基于多模型融合预测算法的智能排产系统

孙 平[1]
SUN Ping

摘 要 在智能制造领域排产方案对提升生产效率至关重要。尽管现有的排产系统已取得一定成效，但未能充分挖掘生产数据的复杂特征和机台运行时长的关联性，限制了排产效果。为此，文章提出了一种多模型融合预测算法，并将其应用于机台智能排产方案推荐系统。该系统将历史运行数据通过标准化处理和特征提取转化为高质量的特征集，利用融合机制综合多模型的优势以精准预测机台的运行时长；基于该预测模型，系统能结合用户设定的生产约束条件，动态生成优化排产方案，最大化设备利用率和生产效率。实验结果验证了该系统在提升机台运行时长预测准确性方面的有效性。

关键词 智能排产；运行时长预测；多模型融合

doi：10.3969/j.issn.1672-9528.2025.10.001

0 引言

工业互联网作为新一轮工业革命的核心驱动力，正深刻推动制造业向智能化、个性化方向转型，尤其在多品种、小批量、高时效的定制化生产场景，例如机械制造、电子加工领域[1]。在生产管理中，传统人工排产方法依赖固定的工时数据来估算作业时间，忽视设备作业时间差异和订单交付期不同，导致排程不准确且复杂，难以高效处理大量数据[2]。在此背景下，工业互联网通过实时数据采集、大数据与 AI 等关键技术，融合设备状态、订单优先级等多维约束，为智能排产提供了动态优化基础，从而实现生产计划的精准可控与实时响应，有效解决个性化定制中的柔性化生产需求，显著提升制造效率与企业竞争力。

机台运行时长预测是智能排产领域的关键环节。传统的时长预测方法包括基于用户预估的方法和线性回归方法，如使用线性回归、多元线性回归或向量回归方法基于历史数据得到预测模型[3-5]。这些方法适用于数据简单的场景，但是对于非线性关系建模能力不足。随着计算能力的提升，机器学习和深度学习方法在时长预测中得到了广泛应用。常用的算法有支持向量机[6]、决策树[7]和随机森林[8-9]等；另外多层感知机 MLP 算法能够有效建模非线性关系，并通过增加层数来提高模型的表达能力[10]。尽管这些方法在特定场景下取得了一定进展，但仍存在明显局限性。例如，现有算法对生产数据中复杂特征的整合能力不足，难以充分考虑设备历史运行状态、工艺流程约束以及动态生产资源等多维信息；此外，前人研究大多基于单个学习器预测作业时长，难以避免单个机器学习模型泛化性差的问题[11]。当前，如何构建适应复杂生产环境并有效融合多源数据的预测与排产模型，已成为智能制造领域急需攻克的难题。

为此，本文提出了一种应用于智能排产方案推荐系统的多模型融合机台运行时长预测算法。该方法在特征数量有限的条件下，通过融合不同机器学习模型的优势进行预测，有效提升了预测性能。此外，本文设计了一种模拟机台运行波动的方案，能够真实地模拟实际生产环境中的波动特性，从而增强预测模型的适用性与鲁棒性。在智能排产方面，本文提出了一种支持用户自定义排产优先级的贪心算法，该方法支持灵活调整排产策略，适应不同企业的多样化生产需求。综合而言，所提出的方法不仅显著提升了机台运行时长预测的准确性和鲁棒性，还通过优化排产方案为企业智能化生产提供了高效的决策支持。

1 智能排产总体模型

智能排产总体模型分为数据输入层、预处理层、算法模型排产层和输出层。数据输入层通过多个接口以 JSON 格式接收机台运行数据、工单信息和机台信息（包括已占用机台）。预处理层对接收到的数据进行标准化处理、特征提取和编码，生成符合算法模型需求的输入特征。算法模型排产层包括 MFP（multi-model fusion prediction）机台运行时长预测模块和智能排产模块。MFP 模型输出单个机台的运行时长预测。智能排产模块通过 MFP 模型预测完单时间内的所有机台预测运行时长，再结合工单信息和设备资源，利用智能排产策略生成优化的排产方案。输出层将预测的机台时长、智能排产方案以及更新后的计划完成时间通过接口返回，供用户进行后续操作。图 1 展示了机台智能排产方案推荐系统的整体架构。

