

分类号\_\_\_\_\_

U D C\_\_\_\_\_

编 号\_\_\_\_\_

密 级\_\_\_\_\_



**南方科技大学**  
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

# 本科生毕业设计（论文）

题 目： 基于 Spring Boot 的神经网络交易系统

姓 名： 顾隽逸

学 号： 11910424

系 别： 计算机科学与工程系

专 业： 计算机科学与技术

指导教师： 唐茗

2023 年 6 月 1 日

# 诚信承诺书

1. 本人郑重承诺所呈交的毕业设计（论文），是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料均真实可靠。

2. 除文中已经注明引用的内容外，本论文不包含任何其他人或集体已经发表或撰写过的作品或成果。对本论文的研究作出重要贡献的个人和集体，均已在文中以明确的方式标明。

3. 本人承诺在毕业论文（设计）选题和研究内容过程中没有抄袭他人研究成果和伪造相关数据等行为。

4. 在毕业论文（设计）中对侵犯任何方面知识产权的行为，由本人承担相应的法律责任。

作者签名：

\_\_\_\_\_年\_\_\_\_月\_\_\_\_日

# 基于 Spring Boot 的神经网络交易系统

顾隽逸

(计算机科学与工程系 指导教师：唐茗)

**【摘要】**神经网络经过不断地发展，在计算能力快速提升的今天成为了当下研究的热点。目前的神经网络模型通过使用数据集训练，从而能够在不同的领域内，完成对应的任务。然而，相较于购买用于训练与测试神经网络模型的数据集，购买神经网络模型则可以帮助购买者更快地将该模型应用于自己所需的领域。为帮助对于神经网络感兴趣，并想要购买高质量神经网络模型的群体，本文提出了一种基于 Spring Boot 的神经网络交易系统，该系统前后端通过 Ajax 交互，用户由管理员、卖家以及买家组成，并在该系统添加了测试神经网络精确度的功能。通过系统内置的线上神经网络精确度测试功能，不仅可以保护买卖双方的数据集中可能存在的隐私问题，也解决了目前网络上购买神经网络模型的不确定性这一问题。

**【关键词】**Spring Boot; 神经网络; 交易系统; 精确度测试

**[ABSTRACT]** Neural networks have continuously evolved, and with the rapid improvement of computing power, this field has become a hotspot of current research. Nowadays, neural network models are trained using datasets, enabling them to complete corresponding tasks in different fields. However, compared to purchasing datasets for training and testing neural network models, buying a neural network model can help buyers apply the model to their desired fields more quickly. To assist individuals interested in neural networks and looking to purchase high-quality neural network models, this article proposes a neural network trading system based on Spring Boot. The system features Ajax interaction between the Frontend and the Backend, and users consist of administrators, sellers, and buyers. Additionally, the system includes a function for testing the accuracy of neural networks. The built-in online neural network accuracy testing function not only protects potential privacy issues in the datasets of both parties but also addresses the uncertainty of purchasing neural network models on the internet.

**[Keywords]** Spring Boot; neural network; trading system; accuracy test

# 目录

<b>1</b>	<b>绪论.....</b>	<b>1</b>
<b>2</b>	<b>需求分析.....</b>	<b>3</b>
2.1	业务需求.....	3
2.1.1	用户界面.....	3
2.1.2	运行环境.....	3
2.2	功能需求.....	3
2.3	Spring Boot 的优点 .....	4
<b>3</b>	<b>系统设计.....</b>	<b>5</b>
3.1	系统功能架构图.....	5
3.2	系统逻辑架构图.....	6
3.2.1	逻辑视图.....	6
3.2.2	业务流程图.....	7
3.2.3	处理视图.....	8
3.3	数据字典.....	9
3.4	用户  界面设计.....	14
3.5	静态页面设计.....	14
3.6	用例图.....	15
<b>4</b>	<b>基本功能.....</b>	<b>16</b>
4.1	买卖功能.....	16
4.2	管理员系统.....	17
4.3	收藏与留言系统.....	18

<b>5</b>	<b>系统神经网络精确度测试功能 .....</b>	<b>19</b>
5.1	神经网络精确度测试的基本流程.....	19
5.2	神经网络精确度测试在系统内的实现.....	19
<b>6</b>	<b>改进方案.....</b>	<b>21</b>
6.1	功能性需求.....	21
6.1.1	测试能力扩展.....	21
6.1.2	优化留言功能.....	21
6.2	非功能性需求.....	21
6.2.1	性能需求.....	21
6.2.2	安全性需求.....	21
<b>7</b>	<b>参考文献.....</b>	<b>22</b>
<b>8</b>	<b>致谢.....</b>	<b>24</b>

# 1 绪论

神经网络可以被用来做很多事情，例如图像分类、物体检测、语音识别等。在图像分类中，神经网络可以用于识别图像中的特定对象。例如，如果想要从一张照片中识别出一个人，那么可以使用深度神经网络来实现这一点。在物体检测中，神经网络可以用于检测经过训练的目标。在语音识别中，神经网络可以用于语音信号的识别。在自然语言处理中，神经网络可以用于文本分类。

目前，国内已有大量对于 VGG 网络实际应用的研究。张建华等人<sup>[1]</sup>基于 VGG16 模型实现了自然条件对于棉花不同病害的准确识别。王羽微等人<sup>[2]</sup>基于 VGG16 网络实现了分类识别性能较好的神经网络模型。柯溢对于 VGG16 识别宠物犬<sup>[3]</sup>易产生过拟合的问题提出了改进方案，识别精确度达到了 96%。钱蓉等人基于 VGG16 进行水稻害虫的智能识别<sup>[4]</sup>进行了研究，使用田间实拍照片训练，准确率为 90.7%。徐佳琦的研究基于使用边缘提取以及 VGG16 网络来识别混凝土裂缝<sup>[5]</sup>，准确性达到了 96%，可较好得解决人工识别的费时费力甚至是安全隐患。廖璐明等人<sup>[6]</sup>通过优化 VGG16 网络以及数据增强方法优化了德国交通标志的数据集，提高了识别率以及收敛速度。曹宇等人基于 VGG16 检测肺结节的研究<sup>[7]</sup>有着 92.56% 的准确率和 94.44% 的高敏感度。徐旭东等人<sup>[8]</sup>通过迁移学习和 VGG 网络来锁定控制图的异常模式。同样，国内也有大量对于 ResNet 应用的研究。张雷等人的研究基于 ResNet 检测人员入侵方法<sup>[9]</sup>，在不同场景下的检测精确度达到 97.8%。金张根等人基于 ResNet 研究了城市生活垃圾的识别<sup>[10]</sup>，使用 Kaggle 数据集，优化参数后的网络精确度达到了 90.0%。廖明霜等人<sup>[11]</sup>修改了 ResNet 的卷积层和池化层，实现了花型和品种的高精确度智能识别。牛智有等人设计了一种基于 ResNet18 的多通道自动入仓原料的饲料自动识别装置<sup>[12]</sup>，在测试集识别的准确率高达 99.4%。张媛等人<sup>[13]</sup>使用 ResNet34 模型解决农田杂草精准防控，相较于未优化的神经网络准确率有较大提高。王斌<sup>[14]</sup>等人改进 ResNet，较好地解决了阿尔兹海默症、轻度认知障碍人群的脑部核磁成像难以分辨的问题。张文景等人<sup>[15]</sup>基于 ResNet18 模型提高了苹果病害的识别准确率以及识别效率。朱家辉等人则对比了 VGG16 与 ResNet50 在植物病害的诊断中的性能差异。

如前所述，神经网络模型可以应用于日常生活中的很多地方，能够在生活中的不同领域内完成对应的任务。因此，神经网络模型的交易是十分有必要的。对于期望省去训练模型的繁琐过程的群体，通过交易获得模型是十分便利的，不同的购买者也能

