



上海大学
SHANGHAI UNIVERSITY

本科毕业论文（设计）

UNDERGRADUATE THESIS (PROJECT)

题 目：基于多智能模型集成的
海洋气象预测方法的研究

学 院： 计算机工程与科学学院

专 业： 智能科学与技术

学 号： 20123093

学生姓名： 苏天尧

指导教师： 辛立明

起讫日期： 2024. 01. 15–2024. 05. 24



姓 名：苏天尧

学号：20123093

论文题目：基于多智能模型集成的海洋气象预测方法的研究

原创性声明

本人声明：所呈交的论文是本人在指导教师指导下进行的研究工作。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已发表或撰写过的研究成果。参与同一工作的其他同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

签 名：苏天尧 日期：2024.5.24

本论文使用授权说明

本人完全了解上海大学有关保留、使用学位论文的规定，即：学校有权保留论文及送交论文复印件，允许论文被查阅和借阅；学校可以公布论文的全部或部分内容。

（保密的论文在解密后应遵守此规定）

签 名：苏天尧 指导教师签名：辛立明 日期：2024.5.24

摘要

随着人民生活水平的不断提高，海洋安全保障在当下愈发重要。在全球气候日趋恶化的大背景下，大气变暖、海平面上升等危害已经引起了人们的广泛关注。这些危害对我国人们的基础生活保障、沿海经济发展以及二十一世纪海上丝绸之路的建设带来了严峻的挑战。因此，海洋气象预测方法的相关研究在当下日益关键。然而，在过去的研究当中，针对海洋气象预测的诸多研究大多依赖基于物理方程的数值型海洋气象学预测，这些方法无非都存在复杂度过高、预测效果不准确等问题。随着近些年来人工智能的发展，海洋气象预测迎来了一轮新的发展方向。

本文在此基础上，提出了一种基于多智能模型集成的气象预测框架，旨在对海雾这一海洋上最为广泛存在的潜在安全隐患做出预测。框架结构主要由三部分构成，海洋因子时序预测单元考虑拟合 24 小时海洋气象水文因子的相关变化，由 Informer、Temporal Fusion Transformer、DeepVAR、N-BEATS 四模型智能加权输出。地区交互单元使用 Graph WaveNet 提取时空间依赖关系，拟合因子输出的残值，这点往往在过去的预报研究中被忽略。最后，海雾分类单元提供最终的预测输出，文中采用 XGBoost 和 LightGBM 智能加权预测概率进而判断海雾的发生与否。

本文数据来源于 ERA5 太平洋水文气象因子数据以及海洋中心提供的能见度数据。能见度被抽象为海雾的严重情况，在能见度 94 与 95 之间划分海雾出现的分界线。选择露点温度、蒸发量、位势高度、平均海面气压、海表温、表面压力、降雨量、相对湿度、风速 U 向、风速 V 向这些因子作为海雾预测的基础。数据在经过标准化、异常值过滤、相关性检测等操作之后进入预报系统。

海雾分类单元的预测在真实数据的分类中取得了 0.90 的平均准确率，以及 0.91 的平均召回率，ROC 曲线覆盖面积达到了 0.98。基于多深度学习模型融合的时序预测单元的最终预测效果相较于单一模型的 MSE 损失降低了约 20%。地区交互单元的输出之后为时序单元的输出做出补足，加和之后的 MSE 损失降低至 0.0062，仅为时序输出单元 MSE 损失的大约 30%。最终的模型输出结果表明，24 小时海雾气象因子输出在随机选择的 2000 条数据测试中取得了大约 80% 的准确率以及 0.89 的 ROC 曲线覆盖面积，由真实数据输入的预测大约取得了 85% 的准确率以及 0.93 的 ROC 曲面覆盖面积。这说明框架中各个单元均表现良好，三方互相协调，构成了一个相对先进、创新的海雾预测框架。

关键词：气象预测；海雾预测；集成学习；深度学习

ABSTRACT

As the standard of living among the populace continues to rise, the imperative for marine safety has become increasingly significant. Against the backdrop of a deteriorating global climate, issues such as atmospheric warming and rising sea levels have garnered widespread attention. These threats pose severe challenges to the basic livelihood security of our citizens, the economic development of coastal regions, and the construction of the 21st Century Maritime Silk Road. Consequently, the study of marine meteorological forecasting methods has become increasingly crucial. Historically, many studies on marine meteorological forecasting have relied on numerical marine meteorology predictions based on physical equations, which often suffer from issues of excessive complexity and inaccurate forecasting results. With the recent advancements in artificial intelligence, marine meteorological forecasting is experiencing a new wave of developmental directions.

This paper proposes a novel weather forecasting framework based on the integration of multiple intelligence models, aimed specifically at predicting sea fog, one of the most common potential safety hazards on the sea. The framework is structured into three main components: the marine factor time series prediction unit, which fits the related changes of marine meteorological and hydrological factors over 24 hours, with an intelligent weighted output from models such as Informer, Temporal Fusion Transformer, DeepVAR, and N-BEATS. The regional interaction unit employs Graph WaveNet to extract spatiotemporal dependencies and fits the residuals of factor outputs, a detail often overlooked in past forecasting research. Finally, the sea fog classification unit provides the ultimate predictive output, using XGBoost and LightGBM to intelligently weigh prediction probabilities and determine the occurrence of sea fog.

Data for this study were sourced from ERA5 Pacific hydro-meteorological factors and visibility data provided by marine centers. Visibility is abstracted as the severity of sea fog, with the boundary between visibility levels 94 and 95 marking the occurrence of fog. Factors such as dew point temperature, evaporation, geopotential height, average sea surface pressure, sea surface temperature, surface pressure, rainfall, relative humidity, wind speed U direction, and wind speed V direction serve as the basis for predicting sea fog. After undergoing standardization, outlier filtering, and correlation testing, the data enter the forecasting system.

The sea fog classification unit achieved an average accuracy of 0.90 and an average recall rate of 0.91 in real data classification, with an ROC curve area of 0.98. The final prediction performance of the multi-deep learning model fusion time series prediction unit reduced the MSE loss by about 20% compared to a single model. Outputs from the regional interaction unit complement the outputs from the time series unit, reducing the cumulative MSE loss to 0.0062, approximately 30% of the time series output unit's MSE loss. Ultimately, the model outputs demonstrated about 80% accuracy and an ROC curve area of 0.89 in tests with 2000 randomly selected data samples, with predictions from real data inputs achieving about 85% accuracy and an ROC curve area of 0.93. This indicates that each unit of the framework performs well, coordinating effectively to form a relatively advanced and innovative sea fog prediction framework..

Keywords: Meteorological Forecasting; Sea Fog Prediction; Ensemble Learning; Deep Learning

目 录

摘 要	I
ABSTRACT	II
1 绪论	1
1.1 研究背景及意义	1
1.2 国内外研究概况	2
1.2.1 海洋海雾领域研究概况	2
1.2.2 基于机器学习的海雾预报的概况	3
1.2.3 深度学习研究技术概况	4
1.2.4 图神经网络研究技术概况	5
1.3 本文研究内容	6
1.4 本文组织结构	7
1.4.1 组织结构	7
1.4.2 技术路线	8
2 面向海雾预报的相关模型概述及数据集选择	9
2.1 海雾气象分类器概述	9
2.1.1 基于多海洋气象因子的海雾预报	9
2.1.2 基于 LightGBM 的海雾分类器	10
2.1.3 基于 XGBoost 的海雾分类器	11
2.1.4 多模型智能融合分类器	13
2.2 海洋气象因子时序预测	13
2.2.1 深度学习时序预测概述	14
2.2.2 基于 Informer 的时序预测模型	14
2.2.3 基于 DeepVAR 的时序预测模型	15
2.2.4 基于 Temporal Fusion Transformer 的时序预测模型	16
2.2.5 基于 N-BEATS 纯前馈网络的时序预测模型	18

2.3 地区海洋气象因子动态扩散拟合	19
2.3.1 消息传递网络中的信息传递	19
2.3.2 基于图卷积神经网络的消息传递网络	20
2.3.3 基于 Graph WaveNet 的单步残差因子拟合	21
2.4 ERA5 太平洋气象信息数据集概述	22
2.4.1 海洋气象因子选取	22
2.4.2 地区时空间建模	24
2.4.3 数据筛选与噪声滤除	25
3 基于多智能模型集成的预报架构构建	26
3.1 海雾气象预报问题分析	26
3.2 基于海雾气象因子的相关组件构建	26
3.2.1 海雾分类单元构建	26
3.2.2 气象因子时序预测单元构建	28
3.2.3 地区因子交互单元构建	29
3.3 基于多智能模型集成的海雾预报框架	30
3.3.1 多智能单元集成	30
3.3.2 数据特征选择与输入	32
3.3.3 模型优化方案	36
3.4 多模型集成框架的模型分析	38
3.4.1 海洋气象因子分析	38
3.4.2 模型智能调优	41
3.4.3 模型训练效率分析	41
4 基于智能集成框架的海雾预报结果分析	45
4.1 实验设置与数据调整	45
4.1.1 数据设置与训练设备	45
4.1.2 模型结构设置	46
4.1.3 学习率优化器设置	47
4.2 二分类器分类效果分析	48
4.2.1 XGBoost 分类效果	48
4.2.2 LightGBM 分类效果	49
4.2.3 联合分类效果	51

4.3 时序预测单元输出结果分析	52
4.3.1 时序预测模型输出结果.....	52
4.3.2 时序单元联合输出结果.....	58
4.4 地区因子交互单元输出结果分析	60
4.4.1 GWNet 输出结果	60
4.4.2 一阶差分最终输出结果.....	62
4.5 智能框架预报能力分析	63
4.5.1 海洋因子最终预测结果.....	63
4.5.2 海雾预报性能	65
结论	68
参考文献	69
附 录 A	71
致 谢	75

1 绪论

本节主要描述了海洋气象预测方法的起源、背景和重要性，并探讨了该领域在国内外的研究进展。基于此，本文进一步明确了研究的具体内容和目标。

1.1 研究背景及意义

随着人民生活水平的不断提升，海洋安全的重要性越发凸显。在航海安全、海洋资源利用以及渔业管理等方面，海洋气象预测扮演着关键角色。然而，海洋气象系统的复杂和不可预测性为准确预测带来了不小的挑战。传统的单一预测模型很难完整地掌握海洋气象变动的复杂性，这常常导致预测结果的误差与不确定性。在我国快速发展的大背景下，开展基于多智能模型集成的海洋气象预测研究显得非常迫切和重要。

海雾，作为航海中常见的一种气象现象，至今为止已经对航海安全构成了极大的威胁。海雾的本质是一种主要由海洋表面水蒸气凝结所形成的低空云，常见于海洋及其邻近地区。这种气象现象主要由于海表较高的水温与上覆较冷空气的接触而触发，当空气温度降至露点以下时，海面上的水蒸气凝结成微小水滴，进而形成海雾。该现象在春末至夏初，以及海洋和陆地温差显著的地区尤为常见。

在学术研究和应用背景中，海雾的预测和管理是海洋气象学和航海安全领域的核心议题。全球化贸易的扩展以及海上航行活动的增加，使得精确预测海雾的生成、发展和消散过程显得尤为重要。海雾不仅严重影响海上和沿海地区的能见度，增加航海和港口作业的风险，还可能对海洋生态系统产生微妙影响，比如通过改变光照条件，间接影响海洋初级生产力。

当前，海雾的研究方法多样，包括但不限于卫星遥感技术、地面气象观测站的数据分析，以及海洋浮标提供的实时环境数据。此外，随着计算技术的进步，数值天气预报模型已成为预测海雾的重要工具。这些模型能够模拟大气中的物理和化学过程，为预测海雾的时空分布提供理论支持。近年来，人工智能技术，特别是机器学习和深度学习技术的引入，已经开始在海雾预测领域显示

出其强大的数据处理和模式识别能力。

实施基于多智能模型集成的海洋气象预测方法研究，不仅具有实际的应用价值，也具备深远的科学意义。这种方法能够提升预测的精确度和鲁棒性，并为决策提供坚实的数据支持。本文通过实践探索解决海洋气象预测中普遍存在的问题，并对国内在该领域的典型智能预测经验及模式进行了归纳和整理，为业界同行研究相应问题及推动相关工作提供了有益的参考。

1.2 国内外研究概况

作为本课题所选择的特定研究对象，海雾预报在过去数十年已经取得了显著的研究进展并达到了较为成熟的阶段。尽管如此，海雾预测的准确性仍受限于气象条件的固有不确定性，这构成了该领域研究的主要限制因素。因此，本节首先回顾了基于传统气象学和人工智能技术在国内外气候预报领域的研究现状，随后将详细讨论本研究采用的多模型集成与深度学习方法的具体应用和研究动态。

1.2.1 海洋海雾领域研究概况

海雾是一种主要发生在海上或沿海地区的天气现象，由于海洋的特殊影响，低层大气中的水雾或冰晶凝结，导致水平能见度降低至 1 公里以下。这种现象对海上运输、渔业、沿岸航空和道路交通等具有显著的负面影响，是中国近海地区一种需要重点关注的灾害性天气。

海雾的形成与多种海洋和大气条件紧密相关。具体来说，海雾的类型主要包括平流雾、混合雾、辐射雾和地形雾，其生成与风向、风速、降水、蒸发、气温、湿度、水温、海流和大气稳定度等因素密切相关。特别是在冷暖海流交界区和涌升的冷流区，海雾的发生尤为频繁，这些区域通常表现出较低的海面温度，这是海雾形成的关键条件之一。同时，稳定的空气层结也有利于海雾的生成和维持。

在过去几十年的海雾的预测方法方面，主要采用的有天气学方法、统计方法和数值预报方法。天气学方法依赖于天气学基本原理和分析方法，强调预报员的经验和对天气系统与海雾发生间关系的理解，但这种方法的准确率和可靠

性相对较低。统计方法则通过对历史数据的分析，建立海雾与相关预测因子之间的统计模型，尽管这种方法能提供一定的预测能力，但它缺乏对海雾物理过程的深入描述，其效果很大程度上依赖于历史观测数据的质量和数量。

随着计算机技术的发展和对海雾宏观及微观结构及其发生和发展规律的深入研究，数值模式已逐渐成为海雾预报中的重要工具。这种方法利用先进的计算模型来模拟和预测海雾的生成和演变过程，提供了一种更为精确和科学的预测方式。

1.2.2 基于机器学习的海雾预报的概况

虽然在此领域取得了一定的进展，但国内外依靠数值预报模型的气候预测仍存在普遍性误差，特别是在极端气候事件的预测上，这些误差尤为突出。以海洋中厄尔尼诺现象的热力温度预测为例，现有数值预报模型难以达到高精度的预测效果。这些限制主要由于数值模型对气候现象的先验理解需求及其数值表达的不足所致。近些年，随着人工智能的发展，这些问题诞生了新的研究方向。当前基于机器学习的海雾预测技术研究的主要进展可以简要描述如下：

（1）线性回归方法

线性回归是统计学中用于预测和数据关系建模的一种方法，特别是用于分析一个或多个自变量与因变量之间的关系。线性回归模型假设因变量与一个或多个自变量之间存在线性关系。

线性回归模型的优势在于模型简单、易于理解和实现；计算需求不高，适用于解释变量与因变量的线性关系。然而对于某些复杂的非线性模式，模型可能很难很好的捕捉数据中的相关趋势。并且线性回归模型对异常值敏感，容易受到极端值的影响。

王厚广[3]和其团队的研究主要聚焦于使用统计分析方法来预测海雾。他们选取了五个关键预测因子，并应用二级判别分析来构建线性回归预测模型，主要针对接下来 24 小时内的海雾情况。该模型在一系列实验中表现出了 82% 的准确率。然而，王的方法主要考虑了气象站点间的气温和气压差异，并未涉及海洋条件，且应用范围局限于单一站点。

于润玲[1]团队则采用了模型输出统计 MOS (Model Output Statistics) 方法，结合多元回归分析，对黄海和东海的海雾进行预测。他们使用的预测因子包括

风向、风速、相对湿度和气海温差，这些因子都是通过综合分析沿海和海岛气象站的数据获得的。这种方法不仅考虑了气象条件，还引入了海洋数据，使得其线性预测模型更为全面和精确。

（2）决策树模型预报方法

决策树模型是一种广泛应用于数据挖掘和机器学习领域的预测和分类方法。决策树通过递归地将数据集分割成越来越小的子集来构建树状结构。每个节点代表一个决策点，这些决策点基于某个属性的值将数据分割成两个或更多的子集。这些属性通常是最能够帮助区分不同类别的特征。树的每个分支代表决策规则的输出，而每个叶节点（树的末端节点）代表一个预测结果。构建决策树的标准主要有信息增益（Information Gain）与基尼不纯度（Gini Impurity）两种。

黄健等人利用汕头、珠海、湛江地面观测站的历史数据和 NCEP 再分析资料，通过分类与回归树（CART）方法建立了一个海雾预报的决策树模型。这个模型使用了 24 小时前的海洋和气象数据作为输入，通过剪枝优化决策树，减少误分率，最终形成了一个简洁且实用的预测流程。该决策树模型在三个站点的 24 小时海雾预测准确性验证中取得了较高的成功率，分别为 83.7%、73.7% 和 82.4%。该模型强调了水汽条件和海面冷却作用的重要性，其清晰的逻辑关系可以有效应用于实际的业务预报中。

1.2.3 深度学习研究技术概况

在深度学习领域，气象预报常被视作一种预测问题，核心挑战在于处理复杂的时间序列数据及其空间依赖性。时间序列分析强调对过去气象数据的连续性和顺序性进行理解，以预测未来的状态。空间依赖性则涉及到气象条件如何在不同地理位置间相互影响。深度学习模型，尤其是 RNN 及其变种如 LSTM，因其能够处理序列数据中的长期依赖而被广泛应用。此外，通过引入注意力机制和图神经网络，深度学习技术能够更加精细地捕捉和分析时间和空间数据的复杂相互作用，从而提高模型在气象预测中的准确性和效率。具体来说，目前研究较为深入的模型主要如下：

（1）RNN (Recurrent Neural Networks)

循环神经网络（RNN）是早期为解决时间序列数据挑战而设计的模型，由 Jeffrey Elman 于 1990 年首次提出，开发了一种全连接的 RNN 架构。RNN 主要

用于序列数据处理，包括在自然语言处理和计算机视觉等应用中。然而，RNN 存在显著的限制，如梯度消失和爆炸问题，这些问题阻碍了其在长序列中维持长期依赖的能力。这一弱点降低了其在复杂序列任务中的效果，例如在气象预报中的应用。

（2）LSTM (Long Short-Term Memory)

在识别出 RNN 的限制后，长短期记忆网络（LSTM）被开发为 RNN 基本架构的增强版。1997 年，Jurgen Schmidhuber 及其同僚引入了 LSTM[15]，通过加入能够更有效管理长期依赖的记忆块来解决 RNN 中的梯度问题。这些记忆块包含三种类型的门：输入门、遗忘门和输出门。输入门调节数据的流入，通过向量乘法过滤输入，而遗忘门管理历史数据的保留或丢弃。这种架构模仿了人类记忆过程，有选择性地保留有用信息并丢弃无关信息，从而克服了传统 RNN 的局限性。2015 年，香港科技大学的一个团队率先将卷积神经网络与 LSTM 结合，用于基于雷达数据的短期预测建模。这种方法在 2017 年进一步被完善，由史行健博士引入 TrajGRU 模型，通过学习时空递归连接，增强了 CNN-LSTM 架构的能力，以更好地捕捉快速变化的气象现象。

（3）Attention 机制

注意力机制[18]的整合已在使用 LSTM 结构的序列处理模型中变得普遍。注意力机制的核心思想是解决输入序列组件的统一重要性问题。它使模型能够关注序列中数据的不同重要性级别。通常在编码器-解码器架构中，编码器将所有输入序列转换为统一的语义表示，然后由解码器使用。这种方法可能导致过长的语义表示，从而阻碍模型性能。通过应用注意力机制，模型可以在解码过程中动态地聚焦于最相关的特征，从而提高总体模型效能。而在时序数据预测方面，Attention 机制的应用也相当成熟。

1.2.4 图神经网络研究技术概况

图神经网络，自 2005 年由 Gori 等人首次提出以来，已经经历了多年的研究与发展，但其广泛应用，特别是在处理包含空间信息的图问题上，是近几年的事情。目前，图神经网络主要分为五大类：图卷积网络、图注意力网络、图自编码器、图生成网络和图时空网络：

（1）图卷积网络（GCN）

图卷积网络通过在图结构上进行卷积操作来学习节点的特征表示，使其能够捕捉到节点间的局部连接模式。这类网络非常适合处理节点分类和图分类问题，

通过利用邻近节点的特征信息，可以有效地提取和利用图的结构性质。

（2）图注意力网络（GAT）

图注意力网络引入了注意力机制来确定节点间的连接强度。这种机制允许模型在聚合邻接节点的特征时，动态地给予不同邻居不同的重要性权重，从而提高了模型对图结构的适应能力和性能。

（3）图自编码器（GAE）

图自编码器用于无监督学习，通过编码器将图的节点编码为连续的向量，然后通过解码器重构图的结构，通常用于图的节点表示学习、图重构或是链接预测等任务。

（4）图生成网络（GGN）

图生成网络旨在生成图结构，这类网络可以从已有的图数据中学习到生成图的分布，并能生成新的图结构。这对于药物设计、蛋白质结构预测等领域尤为重要，因其需要模拟和预测复杂的图形结构。

（5）图时空网络（GSTN）

图时空网络是一种专门设计来捕捉图中时间和空间信息的图神经网络。在此类网络中，图的结构通常通过矩阵表示，矩阵中的每个元素会随时间序列发生变化。这种类型的网络尤其适用于像交通流量测量这样的任务。在交通流量测量中，速度传感器可以看作是图矩阵的元素，其数值随时间变化。因此，这类网络经常用于预测图中元素的未来值。例如，本文中使用的 Graph WaveNet 就是一种基于图卷积网络（GCN）的图时空网络。

1.3 本文研究内容

本文将基于 ERA5 海洋气象数据，使用一个多模态深度学习集成框架对洋面 24 小时之后的海雾发生情况做出预测，该框架将由时序预测器与海雾分类器两部分组成。本文将致力于探究基于多智能模型集成框架在海雾预测领域的可能性，力求提高深度学习技术在捕捉地区时空气象特征的相关能力。

本文的具体研究内容如下：

（1）ERA5 地区海域气象数据与可见度数据建模。

- (2) 基于融合决策树算法（包含 LightGBM 和 XGBoost 两种算法）的联合海雾分类预测。
- (3) 基于深度学习的海洋气象时序数据预测（包含 DeepVAR, Informer, Temporal Fusion Transformer 和 N-BEATS 四种模型的智能集成）。
- (4) 基于图神经网络 Graph WaveNet 的地区时空依赖提取。
- (5) 不同时间窗口的单点预报效果。
- (6) 不同特征选择情况下的海雾分类效果。
- (7) 海洋气象问题中不同因素对预报效果的影响。

1.4 本文组织结构

本小节主要对本文的组织结构及技术路线进行概述。

1.4.1 组织结构

本文共五章。

第一章主要介绍了课题的研究背景与意义、国内外研究概况、研究内容以及本文组织结构与技术路线。

第二章主要介绍了本文所使用的四种时序预测模型，一种图神经网络模型以及两种决策树海雾分类模型。同时介绍了具体的 ERA5 海洋气象数据的提取与建模过程。

第三章主要探讨多智能模型集成框架的实现，包括如何构建具体的时序预测模块和海雾分类模块，其中包含对若干深度学习模型训练过程及训练效果的分析、对模型调优方法的分析、以及海雾预报特征选择上的深入研究。

第四章基于前文的实验结果，继续讨论在不同时序维度上的时序预测效果，以及特征选择对时序预测单元产生的影响。同时，该章节会着重分析考虑空间因素对模型整体预测效果的影响。

