

中国科学技术大学

# 专业硕士学位论文

(工程硕士)



## 基于深度学习的超导瞬变电磁 去噪系统设计与实现

作者姓名： 王钰镗

专业领域： 软件工程

校内导师： 余艳玮 博士

企业导师： 荣亮亮 博士

完成时间： 二〇二三年二月一日



University of Science and Technology of China  
A dissertation for master's degree

(Master of Engineering)



**Design and Implementation of  
Superconducting Transient  
Electromagnetic Denoising System  
Based on Deep Learning**

Author: Wang Yutang

Speciality: Software Engineering

Supervisor: Dr. Yu Yanwei

Advisor: Dr. Rong Liangliang

Finished time: February 1, 2023



## 中国科学技术大学学位论文原创性声明

本人声明所呈交的学位论文，是本人在导师指导下进行研究工作所取得的成果。除已特别加以标注和致谢的地方外，论文中不包含任何他人已经发表或撰写过的研究成果。与我一同工作的同志对本研究所做的贡献均已在论文中作了明确的说明。

作者签名：\_\_\_\_\_

签字日期：\_\_\_\_\_

## 中国科学技术大学学位论文授权使用声明

作为申请学位的条件之一，学位论文著作权拥有者授权中国科学技术大学拥有学位论文的部分使用权，即：学校有权按有关规定向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅，可以将学位论文编入《中国学位论文全文数据库》等有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。本人提交的电子文档的内容和纸质论文的内容相一致。

控阅的学位论文在解密后也遵守此规定。

☒ 公开   ☐ 控阅（\_\_\_\_年）

作者签名：\_\_\_\_\_

导师签名：\_\_\_\_\_

签字日期：\_\_\_\_\_

签字日期：\_\_\_\_\_



## 摘 要

随着我国社会和经济的持续高速发展，对矿产资源的需求进一步的扩大。我国虽我幅员辽阔，但矿产资源人均占有率远低于世界平均人均占有率，且在过去几十年的发展中地表浅层的易探勘矿藏大多已被消耗殆尽。人们不得不研发高性能的探矿设备从而转向更为复杂的深部矿产资源勘探。其中超导瞬变电磁法应其具有灵敏度高，勘探深度大，分辨率高等优势被广泛应用到工程地质勘探等领域。然而在实际应用中，实测的数据包含大量的噪声，严重影响测量数据质量和可靠性。传统的去噪方法需要预知噪声分布并手动选择参数，过程耗时复杂无法满足野外测量电磁信号时需要实时处理数据的需求。随着人工智能与大数据技术的发展，深度学习算法被广泛应用到电磁信号去噪任务中，相比于传统去噪方法深度学习算法无需手动设置参数，无需预知噪声分布，可以实时得到处理结果。

基于此本文以野外数据采集环境下采集人员需要频繁进行数据采集和检测为背景，通过 TensorFlow 深度学习框架研发一个基于 LSTM-CNN 结构的深度学习去噪算法，借助一些开源工具开发了一套基于深度学习的超导瞬变电磁去噪系统。其中，系统包含针对普通用户的数据上传、预处理、模型去噪、数据可视化、数据提交五个功能模块以及针对管理员用户的模型训练、在线预测、用户管理、数据管理四个功能模块。最后根据实际的使用场景，针对本系统的功能和性能进行了测试，并满足了所提出的实际需求。

本课题的意义主要有两点，一是通过参考已有的深度学习电磁信号去噪网络，针对灵敏度更高的超导瞬变电磁信号设计了一个新的深度学习去噪算法，最大限度的发挥了高分辨率信号的优势，促进了深度学习方法在瞬变电磁数据去噪方面的应用。二是设计了一个功能强大电磁信号去噪系统，允许用户对测得的电磁信号进行管理，并根据测量环境灵活适配不同的预设模型进行实时去噪操作，提升了野外实测数据的质量，减少野外矿产资源勘探人员的重复性工作。

**关键词：**深度学习    去噪    超导瞬变电磁

## **ABSTRACT**

With the continuous rapid development of China's society and economy, the demand for mineral resources has further expanded. Although our country is vast, the per capita occupancy rate of mineral resources is much lower than the world average per capita occupancy rate, and in the past decades of development in the surface of the shallow layer of easy to explore mineral deposits have been depleted. People have to develop high performance prospecting equipment so as to turn to more complex deep mineral resources exploration. Among them, superconducting transient electromagnetic method has been widely used in engineering geological exploration due to its high sensitivity, large exploration depth and high resolution. However, in practical applications, the measured data contain a large amount of noise, which seriously affects the quality and reliability of the measured data. The traditional denoising method needs to anticipate the noise distribution and select parameters manually, which is a time-consuming and complicated process that cannot meet the demand for real-time data processing when measuring electromagnetic signals in the field. With the development of artificial intelligence and big data technology, deep learning algorithms are widely used in electromagnetic signal denoising tasks. Compared with traditional denoising methods, deep learning algorithms do not need to manually set parameters and do not need to predict the noise distribution, and can obtain processing results in real time.

Based on this paper, we develop a deep learning denoising algorithm based on LSTM-CNN structure through TensorFlow framework with the background of field data collection environment where collectors need to collect and detect data frequently, and develop a deep learning based superconducting transient electromagnetic denoising system with the help of some open source tools. The system includes five functional modules: data upload, pre-processing, model denoising, result display, and data submission for general users and four functional modules: model training, online prediction, user management, and data management for administrator users. Finally, according to the actual usage scenarios, the functions and performance of this system are tested and meet the proposed practical requirements.

One is to design a new deep learning denoising algorithm for superconducting transient electromagnetic signals with higher sensitivity by referring to existing deep learning electromagnetic signal denoising networks, which maximizes the advantages of high-resolution signals and promotes the application of deep learning methods for



transient electromagnetic data denoising. Second, a powerful electromagnetic signal denoising system is designed to allow users to manage the measured electromagnetic signals and flexibly adapt different preset models for real-time denoising operations according to the measurement environment, which improves the quality of field measurement data and reduces the repetitive work of field mineral resource exploration personnel.

**Key Words:** Deep Learning, Denoising, Superconducting Transient Electromagnetic



## 目 录

摘要	I
ABSTRACT	II
第 1 章 绪论	1
1.1 研究背景和意义	1
1.2 国内外研究现状	2
1.2.1 去噪技术研究现状	2
1.2.2 电磁信号去噪系统研究现状	3
1.3 课题的先进性和实用性	3
1.4 本文主要工作	4
1.5 本文组织结构	4
第 2 章 相关技术介绍	5
2.1 Python 编程语言	5
2.2 神经网络	5
2.2.1 卷积神经网络	5
2.2.2 长短期记忆网络	5
2.3 TensorFlow 框架	6
2.4 PyQt 框架	6
2.5 Django 框架	6
2.6 MySQL 数据库介绍	7
第 3 章 需求分析	8
3.1 概述	8
3.2 功能需求	8
3.3 性能需求	11
3.4 其他需求	12
3.5 本章小结	12
第 4 章 系统设计	13
4.1 系统概述	13
4.2 系统结构设计	13
4.2.1 系统架构与工作逻辑	13
4.2.2 系统功能模块	16

4.3 功能模块详细设计	18
4.3.1 数据录入模块设计	18
4.3.2 预处理模块设计	19
4.3.3 去噪模块设计	22
4.3.4 数据可视化模块设计	23
4.3.5 数据提交与模型下载模块设计	24
4.3.6 模型训练模块设计	25
4.3.7 在线预测模块设计	28
4.3.8 用户管理模块设计	29
4.3.9 数据管理模块设计	29
4.4 数据库详细设计	29
4.4.1 用户信息表	30
4.4.2 管理员信息表	30
4.4.3 模型信息表	31
4.4.4 数据集信息表	31
4.5 本章小结	32
第 5 章 系统实现	33
5.1 开发环境	33
5.2 数据录入模块	33
5.3 预处理模块	35
5.4 去噪模块	38
5.5 数据可视化模块	41
5.6 模型训练模块	41
5.7 在线预测模块	48
5.8 数据管理模块	48
5.9 本章小结	49
第 6 章 系统测试	50
6.1 测试方法与环境	50
6.1.1 测试方法	50
6.1.2 测试环境	50
6.2 系统测试用例	50
6.2.1 算法测试	50
6.2.2 数据录入模块测试	55
6.2.3 预处理模块测试	55

6.2.4 去噪模块测试 . . . . .	55
6.2.5 模型训练模块测试 . . . . .	56
6.2.6 用户管理模块测试 . . . . .	57
6.3 测试结果 . . . . .	57
第 7 章 总结与展望 . . . . .	58
7.1 总结 . . . . .	58
7.2 展望 . . . . .	59
参考文献 . . . . .	60
致谢 . . . . .	63



## 第1章 绪论

### 1.1 研究背景和意义

随着我国社会和经济的持续高速发展,对矿产资源的需求进一步的扩大。我国幅员辽阔,矿产资源丰富,矿产资源种类繁多(168种),储量占世界矿产总之的14.6%,但是我国的人均占有率远低于世界平均人均占有率<sup>[1]</sup>。为了摆脱这一困境,人们不得不研发高性能的探矿设备从而转向更为复杂的深部矿产资源勘探。在深层探矿中,瞬变电磁法(Transient Electromagnetic Method, TEM)具有低阻电性分辨能力高,异常响应形态简单,探测范围大等优点,被广泛应用到工程地质勘探等领域。勘探深度依赖于勘探设备的灵敏度,探测深度理论上最深可以达到几千米<sup>[2]</sup>。超导瞬变电磁法区别于传统瞬变电磁法,使用超导量子干涉器件(Superconducting Quantum Interference Device, SQUID)代替感应线圈作为接收装置。超导TEM系统具有本征噪声低,带宽大的特点。由于SQUID灵敏度的优势,超导瞬变电磁法普遍用于大深度勘探,探测深度达3000m以上。瞬变电磁法的测量原理是利用不接地回线(或电偶源)向地下发送一次脉冲磁场,在磁场的激励下,地下目标产生二次场。二次场不会立即消失,它会按照指数规律来随时间衰减。在地表用接收机接收二次场,该二次磁场的变化,将反映地下介质的电性情况<sup>[3]</sup>。然而接收信号的质量不可避免地会受到退化问题的影响,这是由信号传输和采集过程中设备的环境背景噪声、人为噪声和电子噪声造成的。对退化的TEM信号进行有效去噪对于保证探测的准确性和提高探测深度具有重要意义<sup>[4]</sup>。

瞬变电磁法具有观测区域广,采集数据量大等优点。但电磁数据易受噪声干扰,特别是包含深层地下信息的电磁数据信号更弱,受噪声影响更严重。传统的电磁数据去噪方法大多基于滤波算法如小波变换和卡尔曼滤波以及各种滤波算法变体如小波阈值-指数自适应窗宽滤波器和指数拟合-自适应卡尔曼滤波器。这些方法的去噪效果可以接受但大都需要手动选择参数,预知噪声的一些先验信息,这导致了传统去噪方法需要大量人为调整,耗时长,效率低<sup>[4]</sup>。

近年来,深度学习(Deep Learning, DL)在非常多的科学领域中迅速崛起,在地球物理学界引来了越来越多的关注,并在近几十年来广泛的运用在地球物理学界的各个方面,比如,地震相分类<sup>[5]</sup>、勘探地球物理<sup>[6]</sup>、余震模式分析<sup>[7]</sup>。深度学习方法需要大量的数据,它允许在不事先指定特定模型的情况下利用数据中的非线性依赖关系。通过利用不同的抽象层,深度学习能够发现数据中的低级和高级特征。深度学习方法在TEM去噪领域已经取得一定效果并具有如下优势:①不需要手动的选择去噪算法的参数;②不需要预知噪声的分布。

超导瞬变电磁法中, SQUID 相比传统线圈, 具有灵敏度高、带宽大、噪声低的优势, 相比于一般瞬变电磁法, 能提供更精确, 信噪比更高的数据。然而, 由于在野外实测数据时恶劣的操作条件使得数据缺乏可靠性。现有的基于深度学习的去噪算法难以充分发挥 SQUID 传感器的优势, 需要一个性能更好更适用于超导传感器的去噪方法来发挥发挥 SQUID 传感器的优势。基于深度学习的瞬变电磁去噪方法从不同的思路提供了提升数据质量的方法, 但基于深度学习的电磁信号去噪平台很少, 大多数数据的去噪流程都是嵌入在一个更大的数据解释软件中, 或者将数据单独分离出来进行去噪操作。而基于深度学习的去噪算法开发流程需要生成大量的数据并进行大量的验证工作以及去噪后效果的评估工作, 数据单独分离会增加数据管理和验证算法准确性的难度, 现有的数据解释软件无法满足研究算法的需求, 建立一个专门的针对去噪算法的研究与验证系统会极大的简化去噪算法的开发流程, 最终促进超导瞬变电磁勘探技术的实用化以及产业化。

## 1.2 国内外研究现状

### 1.2.1 去噪技术研究现状

传统的去噪方法包括: 三点滤波、中值去噪法、最小二乘拟合法、三点指数逼近非线性平滑等, 这些方法都是基于经典滤波理论的信号与噪声谱不重叠的原理, 仅在信号谱重叠不严重的场合简单有效, 有一定的局限性<sup>[8]</sup>, 在现代信号处理中, 出现了一种更先进的小波分析法, 它是一种时间-尺度分析方法, 具有多分辨率分析的特点。小波变换由于能同时在时域和频域对信号进行分析, 所以它能有效地区分信号中的突变部分与噪声<sup>[9]</sup>。在小波变换和卡尔曼滤波器的基础上还衍生出了一系列变体, 例如小波阈值-指数自适应窗宽滤波器 (WEF)<sup>[10]</sup>和指数拟合-自适应卡尔曼滤波器 (EF-AKF)<sup>[11]</sup>。

随着机器学习和深度学习在各个领域的兴起, 利用深层神经网络对的深度学习学习方法对瞬变电磁信号进行去噪在最近几年开始发展。例如基于自编码的二次场信号去噪堆叠自动编码器 (SFSDSA)<sup>[12]</sup>, 基于 CNN 的信号到图像转换的瞬态电磁信号深度去噪网络 (TEMDnet)<sup>[4]</sup>, 基于长短期记忆自动编码器的瞬变电磁数据去噪网络 (LSTM-autoencoder)<sup>[13]</sup>, 基于 GAN 的具有噪声学习的瞬态电磁信号深度去噪网络 (TEM-NLnet)<sup>[14]</sup>以及基于最小噪声分数-深度神经网络的瞬态电磁数据去噪网络<sup>[15]</sup>。在大多情况下, 与传统的基于滤波的方法相比, 它们取得了更好的性能。

在国内外, 研究人员都已经在尝试将深度学习技术应用于瞬变电磁信号去噪任务, 但现有的深度学习方法虽然相比与传统方法表现更好, 在面对实测数据



时依然存在还原度不够以及精度不足等问题。特别是面对灵敏度更高的超导瞬变电磁测量方法时，这些不足会被进一步放大，距离实际应用还有一定差距。

### 1.2.2 电磁信号去噪系统研究现状

针对电磁数据处理，国外的一些软件系统如 MTSOFT-2D、Aarhus Workbench 等功能强大，专业齐全，可实现对数据进行解释和可视化。但这些软件大多专注与数据的解释与成像，去噪只作为其数据预处理流程中的一个附带功能。市面上目前并没有一个基于深度学习去噪方法的，针对超导瞬变电磁数据的去噪系统。鉴于此，本课题在国内外相关研究的基础上，设计一个能够保证准确性和鲁棒性的基于深度学习的超导瞬变电磁数据去噪算法，运用此算法，建立一个能够完成瞬变电磁信号实时去噪任务的系统。

## 1.3 课题的先进性和实用性

基于深度学习的超导瞬变电磁去噪系统不同于市面上运用传统去噪算法的电磁数据处理软件，是一个集成深度学习算法的去噪系统。传统的去噪方法通常需要噪声数据的频率和幅值等先验信息，这些方法需要大量的人为调试且面对特征较复杂的噪声时通用性不足，且传统去噪方法发展成熟，很难有突破性的进展。基于深度学习的去噪方法成为研究电磁去噪技术的新方向。深度学习算法直接通过对大量数据的学习，提取出噪声的低级高级特征，相比于传统去噪方法，无需人为设置初始参数，有着良好的全局寻优性能，处理数据时不需要知道先验信息，计算速度快。避免了大量的人为调试工作，大大提高了去噪效率，并使得实时去噪成为可能。在未来甚至可以通过多传感器融合等算法来进一步的分离噪声的特征。本文所使用到的深度学习算法使用仿真数据集开发，基于紫金矿业集团有限公司的大量实际矿产勘探数据验证优化。并且，市面上暂时没有针对超导瞬变电磁信号去噪的软件系统，本系统通过使用针对超导电磁信号优化的深度学习去噪算法可以充分发挥超导瞬变电磁法反演精度高、分辨率高、深度大的优势。同时利用深度学习的去噪方法可以实时给出数据的去噪结果，会为野外探测人员节省大量的重复工作。除此之外本系统将算法研发中的数据集管理，模型训练，效果评估等流程集合到一个平台加快了去噪算法迭代速度并减少大量数据管理的复杂度。

## 1.4 本文主要工作

本课题主要基于深度学习技术，在研究深度学习、地球物理相关理论和技术的基础上，研发一个去噪算法并设计一个专用于超导瞬变电磁的去噪系统，用于实际瞬变电磁数据的数据去噪任务。课题包含以下方面的内容：

(1) 研究瞬变电磁法、超导瞬变电磁法、传统去噪方法、深度学习算法的理论知识、国内外发展现状和发展趋势，对系统进行需求分析，明确课题的实现目标。

(2) 研究本系统的深度学习模型算法，设计模型所使用的数据集，分析不同类型的噪声对算法准确性、鲁棒性的影响，设计适合本系统的深度学习网络结构。

(3) 完成系统架构的设计，包括系统架构、系统功能和存储结构的设计。

(4) 完成模块的详细设计与实现，部署好开发环境，将深度学习算法集成到系统中，完成代码框架的编写。

(5) 分别使用仿真数据和实测数据对本课题所设计的超导瞬变电磁去噪系统进行全面测试和改进，提高系统的稳定性和可靠性。

## 1.5 本文组织结构

本文开发了一个专门针对超导瞬变电磁数据设计的去噪系统，系统的主要去噪算法基于 LSTM-CNN 结构并使用 TensorFlow 开源深度框架搭建。系统可以实时得到去噪结果，应用于实际矿产勘探环境中，为超导瞬变电磁勘探过程提供软件和算法支持并减少野外探测工作量。

首先确定用户需求，通过需求分析确定用户的行为与类型。在此基础上完成系统的概要设计，明确系统涉及到的功能以及对应接口，明确各个模块之间的逻辑关系。在详细设计与实现时采用合适的开发方法，编写功能代码并对模块进行封装，完善细节功能，最后将整合并集成到系统中，完成系统开发工作。

在系统开发完成后进行系统部署并对系统模块和去噪算法进行测试，确定系统的性能、可靠性和去噪的准度都达到设计标准。

## 第 2 章 相关技术介绍

### 2.1 Python 编程语言

本系统开发时采用面向对象 (Object-oriented) 和模块化的方法, 而 Python 是一个非常流行的面向对象、解释型程序设计语言, 用其编写的程序可以在大多数的主流平台运行, 并且具有面向对象的特性, 比如类、封装、继承、多态等足以胜任复杂逻辑的编写和大型程序的开发。同时 Python 提供了丰富的 API(应用程序编程接口) 和丰富强大的程序包模块 (Package Module, 也称为软件包模块), 可以完成从 Web 开发、后台搭建、GUI(图形用户界面) 开发等一系列工作<sup>[16]</sup>。同时本系统开发和运行时涉及大量的数据计算功能且会使用复杂的深度学习算法而 Python 拥有许多优秀的数据处理库如 Pandas 和 NumPy, 对常用的开源深度学习框架如 TensorFlow, PyTorch 也有很好的支持。系统最后要向用户展示数据处理结果, 可以用 Python 对测试结果进行统计分析<sup>[17]</sup>, 如用 Matplotlib 进行数据曲线绘制等<sup>[18]</sup>。基于以上特点选择 Python 作为本系统的主要开发语言。

### 2.2 神经网络

#### 2.2.1 卷积神经网络

卷积神经网络简称 CNN(Convolutional Neural Networks), 是目前广泛应用的神经网络结构之一。卷积神经网络通过权值共享和局部连接等操作<sup>[19]</sup>, 无需设计某个具体的滤波器, 就可以通过大量的训练样本提取到数据中的特征。实际上, 卷积神经网络的训练就是对这些滤波器的训练, 卷积神经网络的参数很大一部分就是这些滤波器的参数。虽然卷积神经网络常用于图像任务中, 但也可以用于一维信号处理任务, 并在如含噪脑电信号样本去噪这样的任务中取得了实用的成果<sup>[20]</sup>, 通过大量前期研究确定将卷积神经网络作为本文的最终去噪模型组成结构之一。

#### 2.2.2 长短期记忆网络

长短期记忆网络简称 LSTM(Long Short Term Memory Network), 是一种循环神经网络 RNN(Recurrent Neural Network), 相对于普通的神经网络这种网络不仅接受上一层网络的输出作为输入还接受前一个时间节点的输出作为输入, 这使得网络可以按照时序方向对其前向传播<sup>[21]</sup>, 这种特点非常适合时序数据的处理任务。虽然简单 RNN 网络在处理长时间的时序数据时会存在梯度消失的问题, 但 LSTM 网络在简单 RNN 的基础上通过增加长程依赖, 设置更新门, 遗忘门,

输出门的方法可以有效解决这个问题。普通 LSTM 网络在时序传播上是单向的,无法有效利用时序数据后段信息,通过增加传播方式为按时序的逆向传播的第二层网络,组成双相 LSTM 网络 (Bi-LSTM) 可以使网络同时利用时序数据前后信息。同 CNN 网络一样 LSTM 网络也已经广泛用于一维时序数据任务中,如对信号噪声干扰多的肌肉疲劳信号识别等问题<sup>[22]</sup>。基于以上调查与原因确定将长短期记忆网络作为本文的最终去噪模型组成结构之一。