够轻松地获得不同的期望效果。

如今网络上已有众多数据买卖的平台，也可以查阅到数据买卖相关的论文。Fan Liang 等人<sup>[17]</sup>以及 Ling Tian 等人<sup>[18]</sup>均发表了对于数据交易做出的研究，数据交易可以减少收集数据以及处理数据所带来的时间成本。然而，数据交易也存在着缺点：数据的质量并无法保证，由于数据集往往是庞大且复杂的，评估数据集质量相对较为困难；每一个用户对于数据集的需求可能完全不同，导致数据交易难以满足用户的个性化数据需求。因此，数据集的交易相较于神经网络模型的交易有着较大缺陷，相较于较为繁琐的数据集质量测试，神经网络模型的测试则较为简单，通过在自定数据集的训练集上训练神经网络，并在同一数据集的测试集上测试该神经网络的精确度，能够较为直观的了解神经网络模型在所需要的领域的精确度；周东宁<sup>[19]</sup>的实验数据表明神经网络模型存在鲁棒性，对抗测试中的轻微扰动对神经网络的影响较小<sup>[20]</sup>，神经网络模型对于同一领域的数据集存在鲁棒性，能够满足用户的个性化需求。

目前，并未找到有关专门用于交易神经网络的平台或是有关神经网络交易的论文，本文旨在解决互联网上的各种功能的神经网络的优劣层次不齐，对想要寻求高质量的现成神经网络模型的群体所造成困扰的这一问题。该群体可能会在购买搜索到的神经网络模型之后，发现其所购买的神经网络模型并不符合自己的预期，然而购买已经发生，像神经网络模型这样的虚拟商品往往在购买后是无法退款的，造成了这一类群众钱财的无意义的浪费。针对这一问题，在神经网络交易平台中加入神经网络精确度测试的模块，可以满足上述群体的需求。

本文将通过开发阶段：需求分析、系统设计；功能展示：基本功能、神经网络精确度测试功能以及改进方案三部分，递进式介绍搭建的神经网络交易系统。



## 2 需求分析

首先,为了尊重用户在训练神经网络时付出的时间、精力,设计出一个基于 Spring Boot 开发的神经网络买卖 web 平台是符合当前的环境的,该平台允许用户购买或是卖出已训练完成的神经网络。

其次,由于是一个基于 Spring Boot 的神经网络交易平台架构,因此,需要搭建的模块有商城门户、买家模块、卖家模块、购物车模块、订单模块、账单模块、后台管理模块、神经网络模型测试模块等。

接着,由于买卖双方的数据集内的数据可能存在一定的隐私性,这可能导致卖家无法向买家提供自己所持有的数据集供买家自行进行测试;也可能导致买家无法将自己所持有的数据集发送给卖家进行测试并返回结果。因此,为保证用户能够购买到心仪的神经网络商品,平台加入小数据集测试,供用户在购买前测试神经网络对于所需方向的准确性,再决定是否购买。这一方式可保证用户能够购买到符合预期的神经网络,而不是因购买到不合适的神经网络之后却无法退款,而导致用户浪费钱财。

以上为对于开发该神经网络模型交易系统的需求分析,下面将通过业务需求以及功能需求两方面分别简要展开叙述系统所能达到的期望目标以及如何达成该目标。

### 2.1 业务需求

#### 2.1.1 用户界面

用户界面是该项目中用户能够看见并且能够与之交互的部分,因此需要设计一个良好的用户界面是十分重要的。该界面需要能够让用户轻松找到自己所需的各项功能,并且功能界面较为简洁。

#### 2.1.2 运行环境

能够在 Lenovo Legion Y7000P2020H 上较为稳定地部署并运行,运行后能够在计算机本地 edge 浏览器使用访问该项目 web 页面,并对各项功能进行使用。

### 2.2 功能需求

需要完善的登录模块,其中包含管理员的登录、会员的注册以及登录;需要管理员的用户管理模块,负责增加或者删除会员的账号等功能;需要完善的商品模块,管理员能够上架或下架卖家新增的商品,卖家能够新增商品以及发货,买家能够购买商品等功能;需要完善的账户模块,买家可以查询到自己账号的余额信息,会员可以编辑账户,如修改姓名、昵称、邮箱、电话等,可以编辑账号简介、查看账户信息等功

能；需要管理员可操作的系统管理模块，如公告系统、资讯系统等功能。

## 2.3 Spring Boot 的优点

由于 Java 是一种静态语言，在使用 Java 的开发过程中往往会显得有些笨重，Java 需要大量的配置以及复杂的部署流程，因此导致了 Java 低下的开发效率<sup>[21]</sup>。然而，Spring Boot 的出现在很大程度上解决了上述问题，Spring Boot 可以以 jar 包的形式独立运行以及内嵌 Tomcat，无需以 war 包形式部署项目。Spring Boot 会根据在类路径中的 jar 包和类，为 jar 包里的类自动配置 Bean，并考虑了大多数的开发场景，同时，对于少数开发场景也可以自定义自动配置<sup>[22]</sup>。

因此，Spring Boot 可快速构建项目，支持主流开发框架的无配置集成，且可独立运行，无需依赖外部 Servlet 容器；极大地提高了开发、部署效率。

### 3 系统设计

开发系统首先需要对系统进行设计，本章将从系统功能架构图、系统逻辑架构图、数据字典、用户 ui 界面设计、静态页面设计、用例图六方面介绍。

系统功能架构图主要展示了系统每一层负责的功能，系统逻辑架构图通过逻辑视图、业务流程图、处理视图三方面展示系统的基本工作流程以及功能，数据字典主要介绍数据库建表，用户 ui 界面设计以及静态页面设计展示了系统前端页面的布局，用例图展示了不同角色在系统内可以使用的功能。

#### 3.1 系统功能架构图

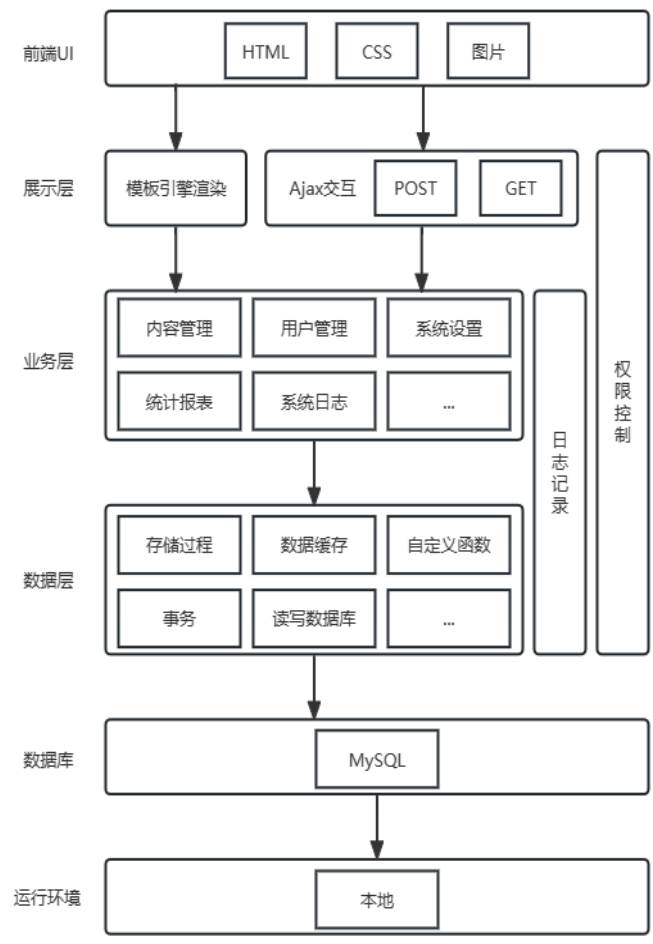


图 1 系统功能架构图

运行环境选择为笔记本（Lenovo Legion Y7000P2020H）本地运行；数据库选择使用 MySQL；数据层实现的是储存过程、数据缓存、自定义函数、事务、读写数据库等；业务层实现的是内容管理、用户管理、系统设置、统计报表、系统日志等；展示层使用模板引擎渲染，Ajax 的 POST 与 GET 作为前后端交互指令；前端 UI 使用 HTML、CSS 以及图片作为基础。▲

