

基于元特征映射与经验黑翅鸢优化的自动化网络流量预测框架

蒋献^{1,2}, 陈嗣元², 苟敬文², 张子天², 诸葛斌², 董黎刚²

(1. 浙江工商大学统计与数据科学学院, 浙江 杭州 310018; 2. 浙江工商大学信息与电子工程学院, 浙江 杭州 310018)

摘要: 针对现有网络流量预测方法中存在的模型与数据模式不匹配、超参数优化效率低下等问题, 提出基于元特征映射与经验黑翅鸢优化的自动化网络流量预测框架。该框架首先通过构建一个包含 38 维元特征的经验库, 实现从数据特征到最优预测模型的智能映射。其次, 设计了一种经验驱动的改进黑翅鸢优化算法, 显著提升了超参数搜索的效率和全局最优解发现能力。实验结果表明, 所提框架在预测精度与计算效率上均全面优于多种基线框架。

关键词: 网络流量预测; 元特征; 超参数优化; 大语言模型

中图分类号: TP302

文献标志码: A

DOI: 10.11959/j.issn.1000-436x.2025215

Automated network traffic prediction framework based on meta-feature mapping and empirical black-winged kite optimization

JIANG Xian^{1,2}, CHEN Siyuan², GOU Jingwen², ZHANG Zitian², ZHUGE Bin², DONG Ligang²

1. School of Statistics and Data Science, Zhejiang Gongshang University, Hangzhou 310018, China

2. School of Information and Electronic Engineering, Zhejiang Gongshang University, Hangzhou 310018, China

Abstract: To address the challenges of inefficient model matching and parameter optimization in network traffic prediction, an automated network traffic prediction framework based on meta-feature mapping and empirical black-winged kite optimization was proposed. The framework began by constructing an empirical repository comprising 38-dimensional meta-features, enabling intelligent mapping from data characteristics to the optimal prediction model. Subsequently, an experience-driven enhanced black-winged kite optimization algorithm (EBKA) was designed, and hyperparameter search efficiency and global optimization capability was improved. Experiments demonstrate that the proposed framework outperforms baseline methods in both prediction accuracy and computational efficiency.

Keywords: network traffic prediction, meta-feature, hyperparameter optimization, large language model

收稿日期: 2025-09-04; 修回日期: 2025-11-16

通信作者: 董黎刚, donglg@zjgsu.edu.cn

基金项目: 国家自然科学基金资助项目(No.W2421086, No.61871468); 浙江省重点研发计划基金资助项目(No.2026C02A1244); 桐乡通用人工智能研究院项目(No.TAGI2-B-2024-0014); 浙江省新型网络标准及应用技术重点实验室(No.2013E10012); 浙江省“十四五”第二批研究生教学改革项目(No.3100JYN4118002G-714); 国家级创新创业训练项目(No.1120XJ2525246/2025, No.1010XJ2524078/2024); 2024年浙江省大学生科技创新活动计划(No.2024R408A047)

Foundation Items: The National Natural Science Foundation of China (No.W2421086, No.61871468), Zhejiang Provincial Key Research and Development Program (No.2026C02A1244), Tongxiang General Artificial Intelligence Research Institute Project (No.TAGI2-B-2024-0014), Zhejiang Provincial Key Laboratory of New Network Standards and Application Technology (No.2013E10012), The Second Batch of Postgraduate Teaching Reform Projects in Zhejiang Province during the “14th Five-Year Plan” Period (No.3100JYN4118002G-714), The National Innovation and Entrepreneurship Training Program (No.1120XJ2525246/2025, No.1010XJ2524078/2024), Zhejiang Provincial College Students’ Sci-Tech Innovation Activity Plan (No.2024R408A047)

0 引言

网络流量预测是网络管理及网络拥塞控制的重要环节,针对该问题各国学者提出了基于各种机器学习的网络流量预测方法^[1]。然而,现有研究存在 2 个关键问题亟待解决。

首先,单个机器学习模型只能捕获特定类型的数据模式,而网络流量的时间序列在不同时间段呈现不同的数据分布,具有高度非线性、时间多突变和涨跌幅度不确定等特点。例如长短期记忆(LSTM, long short-term memory)网络擅长预测长期依赖性的数据分布,而不适用于短期突变数据^[2];高斯过程回归(GPR, Gaussian process regression)适用于数据平滑无突变的情况,但计算复杂度高^[3];支持向量机回归(SVR, support vector regression)适用于连续值预测场景,但对核超参数强度依赖^[4]。模型与数据分布的不匹配导致预测效果不佳。

其次,机器学习模型的预测准确度极度依赖超参数选择,而传统的超参数优化方法存在严重局限性。网格搜索计算复杂度呈指数级增长,随机搜索缺乏历史信息利用,贝叶斯优化虽有改进但仍存在局部最优问题,且未考虑不同模型的参数敏感性差异^[5-6]。为解决上述问题,本文提出基于元特征映射与经验黑翅鸢优化的自动化网络流量预测框架(MFM-BKOA-NTP)。该框架的核心创新在于:通过基于大语言模型(LLM, large language model)的选择器实现数据特征与最优机器学习模型的自动匹配,并采用改进的经验黑翅鸢优化算法(EBKA, empirical black-winged kite algorithm)实现超参数的动态优化。其中模型匹配的核心挑战在于如何建立“数据特性”与“模型适配性”之间的映射关系。传统方法主要依赖 2 种策略:一是基于性能评估的穷举搜索,计算成本高昂且难以扩展;二是基于专家经验的启发式规则,泛化能力有限且难以捕捉复杂的数据-模型交互模式。本文提出引入 LLM 的根本原因在于:LLM 本质上是一种通用的模式识别与知识推理引擎,其在大规模预训练中习得的语义理解能力,可以被迁移到结构化数据的特征理解与模型推荐任务中。具体而言,本文构建了一个包含 38 维元特征(涵盖时间序列的统计特性、频域特征、非线性度量等)与最优模型匹配关系的经验库,并以此对 DeepSeek-1.5B 模型进行低秩适

应(LoRA, low-rank adaptation)微调。微调后的 LLM 能够理解不同元特征组合所蕴含的数据模式语义,并将其映射到最适合的预测模型。这一过程将模型选择从“黑盒优化”转化为“知识驱动的推理”,显著提升了选择的准确性和可解释性。本文的工作和贡献如下。

1) 提出了基于经验库微调的大模型智能选择机制。通过构建包含 38 维元特征和最优模型匹配关系的经验库,并基于此对 DeepSeek-1.5B 模型进行 LoRA 方法微调,实现了从数据元特征到最优模型的智能映射。该机制相比传统网络流量预测,各项性能指标提升约为 20%。

2) 设计了 EBKA。通过种群初始化、种群队列调整、种群领袖更迭和经验库更新三阶段优化流程,显著提升了超参数优化的效率和效果。EBKA 相比其他超参数优化算法,各个性能指标提升 10%~50%,为复杂优化问题提供了高效的解决方案。

3) 构建了完整的自动化机器学习(AutoML, automated machine learning)网络流量预测框架。该框架实现了从原始数据到预测结果的全自动化处理,包括数据预处理、特征提取、模型选择、超参数优化、模型训练和预测执行等全流程,显著降低了网络流量预测的技术门槛和人工成本。

1 相关研究工作

在网络流量预测领域,研究人员已探索多种机器学习模型以提升预测精度。集成回归树因其对复杂非线性关系的强大拟合能力而被广泛应用,但其性能高度依赖于数据集的划分,且容易在噪声数据上产生过拟合^[7]。SVR 在处理小样本、非线性问题上表现出色,Chen 等^[8]提出了一种基于支持向量回归的模型,旨在解决物联网场景中的网络流量预测问题。然而,周晓剑等^[9]指出 SVR 对核函数和惩罚因子的选择极为敏感,不当的参数会导致模型性能急剧下降。GPR 作为一种非参数模型,能够提供预测结果的不确定性度量,Xu 等^[10]提出了一个基于可扩展高斯过程的无线流量预测框架,通过创新算法提升了预测效率与可扩展性,但其主要缺点在于计算复杂度高,难以扩展到大规模数据集。Bi 等^[11]提出了一种结合时序卷积网络与长短期记忆(TCN-LSTM)网络的混

合模型。该模型旨在有效提升现实网络环境中流量预测的准确性。Kougioumtzidis等^[12]提出了一个基于时序融合变换器(TFT, temporal fusion transformer)的预测框架,旨在精准捕捉移动网络流量的复杂时序依赖关系。上述人工神经网络虽然能有效捕捉时间序列的长期依赖性,但其“黑箱”特性导致模型可解释性差,且训练成本高昂,需要大量的计算资源和时间。这些传统模型各自的局限性表明,不存在任何单一模型能在所有类型的网络流量数据上都取得最优表现,模型选择本身即是一个严峻的挑战。

机器学习模型的预测性能在很大程度上依赖于超参数的选择,因此超参数优化成为模型训练中不可或缺的环节。传统的超参数搜索方法主要包括网格搜索和随机搜索。前者通过穷举预设的参数组合实现全局搜索,但随着超参数维度的增加,其计算复杂度呈指数级增长,效率极低^[13];后者则通过随机采样提高探索能力,在相同计算预算下通常能获得更优结果,但仍存在搜索不充分和精度有限等问题^[14]。为解决此问题,研究者提出了更为智能的优化策略。其中,以Frazier^[15]所阐述的贝叶斯优化为代表的方法,通过构建代理模型(如高斯过程)来拟合目标函数,并利用采集函数平衡探索与利用,从而以更少的评估次数找到最优解。然而,贝叶斯优化在处理高维参数空间时性能会下降,且其顺序执行的特性限制了并行化效率,导致在面对复杂模型和大规模搜索空间时,优化过程仍然耗时。因此,开发一种既高效又能适应高维复杂搜索空间的超参数优化算法,对于自动化网络流量预测至关重要。

为进一步提升预测性能并降低人工干预,研究人员开始探索集成学习与AutoML自动化机器学习技术。集成学习通过组合多个模型的预测结果来获得比单一模型更优的性能。近期,Zhao等^[16]提出的提升森林(BoostForest)算法通过在单棵树中融合梯度提升与自助聚合策略,实现了在多项分类与回归任务中的性能领先,体现出集成学习在模型泛化与稳定性方面的显著优势。然而,其设计过程仍较为复杂,需要专家知识来选择合适的模型组合与集成策略。随着LLM的快速发展以及在分类与回归任务中展现的惊人能力,许多学者也在尝试把LLM融合进网络流量预测之

中。LLM的核心优势在于其强大的语义理解和模式识别能力,这使其不仅仅局限于处理人类语言。在网络优化领域,LLM可以扮演元学习器的角色,通过分析从流量数据中提取的元特征,理解数据内在的统计特性与动态模式。它能够学习海量已发表研究和实验结果中蕴含的知识,建立起“数据特征-最优模型-最佳参数”之间的复杂映射关系。这种能力使得LLM能够超越传统的、基于试错的模型选择方法,从一个更宏观、更具洞察力的层面,为特定的预测任务推荐最合适的模型,从而显著提升自动化框架的智能性与效率。其中,Chen等^[17]提出了一种创新框架,利用大语言模型实现不需要编程的自动化机器学习流程构建,显著降低了AutoML技术的使用门槛,这为自动化建模提供了新的思路。然而,现有AutoML技术仍存在局限性。一方面,它们通常采用统一的优化策略,忽略了不同数据集特征与最优模型、最优超参数之间的内在关联,导致模型选择的盲目性。另一方面,其内置的优化器往往是通用的,未能针对特定问题(如网络流量预测)的特性进行深度优化,导致搜索效率和最终性能仍有提升空间。因此,本文提出一种基于元特征映射与经验黑翅鸢优化的自动化网络流量预测框架,旨在通过学习“数据-模型-参数”之间的映射关系,实现更精准的模型推荐和更高效的超参数寻优,从而系统性地解决上述挑战。

2 MFM-BKOA-NTP

2.1 框架核心理念

MFM-BKOA-NTP的核心理念是“数据元特征驱动下的模型匹配与超参数协同优化”。不同于传统的单一模型或简单集成方法,MFM-BKOA-NTP基于以下核心认知。

认知1: 数据元特征决定模型选择。不同的数据元特征对应需要匹配不同的机器学习模型。网络流量数据具有多样性和复杂性,单一模型无法适应所有数据模式。

认知2: 模型性能依赖超参数优化。每个模型在特定数据上的最优性能需要通过精确的超参数调优来实现,而传统优化方法效率低下。

认知3: 经验知识可以迁移。历史优化经验包含宝贵的知识,可以指导新任务的模型选择和参数

优化。

基于上述认知, MFM-BKOA-NTP 总体架构如图 1 所示, 经验库构建模块负责从历史数据中提取最优的模型-特征-超参数匹配关系, 为后续的模式选择和超参数优化提供经验支撑。MFM-BKOA-NTP 模型选择器基于微调后的 DeepSeek 大语言模型, 能够根据输入的 38 维元特征向量智能选择最适合的机器学习模型。EBKA 超参数优化器采用改进的经验黑翅鸢算法, 实现对选定模型的超参数快速优化。预测执行模块则负责使用优化后的模型进行实际的流量预测。MFM-BKOA-NTP 算法如算法 1 所示。