第五章对全文进行了总结，归纳了本文的主要工作与创新点，并指出了进一步需要探究的问题。

1.4.2 技术路线

本文的技术路线如下图所示：

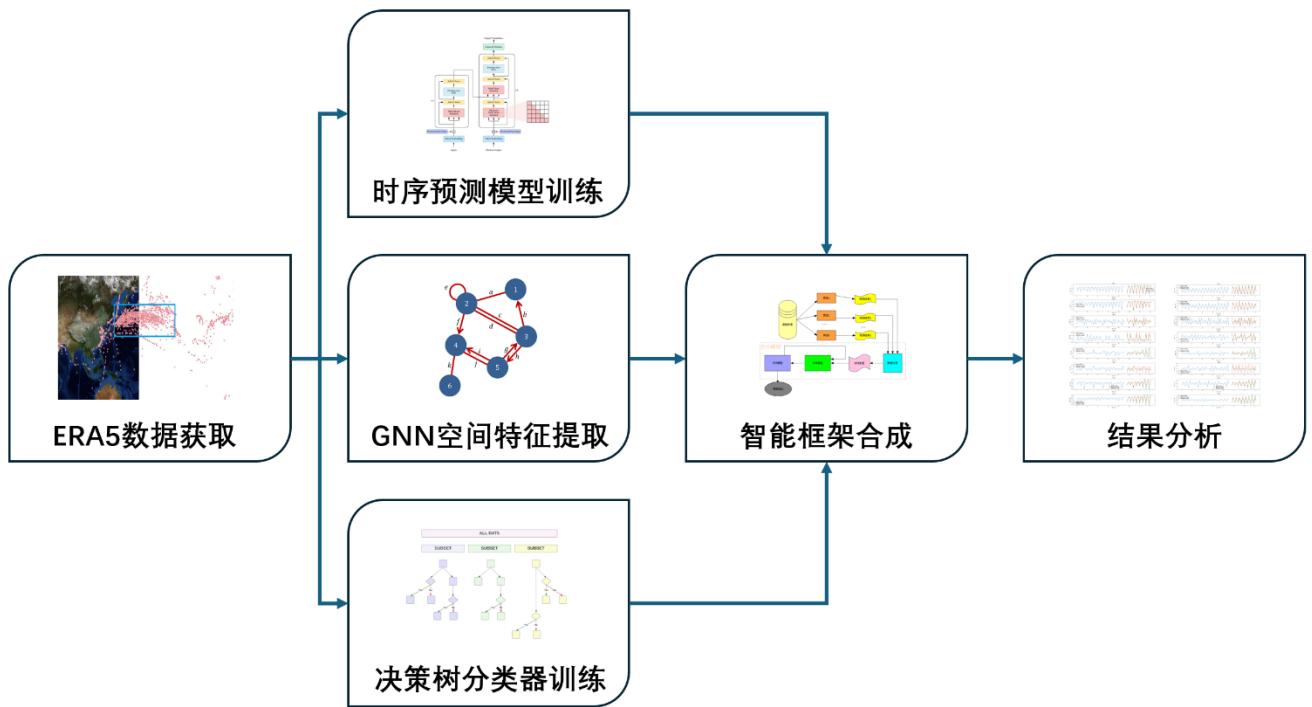


图 1.1 技术路线图

2 面向海雾预报的相关模型概述及数据集选择

本章具体说明了本文中所涉及的所有相关模型的整体结构、所使用的 ERA5 数据集，以及对数据的详细建模过程。

2.1 海雾气象分类器概述

本节将对海雾分类器的整体结构进行概述。

2.1.1 基于多海洋气象因子的海雾预报

近些年来，随着人工智能技术的深入研究，基于机器学习的海雾预报系统正在逐渐完善，其中以决策树为主的相关海雾预报方法在该领域内已经取得了不错的效果。高松等人着重分析了决策树，线性回归，SVM，K 近邻等多种机器学习算法在海雾预报方面的效果，发现决策树算法的效果最为出众，达到了 0.751 的准确率，远高于其他机器学习算法的预报效果。

对于决策树来说，预报因子的选取对决策树的整体效果能够起到至关重要的影响。与海雾发生条件有关的因素主要有：

（1）海温

海温是海雾形成中最关键的因素之一。当海表温度较低而空气较温暖时，温暖的空气接触冷海水面后，空气中的水汽会冷却至露点温度以下，导致水汽凝结成水滴，形成雾。这种现象尤其在春季常见，因为此时海水温度通常比大陆上的空气温度要低。不同的海域由于其地理位置、季节和当地气候的差异，海表温度也会有所不同，从而影响到雾的形成。

（2）露点温度和湿度

露点温度和相对湿度共同作用于海雾的形成过程。露点温度较高时，表明空气中含有较多的水蒸气；相对湿度接近 100% 时，空气接近或达到饱和状态。当海水温度较低，冷空气接触到较暖的海面，海面上方的空气温度可能迅速降

至露点以下，促使空气中的水蒸气凝结，形成海雾。。

（3）风速和风向

风的性质（包括风速和风向）是影响海雾分布和移动的另一个重要因素。一般来说，温和的风速有助于雾的形成，因为它可以帮助维持地面附近的湿润和冷却条件，而不至于将雾快速吹散。风向也同样重要，例如当沿岸地区的风从海上吹向陆地时，可能会携带湿润的空气形成雾。相反，如果风从陆地吹向海洋，可能会带走湿气，减少雾的形成。

（4）海洋和大气压力系统

大气压力系统对海雾的形成也有显著影响。高压系统通常伴随着较冷的空气和清晰的天气条件，这有助于夜间地面快速冷却，从而可能在清晨形成雾。在高压系统的影响下，空气下沉、稳定，这减少了大气的垂直混合，有利于雾的形成和维持。而低压系统通常带来较暖的空气和不稳定的天气条件，这可能会导致雾的消散。

2.1.2 基于 LightGBM 的海雾分类器

LightGBM（Light Gradient Boosting Machine）是由微软提供的一种高效的梯度提升框架，主要用于分类、回归和排序任务。它是基于决策树算法的集成学习技术，优化了传统梯度提升决策树（GBDT）的处理速度和内存使用，特别适合处理大规模数据。

与传统的基于 pre-sorted 算法（如 XGBoost）相比，LightGBM 将连续的特征值转换为离散的 bins，减少了内存消耗同时加快了计算速度。同时为了保持数据分布的一致性，LightGBM 保留梯度大（即更为重要）的数据样本，而对梯度小的数据样本进行随机采样。而在特征较多的情况下，算法会将互斥（即不太可能同时为非零值的特征）的特征进行捆绑，这减少了特征的维度，从而提高了计算效率。

LightGBM 最底层的数学原理为梯度提升策略。梯度提升是通过迭代的方式逐步添加弱学习器（通常是决策树），尝试纠正前一个模型的残差的方法。给定一个训练集 $\{(x_i, y_i)\}_{i=1}^n$ ，模型的目标是最小化以下损失函数：

$$L = \sum_{i=1}^n l(y_i, \hat{y}_i) \quad (2.1.1)$$

其中 l 是损失函数， y_i 是真实值， \hat{y}_i 是预测值。在每一次迭代中，通过添加一个新的决策树 f_t 来逼近负梯度 g_t ，即：

$$f_t = \arg \min_f \sum_{i=1}^n \left[g_{t,i} f(x_i) + \frac{1}{2} \lambda f(x_i)^2 \right] \quad (2.1.2)$$

这里 $g_{t,i}$ 是损失函数 L 关于 \hat{y}_i 的梯度， λ 是正则化项。

同时，LightGBM 融入了多种创新技术，如 Histogram-based 算法、GOSS 和 EFB，提高了处理大数据的速度和效率。这些技术使得 LightGBM 在处理大规模和高维度数据时，相对于其他传统的梯度提升方法，具有明显的优势。

在本文中，LightGBM 作为海雾预报单元的一部分，主要目标是在输入海洋气象因素之后，能够根据这些若干气象因素尽可能高的提高海雾预报的效果，包括尽可能高的精确率、召回率、F1-Score。

2.1.3 基于 XGBoost 的海雾分类器

XGBoost 是一种高效且灵活的梯度提升框架，由陈天奇等人开发。它是基于决策树的集成学习技术，用于分类、回归和排序问题。XGBoost 通过多种优化和正则化技术，提高了模型的预测性能和计算速度，尤其在处理大规模数据时表现出色。

XGBoost 在传统的梯度提升决策树（GBDT）的基础上引入了正则化项，以控制模型的复杂度，从而避免过拟合。并且，XGBoost 采用深度优先策略来进行树的生长，并通过 `max_depth` 参数限制树的深度。此外，它引入了剪枝机制，即使在树达到最大深度后也能从叶子节点回溯并删除不增加额外增益的分支。虽然树的构建本身是顺序的，XGBoost 优化了特征的排序和存储，使得在计算分割点时可以并行化处理。

XGBoost 模型是通过加法模型构建的，其中每一步都会添加一个新的树，来纠正前一轮预测的残差。模型可以表示为：

$$\hat{y}_i^{(t)} = \sum_{k=1}^t f_k(x_i), \quad f_k \in \mathcal{F} \quad (2.1.3)$$

其中， \mathcal{F} 是决策树的函数空间， f_k 是第 k 个树模型， $\hat{y}_i^{(t)}$ 是第 i 个样本在第 t 轮迭代的预测值。

每一轮添加新树时，目标是最小化以下目标函数：

$$\mathcal{L}^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \quad (2.1.4)$$

其中， l 是一个可微的凸损失函数，用于评价预测值 $\hat{y}_i^{(t-1)} + f_t(x_i)$ 和真实值 y_i 之间的误差； Ω 是正则化项，定义为：

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda |w|^2 \quad (2.1.5)$$

这里， T 是树的叶子节点数， w 是叶节点权重的向量， γ 和 λ 是正则化参数。

在每一轮迭代中，寻找最优的树结构 f_t 是通过贪心算法从顶到底逐步构建树。对于每个可能的分割点，计算分割带来的增益：

$$\text{Gain} = \frac{1}{2} \left[\frac{\left(\sum_{i \in I_L} g_i\right)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{\left(\sum_{i \in I_R} g_i\right)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{\left(\sum_{i \in I} g_i\right)^2}{\sum_{i \in I} h_i + \lambda} \right] - \gamma \quad (2.1.6)$$

其中， I 是分割前的节点， I_L 和 I_R 分别是分割后左右两个子节点的样本索引集合， g_i 和 h_i 分别是损失函数对应于 \hat{y}_i 的一阶和二阶导数。

同时，XGBoost 中还存在着一系列增强功能。比如，XGBoost 可以自动学习处理缺失值的策略。在训练时，XGBoost 将数据分为三部分：左子树、右子树和缺失值方向。和随机森林中的特征随机选择类似，XGBoost 中的列抽样可以提高模型的泛化能力，减少过拟合。同时，XGBoost 允许对每一步的更新进行缩放，并添加一个额外的权重参数，以进一步控制模型的更新速度。这些创新使得 XGBoost 在各类实际应用中广泛使用，且都取得了较好的效果。

在本文中，XGBoost 作为海雾预报单元的另一部分，同样要求在输入海洋气象因素之后，能够根据这些若干气象因素尽可能高的提高海雾预报的效果，包括尽可能高的精确率、召回率、F1-Score。

2.1.4 多模型智能融合分类器

在本文提出的海雾分类单元中，上文中涉及的 LightGBM 与 XGBoost 通过智能算法加权融合。具体来说，海雾预报任务被抽象为一个二分类任务，分别为：1-海雾发生，0-海雾不发生。通过多因子输入集成学习模型中，模型会输出当前特征所对应的两种类别的 Softmax 概率，这一概率可以认为是模型对类别的打分，而不同的两种集成学习模型对同一类别的打分情况会存在差异，一种可行的策略是，加权平均两模型的预测结果，让擅长某些样本的模型在这部分的权重更高，另一模型的权重降低，从而使得提高模型的整体预测效果。

本文主要考虑使用 PSO（粒子群扩散算法）作为主要的模型权重优化方案。

粒子群优化（Particle Swarm Optimization，简称 PSO）是一种群体智能优化算法，由 Eberhart 和 Kennedy 在 1995 年基于鸟群觅食行为的社会心理学模型提出。PSO 算法以其实现简单、参数少和收敛速度快等优点，在全球优化问题中得到了广泛的应用，如函数优化、神经网络训练、模糊系统控制等领域。

在 PSO 中，每个优化问题的潜在解都被看作是搜索空间中的一个“粒子”。每个粒子都有其位置和速度，这些位置和速度会根据个体和群体的经验来调整。位置表示在问题空间中的一个潜在解。速度决定了粒子位置更新的方向和幅度。

粒子根据自身的飞行经验和同伴的飞行经验来更新自己的速度和位置。每个粒子都保持着迄今为止找到的最好位置（个体最优解）的记忆，同时群体中的所有粒子共享群体中找到的最好位置（全局最优解）的信息。

2.2 海洋气象因子时序预测

对于某一特定气象因子，如海表温度(sst)而言，其在若干小时内的预测与变化存在较大的不确定性，因此本文通过将多个时序预测模型智能组合，实现了一个智能时序预测单元。本节中先对目前较为先进的时序预测模型进行概述。

2.2.1 深度学习时序预测概述

时序预测（Time Series Forecasting）是一种统计技术，用于基于历史数据预测未来的事件或趋势。这些数据是按时间顺序排列的，并且通常有着固定的时间间隔。时序预测在金融、经济、天气预测、资源管理等领域有着广泛的应用。

时序数据有着若干相当重要的特性。时序数据的平稳性意味着其统计特性（如均值、方差）在整个时间序列中是恒定的。非平稳时间序列预测通常需要先进行差分、对数转换或其他方法来稳定数据的均值和方差。此外，许多时序数据展示出某种形式的周期性变化，这被称为季节性。例如，零售行业的销量可能会在特定的节假日或季节显示出明显的高峰。识别并建模这种周期性是提高预测准确性的关键。而趋势是时间序列中长期增加或减少的方向。时序数据可能包含一个或多个趋势，这可能需要在预测模型中特别处理，以确保预测的准确性。

时序数据预测技术在本文中的主要应用在于对多个因子在 72 小时内的变化做出预测，力求预测 24 小时之后的海洋气候因子的变化情况。

随着技术的进步，深度学习已经成为处理和预测复杂时序数据的强大工具。本文中主要使用了四种相对先进的深度学习模型，分别是 Informer，DeepVAR，Temporal Fusion Transformer 以及 N-BEATS。这四种模型在业界已经被广泛使用并被证明效果良好。

2.2.2 基于 Informer 的时序预测模型

Informer 模型是一种用于处理长序列时间序列预测的 Transformer 模型，它在 2021 年由 Zhou 等人在论文《Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting》中提出。Informer 的主要贡献在于其高效处理长序列数据的能力，通过引入一种新的概率稀疏注意机制（ProbSparse Self-attention）来减少计算复杂度。

传统的 Transformer 模型通过全自注意力机制计算序列中每个元素对其它所

有元素的关注度，这在序列长度很长时会导致巨大的计算负担。Informer 的概率 ProbSparse 自注意力机制选择性地关注序列中的关键信息，显著降低计算复杂度。

稀疏性的引入： ProbSparse 自注意力通过评估每个 key 的影响力，选择最有可能与 query 有高度关联的 key 进行计算。这是通过对 key 向量评估其概率分布来实现的，选择概率最高的前 \sqrt{L} 个 keys 进行注意力计算：

$$s_k = \frac{\sum_{i=1}^L \exp\left(\frac{k \cdot q_i}{\sqrt{d_k}}\right)}{L} \quad (2.2.1)$$

这里， s_k 是计算得到的 key k 的得分， \exp 是指数函数，用于计算 softmax 中的指数项， k 和 q_i 分别是 key 和 query 向量， d_k 是一个归一化因子，用于控制 softmax 的平滑度。

由此计算出的得分中，选择概率最高的前 \sqrt{L} 个 keys 进行注意力计算：

$$\text{Attention}(Q, K', V') = \text{softmax}\left(\frac{QK'^T}{\sqrt{d_k}}\right)V' \quad (2.2.2)$$

这里，Attention 函数表示通过 softmax 函数计算得到的加权和，其中 Q 是 query 矩阵， K 是被选择的 key 子集， K' 是与 K 对应的 value 子集， d_k 同样作为归一化因子出现。

在本文中，Informer 采用自回归方式进行预测，即利用过去的预测结果作为新的输入来逐步生成未来的序列。模型考虑每个因子在时序上的自回归预测。

2.2.3 基于 DeepVAR 的时序预测模型

DeepVAR 是一种用于时间序列预测的深度学习模型，它结合了传统统计学中的向量自回归（VAR）模型和现代的深度学习技术。DeepVAR 通常用于同时预测多个时间序列变量，这些变量之间可能存在相互依赖关系。这种模型特别适用于处理高维数据和捕捉复杂的非线性关系。

DeepVAR 模型的核心思想是将传统的向量自回归（VAR）模型的线性形式扩展到可以包含深度学习结构的非线性形式。在传统的 VAR 模型中，每一个变量的未来值被建模为过去值的线性组合，同时考虑多个时间序列之间的交互作

用。

向量自回归（VAR）模型是一种用于描述多个时间序列和它们之间相互依赖性的统计模型。VAR 模型的基本数学表示如下：

$$Y_t = \nu + \Phi_1 Y_{t-1} + \Phi_2 Y_{t-2} + \cdots + \Phi_p Y_{t-p} + \epsilon_t \quad (2.2.3)$$

其中， Y_t 是一个 $n \times 1$ 的向量，代表在时间 t 的所有变量。 ν 是一个 $n \times 1$ 的常数向量。 $\Phi_1, \Phi_2, \dots, \Phi_p$ 是 $n \times 1$ 的系数矩阵，代表不同时间滞后的影响。 ϵ_t 是一个 $n \times 1$ 的误差向量，通常假设为均值为零且具有常数协方差矩阵的多元正态分布。

在这个模型中，每个变量的当前值被表示为其自身以及其他变量过去值的线性组合，加上一个误差项。系数矩阵 Φ_i 描述了相应滞后时期所有变量之间的相互关系。

DeepVAR 模型将传统的 VAR 模型与深度学习技术结合，使用深度神经网络来捕捉时间序列数据中的复杂非线性关系。其数学表示可以简化为：

$$Y_t = f(Y_{\{t-1\}}, Y_{\{t-2\}}, \dots, Y_{\{t-p\}}; \Theta) + \epsilon_t \quad (2.2.4)$$

其中， f 是一个深度神经网络，其参数为 Θ 。 ϵ_t 是模型误差，通常仍然假设为正态分布。

在这里， f 可以是一个包含多层 LSTM 或 GRU 的网络，这样的网络能够捕捉数据中的长期依赖性和非线性模式。这种表示方式表明，DeepVAR 通过神经网络来学习和近似传统 VAR 模型中的线性组合关系，但同时能够包含更复杂的动态关系。

2.2.4 基于 Temporal Fusion Transformer 的时序预测模型

Temporal Fusion Transformer (TFT) 由 Bryan Lim, Sercan O. Arik, Nicolas Loeff 和 Tomas Pfister 在 Google Cloud AI 团队提出[17]，是一种专用于时间序列预测的深度学习模型，它能够有效地处理多变量输入并应对时间序列数据中的复杂模式。TFT 的设计目的是为了整合和自动选择多种时间相关的特征，包括静态（时间不变）特征、已知的未来输入以及历史时间序列数据。

其核心组件包括多头自注意力机制（见 2.2.2 节），门控机制，时间卷积网络与变量选择网络等。

门控机制的设计主要是为了提高模型的灵活性和处理不同特征的能力，TFT 在其架构中引入了门控机制。这有助于模型学习在不同时间步骤中哪些信息是重要的。其表达式如下：

$$G_t = \sigma(W_g \cdot [X_t, H_{\{t-1\}}] + b_g) \quad (2.2.5)$$

这里， G_t 表示在时间步 t 的门控输出， σ 是 sigmoid 激活函数， W_g 是门控层的权重， $[X_t, H_{\{t-1\}}]$ 表示当前输入 X_t 和前一个时间步的隐藏状态 $H_{\{t-1\}}$ 的拼接， b_g 是门控层的偏置项。

时间卷积网络（TCN）是一种专为处理序列数据设计的神经网络架构。它采用一维卷积层，通过扩张卷积（dilated convolutions）和因果卷积（causal convolutions）来捕捉时间序列的长期依赖关系。扩张卷积允许网络以指数级增长的感受野来覆盖更大的输入范围，而因果卷积确保模型在预测当前值时只能使用过去的信息。

在 TCN 中，卷积操作可以表示为：

$$y(t) = \sum_{\{s=0\}}^{\{s-1\}f(s)} f(s) \cdot x(t - d \times s) \quad (2.2.6)$$

其中， $x(t)$ 是输入序列， $f(s)$ 是卷积核的权重， d 是扩张因子， s 是卷积核的大小， $y(t)$ 是输出序列。

变量选择网络是 TFT 中的一个关键组件，用于自动识别和选择对预测任务最有影响的输入特征。这种机制可以显著提高模型的解释性和性能，通过减少不必要的噪音输入，并专注于那些最具信息量的特征。

变量选择网络通常包括一个学习到的注意力机制，该机制对输入特征的每个维度分配一个权重，这些权重决定了相应特征的重要性。这可以通过一个软注意力机制实现，其中每个特征的权重是通过一个小型神经网络（如全连接层）计算得出的。

假设有 N 个输入特征，每个特征 i 的重要性权重 w_i 可以通过下面的公式计算：

$$w_i = \text{softmax}(W_s \cdot x_i + b_s) \quad (2.2.7)$$

其中， x_i 是特征 i 的输入， W_s 是学习到的权重矩阵， b_s 是偏置项，softmax 函数确保所有权重加起来等于 1，从而使得这些权重可以解释为特征的重要性概率。

2.2.5 基于 N-BEATS 纯前馈网络的时序预测模型

N-BEATS (Neural Basis Expansion Analysis for Time Series Forecasting) 是一种专为时间序列预测设计的深度学习模型，由 Oreshkin 等人在 2020 年提出。它是一个纯前馈神经网络，设计用来直接预测时间序列未来的值，而不依赖于历史输入时间序列的传统递归结构。

N-BEATS 模型的核心思想是基于将时间序列预测问题分解为多个小的、更容易学习的子任务。每个块 (block) 都学习时间序列的一个特定方面，并为未来的时间点提供预测，模型整体的预测是通过整合这些块的输出得到的。

N-BEATS 采用了一种非常规的前馈结构，通过多个堆叠的、完全连接的网络块来预测时间序列的未来值。每个块的结构一般都由输入层，多层感知机和输出层组成。每个块负责输出一部分预测和一组回归系数。每个块都接收整个输入时间序列，并输出对未来值的预测以及对输入时间序列的后向预测。

对于块 b ，其输出可以表达为两部分，其一为预测向量：

$$\hat{y}_b \in R^H \quad (2.2.8)$$

其中 H 是预测的未来步数。

另一部分为回归向量：

$$\hat{x}_b \in R^T \quad (2.2.9)$$

其中 T 是输入时间序列的长度。每个块的输出公式可以表示为：

$$\hat{y}_b = g_{\theta_b}^{\text{forecast}}(x) \quad (2.2.10)$$

这里 $g_{\theta_b}^{\text{forecast}}$ 是块 b 中用于预测未来值的神经网络部分，而 x 表示整个输入时间序列。

对于每个块 b ，其对历史输入数据的重建可以表示为：

$$\hat{x}_b = g_{\theta_b}^{\text{backcast}}(x) \quad (2.2.11)$$

这里 $g_{\theta_b}^{\text{backcast}}$ 是块 b 中用于产生历史时间序列重建的神经网络部分。

模型的整体输出，即所有块预测输出的累加，可以表示为：

$$\hat{y} = \sum_{b=1}^B \hat{y}_b \quad (2.2.12)$$

这里 B 是块的总数， \hat{y}_b 是第 b 个块的输出预测。

每个块通过其 \hat{y}_b 输出提供对输入时间序列的一种解释，即模型是如何理解和重建输入数据的。这种机制增加了模型的解释性，帮助用户理解每个块对预测结果的贡献。

2.3 地区海洋气象因子动态扩散拟合

对于特定地点而言，该地海雾的发生与当地海洋气象因子紧密相关。然而，在更广泛的地区范围内，各个地点的海洋气象因子不仅受到时间变化和地理特性的影响，湿度、气温等环境因素也可能在不同位置间传递和扩散，这种动态交互可能导致预测中出现误差。本节中将详细介绍专门针对海洋气象因子扩散问题所提出的解决策略。

