

基于多模型融合 Stacking 集成学习方式的 负荷预测方法

史佳琪, 张建华

(新能源电力系统国家重点实验室(华北电力大学), 北京市 昌平区 102206)

Load Forecasting Based on Multi-model by Stacking Ensemble Learning

SHI Jiaqi, ZHANG Jianhua

(State Key Laboratory of Alternative Electrical Power System With Renewable Energy Sources (North China Electric Power University), Changping District, Beijing 102206, China)

ABSTRACT: The rapid development of artificial intelligence and machine learning technology provides an innovative solution for load forecasting. A load forecasting method based on multi-model combination under Stacking framework was proposed, associated with the frontier theory research of artificial intelligence. Considering the difference of data observation and training principles, the Stacking based load forecasting model embedded various machine learning algorithms was proposed to utilize their diversified strength. The XGBoost algorithm constructed by tree ensemble model and long-short term memory were involved in Stacking base-learner layer. Swiss load data in ENTSO was used to verify the feasibility of the algorithm in case study. The forecasting results show that the XGBoost, the gradient decision tree and the random forest model can quantify the contribution of the input data through the gain of their own model. The load forecasting results are more accurate when each of base-learner has lower correlation coefficient in Stacking. The results indicate the Stacking ensemble learning based on multi-model has better prediction performance compared with the traditional single model.

KEY WORDS: artificial intelligence; load forecasting; multi-model combination; Stacking ensemble learning; XGBoost; long-short term memory

摘要: 人工智能及机器学习技术的快速发展, 为负荷预测问题提供了崭新的解决思路。该文结合人工智能的前沿理论研究, 提出一种基于多模型融合 Stacking 集成学习方式的负荷预测方法。考虑不同算法的数据观测与训练原理差异, 充分发挥各个模型优势, 构建多个机器学习算法嵌入的 Stacking 集成学习的负荷预测模型, 模型的基学习器包含 XGBoost 树集成算法和长短记忆网络算法。算例使用 ENTSO 中瑞士负荷数据对算法有效性进行了验证。预测结果表明,

XGBoost、梯度决策树、随机森林模型能够通过自身模型的增益情况对输入数据的特征贡献度进行量化分析; Stacking 中各个基学习器的学习能力越强, 关联程度越低, 模型预测效果越好; 与传统单模型预测相比, 基于多模型融合的 Stacking 集成学习方式的负荷预测方法有着较高的预测精度。

关键词: 人工智能; 负荷预测; 多模型融合; Stacking 集成学习; XGBoost; 长短记忆网络

0 引言

人工智能技术的快速发展为当前人类社会带来了巨大变革, AlphaGo 击败围棋高手、无人驾驶汽车上路、医疗机器人 Watson 诊断出世界稀有病例, 标志着人类逐步迈入人工智能时代。能源行业的数字化趋势为人工智能与机器学习技术在电力行业的应用创造了新机遇^[1-3]。人工智能与机器学习的相关技术已在可再生能源预测、负荷预测、电力系统故障诊断与优化调度、电网数据可视化等方面获得了广泛的应用^[4-5]。与此同时, 国务院办公厅于 2017 年 7 月 20 日印发《新一代人工智能发展规划》^[6]提出了面向 2030 年我国新一代人工智能发展的指导思想、战略目标、重点任务和保障措施, 加快建设创新型国家和世界科技强国, 将人工智能的发展上升到国家战略层面。由此可见, 人工智能技术是现代电网发展的必然选择, 是能源电力转型的重要战略支撑。

其中, 电力负荷预测是电力系统发电、规划、调度等部门的一项重要日常工作。精确的负荷预测将有效提升电网调度水平、安全稳定状况和损耗控制能力^[7-8]。负荷预测是人工智能技术在电力系统应

用最广泛、深入的领域之一，国内外学者已经对负荷预测的理论和方法开展了大量的研究。针对于电负荷的预测方法主要包括神经网络^[9]、支持向量机^[10]、小波分析理论^[11]等。此外，以树集成算法和深度学习算法为代表的先进人工智能技术在负荷预测中也取得了良好的应用效果。文献[12]针对传统负荷预测问题泛化性能不强、参数和模型结构确定困难等问题，将灰色投影改进的随机森林回归算法引入短期负荷预测领域。文献[13]将 XGBoost (eXtreme gradient boosting) 与多个算法对比分析，表明所构建的电力负荷 XGBoost 预测模型相比随机森林、贝叶斯和 KNN 方法在计算速度和预测精度方面具有优势。文献[14]将深度置信网络算法引入负荷预测问题当中，并与浅层模型的预测结果对比分析。文献[15]使用深度学习中经典算法长短记忆网络(long-short term memory, LSTM)对用户侧的短期用电负荷进行预测。文献[16]采用变分模态分解技术将原始历史负荷序列分解为一系列特征互异的模态函数，并采用改进粒子群算法优化深度信念网络权值。文献[17]分析了综合能源系统中多种异质能源的预测需求，提出了基于深度结构多任务学习的短期电、热、气负荷联合预测方法。

虽然人工智能及机器学习技术的快速发展，为负荷预测提供了崭新的解决思路。但是，上述文献只是采用了一种单独方式进行负荷预测，由于负荷预测问题的假设空间很大，可能有多个假设在训练集上达到同等性能，若使用单一模型可能由于随机性而导致泛化性能不佳。因此，文献[18-20]寻求使用组合预测的方式，进一步提高模型预测精度。但组合的方式多是采用均值计算方式，求取多种算法模型或同一类型算法不同参数模型的预测均值，并不能体现出不同预测算法数据观测的差异性，各个算法并不能通过取长补短的方式训练出更优异模型，并且这种组合方式没有足够的理论支撑，原理也较为单薄。

本文在分析了负荷预测与人工智能技术的发展背景的基础上，将前沿人工智能技术 XGBoost 算法、LSTM 算法与集成学习方式有效结合，提出了一种 Stacking 集成学习架构下基于多个差异化模型的负荷预测方法。首先介绍了 Stacking 的集成学习方式，以及 XGBoost 算法和 LSTM 算法训练机理；然后，在 Stacking 集成框架下，考虑多种模型的数据观测空间，建立了多模型融合的负荷预测模型；最后，采用 ENTSO 的瑞士负荷数据验证了算

法的有效性。结果表明，基于多模型融合的 Stacking 集成学习方法在电力负荷预测方面有较好的应用效果。

1 算法理论介绍

1.1 基于 Stacking 的集成学习方式

Stacking^[21]集成学习框架首先将原始数据集划分成若干子数据集，输入到第 1 层预测模型的各个基学习器中，每个基学习器输出各自的预测结果。然后，第 1 层的输出再作为第 2 层的输入，对第 2 层预测模型的元学习器进行训练，再由位于第 2 层的模型输出最终预测结果。Stacking 学习框架通过对多个模型的输出结果进行泛化，以获得整体预测精度的提升，如图 1 所示。