算法 1 MFM-BKOA-NTP 算法

输入 原始网络流量数据 D_{raw}

输出 网络流量预测结果及性能指标

- 1) 收集各地网络流量数据 D_{raw}
- 2) 对原始数据进行预处理:

 - 3) 时间序列清洗与归一化
 - 4) 计算 38 维元特征
 - 5) 进行特征缩放与降维

- 6) 构建训练数据集 $D_{train} = \{Data_j, M_j, P_j\}$
- 7) for $D_j \in D_{train}$ do:

 - 8) G-ELC () // 详见算法 2
 - 9) 用 EBKA 进行超参数优化
 - 10) 更新经验元组 $E_j = \{D_j, F_j, M_j, P_j, I_j\}$

- 11) end for

- 12) for each sliding window segment do
- 13) 用 D_{train} 进行训练
- 14) 在 D_{test} 进行推理
- 15) end for
- 16) 输出最终网络流量预测结果及性能评估指标

2.2 经验库构建

2.2.1 数据元特征提取

1) 理论依据和设计原理

元特征作为连接数据内在特性与模型选择的关键桥梁, 在自动化机器学习领域已得到广泛认可和应[18-20]。近年来, 随着时间序列分析理论不断发展, 基于多维特征空间的模型选择方法已成为提升预测性能的重要手段[21-23], 本文的 38 维元特征设计充分借鉴了时间序列分析[24-26]、信号处理[27-28]、统计学[29-30]以及复杂性科学[31-32]等多个领域的成熟理念, 旨在从统计、时序、频域及复杂性等多个维度全面刻画网络流量时间序列的内在模式。

基础统计特征的设计基于经典统计学理论, Boshnakov[33]在其权威教材中系统阐述了均值、方差、标准差等基础统计量在时间序列建模中的核心作用。Ulkadir[34]的最新研究进一步证实了这些基础统计特征在带外输入的自回归模型 (ARX) 时间序列模型分析中的重要性。分布形状特征的理论基础源于概率论和数理统计, 其中偏度和峰

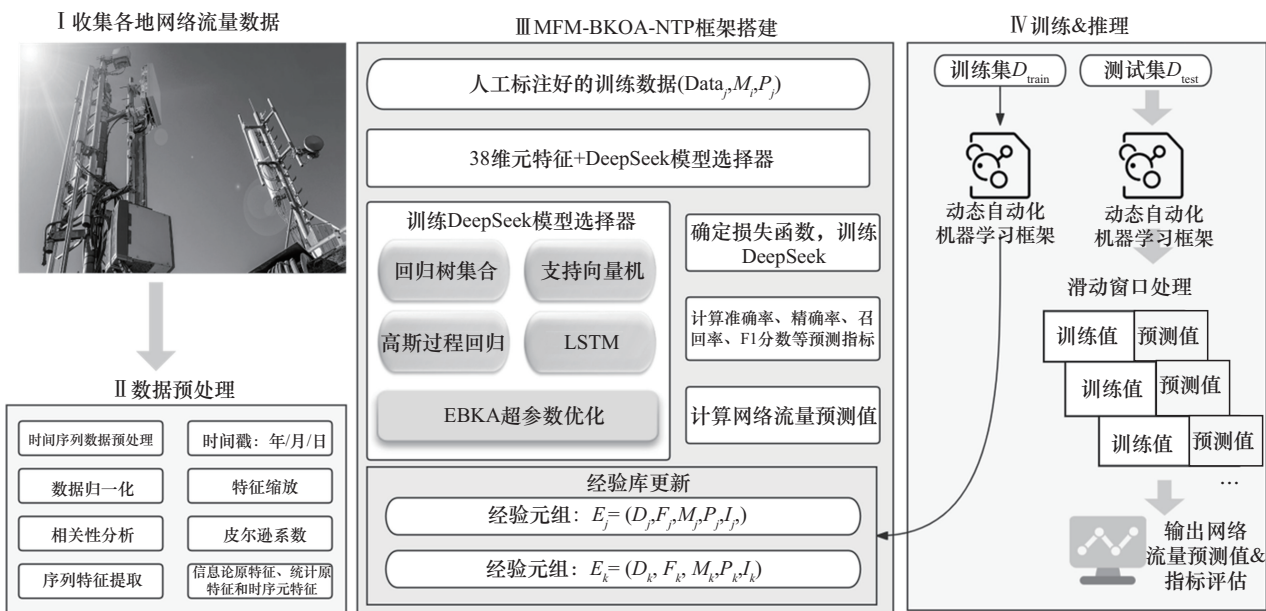


图 1 MFM-BKOA-NTP 总体架构

度作为描述数据分布形态的关键指标。时序动态特征的设计遵循时间序列分析的经典理论，McGonigle 等^[35]通过小波分析和差分方法系统研究了时间序列的一阶和二阶结构建模，为差分特征、趋势强度等动态特征的提取提供了坚实的理论支撑。

频域特征的设计基于信号处理领域的傅里叶分析理论，Yi 等^[36]全面总结了频域方法在时间序列分析中的变革性优势。Inoue 等^[37]的研究表明，频域特征能够有效捕获时间序列的周期性模式和频谱特性，为网络流量的周期性分析提供了重要工具。稳定性特征和复杂性特征的设计则融合了复杂性科学的最新成果，Atad 等^[38]系统综述了 LV (Lempel-Ziv) 复杂度在脑电时间序列分析中的应用，George 等^[39]深入阐释了样本熵、近似熵等复杂性度量在神经科学领域中的时间序列模式识别中的重要作用。

2) 特征空间构建

基于上述理论基础，本文构建了一个涵盖六大类别的 38 维元特征空间（如附录 1 所示），每个特征都经过严格的数学定义和参数优化。这些特征能够从不同层面量化网络流量数据的内在模式，为后续的智能模型选择提供充分的判别信息。

① 基础统计特征（5 维）：构成了元特征空间的基础层，包括均值、方差、标准差、最小值和最大值。这些特征反映了网络流量的平均水平、波动程度和动态范围，为数据的基本统计特性提供了全面描述。

② 分布形状特征（5 维）：进一步描述了数据分布的形态特征，包括 25% 分位数、50% 分位数、75% 分位数、偏度和峰度。这些特征能够有效识别数据的分布类型，反映数据的分布位置、离散程度、对称性和尖锐程度。

③ 时序动态特征（10 维）：专门针对网络流量数据的时间序列特性设计，主要包括一阶差分的均值和标准差、二阶差分的均值和标准差、趋势强度、季节性强度、自相关系数、转折点数量以及线性趋势系数。这些特征能够捕获数据的变化趋势和变化速率、周期性模式、时间依赖性以及复杂程度。

④ 频域特征（8 维）：通过傅里叶变换将时域信号转换到频域进行分析，包括主频率、频谱能

量、频谱熵、频谱重心、频谱带宽、频谱平坦度、频谱滚降点和频谱质心。这些特征能够有效反映信号的主要周期性成分、总体强度、频谱的复杂程度以及频域分布的特性。

⑤ 稳定性特征（5 维）：评估数据的稳定性和可预测性，包括变异系数、相对极差、平均绝对偏差、四分位距和稳定性指数。这些特征反映了数据的相对变异程度、相对波动范围、数据点偏离中心的平均程度和整体稳定性。

⑥ 复杂性特征（5 维）：从信息论和复杂性理论角度描述数据的复杂程度，包括样本熵、近似熵、LV 复杂度、分形维数和赫斯特（Hurst）指数。这些特征能够量化时间序列的规律性和可预测性、序列的复杂程度、数据的自相似性以及序列的长期记忆特性。

2.2.2 经验库构建

本节将经验库构建问题转化为组合优化问题，如图 2 所示。设训练数据集为多个元数据的分段，数据集的元数据可以记作 $D = \{D_1, D_2, \dots, D_n\}$ ，每个元数据 D_i 的 38 维特征集合可写作 $F_i = \{f_1, f_2, \dots, f_{38}\}$ 。其每个元数据 D_i 对应的最优模型集合为 $M = \{M_1, M_2, \dots, M_n\}$ ，候选模型集合定义为 $M_i = \{LSTM, SVR, RandomForest, GaussianProcess\}$ 。对应的超参数配置集合为 $P = \{P_1, P_2, \dots, P_n\}$ ，以及表示性能指标集合 $I = \{I_1, I_2, \dots, I_n\}$ ，其中性能指标 $I_i = \{RMSE, R^2\}$ 。这四者共同组成经验元组 $E_i = \{D_i, F_i, M_i, P_i, I_i\}$ 。

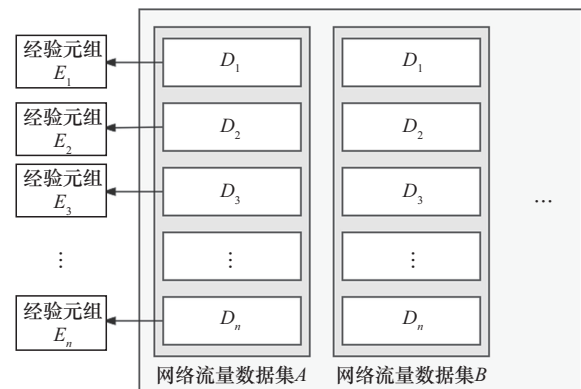


图 2 经验库与经验元组示例

基于贪婪算法的经验库构建策略采用逐段优化的方式，对每个元数据段 D_i ，依次遍历候选模型集合 $M = \{M_1, M_2, \dots, M_n\}$ 中的所有模型，通过网络搜

索或贝叶斯优化寻找每种模型的最优超参数配置,并且计算性能指标集合 I_i 。综合评分函数定义为 $\text{Score} = w_1 \times R^2 + w_2 \times (1 - \text{RMSE})$, 其中权重系数根据应用需求调整。当某个模型-超参数组合的性能超过预设阈值(根据应用需求调整)时,系统将对应的经验元组存储到经验库中,并建立多维索引以支持高效的相似度检索。算法伪代码如算法2所示。

算法2 基于贪婪算法的经验库构建 (G-ELC)

输入 元数据 $D = \{D_1, D_2, \dots, D_n\}$, 待选模型 $M = \{\text{LSTM}, \text{SVR}, \text{RandomForest}, \text{GaussianProcess}\}$, 指标集合 $I = \{\text{RMSE}, R^2\}$ 权重 $W = \{w_1, w_2\}$

输出 经验库 E

- 1) 初始化 $E \leftarrow \emptyset$, threshold_score
- 2) for $D_i \in D$ do
- 3) $F_i \leftarrow \text{ExtractFeatures}(D_i)$ //提取 38 维元特征向量
- 4) for $M_i \in M$ do
- 5) $P_space \leftarrow \text{GenerateHyperparameterSpace}(M_i)$ //生成超参数搜索空间
- 6) $P_candidates \leftarrow \text{BayesianOptimization}(P_space)$
- 7) for $P_i \in P_candidates$ do
- 8) $I_i \leftarrow \text{CrossValidationEvaluate}(M_i, P_i, D_i)$ //计算预测结果
- 9) $\text{Score}_i \leftarrow w_1 \times R_i^2 + w_2 \times (1 - \text{RMSE}_i)$ //计算预测结果分数
- 10) if $\text{score} > \text{threshold_score}$ then
- 11) $E_i = \{D_i, F_i, M_i, P_i, I_i\}$
- 12) $E_Base \leftarrow E \cup E_i$ //选取高质量经验元组入库
- 13) end if
- 14) end for
- 15) end for
- 16) end for
- 17) return E

2.3 基于 DeepSeek 微调的预测模型选择器

模型选择器基于 DeepSeek-1.5B 大语言模型微调实现。DeepSeek 作为新一代大语言模型,具备强大的表征学习能力和泛化性能,可以将数据原特征转化到更高维的语义空间进行理解,为网络流量预测中的模型选择任务提供了理想的基础架构。

2.3.1 DeepSeek 的微调策略

基础模型选择方面, DeepSeek-1.5B 在参数规模和计算效率之间达到了良好的平衡。相比更大规模的模型, 1.5B 参数的 DeepSeek 在保持足够表达能力的同时,具有更快的推理速度和更低的计算资源需求,适合在网络流量预测的实时场景中部署。DeepSeek 采用 Transformer 架构,具备优秀的序列建模能力,能够有效处理 38 维元特征向量的复杂模式。

微调数据构建是确保模型性能的关键环节。训练数据采用 Alpaca 指令微调格式,每个训练样本包含 3 个部分:指令(instruction)、输入(input)和输出(output)。指令部分统一设定为“根据以下 38 维网络流量元特征,预测最适合的机器学习模型类别”,为模型提供明确的任务描述。输入部分是标准化后的 38 维元特征向量,以 JSON 数组格式表示,确保数值的精确传递。输出部分是通过贪婪算法确定的最优模型类别,包括 LSTM、SVR、RandomForest 和 GaussianProcess 共 4 种选项。