1. 福州市数字产业互联科技有限责任公司 福建福州 350007

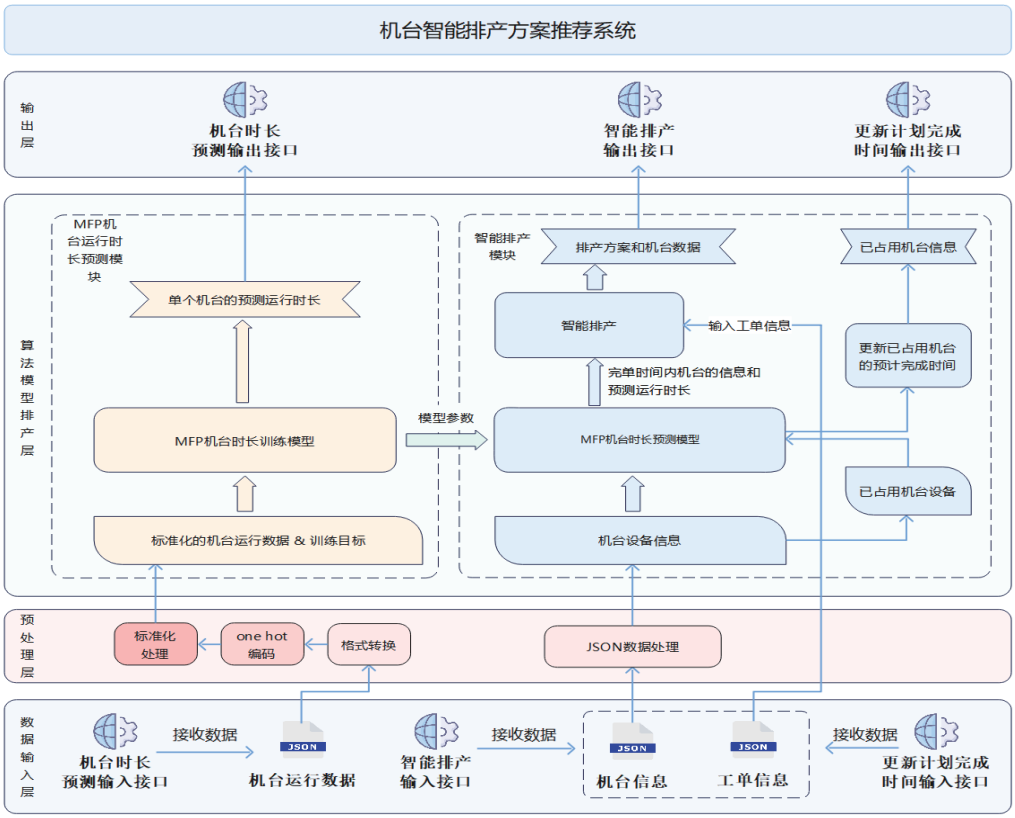

图 1 机台智能排产方案推荐系统的整体架构

1.1 数据预处理模块

在数据预处理中，首先将接收的 JSON 数据转换为 DataFrame 格式，以便进行分析和处理。随后，为提高数据质量，对异常值进行筛选和剔除例如将 Speed=0 或 Density=0 的记录移除。Device_Num（DN）表示机台编号；Run_Time（RT）表示设备的运行时长。

表 1 机台特征信息表

字段名	数据类型	约束条件	说明
Device_Num(DN)	VARCHAR(50)	主键，自增	表的唯一标识符
Date	VARCHAR(50)	非空	数据日期
Run_Time(RT)	INT	非空	设备的运行时长
Speed	FLOAT	非空，且不为零	设备的平均转速
Density	FLOAT	非空，且不为零	产品的平均密度

MFP 模型先对 DN 进行 One-Hot 编码。One-Hot 编码是一种将分类数据（类别变量）转换为数值形式的编码方法，广泛用于机器学习和深度学习中。核心思想是将每个类别映射为一个独热向量（One-Hot Vector），该向量中只有一个位置为 1，其余位置均为 0。使用 One-Hot 编码的优势在于它可以消除类别之间的大小关系、适配机器学习算法、避免信息丢失以及灵活性强等，以便模型能够处理分类机台的设备 DN，便于模型训练时进行区分。继而，对机台的转速和密度进行均值 - 标准差标准化，以消除不同特征之间的量纲差异。如此预处理后，将数据按照表 1 格式写入机台特征信息数据集，供模型训练使用。定义 X 为机台特征集合，经过预处理后的输入 X^{scaled}={DN, Date, RT, Speed, Density}，最后将 X^{scaled} 输入到模型中。

1.2 MFP 多模型融合预测算法

虽然多层感知机（MLP）、决策树（DT）和梯度提升回归（GB）各有优点，但也存在一些局限性：MLP 容易受到训练数据量和计算资源的限制，且在特征较少时可能表现不如预期；决策树则容易出现过拟合，尤其在数据噪声较多时，且对连续特征的处理较为简单；梯度提升回归虽然拟合能力强，但在数据噪声较多时可能导致过拟合，并且训练时间较长。结合企业机台特性信息情况，本文设计了一种融合策略，将 3 种模型结合使用，既发挥各自的优势，又互补不足，提升预测结果的准确性和鲁棒性。