2.3.1 消息传递网络中的信息传递

对于地区之间的若干特定地点，可以用图结构来详细描述这些点之间的交互关系。不同地点之间的气象因子在每个时刻互相交互，形成信息的流动。因此这张图又可以被建模成一个消息传递网络（Message Passing Networks, MPN）。

消息传递网络是一种框架，用于处理图结构数据，通过在图中节点之间迭代地传递信息来更新节点的状态。这种方法基于两个基本假设：节点的重要信息可以通过其相邻节点获得；节点的全局功能可以通过其局部邻居信息逐步构

建。

在消息传递网络中，图中的每个节点都会在多个迭代中与其邻居交换信息。这个过程通常包括三个步骤，即消息的生成，聚合和状态更新。在消息生成阶段，每个节点根据自身和邻居的特征以及边的属性生成消息。这一步骤可以用以下公式概括：

$$m_{\{u \rightarrow v\}} = f(h_u, h_v, e_{\{uv\}}) \quad (2.2.13)$$

其中， $m_{\{u \rightarrow v\}}$ 表示从节点 u 发送到节点 v 的消息。 h_u 和 h_v 分别是节点 u 和 v 的特征状态。 $e_{\{uv\}}$ 是连接 u 和 v 的边的属性。 f 是一个函数，用于从给定的输入生成消息。

之后，节点收集所有来自邻居的消息，并将其合并为一个单一的表示，这通常通过某种形式的聚合函数实现：

$$a_v = \text{Agg}(\{m_{u \rightarrow v} : u \in N(v)\}) \quad (2.2.14)$$

其中 Agg 是聚合函数，如求和、平均或最大值函数。当节点完成信息的聚合之后，节点会根据聚合后的信息和自身的当前状态更新其特征：

$$h'_v = g(h_v, a_v) \quad (2.2.15)$$

其中 h'_v 是更新后的节点状态。 g 是一个更新函数，可能包括非线性变换。通过这种迭代的消息传递过程，网络能够逐渐学习到每个节点在整个图中的角色和功能。

2.3.2 基于图卷积神经网络的消息传递网络

当地区数据被抽象成消息传递网络后，自然而然地借助图神经网络（GNN）进行处理成为了一种直观的方法（见 1.2.4）。然而，传统的 GNN 理论模型在处理大规模图数据时常常面临性能瓶颈，特别是在涉及到大量节点级别更新的场景中。因此，在实际应用中，图卷积神经网络（Graph Convolutional Networks, GCNs）由于其高效处理复杂图结构数据的能力，通常被更广泛地采用。

图卷积神经网络通过在图结构数据上扩展传统的卷积神经网络（CNNs）的

概念而发展起来的。在传统的 CNN 中，卷积运算依赖于欧几里得空间的固定邻域结构（例如，图像中的像素网格）。然而，在图结构数据中，每个节点可能有不同数量的邻居，且图的结构并不规则。因此，传统的卷积操作无法直接应用于图数据，对图数据的卷积操作应使用专用的图卷积技术。

图卷积的关键是如何定义图上的卷积操作。一种流行的方法是使用图的拉普拉斯矩阵来定义卷积。拉普拉斯矩阵是图结构的一种表示，定义为 $L = D - A$ ，其中 A 是图的邻接矩阵，而 D 是节点的度矩阵，其对角线元素 D_{ii} 是节点 i 的度。图卷积操作又可被细分为空间图卷积和谱图卷积，文中使用的 Graph WaveNet 结构就是一种基于空间图卷积技术的先进图卷积神经网络结构。具体来说，空间图卷积方法直接在图的节点上定义卷积操作，将每个节点的特征与其邻居的特征结合起来。一个简单的空间图卷积操作可以表示为：

$$H^{(l+1)} = \sigma\left(D^{-\frac{1}{2}}AD^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) \quad (2.2.16)$$

其中， $H^{(l)}$ 是第 l 层的节点特征矩阵， $W^{(l)}$ 是该层的权重矩阵， σ 是非线性激活函数，如 ReLU。这种方法直接在图的邻接结构上操作，使得卷积操作依赖于节点的局部邻域。

2.3.3 基于 Graph WaveNet 的单步残差因子拟合

本文中，专用于拟合地区地点之间海洋气象因子交互的模型为 Graph WaveNet 图卷积神经网络模型。Graph WaveNet 是一种用于时间序列预测的深度学习模型，它结合了图卷积网络（GCN）和门控卷积网络（WaveNet），特别适用于消息传递网络等类型的图结构数据的预测。

Graph WaveNet 的核心是将图卷积网络（GCN）与因果卷积网络（如 WaveNet）相结合。这种结合不仅可以捕捉时间序列数据的时间依赖性，还可以通过图结构理解空间依赖性。

图卷积网络已在上一节中具体介绍，WaveNet 则是一种自回归模型，使用因果卷积来捕捉序列数据中的时间依赖性。WaveNet 的核心是其因果卷积层，保证在预测当前值时只使用过去的信息。其卷积层可以表示为：

$$x_t = \sigma(W_{f,k} * x_{t-k} + b_f) \odot \tanh(W_{g,k} * x_{t-k} + b_g) \quad (2.2.17)$$

其中，* 表示卷积操作， \odot 表示元素乘法， σ 是 sigmoid 激活函数。 \tanh 是双曲正切激活函数。 $W_{f,k}$ 和 $W_{g,k}$ 是卷积核。 b_f 和 b_g 是偏置项。

在训练 Graph WaveNet 时，通常需要最小化预测值与真实值之间的差异。常用的损失函数是均方误差 (MSE)，表示为：

$$L = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (2.2.18)$$

其中 y_i 是真实值， \hat{y}_i 是模型的预测值， N 是样本数量。

为了拟合地区之间海洋气象因子信息传递所造成的残差，可以先获取时序预测器的输出结果，然后计算出与真实值之间的残差，将残差输入到 Graph WaveNet 中进行训练，输出的结果再与时序预测器加权相加，从而提高预测的精度。

2.4 ERA5 太平洋气象信息数据集概述

本文中使用的数据集是从 ERA5 上获取的太平洋区域气象信息数据。ERA5 是由欧洲中期天气预报中心 (ECMWF) 发布的气象数据集，提供了全球范围内的大气、陆地和海洋气象要素。要素中包括了海表温度，相对湿度等一系列海洋气象因子。同时，从海洋中心获取的数据补充了对应时间地区的可见度数据，可见度在 94 到 95 之间产生明显的区分，一般来说被认为是较为清晰的海雾出现分界线。

2.4.1 海洋气象因子选取

各类型的海雾在特定的海洋环境和天气条件下形成的条件不同。根据国内外海雾研究成果，气温、气温露点差、气海温差、相对湿度和风向风速等气象、水文因子对海雾形成的敏感性较高，接下来将具体介绍本文中使用的相关因子。

(1) 露点温度 (d2m)

露点温度是指空气必须冷却到达饱和并开始凝结形成露珠的温度。当露点温度接近或低于海表面温度时，海雾的形成更有可能发生，因为这增加了空

气湿度饱和的机会。

(2) 蒸发量 (e)

蒸发量是指单位时间内从地表或水体表面蒸发的水量。海上蒸发增加了近地面空气的湿度，有助于雾的形成，尤其是在气温较低时。

(3) 位势高度 (z)

位势高度是指在标准大气条件下，某一压力水平对应的地理高度。虽然位势高度本身不直接影响海雾的形成，但它可以用于确定大气的整体稳定性和垂直结构，这些因素都会间接影响雾的生成。

(4) 平均海面气压 (msl)

平均海面气压是指在海平面上的大气压力。高压通常与稳定的气候条件相关，这有助于雾的形成和维持。

(5) 海表温度 (sst)

海表温度是指海洋表面的温度。较暖的海水可以增加上方空气的湿度，当冷空气流经这些区域时，容易形成海雾。

(6) 表面压力 (sp)

表面压力是指地表的大气压力。低压区可能伴随着气流上升，这通常不利于雾的形成；而高压区则有助于空气下沉和雾的生成。

(7) 降雨量 (tp)

降雨量是指某一时间段内的总降水量。降雨后，地面湿润可以增加局部湿度，有时候这也可能促进雾的形成。

(8) 相对湿度 (r)

相对湿度是指空气中水蒸气含量与该温度下空气可能含有的最大水蒸气量的比例。高相对湿度是雾形成的关键条件，尤其是当相对湿度接近 100% 时。

(9) 风速 U 向 和 风速 V 向

这两个参数分别表示风在水平方向上的两个分量，U 向是东西方向的风速，V 向是南北方向的风速。适度的风速有助于雾的形成和维持，但过强的风则可能将雾层吹散。

2.4.2 地区时空间建模

真实的海雾发生时，往往影响的是一整片海域地区，地区间的各种气象水文因子相互干涉，形成最终的结果。通过 ERA5 和海洋中心（提供能见度）的数据划分出时间从 2010 年 1 月至 2022 年 4 月的数据进行匹配。

选取的地区范围为北纬 38° 至北纬 56°，东经 139° 至西经 22° 的太平洋区域。同时选择对应的可见度数据，仅选择可见度从 90 至 99 的数据，便于在划分海雾出现情况后保证 0-1 样本的均衡性。

此外，为了完成海雾未来预报工作，将数据选取的时间跨度倒推 72 小时，目的是通过前 48 小时的因子时间序列预测后至多 24 小时之后的因子变化情况。由此完成的样本数量为 41740 个，每个样本包含一个对应的可见度关系。

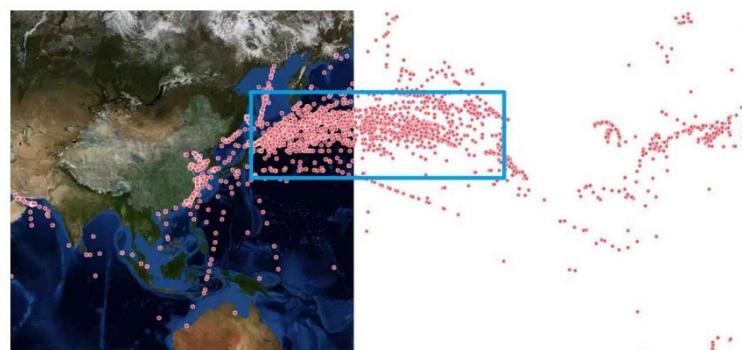


图 2.1 太平洋海域采样范围 红点为采集点

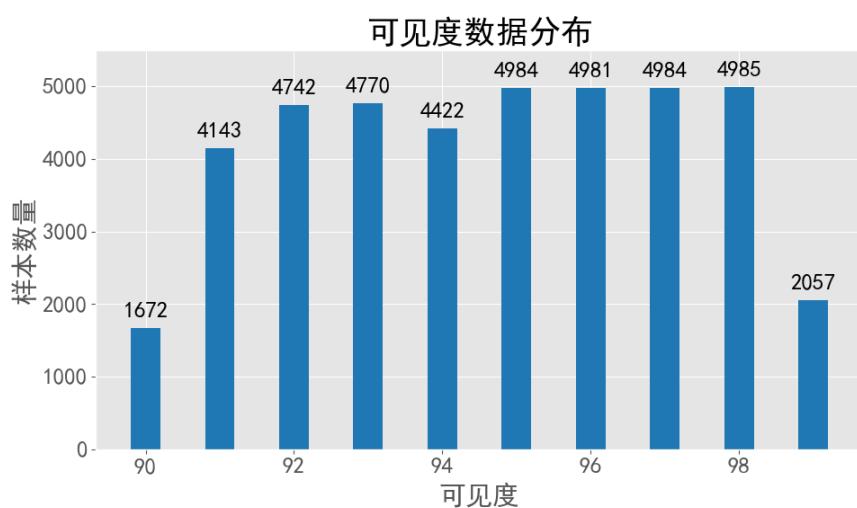


图 2.2 可见度样本数量及分布 基本符合正态分布

进一步地，为了考虑地区中多个样本点之间的消息传递关系，以每个样本点为中心，在其周围匹配 8 个点，相邻点之间的距离为 0.25km，由此构成一个 3×3 大小的点阵。可见度数据仅匹配最中心的原始点。预测的目标是尽可能通过其他 8 个点，提高中心点气象因子的预测精确度，从而提高总体海雾分类正确率。

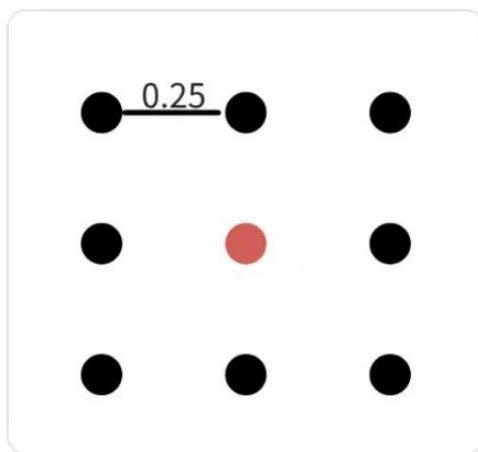


图 2.3 地区样本建模 3×3 地区间点

最终，样本的维度被扩充至 $(41740, 72, 9, 10)$ ，其中样本数量为41740个，每个样本对应72小时的时序，并且每个样本可以被理解为 9 个地区样本组成的点阵区域。每个样本点都对应了海表温度，相对湿度等 10 个特征。

2.4.3 数据筛选与噪声滤除

在实际的大气环境中，除了在 2.4.1 节中讨论的相关因素之外，可能还会存在类似洋流、季风等不确定因素，干扰实际的预测结果。因此在完成数据的获取之后，还应当对异常点作一定的过滤筛选处理。

因此可以从 ERA5 数据中去除位置比较偏的噪声点，对应数据所处时刻的极端天气等情况，完成数据的筛选与噪声滤除。

3 基于多智能模型集成的预报架构构建

前文中已经详尽分析了构建多智能模型集成框架所需的若干模型的实现原理，本章中将具体展开多个智能模型的集成过程、数据维度的筛选与输入，以及模型的调优方案与最终的效果分析。

3.1 海雾气象预报问题分析

在过去的若干年中，海洋气象系统与环球大气系统的复杂和不可预测性为海雾的准确预报带来了不小的挑战。传统的单一预测模型往往很难完整地掌握海洋气象变动的复杂性，这常常导致预测结果的误差与不确定性。

海雾可分为辐射雾、平流雾、地形雾和混合雾等多种类型，每种雾的形成条件都有着或多或少的差异性。此外，海雾形成的频率会随着季节的变化而发生差异。夏季普遍是海雾出现较为频繁的时段，然而为了考虑模型在全年时间跨度上的通用性，选取的数据并没有在时间线上做出筛选。

另一方面，海雾的出现与消散往往变化迅速，在实际情况下的数据采集对时间分辨率有着相当高的要求，这往往会成为海雾预测领域的瓶颈之一。

因此本章提出了一种多模态、多智能模型集成框架，改善了单一模型预测时可能出现的不确定性问题，提高了预报的鲁棒性与精确度。

3.2 基于海雾气象因子的相关组件构建

在第二章中已经详细阐述了框架中用到的所有模型及其实现原理，本节将在此基础上进一步将多个模型集成为相应的框架组件。

3.2.1 海雾分类单元构建

作为海雾预报系统的最终输出部分，海雾分类单元的效果将很大程度上决

定框架综合输出最后的效果上限。至今为止提出的海雾分类模型也几乎都是在这一部分的表现结果不佳。提高分类器的准确率意味着模型整体的输出效果也能进一步提升。因此分类单元的构建至关重要。

分类单元基于两种先进的基于梯度提升决策树（Gradient Boosting Decision Tree, GBDT）机器学习算法：LightGBM 和 XGBoost。这两种算法已在以往研究中展示出对复杂非线性数据模式的高效处理能力，并已经在前文中得到了详尽的评述与分析。

针对特定的海雾预测任务，本文从每个监测区域中心点收集的数据中提取出最后一小时的 10 个关键特征，形成一个具有 41740 个样本的数据集，每个样本由 10 个特征组成。这 10 个特征被用来执行海雾发生与否的二元分类。

在标签的生成上，本文通过将可见度数据进行简化分类处理，定义 90 至 94 的可见度值为类别 1（海雾存在），而 95 至 99 为类别 0（无海雾），将海雾预测任务转化为一个典型的二分类问题。模型的输出是基于这 10 个特征计算得到的类别概率，最终以概率较高的类别作为预测结果。

在模型的训练阶段，LightGBM 与 XGBoost 模型分别独立进行，以确保各自模型能够在其特定的参数设置与结构优化下达到最佳性能。训练完成后，两模型根据相同的输入数据分别对每个样本进行得分预测。随后，通过采用粒子群优化算法（PSO），对两个模型的输出结果进行智能加权组合，探索模型组合中各自输出的最优权重配置，以实现最终预测结果的最大化准确率。

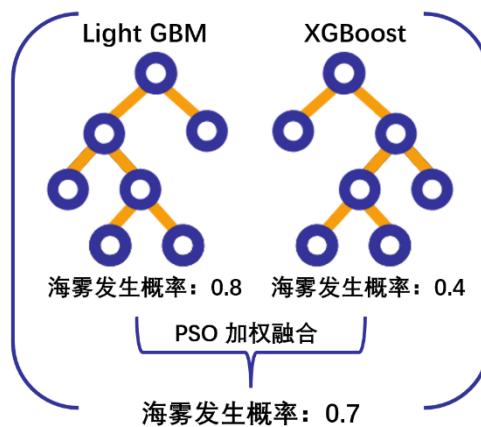


图 3.1 海雾分类单元构建

训练阶段使用使用真实数据中直接提取的数据，但在完成框架的整体构建之后，分类单元接收到的数据将是时序预测单元与交互单元共同给出的当前时

刻气象因子预测结果，分类器输出预测结果，并与海雾 Label 进行比对，从而计算综合预报结果。

3.2.2 气象因子时序预测单元构建

在节进一步提出了一个地区气象因子时序预测单元的设计与实现，该单元专注于在未来 72 小时内对关键气象因子的动态变化进行高精度预测。通过精确模拟这些气象因子的时间序列变化，可以显著提升海雾检测分类器的性能，从而在实际应用中达到更高的预测准确率和可靠性。

时序预测单元采用了四种先进的深度学习模型，具体为 Informer、Temporal Fusion Transformer (TFT)、DeepVAR 以及 N-BEATS。这些模型在处理时间序列预测问题，特别是在气象领域的应用中显示出了卓越的性能。四种模型各有所长，相互弥补。Informer 特别适用于处理长序列数据，能够通过其独特的稀疏自注意力机制，有效降低计算复杂度同时保持长期依赖的捕捉能力。TFT 通过融合历史数据和相关的外部变量来优化预测，其多头注意力机制使得模型能够在预测中灵活考虑时间依赖性和变量间的交互作用。DeepVAR 擅长捕捉和建模多变量时间序列数据的不确定性，在复杂的气象系统影响下仍旧能保持很好的鲁棒性。N-BEATS 能够通过堆叠的方式学习时间序列数据的多种趋势和季节性模式。

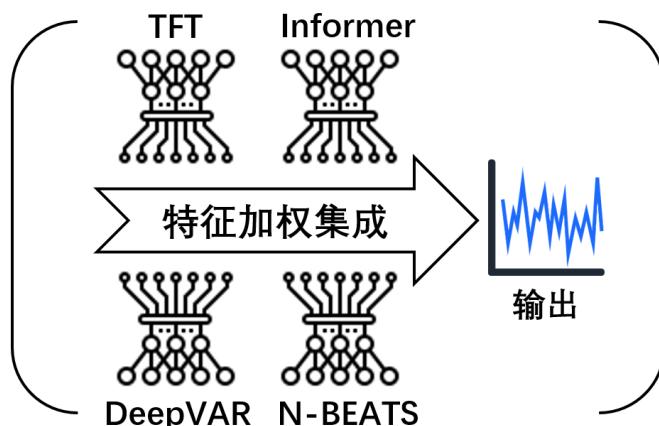


图 3.2 气象因子时序预测单元构建

同时，采用的 ERA5 海洋气象数据集包含了精细化的气象观测数据。对于时序预测单位，文本设置了一个 48 小时的历史数据窗口，目标是预测下一个 24 小时内的总共 10 个气象因子的具体变化。该数据集中每个监测点独立进行模型训练，以确保训练过程中数据的完整性和独立性。

在单独训练各个模型后，文中采用粒子群优化（PSO）算法对模型的输出进行智能加权融合。PSO 算法的角色是优化合成模型中各单一模型的权重分配，以达到整体最优的预测效果。这种加权方法允许系统根据实时数据动态调整各模型输出的影响力，从而更加精确地模拟未来气象条件的变化。

3.2.3 地区因子交互单元构建

本节将深入探讨文中框架的最后一个关键组件：地区因子交互单元的精细构建过程。前文已经详细介绍了时序预测单元的开发与实现，该单元在单点预测方面表现卓越。然而，其分析仍然局限于独立的地理位置，未能考虑区域间复杂的气候相互作用如大气环流和能量转移等，这些因素未被单点分析完全涵盖，可能导致预测精度受限。

针对此问题，本文提出一种创新的多点交互建模方法，通过构建地区内 9 个监测点的 3×3 点阵模型，实现监测点间在每个时间步的信息交互。这种设计能够捕捉由地区整体气候动态引起的数据波动，有效弥补单点模型无法捉摸的区域性特征。

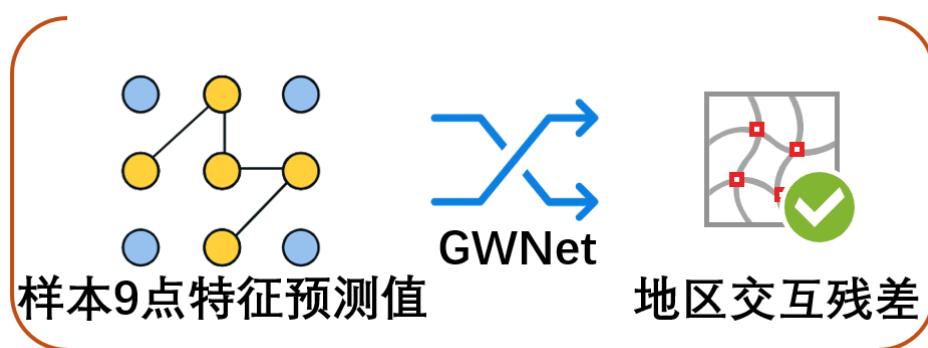


图 3.3 地区因子交互单元构建

在时序预测单元的输出完成后，通过计算模型输出与实际观察值的残差，这些残差数据将被用作地区因子交互单元的拟合标准。残差分析不仅揭示了单点预测的局限性，也提供了关键的反馈信息，用于调整和优化地区间交互模型。

地区因子交互单元采用了前文已经详细介绍过的一种先进图卷积神经网络 Graph WaveNet。该技术结合了图神经网络的空间依赖性捕捉能力与 WaveNet 的时间序列处理优势，使其特别适合处理气象数据中的空间时间依赖性问题。在本单元中，输入数据包括每个监测点的 10 个气象特征和一个静态邻接矩阵，后者定义了监测点之间的连接关系，以便模拟和预测这些点间的复杂相互作用。

通过对 Graph WaveNet 的训练，该网络预测的输出旨在尽可能匹配在后续时间步预测值与实际观测值之间的残差。这种方法不仅优化了信息流的捕捉，也为整体预测模型补充了在单一时序预测阶段可能遗漏的关键信息。这种策略显著提升了模型对区域内气象条件复杂动态的反应能力和预测精度。

通过将时序预测单元和地区因子交互单元的输出进行有效整合，本文实现了一个综合输出结果，具有更高的准确度和可靠性。这种多模型融合的方法不仅缓解了单点预测的局限，还增强了模型对区域气象条件变化的整体响应能力。

3.3 基于多智能模型集成的海雾预报框架

前文中已经详细阐述了多智能模型集成框架中多个组件的独立构成过程，本节将在此基础上进一步精细化组件集成中的相关问题以及优化方案，从而使各个模型之间的协调性进一步提高。

3.3.1 多智能单元集成

前文已经介绍，本文提出的智能集成架构主要由海雾分类单元、气象因子时序预测单元和地区因子交互单元三部分组成。这三个单元分别针对数据集中不同的数据部分进行专门的训练，以实现各自的功能，并最终协同工作，形成一个高度集成的预测系统，如图 3.4。