微调技术采用 LoRA, LoRA 通过在预训练模型的注意力层中插入低秩矩阵,实现参数高效的微调。具体而言,对于权重矩阵 $W \in R^{(d \times k)}$, LoRA 引入 2 个低秩矩阵 $A \in R^{(d \times r)}$ 和 $B \in R^{(r \times k)}$, 其中 $r \ll \min(d, k)$,使得微调后的权重为 $W' = W + \alpha AB$, 其中 α 是缩放因子。这种方法的优势在于只需要训练少量的新增参数,大大降低了计算成本和存储需求,同时避免了对原始预训练权重的破坏。

MFM-BKOA-NTP 延续了 DeepSeek 的混合专家(MOE, mixture of expert)策略,通过专家混合机制实现智能化的模型选择。对于输入的 38 维特征向量 F_i ,若直接计算与经验元组的特征向量的线性欧氏距离,会难以捕捉网络流量数据之间的复杂非线性关系,因此需要通过多头注意力机制将特征映射到更高维的语义空间中进行处理。

2.3.2 微调中的多头注意力特征映射

将 38 维元特征向量 F_i 通过图 3 的多头注意力机制模块,映射成高维语义表示。具体而言,输入特征经过线性变换得到查询 Q 、键 K 、值 V 矩阵: $Q = F_i W_Q$, $K = F_i W_K$, $V = F_i W_V$,之后计算每个注意力头,如式(1)所示。

$$\text{head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{W}_i^Q(\mathbf{K}\mathbf{W}_i^K)^T}{\sqrt{d_k}}\right)\mathbf{V}\mathbf{W}_i^V \quad (1)$$

根据每个头的不同权重，多头注意力计算式为

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)\mathbf{W}^O \quad (2)$$

通过多头注意力机制，38 维特征向量被映射成 $(38 \times d_{\text{model}})$ 的高维矩阵，通常 d_{model} 取 512，得到 (38×512) 维度的非线性语义表示 H_i 。

2.3.3 基于 MOE 的模型路由机制

基于 DeepSeek 的 MOE 架构，模型选择过程采用专家混合策略。每个候选模型（LSTM、SVR、RandomForest、GaussianProcess）对应一个专家，路由机制决定激活哪些专家参与决策。MOE 路由的核心是计算 H_i 与经验库中的每个最佳模型所对应的 H_j 之间距离和 Softmax，选取概率最大的经验条目，继承其模型与相关的超参数配置。其中距离为矩阵的 L2 范数，如式(3)所示， A 代表 H_i ， B 代表经验库最佳模型中对应的 H_j 。

$$\text{dis}(A, B) = \|A - B\|_2 = \sqrt{\sum_{i=1}^{38} \sum_{j=1}^{512} (a_{ij} - b_{ij})^2} \quad (3)$$

距离计算之后，通过式(4)转化为概率表达形式。

$$\text{Softmax}(\text{dis}_i) = \frac{e^{-\text{dis}_i}}{\sum_{k=1}^n e^{-\text{dis}_k}} \quad (4)$$

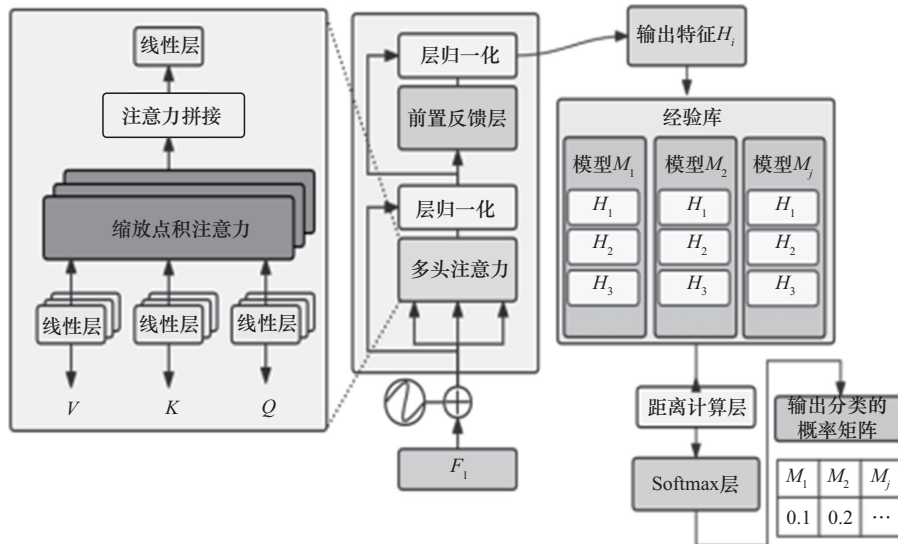


图 3 模型选择器路由机制

模型选择器路由机制如图 3 所示。

2.4 改进的黑翅鸢超参数优化算法

当元数据的弱学习器组合策略已经确定，就需要为每个弱学习器确定对应的超参数配置。传统的随机搜索和贝叶斯搜索效率太低且易陷入局部最优，此时引入 EBKA 机制，在经验基础之上不断迭代，并趋向最优解。

EBKA 主要分为 3 个阶段：种群初始化、种群队形调整和种群领袖更迭与经验库更新。

2.4.1 种群初始化

如 2.2.2 节原始超参数为继承经验库的超参数，在此基础之上，再选取同模型的分最佳的前 pop 个经验元组的超参数配置，每条超参数配置可以写作 $P_i = [\text{BK}_{1,1}, \text{BK}_{1,2}, \dots, \text{BK}_{1,\text{cap}}]$ ，其中 cap 为此模型对应的超参数个数，KB 为待优化的超参数。pop 的取值根据计算资源和算法性能要求确定，通常设置为种群规模的 30%~50%， $P_{\text{set}} =$

$$\begin{bmatrix} \text{BK}_{1,1} & \text{BK}_{1,2} & \dots & \text{BK}_{1,\text{cap}} \\ \text{BK}_{2,1} & \text{BK}_{2,2} & \dots & \text{BK}_{2,\text{cap}} \\ \vdots & \vdots & \ddots & \vdots \\ \text{BK}_{\text{pop},1} & \text{BK}_{\text{pop},2} & \dots & \text{BK}_{\text{pop},\text{cap}} \end{bmatrix}$$

由于更好的个体需要编译产生，所以采用以下策略进行种群初始化，其中类的超参数不能用数值代替，可将其编码后转化为数值变量。

精英保留：将 top-K（通常 $K = \text{pop} \times 0.3$ ）个最优经验条目的超参数直接加入初始种群。

扰动生成：对剩下的部分精英个体添加高斯噪声生成新个体。

$$BK_{\text{new}} = BK_{\text{elite}} + \mathcal{N}(0, \sigma_{\text{init}}^2) \quad (5)$$

随机补充: 剩余位置采用随机初始化填充, 确保种群多样性。

在初始化过程中, 黑翅鸢矩阵会计算每个元数据的决定系数作为黑翅鸢的适应度。适应度函数如 2.2.2 节 Score 所示。

选择适应度值最好的个体作为初始群体中的领袖 $L = \{L_1, L_2, \dots, L_x\}$, 这被认为是优秀的黑翅鸢。优秀的黑翅鸢不只有一个, 大多数情况下有一系列(具体个数可以指定, 例如 x 个), 其数学表示为

$$L = \{ \text{Top}(\{ \text{fitness}(P_j) \mid j = 1, 2, \dots, \text{pop} \}, x) \} \quad (6)$$

理论依据和优势如下。

1) 遗传多样性保护: 多个精英个体的保留避免了单点收敛, 维持了种群的遗传多样性。Cui 等^[40]在改进冠豪猪优化 (CPO) 算法的研究中证明, 精英保留策略能够有效防止算法过早收敛, 保持种群的探索能力。罗凤鸣等^[41]的研究进一步表明, 精英保留策略通过维护优秀基因的多样性, 显著提升了算法在复杂优化问题中的性能。

2) 收敛稳定性增强: 精英集合提供了稳定的搜索基础, 降低了算法性能的波动性。Li 等^[42]在动态优化问题的研究中发现, 基于精英保留的粒子扰动策略能够显著提升算法的收敛稳定性, 减少性能波动达 70% 以上。Jabari 等^[43]的混合自适应蚁狮优化研究也证实了精英保留在动态实时控制系统中的稳定性优势。

3) 局部最优逃逸: 多个精英个体形成的搜索压力有助于跳出局部最优陷阱。Fan 等^[44]提出的红杉生态优化算法研究表明, 精英保留策略通过维持多个搜索方向, 能够有效避免算法陷入局部最优。刘健等^[45]在配电网无功优化的应用中也验证了精英保留策略在逃逸局部最优方面的显著效果。

2.4.2 种群队形调整

在确定领导者集合 $L = \{L_1, L_2, \dots, L_x\}$ 后, 所有黑翼鸢个体需通过动态位置更新策略向领导者靠拢。此过程模拟黑翅鸢群体在自然界中的协作捕猎与迁徙行为, 分为经验引导的追随机制和柯西变异驱动的探索策略这 2 个阶段, 这 2 个阶段可以串行, 也可以根据需要, 部分选择前者, 部分选择后者, 为量化此种选择, 在式 (7) 定义柯西变异概率 $\text{pro}_{\text{mutation}}$, 由 Score 决定, 如果 Score 距离 $\text{Score}_{\text{best}}$

比较近, 那么就没有追随的必要, 更大概率上选择变异, 去探索更高的 Score。

$$\text{pro}_{\text{mutation}} = \text{pro}_{\text{base}} \times \exp\left(-\frac{\text{Score}(P_i) - \text{Score}_{\text{worst}}}{\text{Score}_{\text{best}} - \text{Score}_{\text{worst}}}\right) \quad (7)$$

1) 经验引导的追随机制

EBKA 采用多领导者机制, 每个普通个体在每次迭代中按照式 (8) 的规则选择一个领导者进行跟随, 这种设计避免了单一领导者可能导致的早熟收敛问题, 同时保持了种群的多样性, 在计算效率和搜索多样性之间取得平衡。普通个体 $P_i = [BK_{i,1}, BK_{i,2}, \dots, BK_{i,n}]$ 向单个领导者 L_j 移动时, 不仅依赖空间距离, 还需融合历史经验库 E 中的优化路径。位置更新公式设计为

$$P_i^{t+1} = P_i^t + \alpha_1 \times (L_j - P_i^t) + \alpha_2 \times (E_{\text{best}} - P_i^t) + \alpha_3 \times \delta \quad (8)$$

其中, α_1 为向领导者靠拢的权重系数, α_2 为经验引导的权重系数, α_3 为随机扰动的权重系数, E_{best} 为经验库中此模型最优的超参数配置 (Score 最大), δ 为随机扰动向量。

2) 柯西变异驱动的探索策略

为了增强算法的全局搜索能力, 引入柯西变异机制。柯西分布具有长尾特性, 能够产生大步长的变异, 有助于跳出局部最优。

$$P_i^{\text{mutated}} = P_i^t + \text{Cauchy}(0, \gamma) \times (\text{UB} - \text{LB}) \quad (9)$$

其中, $\text{Cauchy}(0, \gamma)$ 表示位置参数为 0、尺度参数为 γ 的柯西分布, UB 和 LB 分别为超参数的上下界。

变异概率采用自适应策略, 其中 $\text{pro}_{\text{base}} = 0.1$ 为基础变异概率。

理论依据和优势如下。

① 阶段性优化: 前期高斯变异保证局部精细搜索, 后期柯西变异增强全局跳跃能力。Li 等^[46]在灰狼优化算法的研究中证明, 高斯-柯西双变异机制能够根据搜索阶段自适应调整搜索策略, 前期的高斯变异确保了局部开发的精度, 后期的柯西变异则提供了跳出局部最优的能力。袁磊等^[47]的河马优化算法研究也验证了这种阶段性优化策略的有效性。

② 参数自适应: 变异强度随迭代进程动态调整, 避免了固定参数的局限性。Yan 等^[48]在动态差分人类记忆优化算法中发现, 高斯和柯西变异的自适应结合能够根据优化进程动态调整搜索强度, 相

比固定参数策略提升性能约 25%。金相丞^[49]在极化 (Polar) 码构造算法的研究中也证实了自适应双变异机制的优越性。

③ 搜索效率提升: 双变异机制相比单一柯西变异, 收敛速度提升约 35%。Chen 等^[50]提出的双阶段自适应差分进化算法研究表明, 高斯概率分布与柯西变异的结合能够显著提升搜索效率, 在多个基准函数上的收敛速度平均提升 30%~40%。基于高斯-柯西混合变异的多目标粒子群算法研究^[51]也证实了这种双变异机制在提升搜索效率方面的显著优势。