MFP 模型包括训练阶段和预测阶段。在训练阶段，首先对预处理后的机台特征信息 X^{scaled} 中的密度和转速进行波动处理，得到波动后的机台特征数据 X^{fluc} 输入模型进行训练。由于单次波动模拟的结果可能偏离真实情况较多，故迭代 n 次后再将结果输入到融合预测结果模块进行融合。通过不断迭代训练对模型进行调优。在训练阶段，模型通过反向传播和梯度更新机制，逐步调整参数，学习到特征波动对机台运行时长的影响规律。在预测阶段，将待预测机台特征数据输入到模型中，模型将输出机台运行时长预测结果。图 2 为机台运行时长预测架构图。

1.2.1 波动模拟模块

由于输入的密度（Density）和转速（Speed）是由用户自定义设定的，通常情况下用户会设置为整数。然而，在实

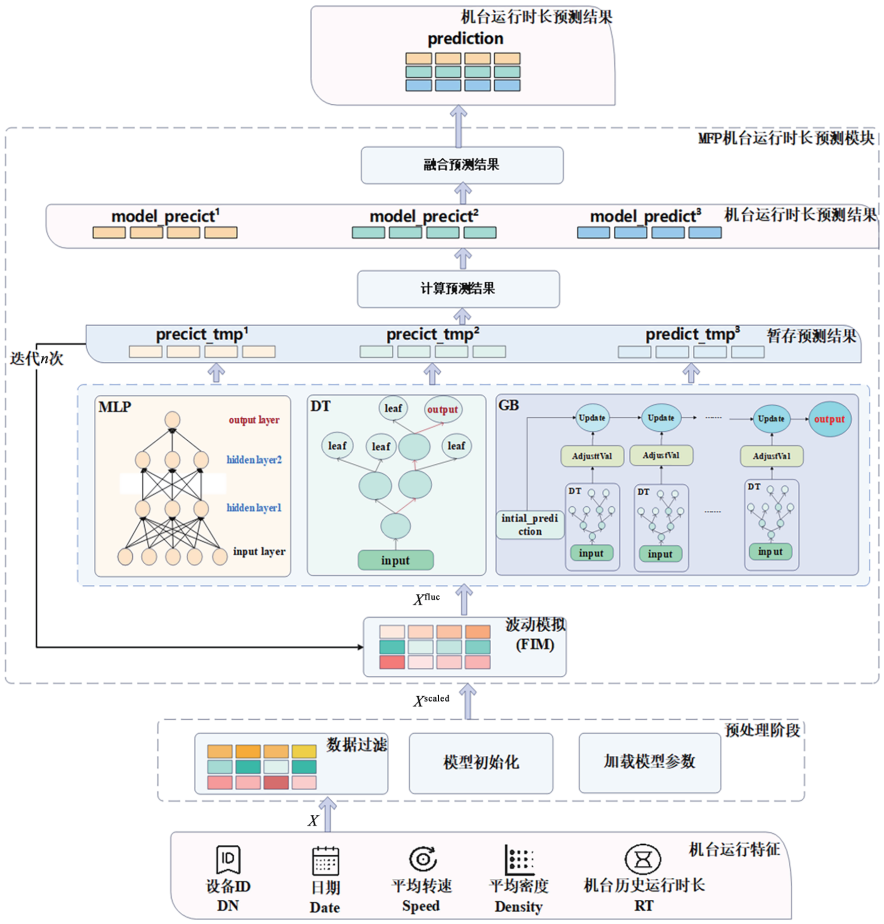

图 2 机台运行时长预测架构图

际生产过程中，由于机台的个体差异，统计出的密度和转速往往与设定值存在一定差异。这使得实际的生产环境与理论设定之间存在一定的偏差。为模拟这些实际环境中的波动情况，本文提出了一种简单有效的数据扰动处理方法，使得扰动后的数据能更真实地反映生产实际情况，提升 MFP 模型在实际生产中的预测效果。

波动模拟模块（FIM-fluctuation interference module），主要针对密度和转速特征进行波动处理，包括两个主要步骤：标准化和波动添加。为降低单次波后预测可能带来的误差，MFP 模型在预测时进行多次波动模拟，增强了预测结果的稳定性和可靠性。公式为：