## 3.2 系统逻辑架构图

### 3.2.1 逻辑视图

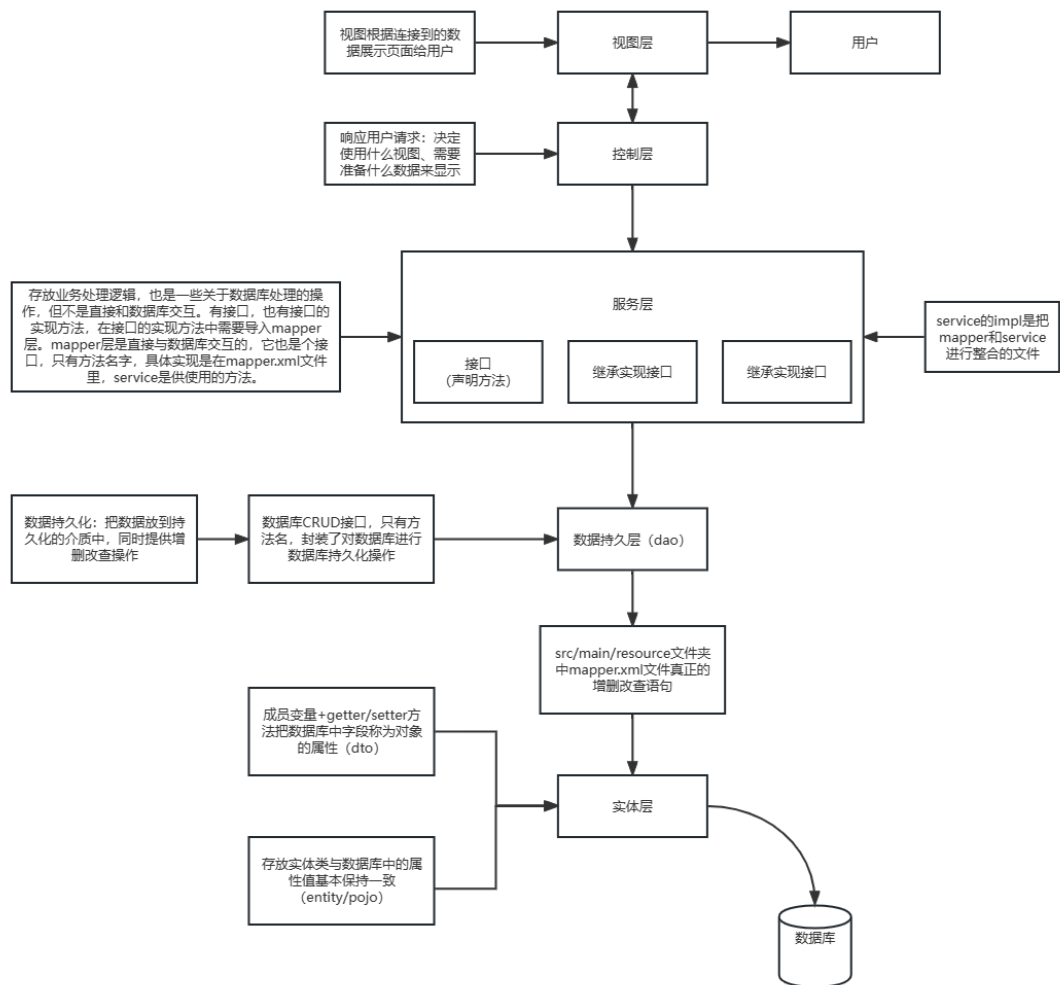


图 2 逻辑视图

本系统采用的是 Spring Boot 的五层结构，视图层、控制层、服务层、数据持久层以及实体层，如图 2 所示。

视图层所连接道德数据展示页面给用户。

控制层任务是响应用户的需求，所需要用到的视图、显示的数据都是由控制层来决定的。因此，控制层主要负责的便是前后端的交互，它先接收前端发来的请求，然后调用服务层，再接收服务层返回的数据，最后将具体的数据和页面发送至前端。

服务层主要分为三个方面：接口、继承实现的接口、接口的实现。接口的作用是声明方法，接口的实现的作用是将数据持久层和服务层进行整合。服务层负责存放业务逻辑处理，包括一些涉及数据库的操作，但不直接与数据库进行交互。

数据持久层中只有方法名，具体实现在 mapper.xml 文件中，对数据库进行数据持久化操作。mapper.xml 文件中储存的是数据库的增删改查语句。

实体层负责存放实体类，与数据库中的属性基本保持一致，一般包括 `getter`、`setter` 方法。

由于 `Spring Boot` 的分层架构，使得服务层和数据库间的更改互相独立：修改服务层的需求，无需修改数据持久层；同样的，如果有对于访问数据库的新需求，也只需要在数据持久层修改，无需修改服务层。

### 3.2.2 业务流程图

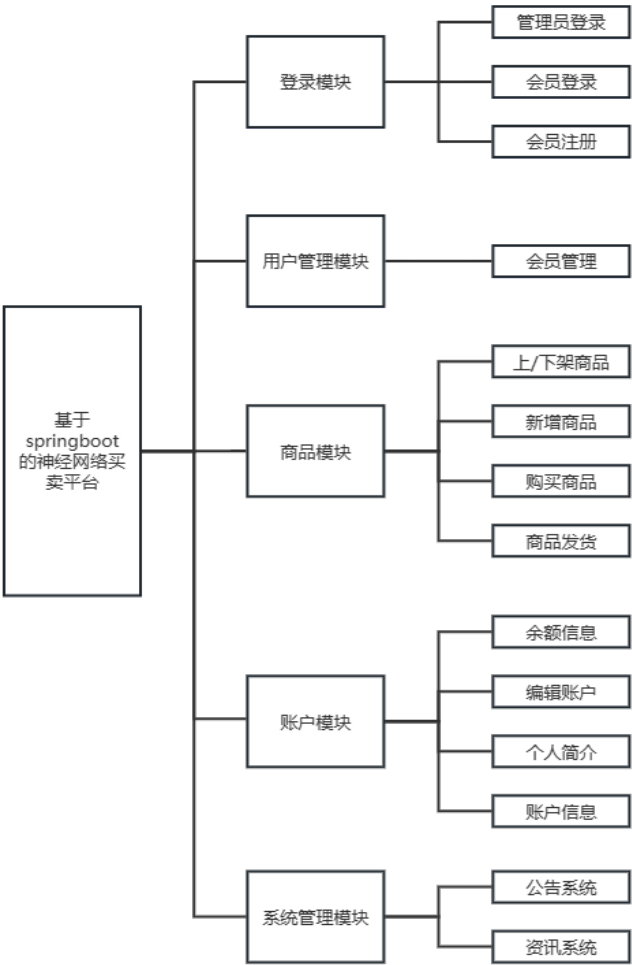


图 3 业务流程图

该系统是基于 `Spring Boot` 的神经网络交易系统，管理员可以通过后台管理会员账号、商品状态、公告系统以及资讯系统；会员可自行上架商品以及商品信息（但需要管理员审核后才能正式上架），可对余额进行充值，编辑账户信息、个人简介等，如图 3 所示。