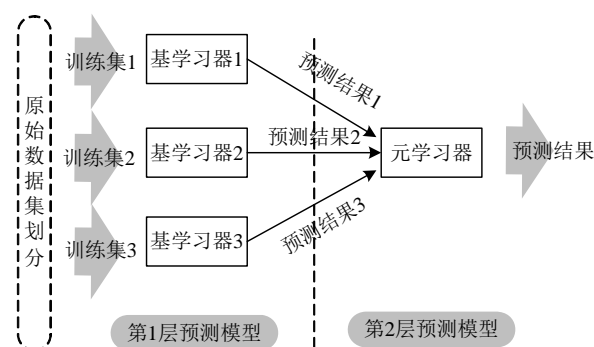


图 1 基于 Stacking 的集成学习方式

Fig. 1 Ensemble learning method based on Stacking

Stacking 集成学习的具体训练方式是：对于数据集 $S=\{(y_n, x_n), n=1, \dots, N\}$ ，其中 x_n 为第 n 个样本的特征向量， y_n 为第 n 个样本对应的预测值， p 为所包含特征数量，即每一个特征向量为 (x_1, x_2, \dots, x_p) 。随机将数据划分成 K 个大小基本相等的子集 S_1, S_2, \dots, S_k 。其中 $S_{-k}=S-S_k$ ，分别定义 S_k 和 S_{-k} 为 K 折交叉验证中的第 k 折测试集与训练集。对于第 1 层预测算法包含 K 个基学习器，对训练集 S_{-k} 用第 k 个算法训练得到基模型 $L_k, k=1, \dots, K$ 。

对于 K 折交叉验证中的第 k 折测试集 S_k 中的每个样本 x_n ，基学习器 L_k 对它的预测表示为 z_{kn} 。在完成交叉验证过程后，将 K 个基学习器的输出数据构成新的数据样本，即： $S_{\text{new}}=\{(y_n, z_{1n}, \dots, z_{kn}), n=1, \dots, N\}$ 。

新产生的数据集就是 Stacking 第 2 层输入数据。使用第 2 层预测算法对这些数据进行归纳得到的元学习器 L_{new} 。Stacking 的配置方式使得第 1 层算法的训练结果能够充分用于第 2 层算法的归纳过程当中，第 2 层算法能够发现并且纠正第 1 层学习算法中的预测误差，以提升模型的精度。Stacking

集成学习的训练流程部分伪代码如下。

输入: 数据集 $S=\{(y_n, x_n), n=1, \dots, N\}$

输出: Stacking 集成学习模型

步骤 1: 将数据划分成 K 个大小基本相等的子集 S_1, S_2, \dots, S_k , 其中 $S_k=S-S_k$

步骤 2: 对第一层 K 个基学习器训练

for $k=1$ to K

基于数据集 S_k 训练第一层的基学习

器 L_k

end

步骤 3: 构成新数据集

$S_{\text{new}}=\{(y_n, z_{1n}, \dots, z_{kn}), n=1, \dots, N\}$

步骤 4: 基于 S_{new} 对第二层预测模型进行 L_{new} 训练。

1.2 长短记忆网络算法机理

LSTM 神经网络是一种在自然语言处理中广泛应用的深度学习算法。LSTM 网络一般由输入层、输出层和隐藏层堆叠组成。LSTM 在控制存储状态方面相比传统的循环神经网络有了巨大提升。LSTM 的记忆单元结构如图 2 所示。

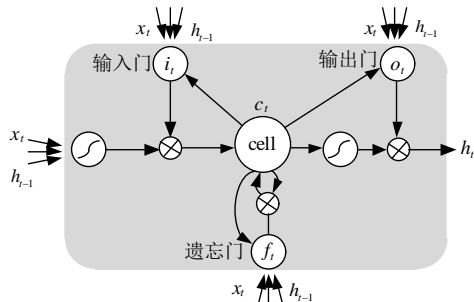


图 2 LSTM 的单元架构

Fig. 2 Framework of LSTM unit

图 2 中, 每一个 LSTM 单元拥有一个细胞元组(cell), 这个元组为具有记忆功能的单元, 其在时刻 t 的状态记为 c_t 。LSTM 单元通过输入门 i_t 、遗忘门 f_t 和输出门 o_t 接收当前状态 x_t 和上一时刻元组的状态 h_{t-1} 。同时, 记忆单元的状态 c_{t-1} 作为内部信息将输入到各个门中。当接收输入信息后, 输入门、遗忘门和输出门将进行内部运算, 判断是否激活该细胞元组。输入门的信号经过非线性函数的变换后, 与遗忘门处理过的记忆单元状态进行叠加, 形成新的记忆单元状态 c_t 。最终, 记忆单元状态 c_t 通过非线性函数的运算和输出门的动态控制形成 LSTM 单元的输出 h_t 。各变量之间的计算公式如下:

$$i_t = \text{sigmoid}[W_{hi} \times h_{t-1} + W_{xi} \times x_t + W_{ci} \times c_{t-1} + b_i] \quad (1)$$

$$f_t = \text{sigmoid}[W_{hf} \times h_{t-1} + W_{xf} \times x_t + W_{cf} \times c_{t-1} + b_f] \quad (2)$$

$$o_t = \text{sigmoid}[W_{ho} \times h_{t-1} + W_{xo} \times x_t + W_{co} \times c_t + b_o] \quad (3)$$

$$c_t = f_t c_{t-1} + i_t \tanh[W_{hg} \times h_{t-1} + W_{xg} \times x_t + b_c] \quad (4)$$

$$h_t = o_t \tanh[c_t] \quad (5)$$

式中: x_t 对应的权重系数分别为 W_{xc} 、 W_{xi} 、 W_{xf} 、 W_{xo} ; h_t 对应的权重系数分别为 W_{hc} 、 W_{hi} 、 W_{hf} 、 W_{ho} ; c_t 对应的权值系数分别为 W_{ci} 、 W_{cf} 、 W_{co} ; 偏置向量分别为 b_i 、 b_c 、 b_f 、 b_o 。LSTM 算法一般采用经典的误差反向传播算法对展开后的网络进行训练在此不加赘述。

1.3 XGBoost 算法机理

XGBoost 是经过优化的集成树模型, 从梯度提升树模型改进和扩展而来。树的集成模型如下所示:

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i), f_k \in F \quad (6)$$

式中: \hat{y}_i 为第 i 个样本的模型预测值; K 为树的数量; F 为树的集合空间; x_i 表示第 i 个数据点的特征向量; f_k 对应第 k 棵独立的树的结构 q 和叶子权重 w 相关状况。