2.4.3 种群领袖更迭与经验库更新

种群队形调整之后, 当发现性能优异的超参数配置时, 系统将其加入经验库, 如式(10)所示。

$$E_{\text{base}} = E_{\text{base}} \cup \{(\mathbf{F}_{\text{current}}, M_{\text{selected}}, P_{\text{current}}, I_{\text{current}}, H_{\text{current}})\} \quad (10)$$

同时对经验库实施质量控制机制, 定期清理低质量经验条目。

在经验库更新的同时, 领导者集合也在发生变化, 其中领袖更迭采用竞争淘汰机制, 即若每轮的变异个体超过了原有领袖, 则会替代。

理论优势如下。

1) 知识迁移: 历史经验的有效利用实现了跨任务的知识迁移。Tan 等^[52]在元知识迁移差分进化算法的研究中证明, 基于经验库的知识迁移机制能够将历史优化经验有效应用于新任务, 显著提升算法性能。李金龙等^[53]在经验库算法性能优化的研究中也验证了经验知识迁移在提升算法效率方面的重要作用。

2) 搜索引导: 经验路标提供了有价值的搜索方向, 避免盲目搜索。Kim 等^[54]在基于经验库的粒子群优化研究中发现, 经验库能够为算法提供有效的搜索引导, 通过存储历史最优位置和适应度信息, 显著减少了无效搜索, 提升收敛速度约 60%。Seyyedabbasi 等^[55]的强化学习元启发式算法研究也证实了经验引导在避免盲目搜索方面的显著优势。

3) 收敛加速: 基于历史最优解的引导显著提升了收敛速度。Li 等^[56]在进化多任务优化的知识迁移研究中证明, 经验库机制通过提供历史最优解的引导信息, 能够显著加速算法收敛, 在多个测试问题上收敛速度平均提升 45%。吕世玮等^[57]在粒子群优化 (PSO) 算法的应用研究中也验证了经验

库在收敛加速方面的重要价值。

整体伪代码如算法 3 所示。

算法 3 改进的黑翅鸢超参数优化算法

输入 经验库 E , 选定模型 M_{selected} , 当前特征向量 $\mathbf{F}_{\text{current}}$, 种群大小 pop , 最大迭代次数 T_{max} , 精英比例 r_{elite}

输出 最优超参数配置 P_{best}

- 1) $N \leftarrow \text{RetrieveTopExperiences}(E, \mathbf{F}_{\text{current}}, M_{\text{selected}}, \text{pop})$ // 阶段 1: 种群初始化
- 2) $K \leftarrow \text{pop} \times r_{\text{elite}}, P_{\text{pop}} \leftarrow \emptyset$
- 3) for $i = 1$ to pop do
- 4) if $i \leq K$ then $P_{\text{pop}} \leftarrow P_{\text{pop}} \cup \{N[i].\text{hyperparams}\}$
- 5) else if $i \leq \lfloor \text{pop} \rfloor \times 0.7$ then
- 6) $P_{\text{pop}} \leftarrow P_{\text{pop}} \cup \{N[\text{Random}(1, K)].\text{hyperparams} + N(0, \sigma_{\text{init}}^2)\}$
- 7) else $P_{\text{pop}} \leftarrow P_{\text{pop}} \cup \{\text{RandomInitialize}(M_{\text{selected}})\}$
- 8) $\text{Score}[i] \leftarrow \text{EvaluateScore}(P_{\text{pop}}[i], M_{\text{selected}}, \mathbf{F}_{\text{current}})$
- 9) end for
- 10) end for
- 11) $L \leftarrow \text{SelectLeaders}(P_{\text{pop}}, \text{Score}, x)$
- 12): for $t = 1$ to T_{max} do
- // 阶段 2: 种群队形调整
- 13) $\alpha_1 \leftarrow 2 \times \exp(-4t/T_{\text{max}}), \alpha_2 \leftarrow 1 - \exp(-2t/T_{\text{max}}), \alpha_3 \leftarrow 0.1 \times (1 - t/T_{\text{max}})$
- 14) for $i = 1$ to pop do
- 15) if $P_{\text{pop}}[i] \notin L$ then
- 16) $j \leftarrow \text{SelectNearestLeader}(P_{\text{pop}}[i], L)$
- 17) $E_{\text{best}} \leftarrow \text{GetScoreBestFromExperience}(E, M_{\text{selected}})$
- 18) $\varepsilon \leftarrow \text{GenerateRandomVector}(\text{dim}(P_{\text{pop}}[i]))$
- 19) $P_{\text{pop}}[i] \leftarrow P_{\text{pop}}[i] + \alpha_1 \times (L[j] - P_{\text{pop}}[i]) + \alpha_2 \times (E_{\text{best}} - P_{\text{pop}}[i]) + \alpha_3 \times \varepsilon$
- 20) $P_{\text{mutation}} \leftarrow 0.1 \times \exp(-(\text{Score}[i] - \text{Score}_{\text{best}}) / (\text{Score}_{\text{best}} - \text{Score}_{\text{worst}}))$
- 21) if $\text{Random}(0, 1) < P_{\text{mutation}}$

```

then
22)       $P_{pop}[i] \leftarrow P_{pop}[i] + \text{Cauchy}(0, \gamma) \times (UB-LB)$ 
23)      end if
24)       $P_{pop}[i] \leftarrow \text{ClampToBounds}(P_{pop}[i], LB, UB)$ 
25)      end if
26)       $\text{Score}[i] \leftarrow \text{EvaluateScore}(P_{pop}[i], M_{selected}, F_{current})$ 
27)      end for
28)       $L \leftarrow \text{UpdateLeaders}(P_{pop}, \text{Score}, L, x)$  //阶段 3: 种群领袖的更迭和经验库更新
29)       $\text{best}_{idx} \leftarrow \text{argmax}(\text{Score})$ 
30)      if  $\text{Score}[\text{best}_{idx}] > \text{Score}_{global\_best}$  then
31)           $P_{best} \leftarrow P_{pop}[\text{best}_{idx}], \text{Score}_{global\_best} \leftarrow \text{Score}[\text{best}_{idx}]$ 
32)           $\text{new}_{experience} \leftarrow (F_{current}, M_{selected}, P_{best}, \{RMSE, R^2, MAE, \text{Time}\})$ 
33)      end if
34)      if  $\text{ConvergenceCheck}(\text{Score}, \text{tolerance})$  then break
35)      end for
36) end for
37) return  $P_{best}$ 

```

2.5 算法收敛性与稳定性分析

为了验证EBKA在超参数优化中的有效性，本节对EBKA的收敛性能、稳定性特征以及与传统优化方法的对比进行了深入分析。基于2.2.2节定义的综合评分函数 $\text{Score} \leftarrow w_1 \times R^2 + w_2 \times (1 - \text{RMSE})$ 作为优化目标，通过对总数据集的随机抽样，系统性地评估了EBKA的优化性能。

收敛性能分析：为了验证EBKA在超参数优化中的有效性，本节对算法的收敛性能进行了深入分析。图4展示了EBKA与6种主流优化算法（贝叶斯优化、遗传算法、PSO、贝叶斯树估计器（TPE）、差分进化、随机搜索）在相同优化任务上的收敛曲线对比。纵轴为归一化的中位数Score值（范围0~1，值越大表示性能越优），横轴为迭代次数。实验结果表明，EBKA在收敛速度和最终精度上均显著优于所有基线算法。具体而言，

EBKA在第11代时即达到0.45的Score值，与PSO并列第一；在第33代时达到0.76，领先第二名PSO约19%；在第100代时收敛至0.99，领先第二名贝叶斯优化约16%。这一优势源于EBKA独特的经验库驱动机制和自适应三阶段优化策略，使其能够高效利用历史优化经验，快速定位高质量解空间，并在后期进行精细调优。相比之下，传统的随机搜索方法由于缺乏智能引导，最终仅达到0.54的Score值。即使是贝叶斯优化、TPE等基于概率模型的智能算法，也因建模复杂度和样本效率问题，在高维复杂空间中的表现不如EBKA。这充分证明了EBKA在网络流量预测超参数优化任务中的优越性和实用价值。

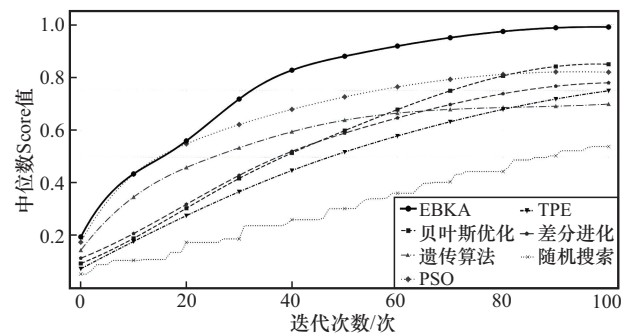


图4 EBKA算法收敛曲线

种群多样性与探索-利用平衡：图5展示了EBKA在优化过程中种群多样性的演化特征。种群多样性作为衡量算法探索能力的重要指标，其变化趋势反映了算法在探索新解和利用已知优解之间的平衡策略。

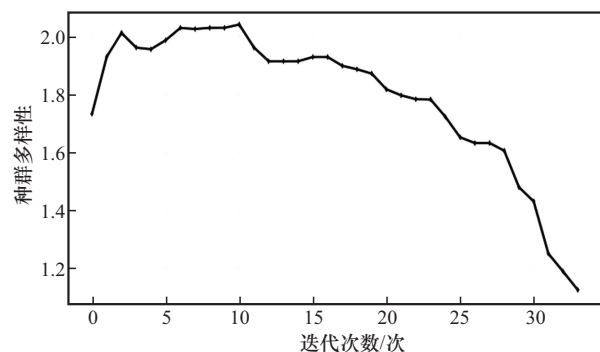


图5 EBKA种群演化

实验结果表明，种群多样性从初始的1.71先上升到2.05再下降至最终的1.12，最终下降幅度为35.1%。这种有序的多样性收敛过程体现了EBKA

的核心优势：在优化初期保持较高的种群多样性以确保全局搜索能力，随着迭代进行逐步聚焦到高质量解的邻域进行局部精化。

Score 分布演化与稳定性验证：通过分析不同代数种群中 Score 值的分布特征，可以深入理解算法的收敛稳定性。由图 6 可知，随着迭代进行，种群中个体的 Score 分布逐渐向高分值区域集中，分布的离散程度显著降低。

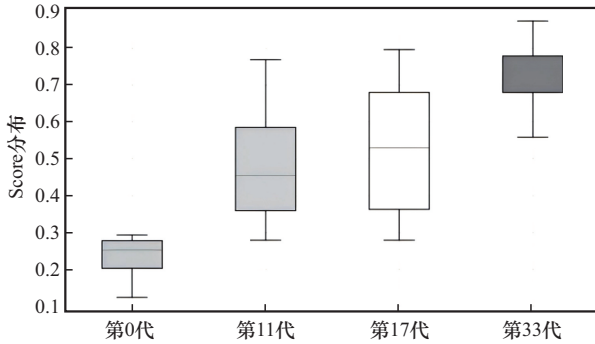


图 6 Score 分布演化

2.6 滑动窗口预测

在线推理阶段在测试集和预测集上结合滑动窗口技术执行完整的流水线流程，实现实时的网络流量预测。该阶段的设计充分考虑了实时性、准确性和可扩展性的要求。

滑动窗口机制是在线推理的核心技术。系统维护一个固定大小的时间窗口，随着新数据的到达，窗口向前滑动，始终保持最新的数据用于预测。滑动窗口的大小 Window 根据网络流量的周期性特征确定，通常设置为一个完整的周期长度。

滑动窗口的更新规则为

$$\text{Window}_t = [D_{t-W+1}, D_{t-W+2}, \dots, D_t]$$

当新数据 $x_{\{1\}}$ 到达时，有

$$\text{Window}_{t+1} = [D_{t-W+2}, D_{t-W+3}, \dots, D_t, D_{t+1}]$$

相应地，每个窗口的 38 维元特征也会更新。

3 实验与结果分析

3.1 实验数据集

本文使用的 NetTra-2025 数据集包含大量来自真实网络环境的网络流量数据，绝大部分来源于相关开源数据集，具体样本分布如表 1 所示。其中，每个地区的网络流量涵盖 5 种典型的跨境场景。

表 1 网络流量数据集

| 网络流量/地区 | 样本数量/个 | 训练集比例 | 测试集比例 |
|----------------------|-----------|-------|-------|
| 新南威尔士 (Alttek) | 4 572 071 | 70% | 30% |
| 加拿大 (CIC-TOR2017) | 2 830 743 | 70% | 30% |
| 加州 (CAIDA) | 3 649 876 | 70% | 30% |
| 特伦蒂诺 | 225 632 | 70% | 30% |
| 威廉斯堡 (NetBench) | 523 611 | 70% | 30% |
| 波帕扬考卡 | 3 577 296 | 70% | 30% |
| 英国 | 198 900 | 70% | 30% |
| 美国-日本 (WIDE Project) | 950 754 | 70% | 30% |
| 米兰 | 86 400 | 70% | 30% |