3.2.3 处理视图

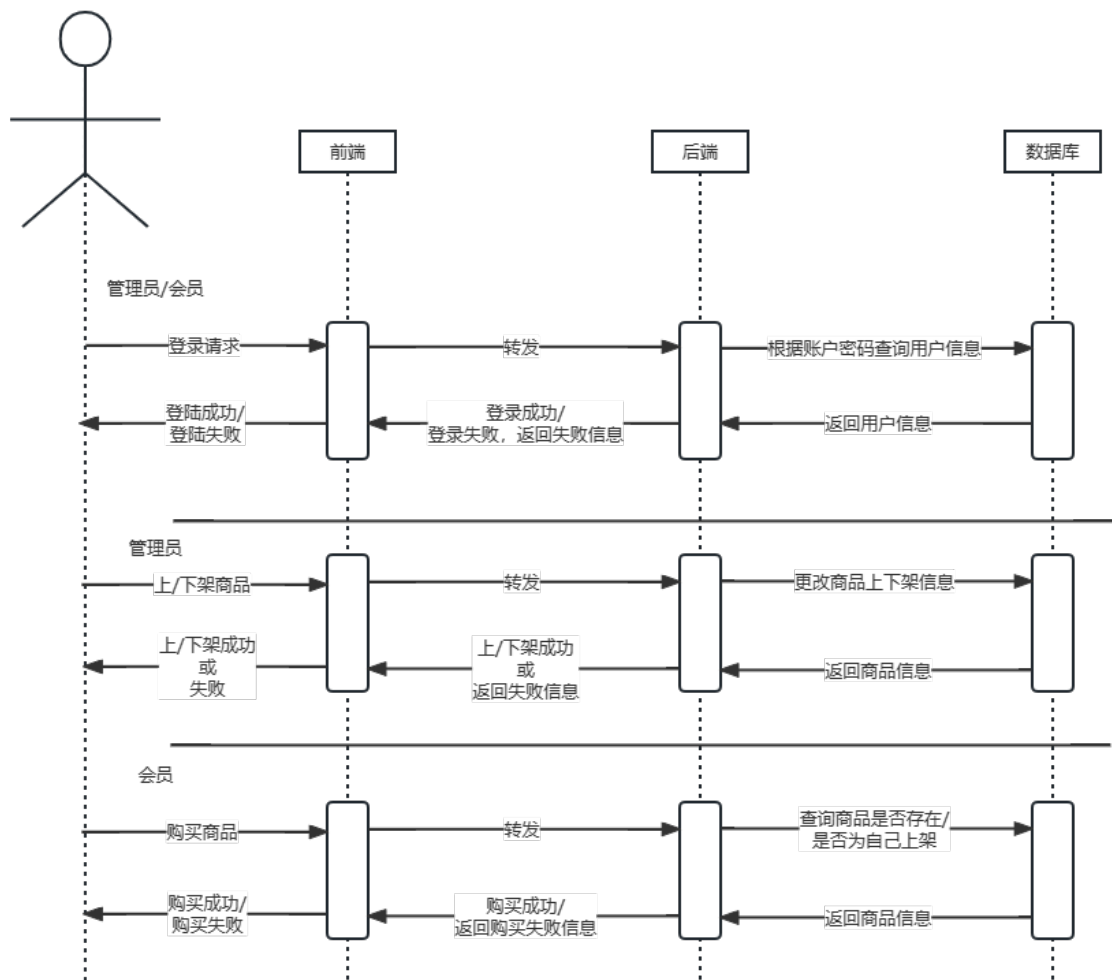


图 4 处理视图

如图 4 所示，处理视图演示了该项目三种较为基础的功能处理。

1) 管理员与会员的登录功能：

用户首先在前端点击登录并输入账号、密码、验证码，选择账号类型后，前端将数据转发至后端，后端通过账号类型、账号、密码在数据库查询账号是否存在，查询成功则逐级返回至前端登陆成功；若在数据库中无法查询到，则在前端显示账号不存在。

2) 管理员上架或下架商品的功能：

管理员可在管理员界面控制上架或下架用户上传的商品，首先在前端选择具体商品并上/下架，前端将请求发送至后端，后端接收到信息后，在数据库中修改商品的上下架状态，并返回成功或失败信息。

3) 会员购买商品的功能：

会员在商城找到心仪的商品后可在前端发起购买请求，前端发送请求至后端，并

查询商品是否存在以及商品是否为该会员本人上架，若商品存在且不为本人上架，则返回购买成功；若商品已不存在或者为本人上架，则返回购买失败。

其他功能，如账户余额充值、修改账户信息、资讯系统管理、公告系统管理等，处理模式与上方三种类似。

### 3.3 数据字典

数据字典是用户可访问的记录数据库，可简单理解为记录后台数据的数据库建表。因此，数据字典的设计需要满足数据库的三大范式要求。

首先是第一范式（1NF），“它要求关系满足一种最基本的条件；它与其它范式不同，不需要诸如函数依赖之类的额外信息。如果某个域的元素被认为是不可分割的单元，那么这个域就是原子的。如果一个关系模式  $R$  的所有属性的域都是原子的，我们称关系模式  $R$  始于第一范式。”<sup>[23]</sup>即第一范式要求 “属于第一范式关系的所有属性都不可再分，数据项不可分”。然而，仅仅通过第一范式来设计数据字典是不够的，第一范式的约束仍然可能导致数据冗余、插入异常、修改异常、删除异常等问题。因此，需要引入规范化的概念，即 “一个低一级的关系模式通过模式分解可以转化为若干个高一级范式的关系模式的集合”。

接下来便需要引入第二范式（2NF），“如果模式  $R$  属于第一范式，并且  $R$  中每个非主属性都完全依赖于关键字，则该关系属于第二范式。”<sup>[23]</sup>简单的，第二范式要求数据库中的每一张表都需要有一个（有且仅有一个）数据项作为主键。

最后则是第三范式（3NF），它要求非主属性既不传递依赖于码（码是可以确定一个元组的所有信息的属性名或属性名组），也不部分依赖于码。可见，第三范式的要求是建立在满足第二范式的基础上的，任何非主属性都不依赖于其他主属性，也因此消除了数据库中的传递依赖。

以下数据字典表 1-10 为本神经网络交易系统主要用于存储数据的数据库表。所有表中数据都已不可再分，满足第一范式。在满足数据库设计的第一范式后，需要利用第二范式来消除数据库中的部分依赖，因此数据字典表 1-10 中均使用自增  $id$  项作为数据库的唯一主键，每一张表只描述一件事情，即表名所描述的。数据字典表 1-10 中各项都完全依赖于自增  $id$ ，满足第三范式。

设计出满足范式的数据库表后，在表创建语句的末尾添加 “ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8 ROW\_FORMAT=COMPACT”，

“ENGINE=InnoDB”即使用 innodb 引擎；“AUTO\_INCREMENT=3”表示插入表内的自增值起始值为 3，为方便测试数据库是否能和后端正确交互，数据库表中预先通过手动插入 1 至 2 条数据，因此，在系统运行中插入的数据主键不会与提前插入的数据主键冲突；“CHARSET=utf8”即数据库默认编码是 utf8；“ROW\_FORMAT=COMPACT”为数据库高效能存放数据，如果一个页中存放的行数越多，其性能就越高。

表 1 appointorder

编号	字段名称	字段含义	字段类型	字段长度	是否主键	默认值
1	id	编号	int	11	是	
2	ddno	订单号	varchar	500		
3	yydate	时间	datetime			
4	fee	费用	double			
5	hyid	会员 id	int	11		
6	mobile	手机号	varchar	500		
7	state	订单状态	int	11		
8	targetid	目标 id	int	11		
9	des	订单备注	longtext			
10	count	商品数量	int	11		
11	danjia	单价	double	18,2		
12	fileurl	文件地址	varchar	2000		

表 2 comment

编号	字段名称	字段含义	字段类型	字段长度	是否主键	默认值
1	id	编号	int	11	是	
2	belongid	被评论 id	varchar	255		
3	cdata	评论内容	longtext			
4	createtime	创建时间	datetime			
5	xtype	文段类型	varchar	255		
6	hyid	会员 id	int	11		
7	agreecount	点赞数	int	11		