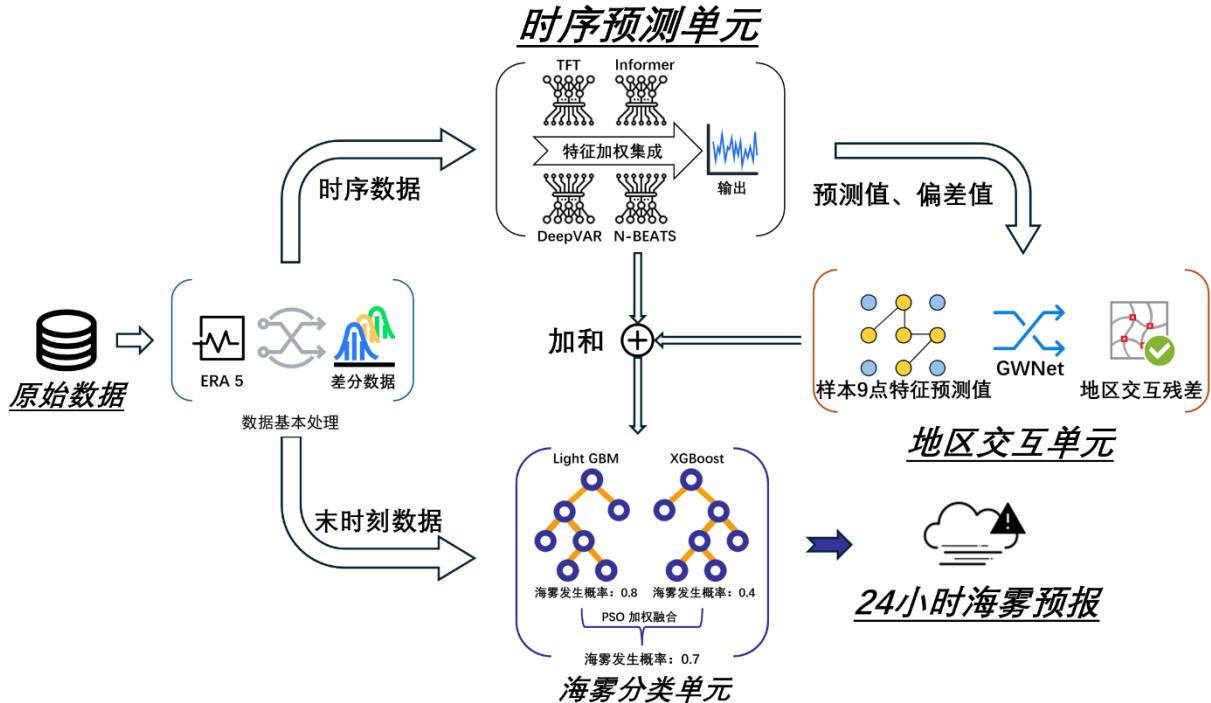


图 3.4 多智能模型集成架构

图中，初始数据首先经过一系列预处理步骤，包括数据标准化和特征选择，以确保输入数据的质量和相关性。数据标准化保证了模型训练的稳定性和有效性，而特征选择则侧重于剔除冗余特征，保留最具预测价值的变量，从而优化模型的性能和计算效率，最终将数据差分，保证数据的稳定性。

差分数据随后进入气象因子时序预测单元。该单元的主要任务是根据历史气象数据预测未来几小时的气象条件，其输出不仅为后续模块提供输入数据，也直接影响整个系统的预测精度。

随后，时序预测单元的输出数据（预测时序）连同其前一时刻的实际观测数据一起输入到地区因子交互单元。该单元会分析和调整时序预测中的空间相关性和地区间的动态交互。该过程从前一时刻的真实数据开始，逐步计算每一步的预测残差，并调整预测，直至达到时间窗口的末端。此单元的输出提供了校正后的时序数据，为最终的预测结果增添了准确性和可靠性。

经过地区因子交互单元处理后的最终时刻数据将被送入海雾分类单元，根据气象条件的变化预测海雾的发生。该分类单元的输出是关于未来几小时内海雾生成与否的预测，对于海上交通安全和相关气象预警系统具有重要意义。

3.3.2 数据特征选择与输入

前文中已经详细阐述了多智能模型集成架构的详细设计方案，本节将着重讨论数据选择部分的相关内容。

在前文的阐述中，文中选取了露点温度 $d2m$ 、蒸发量 e 、位势高度 z 、平均海面气压 msl 、海表温 sst 、表面压力 sp 、降雨量 tp 、相对湿度 r 、风速 U 向、风速 V 向这 10 个海洋气象因子和水文因子来作为判断海雾发生的参考变量。

在完成基本的数据处理之后（包括异常值、空值滤除）第一步处理是对数据作 Z 分数标准化：

$$z = \frac{x - \mu}{\sigma} \quad (3.3.1)$$

其中， x 为原始数据点， μ 为数据点的均值， σ 为特征的标准差， z 为标准化之后的数据点。Z 分数标准化的目的是将数据特征转换为具有零均值和单位方差的形式，因为原始的气象、水文因子的数量级之间存在较大的差异，使用 Z 标准化能够将所有特征规范至同一尺度，从而避免后续算法因数据尺度问题产生的偏差。同时，标准化减少了数据中异常值的影响，因为在计算均值和标准差时，异常值的效应被一定程度上削弱。标准化的结果可见表 3.1。

表 3.1：数据标准化前后结果对比

变量	均值	方差	均值(标准化)	方差(标准化)
露点温度 $d2m$	2.79e+02	6.72e+00	-2.60e-15	1.00
蒸发量 e	-5.11e-05	8.71e-05	1.25e-13	1.00
位势高度 z	2.43e+01	1.73e+02	-2.17e-12	1.00
平均海面气压 msl	1.01e+05	1.15e+03	2.06e-15	1.00
海表温 sst	1.04e+35	1.01e+36	4.01e-13	1.00
表面压力 sp	1.01e+05	1.18e+03	1.64e-15	1.00
降雨量 tp	1.42e-04	4.35e-04	-3.10e-13	1.00
相对湿度 r	8.47e+01	1.37e+01	-2.22e-15	1.00
风速 U 向	2.59e+00	7.17e+00	-5.37e-15	1.00
风速 V 向	1.34e+00	7.03e+00	5.01e-15	1.00

之后可以对各个特征之间的相关性作出基本的分析，查看变量中包含的信息是否存在一定的冗余，使用的方法是计算特征之间的协方差矩阵：

$$C_{ij} = \text{Cov}(X_i, X_j) = \frac{1}{n-1} \sum_{k=1}^n (X_{ki} - \bar{X}_i)(X_{kj} - \bar{X}_j) \quad (3.3.2)$$

其中， X_i 和 X_j 是变量， \bar{X}_i 和 \bar{X}_j 分别是这些变量的平均值。通过计算特征之间相关性可以识别并移除那些与其他变量高度相似或相关的变量，从而减少数据集中的冗余并提高模型的效率和解释性。另一方面，在回归分析等统计建模任务中，特征之间可能会伴随的强烈的多重共线性，即输入变量之间存在高度相关性，这会导致模型的参数估计不稳定，使得模型解释变得困难。通过移除相似或高度相关的变量，可以减轻多重共线性的问题。

这会导致模型的参数估计不稳定，使得模型解释变得困难。通过移除相似或高度相关的变量，可以减轻多重共线性的问题。协方差矩阵可由热力图直观展现，如图 3.5 所示。

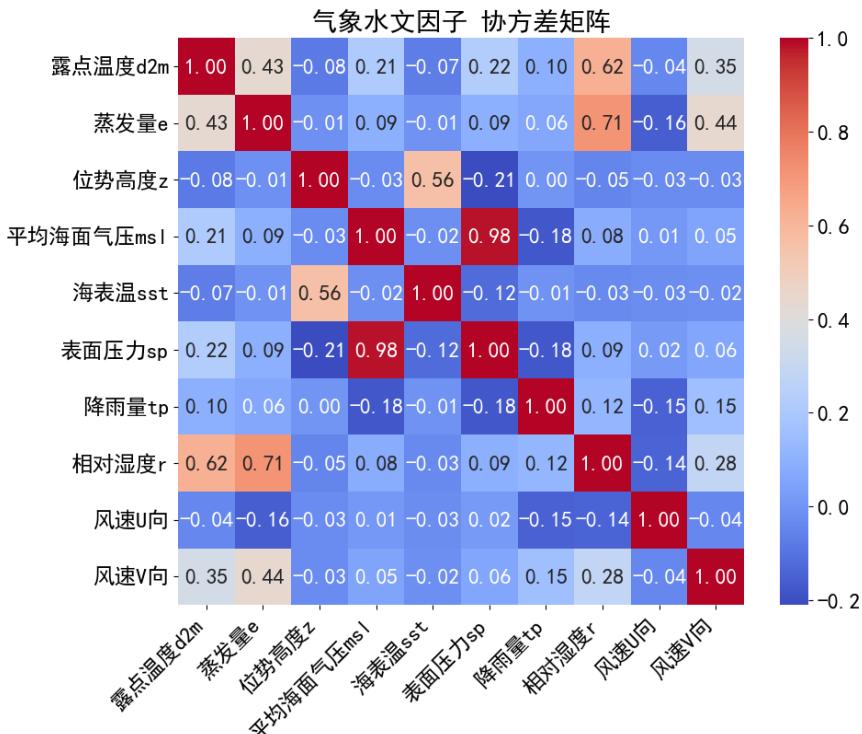


图 3.5 特征协方差热力图 暖色调相关性越高

图中分析可知表面压力与平均海面气压之间存在相当严重的共线关系，相对湿度与多个变量之间也存在一定的相关性。为了进一步判断变量之间是否存

在严重共线关系，可以再引入另一个统计指标，变量膨胀因子（Variance Inflation Factor, VIF）。

变量膨胀因子（VIF）是衡量多重共线性强度的一个统计指标。它提供了一个量化的数值，表明预测变量（自变量）之间有多少程度的线性关联。在回归分析中，多重共线性是一个潜在的问题，因为它可能导致回归模型的参数估计不准确或不稳定。VIF 的计算基于回归模型中各个自变量的 R^2 值：

$$VIF = \frac{1}{1 - R^2} \quad (3.3.3)$$

当 $VIF = 1$ 时，可以认为没有共线性。 $1 < VIF < 5$ 时，可以认为共线性可以接受。 $VIF \geq 5$ 时，认为共线性较强。 $VIF \geq 10$ 时，认为共线性极强。VIF 图可参考图 3.6。

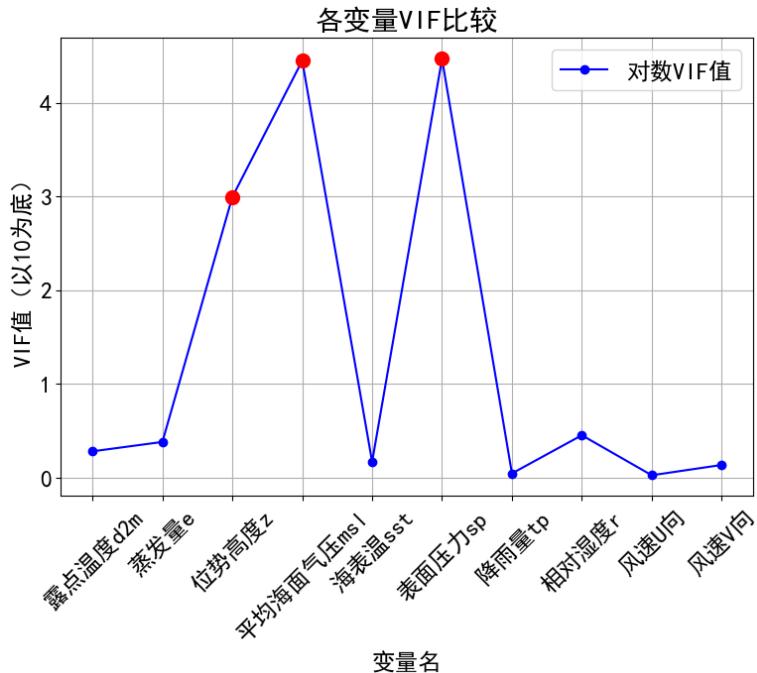


图 3.6 VIF 值折线图 红色点为异常点

图 3.6 考虑到可能出现的极强共线性关系，因此对每一个 VIF 值取以 10 为底的对数，并将 VIF 值大于 10 的值用红点标记。从图中可以看出，位势高度、平均海面气压、表面压力三个因子的共线性相当强，平均海面气压和表面压力的 VIF 值最高。因此先将表面压力这一特征滤除，重新输出 VIF 折线图，如图 3.7。

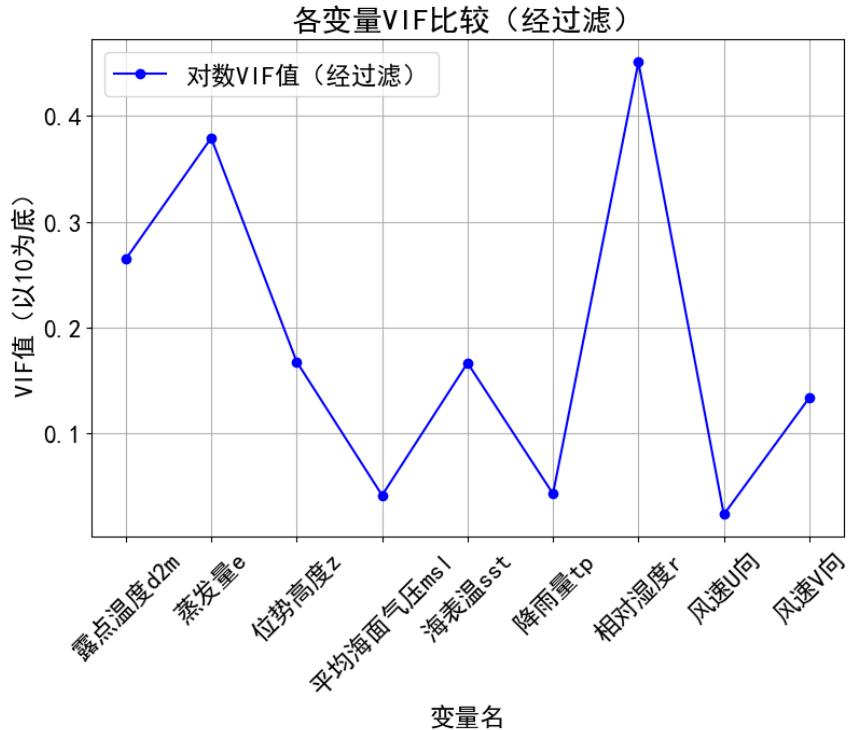


图 3.7 VIF 值折线图（滤）无异常点

从图中可以看出，所有变量的共线性 VIF 值都处在相对健康的位置，不需要进一步的特征筛选。因此，最终敲定参与模型训练的气象、水文因子总计有九个，分别是：露点温度 d2m、蒸发量 e、位势高度 z、平均海面气压 msl、海表温 sst、降雨量 tp、相对湿度 r、风速 U 向、风速 V 向。同时可以输出此时的热力协方图，如图 3.8。

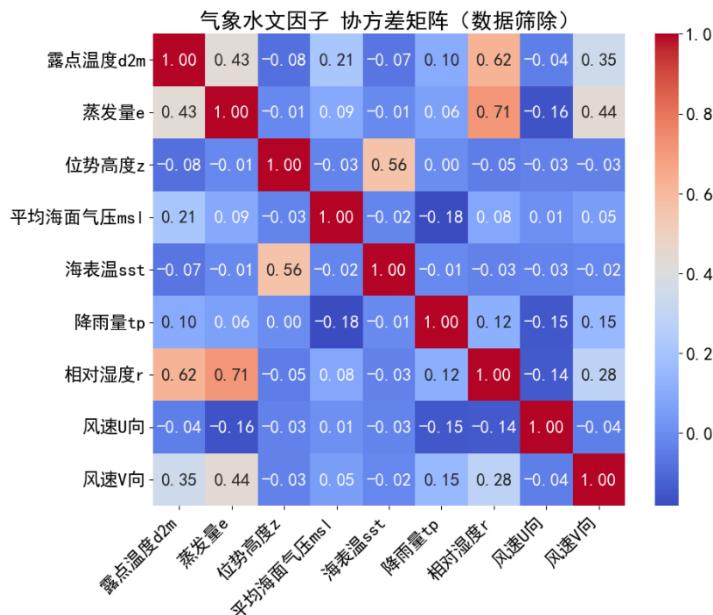


图 3.8 特征协方差热力图（滤）

3.3.3 模型优化方案

完成了数据处理操作之后，数据首先需要进入的单元就是气象因子时序预测单元。该组件中的四种深度学习模型对训练资源的要求都相对较高，并且由于网络较深，训练过程中可能会出现梯度爆炸、梯度消失等情况。因此需要针对这些问题考虑相应的解决措施。

（1）教师强制（Teacher Forcing）

教师强制（Teacher Forcing）是一种在训练序列生成模型时常用的技术，特别是在循环神经网络和其变体中。这种方法涉及在模型训练过程中使用真实的输出序列作为下一个时间步的输入，而不是使用模型自己生成的输出。

假设现有有一个任务，目标是生成一个序列 y_1, y_2, \dots, y_T 。在循环神经网络中，每一个时间步 t 的输出 \hat{y}_t 取决于当前的输入 x_t 和前一时间步的隐藏状态 h_{t-1} ：

$$h_t = f(x_t, h_{t-1}) \quad (3.3.4)$$

$$\hat{y}_t = g(h_t) \quad (3.3.5)$$

其中 f 是 RNN 的递归函数， g 是输出层的函数。

在教师强制训练中， x_t 是从真实的前一时间步的输出 y_{t-1} 而不是预测的输出 \hat{y}_{t-1} 得来的。初始输入 x_1 通常是一个特定的起始符号，如 $<start>$ 。这种方式确保训练数据的每个真实输出都被用作下一个时间步的输入，从而帮助模型学习如何预测序列中的下一个词。这种方法的主要优点是它可以加速收敛并提高模型的性能，因为它总是提供正确的历史信息给模型。

然而，在实际的预测阶段，模型通常无法访问到未来的真实输出。因此，模型必须依赖其自身之前的预测来生成新的输出。这意味着在实际使用中，模型可能表现不如训练阶段，因为它在训练阶段总是接收到正确的输入，而在预测阶段只能依赖于自身的可能不准确的输出。

在本文的框架构建过程中，四种时序预测模型都使用了教师强制机制来提高整体的训练效率，从而改善深度学习所需的计算资源匮乏问题。

（2）Kaiming 初始化

Kaiming 初始化，也被称为 He 初始化，是由 He 等人在 2015 年提出。这种初始化方法是为了解决在深层神经网络中，特别是使用 ReLU 激活函数时，权重初始化可能导致的梯度消失或梯度爆炸问题。

Kaiming 初始化的核心思想是在网络的前向传播和反向传播时，保持数据的方差不变，从而避免梯度消失或爆炸。具体来说，如果网络的每一层都尽可能保持输入信号的方差，那么信号可以更深入地传播到网络，而不会衰减或过度放大。

假设神经网络的每层都有 n 个输入，权重 w 从某种分布中随机初始化，那么为了使层的输出的方差等于输入的方差，权重 w 需要满足以下条件：

$$\text{Var}(w) = \frac{2}{n} \quad (3.3.6)$$

这里的 2 是针对 ReLU 激活函数特别选择的，因为 ReLU 函数 $f(x) = \max(0, x)$ 在输入为正时的梯度为 1，而在输入为负时的梯度为 0。因此，期望激活的一半会是 0（即期望输出的一半是未被激活的），这就要求在初始化时考虑这一点，以保持激活前后的方差一致。

对于非常深的网络，Kaiming 初始化尤其重要，因为错误的初始化方法会导致在多层传播过程中信息丢失。并且在本文的框架中，大部分的激活函数都用 ReLU 函数及其变体来实现，因此 Kaiming 初始化的应用对缓解训练中的梯度爆炸、消失问题来说至关重要。

（3）CosineAnnealingLR 方法

CosineAnnealingLR 是基于余弦退火调度（Cosine Annealing Schedule）的概念是由 Ilya Loshchilov 和 Frank Hutter 在 2016 年提出。它根据余弦函数周期性地调整学习率，帮助模型在训练过程中更好地找到全局最小值并避免陷入局部最小值。

CosineAnnealingLR 通过模拟退火的思想，使用一个类似于余弦波形的调度策略来调整学习率。具体而言，学习率随着每个 epoch 的变化而按照余弦函数进行周期性调整：

$$\text{lr}(t) = \text{lr}_{\min} + \frac{1}{2}(\text{lr}_{\max} - \text{lr}_{\min}) \left(1 + \cos \left(\frac{T_{\text{cur}}}{T_{\max}} \pi \right) \right) \quad (3.3.7)$$

其中， lr_{\max} 是初始学习率（或最大学习率）。 lr_{\min} 是一个很小的学习率，通常接近于 0。 T_{\max} 是半周期的长度，即学习率从 lr_{\max} 降低到 lr_{\min} 所需的 epoch 数。 T_{cur} 是当前 epoch 数对 T_{\max} 取余的结果。

相较于传统的学习率衰减方法（如指数衰减或线性衰减）通常会导致学习率单调递减，这可能使得训练过程过早地收敛到局部最小值。Cosine Annealing 通过周期性地“重启”学习率，提供了一种机制，允许模型在学习率较高时跳出局部最小值陷阱，探索更广泛的参数空间，增加找到更好全局最小值的机会。对于本文中结构相对复杂的深度学习模型而言，合适的学习率衰减算法能够使得学习效率事半功倍。

3.4 多模型集成框架的模型分析

前文中着重探讨了多智能模型集成框架的构建以及一系列优化手段，本节中将着重展开前文中使用到的方法所带来的影响。

3.4.1 海洋气象因子分析

在第 3.3.2 节中，本文详细地介绍了特征处理的关键步骤，包括但不限于特征的标准化和筛选流程。本节进一步扩展了这一讨论，系统地分析了海洋气象因子在进行数据筛选前后与可见度之间的相关性变化。此外，第 2.4.3 节专门探讨了噪声数据的识别与剔除过程，通过表 3.2 提供的数据，本文直观展示了去除噪声点后，数据与标签之间相关性的具体变化。

分析表明，在实施精细化的噪声过滤策略后，相关性的提升幅度可达 22%。这一显著的性能提升不仅验证了噪声剔除方法的高效性，也强调了数据预处理在增强统计模型预测精度和可靠性方面的重要性。

表 3.2：噪声滤除前后结果对比

变量	相关性(噪)	相关性排序(噪)	相关性	相关性排序
露点温度 d2m	-0.22	3	-0.40	3
蒸发量 e	-0.32	2	-0.51	2
位势高度 z	0.04	7	0.08	6
平均海面气压 msl	-0.17	4	-0.29	4
海表温 sst	-0.05	6	0.04	7
降雨量 tp	0.02	8	0.03	8
相对湿度 r	-0.32	1	-0.54	1
风速 U 向	0.02	9	0.04	9
风速 V 向	-0.14	5	-0.25	5

此外，各个海洋气象因子、水文因子本质上依旧是时序数据的形式为主，若希望取得较好的预测效果，则需要满足时序数据预测的相关特性。图 3.9 中对原始数据随机采样了一个样本点，并展示了其中 72 小时时序变化情况。