（1）标准化

$$Z^{\text{scaled}} = \frac{Z-\mu}{\sigma} \tag{1}$$

式中：$Z \in \{\text{Density, Speed}\}$，表示密度或转速的原始值；$\mu$ 表示该特征在训练集中的平均值；σ 表示该特征在训练集中的标准差；Z^{scaled} 表示标准化后的结果。

（2）波动添加

$$Z_i^{\text{fluc}} = Z^{\text{scaled}} + \Delta[Z]_i \tag{2}$$

式中：$\Delta[Z]_i$表示第 i 次波动的值。

在图 2 的表述中，将密度和转速波动过后的输入表示为 X^{fluc}。

1.2.2 多层感知器模型

多层感知机（multi-layer perceptron, MLP）是一种前馈神经网络，其基本结构包括输入层、若干隐藏层和输出层，每一层由多个神经元（节点）组成。MLP 由多个全连接层（fully connected layer）堆叠而成，每一层可以表示为：

$$\boldsymbol{X}^{(l)} = \boldsymbol{W}^{(l)}\boldsymbol{X}^{(l-1)} + \boldsymbol{b}^{(l)} \tag{3}$$

式中：$\boldsymbol{X}^{(l)}$ 为第 l 层的输出（即激活值）；$\boldsymbol{X}^{(l-1)}$ 是第 l-1 层的输出，作为第 l 层的输入，其中第 0 层的输入 $\boldsymbol{X}^{\text{fluc}}$ 为经过标准化和波动模拟后特征；$\boldsymbol{W}^{(l)}$ 是第 l 层的权重矩阵；$\boldsymbol{b}^{(l)}$ 是第 l 层的偏置向量；f 是激活函数，如 ReLU、SigmoDevice_Num、Tanh 等。

MLP 使用均方误差函数 L_1 来优化预测结果。

$$L_1(y,\hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y})^2 \tag{4}$$

式中：y_i 为实际值；$\hat{y}$为平均预测值。

1.2.3 决策树模型

决策树（DT）是一种树形结构的机器学习模型，用于分类和回归任务。其基本思想是将数据集划分成更小的子集，同时对应的决策树逐渐构建完成。以下是决策树的主要公式及说明：

（1） 信息增益（information gain, IG）

决策树的分裂依据是最大化信息增益，衡量划分前后不确定性的减少程度。公式为：

$$\text{IG} = H(S) - \sum_{i=1}^{n}\frac{|S_i|}{|S|}H(S_i) \tag{5}$$

按输入的特征进行划分，共有 Density、Date、Speed、DN(DeviceNum)、RT(Run_time) 五个特征，决策树将按特征进行划分，$|S|$ 是数据集 S 基态的特征数量，即 $|S| \subseteq X$，$H(S)$ 是划分前的熵，当以某一特征（例如 Density）为基础进行划分后，数据集被分为若干个子集 $S_1, S_2, \cdots, S_k$，$H(S_i)$ 表示每个子集的熵；S_i 表示第 i 个子集的大小。通过选择信息增益最大的属性进行划分计算熵 $H(S)$ 的公式为：

$$H(S) = -\sum_{k=1}^{m} p_k \log_2(p_k) \tag{6}$$

式中：p_k 是 S 中第 k 类样本的概率。

通过计算数据集划分前后熵的差值，选择信息增益最大的特征进行划分，从而减少数据的混乱度。

（2）基尼指数（Gini Index）

基尼指数是另一种衡量不纯度的方法，其公式为：

$$G(S)=1-\sum_{k=1}^{m}p_k^2 \tag{7}$$

式中：$G(S)$ 为数据集 S 的基尼指数；p_k 为数据集中第 k 类样本的概率。

划分后的加权基尼指数为：

$$G_{\text{split}}=\sum_{i=1}^{n}\frac{|S_i|}{|S|}G(S_i) \tag{8}$$

③ 损失函数

用均方误差（MSE）评估划分效果，DT 使用该误差函数 L_2 来优化预测结果。

$$L_2(y,\hat{y})=\frac{1}{n}\sum_{i=1}^{n}(y_i-\hat{y})^2 \tag{9}$$

1.2.4 梯度提升回归模型

梯度提升（GB）是一种集成学习算法，通过逐步构建一系列弱预测模型（通常是决策树）来优化目标函数。以下是梯度提升回归的公式和算法流程。

（1）目标函数：对于回归问题，目标是最小化一个损失函数 $L_3(y, F(x))$，通常使用均方误差（MSE）作为损失函数：

$$L_3(y,F(x))=\frac{1}{n}\sum_{i=1}^{n}(y_i-F(x_i))^2 \tag{10}$$

式中：y_i 为真实值；$F(x_i)$ 为模型的预测值。

（2）预测模型的构建：梯度提升通过迭代构建模型。第 m 次迭代的预测模型表示为：

$$F_m(x)=F_{m-1}(x)+\nu\cdot h_m(x) \tag{11}$$

式中：$F_m(x)$ 是第 m 次迭代的模型预测值；$F_{m\text{-}1}(x)$ 是前 m-1 次迭代的预测值；$h_m(x)$ 是第 m 棵树（即弱学习器）的预测机台运行时长；ν 是学习率（Step Size），用于控制每棵树的贡献。