表 3 fans

编号	字段名称	字段含义	字段类型	字段长度	是否主键	默认值
1	id	编号	int	11	是	
2	targetid	目标 id	int	11		
3	actionid	会员 id	int	11		
4	createtime	创建时间	datetime			

表 4 huiyuan

编号	字段名称	字段含义	字段类型	字段长度	是否主键	默认值
1	id	编号	int	11	是	
2	accountname	账号名称	varchar	255		
3	address	地址	varchar	255		
4	des	备注	varchar	255		
5	email	邮箱	varchar	255		
6	idcardno	卡号	varchar	255		
7	logtimes	登陆次数	int	11		
8	mobile	手机号	varchar	255		
9	name	姓名	varchar	255		
10	nickname	昵称	varchar	255		
11	paypwd	支付密码	varchar	255		
12	password	账号密码	varchar	255		
13	regdate	注册时间	datetime			
14	sex	性别	varchar	255		
15	status	账号类型	int	11		
16	touxiang	头像	varchar	255		
17	yue	余额	float			
18	fanscount	粉丝数	int	11		

表 5 jiaodiantu

编号	字段名称	字段含义	字段类型	字段长度	是否主键	默认值
1	id	编号	int	11	是	
2	href	图片路径	varchar	255		
3	title	标题	varchar	255		
4	tupian	图片	varchar	255		
5	xtype	类型	varchar	255		

表 6 leaveword

编号	字段名称	字段含义	字段类型	字段长度	是否主键	默认值
1	id	编号	int	11	是	
2	dcontent	留言内容	longtext			
3	pubtime	发布时间	datetime			
4	hyid	会员 id	int	11		
5	replytime	回复时间	datetime			
6	replyren	回复者	varchar	500		
7	replycontent	回复内容	longtext			
8	state	留言状态	int	11		

表 7 notice

编号	字段名称	字段含义	字段类型	字段长度	是否主键	默认值
1	id	编号	int	11	是	
2	clickcount	点击数	int	11		
3	dcontent	公告内容	text			
4	pubren	发布者	varchar	255		
5	pubtime	发布时间	datetime			
6	title	标题	varchar	255		

表 8 shoucang

编号	字段名称	字段含义	字段类型	字段长度	是否主键	默认值
1	id	编号	int	11	是	
2	targetid	目标 id	int	11		
3	targetname	目标名称	varchar	255		
4	hyid	会员 id	int	11		
5	sctime	收藏时间	datetime			
6	tupian	图片	varchar	255		
7	href	目标路径	varchar	255		

表 9 xinxi

编号	字段名称	字段含义	字段类型	字段长度	是否主键	默认值
1	id	编号	int	11	是	
2	againstcount	点踩数	int	11		
3	agreecount	点赞数	int	11		
4	clickcount	点击数	int	11		
5	dcontent	资讯内容	longtext			
6	lmid	栏目 id	int	11		
7	pubren	发布者	varchar	255		
8	pubtime	发布时间	datetime			
9	title	标题	varchar	255		
10	tuijian	是否推荐	int	11		
11	tupian	图片	varchar	500		
12	state	状态	int	11		
13	subtypeid	子项目 id	int	11		
14	tagid	目标 id	int	11		

3.4 用户 ui 界面设计

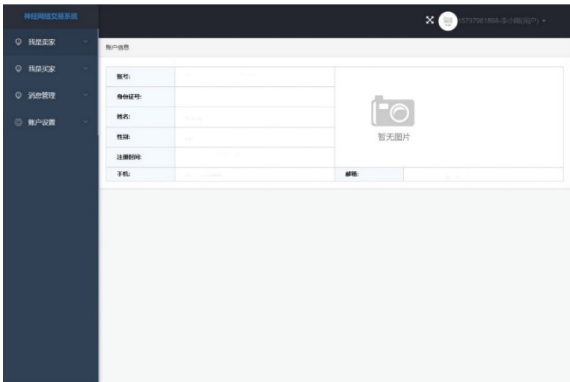


图 5 用户 ui 界面

用户 ui 界面采用较为经典的左右布局，左侧为功能栏，功能栏的菜单模式采用垂直菜单，垂直菜单内嵌下拉式子菜单；右侧显示点击子菜单后的具体页面内容。本设计图展示的为账户信息功能的设计，右侧依次为账号、身份证号、姓名、性别、注册时间、手机、邮箱、头像。

3.5 静态页面设计

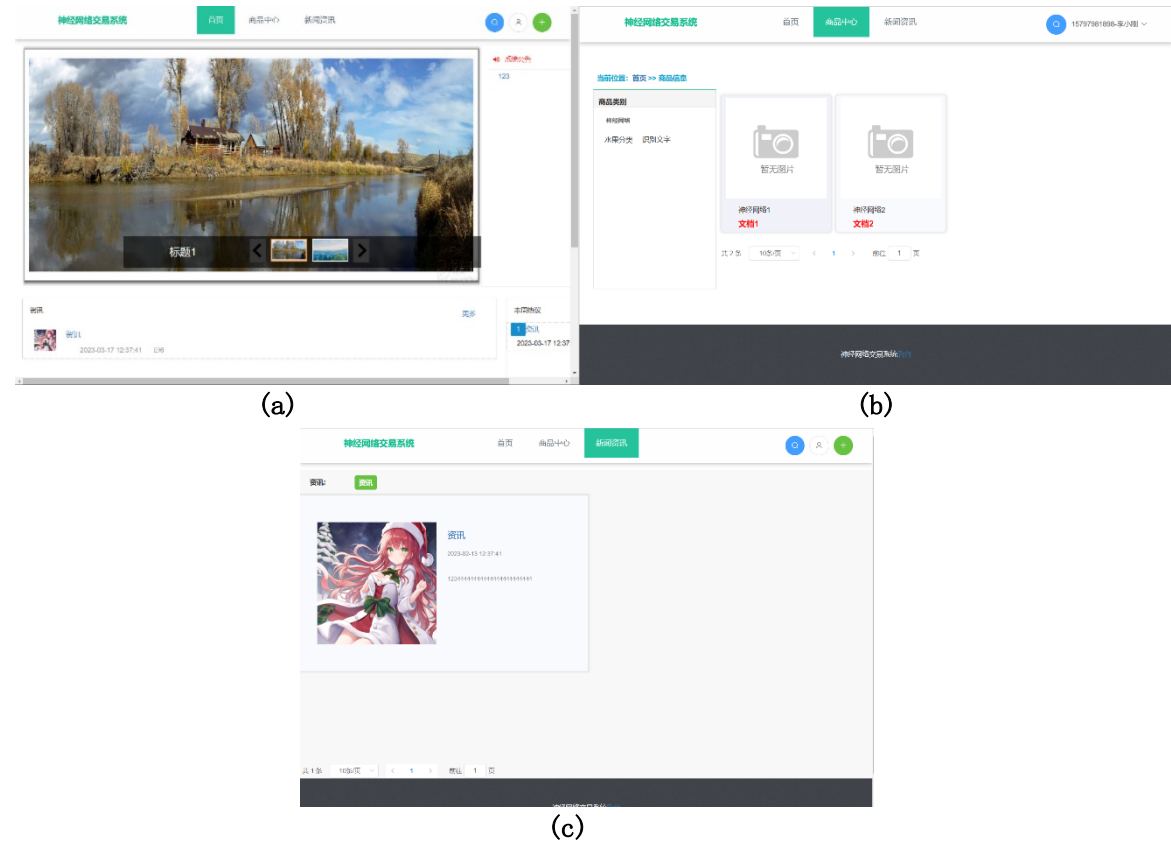


图 6 静态页面

首页页面如图 6(a) 所示，总体采用上中下结构，顶部从左至右依次为网站名称、菜单栏、查询、用户登录、用户注册；中部焦点图占用较大面积，主要用途为广告轮

播，焦点图下方则是资讯栏，点击即可查看管理员发布的资讯，焦点图右侧为系统公告，系统公告为红色，更加显眼，点击便可查看管理员发布的公告。

商品中心页面模块如图 6（b）所示，采用左右结构，左侧为商品的总体分类以及对应总体分类的子分类，可通过点击分类跳转，右侧于第一次进入商城时显示所有商品，点击具体分类后将切换为具体分类下的对应商品。