## 2.3 TensorFlow 框架

TensorFlow 是一个机器学习和深度学习架构,可以在大规模和异构环境中运行<sup>[23]</sup>。TensorFlow 框架已经实现了常见的深度学习网络结构如卷积神经网络<sup>[24]</sup>,长短期记忆网络,同时 TensorFlow2 框架无需定义计算图,可以像 Python 程序一样写函数,写完直接调用,搭建和训练模型过程简便,利用 keras 接口又可以高度自定义网络结构和损失函数。打包好的模型可以在大多数平台调用,对各种硬件都有很好的支持。基于此选择此框架对本系统涉及的深度学习算法进行开发、迭代和部署等工作。

## 2.4 PyQt 框架

PyQt 是 Python 针对 Qt 提供的一种功能性扩展,Qt 是一个基于 C++ 语言的图形用户界面应用程序开发框架,PyQt 使用 Python 语言实现了 Qt 库的大部分功能,它有基本的 Qt 库函数,和超过 300 个类、将近 6000 个函数和方法<sup>[25]</sup>。利用 PyQt 开发的程序可以在所有的主流操作系统上运行,故选用此框架作为本系统客户端开发工具。

## 2.5 Django 框架

Django 是一个开放源代码的 Web 应用框架,由 Python 编写,具有完整建站能力。使用 Django,只要很少的代码,Python 的程序开发人员就可以轻松地完成一个正式网站所需要的大部分内容,并进一步开发出全功能的 Web 服务。同时 Django 本身基于 MVC 模型,即 Model (模型)+View (视图)+Controller (控制器)设计模式,因此天然具有 MVC 的出色基因:开发快捷、部署方便、可重用性高、维护成本低等。Python 加 Django 是快速开发、设计、部署网站的最佳组合。<sup>[26]</sup>

## 2.6 MySQL 数据库介绍

MySQL 是一个关系型数据库管理系统,它是当今最流行的开源 SQL 数据库管理系统。该数据库系统使用结构化查询语言 (SQL) 支持多线程、多用户和 SQL 数据库管理系统等功能。拥有快速、可靠和易于使用的特点<sup>[27]</sup>。同时 MySQL 对各种操作系统都有很好的支持并多种编程语言提供了 API,可以很方便的通过 Python 语言进行调用。

## 第3章 需求分析

### 3.1 概述

基于深度学习的超导瞬变电磁去噪系统需满足矿产勘探技术人员对于实际测量数据的输入与去噪需求。在这个过程中软件要解决的问题主要分为以下几点：

#### (1) 预处理

实测数据会因为复杂的野外情况产生如坏点，断层之类的损坏。这样的数据无法直接去噪，需要通过一定的数据预处理方法进行修复。如果修复失败需要标记数据无效并反馈给用户。

#### (2) 去噪方案选择

野外测取数据时情况多变，根据不同的测取环境和不同参数需要选择不同的去噪方案，这个阶段用户可以选择使用上传自己的去噪模型、选择软件内置的模型或让软件根据输入的参数自动决定去噪模型这几种方法来决定去噪方案。

#### (3) 软件系统性能

由于在野外实际测得数据难度大，特定数据只能在特定地点测得，离开相应地点再返回重测的代价较大。要求软件要实时给出去噪结果，用户依据去噪结果实时决定是否要重测数据，这对软件的性能提出了较高的要求。

#### (4) 数据显示

在使用系统时用户需要迅速获得去噪的结果与评估，需要一个可以清晰显示去噪后数据的分布情况，允许用户更改参数，查看不同方法的去噪效果对比的数据显示功能。此功能应当简单便捷，交互性强。

#### (5) 多模型训练

为了提高去噪效果，最大程度利用环境参数，系统需要根据不同的应用场景，训练不同的模型。要求系统提供多模型训练的功能，可以快捷高效的完成不同参数模型的训练。

### 3.2 功能需求

基于深度学习的超导瞬变电磁去噪系统包括了两种用户角色——普通用户和系统管理员，不同的角色对于系统有不同的需求。

超导瞬变电磁去噪系统的主要服务对象是用户，本系统的设计与实现目的就是帮助用户通过深度学习算法实现超导瞬变电磁数据去噪和分析工作。用户可将数据上传到软件平台，输入超导瞬变电磁测量系统的系统参数，系统会自动

对数据进行预处理，并根据参数自动选择已有的训练模型进行去噪，同时用户也可以自己上传或指定去噪模型。最后去噪的结果与评价会通过图像可视化出来，用户可以查看不同去噪模型的去噪效果对比并保存去噪后的数据和图像结果。对于不适用于已有训练模型的测量系统和系统参数，用户可以提交测量系统参数到后台反馈。此外，用户还可以根据实际探矿结果，提交实测的电磁响应数据和对应的地下电阻率数据计算出的理论电磁响应数据进行反馈。系统管理员可登录后台管理系统，审核用户提交的数据或参数，针对相应的场景训练不同模型，增加数据集数据或更改现有模型参数，提高系统的训练效果和通用性。

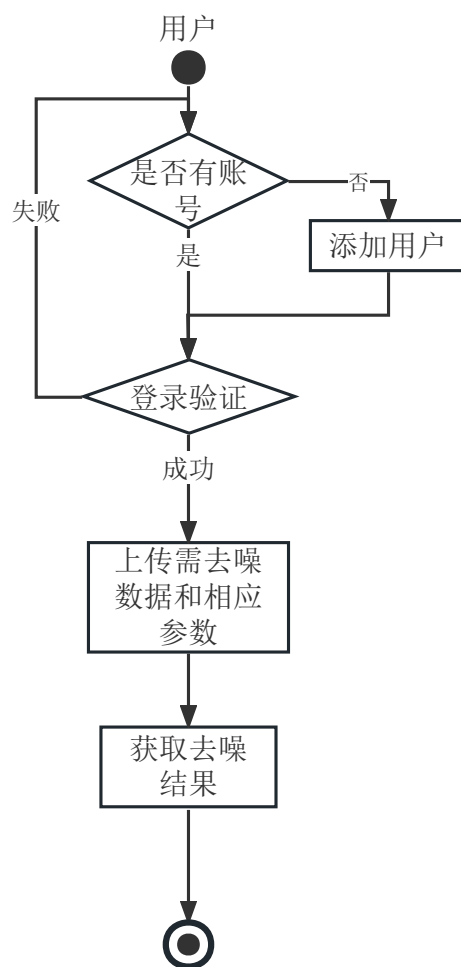


图 3.1 在线去噪任务活动图

根据以上需求绘制了系统用例图，用例图如图 3.2 所示

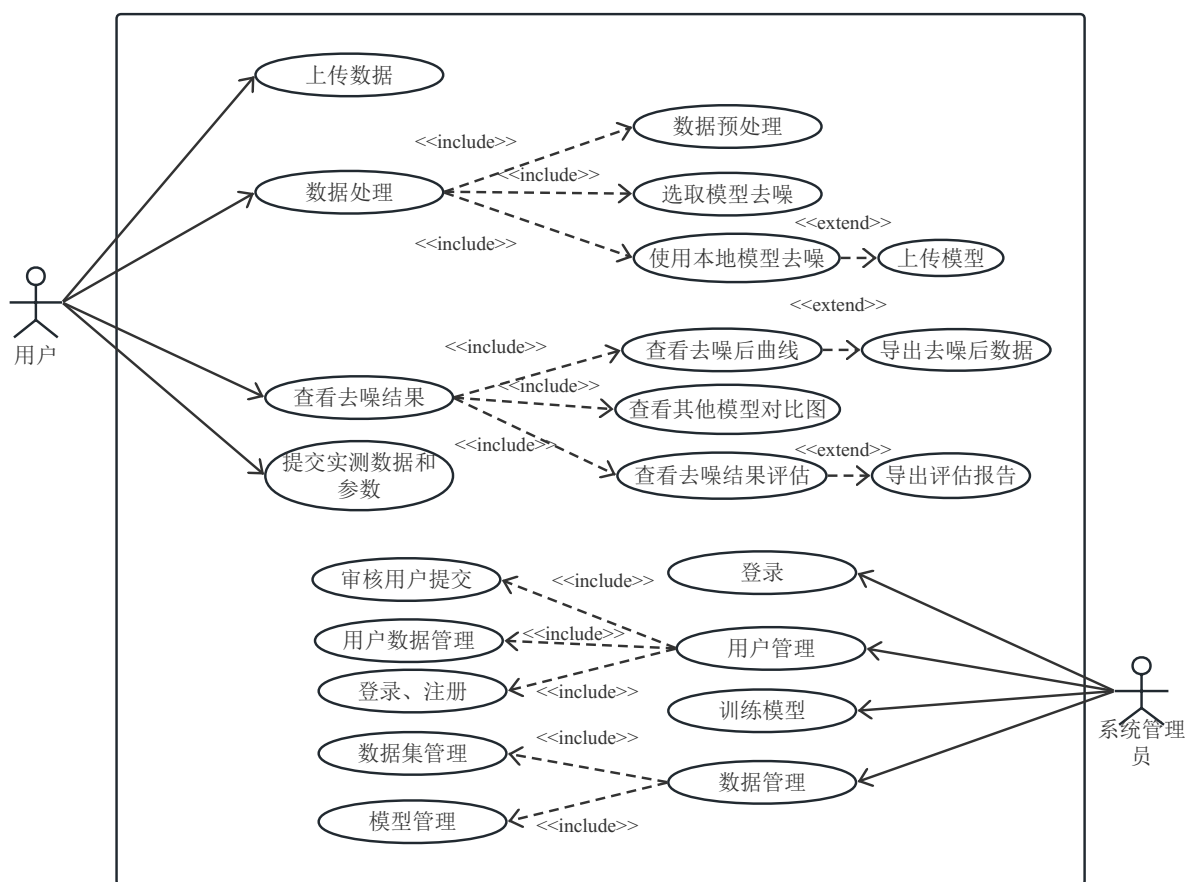


图 3.2 系统用例图

图中描述了超导瞬变电磁系统的主要功能和用户操作。该系统的用户包括普通用户和管理员两种角色。

从普通用户的角度来看，超导瞬变电磁去噪系统需要提供以下功能：

#### (1) 账户信息管理

用户首次使用系统时，需要填写账户密码等个人信息，方便之后使用在线功能。

#### (2) 数据上传与预处理

用户将指定格式的实测数据上传到系统，测量系统参数可通过读取本地文件方式上传到系统，或者可以手动选择测量系统参数。根据上传的测量参数系统完成对数据的预处理操作，返回数据预处理的结果。

#### (3) 选取模型去噪

用户提交去噪请求，系统根据用户上传数据和测量系统参数选择合适的已有模型进行去噪。或者，考虑到实际测量环境，用户可以使用本地去噪模型或指定已有模型(包括传统去噪算法)进行去噪，整个过程实时完成并返回去噪结果。



#### (4) 查看去噪结果

去噪结果包含电磁信号曲线去噪图、其他模型去噪结果对比图、去噪压制比。其中，电磁信号去噪图包含有单个曲线或多个曲线去噪图，去噪结果可通过抽道算法转化为剖面图片，当存在多条测线数据时可以通过多个一维数据生成区域二维图像，生成的图像可以调整比例。其他模型去噪对比图用于参考对比，去噪压制比图包含两类数据的分析一种是真实数据 (未知理论曲线) 一种是模拟数据 (已知理论曲线)，用于评估去噪效果。

#### (5) 提交勘探结果数据和参数

本系统应用于矿产资源勘探场景，用户可提交一些实际勘探结果数据，数据包括使用超导瞬变电磁装置采集到的电磁响应数据和实际挖矿过程中得到地下电阻率数据计算出的理论电磁响应数据，提交给后台进行反馈。对于系统中已有模型覆盖不到的测量参数，可提交系统参数给后台进行反馈。

从系统管理员的角度，系统需要具有以下功能：

##### (1) 审核用户提交的数据

根据上文所述的用户反馈的数据和参数，系统管理员可登录系统，在审核模块审核数据是否存在异常，如果数据审核通过，该数据添加到待训练数据集，如果测量参数审核通过，管理员根据测量参数创建仿真数据集。

##### (2) 训练模型

系统管理员可根据不同的使用场景训练多个不同的模型，训练好的模型以 h5 文件保存在服务器中方便调用，同时管理员也可以根据用户反馈的数据和参数训练新的模型或调整已经训练好的模型。

##### (3) 数据管理

数据管理分为数据集管理和模型管理。数据集管理可对单个数据集进行增删改查，也可根据用户提交的实测数据制作新增数据集。模型管理可以增加和删除单个模型。

##### (4) 用户注册

本系统根据实际应用情况，用户需要注册时需要系统管理员将账户和密码添加到系统中。

### 3.3 性能需求

#### (1) 响应时间

系统的主要响应时间是系统获取电磁响应数据后给出去噪结果这一时长，系统响应用户从 GUI 界面输入的去噪指令，时间不能多于 1 秒，对于系统运行负荷较重时，该响应时间也不能超过 3 秒。

(2) 准确率

本系统作为电磁数据去噪系统，必须保证在实际使用场景中，数据的去噪结果有比较高的准确率。作为去噪结果评估标准，去噪后噪声压制超过 30 倍。

(3) 稳定性

本系统要求能提供稳定的应用服务，系统能够正常的运行，不会在使用过程中出现严重的故障。当运行环境发生变化后，本系统仍然能保持较好的运行效果。

(4) 鲁棒性

对于常用的测量系统参数，本系统都能提供算法支持。并且，本系统的算法应当能适应数据中存在的异常，对带有异常的数据也能保持计算结果的准确性。

### 3.4 其他需求

(1) 界面需求

界面需求要求系统给用户提供简洁易懂的操作服务。界面设计应当交互性强，操作方便，使用逻辑符合行为逻辑。界面美观清晰，为用户提供良好的体验。对于管理员用户，界面应当清晰明了，操作简单。

(2) 数据存储

系统需提供数据存储功能，包括操作日志、系统异常状况记录等。系统需对所有的数据进行安全的存储，保证数据的安全性和准确性。

(3) 接口需求

本系统需在公共互联网上通信，实现数据的传送和接收。用户平台与系统服务器之间应提供通信接口。系统管理员平台与系统服务器之间也需提供通信接口。

### 3.5 本章小结

本章首先确定了本系统要解决的问题，从用户和系统管理员两个视角明确了用户对本系统的功能需求。根据这些需求进一步的划分了系统需要实现的功能，同时通过对系统使用场景的分析，确定了系统的性能需求和其他非功能需求。最后针对这些需求提出对应的指标，为接下来系统的具体设计建立了基础。

## 第4章 系统设计

### 4.1 系统概述

本系统基于深度学习算法，为用户提供实际超导瞬变电磁测量数据的去噪与评估功能。为管理员提供数据管理和模型训练功能。系统整体分为用户子系统和管理员子系统，分别为用户和管理员提供不同的功能。

### 4.2 系统结构设计

#### 4.2.1 系统架构与工作逻辑

本系统采用了 C/S 架构。系统包括客户端和服务端两个主要部分。客户端包括数据录入、数据预处理、数据展示等功能，而服务端主要用于数据处理和模型训练。客户端和服务端之间通过网络进行通信，可以进行数据上传和下载，在线预测等操作。

系统的架构图如图4.1所示

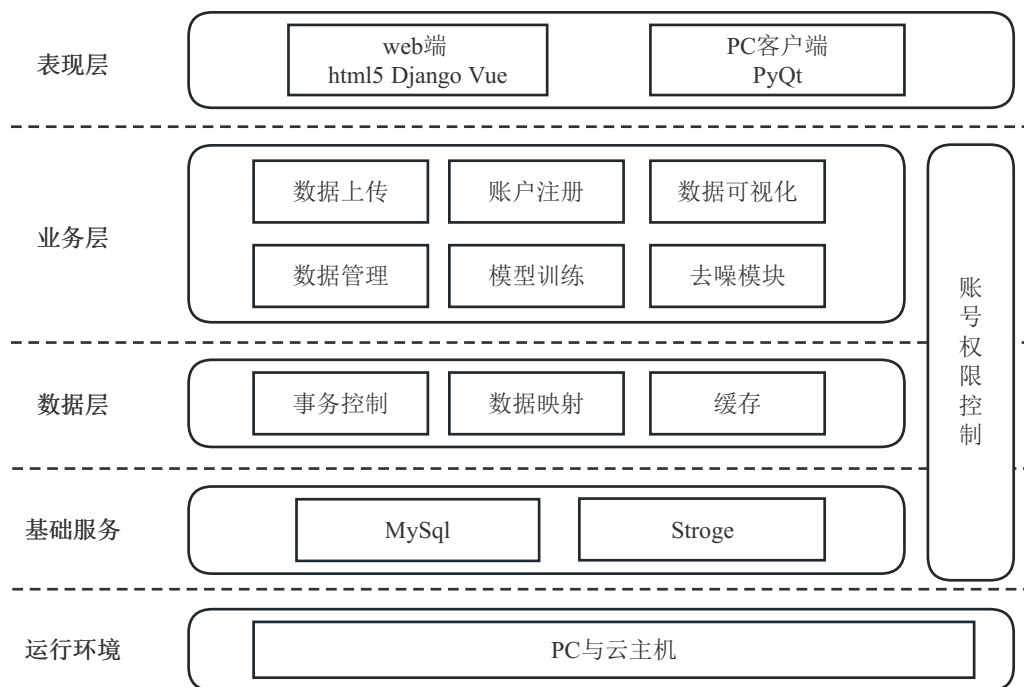


图 4.1 系统架构图

其中客户端包括用户界面、本地数据库、数据录入模块、数据预处理模块、数据可视化模块、数据上传模块等组件。用户界面为用户提供了友好的交互界面，让用户可以方便地进行数据录入、数据预处理和数据展示等操作。本地数据

库用于存储本地数据，可以对录入的数据进行存储和查询。数据录入模块可以通过手动输入和读取本地文件的方式将测量参数和原始数据录入到本地数据库中。数据预处理模块可以对录入的数据进行预处理和去噪，提高数据的质量。数据可视化模块可以将处理后的数据进行可视化展示，方便用户查看。

服务端包括计算机集群、服务器数据库、数据处理模块、模型训练模块等组件。计算机集群用于进行数据处理和模型训练等任务，可以提高数据处理和模型训练的效率。服务器数据库用于存储服务器端数据，包括用户提交的数据、上传的数据集等。数据处理模块可以对用户提交的数据进行处理和去噪，提高数据的质量。模型训练模块可以根据用户提交的数据集进行模型训练，并保存训练好的模型。

本系统客户端的工作逻辑如图4.2所示。

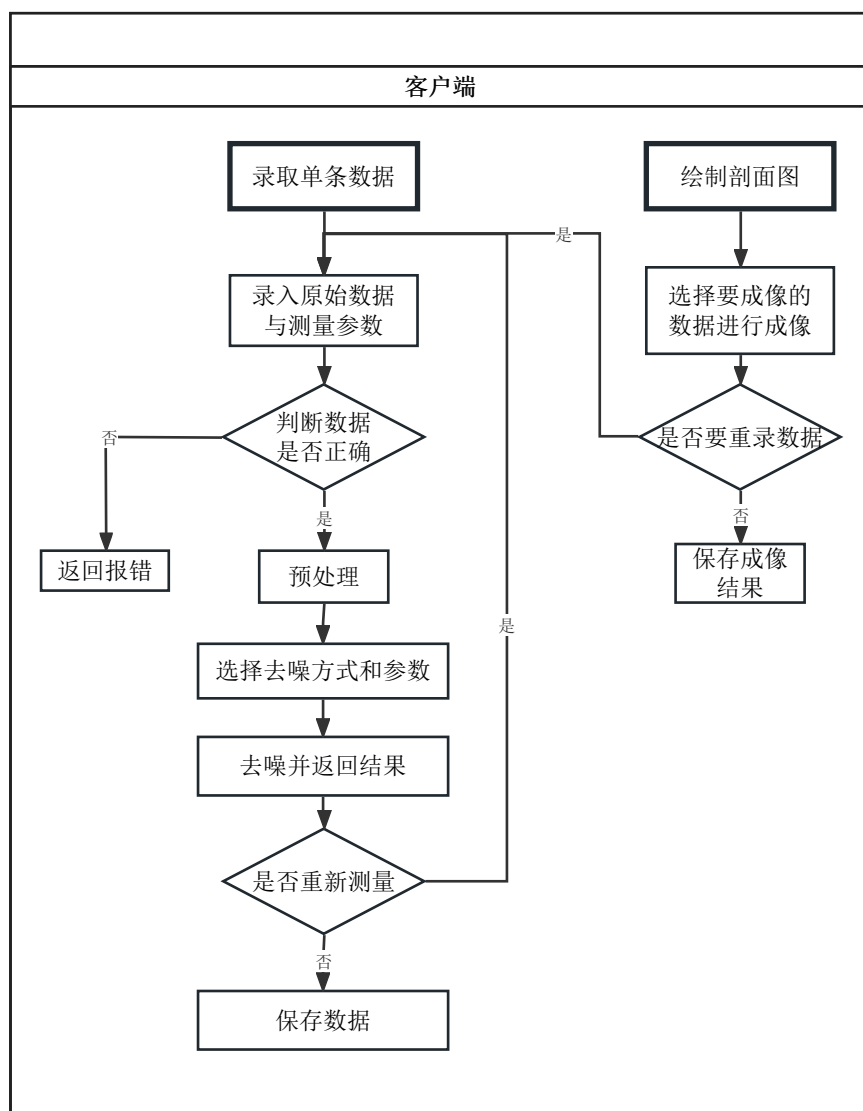


图 4.2 客户端工作逻辑

图中描绘了客户端独立完成任务，主要包括对单条数据的各种处理和联合多条数据进行剖面图绘制。

具体的工作流程为：用户在客户端选择要录入的原始数据并输入相关测量参数，输入完成后进行预处理步骤。在预处理阶段如果发现数据输入有误或参数错误则反馈报错信息。预处理完成后用户可以设定去噪参数，主要选项包括本地去噪或在线去噪以及手动选择去噪模型或自动选择。数据去噪完成后可以查看本条数据的去噪效果以及一些去噪前后数据的噪声水平对比等参数并自动在本地进行保存。当一条测线的所有数据点测量并去噪完毕后可以根据这些数据自动绘制剖面图，此图可以更全面的反应数据的质量和异常情况方便野外测绘人员及时进行调整或者重测工作。