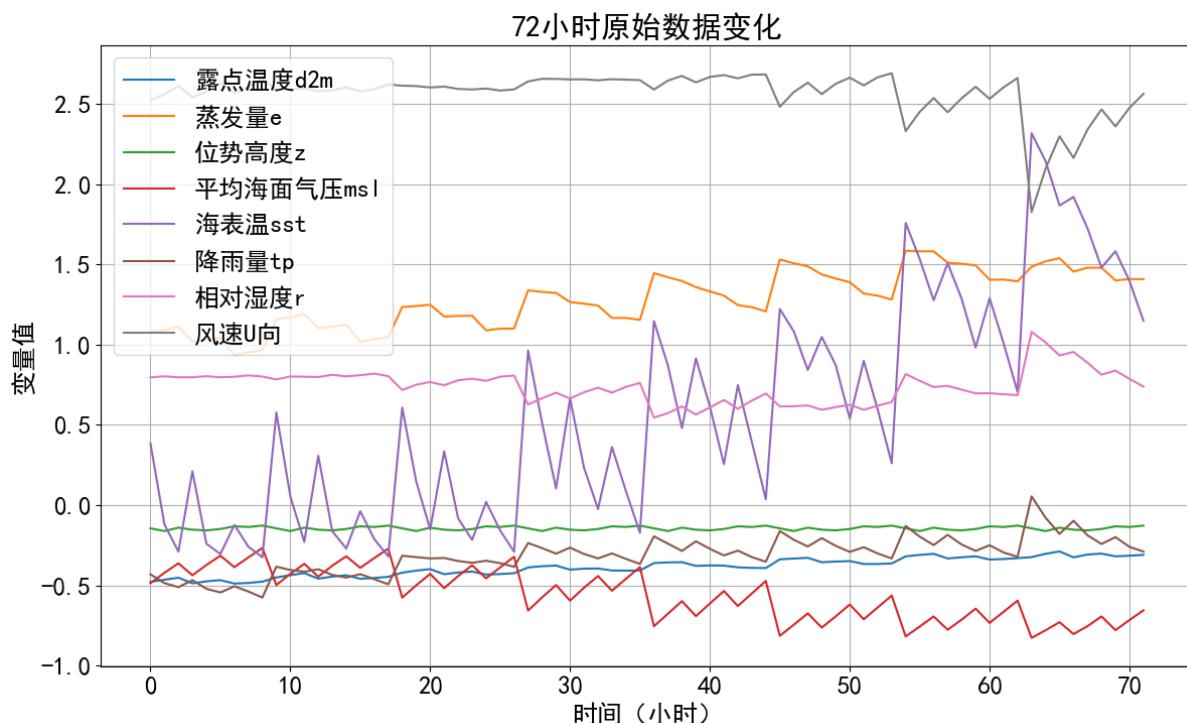


图 3.9 72 小时原始时序数据 稳定性较差

从图中可以看出，原始数据明显缺乏稳定性，多个因子存在较大的变化趋势。一种缓解该情况的手段是数据差分，其中最为常见的差分方式为一阶差分，定义为序列中连续两个观测值之间的差异。往往很多数据在完成一阶差分之后就已经能达到很好的平稳效果：

$$\Delta y_t = y_t - y_{t-1} \quad (3.3.8)$$

在预测完成之后，还需要将差分后的数据恢复至其原始的形态，因此需要使用反差分手段。反差分是差分的逆过程，可表示如下：

$$y_t = \Delta y_t + y_{t-1} \quad (3.3.9)$$

在实际应用中，模型从一阶差分的数据进行预测，预测结束后，使用上述反差分公式逐步恢复出原始序列的预测值。这个过程需要从已知的最后一个原始数据点开始，依次加上差分序列中的相应值。

一阶差分之后的效果可见图 3.10。

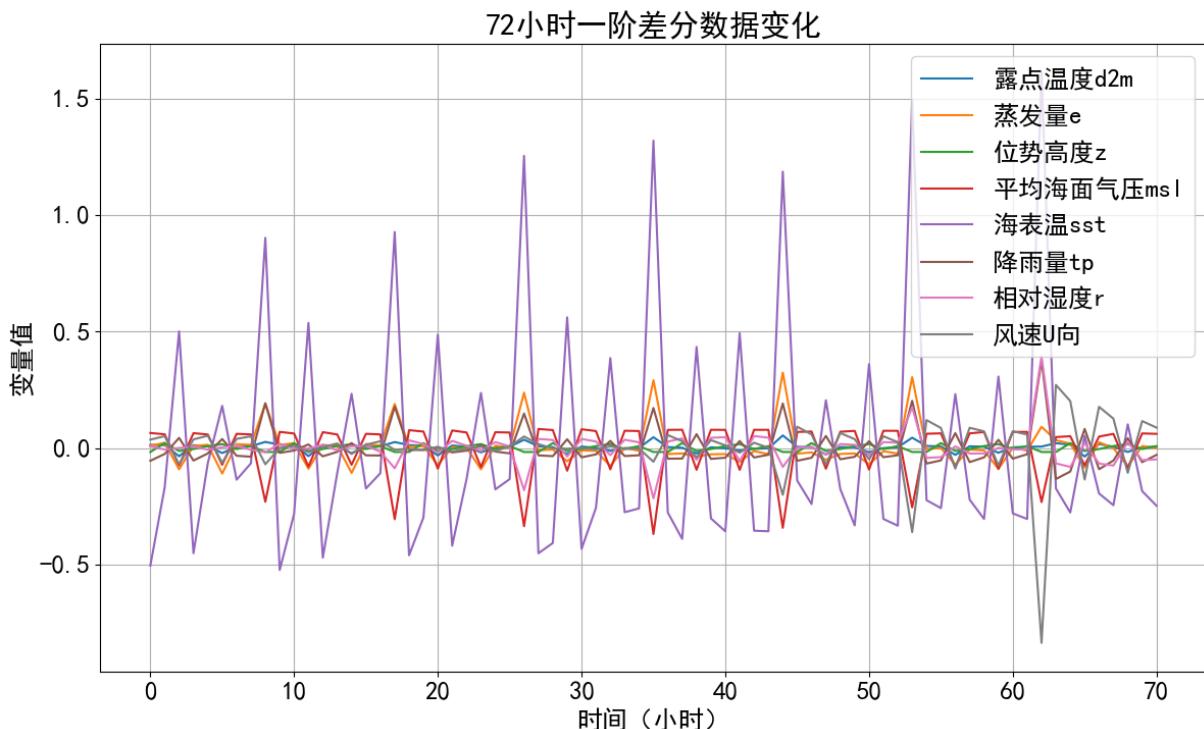


图 3.10 72 小时差分数据 稳定性较好

可以看到，相同样本点的 9 个因子数据在完成一阶差分之后几乎都在 $y = 0$ 附近均匀波动，一阶差分的效果出色。同时，海温 sst 的一阶差分值为恒定值，可以暂时剔除，不参与时序预测，最后以初始值作为输出结果。

3.4.2 模型智能调优

本文中用到的模型数量众多，因此在学习率、神经元个数、梯度提升树叶子节点数量等方面需要一定的调参手段。在实际操作过程中，主要使用 Optuna 调优技术来完成模型的智能超参数调优。Optuna 最主要的优化算法为 Tree-structured Parzen Estimator (TPE)，TPE 算法是贝叶斯优化的一种实现。

而贝叶斯优化则是一种全局优化策略，用于找到目标函数的最小值（或最大值），特别适用于目标函数计算成本高的情况。其核心思想是构建一个概率模型来近似目标函数，然后基于该模型来预测最有可能达到最优值的参数。

在使用 TPE 算法优化学习率的过程中，主要通过构建历史参数性能的概率模型来推断可能导致更优性能的新学习率值。

3.4.3 模型训练效率分析

本节中将展示四个时序模型的训练效果。通过 TensorBoard 可以直观看出 Informer、TFT、N-BEATS 和 DeepVAR 的训练收敛情况。

可以看出，四个模型在若干轮训练之后都能在测试数据集上收敛，Informer、TFT、N-BEATS 和 DeepVAR 的最优 Loss 依次为 0.0204, 0.0233, 0.0209, 0.0225，均处于较优位置。

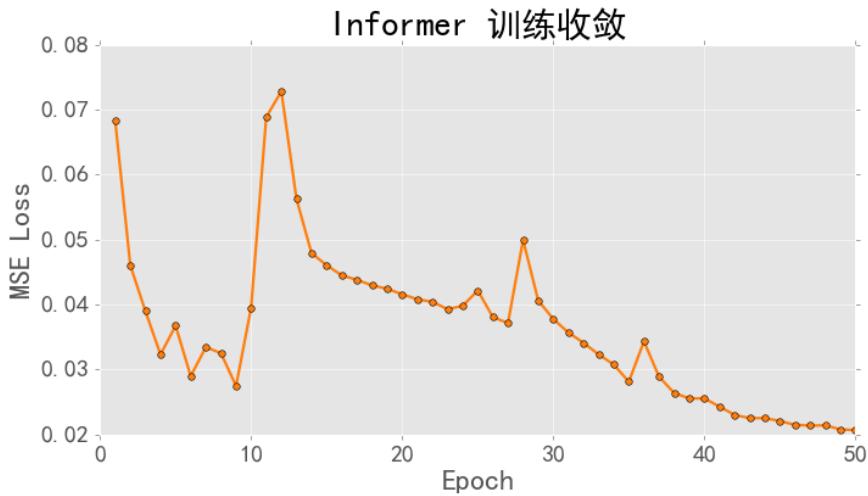


图 3.11 Informer 训练集收敛 50 轮 epoch

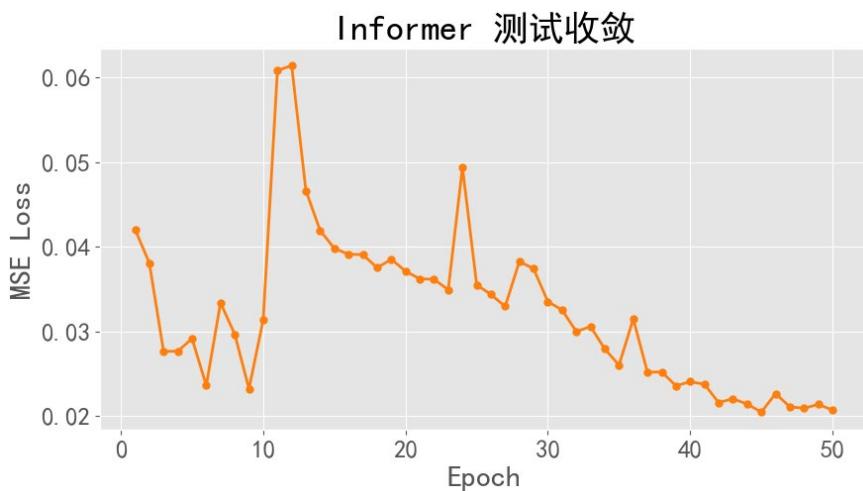


图 3.12 Informer 测试集收敛 50 轮 epoch

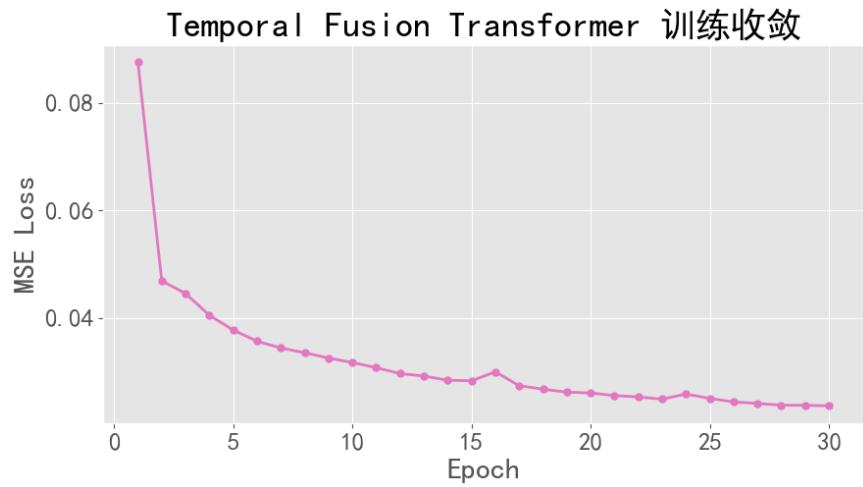


图 3.13 TFT 训练集收敛 30 轮 epoch

Temporal Fusion Transformer 测试收敛

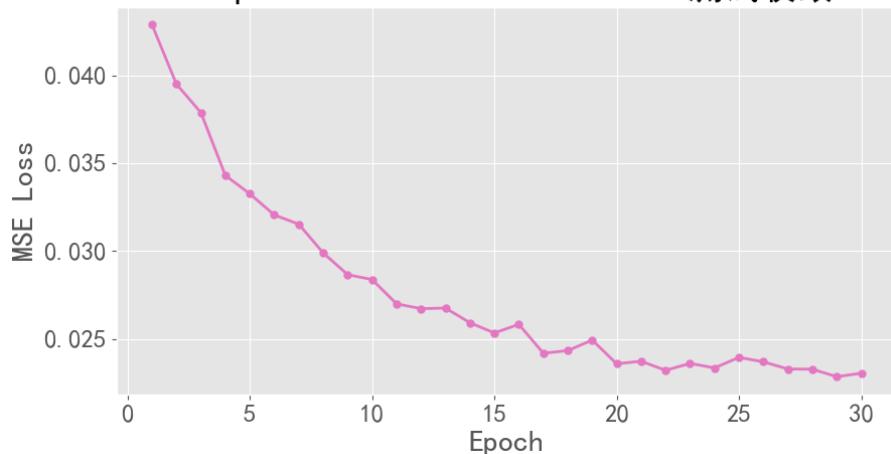


图 3.14 TFT 测试集收敛 30 轮 epoch

N-BEATS 训练收敛

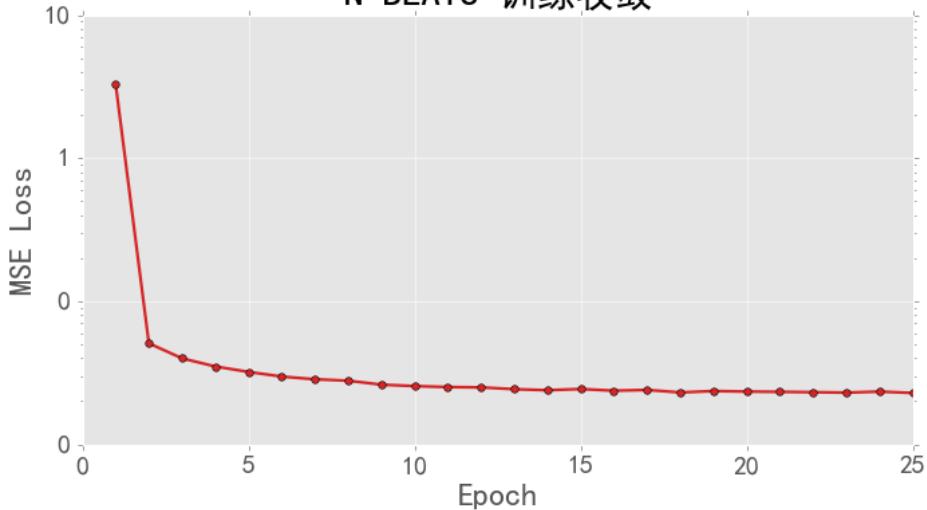


图 3.15 N-BEATS 训练集收敛 25 轮 epoch

N-BEATS 测试收敛

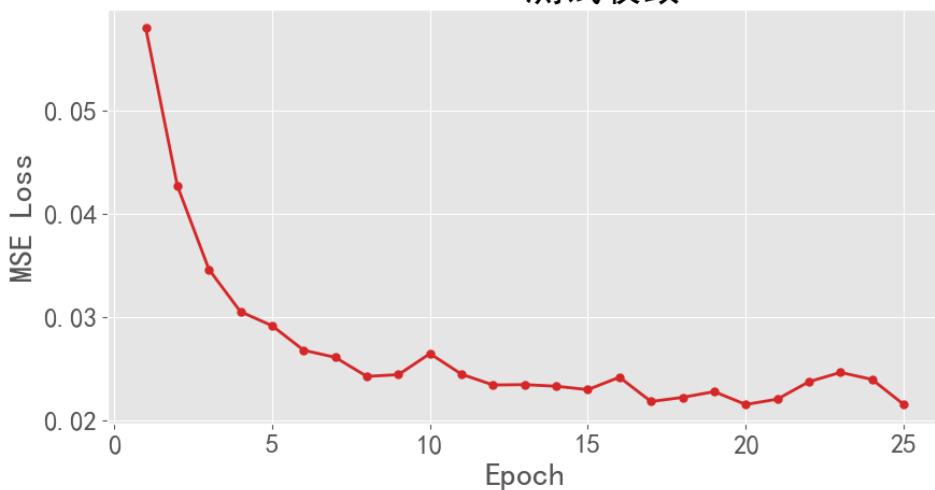


图 3.16 N-BEATS 测试集收敛 25 轮 epoch

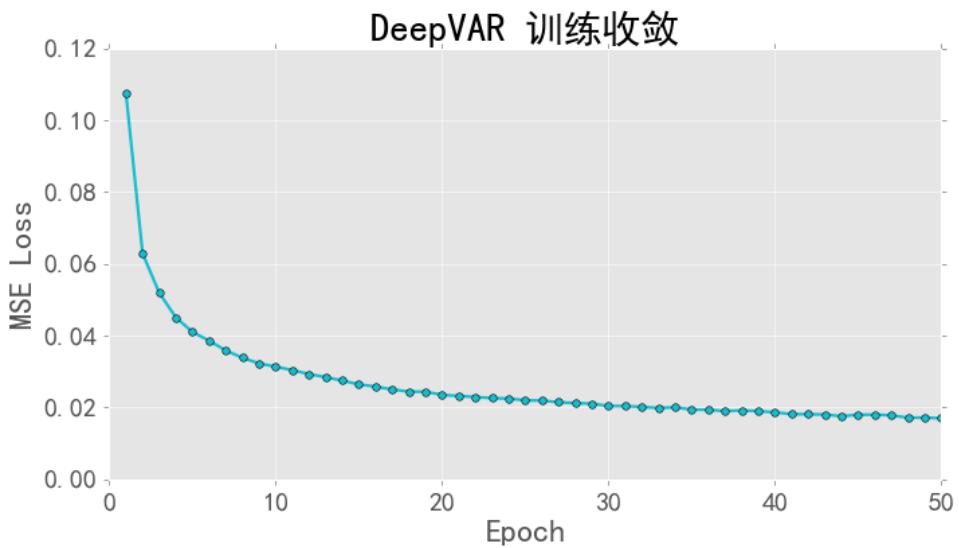


图 3.17 DeepVAR 训练集收敛 50 轮 epoch

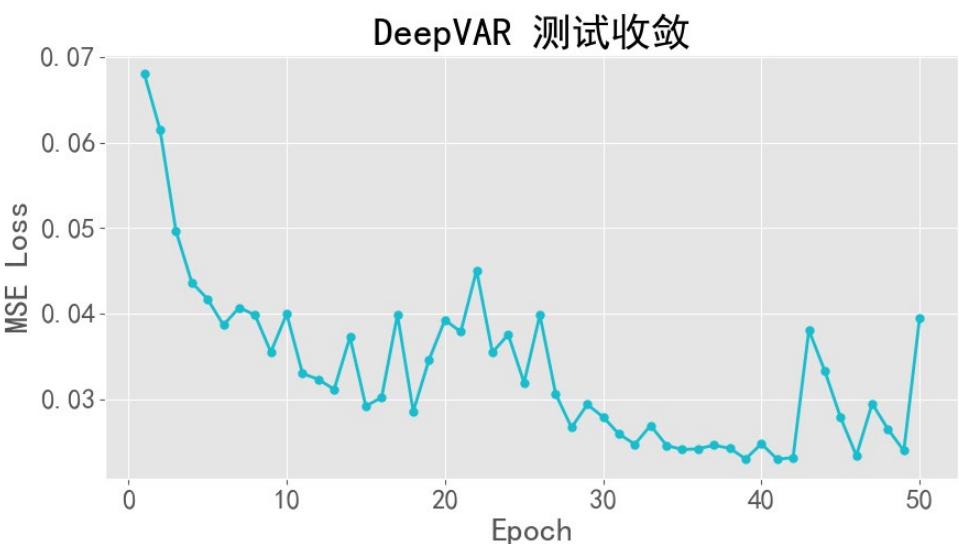


图 3.18 DeepVAR 测试集收敛 50 轮 epoch

4 基于智能集成框架的海雾预报结果分析

到目前为止，本文中基于多智能模型的集成框架的构建及实现已经较为完成地阐述完成。本章将集中于分析与评估框架中各个单元的表现效果以及最终的输出情况。

4.1 实验设置与数据调整

本节中将首先介绍集成框架中各个单元的输入输出维度设置以及训练设备详情。

4.1.1 数据设置与训练设备

实验中使用的训练设备如表 4.1 可见。

表 4.1：训练设备

设备	参数
CPU	Intel i9 13900HX
GPU_1	RTX 4060 8GB
GPU_2	RTX 2070 8GB
RAM_1	32 GB
RAM_2	16 GB
Python	3.10
Pytorch	2.0.0

对于 ERA5 获取的原始数据，最终时刻的数据送入分类单元中。总计 72 小时的数据作一阶差分，差分获得起始点数据，以及差分获得的 71 小时的差分值。送入时序预测单元的数据，batch size 为 180，恰好包含 20 个原始样本，每个样本有 47 个时序窗口，目标是预测未来 24 个时刻的时序目标。每个时刻都有 8 个特征（剔除了海温 sst 的恒定值）。

从时序预测单元中输出的 24 小时一阶差分数据与真实的 24 小时差分数据相减，得到 24 小时差分数据的残差，这部分输入至 GWNet 中完成地区交互训练，之后将 GWNet 的输出与预测差分相加，即可获得最终的 24 小时差分数据预测。之后将差分数据还原至未差分状态，取最后一个时刻的数据输入至分类单元中，即可获得最终的输出结果。

4.1.2 模型结构设置

本节将主要介绍各单元中的模型结构及参数设置。

时序预测单元中主要包含四个模型，分别是 Informer、TFT、N-BEATS 和 DeepVAR，接下来依次展示四种模型的参数详情。

（1） Informer:

Informer 结构中，主要使用了高效的 Prob Sparse 注意力机制，将 `d_model` 的值设置为 512，编码器与解码器均使用 LSTM 层为核心，输出采用一个全连接层。

（2） TFT:

TFT 模型结构包括多层 LSTM 编解码器和带门控残差网络的特征转换模块。在完成编码与解码之后，使用 GRN 网络进一步提取特征，最后通过一个全连接层与残差 Sigmoid 获得最终输出。模型内的隐层维度均为 64。

（3） N-BEATS:

N-BEATS 包含多个 N-Beats 块和一个输出层的模型结构。每个块均由多个全连接层连接。最终由一个全连接层输出最终结果。

（4） DeepVAR:

DeepVAR 模型包括双向 GRU 编码器、多层次全连接解码器、批量归一化和注意力机制，通过转换隐层输出为均值和对数方差向量。隐层维度设置为 256。

海雾分类单元中主要由 LightGBM 与 XGBoost 两种分类模型构成。

（1） LightGBM:

LightGBM 分类器使用 GBDT 算法，设置 31 个叶子节点，学习率为 0.01，特征抽样比例为 0.9，数据抽样比例为 0.8，每 5 次迭代进行一次 bagging，适应

非平衡数据集。

(2) XGBoost:

XGBoost 输出二分类概率，使用基于树的模型，树的最大深度为 5，学习率为 0.01，数据抽样比例为 0.8，特征抽样比例为 0.9，并使用比例权重来处理数据不平衡。

地区因子交互单元主要由 GWNet 构成，其结构如图 4.7 所示。

GraphWaveNet 包含两个图卷积层（分别将 8 维特征扩展到 16 维和 32 维）和两个时间卷积层（将 288 个输入通道压缩到 64 个，再扩展到 72 个输出通道）。从而同时处理时间与空间上的依赖关系。

4.1.3 学习率优化器设置

在时序预测单元与地区交互单元方面，经由 optuna 调优结果得到的学习率大致设置为 0.001 左右可保持较快的收敛速度，同时采用 AdamW 优化器与 0.01 的梯度衰减，同时设置 Kaiming 初始化在 relu 层上的 bias 初始值为 0.01。

在分类单元方面，设置 early stop 情况下的 LightGBM 与 XGBoost 的停止训练轮数依次为 657 与 1329（设置 10 轮正确率没有明显优化则停止训练）。

4.2 二分类器分类效果分析

本节着重分析海雾分类单元的分类效果，主要效果分析使用随机打乱的约 2000 多条测试数据。

4.2.1 XGBoost 分类效果

XGBoost 分类器最终的分类结果如表 4.2 所示。

表 4.2：XGBoost 训练与测试结果

变量	precision	recall	f1-score	总数
无海雾（训练）	0.97	0.91	0.94	7592
有海雾（训练）	0.89	0.97	0.93	5904
无海雾（测试）	0.94	0.88	0.91	1872
有海雾（测试）	0.86	0.93	0.89	1503