XGBoost 模型损失函数 L 包含两部分:

$$L = \sum_{i=1}^n l(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (7)$$

式中: 第 1 部分为预测值 \hat{y}_i 和目标真实值 y_i 之间的训练误差; 第 2 部分为树的复杂度之和, 是用于控制模型的复杂度的正则项, 即

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \|w\|^2 \quad (8)$$

式中 γ 和 λ 表示对模型的惩罚系数。

式(7)在序列最小化过程中, 每一轮加入的增量函数 $f_t(x_i)$ 尽可能使损失函数最大程度的减小。第 t 轮的目标函数可写为

$$L^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \quad (9)$$

对于式(9)采用二阶泰勒展开的方式来近似目标函数。定义第 j 棵树每一叶子中的样本集合为 $I_j = \{i | q(x_i) = j\}$ 。其中, $g_i = \partial_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})$, $h_i = \partial_{\hat{y}_i^{(t-1)}}^2 l(y_i, \hat{y}_i^{(t-1)})$ 分别为损失函数的一、二阶导数。由此可得

$$\begin{aligned}
L^{(t)} &\cong \sum_{i=1}^n [g_i f_i(x_i) + \frac{1}{2} h_i f_i^2(x_i)] + \Omega(f_i) \cong \\
&\sum_{i=1}^n [g_i f_i(x_i) + \frac{1}{2} h_i f_i^2(x_i)] + \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2 \cong \\
&\sum_{j=1}^T [(\sum_{i \in I_j} g_i) w_j + \frac{1}{2} (\sum_{i \in I_j} h_i + \lambda) w_j^2] + \gamma T \quad (10)
\end{aligned}$$

定义 $G_j = \sum_{i \in I_j} g_i$, $H_j = \sum_{i \in I_j} h_i$, 可得

$$L^{(t)} \cong \sum_{j=1}^T [G_j w_j + \frac{1}{2} (H_j + \lambda) w_j^2] + \gamma T \quad (11)$$

对 w 求偏导可得

$$w_j = -\frac{G_j}{H_j + \lambda} \quad (12)$$

将权值代入目标函数, 可得

$$L^{(t)} \cong -\frac{1}{2} \sum_{j=1}^T \frac{G_j^2}{H_j + \lambda} + \gamma T \quad (13)$$

损失函数越小代表模型越好, 采用贪心算法对子树进行划分, 并枚举可行的分割点, 即每次对已有的叶子加入新的分割, 并计算因此获得的最大增益。增益 L_{Gain} 的计算方式如下:

$$L_{\text{Gain}} \cong \frac{1}{2} \left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma \quad (14)$$

式中: 第 1、2 项分别表示左、右子树分裂后产生的增益; 第 3 项为不进行子树分裂的增益。

2 基于多模型融合 Stacking 集成学习方式的负荷预测模型分析

集成学习(ensemble learning)是一种建立在统计学习理论基础之上的多算法融合的机器学习方法, 一般情况下, 对于单一预测模型而言, 其预测准确率是呈现边际效用递减的趋势, Stacking 集成学习方式是组合来自多个预测模型的信息以生成新模型的模型集成技术。将不同的机器学习算法通过不同的方式结合在一起, 以此获得比单一算法更优越的性能。在 Stacking 集成学习模型中, 既要分析每个基学习器的单独预测能力, 同时也要综合比较各个基学习器的组合效果, 使得 Stacking 集成学习模型获得最佳预测效果。

立足于基学习器的预测能力而言, 本文在 Stacking 模型第 1 层中除了选择 XGBoost 算法与 LSTM 算法以外, 同时也选择了预测性能较为优异的若干模型作为基学习器。这是因为学习能力较强的基模型有助于模型整体的预测效果提升。其中,

随机森林(random forest, RF)和梯度提升决策树(gradient boosted decision tree, GBDT)分别采用 Bagging 和 Boosting 的集成学习方式, 有着出色的学习能力与严谨数学理论支撑, 在各个领域获得了广泛的应用。支持向量机(support vector machine, SVM)对于解决小样本、非线性及高维度的回归问题表现特有的优势。KNN 因为其理论成熟、训练高效等特点也有着良好的实践应用效果。第 2 层选择泛化能力较强的模型, 从中归纳出并纠正多个学习算法对于训练集的偏置情况, 并通过集合方式防止过拟合效应出现。综上所述, Stacking 集成模型第 1 层初步选择基学习器分别为 XGBoost、LSTM、GBDT、RF、KNN、SVM, 第 2 层选择 XGBoost 作为元学习器, 模型架构如图 3 所示。

另一方面, 为了获得最优预测效果, 在 Stacking 模型第 1 层中还需要选择差异度较大的模型作为基学习器。这是考虑到针对不同的算法模型, 其本质是在不同的数据空间角度和数据结构角度来观测数据, 再依据算法的观测状况及自身算法原理建立相应模型。因此, 选择差异度较大的算法能够最大程度体现不同算法的优势, 使得各个差异化模型取长补短。本文采用 Pearson 相关系数对各个模型的误差差异度进行计算, 以此分析不同基学习器的关联程度, 二维向量的 Pearson 相关系数计算方法为

$$r_{xy} = \frac{\sum_{i=1}^m (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^m (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^m (y_i - \bar{y})^2}} \quad (15)$$

式中 \bar{x} 和 \bar{y} 分别为各向量中元素的平均值。

值得注意的是, 元学习器的训练集是由基学习器的输出产生的, 直接使用基学习器的训练集来产生次级训练集的话, 可能会产生严重的过拟合。为了防止数据被双层学习器重复学习, 避免“过拟合”效应的发生, 需要合理的对数据使用过程进行划分。根据所选择的 6 个基学习器, 需要先将原始训练数据集按照时间维度分为 6 个子数据集, 并保证每一块数据 ID 互不重叠, 对于单一基学习器, 使用 1 个数据块作为验证集, 对应的其余 5 个数据块作为训练集。每个基学习器针对自己的测试数据集都可输出一个预测结果, 并且这 6 个结果最终又可合并为新的数据集, 新生成数据集与原始数据集大小相同, 如图 3 所示。从而实现了所有数据从输