本系统服务器端的工作逻辑如图4.3所示。

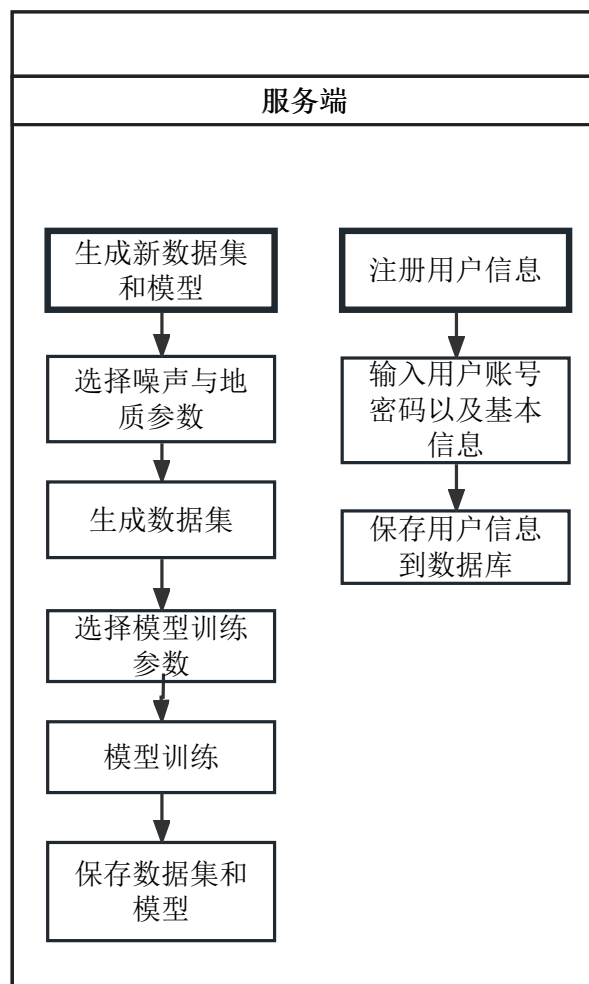


图 4.3 服务器端工作逻辑

图中描绘了服务器端独立完成任务，主要包括对数据集和模型的各种操作和对用户信息的管理。

具体的工作流程为：系统管理员可以在服务器后台界面管理已有的数据集

和模型，根据需要设置新的噪声水平和地质类型等参数并生成相应数据集进行模型训练。同时系统管理员可以在服务器后台管理普通用户账户进行注册管理等操作。

本系统客户端和服务端端的交互工作逻辑如图4.4所示。

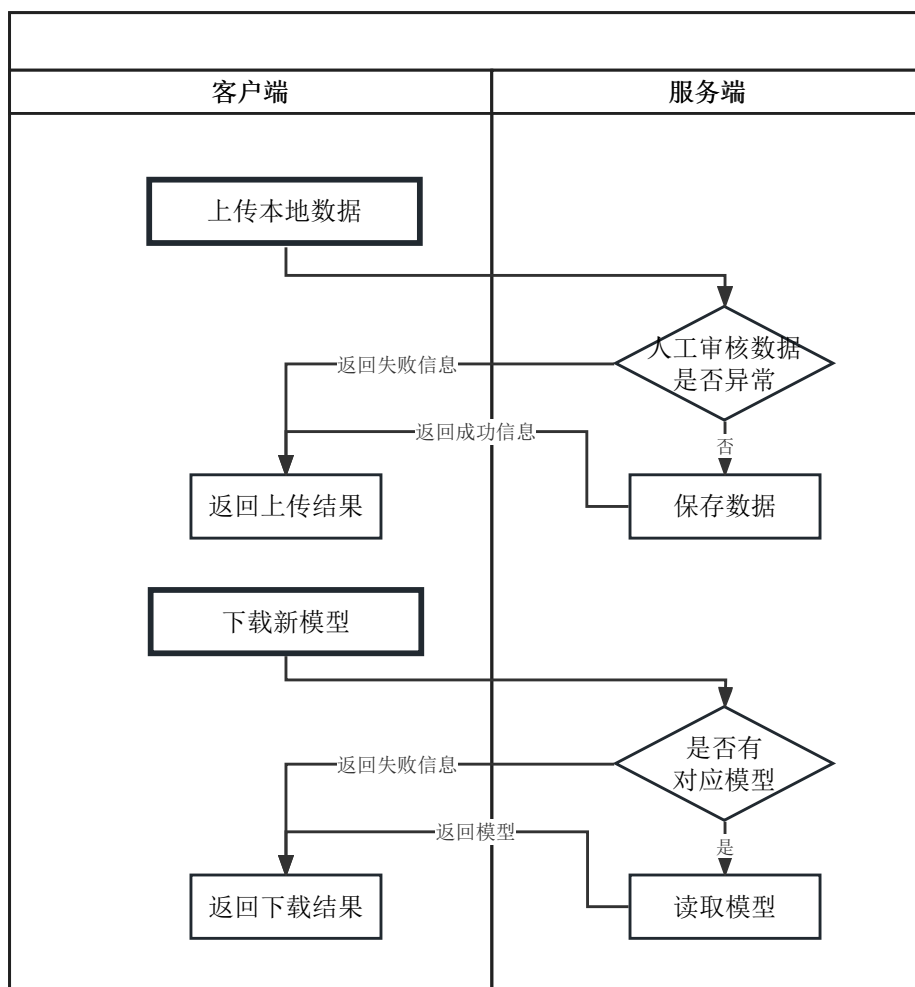


图 4.4 客户端和服务端端的交互工作逻辑

需要交互工作包括，用户在客户端选择测得的数据以及测量参数提交到服务器端帮助之后的模型更新工作，用户通过服务器端下载更新的模型。系统管理员对普通用户上传的数据进行审核并在审核通过后将数据加入到现有数据集中。

#### 4.2.2 系统功能模块

在分析完本系统的处理逻辑之后，接下来对系统的功能模块进行详细划分，根据本系统的特点将功能模块分为用户子系统模块和管理员子系统模块两部分，两部分通过相应的通信模块进行交互。

本系统的功能模块如图4.5所示，其中用户子系统分为数据录入、预处理、去噪、数据可视化、数据提交与模型下载，管理员子系统分为模型训练、在线预测、用户管理、数据管理一共九个功能模块。

同时在表 4.1 对系统涉及到的功能模块进行进一步说明。

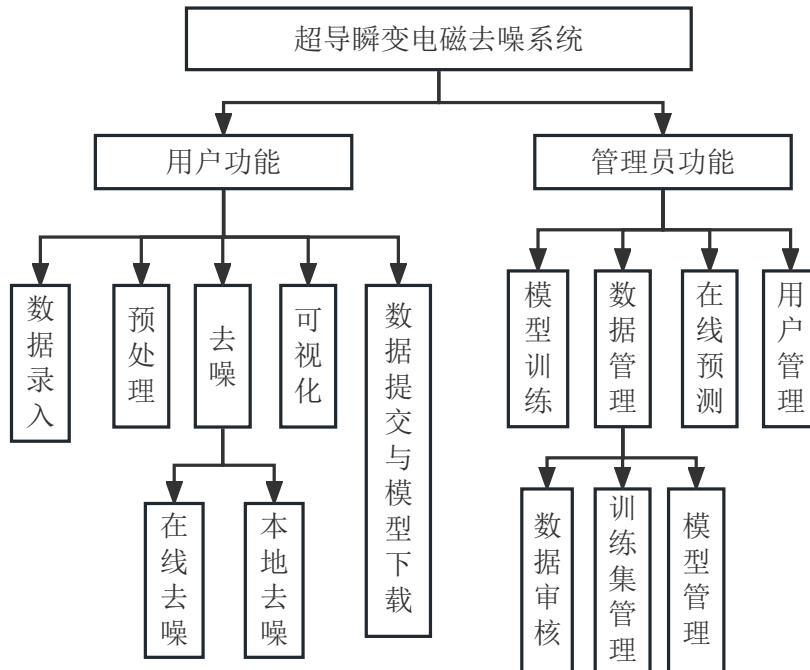


图 4.5 系统模块图

表 4.1 模块功能描述表

模块名称	功能
数据录入模块	录入需要处理的原始数据
预处理模块	将原始数据进行一定处理得到去噪前的数据
去噪模块	对经过预处理后的数据进行去噪操作
数据可视化模块	将本系统处理数据时的结果可视化呈现给用户
数据提交与模型下载模块	向服务器端提供测量的数据样本，从服务器端下载最新模型
模型训练模块	为系统管理员提供训练新模型的功能
在线预测模块	此模块为系统管理员和普通用户提供在线预测的功能
用户管理模块	为系统管理员提供注册管理用户账户等功能
数据管理模块	数据管理模块提供数据集、模型的管理以及数据集生成等功能

### 4.3 功能模块详细设计

以上介绍了本系统的运行逻辑和总体架构设计，本小节对系统涉及的各个功能模块进行更详细的设计。

#### 4.3.1 数据录入模块设计

数据采集用户根据测绘计划需要，向系统中加入新的超导瞬变信号原始数据，加入数据的方式需要适合实际场景且简单直观。

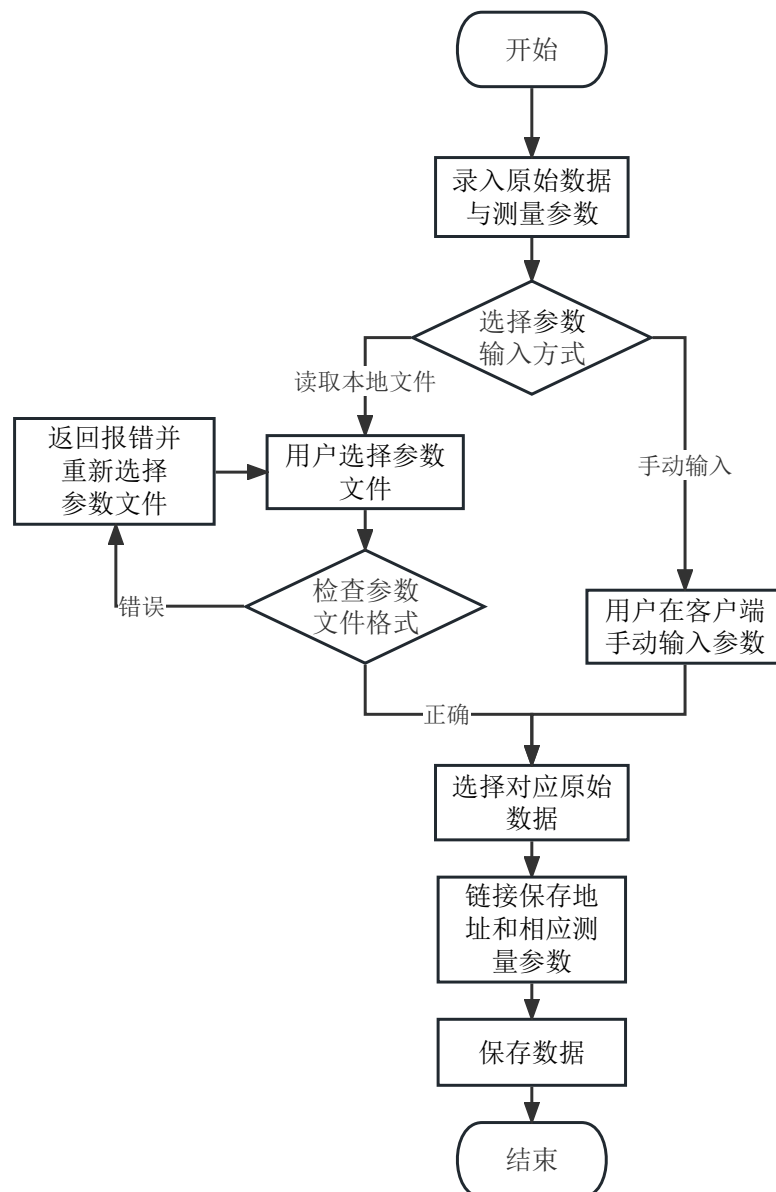


图 4.6 数据录入流程



如图4.6所示,用户可通过手动输入和读取本地文件两种方式将测量参数输入系统中,两种方式对应了两种不同的用户使用场景,手动输入对应用户在野外采集数据时需要实时输入当时的测量参数。读取本地文件对应用户在计划野外测量方案时将测量参数预设好,或在测量结束后进行测量参数的补录,读取本地文件获取参数要求文件格式需要符合系统要求,不符合则会提示错误。通过本地文件录入的测量参数可通过手动输入进行覆盖。

在测量参数输入完成后可以进行原始数据的录入,原始数据的录入过程在本系统体现为将原始数据在本地的保存地址和相应的测量参数进行连接并存储进数据库的过程。本系统不提供原始数据的实际测量功能,超导瞬变电磁信号的原始数据是一种后缀为 `tdms` 的二进制文件<sup>[28]</sup>,超导瞬变电磁信号的测量需要专门的硬软件完成,python 中有名为 `nptdms` 的第三方库可以对此文件进行读取,并转化为 `pandas DataFrame` 类型的文件<sup>[29]</sup>。

数据录入完成后,可通过可交互图片的方式显示出来,用户可以查看原始数据细节。

### 4.3.2 预处理模块设计

当原始数据录入完成还需一些预处理步骤才能得到需要去噪的数据。用户可通过预处理模块对原始数据进行预处理操作,预处理操作具体包含断层补偿,跳点检测修复,同步分割三个步骤。

#### (1) 断层补偿

由于超导瞬变电磁的探测仪器是一个非线性元件,即数据的输出和环境磁场变化的输入是非线性的关系。为了将仪器的输入和输出转换成线性关系,同时提高灵敏度,探测器必须工作在负反馈调节电路中。然而当环境磁场的输入大于负反馈调节电路的调节水平时会造成“失锁”现象,具体表现在测得的信号中会发现信号出现了阶跃和断层,这样的数据是完全失真的,无法进一步使用,因此阶跃点去除成为瞬变电磁信号处理的重要步骤<sup>[30]</sup>。在本系统中将其做为预处理的第一个步骤也是最核心的步骤。

如图4.7所示,断层补偿的实际步骤主要包括,首先基于超导瞬变电磁获取信号 `B`,对 `B` 选定一个周期个数 `N` 来作为计算相似度的周期数,将这 `N` 个周期组成的区间称为 `Q`,因为原始数据每个周期长度是有细微变化的,根据起始点的不同 `Q` 的长度也不同。从信号的第一个点开始选取第一个区间 `Q`,计算其长度,之后计算区间 `Q` 和 `Q` 之后同样长度的区间 `Q'` 的均方根。当均方根大于某一阈值时代表下一个区间出现了断层或阶跃点,转入阶跃点的步骤,如果均方根小于阈值说明下一个区间数据正常。此时从区间 `Q` 后选取 `N` 个周期组成新的区间 `Q` 重复步骤 S1 直到数据结束。

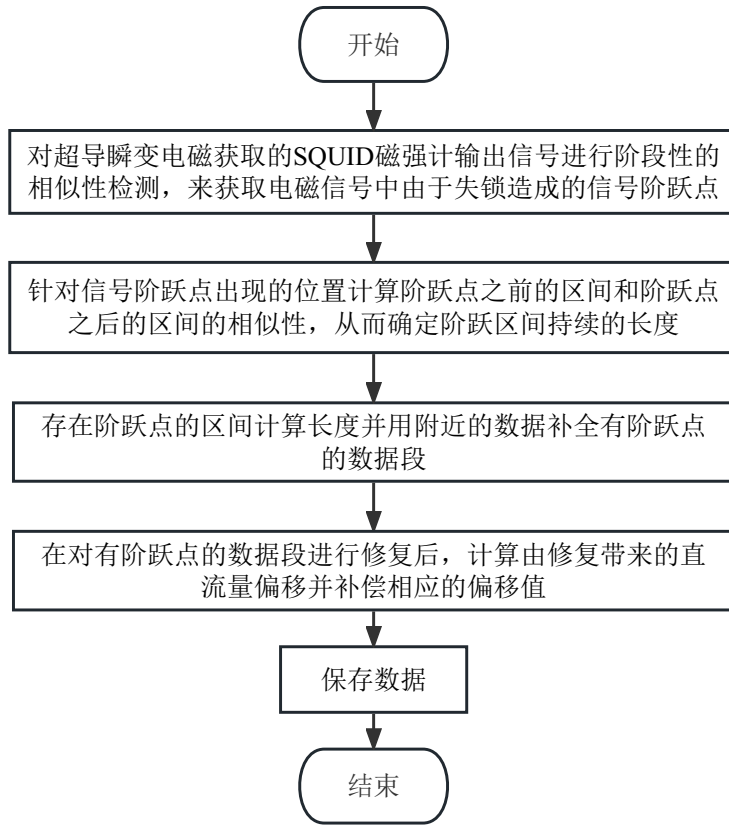


图 4.7 断层补偿流程

具体补偿方法的流程如图4.8所示，其中具体的计算区间相似度方法为：从第一个数据点开始，向后数  $N$  个周期组成区间  $Q$  并把  $Q$  的长度计为  $m$ ，从  $Q$  的结尾向后数  $m$  个数据点，将这段数据  $Q'$  与  $Q$  计算均方根，均方根越大代表两段数据越不相似，如果没有断层出现两段数据的相似度会很高，其均方根会很小。当计算的均方根大于一个阈值时，转入处理阶跃点的步骤。如果均方根小于阈值，那就说明  $Q$  的结尾向后数  $m$  个数据点是正常的数据。此时从  $Q$  的结尾向后数  $N$  个周期组成新的区间  $Q$  并更新相应的长度  $m$ ，重复以上的过程直到数据结束。

相似度的计算公式为均方差，具体公式如下：

$$J = \frac{1}{2m} \sum_{i=1}^m (Q_i - Q'_i)^2 \quad (4.1)$$

在转入阶跃点处理流程后以区间  $Q$  为标准计算区间  $Q$  之后相同长度数据  $Q'$  和区间  $Q$  的相似度，如果相似度未到达阈值，则将  $Q'$  的起始点后移一个(长度不变)再次计算相似度，直到达标为止。当  $Q'$  相似度与  $Q$  足够高时可以认为阶跃区间在  $Q$  的最后一个点和  $Q'$  的第一个点之间。至此阶跃区间持续的长度计算完毕。其中，在阶跃点处理流程中计算相似度时公式与上文的公式一致。不同的是，由于阶跃点会带来直流量的偏移，所以在计算相似度之前需要分别计算  $Q$

和  $Q'$  的平均值，并将  $Q'$  中所有数据点的值减去  $Q$  和  $Q'$  的平均值的差值，再计算相似度，以消除直流偏移带来的影响。具体的平均值差值计算方法如下：

$$D = \frac{1}{m} \sum_{i=0}^m Q_i - \frac{1}{m} \sum_{i=0}^m Q'_i \quad (4.2)$$

在上一个步骤计算出阶跃区间的长度和位置后，将离阶跃区间最近的一个完整周期即  $Q$  区间的最后一个周期复制若干遍，用此段复制的数据来替换掉阶跃区间的数据。周期复制的次数确定方法为：设阶跃区间长度为  $L$ ，复制的单个周期长度为  $n$ ，复制次数为  $t = \lfloor L/n \rfloor + 1$  复制的数据会略长于阶跃区间长度，在替换时截取复制数据的前  $L$  个点，使之和阶跃区间长度相等。在阶跃区间替换修复完成后，将修复完成的阶跃区间之后的所有数据都减去之前计算出的  $Q$  和  $Q'$  的平均值的差值，以消除直流偏移，从修复完成的阶跃区间的最后一个点开始重复查找阶跃点的步骤，直到数据结束。

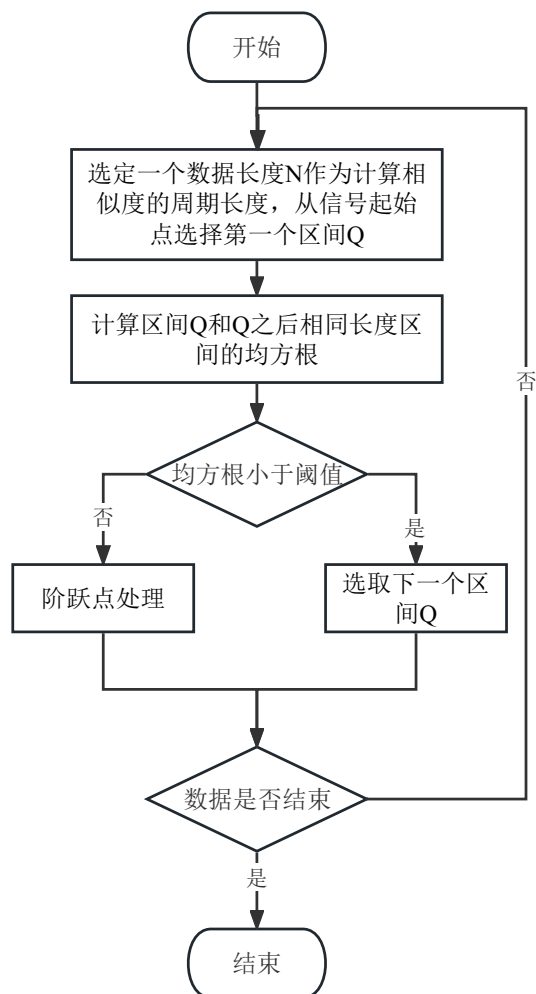


图 4.8 补偿具体方法

### （2）跳点修复

跳点是原始数据上出现的离群点，出现的原因多样包括随机噪声或仪器的不稳定等，这种噪声无法被避免，在预处理阶段通过一些常用的对离群点进行检测和修复的算法处理原始数据可以增加数据的精度，减少之后去噪的难度。

### （3）同步分割

瞬变电磁信号是一个指数衰减的信号，但由于信号衰减很快所以在实际测量时硬件会以一个固定频率发出脉冲信号并记录反射的电磁信号，反射的电磁信号频率和发射频率保持一致。而系统输入的原始信号就是这样的信号，是一个周期性的电磁数据，将其以频率的倒数作为周期进行分割并叠加后就得到了一个指数衰减的瞬变电磁信号，这样的叠加减少的数据在测量过程中产生的随机误差，具体的分割频率等参数在之前的输入测量参数时已经记录到系统当中，整个分割并叠加的过程就是同步分割，这个步骤后得到的数据就是接下来需要去噪的数据。