最终，测试集上的正确率为 0.9016，比较发现，无海雾情况下准确率较高，但召回率较低。可以查看分类的混淆矩阵与 ROC 曲线，如图 4.8，4.9 所示。

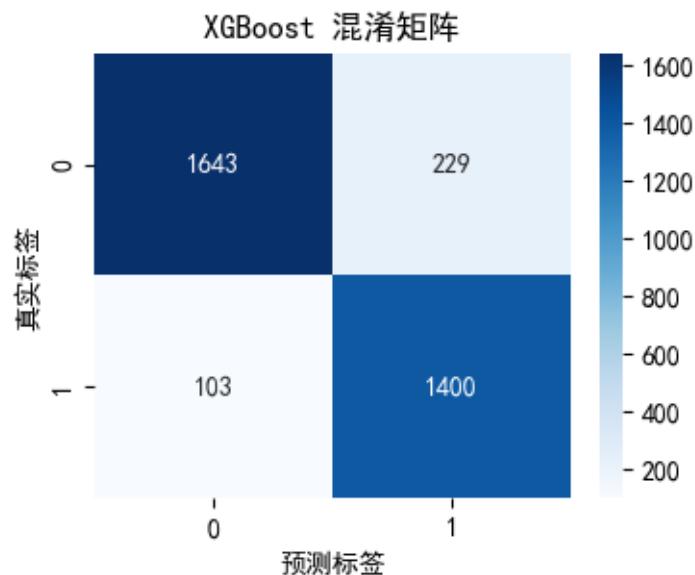


图 4.1 XGBoost 混淆矩阵

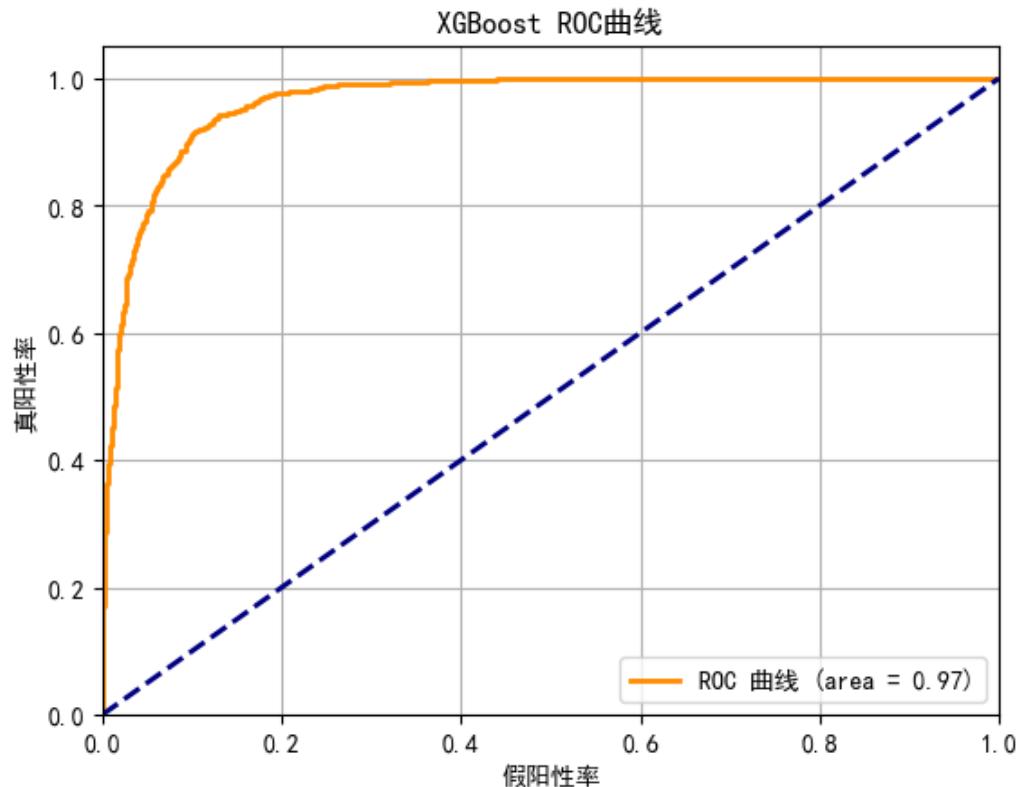


图 4.2 XGBoostROC 曲线

从 ROC 曲线可以看出，XGBoost 分类器表现优秀，与 X 轴围成的面积达到了 0.97。

4.2.2 LightGBM 分类效果

LightGBM 分类器最终的分类结果如表 4.3 所示。

表 4.3：LightGBM 训练与测试结果

变量	precision	recall	f1-score	总数
无海雾（训练）	1.00	1.00	1.00	7592
有海雾（训练）	1.00	1.00	1.00	5904
无海雾（测试）	0.93	0.89	0.91	1872
有海雾（测试）	0.87	0.91	0.89	1503

最终，测试集上的正确率为 0.9019，同样比较发现，无海雾情况下准确率较高，但召回率较低。可以查看分类的混淆矩阵与 ROC 曲线，如图 4.10，4.11 所示。

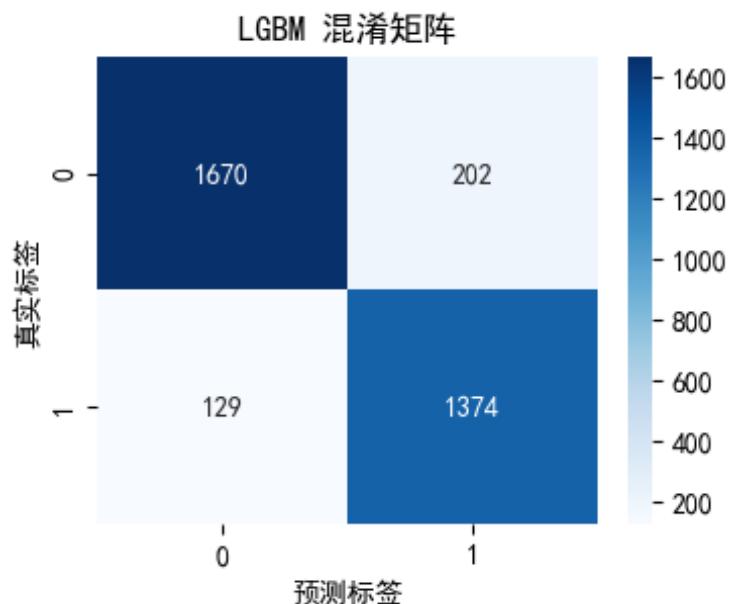


图 4.3 LightGBM 混淆矩阵

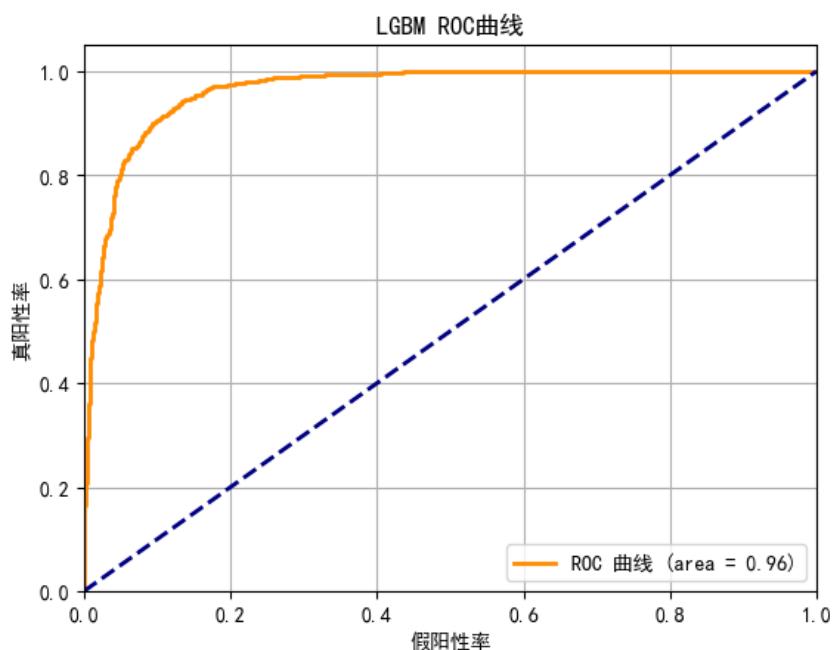


图 4.4 LightGBM ROC 曲线

从 ROC 曲线可以看出，LightGBM 分类器表现同样出色，与 X 轴围成的面积达到了 0.96。

4.2.3 联合分类效果

在完成了两个模型的独立训练之后，使用 PSO 拟合加权系数，可以获得 XGBoost 与 LightGBM 的加权系数为 0.81 与 0.19，加权联合输出预测标签的概率，从而可以获得表 4.4 的输出结果。

表 4.4：模型联合预测结果

变量	precision	recall	f1-score	总数
无海雾（测试）	0.94	0.88	0.91	1872
有海雾（测试）	0.86	0.93	0.89	1503

最终模型输出的准确率为 0.9025，比两个分类器单独预测的效果都要好。并且召回率有了一定的提升。可以查看混淆矩阵与 ROC 曲线，如图 4.12、4.13。

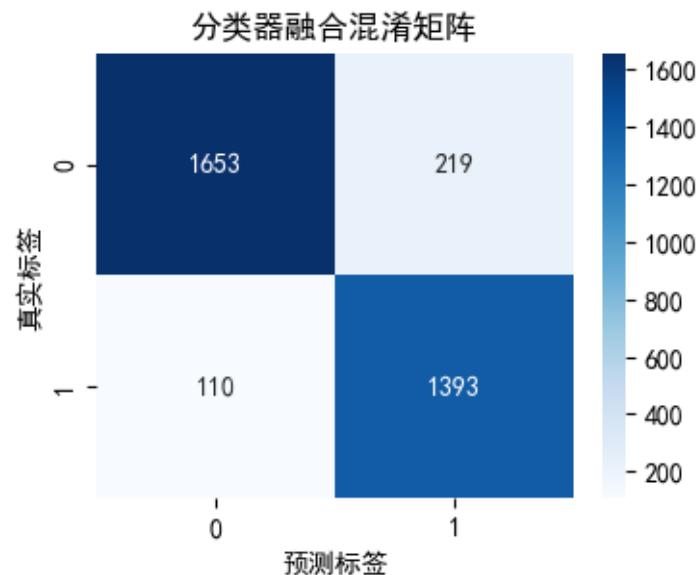


图 4.5 联合分类混淆矩阵

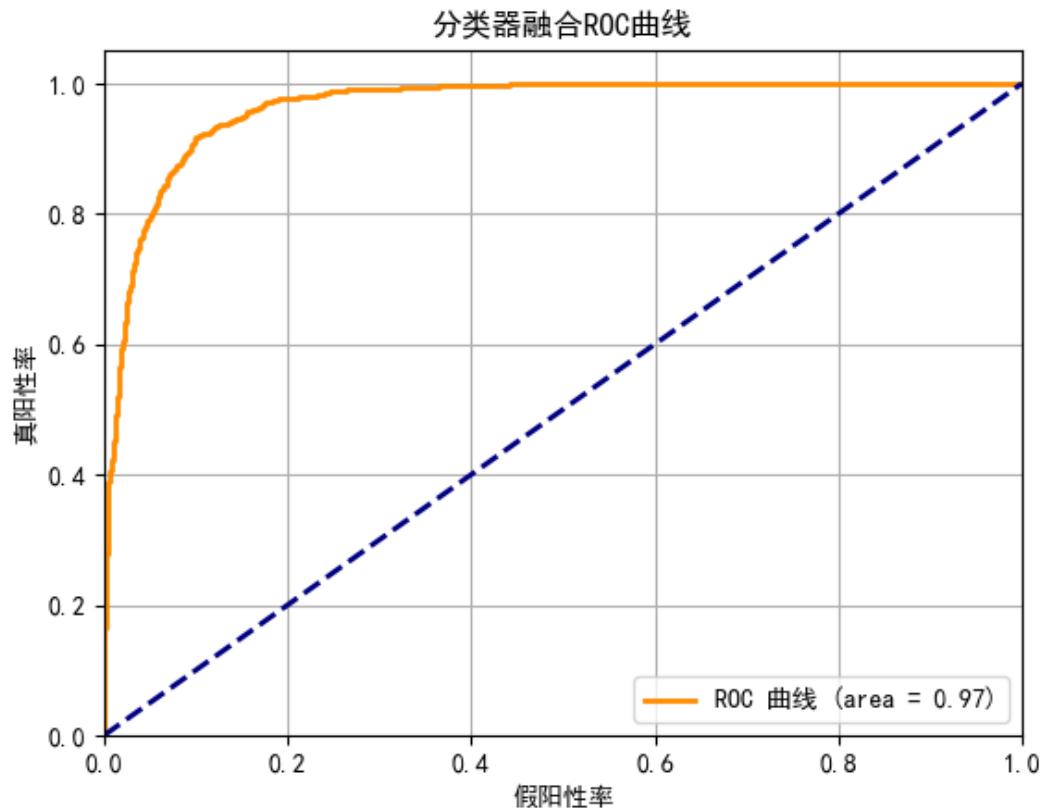


图 4.6 联合分类 ROC 曲线

从混淆矩阵中可以看出，模型在测试集上的分类效果比起两种单独模型效果均更好，错分率降低。ROC 曲线围成的面积为 0.97，也到达了非常好的效果。说明模型联合预测效果优秀，模型融合成功。

4.3 时序预测单元输出结果分析

本节着重分析时序预测单元的输出结果，主要考虑展示通过 47 个时序差分点完成的后 24 个时序差分的预测效果。

4.3.1 时序预测模型输出结果

本节将首先展示 24 个时序预测点的输出结果，依次展示 Informer、TFT、N-BEATS 和 DeepVAR 四种模型。四种模型的 MSE 可见表输出效果可见表 4.5。

表 4.4：模型联合预测结果

Loss	Informer	TFT	N-BEATS	DeepVAR
MSE	0.0216	0.0223	0.0234	0.0242

可以看到，四种模型在训练收敛时在测试集上的效果排序依次是 Informer、TFT、N-BEATS、DeepVAR。从图 4.14 到图 4.17 可以看到四种模型的输出结果。