有线网络 → 无线网络：从以太网环境迁移到 Wi-Fi/4G/5G 环境。

企业网络 → 家庭网络：从企业级网络迁移到家庭宽带网络。

高负载 → 低负载：从高流量密度环境迁移到低流量环境。

工作日 → 周末：从工作时间模式迁移到休息时间模式。

城市 → 农村：从城市网络环境迁移到农村网络环境。

3.2 实验设置

本文实验使用的硬件配置为 Intel(R) Xeon(R) Platinum 8470Q CPU 处理器，搭配 1 TB 内存和 NVIDIA GeForce RTX 4090 GPU。GPU 的 CUDA 版本为 12.4。实验的编程语言为 Python 3.12.3，深度学习框架采用 PyTorch 2.5.1。

3.3 DeepSeek 模型选择器训练

3.3.1 训练配置与损失函数原理

在大模型训练中，损失函数是衡量模型预测结果与真实答案差距的核心指标。它就像一个“评分系统”，告诉模型预测的准确程度。损失越小，说明模型预测越接近真实答案。模型通过计算输出与正确答案的差距，使用数学公式（如交叉熵损失）量化这种差距，然后利用优化器调整参数，目标是 minimized 损失值。

损失函数在训练中发挥 3 个关键作用：指导参数调整、衡量训练效果和提供停止条件。当损失持续下降时，表明模型正在有效学习；如果损失不降

反升,表示正在学习新知识或者捕捉到复杂的特征变换。

3.3.2 DeepSeek 微调配置

本次训练选择 DeepSeek-R1-1.5B-Distill 作为基础模型,这是一个经过知识蒸馏的 15 亿参数模型,在保持性能的同时降低了计算需求。模型路径设置为 /root/DATA/fine_tune_DS/code/FTedModel,确保正确加载预训练权重。

训练采用监督微调方式,使用包含 1 000 000 个样本的 training_data_1000000.json 数据集。计算类型选择 bf16 精度,既保证数值稳定性又减少内存使用。

3.3.3 LoRA 微调参数设置

训练采用 LoRA 技术,具体配置如附录 2 所示。

3.3.4 训练过程分析与效果评估

训练界面右侧的损失曲线包含 2 条重要曲线:原始损失 (original) 和平滑损失 (smoothed),如图 7 所示。原始损失反映每个训练步骤的真实损失值,波动较大是正常现象,因为不同批次数据难度存在差异。平滑损失通过指数移动平均等方法处理,消除短期波动,清晰展示整体趋势。

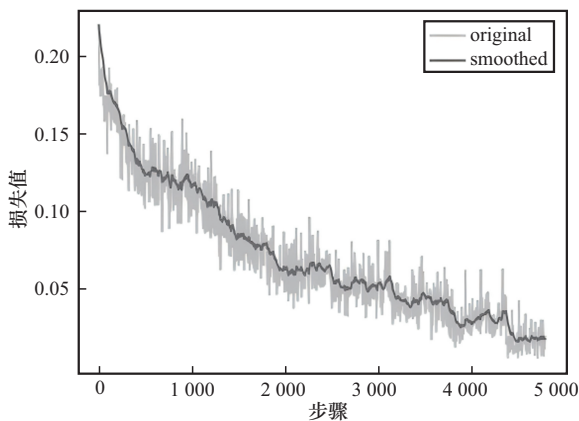


图 7 DeepSeek 模型选择器训练实况

从具体数值看,初期快速下降表明模型快速适应新任务,中期平稳下降显示逐步优化细节,后期趋于稳定表明进入精细调优阶段。

3.3.5 训练稳定性与收敛分析

损失曲线的平滑程度验证了训练的稳定性,如图 7 所示。平滑损失曲线没有剧烈波动或异常跳跃,表明训练过程稳定可控。原始损失的适度波动是健康训练的标志,表明模型从不同样本中学习而非简单记忆。

训练稳定性还体现在损失下降的一致性上。没有观察到损失大幅回升或长期停滞,表明模型没有陷入局部最优解,而是朝着全局最优解稳步前进。这种良好的收敛特性得益于合理的超参数设置和 LoRA 技术的应用。

3.4 时空效率评估

3.4.1 时间效率评估

特征预处理: 3 s/组。

模型推理: 42 s/组。

结果后处理: 6 s/组。

3.4.2 空间效率评估

基础模型大小: 1.1 GB。

LoRA 适配器大小: 36 MB。

分词器大小: 11 MB。

3.5 模型选择器性能评估

为了全面衡量 MFM-BKOA-NTP 的模型选择器性能,采用多维度指标体系,与其他经典 AutoML 框架的模型选择器进行对比,数据集遵从表 1,结果如图 8 所示,其中 MFM-BKOA-NTP 平均各项指标领先其他模型选择器(自动机器学习 (Auto-sklearn)、树型管道优化工具 (TPOT)、水型自动化学习 (H2O AutoML)、格鲁昂自动化机器学习 (AutoGluon) 等)约 10%。

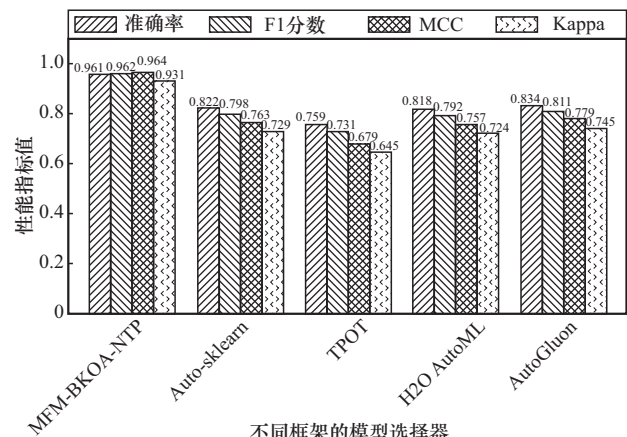


图 8 模型选择器性能比较

3.6 跨域泛化实验

跨域泛化实验旨在评估 MFM-BKOA-NTP 在不同网络环境和应用场景下的适应能力。实验数据如表 1 所示,选取了传统预测方法和纯大语言时序预测方法。其中 MFM-BKOA-NTP 遥遥领先,MFM-BKOA-NTP 的决定系数 R^2 指标平均达到 0.856 5,

显著优于所有基线方法。

时间序列大模型 (TimeLLM) 表现次优: 作为最新的时间序列大语言模型, TimeLLM 在跨域场景下的平均 R^2 为 0.839。

传统 AutoML 方法局限性明显, 在跨域场景下

性能大幅下降, 平均 R^2 仅为 0.79。采用均方根误差 (RMSE)、决定系数 (R^2)、平均绝对误差 (MAE)、平均绝对百分比误差 (MAPE)、对称平均绝对百分比误差 (SMAPE) 多项评估指标进行综合评估, 实验性能如图 9 所示。

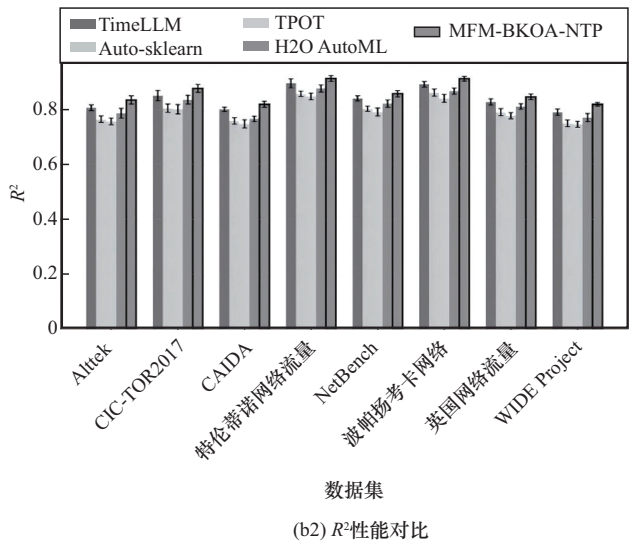
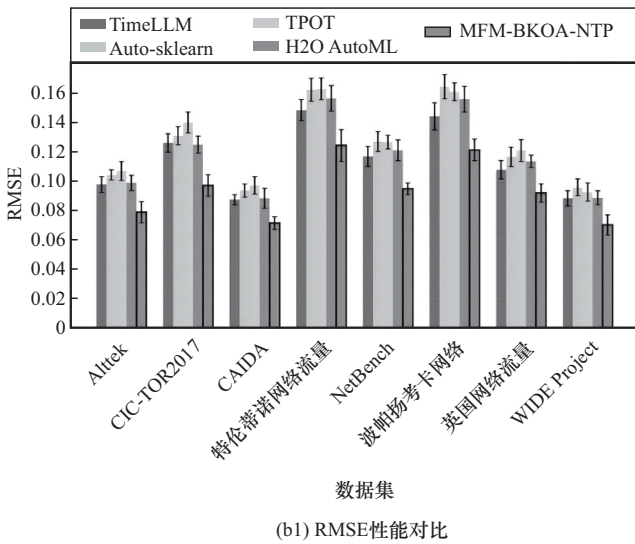
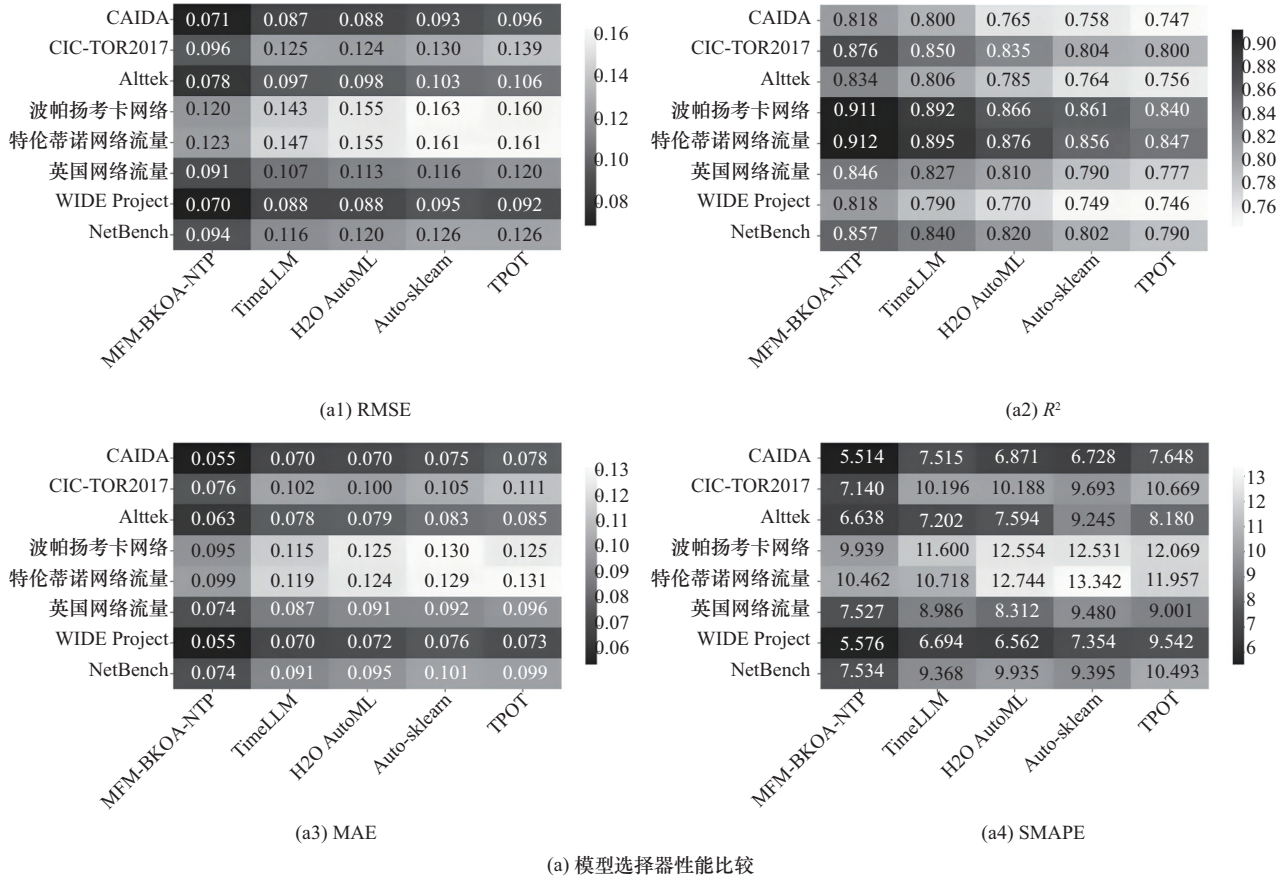