## 4.3.3 去噪模块设计

去噪模块对之前经过预处理的瞬变电磁信号数据进行去噪操作，此模块是本系统的核心模块，用户在通过使用本模块时可以选择不同的去噪方式和去噪参数进行去噪操作。图4.9为去噪总流程，本模块提供本地去噪和在线去噪两种工作模式。

### （1）本地去噪

本地去噪对应用户在野外录取数据时需要进行实时去噪工作，如图4.10所示这种模式用户可以选择已在本地的去噪模型。本系统封装了小波去噪和经验模态分解 (EMD) 两种常用传统的去噪算法，使用这两种算法需要预设一些去噪参数，若用户选择不输入去噪参数系统会使用默认参数进行去噪。用户也可以选择深度学习去噪算法进行去噪，本系统的深度学习去噪模型都由 TensorFlow 框架搭建和训练，并保存为 h5 模型文件格式。在选择模型时会将一个标准数据输入模型来检测模型是否符合系统要求，若不符合会返回报错信息。在使用深度学习算法去噪时系统会通过数据的测量参数自动选择去噪模型，用户也可以指定保存在本地的模型进行去噪。

### （2）在线去噪

由于使用的深度学习去噪模型网络结构较为复杂，在性能较低的硬件上运行可能会有较长的等待时间，所以当用户在良好的网络使用本系统时可以选择在线去噪模式，此模式下之可以选择储存在服务端去噪模型，服务端只提供深度学习去噪模型，去噪模型会由系统管理员进行定期跟新。用户若是想使用在线去噪功能需要登陆用户账号，考虑到实际使用情况本系统暂时不支持用户自行

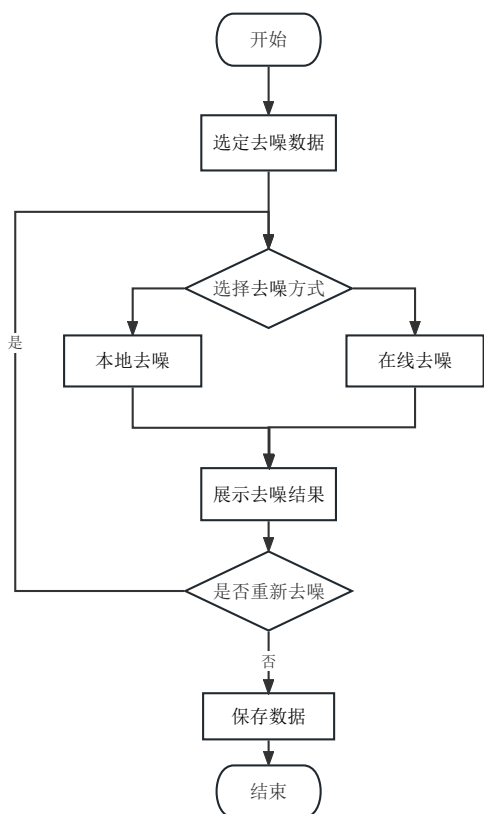


图 4.9 总体去噪流程

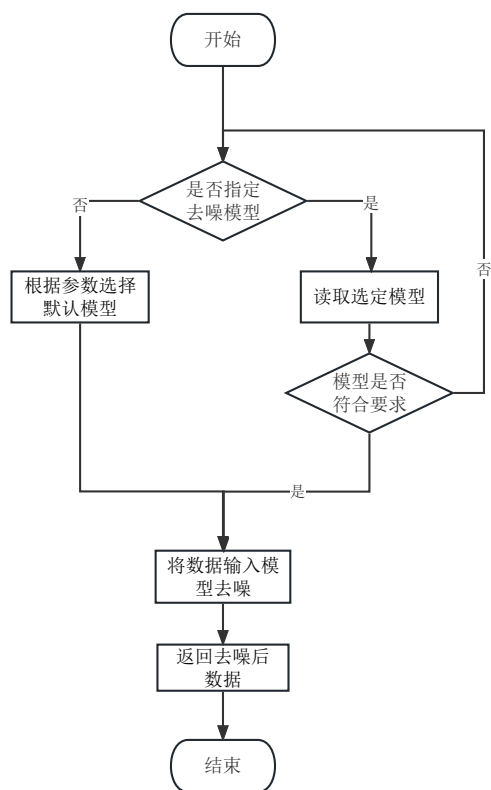


图 4.10 本地去噪流程

注册账户的功能，需要系统管理员手动将用户信息加入到后台数据库中。使用在线去噪时的操作与使用本地模型去噪时一致，用户可以指定去噪模型或由系统自动决定，确定模型后，系统会将待去噪数据和对应的测量参数上传到服务器后台，在服务器计算完成后返回去噪后数据到本地系统。

#### 4.3.4 数据可视化模块设计

数据可视化模块是为了方便用户对数据有更直观的理解而设立的模块，虽然本系统在数据处理的各个阶段都支持数据的导出，但考虑到本系统的主要使用人群，有一个直观的数据可视化模块会大大提升用户的使用体验，并提升工作效率。

本系统涉及到的数据种类多样，不同的数据有不同的展示方法，数据可视化模块将这些数据展示方法进行整合和封装，方便系统在需要时进行调用。此模块独立于去噪流程，在需要成图时调用相应接口即可，模块的主要功能包括将一维的电磁信号数据转化为可交互图片，可以进行缩放、拖动、保存和读取光标所在位置数据等操作，并且支持多个同类型的一位电磁信号时显示在同一张图中。同时对于一组瞬变超导电磁数据可以通过特定算法进行二维成像，此图也是用

户判断数据质量和地下空间分布的重要依据。

图4.11为数据可视化模块的工作流程

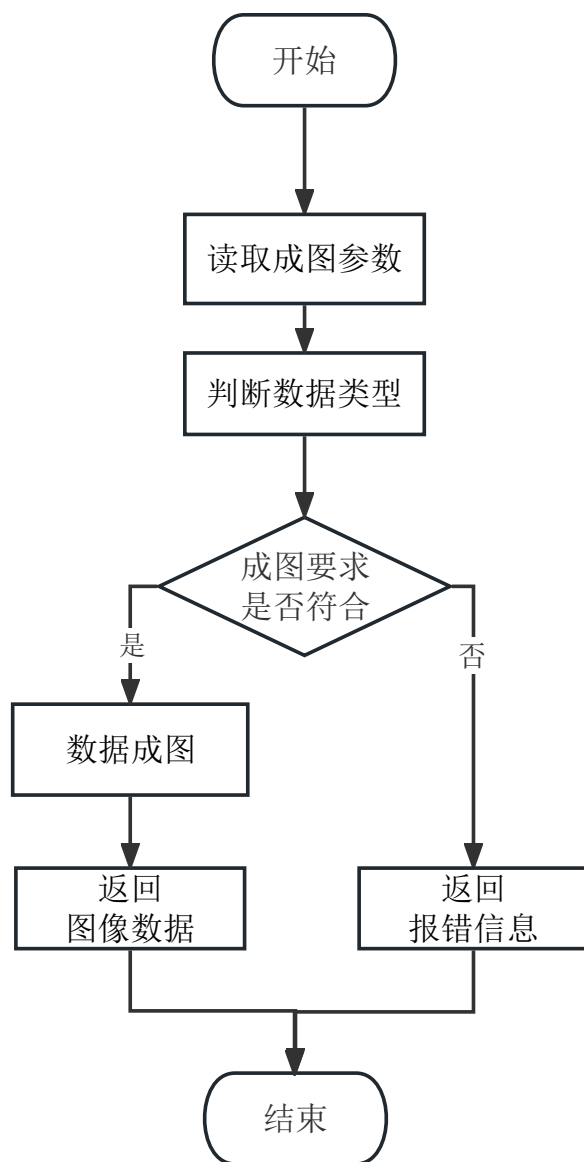


图 4.11 数据可视化流程

#### 4.3.5 数据提交与模型下载模块设计

数据提交与模型下载模块主要包括提交和下载两个功能，数据提交允许用户向服务器后台上传最新的野外实地测量数据，方便系统管理员在更新数据库与模型。而模型下载允许用户从服务器下载最新或指定的模型保存在本地。与在线去噪功能类似，这两个功能都需要用户的账号已存在于服务器后台的数据库中。

### 4.3.6 模型训练模块设计

此模块为系统管理员提供训练新模型的功能，根据系统设计的需求此模块提供了一系列用于模型训练的功能和对应接口。模型训练过程可以设置训练的迭代次数，学习率等超参数，考虑到实际需求模型开发的工作在前期算法研究时已经完成，在此阶段不做调整本模块也不支持更改网络结构或自定义网络结构。模型训练模块工作流程如图4.12所示，

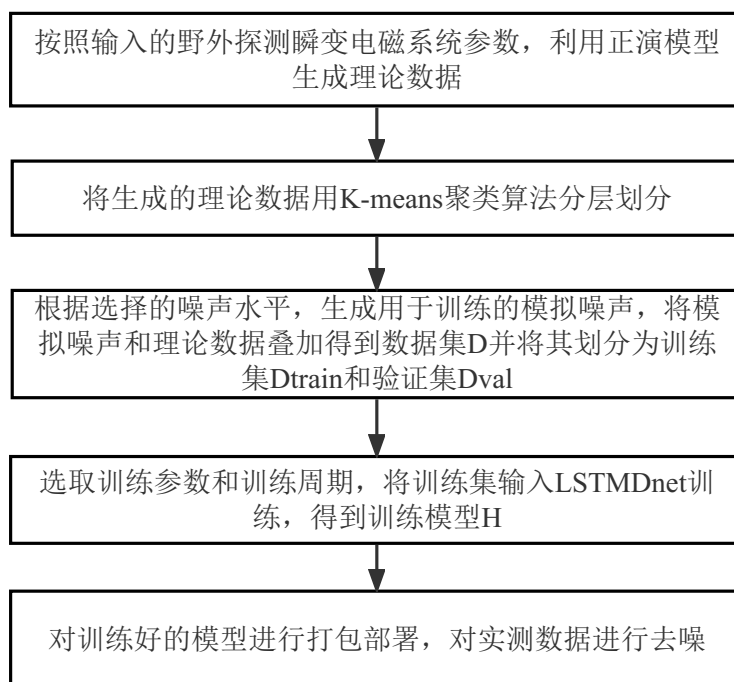


图 4.12 模型训练模块工作流程

管理员在训练新模型时需要先选择使用的数据集，新的数据集由正演算法生成，生成数据集时管理员需要输入包括数据集大小、噪声水平、电磁信号类型等超参数，之后模块会生成相应的数据集，对生成的数据集管理员还可以进行数据优化。由于瞬变电磁信号数据的理论数据通过一个正演算法生成，且此算法生成的数据分布并不均匀<sup>[31]</sup>，数据优化功能可以对生成的数据集进行聚类并根据结果对数据分层抽样，这样优化过的数据集的数据分布会更加均匀，用优化过的数据集训练的模型会有更好的表现。确定数据集后管理员需要输入训练参数，如果不输入系统会填充默认参数。在模型训练结束后可以查看模型的损失曲线，且可以将一组标准数据输入训练好的模型来检测模型的效果并以图片等方式呈现出来，管理员可以更具其表现决定是否重新训练。当模型训练结束后会自动保存到数据库中。

#### (1) 算法描述

本系统用的到的深度学习模型是一个基于全连接神经网络、双向长短期记忆神经网络和卷积神经网络架构的深度学习算法 (LSTMDnet)。它的输入是叠加后的超导瞬变电磁数据、输出是去噪之后的电磁数据。设  $s$  为超导瞬变电磁的理论信号

$$s = v(t) = \frac{C}{\tau} \sum_{k=1}^{\infty} \exp\left(-\frac{k^2 t}{\tau}\right) + B \quad (4.3)$$

其中  $C$  由地下局部导体的值、深度、真空磁导率、发射线圈半径和发射电流决定,  $C$  与导体的电阻率无关并设置为常数,  $\tau$  是导体的时间常数,  $B$  代表直流偏置。真实的真实测得的带有噪声信号可以表示为下。

$$y = s + \epsilon \quad (4.4)$$

而去噪的过程可以表示为对误差函数求解最小值

$$\hat{s} = \operatorname{argmin} \frac{1}{2} |y - s|^2 + \lambda(s) \quad (4.5)$$

如图4.13所示, 网络由输入、全连接、双向 LSTM、卷积和输出模块构成, 输入数据是归一化且添加噪声长度为  $S$  的 TEM 信号  $x$ 。归一化且添加噪声的 TEM 信号被送入输入层, 然后是四个分别具有 1000、500、250、500 个神经元的全连接层, 相邻全连接层通过节点数分别为 500、250、500、1000 的双向 LSTM 层连接。之后通过两个卷积块, 通道数扩展为 64。

## (2) 数据集制作

我们使用超导瞬变电磁的一维正演模型来构建训练集。采用等对数间隔对电阻率模型的深度方向进行剖分, 划分为  $c$  层电阻率层状模型, 总深度为  $h(m)$ , 电阻率变化范围为  $R1-R2(\text{ohm/m})$ 。使用伪随机方法生成模型的电阻率, 即下一层的电阻率与当前层相同的概率为 70%。正演模型的输出是按照等对数间隔划分时间的的电磁响应数据, 时间道  $t$  满足  $t > 2c$ 。将拟定的反演区域划分为  $M * N$  的网格, 每个网格生成  $L$  个样本, 总共生成  $M * N * L$  个样本, 将这些样本进行插值转化为实测数据的格式, 每个数据插值后的长度为  $S$ , 这些数据作为训练时的理论数据。根据对实际数据噪声的分析结果生成噪声数据, 噪声数据包含两部分, 一部分为在无激励条件下测得的环境噪声, 一部分为对噪声分析后利用相似频率的低频高频信号合成的模拟噪声。将噪声数据和理论数据叠加得到用于训练的数据集  $D$ 。

为方便模型训练, 将数据集进行归一化, 由于输入的实际数据可能有负值, 在归一化时, 选取数据的起始和结束点作为归一化的上限和下限。训练集包含两部分内容: ①带有噪声的电磁数据  $X$ ; ②电磁响应理论数据  $Y$ 。对于数据集  $D$



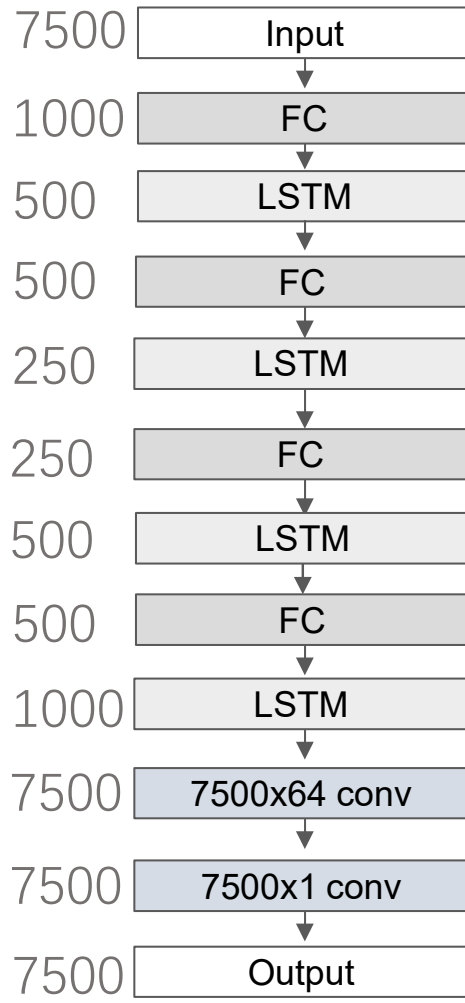


图 4.13 网络结构

中的所有  $m$  个样本,  $(X^i, Y^i), i = 1, \dots, m$ , 根据以下公式计算, 得到处理后的  $m$  个样本  $(x^i, y^i), i = 1, \dots, m$ ,

$$x^i = (X^i - X_s^i) / (X_0^i - X_s^i) \quad (4.6)$$

$$y^i = (Y^i - Y_s^i) / (Y_0^i - Y_s^i) \quad (4.7)$$

其中,  $X = X^1, \dots, X^m, Y = Y^1, \dots, Y^m$ 。理论数据  $Y$  按照带噪数据  $X$  的参数归一化。

### (3) 训练过程

采用小批量梯度下降法来更新参数, 其工作原理是: 对于一批  $n$  个样本  $(x^i, y^i), i = 1, \dots, n$ , 训练 Tem-denoise-net 来完成最小化损失函数的目的, 损失函数定义为均方差:

$$MSE(\theta) = \frac{1}{s} \cdot \frac{1}{n} \sum_{i=1}^n \sum_{k=0}^s (f_{\theta}(x_k^i) - y_k^i)^2 \quad (4.8)$$

其中  $s$  是向量  $y$  的长度。采用 adam 优化器指数衰减学习率和早期停止方法，激活函数为 SeLu 激活函数。通过每次迭代之后预测验证集，并给出训练集和验证集的损失，以此来判断模型是否收敛以及是否过拟合和欠拟合。根据损失函数迭代调整 Tem-denoise-net 中的参数  $\theta$ ，直到训练达到收敛状态，获得训练好的模型 H。

### 4.3.7 在线预测模块设计

此模块为系统管理员和普通用户提供在线预测的功能，系统管理员可以通过此模块调用系统内的模型对特定数据进行去噪，并查看去噪结果，查看方式为可交互图片与客户端保存一致。普通用户在使用在线去噪功能时也是通过调用此模块实现，具体的调用流程如图4.14所示

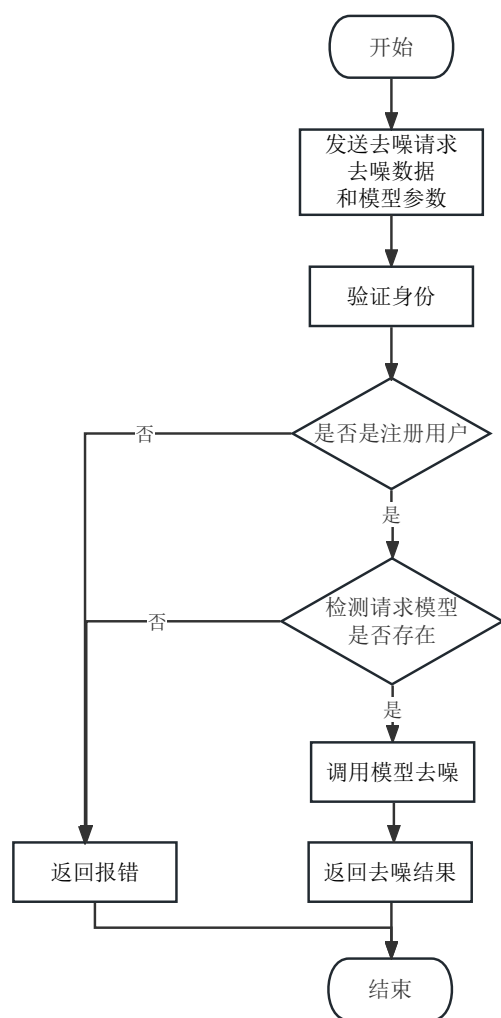


图 4.14 在线预测流程

用户在用客户端的去噪模块发出在线去噪请求并发送需去噪的数据和模型参数，在服务端的在线预测模块接受到请求后会验证请求者的身份，如果是在系

统中已经注册过的用户则进行在线去噪操作，首先模块会将请求中的模型参数进行搜索，查看模型参数是否格式正确并且检测对应的模型是否在系统数据库中。之后模块会读取对应模型将数据进行去噪并返回去噪后的结果，根据实际的需求返回结果只包含去噪后的结果或报错信息，具体的去噪后数据分析过程由客户端完成。

#### 4.3.8 用户管理模块设计

用户管理模块为系统管理员提供注册管理用户账户等功能。普通用户在使用本系统时会涉及到一些需要请求在线服务器完成的任务，而为了防止任何请求都被响应的情况发生，服务器需要对收到的请求进行身份验证，只有已经注册在系统中的用户提交的请求才能被通过。因为本系统暂时仅限内部人员使用，所以暂时不允许用户主动进行注册，普通用户需要通过系统管理员完成账户注册的过程。在注册时管理员将用户账户、密码和用户信息填入对应数据库中，用户在和在线服务器进行交互时需要在请求中包含用户账户和密码，服务器进行核实后才会响应。管理员也可以通过用户管理模块查看用户的信息，删除现有用户账户等操作。

#### 4.3.9 数据管理模块设计

数据管理模块主要功能包括数据集的生成和系统涉及到的各种数据的管理。本系统涉及到的数据主要分为，超导瞬变电磁信号数据、训练数据集、模型数据以及用户数据。系统管理员可以利用数据管理模块对系统中的数据进行添加、移动、删除等操作。

### 4.4 数据库详细设计

数据库管理系统是一个能够提供数据录入、修改、查询的数据操作软件，具有数据定义、数据操作、数据存储与管理、数据维护、通信等功能，且能够允许多用户使用能实现配置管理系统的数据持久化，是系统的重要组成部分。对于超导瞬变电磁去噪系统，需要存储用户信息、数据集信息、模型信息等数据，引入数据库管理系统，可以方便开发人员建立、使用和维护数据，对数据进行统一的管理和控制，以保证数据的安全性和完整性<sup>[32]</sup>。本系统采用 MySQL 数据库来存储普通用户信息、管理员信息、数据集信息、模型信息等数据。