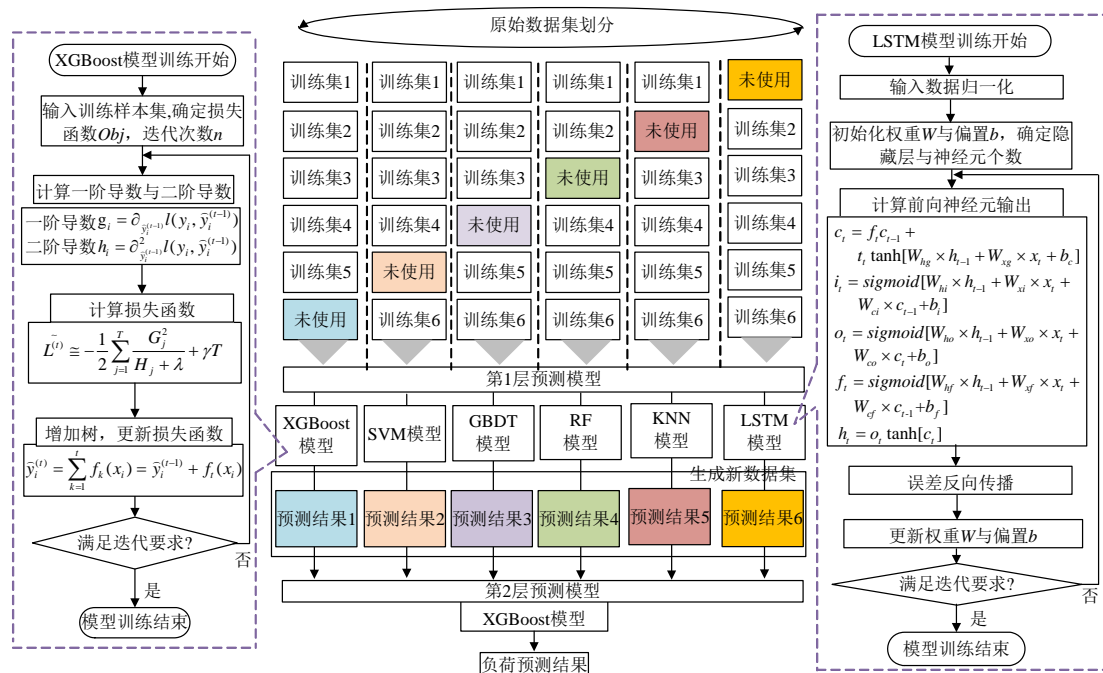


图3 Stacking 框架下基于多模型融合的负荷预测方法

Fig. 3 Load forecasting research based on multi-model under Stacking framework

入特征到输出特征的特征变换, 而且因为每个基学习器预测的数据块均未参与到该学习器的训练, 这样的配置使得所有数据只在模型训练时使用了一次, 有效防止过拟合的发生。

此外, 特征选取往往是建立模型极为重要的一步, 特征的好坏对预测效果起到决定性作用。一般情况下, 常规方法是根据人工经验进行特征工程构建。本文采用人工经验与模型评分相结合的方式选择对特征进行选择。模型评分的计算是使用了树集成模型的特征贡献度分析功能, XGBoost、RF、GBDT 模型可以在训练的过程中根据树的增益情况, 计算各个特征的评分, 表明每个特征对模型训练的重要性, 该功能有助于特征选择的有效性验证与辅助决策, 使得建模过程中保留优秀特征, 删除冗余特征。

Stacking 框架下基于多模型融合的负荷预测方法训练流程如下:

- 1) 使用 XGBoost、GBDT、RF 算法对输入特征贡献度进行分析, 辅助特征选择。
- 2) 分析各个算法的误差分布情况, 选择差异较大算法作为第 1 层预测模型。划分原始数据集, 使用交叉验证方式, 优选各个模型的最优超参数。
- 3) 使用划分后的数据集对 Stacking 中的第 1 层预测算法分别训练, 并输出预测结果, 生成新的数据集。
- 4) 使用新生成的数据集, 对 Stacking 中第 2

层算法进行训练, 基于多模型融合的 Stacking 集成学习算法训练完毕。

3 算例分析

实验数据中的负荷数据来源于 ENTSO(Europe electricity transmission system operator)所提供的瑞士用电负荷情况, 天气数据采用 Dark Sky 网站所提供的天气数据。其中, 2015 年 1 月至 2016 年 12 月数据作为训练数据, 训练数据包含了每天 24 个采样点的负荷情况(采样周期 1h)。预测目标为未来一小时的负荷情况, 预测时段为 2017 年的负荷状况。相关模型在 Matlab 2017a 与 Python3.6 环境下完成编程计算。预测评价指标采用平均相对误差 e_{MAPE} (mean absolute percentage error, MAPE), 相对均方误差 e_{RMSE} (root mean square error, RMSE)和最大误差 e_{MAX} , 如下所示:

$$e_{MAPE} = \frac{1}{n} \sum_{i=1}^n \left(\frac{x(i) - y(i)}{x(i)} \right) \times 100\% \quad (16)$$

$$e_{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n [x(i) - y(i)]^2} \quad (17)$$

$$e_{MAX} = \max[x(i) - y(i)] \quad (18)$$

式中: $x(i)$ 和 $y(i)$ 分别表示 i 时刻的实际值和预测值; n 为样本数量。

3.1 输入特征贡献度分析

首先, 通过人工经验选择负荷历史信息、天气信息、日历规则作为输入数据。其中, 负荷信息包

含了前 7h 历史负荷数据, 天气数据包含了瑞士 8 个最大城市的温度信息, 及每个城市气象状况映射的 3 种天气类型, 日历规则信息包含了预测目标对应的小时、星期、月度及是否属于节假日。所选输入信息及处理方式如表 1 所示。为了防止各个连续数据间互相影响, 对于映射天气类型、小时、星期、月度等特征采用了独热编码(one-hot)的离散处理形式。XGBoost、GBDT、RF 算法计算的特征贡献度如图 4 所示。

表 1 输入属性形式及编号
Tab. 1 Form and serial number of input attribute

特征范畴	特征属性	处理方式	编号
历史信息	前 7h 历史负荷数据	连续	1—7
	温度信息	连续	8—15
天气信息	天气映射类型	离散	16—9
	节假日	离散	40
日历规则	星期	离散	41—47
	小时	离散	48—71
	月度	离散	72—83

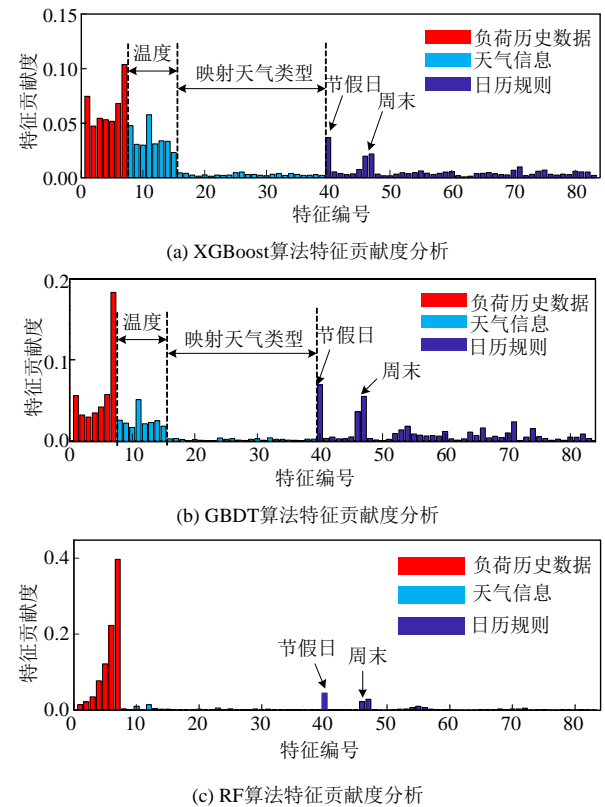


图 4 不同算法的特征贡献度分析

Fig. 4 Feature contribution research based on different algorithms

由图 4 可知, 负荷历史信息往往对预测目标产生较大影响, 越临近预测目标的历史负荷信息特征重要性越高, 预测目标前 1h 的历史负荷数据特征贡献度最高。同时, 温度信息、节假日与周末对用