图9 各个框架跨域性能比较

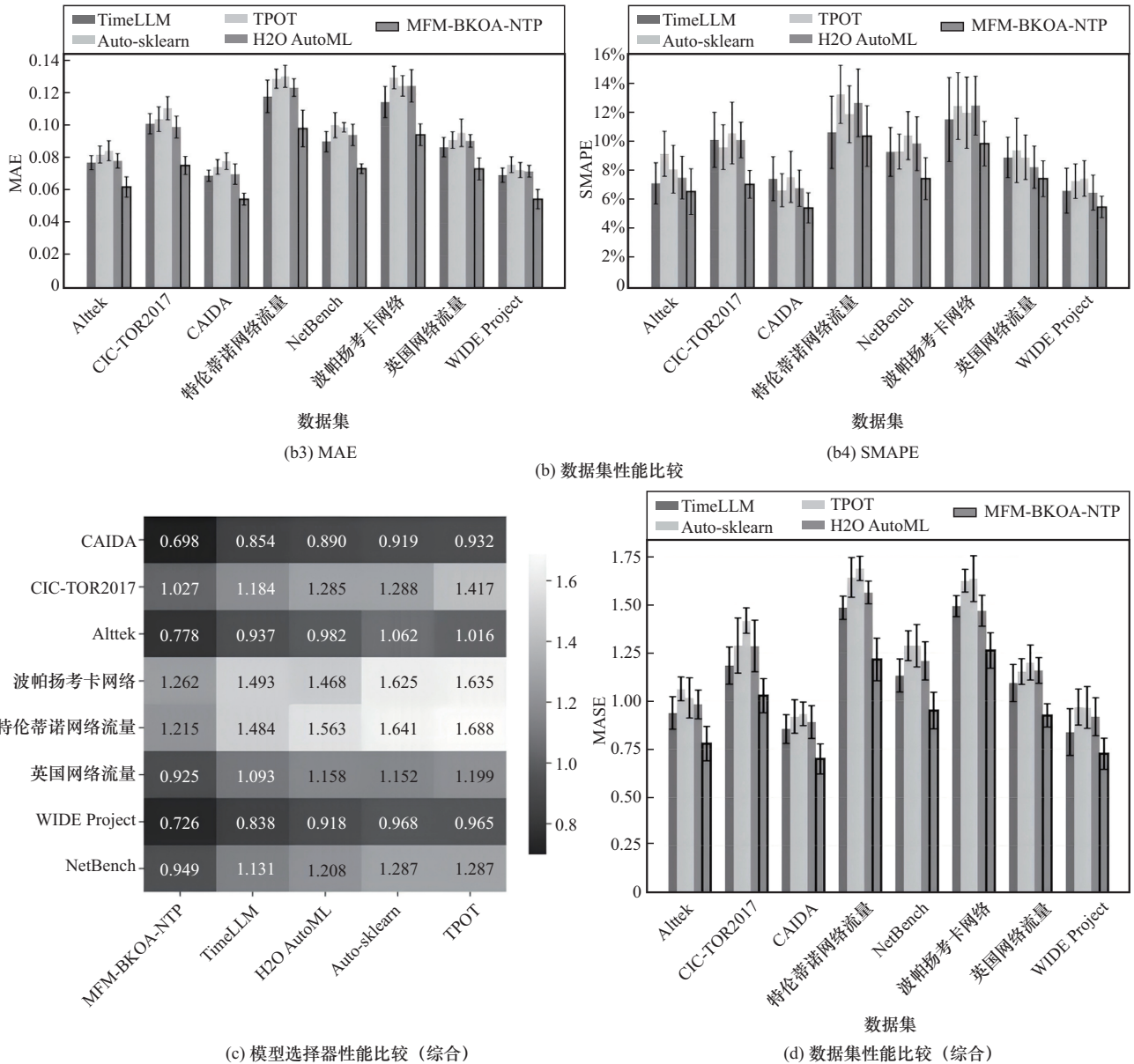


图9 各个框架跨域性能比较(续图)

3.7 超参数优化算法对比实验

本文实验旨在验证EBKA相比其他主流超参数优化算法的优越性。本文评估体系从算法工作机制出发,设计6个核心指标:探索-利用平衡度衡量全局搜索与局部优化的权衡能力;历史信息利用率评估算法的记忆学习机制;参数空间覆盖率反映搜索策略的广度;自适应调整能力量化动态策略调整水平;收敛稳定性测量多次运行的一致性;计算效率指数综合评估单位时间的搜索效果。各指标均归一化至0~1,全面揭示不同优化器的原理特征与性能差异。实验结果如图10所示,充分证明了EBKA在超参数优化任务中,性能和效率2个方面

的显著优势。

3.8 消融实验

为了系统性地分析MFM-BKOA-NTP各核心组件的实际贡献度并提供细粒度的性能解释,设计了一系列详尽的消融实验。通过在完整框架的基础上依次移除关键组件,定量评估各模块对整体预测性能的影响程度,从而验证框架设计的合理性和各组件的必要性。

3.8.1 实验设置

消融实验包括以下5种配置。

完整框架(基准):采用完整MFM-BKOA-

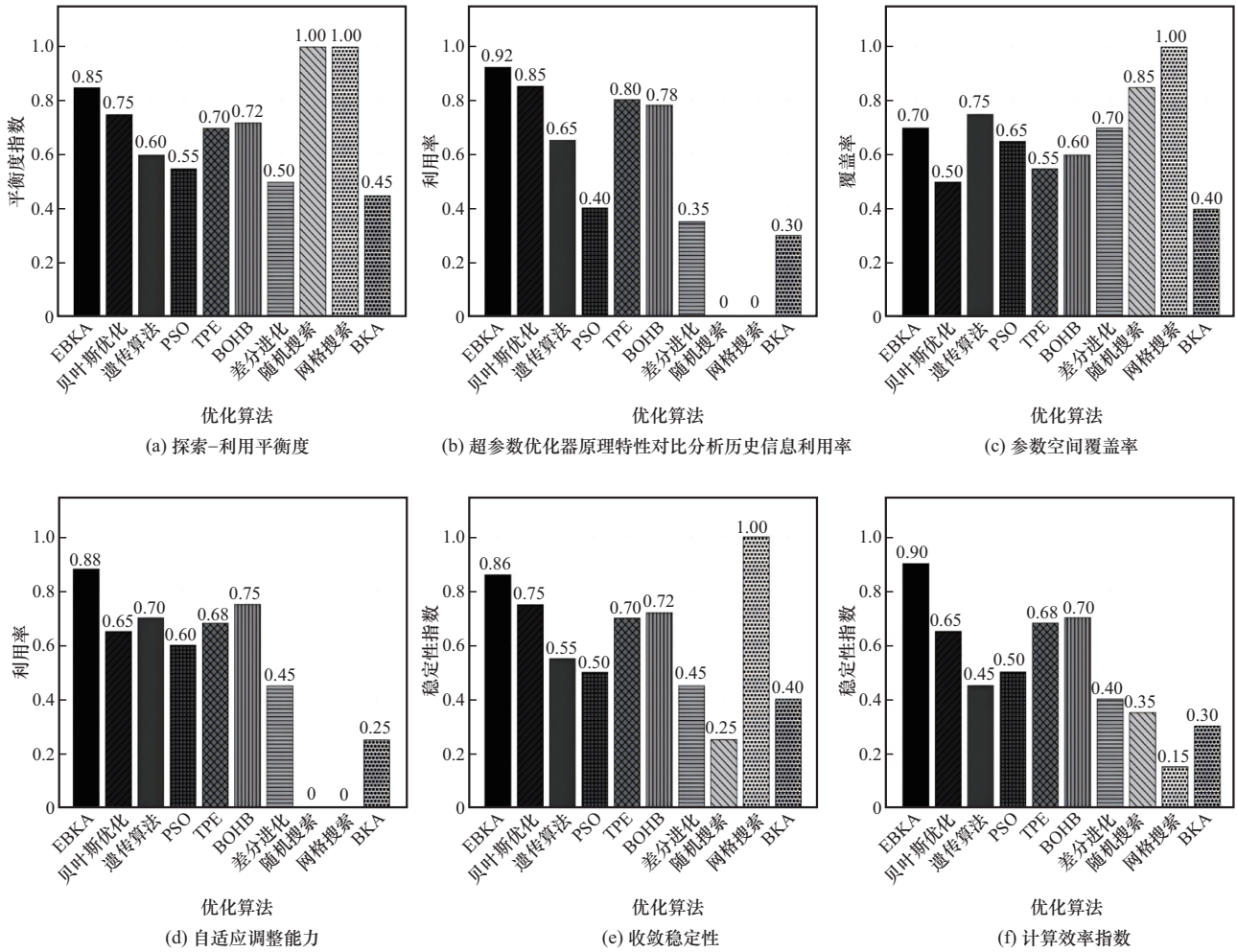


图 10 EBKA 与其他超参数优化器的对比

NTP, 包含所有核心组件, 作为性能评估的基准线。

移除经验库检索: 移除经验库模块, 使 EBKA 优化器和模型选择器无法利用历史优化经验, 退化为标准的启发式搜索策略。

移除模型选择器: 移除基于 DeepSeek 大模型微调的智能模型选择器, 采用随机选择的传统策略。

移除精英保留: 移除 EBKA 中的精英保留机制, 使算法无法保持历代最优解, 从而导致优化过程中的性能退化。

移除柯西变异: 移除 EBKA 中的柯西变异策略, 降低算法跳出局部最优的能力, 影响全局搜索效果。

3.8.2 实验结果与分析

为了更直观地展示各组件移除对不同性能指标的具体影响程度, 绘制了组件重要性热力图, 如图 11 所示。该热力图以颜色深浅表示性能下降的严重程度, 颜色越深表示该组件对相应指标的影响越大, 可见经验库组件和模型选择器是 MFM-BKOA-NTP

的最重要组件。在全部 10 个网络流量数据集上对上述配置进行了 5 次独立重复实验, 从图 11 的实验结果可以得出以下重要结论。

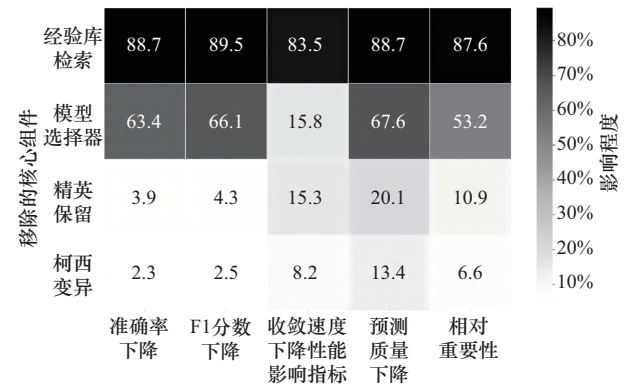


图 11 MFM-BKOA-NTP 核心组件的重要性

经验库检索的决定性作用: 移除经验库检索导致了最为严重的性能下降。具体而言, RMSE 从基

准的 0.092 5 激增至 0.195 0 (增长 110.8%), R^2 从 0.883 6 大幅下降至 0.620 0 (下降 29.8%), MAE 和 MAPE 分别恶化 104.4% 和 122.0%。这一结果充分证明了利用历史优化经验指导新任务的关键价值。经验库不仅能够为模型选择提供先验知识,更能显著提升超参数优化的效率和效果,避免了从零开始的盲目搜索。

模型选择器的重要贡献: 移除智能模型选择器同样造成了显著的性能衰减。RMSE 增加 64.3% 至 0.152 0, R^2 下降 16.8% 至 0.735 0, MAE 和 MAPE 分别增长 60.9% 和 67.5%。这些数据有力验证了基于数据元特征进行智能模型匹配的机制远优于随机选择或固定模型策略。通过 DeepSeek 大模型的语义理解能力, 框架能够准确识别不同网络流量数据的内在模式特征, 从而选择最适配的预测模型。

EBKA 组件的必要性: 精英保留和柯西变异作为 EBKA 的核心改进机制, 其重要性同样得到了实验验证。移除精英保留导致 RMSE 增长 13.0%, R^2 下降 3.1%; 移除柯西变异使 RMSE 增加 7.6%, R^2 下降 1.7%。虽然这 2 个组件的影响相对较小, 但它们对于提升超参数优化的稳定性和全局搜索能力仍然发挥着不可替代的作用。

性能下降的层次性特征: 从各组件移除后的性能下降幅度可以清晰地看出框架组件的重要性层次: 经验库检索 > 模型选择器 > 精英保留 > 柯西变异。这一层次结构与理论分析完全吻合, 进一步支撑了经验库检索与模型选择器为最关键模块的核心结论。

计算效率的影响分析: 从训练时间和预测时间

的变化可以看出, 移除经验库检索不仅严重影响预测精度, 还显著增加了计算开销 (训练时间从 245.6 s 增至 425.8 s), 这是因为缺乏历史经验指导导致优化过程需要更多的搜索迭代。相比之下, 其他组件的移除对计算效率的影响相对较小。

4 结束语

本文针对网络流量预测中模型与数据模式不匹配及超参数优化效率低下的挑战, 创新性地提出了基于元特征映射与经验黑翅鸢优化的自动化网络流量预测框架。该框架通过构建 38 维元特征经验库并微调 DeepSeek-1.5B 模型, 实现了数据特征与最优模型的智能匹配, 显著提升了模型选择的准确性。同时, 引入改进的经验黑翅鸢优化算法, 有效解决了传统超参数优化方法效率低和易陷入局部最优的问题, 大幅提高了优化效率和模型性能。