图4.15为本系统的 ER 关系图。

其中实体有：用户，管理员，模型，数据集。关系有：管理，使用，训练，生成各个实体之间的关系除了数据集训练模型为一对一以外，其他关系都为一对多。

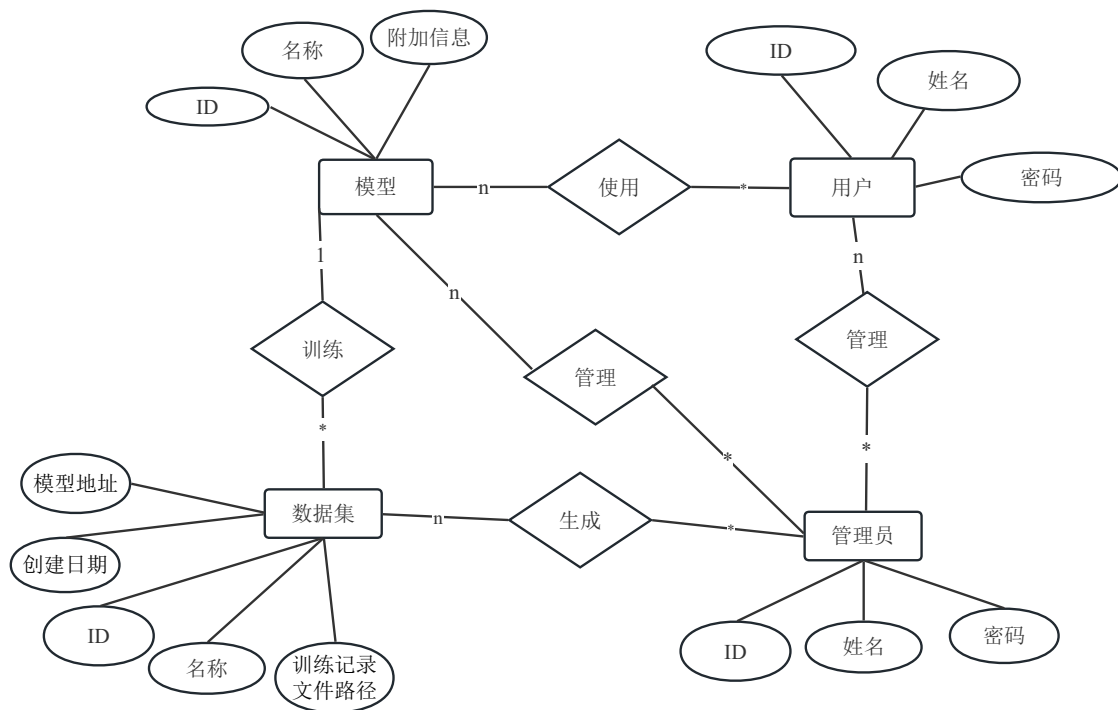


图 4.15 ER 关系图

#### 4.4.1 用户信息表

用户信息表记录了用户相关的信息，包括用户姓名，ID 和密码，这两个属性是用户在使用在线功能时必须的信息。其中用户 ID 作为该数据表的主键，其他属性都不能为空。用户的信息由系统管理员填入表中，需要修改时也通过系统管理员进行修改。用户信息表内容如表 4.2 所示。

表 4.2 用户信息表

字段	字段说明	类型	是否主键	是否为空	是否外键
id	用户 ID	Long	是	否	否
user_name	用户姓名	Varchar(50)	否	否	否
user_pwd	用户密码	Varchar(50)	否	否	否

#### 4.4.2 管理员信息表

管理员信息表与用户信息表类似记录了管理员相关的信息，包括管理员姓名，ID 和密码，管理员 ID 和密码是管理员登陆系统后台时必须的信息。其中管理员 ID 作为该数据表的主键，其他属性都不能为空。管理员的信息由初始管理员填入表中，之后的修改可由管理员自行完成，每个管理员都只能修改自己的信息。管理员信息表内容如表 4.3 所示。

表 4.3 管理员信息表

字段	字段说明	类型	是否主键	是否为空	是否外键
id	管理员 ID	Long	是	否	否
manager_name	管理员姓名	Varchar(50)	否	否	否
manager_pwd	管理员密码	Varchar(50)	否	否	否

#### 4.4.3 模型信息表

模型信息表记录了模型相关的信息，包括模型名称，模型 ID，模型训练记录的地址，模型保存的地址，模型创建日期和训练模型所用的数据集 ID，其中模型 ID 作为该数据表的主键，除模型训练文件和模型创建日期外的其他属性都不能为空。模型的信息由系统在生成模型时自动填入，也可以在模型添加的过程中由用户手动输入，模型的数据集 ID 作为数据集信息表的主键可以查询所用到的数据集信息。模型信息表内容如表 4.4 所示。

表 4.4 模型信息表

字段	字段说明	类型	是否主键	是否为空	是否外键
id	模型 ID	Long	是	否	否
model_name	模型名称	Varchar(50)	否	否	否
model_log	模型训练记录文件路径	Varchar(50)	否	是	否
model_path	模型地址	Varchar(50)	否	否	否
model_cre_time	模型创建日期	DateTime	否	是	否
dataset_id	数据集 ID	Long	否	否	是

#### 4.4.4 数据集信息表

数据集信息表记录了数据集相关的信息，包括数据集名称，数据集 ID，数据集描述，数据集保存的地址，数据集创建日期和一些生成数据集所用到的瞬变电磁与地下空间参数，其中数据集 ID 作为该数据表的主键，除数据集描述和数据集创建日期外的其他属性都不能为空。数据集的信息在系统生成数据集时自动填入，值与用户在生成数据集时所输入的参数保持一致。数据集信息表内容如表 4.5 所示。

表 4.5 数据集信息表

字段	字段说明	类型	是否主键	是否为空	是否外键
id	数据集 ID	Long	是	否	否
ds_name	数据集名称	Varchar(50)	否	否	否
ds_desc	数据集描述	Varchar(200)	否	是	否
ds_path	数据集地址	Varchar(50)	否	否	否
ds_create_time	数据集创建时间	DateTime	否	是	否
ds_mod_time	数据集修改时间	DateTime	否	是	否
ts_channel	电磁响应数据时间道数	Int	否	否	否
ts	电磁响应数据时间道	Varchar(50)	否	否	否
on_time	测量系统开启时间	Float(4)	否	否	否
off_time	测量系统关断时间	Float(4)	否	否	否
off_edge	测量系统关断沿	Float(4)	否	否	否
current	测量系统发射电流大小	Float(4)	否	否	否
emis_dist	测量系统发射点距离	Int	否	否	否
offset_dist	测量系统偏移距范围	Varchar(50)	否	否	否
res_range	数据集涵盖电阻率范围	Varchar(50)	否	否	否
layer_num	地下空间划分层数	Int	否	否	否
layer_thic	地下空间每层厚度	Varchar(50)	否	否	否

## 4.5 本章小结

本章对超导瞬变电磁去噪系统的各个模块进行了详细设计，梳理了各个模块所实现的具体功能，明确了各模块的工作逻辑，同时对数据库的设计进行了详细介绍。为下一步的详细实现起到了指导作用。

## 第5章 系统实现

在本章中以第四章系统架构设计为基础对整个系统进行实现，并描述系统关键模块的具体实现细节。

### 5.1 开发环境

在进行系统的详细实现之前需确定系统的开发环境，方便之后的开发工作。如第二章中提到的，本系统采用面向对象的模块化开发方法，主要编程语言为 Python。开发工作包括 web 端前后端的开发，客户端 GUI 开发，以及算法的研发工作，基于此确定了本系统的开发环境基于 Window 系统，使用 TensorFlow 框架开发算法，Django 作为后端框架，Vue.js 作为前端框架，PyQt 作为 GUI 开发框架。

### 5.2 数据录入模块

系统需要原始文件测量信息和原始数据的保存地址来将数据转化为可以处理和展示的数据。在输入原始数据时默认通过读取测量信息文件来进行数据输入。在具体实现时首先读取测量信息文件，信息文件以 CSV 文件存储，利用 Python 的标准库 Tkinter<sup>[33]</sup>生成弹窗选择文件路径，测量信息中会包含原始文件的储存地址，将地址输入解析原始文件函数，再通过 nptdms 库下的 TdmsFile 方法解析原始数据信号，就可得到原始数据文件，解析的信号以 DataFrame 形式返回方便后续处理。用户也可以选择手动进行参数输入和原始数据文件选择。

图5.1为数据录入模块类图。

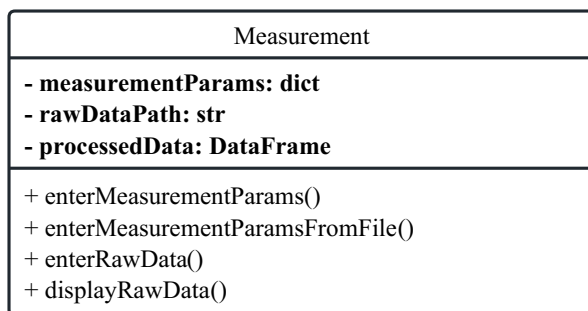


图 5.1 数据录入模块类图

在上述类图中，我们可以看到系统中的主要类为 Measurement 类，其包含了测量参数和原始数据相关的属性和方法。Measurement 类有三个属性，分别是 measurementParams，用于存储测量参数的字典；rawDataPath，用于存储原始数

据的本地路径；processedData，用于存储处理后的数据。

Measurement 类有四个方法，分别是 enterMeasurementParams()，用于手动输入测量参数；enterMeasurementParamsFromFile()，用于从本地文件中读取测量参数；enterRawData()，用于将原始数据与测量参数连接并存储进数据库；displayRawData()，用于通过可交互图片的方式显示原始数据。

图5.2和5.3分别为手动填入数据和数据选择弹窗时系统截图。

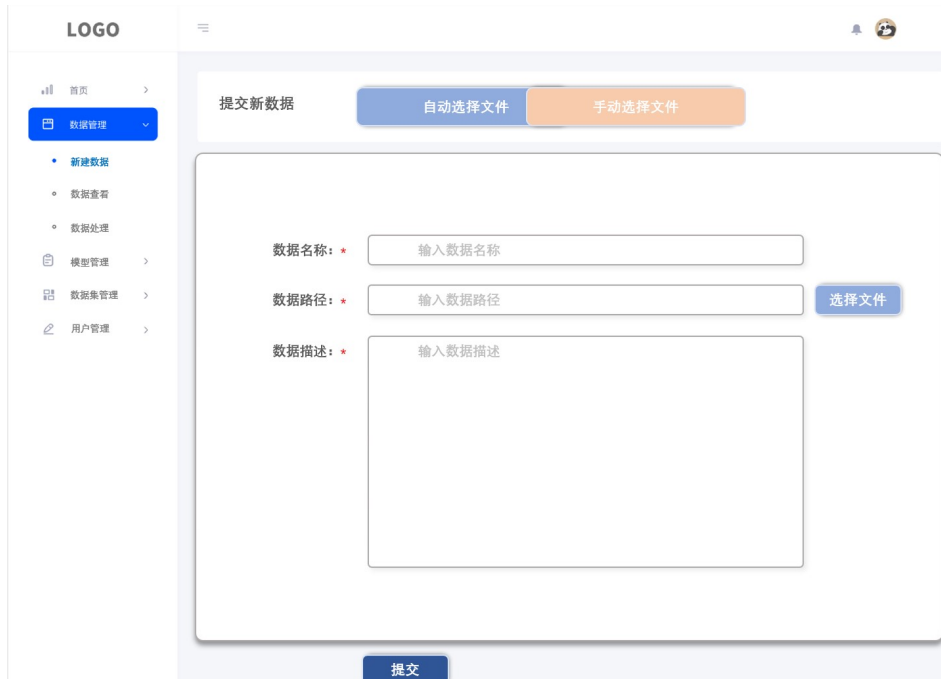


图 5.2 手动填入数据

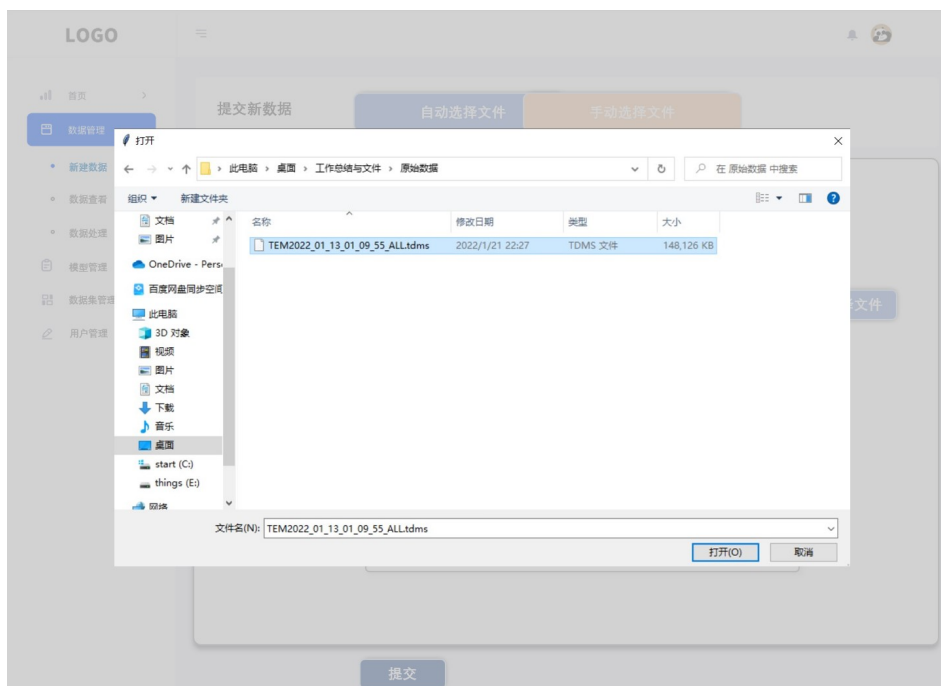


图 5.3 数据选择弹窗



### 5.3 预处理模块

预处理模块主要对输入的原始数据进行断层补偿，跳点检测修复，同步分割等操作。由于原始数据包含的数据点较多，出于性能考虑此模块在接受到数据后会将数据从 DataFrame 格式转化为 Numpy 矩阵，虽然预处理模块的三个主要功能是独立的，但在实际使用时，原始必须经过这三个流程才能转化为待去噪数据，且这三个流程中涉及到的参数都是根据原始数据的测量参数计算得到的。考虑到系统的实际的使用情况，在实现时会将这三个步骤依次调用并向用户屏蔽过程细节，直接展示处理完成的结果。表现在具体操作时，用户只需要选择需要预处理的原始文件，系统会自动进行预处理并返回最终的处理结果，过程中没有需要选择的参数，这样简化流程可以使测绘人员在实际的野外应用场景中快速确定是否需要重测某条数据，提升工作效率。

图5.4为预处理模块类图。

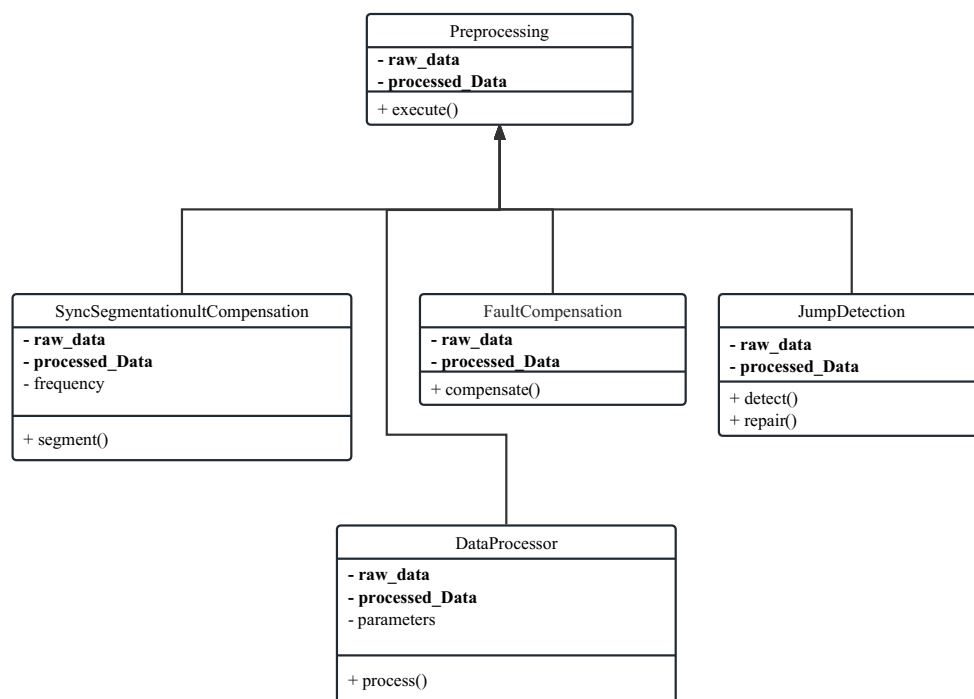


图 5.4 预处理模块

这个类图展示了 4 个类：Preprocessing, FaultCompensation, JumpDetection, 和 SyncSegmentation。另外还有一个 DataProcessor 类，用于调用三个预处理步骤来完成整个预处理过程。

Preprocessing 类是一个抽象类，定义了 raw\_data 和 processed\_data 两个属性，以及一个 execute() 抽象方法。该方法将会在 DataProcessor 类中被具体实现。

FaultCompensation 类实现了 Preprocessing 抽象类，并添加了 compensate() 方法，该方法用于执行断层补偿操作。

JumpDetection 类也实现了 Preprocessing 抽象类，并添加了 detect() 和 repair()

方法，该方法用于执行跳点检测和修复操作。

SyncSegmentation 类同样实现了 Preprocessing 抽象类，并添加了 segment() 方法，该方法用于执行同步分割操作。

DataProcessor 类用于将上述三个类的方法调用起来，完成整个预处理过程。它具有 raw\_data 和 processed\_data 两个属性，以及一个 parameters 属性，该属性存储了需要使用的参数（例如，分割频率等）。process() 方法用于调用三个预处理步骤，并返回最终的预处理结果。

接下来描述每段工作的详细流程。

### (1) 断层补偿

数据的断层对后续处理影响最大，所以在预处理过程中先进行断层补偿操作。具体实现时先进行相似度的检查，

相似度的计算函数如下：

```
1 def Signal_Check_Error(data_test, data_check, axis):
2     '''平均均方相对误差公式'''
3     V_j = data_test[:, axis] # 实验点数据
4     V_j_check = data_check[:, axis] # 检查点数据
5     V_average = (V_j + V_j_check) / 2 # 实验点与检查点之间平均值
6     Mse = np.sqrt((np.mean((V_j - V_j_check) / V_average) ** 2) / 2) # 单点
7     # 均方相对误差
8     return Mse
```

将计算的误差与预设的阈值进行对比后，大于阈值周期的数据会被标记出来并被删除，之后用相邻正常周期的数据进行填充，修复后的结果如图5.5所示。

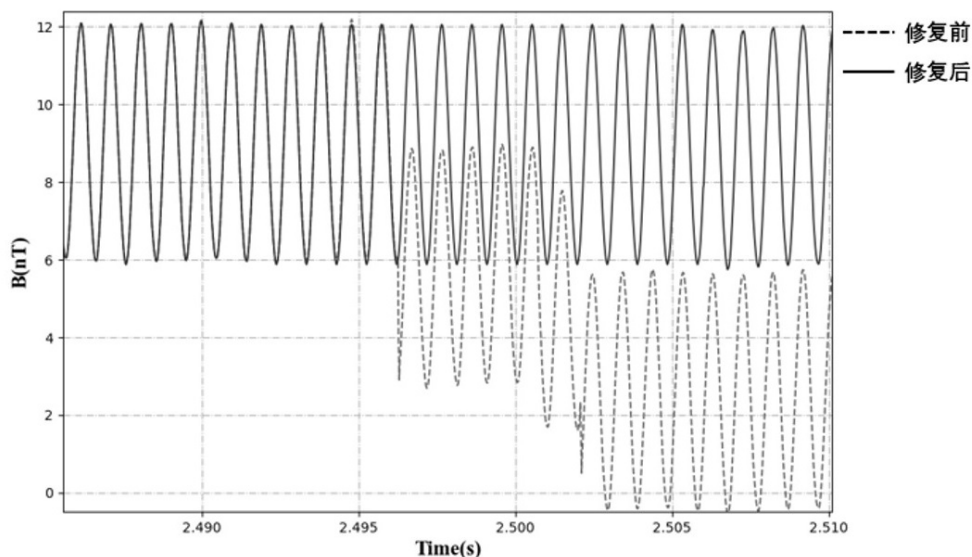


图 5.5 断层修复对比

## (2) 跳点修复

原始数据中存在跳点或离群点会对后续的数据处理产生很大的影响，跳点或离群点通常随机出现，无法用断层补偿中用到的相邻周期相似性检测的方法定位，但原始数据是一个周期且连续的一维数据，可以用计算差分的方法来检测跳点的出现。具体操作时对原始信号进行二阶差分，从二阶差分后的信号中提取出最大值与最小值并将两者差值的绝对值作为本周期信号的跳点特征值，组成坏点特征值集合，设定参数构建 Kmeans 聚类模型进行聚类，在跳点特征值集合中自动找出偏离点，从而锁定坏跳所在周期。定位跳点之后将跳点的值用前后周期对应位置的平均值进行替换，从而完成跳点的修复。