电负荷影响也较大。考虑到映射天气类型、小时、星期、月度等特征采用了独热编码形式, 使得特征贡献度较为稀疏分散, 但是对于模型训练都起到一定作用。通过对模型特征贡献度分析, 侧面验证了本文特征选择的有效性。

此外, XGBoost、GBDT 和 RF 是根据树的增益情况获得特征贡献度评分, 其贡献度评分是与各个特征在模型训练中的利用效率直接相关的。对图 4 进一步分析可知, XGBoost 与 GBDT 的特征贡献度较为相似, 该 2 类算法能够充分挖掘输入信息的数据内部特征, 对连续型和离散型特征都有较好的应用。而 RF 主要关注了影响较大的几种特征类型, 对于离散稀疏型特征并不敏感。其观测数据方式与特征利用情况对训练结果也产生一定影响, XGBoost 与 GBDT 的训练结果往往好于 RF 算法(如表 2)。

表 2 各算法的超参数及其单独预测的误差
Tab. 2 Hyper-parameters and forecasting error of different single algorithms

算法名称	超参数集	单模型预测误差		
		$e_{MAPE}/\%$	e_{RMSE}/MW	e_{MAX}/MW
XGBoost	树深度 6, 学习率 0.09, 树的数目 600, 最小叶子节点样本权重 4, 随机采样的比例 0.08, 叶子节点分裂最小损失 0.1, 模型复杂度的权重值 30	1.68	156.54	984.43
	LSTM 神经元数量 40、20, 隐藏层数量 2	1.78	172.14	1027.32
GBDT	树深度 5, 学习率 0.1, 树的数目 300, 最小叶子节点样本权重 4, 随机采样的比例 1	1.80	184.14	1220.54
	RF 叶子节点最小样本数量 1, 节点划分最小样本数量 2, 树的数目 50	1.92	189.51	1436.80
SVM	核函数为径向基函数, 惩罚系数 100, 核函数系数 10^{-4}	2.03	198.47	1398.49
KNN	邻居数量 5	2.28	219.75	1432.20

3.2 各模型的超参数选择与关联度分析

为了使得 Stacking 性能达到最优, 需要立足于各个基模型学习能力与各模型之间关联度两方面展开分析。

首先, 设计实验将各个基学习器在原始数据集上单独预测的结果进行比较分析。针对各个基学习器采用交叉验证方式, 将划分后的数据进一步分为训练集与验证集, 分别观测使用不同超参数集训练后, 模型在验证集的预测效果, 从而选择各个模型的最优超参数集。以四季度(10、11、12 月)数据为例, 各模型超参数集以及单模型预测

性能如表2所示。

由表2可知,各算法单独进行预测时,XGBoost的预测误差较小,这是因为XGBoost对损失函数进行了二阶泰勒展开,优化过程使用了一阶和二阶导数信息进行更新迭代,使得模型训练更充分。此外,考虑到负荷数据的变化是一个连续的过程,当前时刻的负荷与上一时刻的负荷之间并不是相互独立的。所以,每一时刻的负荷变化不仅依赖于当前时刻的输入特征,而且与过去的输入特征也有关。LSTM算法可以有效使用前几次训练的累积信息,预测性能也较为优越。

Stacking框架集成了多样化的预测算法,能够充分利用各个算法从不同的数据空间和结构来观测数据。所以,第1层的基学习器除了要选择性能优异的算法之外,要尽可能的加入不同种类的预测算法。为了选择最佳的基模型进行组合,分别设计实验将各个基学习器单独进行负荷预测所产生的预测误差分布情况进行比较,采用二维向量的Pearson相关系数为相关性指标,各算法误差相关性分析如图5所示。

由图5可知,各算法的误差相关普遍较高,这是由于各算法学习能力较强,训练中学习数据中的

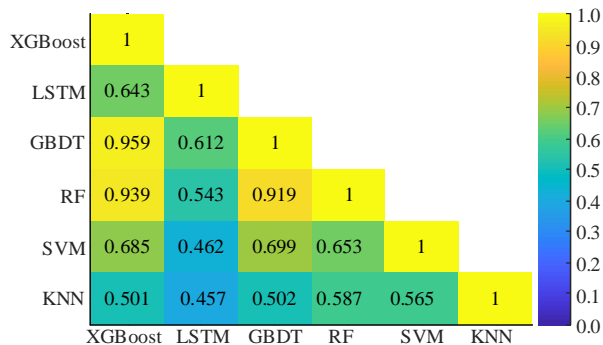
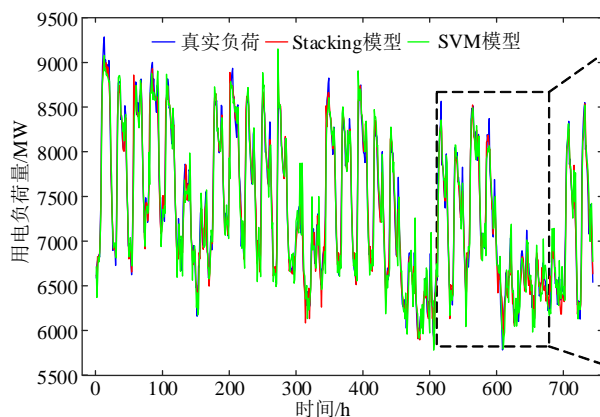


图5 各模型预测误差的相关性分析

Fig. 5 Correlation analysis of forecasting error for each model



(a) 3月负荷预测结果

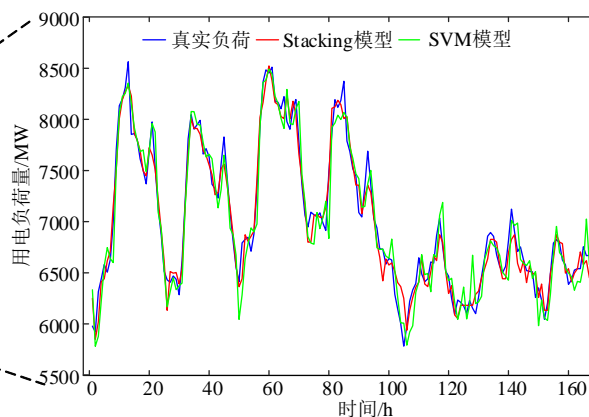
固有误差不可避免。其中,XGBoost、GBDT、RF算法的误差相关度最高,这是因为该3类算法虽然原理有些许不同,总体上仍然属于树的集成算法,其数据观测方式存在较强相似性。LSTM、SVM、KNN的训练机理差距较大,因此误差相关性也较低。因此,选择XGBoost、LSTM、SVM、KNN 4种算法作为Stacking集成学习中的基模型。

3.3 Stacking模型预测性能分析

为了验证Stacking集成学习模型的预测性能,将单模型(选择SVM)与Stacking集成模型(选择XGBoost、LSTM、SVM、KNN)预测效果分别进行对比分析。考虑到不同季节、不同月度负荷形态各异,分别选取了具有代表性月度的负荷数据验证算法可行性。其中图6(a)、(c)、(e)、(g)为3月、6月、9月、12月全月的负荷预测情况,图6(b)、(d)、(f)、(h)选取了3月、6月、9月、12月中某一周的负荷预测情况。