实验结果充分验证了 MFM-BKOA-NTP 的优越性。在多项性能指标上, MFM-BKOA-NTP 在预测精度、计算效率和跨域泛化能力方面均显著超越了现有方法, 特别是其模型选择器和经验库组件在消融实验中展现出关键贡献。MFM-BKOA-NTP 极大地降低了网络流量预测的技术门槛和人工成本, 为实际应用提供了高效、可靠的解决方案。

未来研究将进一步探索 MFM-BKOA-NTP 在更复杂网络环境和新型流量模式下的适应性, 并结合联邦学习等技术, 以应对数据隐私和分布式部署的挑战, 持续提升网络流量预测的智能化水平和应用价值。

附录 1 38 维度元特征具体细节

38 维度元特征具体细节如表 3 所示。

表 2 各模块对整体预测性能的影响程度

| 方法 | RMSE | R^2 | MAE | MAPE | SMAPE | 95%分位数 预测误差 | 训练时间/s | 预测耗时/s |
|--------------|----------------------|---------------------|----------------------|---------------------|--------------------|----------------------|-------------------|-----------------|
| 完整框架 (基准) | 0.092 5 | 0.883 6 | 0.072 4 | 6.24% | 11.43 | 0.166 5 | 245.6 | 3.2 |
| 移除经验 库模块 | 0.195 0 (+110.8%) | 0.620 0 (-29.8%) | 0.148 0 (+104.4%) | 13.85% (+122.0%) | 22.50% (+96.9%) | 0.370 5 (+122.5%) | 425.8 (+73.4%) | 5.8 (+81.2%) |
| 移除模型 选择器 | 0.152 0 (+64.3%) | 0.735 0 (-16.8%) | 0.116 5 (+60.9%) | 10.45% (+67.5%) | 17.85% (+56.2%) | 0.273 6 (+64.3%) | 315.2 (+28.3%) | 4.1 (+28.1%) |
| 移除精英 保留 | 0.104 5(+13.0%) | 0.856 5 (-3.1%) | 0.081 5 (+12.6%) | 7.15% (+14.6%) | 12.85% (+12.4%) | 0.188 1 (+13.0%) | 285.5 (+16.2%) | 3.7 (+15.6%) |
| 移除柯西 变异 | 0.099 5 (+7.6%) | 0.868 5 (-1.7%) | 0.077 5 (+7.0%) | 6.75% (+8.2%) | 12.20% (+6.7%) | 0.179 1 (+7.6%) | 268.4 (+9.3%) | 3.5 (+9.4%) |

表 3 38 维度元特征具体细节

| 序号 | 特征名 | 符号 | 数学定义 | 复杂度 |
|----|------------|---------------------|--|--------------|
| 1 | 均值 | μ | $\mu = \frac{1}{N} \sum_{i=1}^N x_i$ | $O(N)$ |
| 2 | 方差 | σ^2 | $\sigma^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \mu)^2$ | $O(N)$ |
| 3 | 标准差 | σ | $\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \mu)^2}$ | $O(N)$ |
| 4 | 最小值 | x_{\min} | $x_{\min} = \min_{1 \leq i \leq N} \{x_i\}$ | $O(N)$ |
| 5 | 最大值 | x_{\max} | $x_{\max} = \max_{1 \leq i \leq N} \{x_i\}$ | $O(N)$ |
| 6 | 25%分位数 | Q_1 | $Q_1 = x_{(\lfloor 0.25N \rfloor)}$ | $O(N \lg N)$ |
| 7 | 50%分位数 | Q_2 | $Q_2 = \begin{cases} x_{(\frac{N}{2})} & ,N \text{ 是偶数} \\ \frac{x_{(\lfloor \frac{N}{2} \rfloor)} + x_{(\lfloor \frac{N}{2} \rfloor + 1)}}{2} & ,N \text{ 是奇数} \end{cases}$ | $O(N \lg N)$ |
| 8 | 75%分位数 | Q_3 | $Q_3 = x_{(\lfloor 0.75N \rfloor)}$ | $O(N \lg N)$ |
| 9 | 偏度 | S | $S = \frac{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^3}{\sigma^3}$ | $O(N)$ |
| 10 | 峰度 | Kurt | $K = \frac{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^4}{\sigma^4} - 3$ | $O(N)$ |
| 11 | 一阶差分均值 | μ_{Δ_1} | $\mu_{\Delta_1} = \frac{1}{N-1} \sum_{i=2}^N (x_i - x_{i-1})$ | $O(N)$ |
| 12 | 一阶差分标准差 | σ_{Δ_1} | $\sigma_{\Delta_1} = \sqrt{\frac{1}{N-2} \sum_{i=2}^N (\Delta_1 x_i - \mu_{\Delta_1})^2}$ | $O(N)$ |
| 13 | 二阶差分均值 | μ_{Δ_2} | $\mu_{\Delta_2} = \frac{1}{N-2} \sum_{i=3}^N (\Delta_1 x_i - \Delta_1 x_{i-1})$ | $O(N)$ |
| 14 | 二阶差分标准差 | σ_{Δ_2} | $\sigma_{\Delta_2} = \sqrt{\frac{1}{N-3} \sum_{i=3}^N (\Delta_2 x_i - \mu_{\Delta_2})^2}$ | $O(N)$ |
| 15 | 趋势强度 | T_s | $T_s = 1 - \frac{\text{Var}(R_t)}{\text{Var}(x_t)}$ | $O(N)$ |
| 16 | 季节性强度 | S_s | $S_s = 1 - \frac{\text{Var}(R_s)}{\text{Var}(x_t - T_t)}$ | $O(N)$ |
| 17 | 滞后 1 自相关系数 | ρ_1 | $\rho_1 = \frac{\sum_{i=2}^N (x_i - \mu)(x_{i-1} - \mu)}{\sum_{i=1}^N (x_i - \mu)^2}$ | $O(N)$ |
| 18 | 滞后 2 自相关系数 | ρ_2 | $\rho_2 = \frac{\sum_{i=3}^N (x_i - \mu)(x_{i-2} - \mu)}{\sum_{i=1}^N (x_i - \mu)^2}$ | $O(N)$ |
| 19 | 转折点数量 | TP | $\text{TP} = \sum_{i=2}^{N-1} \mathbb{I}[(x_{i-1} - x_i)(x_i - x_{i+1}) > 0]$ | $O(N)$ |
| 20 | 线性趋势系数 | β | $\beta = \frac{N \sum_{i=1}^N i \cdot x_i - \sum_{i=1}^N i \sum_{i=1}^N x_i}{N \sum_{i=1}^N i^2 - (\sum_{i=1}^N i)^2}$ | $O(N)$ |
| 21 | 主频率 | f_d | $f_d = \arg \max_f X(f) ^2$ | $O(N \lg N)$ |

续表

| 序号 | 特征名 | 符号 | 数学定义 | 复杂度 |
|----|----------------|----------|---|-------------|
| 22 | 频谱能量 | E_s | $E_s = \sum_{k=0}^{\frac{N}{2}} X_k ^2$ | $O(M\lg N)$ |
| 23 | 频谱熵 | H_s | $H_s = - \sum_{k=0}^{\frac{N}{2}} p_k \lg p_k$ | $O(M\lg N)$ |
| 24 | 频谱重心 | C_s | $C_s = \frac{\sum_{k=0}^{\frac{N}{2}} k \cdot X_k ^2}{\sum_{k=0}^{\frac{N}{2}} X_k ^2}$ | $O(M\lg N)$ |
| 25 | 频谱带宽 | B_s | $B_s = \sqrt{\frac{\sum_{k=0}^{\frac{N}{2}} (k - C_s)^2 \cdot X_k ^2}{\sum_{k=0}^{\frac{N}{2}} X_k ^2}}$ | $O(M\lg N)$ |
| 26 | 频谱平坦度 | F_s | $F_s = \frac{\sqrt{\frac{N}{2} \prod_{k=0}^{\frac{N}{2}} X_k ^2}}{\frac{1}{N/2 + 1} \sum_{k=0}^{\frac{N}{2}} X_k ^2}$ | $O(M\lg N)$ |
| 27 | 频谱滚降点 | R_s | $R_s = \min \left\{ k: \sum_{j=0}^k X_j ^2 \geq 0.85 \sum_{j=0}^{\frac{N}{2}} X_j ^2 \right\}$ | $O(M\lg N)$ |
| 28 | 频谱质心 | C_{sn} | $C_{sn} = \frac{C_s}{f_{\max}}$ | $O(M\lg N)$ |
| 29 | 变异系数 | CV | $CV = \frac{\sigma}{ \mu }$ | $O(N)$ |
| 30 | 相对极差 | RR | $RR = \frac{x_{\max} - x_{\min}}{ \mu }$ | $O(N)$ |
| 31 | 平均绝对偏差 | MAD | $MAD = \frac{1}{N} \sum_{i=1}^N x_i - \mu $ | $O(N)$ |
| 32 | 四分位距 | IQR | $IQR = Q_3 - Q_1$ | $O(N)$ |
| 33 | 稳定性指数 | SI | $SI = 1 - \frac{CV}{CV_{\max}}$ | $O(N)$ |
| 34 | 样本熵 | SampEn | $\text{SampEn}(m, r, N) = -\ln\left(\frac{A}{B}\right)$ | $O(N^2)$ |
| 35 | 近似熵 | ApEn | $\text{ApEn}(m, r, N) = \phi(m) - \phi(m + 1)$ | $O(N^2)$ |
| 36 | Lempel-Ziv 复杂度 | LZC | $LZC = \frac{c(n)}{\frac{n}{\lg n}}$ | $O(N^2)$ |
| 37 | 分形维数 | FD | $FD = \frac{\lg N}{\lg N + \lg\left(\frac{L}{a}\right)}$ | $O(N^2)$ |
| 38 | Hurst 指数 | H | $H = \frac{\lg\left(\frac{R}{S}\right)}{\lg(n)}$ | $O(N^2)$ |

附录 2 LoRA 微调训练配置与复现细节

LoRA 微调训练配置与复现细节如表 4 和表 5 所示。

表 4 LoRA 配置

| 类别 | 配置说明 |
|------------|-------------------------------------|
| LoRA 秩 | $r = 8$, 控制低秩矩阵维度, 平衡参数量与表达能力 |
| LoRA 缩放系数 | $\alpha = 16$, 用于控制 LoRA 层对输出的贡献程度 |
| LoRA 随机丢弃 | 随机失活率 (dropout) 为 0.02, 正则化以防过拟合 |
| LoRA 学习率比例 | LoRA 层学习率为主网络的 2.48 倍 |
| rsloa | 启用改进的 LoRA 实现, 提高训练稳定性 |

表 5 关键训练超参数

| 类别 | 配置说明 |
|--------|---|
| 学习率 | 主网络 lr = 5×10^{-5} ; LoRA 层 lr = 1.24×10^{-4} |
| 学习率调度 | 线性衰减, 前 500 步预热 |
| 最大梯度范数 | 设置 1.0, 用于梯度裁剪防止梯度爆炸 |
| 训练步数 | total _{steps} = 5 000 |
| 数据划分 | 训练、验证、测试分别为 80%、10%、10% |
| 早停机制 | 验证集损失连续 5 次无下降触发 |
| 精度 | 全精度 (bf16 可选), 量化方法是比特字节量化法 (bnb) |
| 硬件与环境 | NVIDIA RTX 4090 GPU, CUDA 12.4 |
| 框架与版本 | Python 3.12.3, LlamaFactory |

参考文献:

- [1] 王升辉, 裘正定. 结合多重分形的网络流量非线性预测[J]. 通信学报, 2007, 28(2): 45-50, 57.
WANG S H, QIU Z D. Network traffic nonlinear prediction combined with multifractal[J]. Journal on Communications, 2007, 28(2): 45-50, 57.
- [2] SINGH S, HUSSAIN S, BAZAZ M A. Short term load forecasting using artificial neural network[C]//Proceedings of the 2017 Fourth International Conference on Image Information Processing (ICIIP). Piscataway: IEEE Press, 2017: 1-5.
- [3] LIU H T, CAI J F, ONG Y S, et al. Understanding and comparing scalable Gaussian process regression for big data[J]. Knowledge-Based Systems, 2019, 164: 324-335.
- [4] IRANMEHR A, MASNADI-SHIRAZI H, VASCONCELOS N. Cost-sensitive support vector machines[J]. Neurocomputing, 2019, 343: 50-64.
- [5] SCHOOT R V D, DEPAOLI S, KING R, et al. Bayesian statistics and modelling[J]. Nature Reviews Methods Primers, 2021, 1: 1.
- [6] ZHOU Z H. Machine learning[M]. Berlin: Springer, 2021
- [7] 崔佳旭, 杨博. 贝叶斯优化方法和应用综述[J]. 软件学报, 2018, 29(10): 3068-3090.
CUI J X, YANG B. Survey on Bayesian optimization methodology and applications[J]. Journal of Software, 2018, 29(10): 3068-3090.
- [8] CHEN X, LIU Y N, ZHANG J K. Traffic prediction for Internet of Things through support vector regression model[J]. Internet Technology Letters, 2022, 5(3): e336.
- [9] 周晓剑, 肖丹, 付裕. 增量式 Huber-支持向量回归机算法研究[J]. 运筹与管理, 2022, 31(8): 137-142.
ZHOU X J, XIAO D, FU Y. A study of incremental Huber-support vector regression algorithm[J]. Operations Research and Management Science, 2022, 31(8): 137-142.
- [10] XU Y, YIN F, XU W J, et al. Wireless traffic prediction with scalable Gaussian process: framework, algorithms, and verification[J]. IEEE Journal on Selected Areas in Communications, 2019, 37(6): 1291-1306.
- [11] BI J, ZHANG X, YUAN H T, et al. A hybrid prediction method for realistic network traffic with temporal convolutional network and LSTM[J]. IEEE Transactions on Automation Science and Engineering, 2022, 19(3): 1869-1879.
- [12] KOUGIOUMTZIDIS G, POULKOV V K, LAZARIDIS P I, et al. Mobile network traffic prediction using temporal fusion transformer[J]. IEEE Transactions on Artificial Intelligence, 2025, 6(10): 2685-2699.
- [13] NDIAYE E, LE T, FERCOPO O, et al. Safe grid search with optimal complexity[J]. arXiv Preprint, arXiv: 1810.05471, 2018.
- [14] BERGSTRA J, BENGIO Y. Random search for hyper-parameter optimization[J]. Journal of Machine Learning Research, 2012, 13: 281-305.
- [15] FRAZIER P I. Bayesian optimization[C]//Recent Advances in Optimization and Modeling of Contemporary Problems. Piscataway: IEEE Press, 2018: 255-278.
- [16] ZHAO C M, WU D R, HUANG J, et al. BoostTree and BoostForest for ensemble learning[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(7): 8110-8126.
- [17] CHEN S H, ZHAI W H, CHAI C, et al. LLM2AutoML: zero-code AutoML framework leveraging large language models[C]//Proceedings of the 2024 International Conference on Intelligent Robotics and Automatic Control (IRAC). Piscataway: IEEE Press, 2024: 285-290.
- [18] WANG X Z, WIRTH A, WANG L. Structure-based statistical features and multivariate time series clustering[C]//Proceedings of the 2007 Seventh IEEE International Conference on Data Mining. New York: ACM Press, 2007: 351-360.
- [19] FULCHER B D. Feature-based time-series analysis[C]//Feature Engineering for Machine Learning and Data Analytics. Boca Raton: CRC Press, 2018: 87-116.
- [20] CERQUEIRA V, TORGO L, SOARES C. A case study comparing machine learning with statistical methods for time series forecasting: size matters[J]. Journal of Intelligent Information Systems, 2022, 59(2): 415-433.
- [21] ZHANG R, SUN Y, WEN H, et al. Time series analysis in frequency domain: a survey of open challenges, opportunities and benchmarks[J]. Proceedings of the ACM on Measurement and Analysis of Computing Systems, 2018, 37(4): 111.
- [22] GUNASEKARAN S, KEMBAY A, LADRET H, et al. A predictive approach to enhance time-series forecasting[J]. Nature Communications, 2025, 16: 8645.
- [23] ZHAO Q H, LAI F, MO X Y. Exploring time series analysis in frequency domain with complex-valued spectral attention and bidirectional variable mamba[J]. The Journal of Supercomputing, 2025, 81(8): 875.

- [24] KONG X J, CHEN Z H, LIU W Y, et al. Deep learning for time series forecasting: a survey[J]. *International Journal of Machine Learning and Cybernetics*, 2025, 16(7): 5079-5112.
- [25] LIM B, ZOHREN S. Time-series forecasting with deep learning: a survey[J]. *Philosophical Transactions Series A, Mathematical, Physical, and Engineering Sciences*, 2021, 379(2194): 20200209.
- [26] 杨海民, 潘志松, 白玮. 时间序列预测方法综述[J]. *计算机科学*, 2019, 46(1): 21-28.
YANG H M, PAN Z S, BAI W. Review of time series prediction methods[J]. *Computer Science*, 2019, 46(1): 21-28.
- [27] PRYAMIKOV A. Deep learning as a highly efficient tool for digital signal processing design[J]. *Light: Science & Applications*, 2024, 13: 248.
- [28] SWINNEY C J, WOODS J C. Unmanned aerial vehicle operating mode classification using deep residual learning feature extraction[J]. *Aerospace*, 2021, 8(3): 79.
- [29] CASINI A, PERRON P. Change-point analysis of time series with evolutionary spectra[J]. *Journal of Econometrics*, 2024, 242(2): 105811.
- [30] MASINI R P, MEDEIROS M C, MENDES E F. Machine learning advances for time series forecasting[J]. *Journal of Economic Surveys*, 2023, 37(1): 76-111.
- [31] 万晨, 李文中, 丁望祥, 等. 一种基于自演化预训练的多变量时间序列预测算法[J]. *计算机学报*, 2022, 45(3): 513-525.
WAN C, LI W Z, DING W X, et al. A multivariate time series forecasting algorithm based on self-evolution and pre-training[J]. *Chinese Journal of Computers*, 2022, 45(3): 513-525.
- [32] FOTOVVIKHAH F, AHMEDY I, NOOR R M, et al. A systematic review of AI-based techniques for automated waste classification[J]. *Sensors*, 2025, 25(10): 3181.
- [33] BOSHPANOV G N. Introduction to time series analysis and forecasting[J]. *Journal of Time Series Analysis*, 2016, 37(6): 864.
- [34] ULKADIR S J. ARX time series model analysis with wavelets shrinkage (simulation study)[J]. *Iraqi Journal of Statistical Sciences*, 2024, 23(1): 1-15.
- [35] MCGONIGLE E T, KILLICK R, NUNES M A. Modelling time-varying first and second-order structure of time series via wavelets and differencing[J]. *Electronic Journal of Statistics*, 2022, 16(2): 4936-4981.
- [36] YI K, ZHANG Q, FAN W, et al. A survey on deep learning based time series analysis with frequency transformation[C]//*Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*. New York: ACM Press, 2025: 6206-6215.
- [37] INOUE M, KOSHIKIYA Y. Enhanced frequency and time domain feature extraction for communication infrastructure type classification using optical fiber sensing[J]. *Optical Fiber Technology*, 2024, 87: 103859.
- [38] ATAD D A, MEDIANO P A M, ROSAS F E, et al. Meditation and complexity: a review and synthesis of evidence[J]. *Neuroscience of Consciousness*, 2025, 2025(1): niaf013.
- [39] GEORGE S V. Chaos to clarity: interpreting time series complexity metrics with an application to depression[J]. *Discover Mental Health*, 2025, 5(1): 97.
- [40] CUI S Y, LI H, QIU J J, et al. An improved CPO algorithm based on elite preservation strategy[C]//*Proceedings of the 2024 10th International Conference on Systems and Informatics (ICSAI)*. Piscataway: IEEE Press, 2024: 1-9.
- [41] 罗凤鸣, 吕方林, 侯宗琰. 基于精英保留策略与爆炸算子的改进遗传算法[J]. *西华大学学报(自然科学版)*, 2018, 37(3): 83-88.
LUO F M, LYU F L, HOU Z Y. An improved genetic algorithm based on elite retention strategy and explosion operators[J]. *Journal of Xihua University (Natural Science Edition)*, 2018, 37(3): 83-88.
- [42] LI F, PAN H, JI Y L, et al. Particle swarm optimization based on particle perturbation and elite preservation strategies for dynamic optimization problems[J]. *Expert Systems with Applications*, 2025, 279: 127423.
- [43] JABARI M, IZCI D, EKINCI S, et al. Hybrid adaptive ant lion optimization with traditional controllers for driving and controlling switched reluctance motors to enhance performance[J]. *Scientific Reports*, 2025, 15: 12898.
- [44] FAN S J, WANG R C, SU K, et al. A sequoia-ecology-based meta-heuristic optimisation algorithm for multi-constraint engineering design and UAV path planning[J]. *Results in Engineering*, 2025, 26: 105130.
- [45] 刘健, 李京航, 柏小丽. 基于精英保留策略遗传算法的配电网无功优化[J]. *电气技术*, 2015, 16(4): 35-38.
LIU J, LI J H, BAI X L. Reactive power optimization of distribution network using genetic algorithm with elitist strategy[J]. *Electrical Engineering*, 2015, 16(4): 35-38.
- [46] LI K W, LI S H, HUANG Z C, et al. Grey Wolf Optimization algorithm based on Cauchy-Gaussian mutation and improved search strategy[J]. *Scientific Reports*, 2022, 12: 18961.
- [47] 袁磊. 融合多策略的改进河马优化算法及其应用[J]. *海南大学学报(自然科学版中英文)*, 2025, 43(3): 289-296.
YUAN L. Application of improved hippopotamus optimization algorithm integrating multi-strategy[J]. *Natural Science of Hainan University*, 2025, 43(3): 289-296.
- [48] YAN J L, HU G, SHU B. MGCHMO: a dynamic differential human memory optimization with Cauchy and Gauss mutation for solving engineering problems[J]. *Advances in Engineering Software*, 2024, 198: 103793.
- [49] 金相丞, 钱博. 基于 CA-GA 的 Polar 码构造算法研究[J]. *无线通信*, 2024(5): 92-103.
JIN X C, QIAN B. Research on polar code construction algorithm based on CA-GA algorithm[J]. *Hans Journal of Wireless Communications*, 2024(5): 92-103.
- [50] CHEN B Z, OUYANG H B, LI S, et al. Dual-stage self-adaptive differential evolution with complementary and ensemble mutation strategies[J]. *Swarm and Evolutionary Computation*, 2025, 93: 101855.
- [51] 舒一鸣, 戴毅茹. 基于高斯-柯西混合变异的多目标粒子群算法[J]. *计算机与数字工程*, 2024, 52(6): 1593-1597, 1603.
SHU Y M, DAI Y R. Multi-objective particle swarm optimization algorithm based on Gaussian-cauchy mixture mutation[J]. *Computer & Digital Engineering*, 2024, 52(6): 1593-1597, 1603.
- [52] TAN K C, FENG L, JIANG M. Evolutionary transfer optimization - a new frontier in evolutionary computation research[J]. *IEEE Computational Intelligence Magazine*, 2021, 16(1): 22-33.
- [53] 李金龙, 黄英. 模糊物元分析在经验库算法中的性能优化[J]. *计算机系统应用*, 2009, 18(1): 50-54.
LI J L, HUANG Y. Performance optimization of experience-database based on fuzzy matter-element analysis[J]. *Computer Systems & Appli-*

cations, 2009, 18(1): 50-54.

- [54] KIM J J, PARK S Y, LEE J J. Experience repository based Particle Swarm Optimization for evolutionary robotics[C]//Proceedings of the 2009 ICCAS-SICE. Piscataway: IEEE Press, 2009: 2540-2544.
- [55] SEYYEDABBASI A. A reinforcement learning-based metaheuristic algorithm for solving global optimization problems[J]. Advances in Engineering Software, 2023, 178: 103411.
- [56] LI J Y, ZHAN Z H, TAN K C, et al. A meta-knowledge transfer-based differential evolution for multitask optimization[J]. IEEE Transactions on Evolutionary Computation, 2022, 26(4): 719-734.
- [57] 吕世玮, 黄德辅, 高聪, 等. PSO 优化算法在尾矿库空间预测中的应用[J]. 有色金属(矿山部分), 2024, 76(1): 126-132.
- LYU S W, HUANG D Y, GAO C, et al. Application of PSO optimization algorithm in tailings pond spatial prediction[J]. Nonferrous Metals (Mining Section), 2024, 76(1): 126-132.

[作者简介]



蒋献 (1988-), 男, 浙江兰溪人, 浙江工商大学博士生, 主要研究方向为智慧网络与智慧教育。



陈嗣元 (2004-), 男, 河南新乡人, 浙江工商大学硕士生, 主要研究方向为时间序列、网络安全与 AI 安全、具身智能关键技术。



苟敬文 (2000-), 男, 四川巴中人, 浙江工商大学硕士生, 主要研究方向为智慧网络。



张子天 (1988-), 男, 河北石家庄人, 博士, 浙江工商大学副研究员, 主要研究方向为基于机器学习的网络流量预测与资源管理。



诸葛斌 (1976-), 男, 浙江兰溪人, 博士, 浙江工商大学教授, 主要研究方向为网络和通信技术、互联网技术和网络安全。



董黎刚 (1973-), 男, 浙江上虞人, 博士, 浙江工商大学教授, 主要研究方向为智能网络与智慧教育。