跳点修复后的结果如图5.6所示。

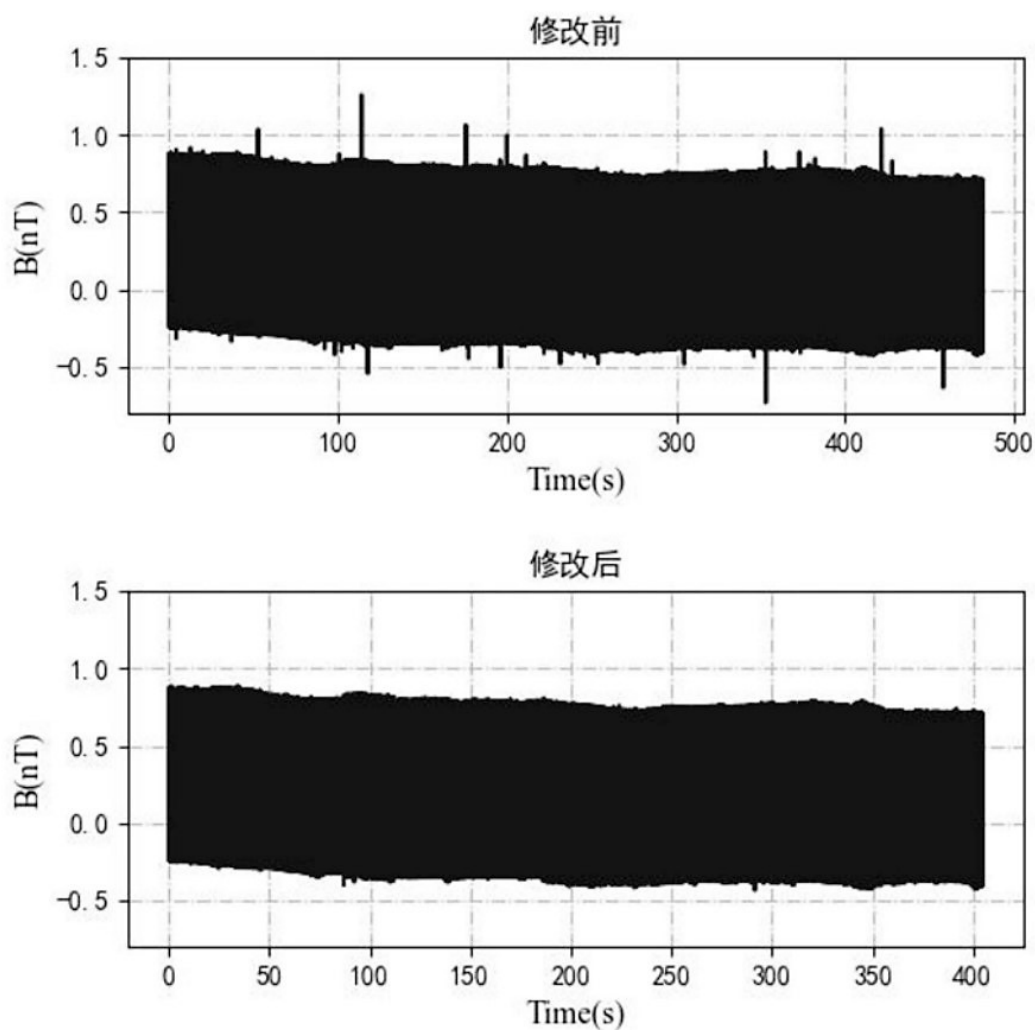


图 5.6 跳点修复结果

### (3) 同步分割

通过前两个步骤将原始数据处理完整之后，就可以将原始数据转化为待处理的数据。如第四章中提到的，原始数据是一个周期信号，将数据按周期切割叠加就可得到我们需要去噪的数据，叠加的过程也可以去除一些周期性的噪声。具体实现时由于原始数据不一定刚好整数个周期，所以先进行取整，同时考虑到性能，再实现时避免时用循环来叠加，而是利用 `numpy` 的矩阵操作进行切割和叠加。

同步分割后的结果如图5.7所示。

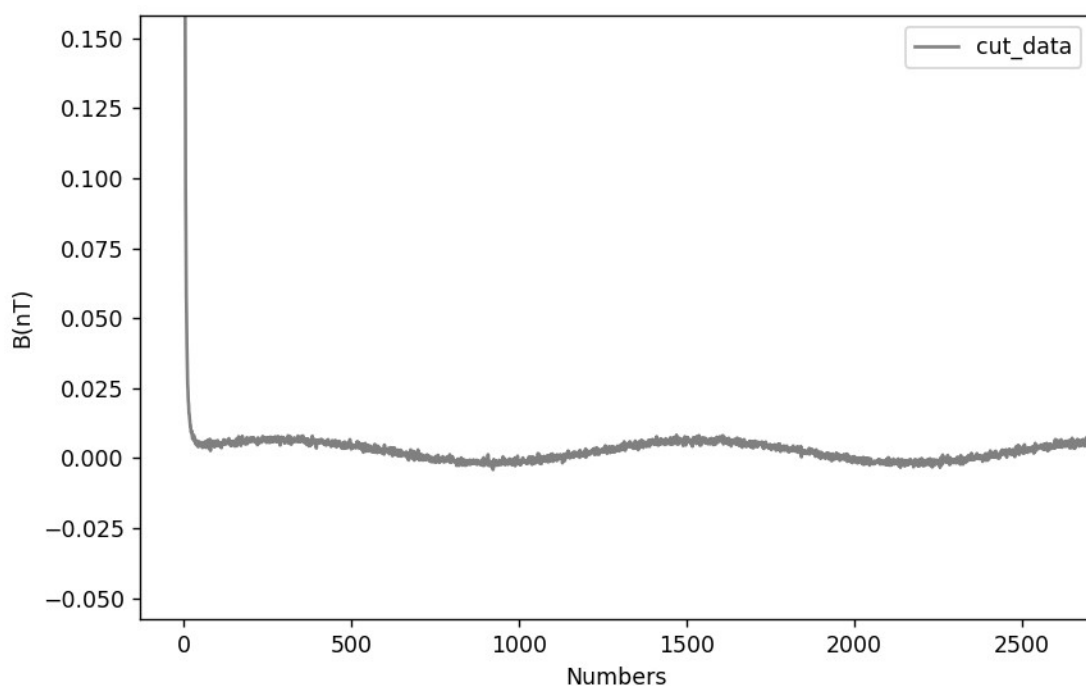


图 5.7 同步分割结果

## 5.4 去噪模块

用户选择经过预处理的数据后就可以用去噪模块对数据进行去噪操作，此阶段用户可以指定去噪模型，如图在模型选择界面可以查看已经在系统中的模型，用户可以在此界面选定模型，模型以 `h5` 文件的模式保存，选定模型时系统会调用 `keras` 模块的 `models.load_model` 方法，这样可以避免加载整个 `TensorFlow` 库造成不必要的内存和性能开销。使用添加模型的功能将本地模型添加到系统中，选定模型之后便可进行去噪操作。因为模型的输入是根据模型结构决定的固定值，而实际需要去噪的数据的长度并不确定，所以在输入模型之前需要对数据进行格式统一，去噪的数据都是一维数据，当数据长度与模型的输入不一致时对数据进行长度转化。标准的模型输入是一个 17500 的张量，在数据输入模型之前对数据的长度进行判断如果长于 7500 对数据进行下采样操作，实现时用

Pandas.date\_range 函数重构数据时间轴并用 Pandas.resample 进行重新采样。当数据小于所需长度时用 scipy 库中的 interpolate.UnivariateSpline 直接对数据进行插值。

训练模型的数据都经过归一化处理，实际应用时去噪的数据也需经过归一化处理，但由于数据的特殊性，简单通过最大最小值来归一效果不佳。根据瞬变电磁信号理论数据是一个指数衰减函数的特点，将数据第一个值定义为最大值，最后一个值定义为最小值来归一化会更符合理论情况，虽在实际测量的数据不完全符合理论情况，最大最小值未必是第一个和最后一个值，但在实际实践中这样的归一化方法表现更加突出，这也是本系统所采用的归一化方法。

归一化的代码如下：

```

1 def MinMax( data , s=1,e=0.0011):
2     N = len( data[0])
3     b = int(N*0.99)
4     ret = np.empty( data.shape)
5     num_list = np.empty((len(data),2))
6     for i in range(len(data)):
7         start = data[i][0]
8         end = data[i][b]
9         temp = data[i] * ((s-e)/(start-end))
10        num_list[i][0] = ((s-e)/(start-end))
11        num_list[i][1] = (temp[b]-e)
12        temp = temp - (temp[b]-e)
13        ret[i] = temp
14    return ret , num_list

```

代码中 start 和 end 分别代表第一个点和最后一个点对应的数据值，归一化后这两个数据对应的值分别为 0 和 1，其他数据等比例缩放。

除了本地去噪外用户也可以使用在线去噪功能，与本地去噪一样也需要在本地进行长度转换，之后上传数据和模型参数，得到去噪后结果，图5.8和图5.9分别为去噪前后全局对比和去噪前后局部对比。

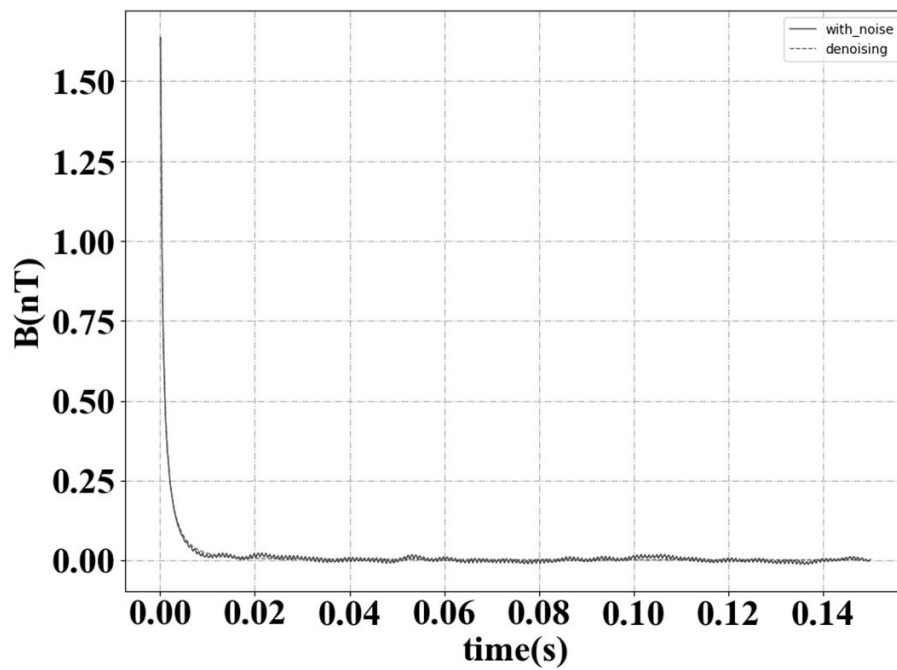


图 5.8 去噪前后全局对比

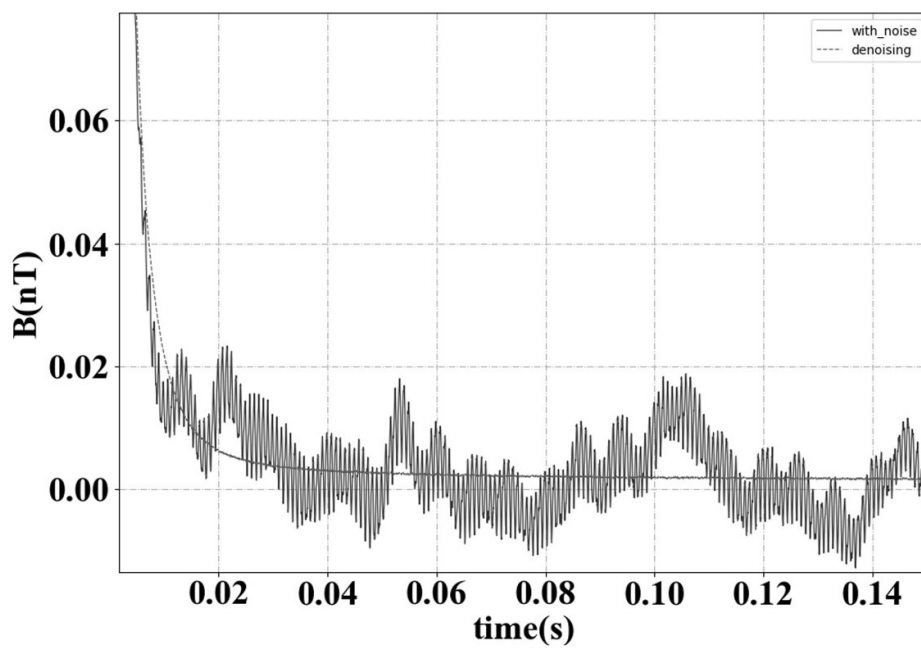


图 5.9 去噪前后局部对比

## 5.5 数据可视化模块

此模块负责各种数据的成图，本系统涉及到的成图数据主要为数据处理过程中的各种一维时序数据，以及对处理结果进行评估时去噪前后噪声水平评估数据。对于大部分二维图形成图本模块采用 Python 的第三方可视化库 Plotly 绘制，利用 Plotly 库可以很轻松的制作交互式的图形，对本模块涉及到的折线图、散点图、柱状图都有很好的支持，此外其实现是基于 Django 框架，非常方便将其成图嵌入网页中<sup>[34]</sup>。虽然 Plotly 可以满足大部分成图需求，但在面临大超数据时会产生卡顿，在实现过程中发现在生成一部分较大的原始数据时会有明显卡顿，所有对于原始数据的成图采用 matplotlib 库实现，此可视化库对大数据的支持较好。

考虑到此模块涉及到多数据成图，在实现时将画图函数统一接口，固定需要传入的参数为：成图类型，x 轴 y 轴名称，图像名称以及数据列表方便每次调用，数据列表中的一个元素代表要绘制的一条数据，内容包括 x 轴值，y 轴值，标签，颜色等等信息。以下为成图函数的核心代码

```

1 def show_line( data_list ,xname,yname,title ): #折线图函数
2     plt.title(title)
3     for item in data_list: #加载数据列表内数据
4         plt.plot(item['X'], item['Y'], color=item['color'], label=item['label']
5                 )
6     plt.legend() # 显示图例
7     plt.xlabel(xname,fontdict={"family": 'Times New Roman', 'weight': 'bold'
8                               , 'size': 14})
9     plt.ylabel(yname,fontdict={"family": 'Times New Roman', 'weight': 'bold'
10                              , 'size': 14})
11    plt.grid(linestyle='-.')
12    plt.show()
13
14 def load_show( data_list ,xname,yname,title ,show_func): #绘图接口函数
15     func = fun_dic[show_func] #读取成图函数
16     func(*[ data_list ,xname,yname,title ,show_func ])

```

其中 load\_show() 函数为绘图接口，接口中的 show\_func 变量是之后成图需要调用的具体成图函数，这样不管是绘制怎样的图，一共有多少数据都可以用这个统一接口实现。

图5.10、5.11、5.12 为分别为数据可视化模块主要生成的三种绘图示例。

## 5.6 模型训练模块

此模块实现时主要包含的步骤有理论数据生成，噪声添加，和模型训练。同预处理模块一致本模块在实际调用时也会向用户屏蔽过程细节，用户只需要提前设定好理论地下空间参数，数据集大小和噪声范围即可，系统会根据这些参数自动训练模型并将训练的模型保存在数据库中，下面分步骤介绍训练过程的详细实现。

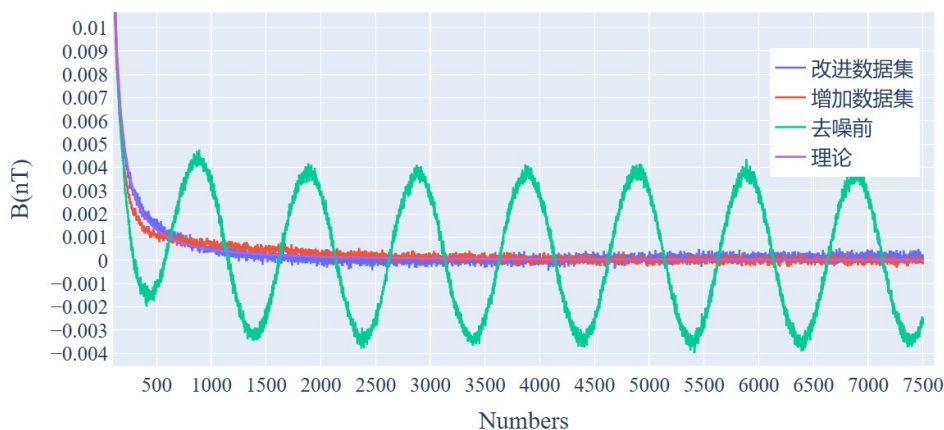


图 5.10 多条数据去噪对比

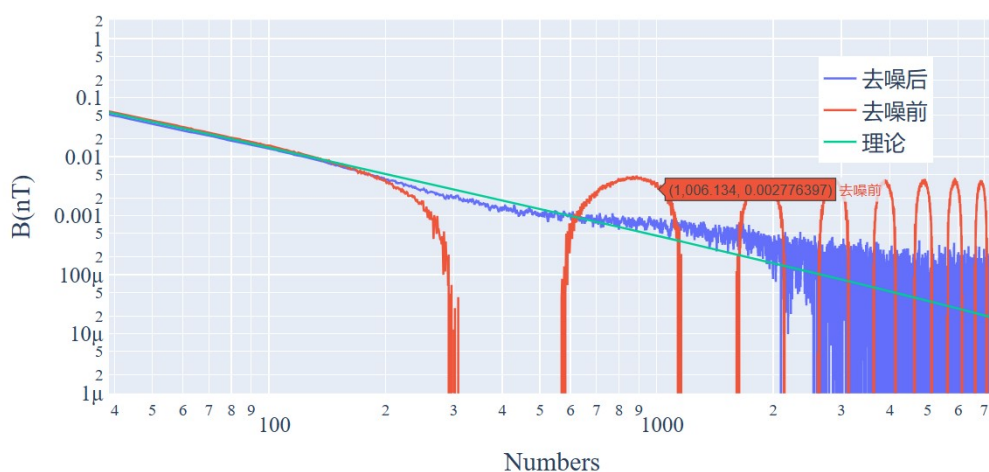


图 5.11 数据去噪前后对比 (log 轴)

### (1) 理论数据生成

首先进行理论数据生成，如第四章中提到的理论数据是由一个正演算法生成的，此算法较为复杂且计算量大实际运用时会把算法打包成 `exe` 文件，需要生成时直接调用。在调用时需要传入一些想要模拟地质层数据的参数。一般生成一次数据集会有百万条数据，所以在调用 `exe` 程序时会采取多进程调用，多进程调用的 `exe` 产生的数据分别保存在单独的文件中，所有进程任务结束后再进行数据的合并，这样可以避免多进程文件同步的问题。

数据生成后可以对数据进行优化，正演模型是按照给定参数模拟地下空间来生成数据的，数据的分布也会和模拟的分布一样，然而这样数据直接作为数据集会导致数据偏向大概率产生的情况，而在实际情况中，我们希望数据集中的数据分布越均匀越好，这样模型对任何条件下的数据都有很好的预测能力。所以对生成的数据集中的数据进行聚类，根据数据集的大小一般分为 9 类，每一类选取 1100 条数据，如果某一类没有达到这个阈值则全部选中。最后得到一个大小为 10000 的数据集，此数据集作为最终模型训练的 `label`。



实际噪声去噪前后噪声压制对比

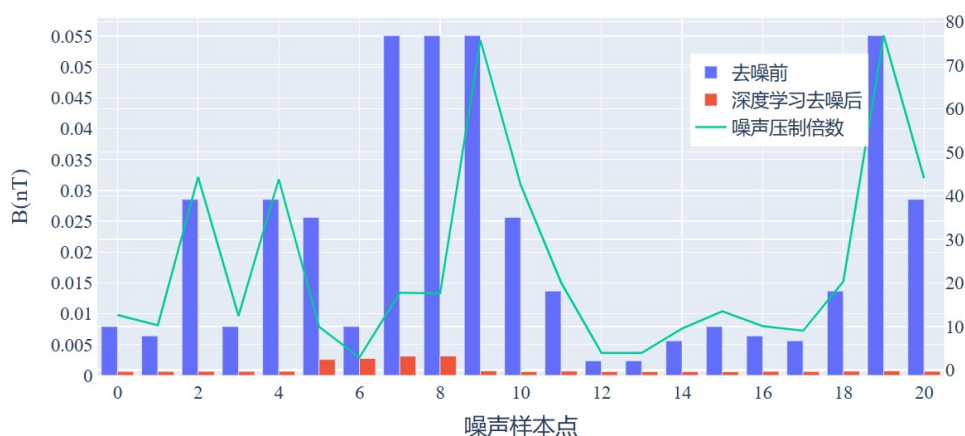


图 5.12 去噪前后指标

数据集聚类结果如图5.13所示。

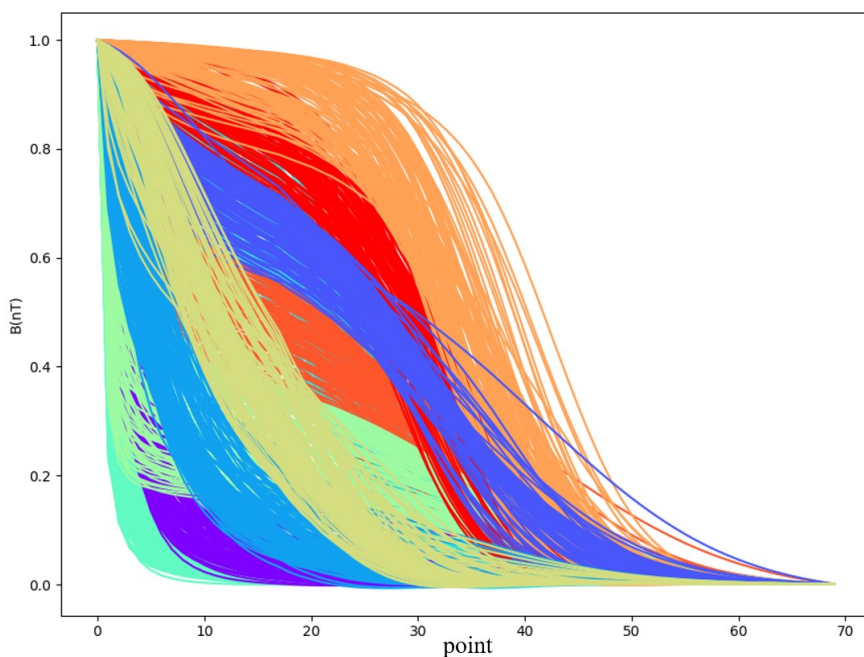


图 5.13 数据集聚类结果

图中不同颜色数据代表聚类的不同结果，可以看出不同结果数据的分布有很大区别。

## (2) 噪声数据生成

在得到理论数据的 label 之后，需要叠加噪声生成训练数据，噪声的分布应该尽可能模拟真实的噪声，基于前期的研究将添加的噪声分为三种噪声的叠加，分别是高频工频噪声、低频工频噪声和白噪声。其中工频噪声是噪声水平 10-60pT 频率 10-75Hz 的正弦波，白噪声为 10-20pT 高斯白噪声。随机组合产生大小 10000 条的数据集，将这些噪声和理论数据。