由图6可知,不同月度之间的负荷曲线形态有着明显差异,12月、3月的用电负荷整体高于6月、9月。在各个月度的负荷预测中,采用Stacking集成模型的预测效果均优于传统单模型SVM预测效果。值得注意的是,周末、节假日、工作日用电量也各不相同。在平日中,2种方法预测精度差异并不明显,Stacking集成模型预测结果略好于单模型SVM。但是在节假日期间,如12月最后一周(圣诞节假期),可以看到负荷曲线波动性明显增大,当负荷形态产生较大变化时,单模型SVM对于随机性更强的节假日负荷变化不敏感。此时,Stacking模型仍能较好跟踪负荷的变化趋势。

从理论层面分析Stacking集成模型优于单模型的原因,是因为Stacking模型充分发挥各个算法自身优势,摒弃了各个算法中预测效果较差的环节。此外,考虑到负荷预测模型训练过程中的假设空间



(b) 3月某周负荷预测结果

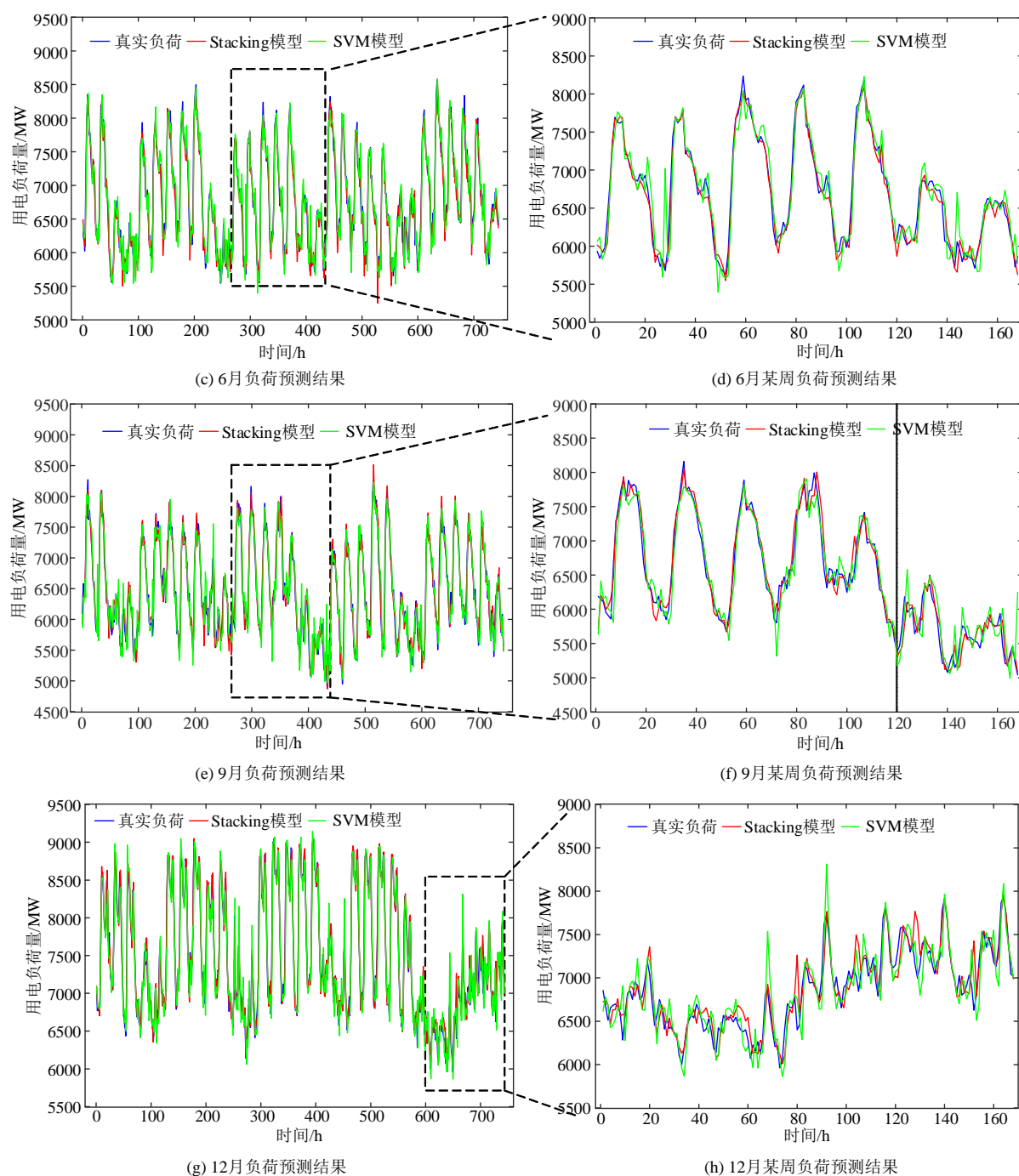


图 6 单模型与集成学习模型预测误差的对比

Fig. 6 Forecasting error analysis between ensemble learning model and single model

往往很大, 可能有多个假设在训练集上达到同等性能, Stacking 集成学习的方式有效减少单一模型泛化性能不佳的风险。另一方面, 从模型优化角度来看, 单一模型训练的优化过程中, 模型往往会有陷入局部最小点的风险, 有的局部极小点所对应的模型泛化性能可能较差, 而通过多个基学习器运行之后进行结合, 可有效减少陷入局部极小点的风险。因此, 采用集成学习方式后预测精度有所提升。

为了进一步验证 Stacking 集成模型中基学习器选择对预测结果的影响, 表 3 分别给出不同季度、

不同基学习器组合方式的预测结果。其中, Stacking 模型 1 使用了随机选择基学习器组合方式, Stacking 模型 2 使用了相关性最小基学习器组合方式(选择 XGBoost、LSTM、SVM、KNN)。

预测结果显示, 使用不同的基学习器对预测结果有较大影响, 除了在第 3 季度中两种组合方式的预测精度基本持平以外。在其他季度中, 使用相关性最小基学习器的 Stacking 模型都比随机选择基学习器的 Stacking 模型表现更优异。一方面是由于选择学习能力强的基学习器能够整体提升 Stacking 模

型的预测能力。另一方面是因为针对不同的机器学习算法，其本质是在不同的数据空间观测数据，再根据自身算法规则构建相应模型。而选择差异度较大的算法能够最大程度体现不同算法的优势，基模型的多样性和差异性使得集成结果会更加稳健、精确，从而使得预测效果获得更大的提升。

表 3 不同基模型组合方式的 Stacking 算法负荷预测误差

Tab. 3 Forecasting error of Stacking models based on different base-learner aggregation style