新闻资讯页面如图 6（c）所示，所有资讯呈一列两条排列，各资讯的左侧为该咨询所配的图片、右侧为资讯的标题、发表时间、具体内容。

3.6 用例图

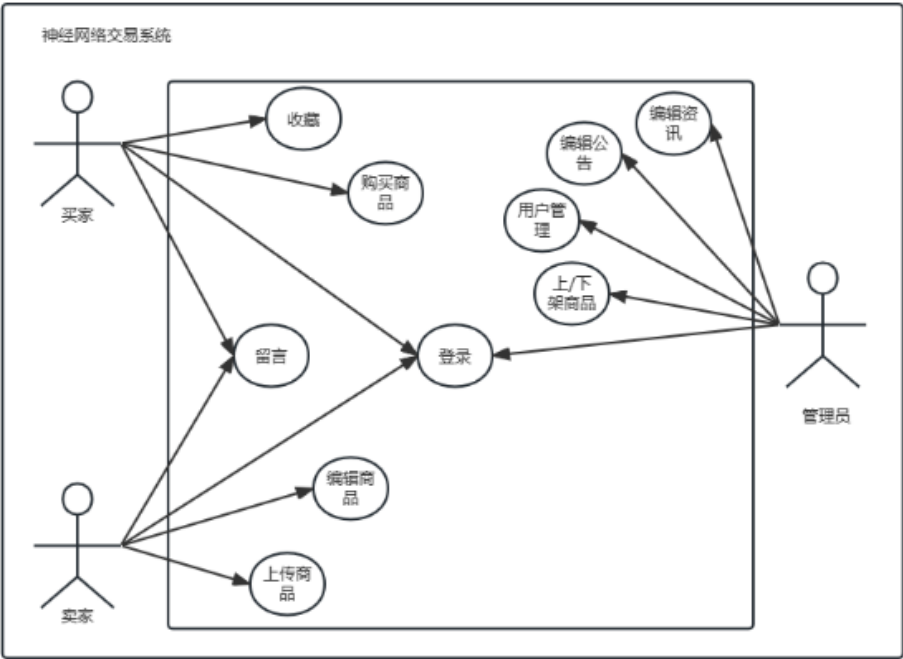


图 7 用例图

系统的用户由三类参与者组成：买家、卖家以及管理员。管理员主要负责编辑咨询、编辑公告、用户管理以及上/下架商品；买家可以使用的功能有登录、收藏（商品、资讯等）、购买商品以及留言；卖家可以使用的功能有登录、上传商品、编辑商品以及留言。

4 基本功能

基本功能主要通过买卖功能、管理员系统以及收藏与留言系统来展示。

4.1 买卖功能

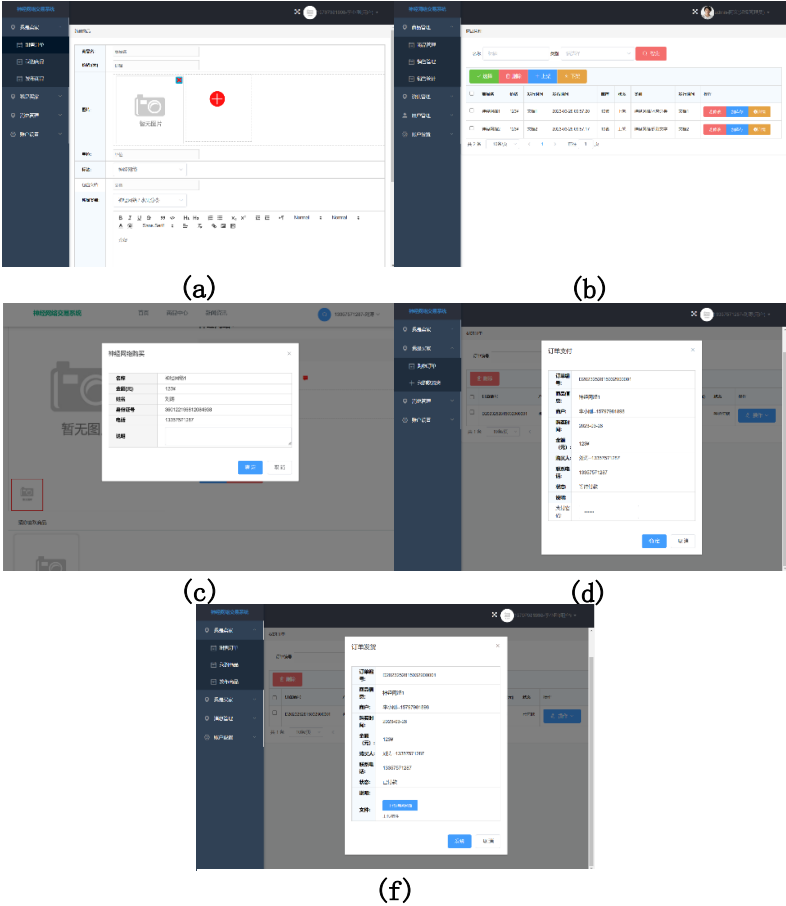


图 8 买卖流程

买卖功能如图 10 (a) – (f) 所示，首先，卖家点击发布商品后进入新建商品界面，在填写完商品参数后点击提交按钮；其次，管理员通过管理员后台审核卖家新建的物品，审核完成后上架；再次，买家可在系统中看到卖家发布的神经网络商品，点击购买并付款；然后，卖家通过卖家后台上传神经网络并点击发货；最后，买家通过买家后台订单系统下载神经网络，至此，交易完成。