图5.14和图5.15分别为真实测得的噪声示例与模拟噪声示例。

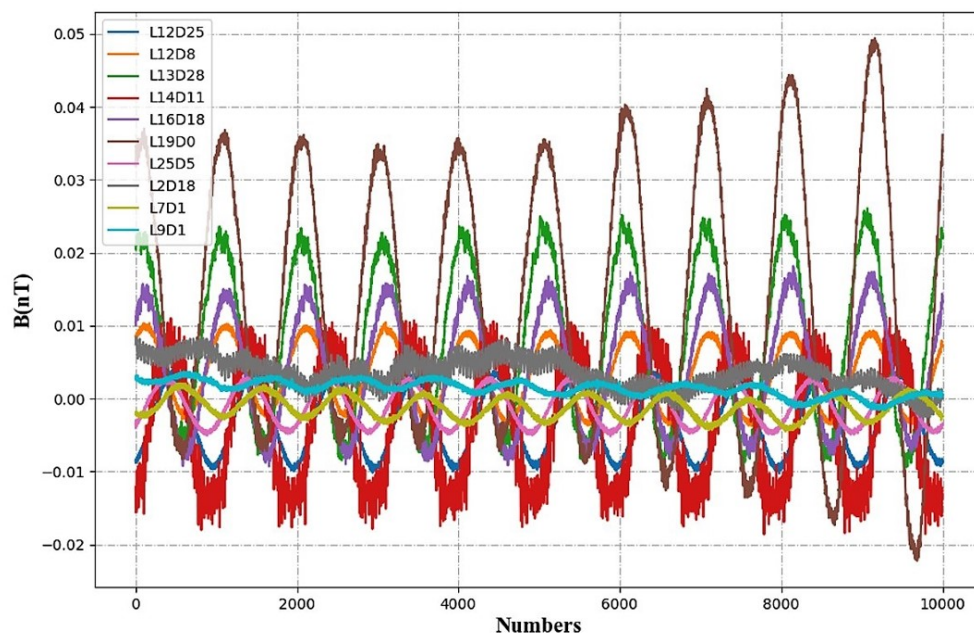


图 5.14 野外实测噪声示例

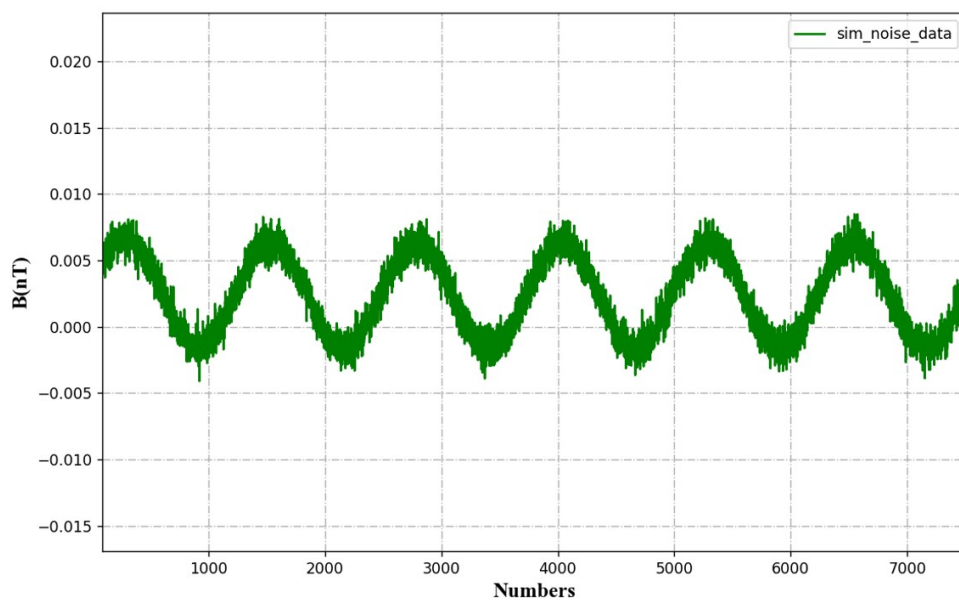


图 5.15 模拟噪声示例

## (3) 模型训练

模型训练环境为如表 5.1 所示：

**表 5.1 模型训练环境**

类别	配置
CPU	6 x Intel Xeon Gold 6142
GPU	RTX 3080 12GB 显存
内存	26GB
操作系统	Ubuntu20.04
软件环境	CUDA11.2, TensorFlow2.7.0, Python3.9

模型训练过程中标准数据集为 10000 条数据，将这 10000 条数据分为两个部分，80% 作为训练集，20% 作为验证集，根据实际训练数据的特点，损失函数采用了自定义的 MSE 函数，自定义部分为每个维度的权重不一样，数据前段波动更大且信息量更大，所以设置了较大的权重。

自定义损失函数如下：

```
1 def customized_mse(y_true, y_pred):
2     a_num = y_true.shape[2]
3     x = 40
4     a_list_0 = np.linspace(8,6,x)
5     a_list_1 = np.linspace(1,1,a_num-x)
6     a_list = np.hstack((a_list_0, a_list_1))
7     sq = tf.square(y_pred - y_true)
8     a_list_temp = a_list.reshape((-1,1,a_num))
9     return tf.reduce_mean(a_list_temp * sq)
```

其中 `a_list_0` 代表第一阶段的损失权重是一个线性减少的值，`a_list_1` 代表第二阶段的损失权重均为 1，平方和求平均操作均调用 TensorFlow 库提供的函数，这样可以将自定义函数整合入训练过程。

由于本系统设计的神经网络模型包含 CNN 结构，但处理的是一维度数据，所以在选择填充方法时需要选择镜像填充，TensorFlow 库中的 CNN 的可选填充方法中只有固定值和零填充，在模型搭建时需要实现自定义的填充方法。

自定义镜像填充函数如下：

```

1 def pad(a ,kernel_size , strides=1,dilation_rate=1,mode = 'SYMMETRIC'):
2     strides=1
3     dimen = len(a.shape)
4     w = a.shape[1]
5     f = kernel_size*dilation_rate
6     s = strides
7     pp = (w-dilation_rate)*s-w+f
8     if(pp%2==0):
9         p1=p2=pp//2
10    else:
11        p1=pp//2+1
12        p2=p1-1
13    if(dimen == 3):
14        a = tf.pad(a,[[0,0],[p1,p2],[0,0]],mode=mode)
15    elif(dimen == 4):
16        a = tf.pad(a,[[0,0],[p1,p2],[p1,p2],[0,0]],mode=mode)
17    return a

```

训练时采用 **adam** 优化器和指数衰减学习率和早期停止方法，训练的最大迭代数为 7000epoch，当连续 50 个 epoch 的 loss 没有减少时认为模型收敛结束训练，如果达到最大迭代次数则认为模型没有收敛，此时可以放弃训练或选择调整学习率等参数后重新训练。

图5.16为模型训练的流程圖：

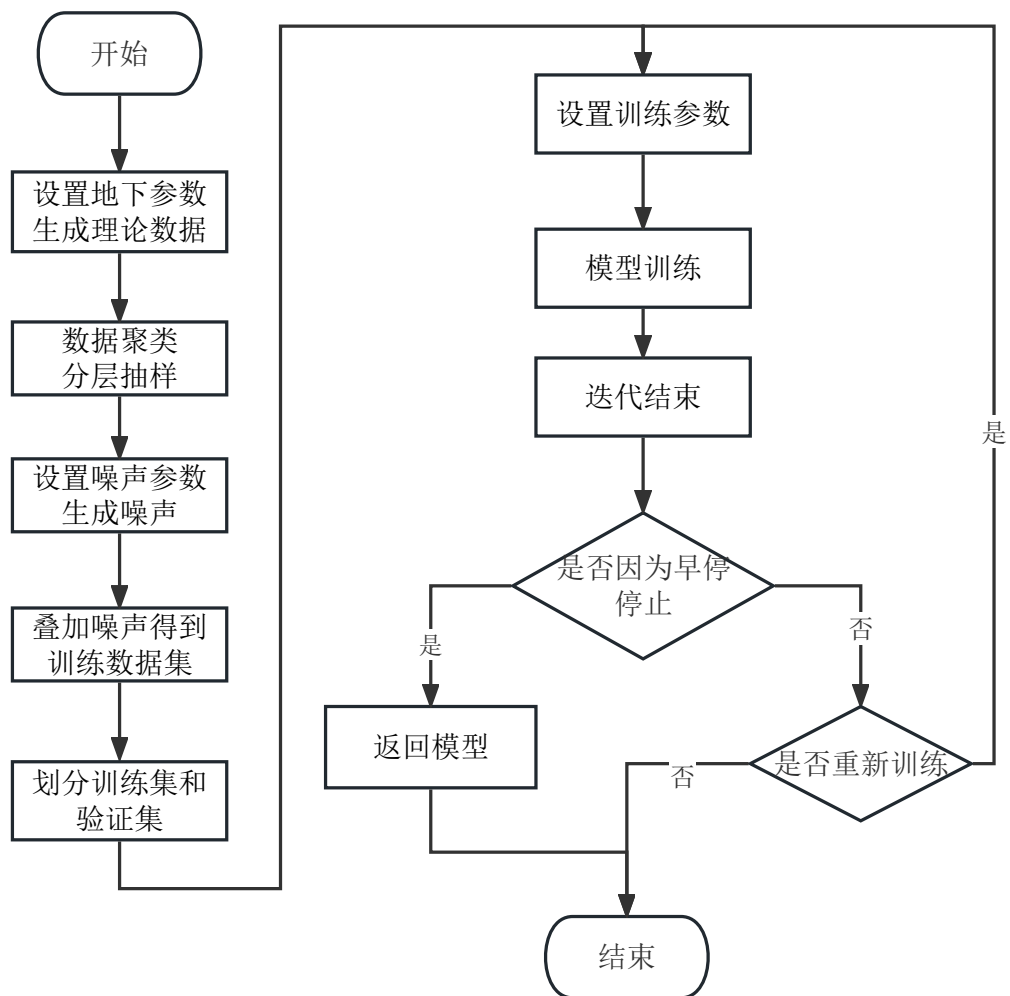


图 5.16 模型训练流程图

## 5.7 在线预测模块

在线预测功能调用前，需要用户更新在线模块的信息，具体过程为：用户在去噪界面点击在线模型更新。之后在线预测模块会发送一个请求到服务端后台，本系统用户客户端和服务端通信通过 HTTP 请求和响应实现。HTTP 请求在 QT 框架中相应的类为 `QNetworkRequest`，HTTP 响应相应的类为 `QNetworkReply`。发送的请求中包括信息有：用户账户，密码，读取现有模型指令。服务器在接受到请求后会验证用户账户，密码。如果用户存在模块会执行读取现有模型指令，并返回客户端服务器现有的模型列表。在线预测模块会将返回的信息加入到在线模型列表中，这些返回信息并不包括模型本身，只有模型名称，ID，描述等参数。在客户端发送请求时会设置一个 2000ms 的超时时间，超过这个时间没有得到响应会认为更新模型失败并弹出提示。

当用户在模型列表中选择在线模型并点击去噪后，客户端会发送一个带有去噪数据和模型 ID 以及去噪命令的请求，同样的服务器会对相应的信息进行验证，通过后会尝试调用模型 ID 所指的模型对去噪数据进行预测，成功会直接返回预测值及去噪后数据。客户端接受到后会提示去噪完成。若用户没有及时更新在线模型列表，选择了一个服务器已经删除的模型，服务器在匹配模型 ID 时会匹配失败并返回模型不存在请更新模型列表的信息。

## 5.8 数据管理模块

管理员在系统后台可以看到所有在系统中的数据，如第四章中提到的本模块主要维护的数据表包括用户信息，数据库信息，数据集信息，模型信息，数据点信息。对于每个信息表，本模块都提供了查询更改和删除的功能。

图5.17为查看数据列表时系统截图。

除此之外本系统涉及到的数据，特别是原始数据名包含丰富信息，为了有效利用这些信息，数据管理模块实现了对名称关键字提取的功能，提取的关键字为时间，测量地点，类型等。提取的信息会作为数据附加信息的一部分，方便用户查看。

信息提取的实现代码如下：

```

1  def get_information(name,key_words_list):
2      name_list = name_list.split('_')
3      key_words_info = {}
4      for keyword in key_words_list:
5          for word in name_list:
6              if(keyword in word):
7                  temp = word.split(':')[1]
8                  key_words_info[keyword] = temp
9  return key_words_info

```



图 5.17 数据列表界面

代码中 `key_words_list` 包含了所有要提取的关键字，本系统数据的命名规则都是关键字后为冒号，关键字之间为下划线如 `location:yunnang_date:2022-12-03` 所有查找时先用下划线分割文件名，匹配到关键字后再用冒号分割就可得到关键字对应的信息。

## 5.9 本章小结

本章通过梳理数据处理过程中重要模块的实现方法、核心代码和部分系统运行截图表述了系统模块的具体实现细节，并从数据预处理，数据集生成，模型训练这几个步骤表述了深度学习去噪算法的模型训练过程，在下一章中会对本章所实现的模块进行完整的测试并给出相应的测试结果。

## 第6章 系统测试

系统测试是正式交付系统前的重要步骤，此阶段旨在发现编写代码中的逻辑错误，检验功能性不足，确定是否系统不满足需求的地方<sup>[35]</sup>。本章主要对所实现的基于深度学习的超导瞬变电磁去噪系统进行测试，本章从测试方法与环境、测试用例以及测试结果三个步骤介绍测试工作，确定系统的不足之处并加以改进。

### 6.1 测试方法与环境

#### 6.1.1 测试方法

考虑到本系统主要用户为专业的测绘人群，对本系统所使用的深度学习算法有相应的标准，本文在测试系统之前先对算法进行性能测试，保证使用过程中算法的可靠性。对系统的测试主要采用黑盒测试的方法，即在不考虑模块和程序的内部结构与逻辑的情况下，对系统的功能进行测试。主要测试的项目包括系统的各个功能是否正常完成了规定的工作，是否得到了预期的结果，对于异常的输入与数据是否能进行辨别，遇到错误操作时是否能给出相应的反馈与提示。

#### 6.1.2 测试环境

表 6.1 系统测试环境配置表

类别	配置
CPU	Intel CPU 通用处理器
CPU 主频	>2.5GHz
内存	>4GB
硬盘	>20GB
操作系统	Windows 10

### 6.2 系统测试用例

#### 6.2.1 算法测试

算法测试数据分为仿真模拟数据和真实数据，所有数据从未在训练中使用过。测试时采用均方差 (MSE) 作为算法准确性量化指标衡量算法去噪后数据与理论数据的差距，除此之外信噪比 (SNR) 也常作为去噪任务中的衡量指标，其



定义如公式 6.1 所示为数据噪声和信号的功率或振幅的比值<sup>[36]</sup>，测试时选择信噪比作为性能量化指标，衡量算法对数据中噪声的压制效果。

SNR 定义:

$$SNR(dB) = 10 * \log_{10} \frac{P_{signal}}{P_{noise}} = 20 * \log_{10} \frac{A_{signal}}{A_{noise}} \quad (6.1)$$

其中

$P_{signal}$  为信号功率 (Power of Signal)。

$P_{noise}$  为噪声功率 (Power of Noise)。

$A_{signal}$  为信号振幅 (Amplitude of Signal)。

$A_{noise}$  为噪声振幅 (Amplitude of Noise)。

#### (1) 仿真数据测试

测试数据中理论数据分为两类数据，仿真数据和模拟地下空间生成数据，前者由算法随机参数生成，数据分布广泛但没有物理意义，用于检测模型的泛化能力。后者由特定的地下空间计算得到，数据较少，分布较窄但有对应的地下空间，可以用于成图来检测算法对后续成图等流程的影响。用于检测的噪声分为模拟噪声和实际噪声，模拟噪声模拟实际噪声的分布情况生成，其噪声分布与实际噪声保持一致，实际噪声则直接由野外采集得到。

所有测试的模拟噪声均匀分布在不同的噪声水平下，通过计算去噪后数据与理论数据的 MSE 和去噪前后的 SNR 比值检验算法。

①模拟噪声仿真数据测试，测试数据为未训练的仿真数据添加模拟噪声进行测试

在测试的 11000 个数据中，平均 SNR 提升 42.05，MSE 降低 17.29 倍，11000 条数据的去噪用时开销为 2 秒，均达到设计标准。图6.1和图6.2分别为去噪前后 SNR 对比和 MSE 对比，图中数据点按照去噪前的噪声水平排序。

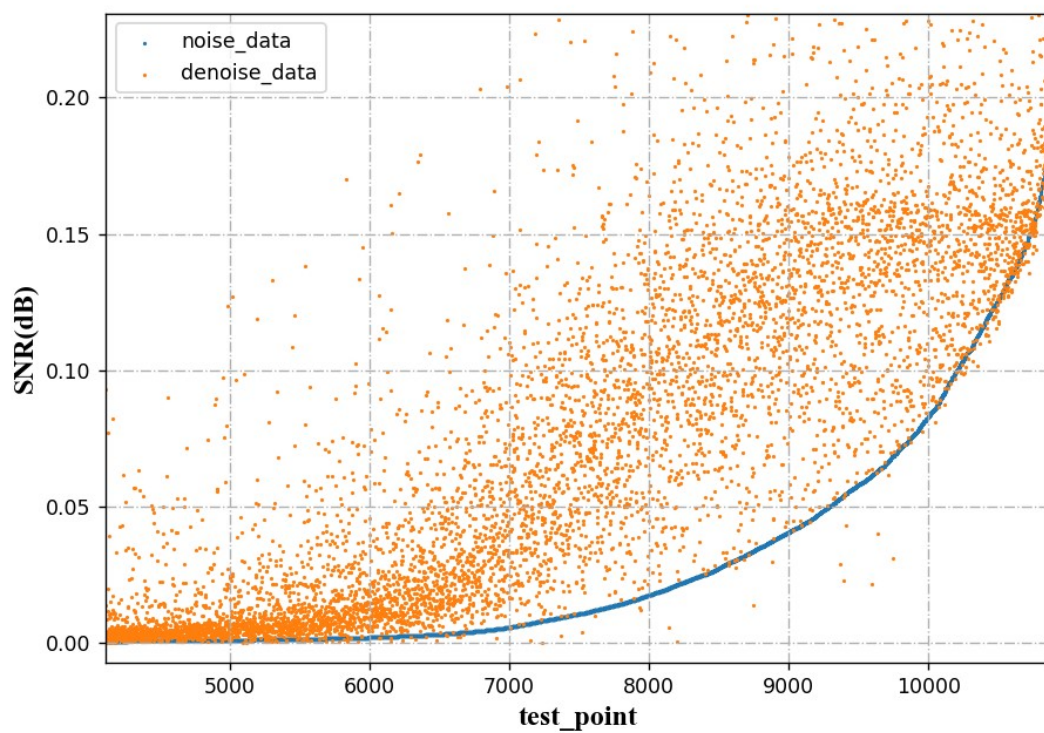


图 6.1 去噪前后 SNR 对比

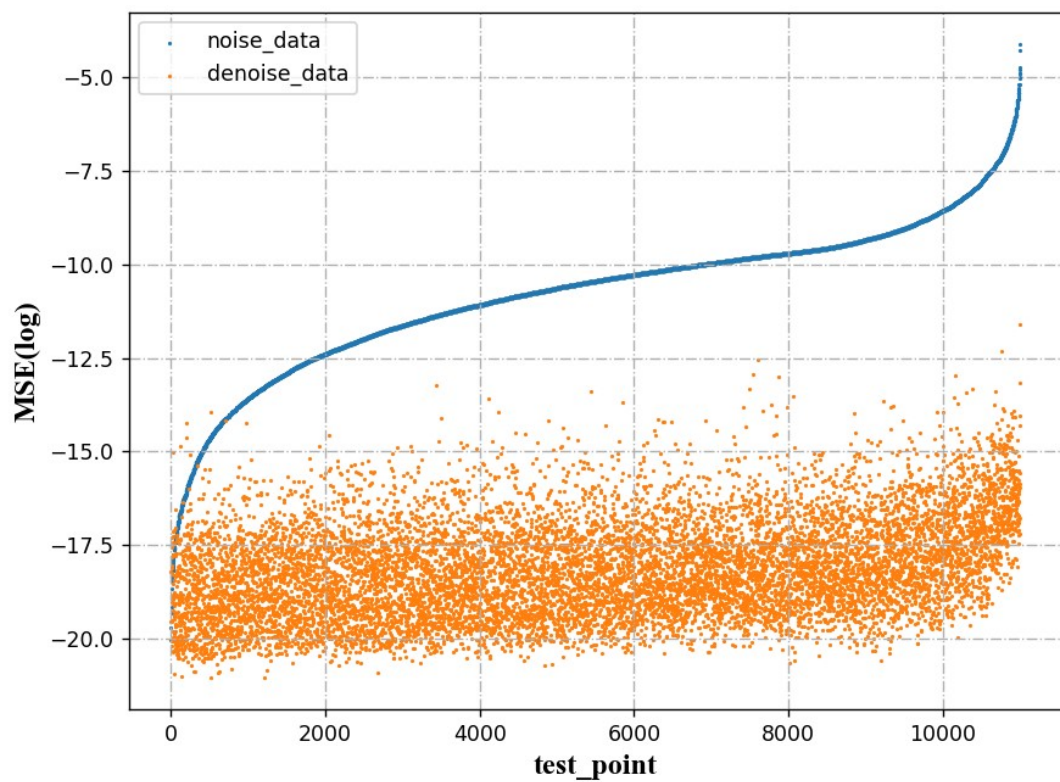


图 6.2 去噪前后 MSE 对比

②模拟噪声模拟地下空间测试，模拟一个已知的地下空间，进行一维正演模拟，得到理论电磁响应数据，对此数据添加添加野外的实测噪声进行测试，对于一组有物理意义的瞬变电磁信号数据，会通过绘制剖面图的方法测试算法的性能，测绘人员通过剖面图便可判断去噪效果好坏。