季度	模型名称	包含算法名称	$e_{MAPE}/e_{RMSE}/e_{MAX}/$		
			%	MW	MW
一季度 (1、2、3 月份)	Stacking 模型 1 (随机选择基学习器)	RF、SVM、 KNN	1.51	160.5 4	1002.31
	Stacking 模型 2 (本文模型)	XGBoost、 LSTM、SVM、 KNN	1.45	144.3 0	856.22
二季度 (4、5、6 月份)	Stacking 模型 1 (随机选择基学习器)	RF、KNN	1.97	178.5 4	1251.31
	Stacking 模型 2 (本文模型)	XGBoost、 LSTM、SVM、 KNN	1.71	156.4 7	1176.65
三季度 (7、8、9 月份)	Stacking 模型 1 (随机选择基学习器)	GBDT、LSTM、 RF、 KNN	1.86	168.2 9	1198.95
	Stacking 模型 2 (本文模型)	XGBoost、 LSTM、SVM、 KNN	1.85	165.6 5	1214.43
四季度 (10、11、12 月份)	Stacking 模型 1 (随机选择基学习器)	LSTM、SVM、 KNN	1.67	169.1 9	1008.50
	Stacking 模型 2 (本文模型)	XGBoost、 LSTM、SVM、 KNN	1.52	149.2 1	942.12

4 结论

本文借鉴了人工智能及机器学习领域的前沿算法技术。在 Stacking 集成算法模型中，充分利用不同算法从不同角度对数据空间与结构进行观测，使得不同算法能够取长补短，获得最优预测结果。通过算例测试表明，模型建立之前进行特征贡献度分析，能够有效量化各个特征的重要性。包含了 LSTM 与 XGBoost 的 Stacking 集成学习算法在负荷预测中有着良好的精度，在电力负荷预测方面有较高应用价值。

在今后的工作中，将进一步针对以下问题开展深入探讨，Stacking 的框架设计比较复杂，对于基模型要训练多次，即使在训练 Stacking 模型的时候对每个基学习器减少若干数据量，计算时间仍然较长。因此，未来研究中有必要布置分布式计算的相

关环境，对不同基模型分别建模，采用分而治之的思想，有效减小算法时间复杂度。

参考文献

[1] 薛禹胜, 赖业宁. 大能源思维与大数据思维的融合: (一) 大数据与电力大数据[J]. 电力系统自动化, 2016, 40(1): 1-8.
Xue Yusheng, Lai Yening. Integration of macro energy thinking and big data thinking: part one big data and power big data[J]. Automation of Electric Power Systems, 2016, 40(1): 1-8 (in Chinese).

[2] 王继业, 季知祥, 史梦洁, 等. 智能配用电大数据需求分析与应用研究[J]. 中国电机工程学报, 2015, 35(8): 1829-1836.
Wang Jiye, Ji Zhixiang, Shi Mengjie, et al. Scenario analysis and application research on big data in smart power distribution and consumption systems [J]. Proceedings of the CSEE, 2015, 35(8): 1829-1836(in Chinese).

[3] 尚宇炜, 马钊, 彭晨阳, 等. 内嵌专业知识和经验的机器学习方法探索(一): 引导学习的提出与理论基础 [J]. 中国电机工程学报, 2017, 37(19): 5560-5571.
Shang Yuwei, Ma Zhao, Peng Chenyang, et al. Study of a novel machine learning method embedding expertise part I: proposals and fundamentals of guiding learning [J]. Proceedings of the CSEE, 2017, 37(19): 5560-5571(in Chinese).

[4] 史佳琪, 张建华. 基于深度学习的超短期光伏精细化预测模型研究[J]. 电力建设, 2017, 38(6): 28-35.
Shi Jiaqi, Zhang Jianhua. Ultra short-term photovoltaic refined forecasting model based on deep learning [J]. Electric Power Construction, 2017, 38(6): 28-35(in Chinese).

[5] 魏东, 龚庆武, 来文青, 等. 基于卷积神经网络的输电线路区内外故障判断及故障选相方法研究[J]. 中国电机工程学报, 2016, 36(S1): 21-28.
Wei Dong, Gong Qingwu, Lai Wenqing, et al. Research on internal and external fault diagnosis and fault-selection of transmission line based on convolutional neural network[J]. Proceedings of the CSEE, 2016, 36(S1): 21-28(in Chinese).

[6] 国务院. 国务院关于印发新一代人工智能发展规划的通知[R]. 北京: 国务院, 2017.
State Council. Circular of the state council on the issuance of a new generation of artificial intelligence development plan [R]. Beijing: State Council, 2017(in Chinese).

[7] 张素香, 赵丙镇, 王风雨, 等. 海量数据下的电力负荷短期预测[J]. 中国电机工程学报, 2015, 35(1): 37-42.
Zhang Suxiang, Zhao Bingzhen, Wang Fengyu, et al. Short-term power load forecasting based on big

