

23.02

最低预测销量

18.38

日期	预测销量 	趋势
2025/6/11	19.75	-
2025/6/12	19.00	∨ 0.75
2025/6/13	20.25	∧ 1.25
2025/6/14	18.88	∨ 1.36
2025/6/15	18.38	∨ 0.50
2025/6/16	20.81	∧ 2.42
2025/6/17	23.02	∧ 2.22

模型管理

模型管理模块允许用户查看、导出和删除已训练的模型，也支持导入外部模型。

操作步骤：

- 1. 点击左侧菜单的"模型管理"
- 2. 查看模型列表
- 3. 使用过滤器筛选特定产品或模型类型
- 4. 点击"详情"查看模型详细信息
- 5. 点击"导出"下载模型文件
- 6. 点击"删除"移除不需要的模型
- 7. 点击"导入模型"上传外部训练的模型文件



系统组件

1. 前端UI

- 基于Vue.js和Element Plus构建的现代化界面
- 蓝色主题的沉浸式用户体验
- 响应式设计，适配不同设备屏幕

2. 后端API

- 基于Flask的RESTful API
- 支持数据上传、模型训练、销售预测和模型管理
- Swagger API文档支持

3. 预测模型

- **mLSTM模型**：多层长短期记忆网络，适合序列数据预测
- **Transformer模型**：基于自注意力机制，捕捉长期依赖关系
- **KAN模型**：Kolmogorov-Arnold网络，具有高精度的函数拟合能力

4. 数据管理

- 支持Excel格式的销售数据导入导出
- 历史销售数据可视化
- 预测结果可视化与导出

命令行操作指南

除了图形界面外，系统也提供命令行操作方式，适合高级用户和自动化脚本使用。

快速入门

在项目根目录下，运行以下命令启动命令行界面：

```
python run_pharmacy_prediction.py
```

主菜单导航

启动后，您将看到主菜单界面：

```
=====
🏪 药店单品销售预测系统 🏪
=====
1. 训练所有药品的销售预测模型
2. 训练单个药品的销售预测模型（Transformer）
3. 训练单个药品的销售预测模型（mLSTM）
4. 训练单个药品的销售预测模型（KAN）
5. 查看已有预测结果
6. 使用已训练的模型进行预测
7. 比较不同模型的预测结果
8. 模型管理
0. 退出
=====
```

功能详解

训练模型

系统支持三种主要的模型训练方式：

- **训练所有药品模型：**选择主菜单中的选项 1
- **训练单个药品模型：**选择选项 2 、 3 或 4 ，分别使用Transformer、mLSTM或KAN模型

查看预测结果

选择主菜单中的选项 5 ，系统会显示已有的预测结果列表。

使用模型预测

选择主菜单中的选项 6 ， 可以使用已训练的模型进行预测。

比较模型预测结果

选择主菜单中的选项 7 ， 可以比较不同模型对同一产品的预测结果。

模型管理

选择主菜单中的选项 8 ， 进入模型管理子菜单：

```
=====
🏠 药店销售预测系统 - 模型管理工具 🏠
=====

1. 查看所有模型
2. 查看特定产品的模型
3. 查看特定模型的详细信息
4. 使用模型进行预测
5. 比较不同模型的预测结果
6. 删除模型
7. 导出模型
8. 导入模型
0. 退出
=====
```

命令行参数

许多功能也可以通过命令行参数直接调用，例如：

```
# 使用mLSTM模型训练P001产品的销售预测模型
python run_pharmacy_prediction.py --train P001 --model mlstm

# 比较P001产品的不同模型预测结果
python run_pharmacy_prediction.py --compare P001
```

模型管理命令行工具

模型管理功能也可以通过独立的命令行工具使用：

```
# 列出所有模型
python model_management.py --action list

# 查看特定产品的模型详情
python model_management.py --action details --product_id P001 --model_type mlstm

# 使用特定模型进行预测
python model_management.py --action predict --product_id P001 --model_type mlstm
```

API服务使用

启动API服务

运行以下命令启动API服务：

```
python api.py
```

默认情况下，API服务会在 <http://localhost:5000> 上运行。

访问API文档

启动服务后，访问 <http://localhost:5000/swagger/> 可以查看所有API接口说明并进行测试。

使用API示例

以下是一些基本的API使用示例：

```
# 获取产品列表
curl -X GET "http://localhost:5000/api/products"

# 获取特定产品销售数据
curl -X GET "http://localhost:5000/api/products/P001/sales"

# 启动模型训练
curl -X POST "http://localhost:5000/api/training" \
  -H "Content-Type: application/json" \
  -d '{"product_id": "P001", "model_type": "mlstm"}'

# 获取预测结果
curl -X POST "http://localhost:5000/api/prediction" \
  -H "Content-Type: application/json" \
  -d '{"product_id": "P001", "model_type": "mlstm", "days": 7}'
```

常见问题解答

问题1：如何选择最合适的预测模型？

回答：三种模型各有特点：

- mLSTM：适合较短期的预测，训练速度快
- Transformer：适合中长期预测，对季节性变化敏感
- KAN：适合复杂模式识别，通常有最高的准确率但训练时间较长

根据预测周期和数据特点选择合适的模型。一般情况下，如果不确定，可以使用KAN模型获得最佳效果。

问题2：为什么模型训练失败？

回答：常见原因包括：

- 数据量不足：确保至少有30天以上的销售数据
- 数据异常：检查数据中是否有缺失值或异常值
- 服务器资源不足：大型模型训练需要足够的计算资源

问题3：如何提高预测准确率？

回答：

- 提供更多历史数据
- 增加训练轮次 (epochs)
- 结合多个模型的预测结果
- 加入更多相关特征 (如节假日、天气等)

问题4：系统支持哪些数据格式？

回答：目前仅支持Excel(.xlsx)格式的销售数据文件。

问题5：训练速度慢

回答：

- 检查是否正在使用GPU加速
- 减小批大小 (batch_size)
- 减少训练轮次 (epochs)
- 考虑使用更简单的模型

问题6：模型保存失败

回答：

- 检查磁盘空间是否充足
- 确保有写入权限
- 尝试手动创建predictions目录

系统要求

- **后端**：Python 3.10或更高版本，安装所有requirements.txt中的依赖
- **前端**：现代浏览器(Chrome, Firefox, Edge等)