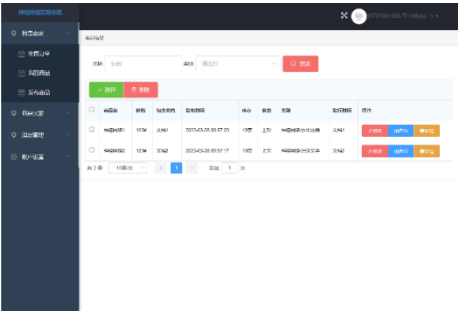


图 9 上架总览

卖家可以在卖家后台系统查看自己上架的商品，并对商品详细数据进行修改、加库存、查看详情等操作，如图 11 所示。

## 4.2 管理员系统

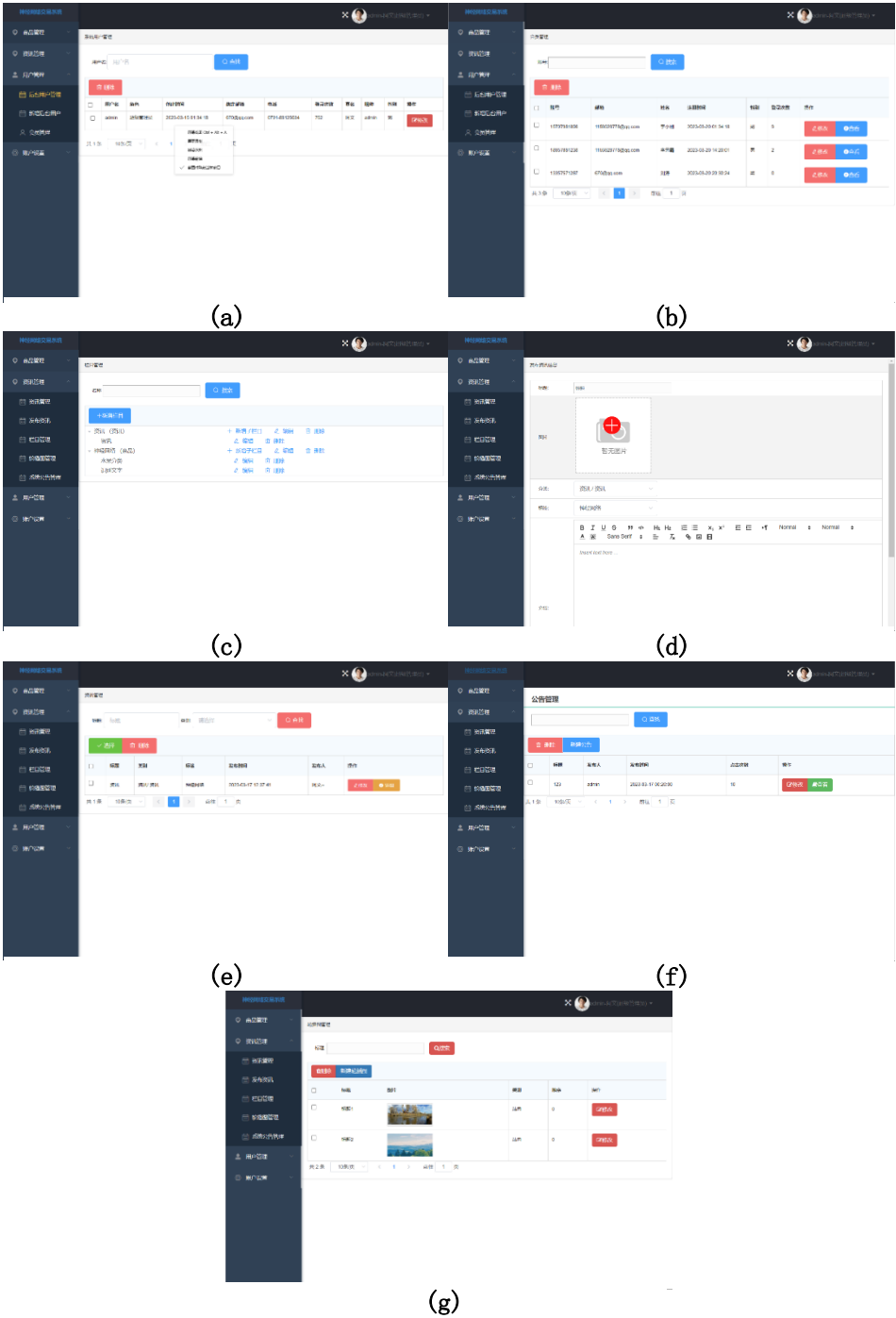


图 10 管理员系统功能

超级管理员的特殊权限有修改超级管理员的账户信息；对系统的普通管理员的数量进行增加或者删除；对普通管理员的信息进行修改，如图 12 (a) 所示。

管理员可以修改会员的账户信息，还可以删除会员账号，如图 12 (b) 所示；对

资讯以及商品分类栏目数量进行增加或者删除，如图 12（c）所示；可以在系统内发布资讯，并对已有资讯进行编辑或者删除，如图 12（d）（e）所示；能够在系统中新建公告以及对已有公告进行编辑或删除，如图 12（f）；可以对首页轮播图的详细信息进行修改，同时也可以新建轮播图或者删除已有轮播图，如图 12（g）所示。

### 4.3 收藏与留言系统

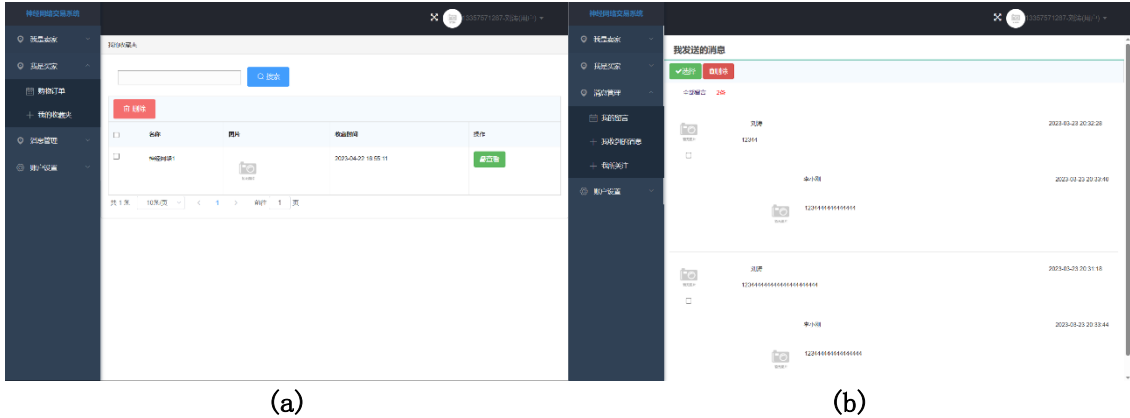



图 11 收藏与留言系统

为防止系统内总商品数量过大导致的用户难以找到先前查找到的心仪商品，添加了收藏系统来帮助用户快速定位，如图 13（a）所示；为保证买卖双方的有效沟通，在系统内添加了留言系统，提高买卖双方的沟通效率，如图 13（b）所示。



## 5 系统神经网络精确度测试功能

### 5.1 神经网络精确度测试的基本流程

神经网络的训练以及测试精确度的流程较为固定。以下流程以图像识别类神经网络的训练及测试展开讨论。

首先，需要获取所需的参数，如神经网络模型路径、学习率、数据集路径、GPU 设置（若 GPU 无 CUDA 环境则只能使用 CPU 来运行）等，这些参数均通过终端命令行输入，其余尚未定义的参数均采用默认值。

其次，根据数据集的不同可分为两种情况：第一种便是直接使用 python 库自带的数据集，如 MNIST、CIFAR-10 等，这一类数据集已有完善的函数可供调用，直接调用函数即可；第二种则是使用自己的数据集，这一情况往往需要对数据集内的数据进行预处理，较为重要的一步则是使图片的格式符合神经网络的输入格式。此处以 VGG 网络为例，VGG 网络的输入要求是  $224*224*3$ ，限制图像尺寸在图像分类的神经网络中是十分常见的，这样做的原因是：更大的图片需要使用更大的卷积层深度来减小输出层的参数，但深度过高将会导致模型退化，而图像分类只需要提取到特定的特征即可，因此，图片大小并不影响神经网络进行特征的提取，使用更小的图片不仅能减小运算，还能够降低神经网络模型复杂度，降低过拟合的风险。对图像尺寸进行预处理后，则需要将这些图像转变为数据集，较为常用的方法便是将不同种类的物品的图像存放于不同的文件夹，并编号，文件夹以文件夹内存放的内容来命名。

然后，需要初始化神经网络模型和加载数据集，训练过程则是一个循环，该循环遍历训练集内所有数据，在每一代中正向传播，通过损失函数计算差异，反向传播，计算梯度，并更具梯度更新网络的参数。测试过程则相对简单，仅需要将数据传入神经网络并将输出与答案对比，并计算精确度即可。

以上为神经网络训练以及测试的基本流程，本系统内有关神经网络精确度测试模块的后端代码根据该流程编写并调用。

### 5.2 神经网络精确度测试在系统内的实现

用户上传的数据集必须为.zip 后缀的压缩包，数据集内的所有图片均为.jpg 格式。压缩包内包含训练集以及测试集，数据集内需要通过文件夹来对所需要测试的数据进行分类，并对分类名进行定义，对各分类内的数据进行编号。