- data[J]. Proceedings of the CSEE, 2015, 35(1): 37-42(in Chinese).
- [8] 康重庆, 牟涛, 夏清. 电力系统多级负荷预测及其协调问题: (一)研究框架[J]. 电力系统自动化, 2008, 32(7): 34-38.
- Kang Chongqing, Mu Tao, Xia Qing. Power system multilevel load forecasting and coordinating part one: research framework[J]. Automation of Electric Power Systems, 2008, 32(7): 34-38(in Chinese).
- [9] Khotanzad A, Afkhami-Rohani R, Maratukulam D. ANNSTLF- artificial neural network short-term load forecaster- generation three[J]. IEEE Transactions on Power Systems, 1998, 13(4): 1413-1422.
- [10] 谢敏, 邓佳梁, 吉祥, 等. 基于信息熵和变精度粗糙集优化的支持向量机降温负荷预测方法[J]. 电网技术, 2017, 41(1): 210-214.
- Xie Min, Deng Jialiang, Ji Xiang, et al. Cooling load forecasting method based on support vector machine optimized with entropy and variable accuracy roughness set[J]. Power System Technology, 2017, 41(1): 210-214 (in Chinese).
- [11] 沈沉, 秦建, 盛万兴, 等. 基于小波聚类的配变短期负荷预测方法研究[J]. 电网技术, 2016, 40(2): 521-526.
- Shen Chen, Qin Jian, Sheng Wanxing, et al. Study on short-term forecasting of distribution transformer load using wavelet and clustering method[J]. Power System Technology, 2016, 40(2): 521-526(in Chinese).
- [12] 吴潇雨, 和敬涵, 张沛, 等. 基于灰色投影改进随机森林算法的电力系统短期负荷预测[J]. 电力系统自动化, 2015, 39(12): 50-55.
- Wu Xiaoyu, He Jinghan, Zhang Pei, et al. Power system short-term load forecasting based on improved random forest with grey relation projection[J]. Automation of Electric Power Systems, 2015, 39(12): 50-55(in Chinese).
- [13] Li Guangye, Li Wei, Tian Xiaolei, et al. Short-term electricity load forecasting based on the XGBoost algorithm[J]. Smart Grid, 2017, 7(4): 274-285.
- [14] Ryu S, Noh J, Kim H. Deep neural network based demand side short term load forecasting[J]. Energies, 2017, 10(1): 3.
- [15] Kong Weicong, Dong Zhaoyang, Jia Youwei, et al. Short-term residential load forecasting based on LSTM recurrent neural network[J]. IEEE Transactions on Smart Grid, 2019, 10(1): 841-851.
- [16] 梁智, 孙国强, 李虎成, 等. 基于 VMD 与 PSO 优化深度信念网络的短期负荷预测[J]. 电网技术, 2018, 42(2): 598-606.
- Liang Zhi, Sun Guoqiang, Li Hucheng, et al. Short-term load forecasting based on VMD and PSO optimized deep belief network[J]. Power System Technology, 2018, 42(2): 598-606(in Chinese).
- [17] 史佳琪, 谭涛, 郭经, 等. 基于深度结构多任务学习的园区型综合能源系统多元负荷预测[J]. 电网技术, 2018, 42(3): 698-706.
- Shi Jiaqi, Tan Tao, Guo Jing, et al. Multi-task learning based on deep architecture for various types of load forecasting in regional energy system integration [J]. Power System Technology, 2018, 42(3): 698-706(in Chinese).
- [18] 刘文霞, 龙日尚, 徐晓波, 等. 考虑数据新鲜度和交叉熵的电动汽车短期充电负荷预测模型[J]. 电力系统自动化, 2016, 40(12): 45-52.
- Liu Wenxia, Long Rishang, Xu Xiaobo, et al. Forecasting model of short-term EV charging load based on data freshness and cross entropy[J]. Automation of Electric Power Systems, 2016, 40(12): 45-52(in Chinese).
- [19] Li Zhiyi, Liu Xuan, Chen Liyuan. Load interval forecasting methods based on an ensemble of Extreme Learning Machines[C]//Proceedings of 2015 IEEE Power & Energy Society General Meeting. Denver, CO, USA: IEEE, 2015: 1-5.
- [20] Zhang Rui, Dong Zhaoyang, Xu Yan, et al. Short-term load forecasting of Australian national electricity market by an ensemble model of extreme learning machine [J]. IET Generation, Transmission & Distribution, 2013, 7(4): 391-397.
- [21] Wolpert D H. Stacked generalization[M]. Boston: Springer, 2017: 6-10.



史佳琪

收稿日期: 2018-09-04.

作者简介:

史佳琪(1990), 男, 博士研究生, 研究方向为分布式电源及负荷预测, 区域综合能源系统优化调度, phdshijiaqi@foxmail.com;

张建华(1952), 男, 教授, 博士生导师, IET 会士, 主要研究方向为配电网规划、电力系统安全应急等。

(责任编辑 乔宝榆)

Load Forecasting Based on Multi-model by Stacking Ensemble Learning

SHI Jiaqi, ZHANG Jianhua

((North China Electric Power University))

KEY WORDS: artificial intelligence; load forecasting; multi-model combination; Stacking ensemble learning; XGBoost; long-short term memory

The rapid development of artificial intelligence and machine learning technology provides an innovative solution for load forecasting. A load forecasting method based on multi-model combination under Stacking framework is proposed in this paper, associated with the frontier theory research of artificial intelligence. Firstly, the mechanism of Stacking ensemble learning is introduced. The XGBoost algorithm constructed by the tree model and the deep learning algorithm represented by long and short memory network are presented. Then considering the difference of data observation and training principles, the Stacking based load forecasting model embedded various machine learning algorithms is proposed to utilize their diversified strength.

In Stacking Framework. Firstly, the original dataset is divided into several sub-datasets which are used to train each base-learner at the first layer of forecasting model. Then, the output of the first layer is deployed as the input of the second layer to train the meta-learner. The Stacking framework considerably summarizes the output of multi-model to improve the overall forecasting performance.

We use the Swiss load data in ENTSO to verify the effectiveness of the algorithm. Stacking framework integrates a variety of forecasting algorithms, which can make full use of each algorithm to observe data from different perspectives. Thus, the base-learners of first layer

with diversified forecasting algorithms are essential to build Stacking model. To select the best combination of base-learner, Pearson correlation coefficients are used to assess the distribution of forecasting errors produced by different set of base-learners. Correlation analysis of forecasting error for each model is shown in Fig. 1. The load forecasting results are more accurate when each of base-learner has lower correlation coefficient in Stacking.

To further verify the learning and generalization capabilities of the proposed model, the same input dataset is used to make a comparative study on forecasting results between Stacking ensemble model and single model (SVM). Fig. 2 illustrates the load forecasting curve in June. The results indicate the Stacking ensemble learning based on multi-model has better prediction performance compared with the traditional single model.

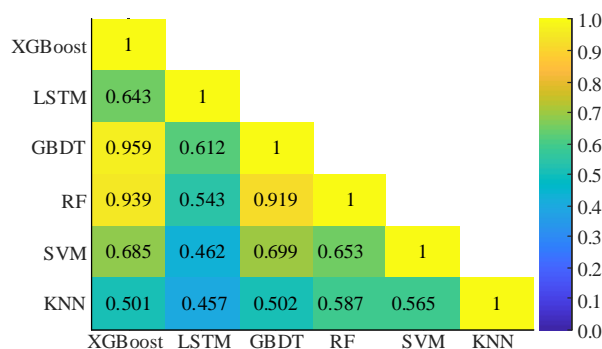


Fig. 1 Correlation analysis of forecasting error for each model

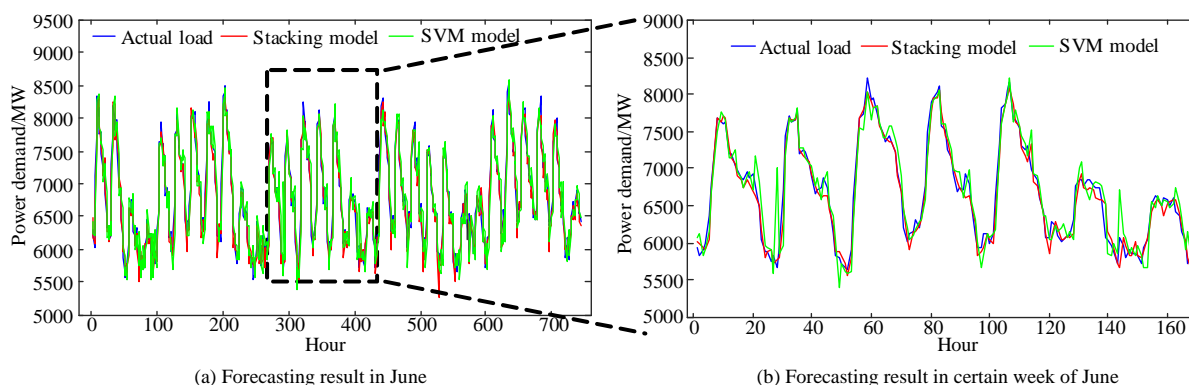


Fig. 2 Forecasting error analysis between ensemble learning model and single model