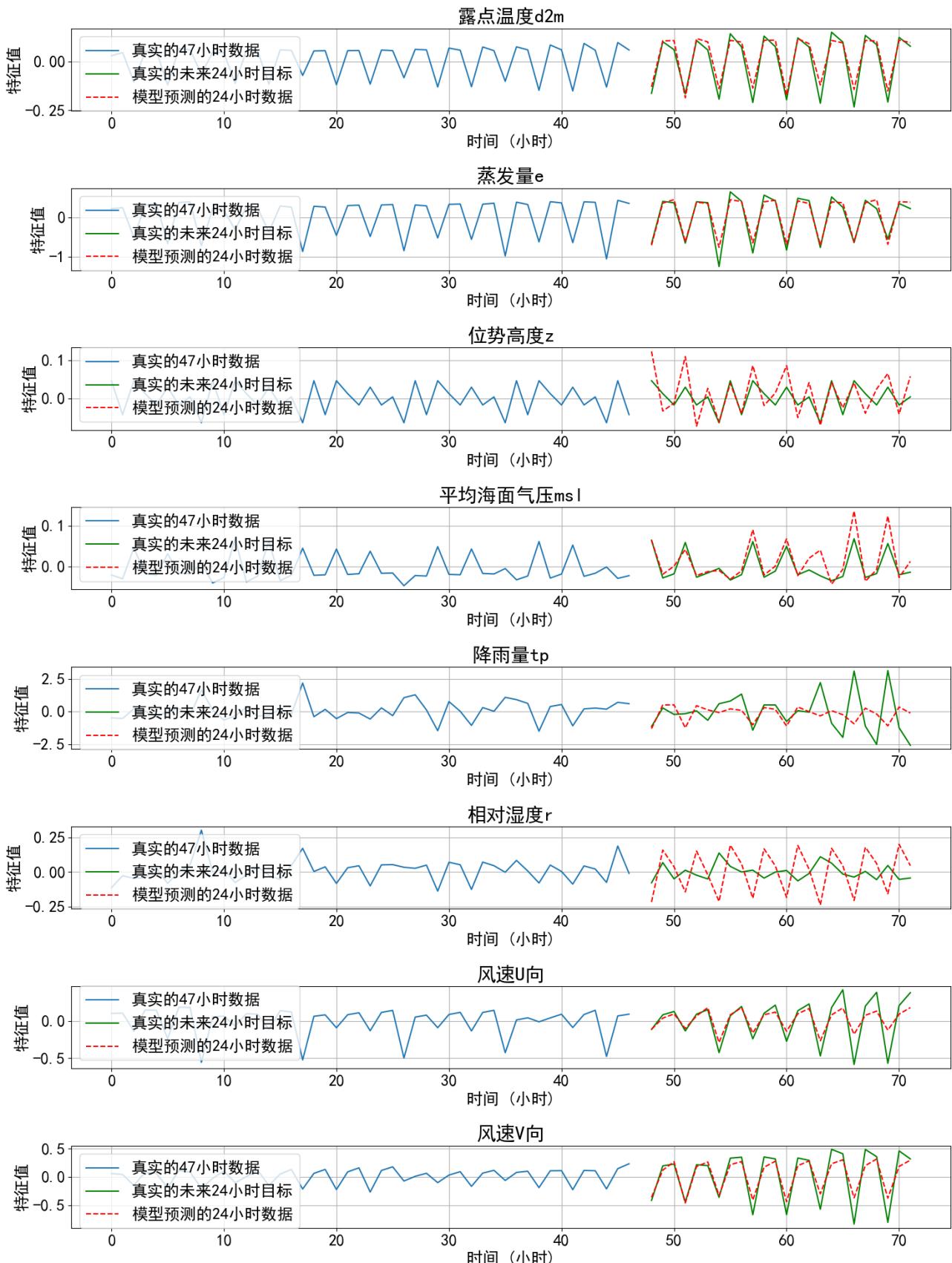


图 4.7 Informer 输出结果 降雨量、相对湿度拟合较差

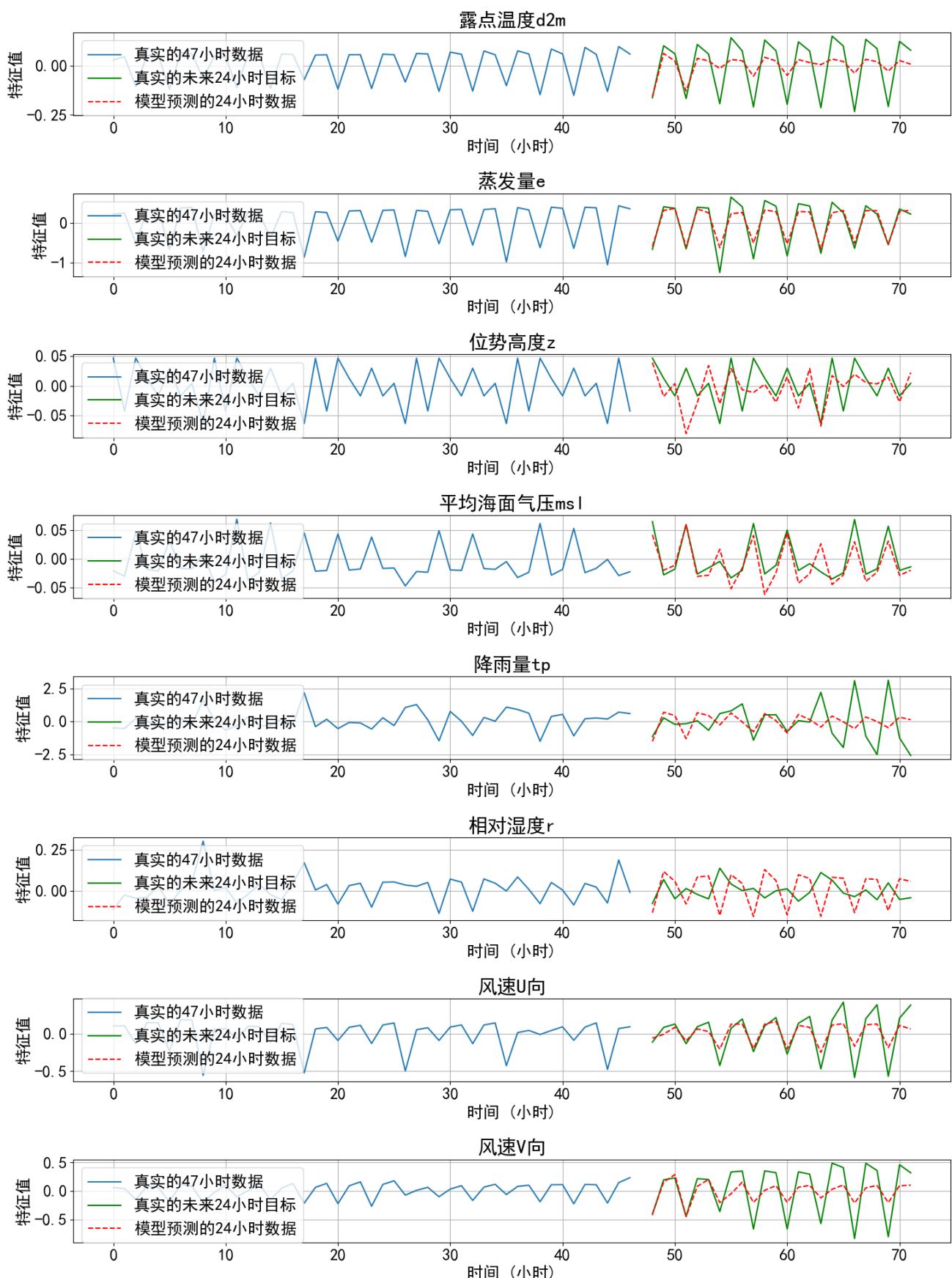


图 4.8 TFT 输出结果 降雨量、相对湿度拟合较差

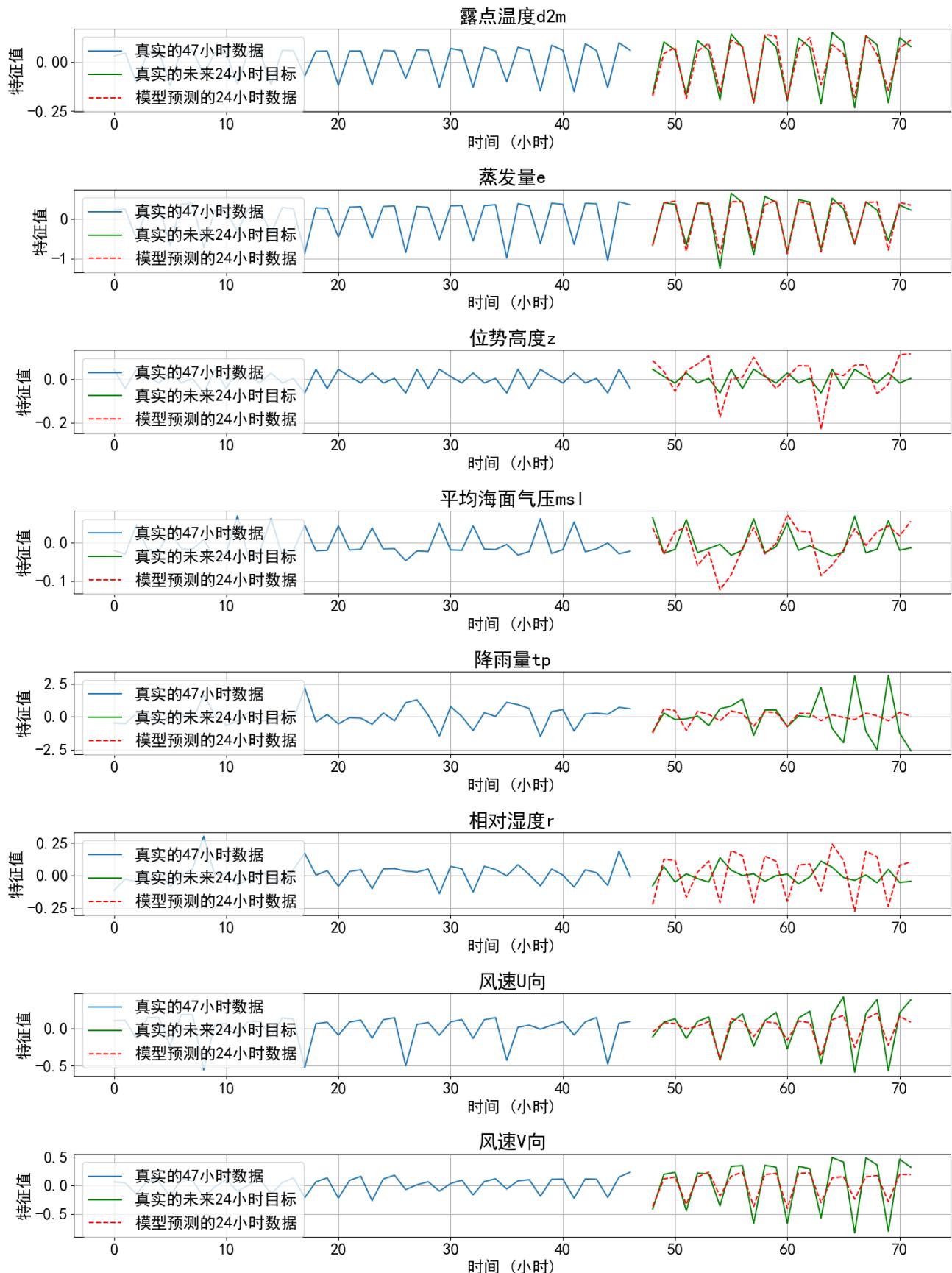


图 4.9 N-BEATS 输出结果 降雨量、相对湿度拟合较差

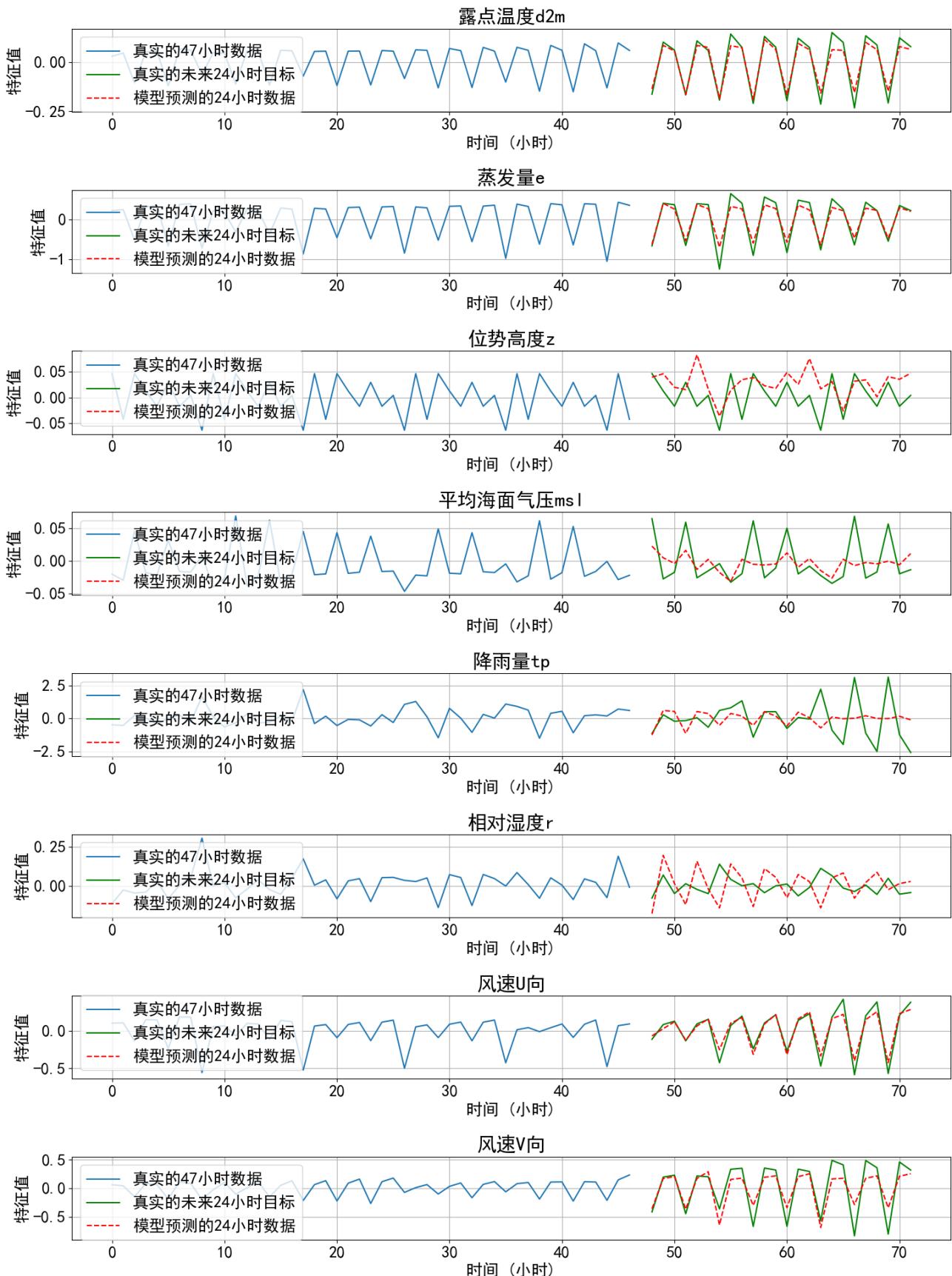


图 4.10 DeepVAR 输出结果 降雨量、相对湿度拟合较差

比较可以看出，Informer 模型主要在降雨量与相对湿度两个因子上的预测效果较差，其他因子的预测效果都较为出色。TFT 的预测相对保守，不易出现骤然大幅度的偏差，这点较 Informer 来说是相对出色的。但是依旧在降雨量和相对湿度两个因子上的预测效果不佳。N-BEATS 模型在降雨量因子上能够在初期保持相对精确的预测，但是随着降雨量骤然变动，N-BEATS 依旧无法及时做出响应。DeepVAR 模型在第 1, 2, 7, 8 个因子上的预测精确度极好，但缺点是预测相对极端，可能出现较大的偏差。

4.3.2 时序单元联合输出结果

完成训练的四个模型采用 PSO 拟合 8 个特征维度总计 32 个权重。拟合完成之后作为时序预测单元的联合输出，其 Loss 达到了 0.0193，比起任意一个单独的时序预测器都是较大的改进。输出结果如图 4.18。从图中可以看出，除了降雨量和相对湿度两个因子以外的所有因子的预测结果都十分优秀，模型融合效果较好。

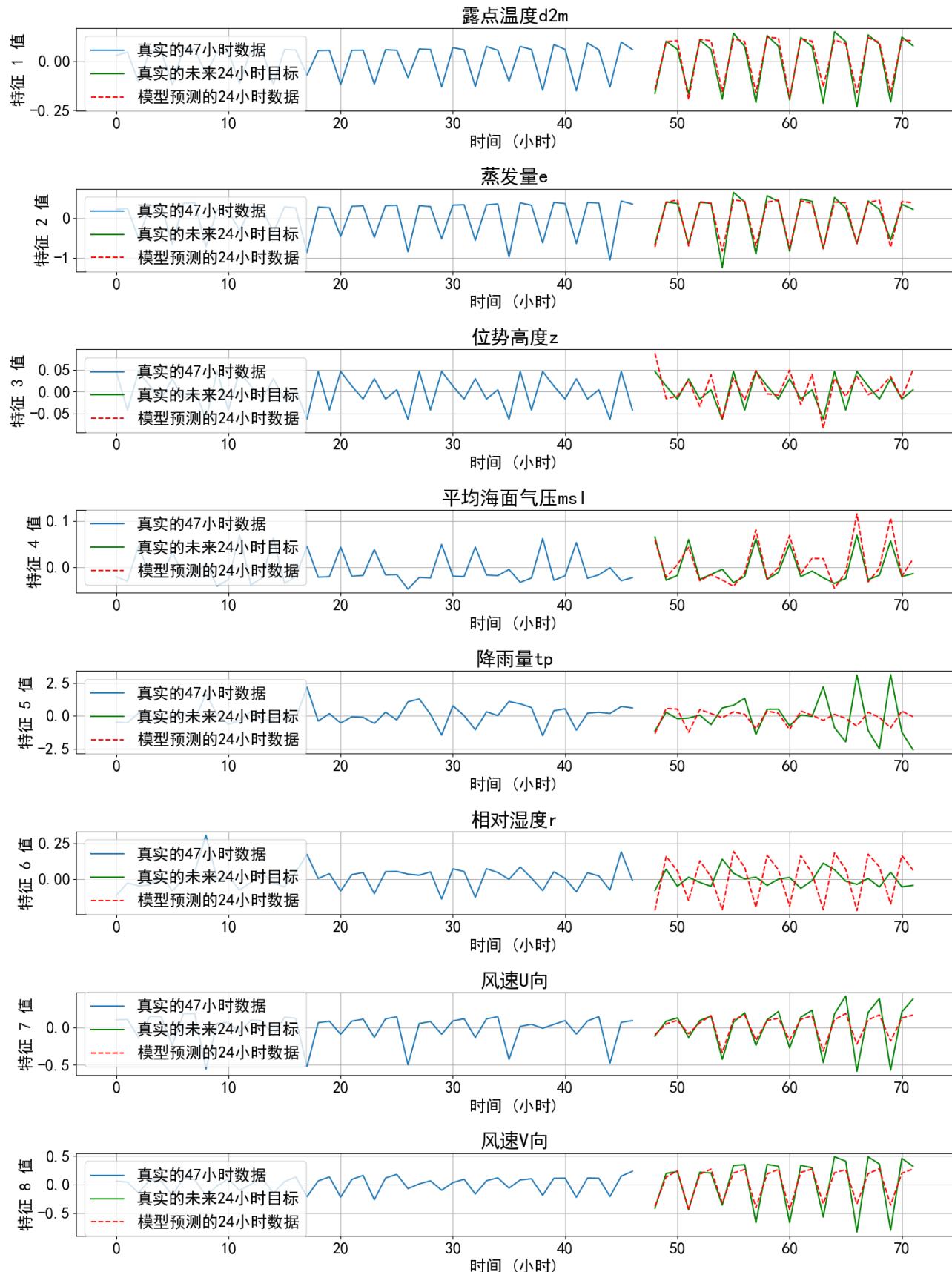


图 4.11 时序单元联合输出结果

4.4 地区因子交互单元输出结果分析

本节着重分析地区因子交互单元的输出结果，根据输入的 24 个时刻的预测数据来拟合此 24 时刻数据与真实值的残差。

4.4.1 GWNet 输出结果

本节中将首先展示 GWNet 的拟合效果，GWNet 作为一种结合了时空间维度处理的神经网络，在残差预测上的表现较为优秀，预测的 loss 达到了 0.0203，输出结果可见图 4.19。可以看到，之前拟合偏差相对较大的相对湿度与降雨量两因子在残差预测部分的结果较好，总体的预测效果表现优异。

下一步就可以将时序单元整体的输出结果与残差预测的结果相加，得到最终的输出结果。

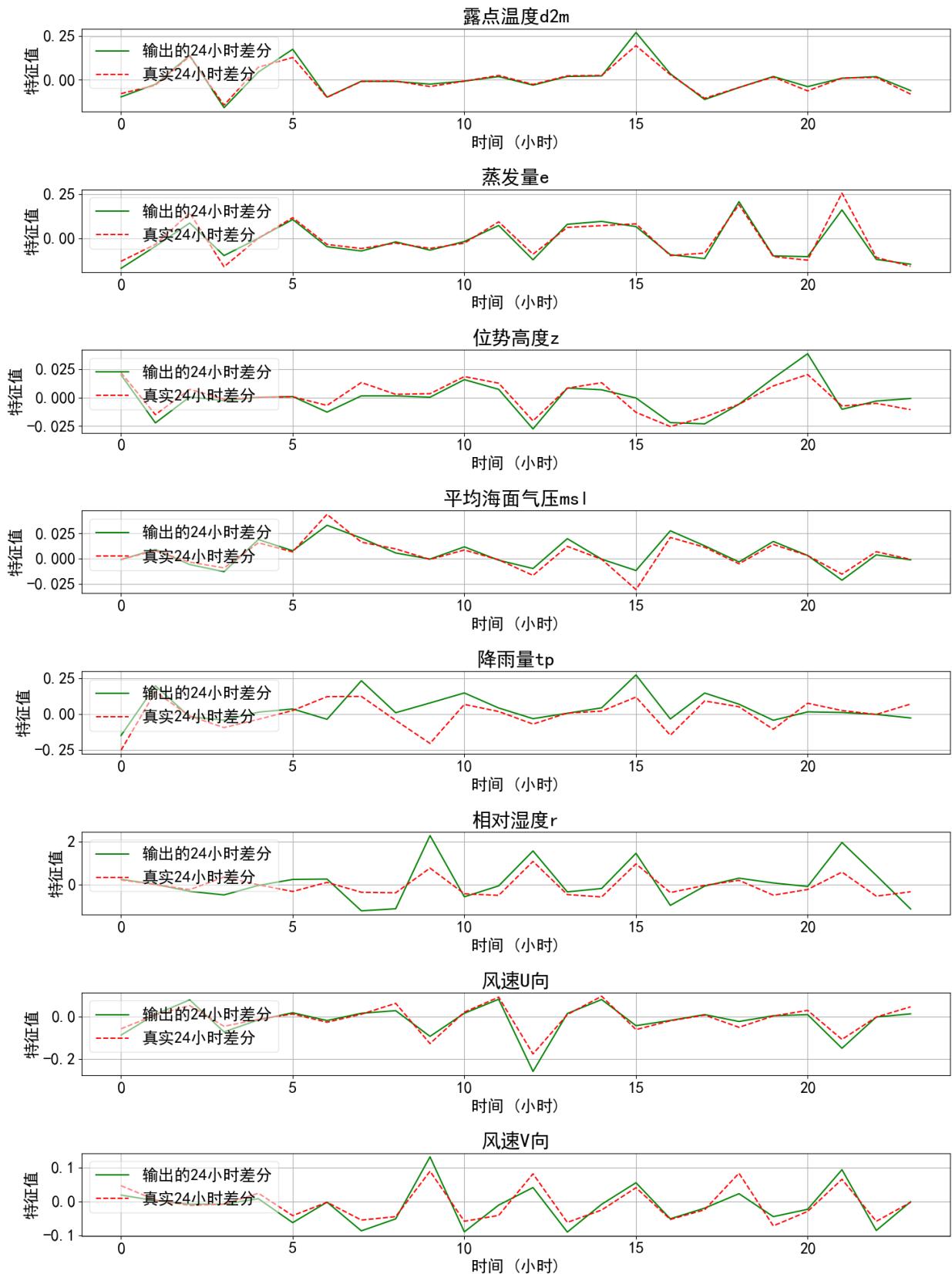


图 4.12 时序单元联合输出结果

4.4.2 一阶差分最终输出结果

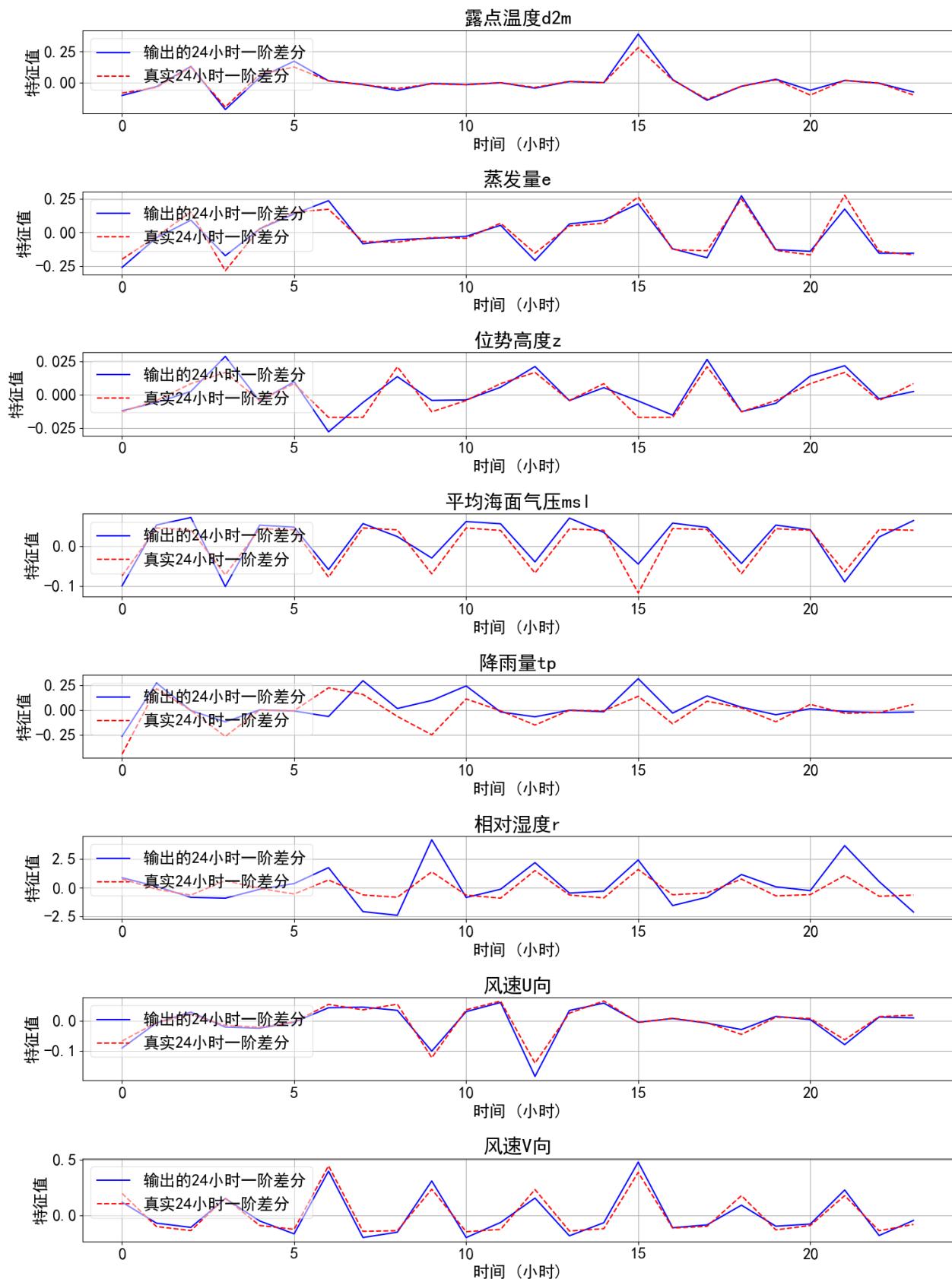


图 4.13 一阶差分最终预测结果

从图 4.20 可以看到最终一阶差分的输出结果。在时序预测器中，除了降雨量与相对湿度的预测效果较差，其他的预测效果都很好，经过了 GWNet 的拟合，预测效果相对均衡，每个变量的预测效果都较好，最终 MSE loss 达到了 0.0062，说明模型的融合效果出色，能够将海洋气象水文因子的相关信息完整预测。之后需要还原为未差分状态的数据，并且完成最终的气象预报。

4.5 智能框架预报能力分析

本节之前的所有内容已经完成了一个相当综合的海洋因子差分输出，在本节内，差分数据将还原回原维度数据，并且通过海雾分类单元完成最后的预报工作。

4.5.1 海洋因子最终预测结果

通过利用数据处理部分保存的时序数据初始值，可以很轻松地还原数据至原维度，还原结果可见图 4.21。

结果可以看到，位势高度，降雨量、相对湿度三个因子的偏差较大，其他几个因子的效果都较好。并且由于海表温度的偏差较小，直接使用原数据作为这一因子的预测输出。

到此为止，海洋气象水文因子已经全部完成预测，接下来需要在分类单元中测试预报效果。

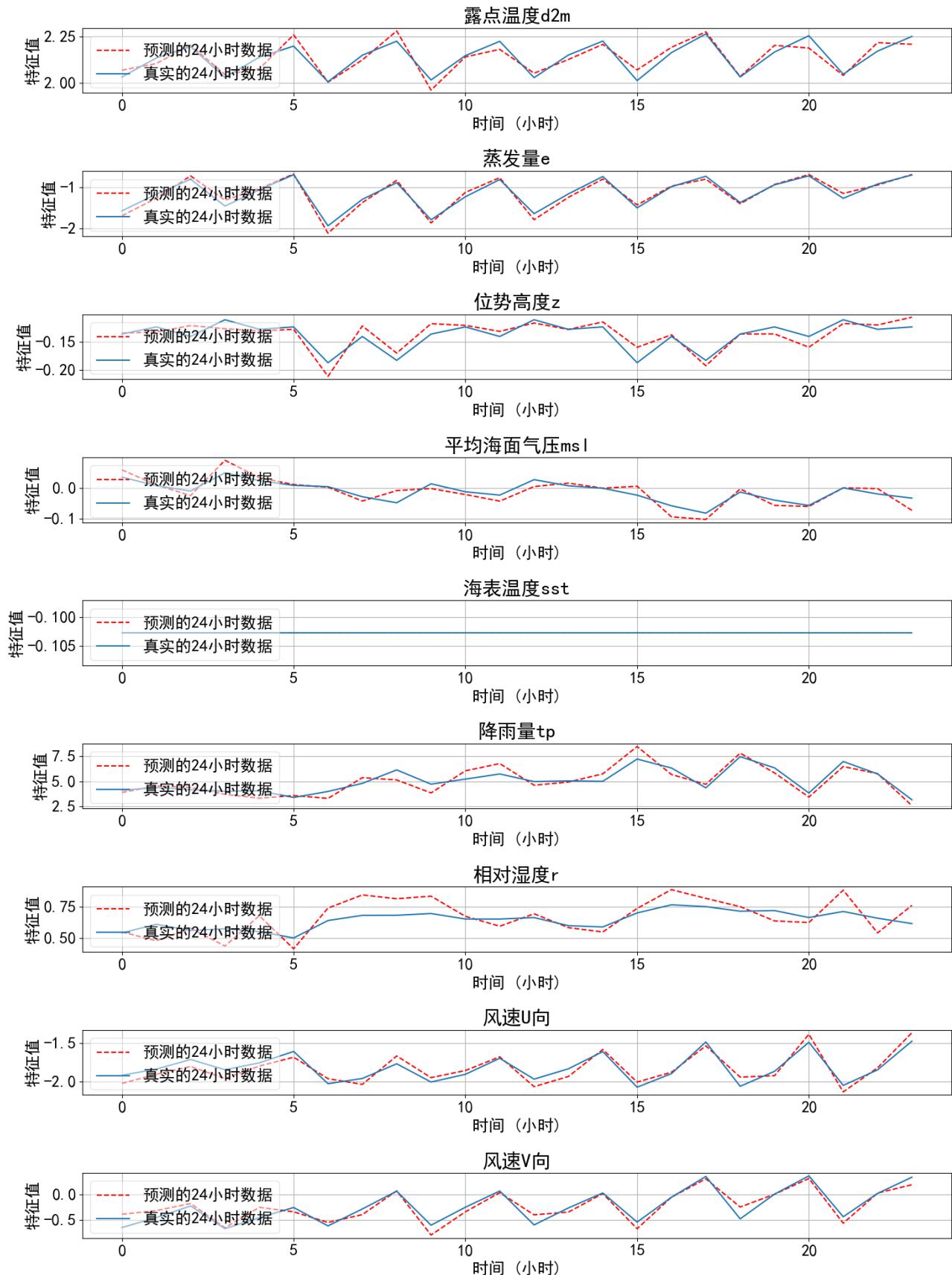


图 4.14 海洋因子最终预测结果

4.5.2 海雾预报性能

本节中将分析框架最终的预报结果。这也是框架最终的输出。随机选取了 2000 条样本中心点区域数据，分别有 9 个特征。并且通过下标完成了时序预测单元和地区交互单元的预测与合并，最终的输出 2000 条预测样本作为预报数据。