Java 前端通过 POST 命令将订单编号以及测试集路径发送至后端，后端通过订单编号查找神经网络路径，并通过终端指令“python D:\Projects\test\mission.py --dataset 数据集路径 --model 神经网络路径”调用 mission.py 这一测试程序。Python 程序通过 get\_args 方法获取到终端指令内的数据集路径以及神经网络路径，并使用 shutil.copy 函数将数据集以及神经网络各自从原文件夹复制到专门负责处理测试需求的文件夹内，尽量避免因为路径可能导致的问题。接着，使用 zipfile.extractall(r'.')函数来解压数据集到当前文件夹，由于解压缩后的文件名可能由于编码问题导致乱码，此处需要对压缩包内的文件名进行预处理，即先使用 CP437 格式编码，再通过 GBK 格式解码，保证文件名无乱码。然后便是初始化神经网络，在训练过程中计算损失函数，并通过 SGD（随机梯度下降）优化，此方法内的学习率设置为 0.001，动量设置为 0.9，并在每一代输出损失值。最后，由于 Python 程序的逐行输出，Java 程序内需要使用 StringBuilder 来储存输出，并将测试结果发回前端，展示给用户，如图 14 所示。

神经网络:	Net.py
说明:	
数据集:	<div>上传数据集</div> <div>上传附件</div> <div>data1.zip</div> <div>测试</div>
测试结果:	[1,10]loss:2.305;[1,20]loss:2.125;[2,10]loss:1.958;[2,20]loss:1.765;Finished Training;Accuracy of the network:23%

图 12

## 6 改进方案

### 6.1 功能性需求

#### 6.1.1 测试能力扩展

目前该测试工具仅能测试图像分类型神经网络，无法满足更多测试需求，如音频转文字类神经网络等，需要查阅资料并在 Python 测试文件内加入更多分支以完善测试功能。

#### 6.1.2 优化留言功能

目前的留言功能在买卖双方略微沟通后便会产生较多的冗余信息，导致买卖双方翻阅留言记录繁琐，重要信息难以找到等缺点，日后查阅资料，争取能够达到类似即时通讯工具的留言界面。

### 6.2 非功能性需求

#### 6.2.1 性能需求

目前所有的测试均为本地单用户无其他压力的环境下进行，各场景平均响应时间较快。但还未考虑到高并发环境下的性能测试，在添加高并发后可能会发生短链接完全占用端口等报错问题，需要设置 Tomcat 最大线程连接数、等待超时的阈值等。

#### 6.2.2 安全性需求

目前用户在身份验证以及账户信息等传输过程中并未加密，可能导致用户的隐私信息泄露，资产被盗用等问题。

对于敏感数据，如姓名、手机号、身份证号、密码等，可以通过对这些数据进行脱敏来达到保护用户隐私的效果。常用的方法有：截断法，即使用“\*”替代真实值中间的一部分；对称加密，即可逆的脱敏方法，通过加密密钥和算法进行加密，但需要对密钥的安全性进行设计。

## 7 参考文献

- [1] 张建华,孔繁涛,吴建寨等.基于改进 VGG 卷积神经网络的棉花病害识别模型[J].中国农业大学学报,2018,23(11):161-171.辜瑞帆,李祥,任维民.基于 ResNet50 改进模型的图像分类研究[J].现代电子技术,2023,46(04):107-112.DOI:10.16652/j.issn.1004-373x.2023.04.020.
- [2] 王羽微,程远,毕海等.基于深度学习 VGG 网络模型的海洋单细胞藻类识别算法[J].大连海洋大学学报,2021,36(02):334-339.DOI:10.16535/j.cnki.dlhyxb.2020-161.
- [3] 柯溢,卢亚玲.基于 VGG16 的宠物犬识别方法[J].电子制作,2020,No.407(21):42-45.DOI:10.16589/j.cnki.cn11-3571/tn.2020.21.016.
- [4] 钱蓉,孔娟娟,朱静波等.基于 VGG-16 卷积神经网络的水稻害虫智能识别研究[J].安徽农业科学,2020,48(05):235-238.
- [5] 徐佳琦.基于边缘提取与 VGG16 深度卷积神经网络的混凝土裂缝识别方法研究[J/OL].施工技术(中英文):1-6[2023-04-17].<http://kns.cnki.net/kcms/detail/10.1768.TU.20230413.1812.014.html>.
- [6] 廖璐明,张伟.基于改进 VGG16 网络的混合批量训练交通标志识别[J].电子科技,2021,34(08):8-13.DOI:10.16180/j.cnki.issn1007-7820.2021.08.002.
- [7] 曹宇,邢素霞,逢键梁等.基于改进的 VGG-16 卷积神经网络的肺结节检测[J].中国医学物理学杂志,2020,37(07):940-944.
- [8] 徐旭东,马立乾.基于迁移学习和卷积神经网络的控制图识别[J].计算机应用,2018,38(S2):290-295.
- [9] 张雷,鲍蓉,朱永红等.基于 CSA-ResNet 的人员入侵检测方法[J].计算机工程,2023,49(04):297-302+311.DOI:10.19678/j.issn.1000-3428.0066093.
- [10] 金张根,曹杨,于红绯等.基于 ResNet-18 网络的城市生活垃圾识别方法研究[J].现代计算机,2023,29(02):73-77.
- [11] 廖明霜,罗远远.基于 ResNet 对花朵分类研究[J].农业与技术,2023,43(02):65-68.DOI:10.19754/j.nyyjs.20230130016.
- [12] 牛智有,于重洋,吴志陶等.基于改进 ResNet18 模型的饲料原料种类识别方法[J].农业机械学报,2023,54(02):378-385+402.
- [13] 张媛,陈西曲.基于改进 ResNet34 模型的杂草识别方法[J].武汉轻工大学学报,2023,42(01):86-94.
- [14] 王斌,吴晓红,辜蕊等.基于改进 ResNet 的阿尔兹海默症分类网络[J].智能计算机与应

- 用,2023,13(03):69-76+82.
- [15] 张文景,蒋泽中,秦立峰.基于弱监督下改进的 CBAM-ResNet18 模型识别苹果多种叶部病害[J/OL].智慧农业(中英文):1-11[2023-04-17].<http://kns.cnki.net/kcms/detail/10.1681.S.20230413.1825.002.html>.
- [16] 朱家辉,苏维均,于重重.基于卷积神经网络的农作物叶片病害诊断[J].植物检疫,2021,35(02):28-32.DOI:10.19662/j.cnki.issn1005-2755.2020.00.042.
- [17] F. Liang, W. Yu, D. An, Q. Yang, X. Fu and W. Zhao, "A Survey on Big Data Market: Pricing, Trading and Protection," in IEEE Access, vol. 6, pp. 15132-15154, 2018, doi: 10.1109/ACCESS.2018.2806881.
- [18] L. Tian, J. Li, W. Li, B. Ramesh and Z. Cai, "Optimal Contract-Based Mechanisms for Online Data Trading Markets," in IEEE Internet of Things Journal, vol. 6, no. 5, pp. 7800-7810, Oct. 2019, doi: 10.1109/JIOT.2019.2902528.
- [19] 周东宁. 卷积神经网络的鲁棒性分析[D].大连理工大学,2022.DOI:10.26991/d.cnki.gdllu.2022.001995.
- [20] 王赞,闫明,刘爽等.深度神经网络测试研究综述[J].软件学报,2020,31(05):1255-1275.DOI:10.13328/j.cnki.jos.005951.
- [21] 张峰.应用 SpringBoot 改变 web 应用开发模式[J].科技创新与应用,2017,No.207(23):193-194.
- [22] 吕宇琛.SpringBoot 框架在 web 应用开发中的探讨[J].科技创新导报,2018,15(08):168+173.DOI:10.16660/j.cnki.1674-098X.2018.08.168.
- [23] 陈怿. 数据库范式分解理论研究[D].吉林大学,2008.

## 8 致谢

感谢唐茗老师对我的课题的耐心指导以及提出的宝贵意见和建议，感谢张进老师对我的论文修改提出的宝贵意见和建议，感谢所有授课老师，感谢支持我的家人和朋友。