图6.3为去噪前后剖面图对比，去噪后剖面图越接近理论越好。

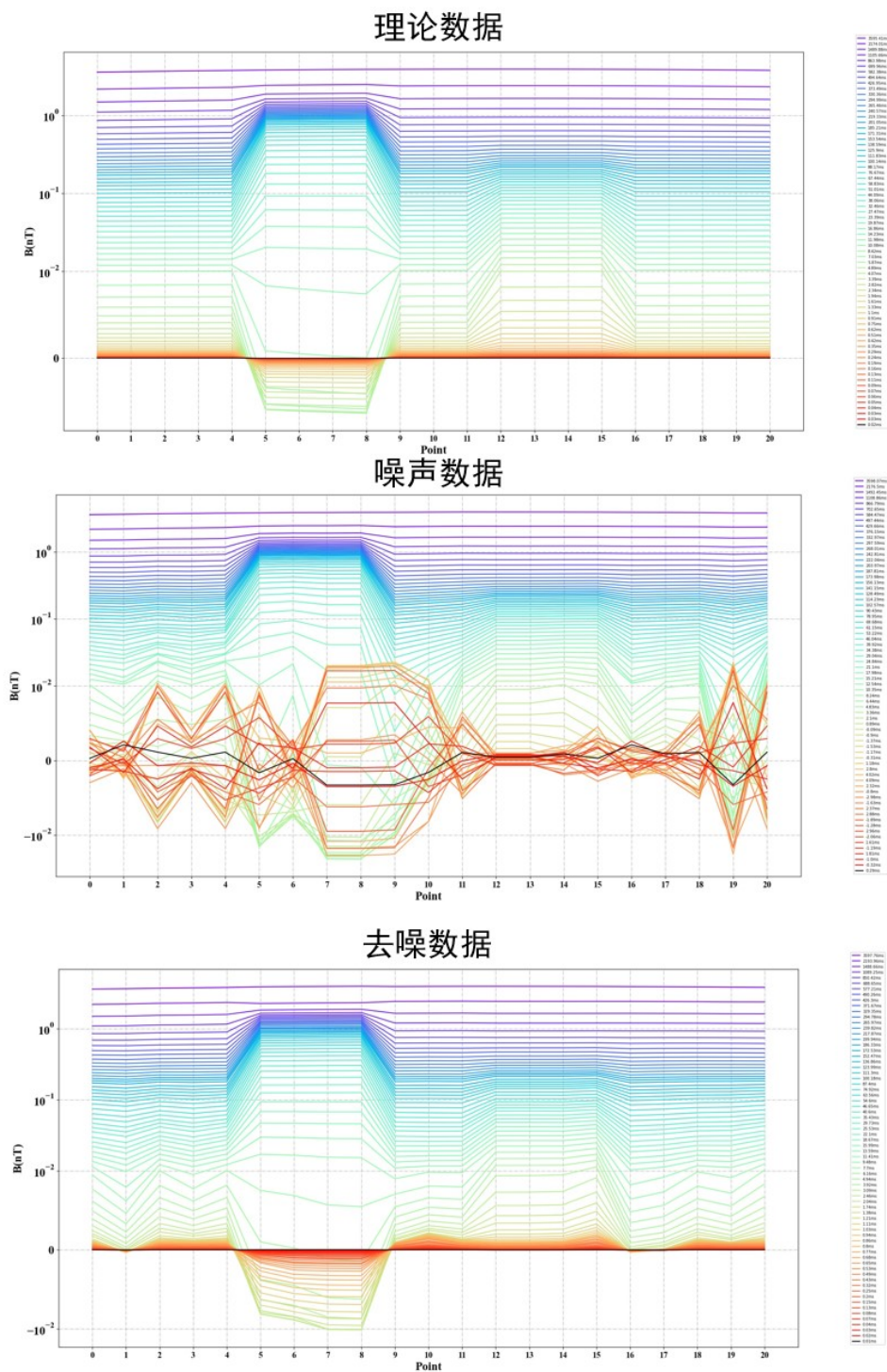


图 6.3 去噪前后剖面图对比

## (2) 真实数据测试

对野外实测数据进行去噪测试，由于测试的野外数据的地点还没有被开采或探秘过，所以只对比去噪前后的剖面图。

图6.4为去噪前后剖面图对比，去噪后剖面图越接近理论越好。

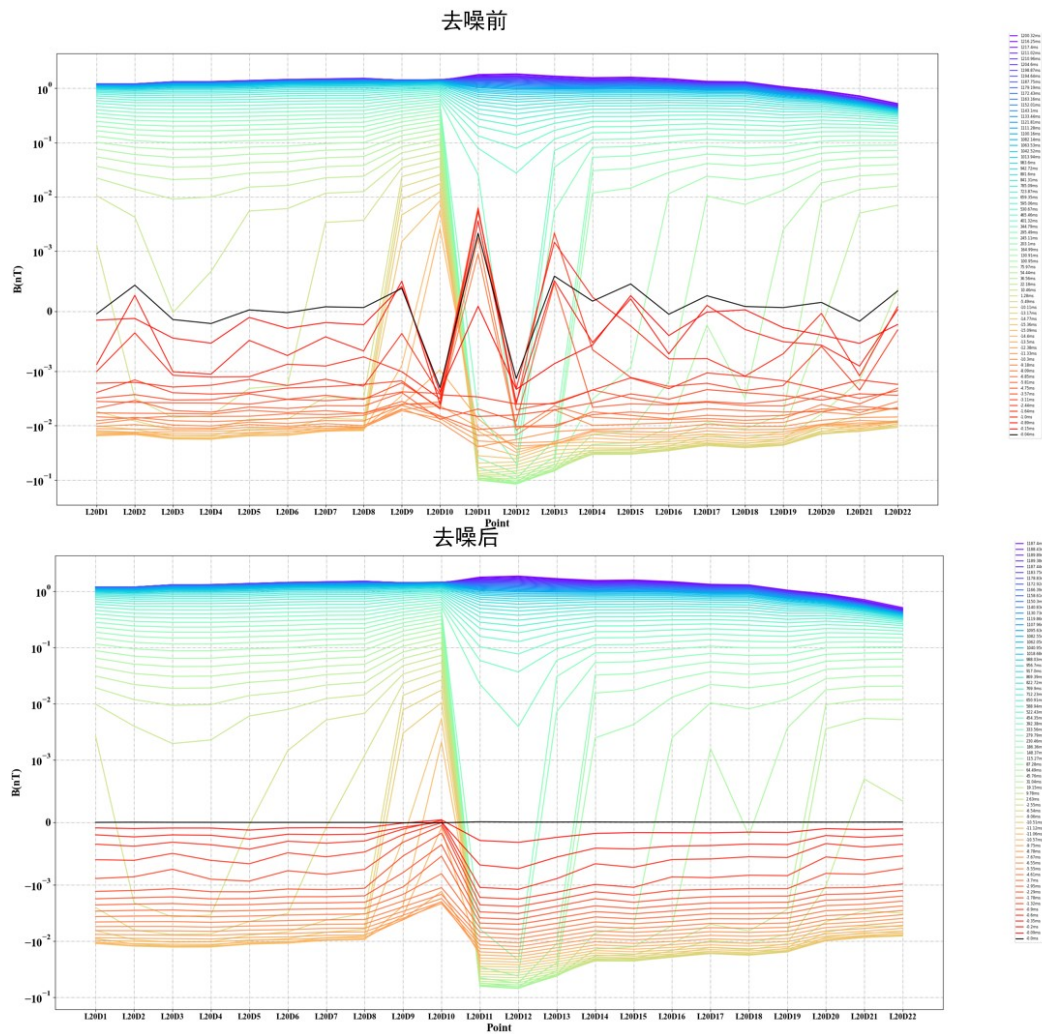


图 6.4 实测数据去噪前后剖面图对比

实测数据去噪结果由测绘人员检验，效果良好可以作为实际应用的去噪算法。



### 6.2.2 数据录入模块测试

该测试主要测试原始数据在录入系统时用户是否能够顺利的完成录入操作，相关的参数是否能顺利读取，遇到错误信息和数据时是否能给出提示。测试用例如表 6.2 所示。

**表 6.2 数据录入模块功能测试用例表**

编号	输入	预期结果	实际结果
1	选择自动输入数据	弹出选择文件界面	同预期结果
2	选择手动输入数据	弹出参数输入界面	同预期结果
3	选择参数文件	将对应文件添加至数据库中并提示添加成果	同预期结果
4	输入错误参数文件	提示添加失败输入有误	同预期结果
5	删除选中数据	提示删除成功	同预期结果

### 6.2.3 预处理模块测试

该测试主要测试，用户在进行预处理操作时模块是否能够完成正常的数据处理，在预处理的三个流程中出现错误是否可以反馈相应的错误信息。

测试用例如表 6.3 所示。

**表 6.3 预处理模块测试用例表**

编号	输入	预期结果	实际结果
1	选择需要预处理的数据	显示选择数据预览	同预期结果
2	点击开始预处理	处理结束，提示操作成功	同预期结果
3	输入数据小于最小处理长度	预处理失败，提示长度过小	同预期结果
4	输入断层间隔过大数据	正常预处理，提示断层间隔过大	同预期结果
5	查看预处理结果	显示可交互图表	同预期结果

### 6.2.4 去噪模块测试

该测试主要测试，用户在进行去噪操作时模块是否能够完成正常的数据处理，用户是否能顺利选择去噪模式，是否可以自由指定去噪模型，当在线去噪模式时用户账号错误或连接失败时能否返回正确的提示，对于去噪后的数据能否正常显示，对去噪后的评价指标能否正常显示。

测试用例如表 6.4 所示。

表 6.4 去噪模块用例表

编号	输入	预期结果	实际结果
1	选择本地去噪模式	转到本地模型列表	同预期结果
2	选择在线去噪模式	拉取在线模型列表	同预期结果
3	选择在线模式时账户错误	操作失败，提示账户错误	同预期结果
4	选择在线模式时密码错误	操作失败，提示密码错误	同预期结果
5	在模型列表选择需要去噪的模型	将选择的模型加载到系统中，提示选择成功	同预期结果
6	选择需要去噪的数据	显示选择数据预览	同预期结果
7	点击开始去噪	处理结束，提示去噪成功，并显示去噪后数据和评价指标	同预期结果

### 6.2.5 模型训练模块测试

该测试主要测试，管理员用户在进行模型训练时模块是否能够完成正常的参数输入，数据集生成，模型训练参数设定，模型训练等操作，以及当输入的数据集参数有误时是否能提示相应的错误，训练失败时是否返回相应的失败信息。

测试用例如表 6.5 所示。

表 6.5 模型训练模块用例表

编号	输入	预期结果	实际结果
1	不输入数据集与训练参数，点击开始训练	采用默认参数生成数据集和训练，提示采用默认参数	同预期结果
2	输入正常数据集参数不输入训练参数，点击开始训练	默认训练参数训练，提示采用默认训练参数	同预期结果
3	不输入数据集参数输入正常训练参数，点击开始训练	采用默认数据集参数生成数据集训练，提示采用默认数据集参数	同预期结果
4	输入正常数据集与训练参数，点击开始训练	提示开始训练，训练完成后提示训练完成	同预期结果
5	训练结束后模型未收敛	模型训练失败，提示是否重设参数重新训练	同预期结果

### 6.2.6 用户管理模块测试

该测试主要测试管理员用户在进行用户账户的添加, 查询, 删除以及修改信息时, 系统能否做出正确的操作, 以及返回对应的信息。

测试用例如表 6.6 所示。

**表 6.6 用户管理模块用例表**

编号	输入	预期结果	实际结果
1	输入不存在的账户和密码, 点击添加	添加成功, 提示添加成功	同预期结果
2	输入已经存在的账户和密码, 点击添加	添加失败, 提示添加账户已存在	同预期结果
3	输入账户, 点击查询	返回查询结果	同预期结果
4	选择账户, 点击删除	删除对应账户, 提示删除成功	同预期结果
5	选择账户, 点击修改	弹出修改信息界面	同预期结果

## 6.3 测试结果

本章首先对系统所用的深度学习去噪算法进行测试设计并完全执行了相应测试, 测试结果显示算法的性能良好, 准确性高且响应时间短, 各指标均达到设计系统时设置的指标。其次对系统功能进行了测试, 测试结果显示系统运行状况良好, 各个功能均完整了设定的操作。各个模块之前的调用和跳转正常, 整个系统运行时响应速度快, 对与错误数据和操作有对应的提示。综上所述, 通过各种测试用例可知, 系统在保证算法正确性的同时运行平稳易用性好, 可以顺利完成实际应用场景下的各种工作。

## 第7章 总结与展望

### 7.1 总结

本文使用基于神经网络的深度学习去噪算法、MySQL 数据库、PyQt 框架、Python web 等技术，完成了一个基于深度学习的超导瞬变电磁去噪系统。通过分析野外测绘人员的实际工作环境和工作任务，明确了本系统需要满足的需求，基于此进行了系统的概要设计以及详细实现，并在之前章节对系统各个模块功能和实现方法进行详细介绍。本文不仅描述了本系统的实现过程，同时也提供了后续算法升级和系统维护的必要信息，为以后实现更完善的瞬变电磁去噪系统提供了指导方案。

以下总结本文主要工作：

1. 本文一开始阐述了实现一个针对超导瞬变电磁的深度学习算法的重要性以及开发一个可以实时去噪的瞬变电磁信号去噪系统对探测领域的实际意义。对各个时期的瞬变电磁去噪方法进行了研究和分析，对市面上有去噪功能的软件进行了调研，为之后研发适用于超导瞬变电磁的去噪算法以及开发相应的去噪系统提供了指导。

2. 通过对实际应用情景的分析，确定了本系统面向的用户人群，以及用户对需求的需求。

3. 确定需求之后，对系统完成这些需求所应有的功能模块进行了设计和实现，并且通过测试确定本系统可以满足用户的需求。

本文通过对最新的同类去噪算法进行研究针对本系统去噪数据进行改进，研发了性能和准确度良好，支持实时处理的超导瞬变电磁信号去噪算法，从而使一个实时去噪系统成为可能。基于此开发了一个支持灵活调用不同模型，支持生成新数据集和训练新模型，支持对系统所涉及到的各种数据进行直观的展示，拥有简单易用的模型与数据管理模块的超导瞬变电磁信号去噪系统。

野外测绘人员可以利用本系统完成测绘数据的实时去噪工作，避免了需要将所有数据测绘完成进行分析后才能判断数据是否有效的现状。测绘人员通过本系统在一条数据测量完成后就可以查看数据原始曲线，预处理后曲线以及去噪后曲线，根据这些数据结果测绘人员可以马上决定是否要重测数据，或对测量参数进行调整。这会节省野外测绘人员大量的经历和时间，避免大量的无效操作和无效数据，在实际工作中大大提高了测绘人员的工作效率，最后本系统还实现了新模型添加与训练的功能，野外测绘人员可以很方便的通过系统服务器端获取最新的模型，管理员也可以通过系统服务器后端根据实际的需求训练新的模型，这些功能使得本系统拥有更广泛的使用场景，进一步提示工作效率。



## 7.2 展望

本性实现的去噪模型虽然达到了预期目标但仍有很多不足之处，比如对个数据的支持不够灵活需要特定格式才可以运行，系统客户端不支持多进程的操作当用户电脑性能较弱或处理超大数据时系统可能会响应缓慢，对生成的数据集和模型没有自动清除的功能，在用户使用期间可能会有很多无用的数据集或模型需要手动删除。这些不足和问题会成为下一阶段完善系统的目标。

## 参 考 文 献

- [1] 谢立军. Low-Tc SQUID 瞬变电磁系统的大功率发射机关键技术研究[D]. 吉林大学, 2016.
- [2] 滕吉文. 创新第二深度空间资源等. 中国地球科学联合学术年会论文集（六）——专题 12: 华北克拉通演化、专题 13: 陆陆碰撞带深部结构和动力学意义[C]. 2017:1.
- [3] 张黎, 王飞. 瞬变电磁法反演成像算法研究[J]. 品牌与标准化, 2015.
- [4] CHEN K, PU X, REN Y, et al. Temdnet: A novel deep denoising network for transient electromagnetic signal with signal-to-image transformation[J]. IEEE Transactions on Geoscience and Remote Sensing, 2020, 60: 1-18.
- [5] LIM J S. Reservoir properties determination using fuzzy logic and neural networks from well data in offshore korea[J]. Journal of Petroleum Science and Engineering, 2005, 49(3-4): 182-192.
- [6] DEVRIES P M, VIÉGAS F, WATTENBERG M, et al. Deep learning of aftershock patterns following large earthquakes[J]. Nature, 2018, 560(7720): 632-634.
- [7] XUE G, LI H, HE Y, et al. Development of the inversion method for transient electromagnetic data[J]. IEEE Access, 2020, 8: 146172-146181.
- [8] 刘家富, 雷宛, 张扬, 等. 瞬变电磁信号去噪方法对比分析与评价[J]. 工程勘察, 2014(11): 88-93.
- [9] 李貅, 宋建平, 马宇, 等. 基于小波分析的 TEM 信号提取[J]. 煤田地质与勘探, 2005, 33 (2): 72-75.
- [10] JI Y, LI D, YU M, et al. A de-noising algorithm based on wavelet threshold-exponential adaptive window width-fitting for ground electrical source airborne transient electromagnetic signal[J]. Journal of Applied Geophysics, 2016, 128: 1-7.
- [11] JI Y, WU Q, WANG Y, et al. Noise reduction of grounded electrical source airborne transient electromagnetic data using an exponential fitting-adaptive kalman filter[J]. Exploration Geophysics, 2018, 49(3): 243-252.
- [12] LIN F, CHEN K, WANG X, et al. Denoising stacked autoencoders for transient electromagnetic signal denoising[J]. Nonlinear Processes in Geophysics, 2019, 26(1): 13-23.
- [13] WU S, HUANG Q, ZHAO L. De-noising of transient electromagnetic data based on the long short-term memory-autoencoder[J]. Geophysical Journal International, 2021, 224(1): 669-681.
- [14] WANG M, LIN F, CHEN K, et al. Tem-nlnet: A deep denoising network for transient electromagnetic signal with noise learning[J]. IEEE Transactions on Geoscience and Remote Sensing,

- 2022, 60: 1-14.
- [15] SUN Y, HUANG S, ZHANG Y, et al. De-noising of transient electromagnetic data based on the minimum noise fraction-deep neural network[J]. IEEE Geoscience and Remote Sensing Letters, 2022.
- [16] 赵军. Python 程序设计第一课[M]. 清华大学出版社, 2018.
- [17] CHUN W. Core python programming: Vol. 1[M]. Prentice Hall Professional, 2001.
- [18] ARI N, USTAZHANOV M. Matplotlib in python[C]//2014 11th International Conference on Electronics, Computer and Computation (ICECCO). IEEE, 2014: 1-6.
- [19] 张顺, 龚怡宏, 王进军. 深度卷积神经网络的发展及其在计算机视觉领域的应用[J]. 计算机学报, 2019, 042(003): 453-482.
- [20] 吴晓军, 孙维彤, 张玉梅, 等. 基于一维残差卷积神经网络的脑电信号去噪方法[Z]. 2019.
- [21] 刘建伟, 宋志妍. 循环神经网络研究综述[J]. 控制与决策, 2022, 37(11): 16.
- [22] 王君洪, 孙少明, 孙怡宁, 等. 基于改进小波去噪和长短时记忆网络的肌肉疲劳识别模型研究[J]. 生物医学工程学杂志, 2022, 39(3): 9.
- [23] ABADI M, BARHAM P, CHEN J, et al. Tensorflow: a system for large-scale machine learning.[C]//Osd: Vol. 16. Savannah, GA, USA, 2016: 265-283.
- [24] 尹宝才, 王文通, 王立春, 等. 深度学习研究综述[J]. 北京工业大学学报, 2015, 41(1): 48-59.
- [25] 肖文鹏. 用 PyQt 进行 Python 下的 GUI 开发[J]. 中文信息: 程序春秋, 2002(7): 73-75.
- [26] VINCENT W S. Django for beginners: Build websites with python and django[M]. Welcome-ToCode, 2021.
- [27] LETKOWSKI J. Doing database design with mysql[J]. Journal of Technology Research, 2015, 6: 1.
- [28] 王世阳, 毕祥军, 王平. 基于 LabVIEW 多通道应变采集系统设计[J]. 国外电子测量技术, 2017, 36(8): 83-87.
- [29] SCHOKKER D, ATHANASIADIS I N, VISSER B, et al. Using a data lake stack in animal sciences[Z]. 2019.
- [30] 毛鑫鑫, 毛立峰, 杨聪, 等. 半航空瞬变电磁数据小波降噪方法研究[J]. 地球物理学进展, 2021, 36(3): 1287-1296.
- [31] 刘伟. 瞬变电磁正演与反演方法进展及其应用[C]//第十三届全国数学地质与地学信息学术研讨会论文集. 2014.
- [32] 赵静. 数据库管理系统的现状及发展方向[J]. 电脑编程技巧与维护, 2009(12): 2.
- [33] 李刚. Python: 基于实例的 GUI(Tkinter) 编程[J]. 电脑编程技巧与维护, 2022(6): 7-9.
- [34] DABBAS E. Interactive dashboards and data apps with plotly and dash: Harness the power of a fully fledged frontend web framework in python—no javascript required[M]. Packt Publishing

Ltd, 2021.

- [35] 李家宏. GB/T25000.51-2016 标准检测方法的验证[J]. 中国认证认可, 2018(4): 4.
- [36] STATHAKI T. Image fusion: Algorithms and applications[M]. Image Fusion: Algorithms and Applications, 2008.

## 致 谢

随着毕业设计的完成我的研究生生活也接近尾声，在三年的研究与实习生活中我获得了很多老师和同学的帮助与关怀，在此，我要对在研究生生涯以及毕业设计撰写过程中帮助我一起成长和学习的老师和同学致以最诚挚的感谢。首先在研究学习期间，我有幸得到了两位老师的教导，他们是：我的导师，中国科大的余艳玮博士，和我的企业导师，中科院上海微系统所的荣亮亮博士。两位深厚的学术功底，严谨的工作态度和敏锐的科学洞察力使我受益良多，每当遇到问题时总能给予我关键的指导，衷心感谢他们这段时间给予我的悉心教导和热情帮助。

同时我要感谢我的同学对我的论文工作给予的帮助，感谢我的室友和同学在和我相处的时间里对我的提升，感谢我的家人对我一如既往的支持，这些都是我不断前行的动力，在此深表谢意。