首先可以对比这组数据在预测单元的表现。可见表 4.5。

表 4.5：最终输出结果

变量	precision	recall	f1-score	总数
无海雾（预测）	0.78	0.86	0.82	1094
有海雾（预测）	0.81	0.72	0.76	906
无海雾（真实）	0.88	0.85	0.87	1094
有海雾（真实）	0.83	0.87	0.85	906

预测数据最终的 Accuracy 为 0.7945，真实数据的 Accuracy 为 0.858，预测结果与使用真实数据分类的正确率较为接近。预测数据在无海雾标签的召回率甚至高于真实数据，这表明框架的预报性能出色。可以进一步查看混淆矩阵与 ROC 曲线。

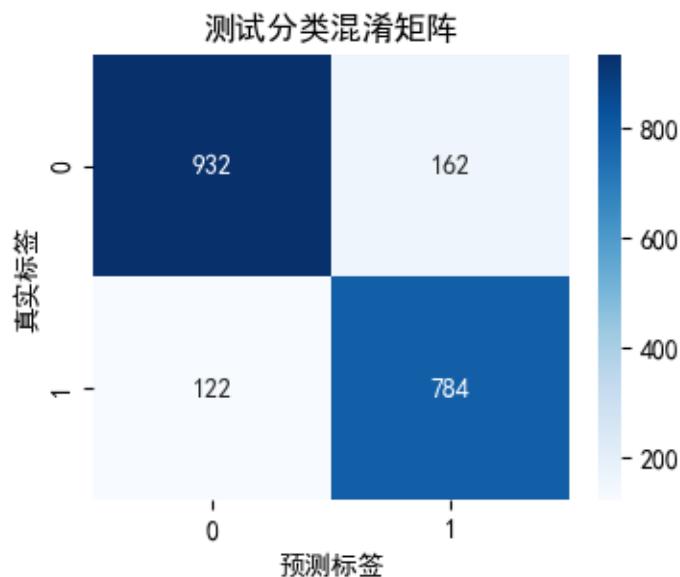


图 4.15 测试类别的混淆矩阵

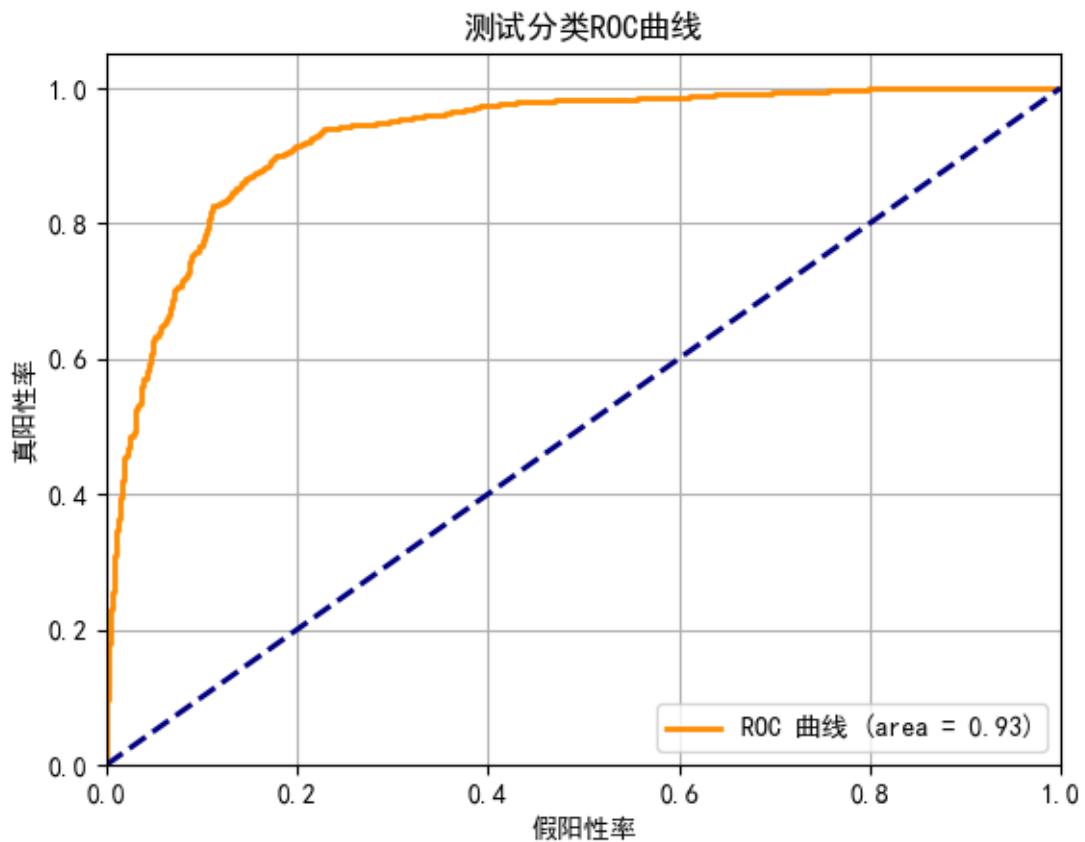


图 4.16 测试类别的 ROC 曲线

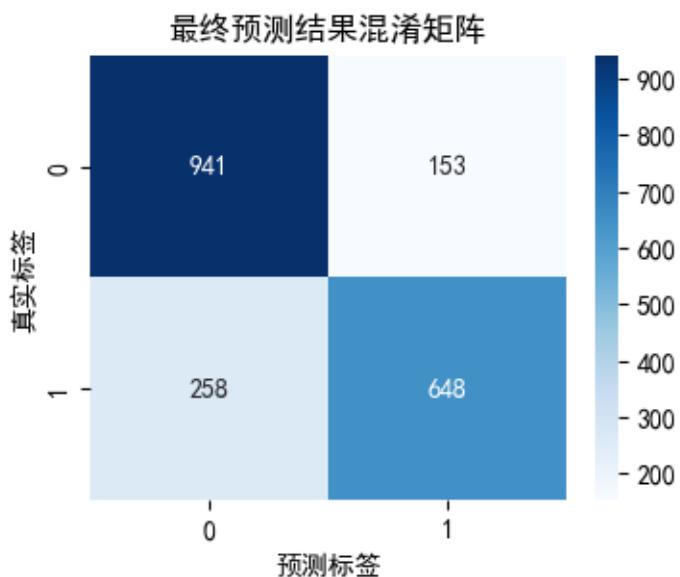


图 4.17 最终预测结果的混淆矩阵

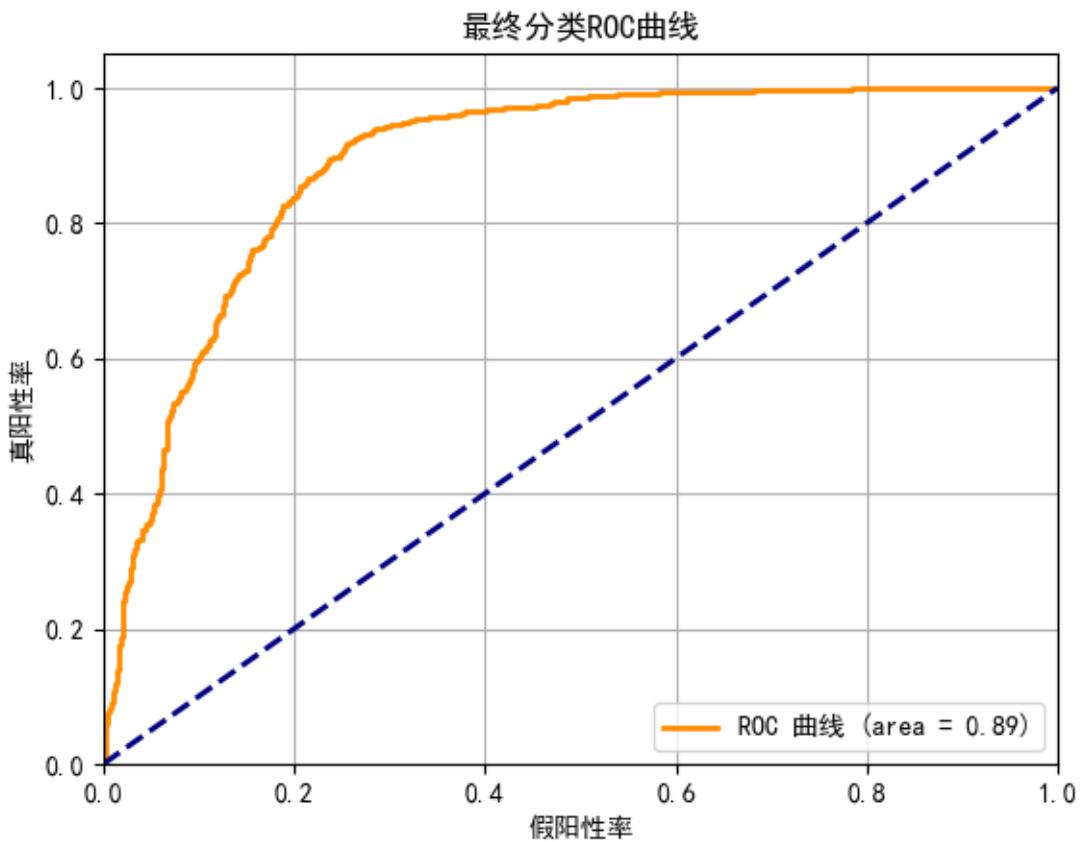


图 4.18 最终预测结果的混淆矩阵

可以看到，多智能模型集成框架对海雾预报的 ROC 曲线与 X 轴围成的面积为 0.89，分类效果出色，框架表现较好，能够较好地预报 24 小时之后的海雾出现情况。

结论

至今为止，由于全球气候变化的复杂性和不可预测性等因素，海雾的预测和预报仍面临诸多挑战。伴随着近些年我国海上丝绸之路的发展，海洋安全的保障愈发迫在眉睫。然而，海雾作为一种严重威胁海上船只航行安全的气象航海危险因素，其强烈的突发性使得精确预测变得极为困难。

在此基础上，本文开创性地提出了一种新的多智能模型集成海雾气象预报框架，实现了海雾 24 小时预报 80%左右的准确率。本文主要从三大方面完成框架的设计。首先使用 Informer、Temporal Fusion Transformer、DeepVAR、N-BEATS 四种深度学习时序预测模型完成加权集成、构建了一个时序预测单元来预测 24 小时的气象水文因子变化。之后使用 Graph WaveNet 图卷积神经网络完成空间地区交互因素的拟合，完成一个地区交互单元。最后利用 LightGBM 与 XGBoost 设计一个海雾分类判别单元，完成海雾出现情况的判别。

通过本文提出的多智能模型集成框架，海雾预测的准确性得到了显著提升。时序预测单元改善了过去方法在处理气象水文数据时的限制，提供了一种更灵活的方法来处理和预测气象数据的时间序列，为海雾的形成条件提供了科学的预测基础。地区交互单元考虑不同地区间的气象因素互动，这在传统海雾预报中往往被忽视，它使得预测模型综合关注区域的气象动态，提高了模型对复杂空间关系处理的能力，使得预测结果更为全面和准确。海雾分类单元摒弃了传统诸多研究中使用决策树的方法，使用梯度提升树集成学习完成海雾预报。这一环节直接关联到海雾的预警发出，提高预警的精确度和实用性，对于海上交通安全和相关经济活动具有重大的实际意义。

然而，本文提出的智能框架仍然存在一定的局限。首先，在时序数据的预测方面，由于全球大气、水文因素存在较大的突变性与不确定性，长期预测会面临许多复杂问题，使用深度学习完成预测在模型规模等方面仍有较大的发展空间。事实上，近些年随着大模型技术的发展，使用大模型完成全球气候预测将为未来的发展趋势，这也是本文的改进空间之一。其次，本文提出的智能框架依旧停留在理论状态，目前没有基于本框架构建较为完善的智能预测系统，因此还未能将本文的想法投入实际应用当中，希望本文能为未来该领域的研究者提供一种立体、新颖的问题解决思路。

参考文献

- [1] 史得道, 吴振玲, 高山红, 等. 海雾预报研究综述[J]. 气象科技进展, 2016 (2): 49-55.
- [2] 叶朗明, 赵建峰. 海雾的特点及预报[J]. 广东科技, 2009, 18(4): 38-39.
- [3] 王厚广, 曲维政. 青岛地区的海雾预报[J]. 海洋预报, 1997, 14(3): 52-57.
- [4] 黄彬, 陈涛, 陈炯, 等. 黄渤海海雾数值预报系统及检验方法研究[J]. 气象科技, 2009, 37(3): 271-275.
- [5] 黄健, 黄辉军, 黄敏辉, 等. 广东沿岸海雾决策树预报模型[J]. 应用气象学报, 2011, 22(1): 107-114.
- [6] 殷美祥, 罗瑞婷, 陈荣泉, 等. 基于 GRU 神经网络的雷州半岛近海岸能见度 短临预报研究[J]. Journal of Tropical Meteorology (1004-4965), 2023, 39(2).
- [7] 陈锦鹏, 林辉, 吴雪菲, 等. 基于集成学习的沿海低能见度天气分类预报方 法[J]. Journal of Tropical Meteorology (1004-4965), 2023, 39(5).
- [8] 周福, 钱燕珍, 金靓, 等. 宁波海雾特征和预报着眼点[J]. 气象, 2015, 41(4): 438-446.
- [9] 覃梦娇. 基于深度学习的海洋环境时空预测方法 [D]. 浙江大学, 2021. DOI:10.27461/d.cnki.gzjdx.2021.002619.
- [10] 梁宏涛, 刘硕, 杜军威, 等. 深度学习应用于时序预测研究综述[J]. Journal of Frontiers of Computer Science & Technology, 2023, 17(6).
- [11] Lim B, Zohren S. Time-series forecasting with deep learning: a survey[J]. Philosophical Transactions of the Royal Society A, 2021, 379(2194): 20200209.
- [12] Koračin D. Modeling and forecasting marine fog[J]. Marine fog: challenges and advancements in observations, modeling, and forecasting, 2017: 425-475.
- [13] Marine fog: challenges and advancements in observations, modeling, and forecasting[M]. Springer, 2017.
- [14] Croft P J, Pfost R L, Medlin J M, et al. Fog forecasting for the southern region: A conceptual model approach[J]. Weather and Forecasting, 1997, 12(3): 545-

556.

- [15] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.
- [16] Zhou H, Zhang S, Peng J, et al. Informer: Beyond efficient transformer for long sequence time-series forecasting[C]//Proceedings of the AAAI conference on artificial intelligence. 2021, 35(12): 11106-11115.
- [17] Lim B, Arik S Ö, Loeff N, et al. Temporal fusion transformers for interpretable multi-horizon time series forecasting[J]. International Journal of Forecasting, 2021, 37(4): 1748-1764.
- [18] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.
- [19] Fatouros G, Makridis G, Kotios D, et al. DeepVaR: a framework for portfolio risk assessment leveraging probabilistic deep neural networks[J]. Digital finance, 2023, 5(1): 29-56.
- [20] Shumway R H, Stoffer D S, Shumway R H, et al. ARIMA models[J]. Time series analysis and its applications: with R examples, 2017: 75-163.
- [21] Oreshkin B N, Carpov D, Chapados N, et al. N-BEATS: Neural basis expansion analysis for interpretable time series forecasting[J]. arXiv preprint arXiv:1905.10437, 2019.
- [22] Wu Z, Pan S, Long G, et al. Graph wavenet for deep spatial-temporal graph modeling[J]. arXiv preprint arXiv:1906.00121, 2019.
- [23] Ke G, Meng Q, Finley T, et al. Lightgbm: A highly efficient gradient boosting decision tree[J]. Advances in neural information processing systems, 2017, 30.
- [24] Chen T, Guestrin C. Xgboost: A scalable tree boosting system[C]//Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016: 785-794.

附录 A

表 A1：多种模型二分类效果对比

模型	precision	recall	f1-score	ROC
XGBoost	0.91	0.90	0.90	0.98
LightGBM	0.90	0.91	0.90	0.97
MLP	0.88	0.79	0.83	0.91
SVM	0.85	0.74	0.79	0.88

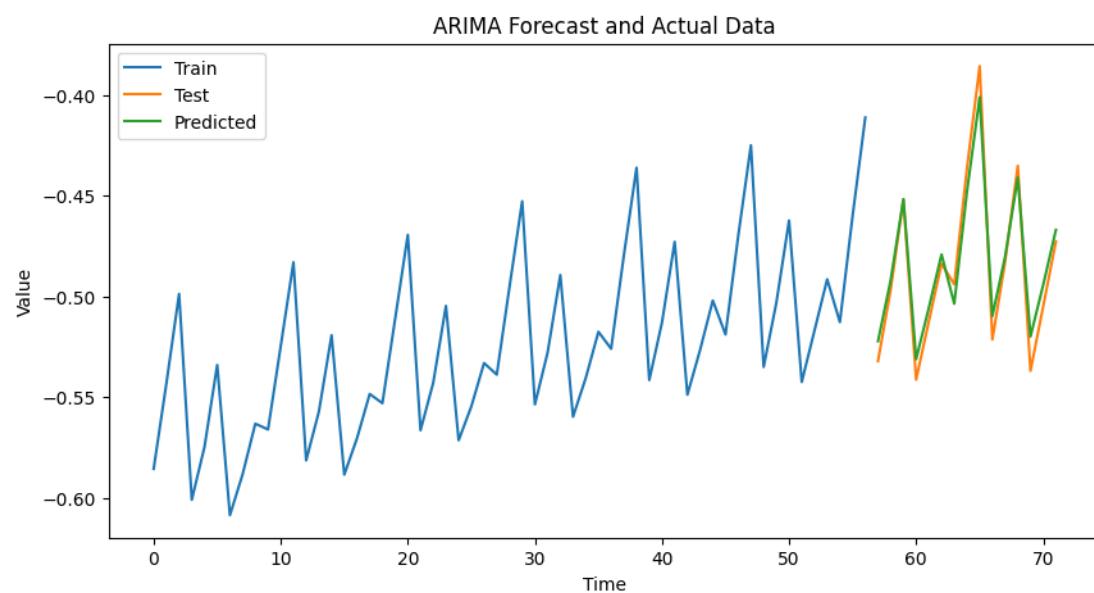


图 A1 Arima 模型预测效果图（单一特征预测）

附录 B

```
Informer(  
    (attention): ProbSparseAttention(  
        (query): Linear(in_features=512, out_features=512, bias=True)  
        (key): Linear(in_features=512, out_features=512, bias=True)  
        (value): Linear(in_features=512, out_features=512, bias=True)  
    )  
    (encoder): Encoder(  
        (lstm): LSTM(8, 512, batch_first=True)  
    )  
    (decoder): Decoder(  
        (lstm): LSTM(512, 512, batch_first=True)  
        (fc): Linear(in_features=512, out_features=512, bias=True)  
    )  
    (fc): Linear(in_features=512, out_features=8, bias=True)  
)
```

图 B1 Informer 结构图

```
TemporalFusionTransformer(  
    (encoder_lstm): LSTM(8, 64, num_layers=2, batch_first=True)  
    (decoder_lstm): LSTM(64, 64, num_layers=2, batch_first=True)  
    (grn): GatedResidualNetwork(  
        (fc1): Linear(in_features=64, out_features=64, bias=True)  
        (fc2): Linear(in_features=64, out_features=8, bias=True)  
        (gate): Linear(in_features=64, out_features=8, bias=True)  
        (activation): ReLU()  
        (sigmoid): Sigmoid()  
    )  
)
```

图 B2 TFT 结构图

```
NBeatsModel(  
    (blocks): ModuleList(  
        (0-3): 4 x NBeatsBlock(  
            (fc1): Linear(in_features=376, out_features=512, bias=True)  
            (relu1): ReLU()  
            (fc2): Linear(in_features=512, out_features=512, bias=True)  
            (relu2): ReLU()  
            (fc3): Linear(in_features=512, out_features=376, bias=True)  
        )  
    )  
    (output_layer): Linear(in_features=1504, out_features=192,  
    bias=True)  
)
```

图 B3 N-BEATS 结构图

```
DeepVAR(  
    (encoder): GRU(8, 128, num_layers=2, batch_first=True, dropout=0.2,  
    bidirectional=True)  
    (encoder_bn): BatchNorm1d(256, eps=1e-05, momentum=0.1, affine=True,  
    track_running_stats=True)  
    (hidden_to_mean): Linear(in_features=256, out_features=64, bias=True)  
    (hidden_to_logvar): Linear(in_features=256, out_features=64, bias=True)  
    (attention): Linear(in_features=256, out_features=1, bias=True)  
    (decoder_fc1): Linear(in_features=64, out_features=256, bias=True)  
    (decoder_bn1): BatchNorm1d(256, eps=1e-05, momentum=0.1, affine=True,  
    track_running_stats=True)  
    (decoder_fc2): Linear(in_features=256, out_features=256, bias=True)  
    (decoder_bn2): BatchNorm1d(256, eps=1e-05, momentum=0.1, affine=True,  
    track_running_stats=True)  
    (decoder_fc3): Linear(in_features=256, out_features=192, bias=True)  
)
```

图 B4 DeepVAR 结构图

```
params = {  
    'objective': 'binary', # 二分类问题  
    'metric': 'binary_logloss', # 评估指标为二分类对数损失  
    'boosting_type': 'gbdt', # 使用 GBDT 算法  
    'num_leaves': 31, # 叶子节点数  
    'learning_rate': 0.01, # 学习率  
    'feature_fraction': 0.9, # 特征的子抽样比例  
    'bagging_fraction': 0.8, # 数据的子抽样比例  
    'bagging_freq': 5, # bagging 的频率  
    'is_unbalance': False # 如果数据集不平衡, 设置为 True  
}
```

图 B5 LightGBM 结构图

```
params = {
    'objective': 'binary:logistic', # 二分类问题，输出概率
    'eval_metric': 'logloss', # 评估指标为对数损失
    'booster': 'gbtree', # 使用基于树的模型
    'max_depth': 5, # 树的最大深度
    'eta': 0.01, # 学习率
    'subsample': 0.8, # 数据的子抽样比例
    'colsample_bytree': 0.9, # 特征的子抽样比例
    'scale_pos_weight': sum(y_train == 0) / sum(y_train == 1), # 用于
    处理不平衡数据
}
```

图 B6 XGBoost 结构图

```
GraphWaveNet(
    (graph_conv1): GraphConvolution(
        (fc): Linear(in_features=8, out_features=16, bias=True)
    )
    (graph_conv2): GraphConvolution(
        (fc): Linear(in_features=16, out_features=32, bias=True)
    )
    (temporal_conv1): TemporalConvolutionLayer(
        (conv): Conv1d(288, 64, kernel_size=(2,), stride=(1,), padding=(1,))
    )
    (temporal_conv2): TemporalConvolutionLayer(
        (conv): Conv1d(64, 72, kernel_size=(2,), stride=(1,), padding=(1,))
    )
)
```

图 B7 GWNet 结构图

致 谢

本次毕业设计到此也已接近尾声，回想数月以来在电脑前的日日夜夜，我不由感慨万千。这数月，乃至数年以来，我受到了太多优秀的老师与同学们的帮助，此时此刻无论怎样的致谢都无法完整传达我的谢意。

首先，我想向我的导师徐凌宇教授表达我最诚挚的感谢。在我的人生即将踏入社会的这个阶段，徐老师作为我的引路人，指引着我向着人工智能的更高层进发。我感谢徐老师的耐心教导，在与老师的交谈中，我明白了到目前为止的人工智能究竟在做些什么，知道了我们还需要做些什么，知道了自己还欠缺些什么，这些想法是我此前从未有过的。在此，我想要向徐老师表达我最由衷的感谢。

回想在开题初期，我曾有过无数个想法希望在设计中实现，但当我真正面对模型出现的诸多问题时，我才发现自己在 AI 领域的造诣远远达不到能够以“合格”一词来描述。课堂上学到的知识如果没有真正实践过，很多问题是很难发现的，这是本次毕设真正教会我的。在完成这篇论文之时，我曾遇到过许多困难，也请教过许多老师和同学，在此，我希望向这一路上所有帮助过我的老师与同学表达我诚挚的感激。

苏天尧

于上海大学图书馆

2024 年 05 月 19 日