（3）梯度提升的核心：在每次迭代中，算法根据损失函数 L 的负梯方向来拟合新模型 $h_m(x)$。对回归问题，负梯度可以看作残差 r_m：

$$r_m=y_i-F_{m-1}(x_i) \tag{12}$$

式中：新模型 $h_m(x)$ 用来拟合这些残差。

（4）最终预测值： 在所有迭代完成后，最终的预测值为所有弱学习器的加权和：

$$F(x)=\sum_{m=1}^{M}\nu\cdot h_m(x) \tag{13}$$

式中：M 为树的总数。

1.2.5 融合预测结果

为了充分地融合每次波动模拟的计算结果，使用预测结果的均值，具体过程用公式表示为：

（1）预测运行时长

$$\text{model\_predict}^k=\frac{\sum_{i=1}^{n}\text{model}^k(X_i^{\text{fluc}})}{n} \tag{14}$$

式中：model_predict^k 表示最终预测运行时长结果；$\text{model}^k(X_i^{\text{fluc}})$ 表示第 k 个模型对于特征第 i 次波动后的预测结果，其中$\forall k\in\{1,2,3\}$；n 表示对数据进行模拟波动的次数，设定 MLP 模型（多层感知机）、GB 模型（梯度提升回归模型）和 DT 模型（决策树模型）分别对应于 model^1、model^2、model^3。

（2）融合模型预测结果

$$\text{prediction}=\sum_{k=1}^{3}w_k\times\text{model\_predict}^k \tag{15}$$

式中：predict 表示融合后的预测结果；w_k 表示每个模型预测结果的权重，为可学习的参数，$\forall k\in\{1,2,3\}$。

（3）计算损失

$$L=\sum_{k=1}^{3}\alpha_k\cdot L_k \tag{16}$$

式中：L 表示总损失；α_k 表示可学习的参数；L_k 表示 MLP、GB、DT 三个模型的损失；模型根据 α_k 自适应调整每个模型的损失权重，其中$\forall k\in\{1,2,3\}$。

1.3 智能排产模块

1.3.1 更新计划完成时间

由于在实际生产过程中，机台可能受到多种因素的干扰，例如设备故障、操作失误等，导致预计完成时间与实际情况产生偏差。智能排产方案推荐系统需要实时监控机台的运行状态，并结合当前生产进度和历史数据进行动态调整。故在启动智能排产前，本文设计了一个更新计划完成时间的接口，用于修正机台的预计完成时间，以便更好地支持后续的排产工作。之后通过整合模型的预测结果与实际生产状况，系统能够精准地修正预计完成时间，从而提升生产计划的准确性和可靠性。

1.3.2 用户自定义排产优先级

在排产过程中用户可以自定义排产优先级 P={ 空闲机台优先 , 占用机台优先 盘头剩余时间短优先 , 产量高优先 , 预计占用时间短优先 }。用户输入期望完成时间 expectComplete-Time，自定义排产优先级 $\text{PriotiryList}=\{p_i|\,p_i\in P,\,1\le i\le 5\}$，所有可排产机台 devices 和排产总产量 sumOutput。然后，根据用户输入的优先级 PriotiryList，先计算出每个 p_i 对应的机台排列顺序SORT_{p_i}（$1\le i\le 5$）。再依次遍历SORT_{p_i}中的所有机台，累加每个机台的预计产量到预计总产量中，如果预计总产量小于总产量则将机台加入排产列表 List 中。最终，若预计总产量大于排产总产量则将排产列表 List 返回；若小于

则无法满足排产需求。自定义优先级排产算法流程图如图 3 所示。

```
算法：自定义优先级排产算法
Input：用户自定义优先级 PriotiryList；
期望完成时间 expectCompleteTime;
排产总产量 sumOutput;
所有可排产机台 devices
Output:排产列表 List
定义：totalOutput 是预计总产量
通过 MFP 模型计算 devices 的预计产量
分别计算四个不同优先级下的机台排列顺序
SORT ={SORT_P1,SORT_P2,SORT_P3,SORT_P4,SORT_P5}
    totalOutput=0;
    //预计总产量小于总产量则进行排产
    While totalOutput<sumOutput do
      for sort in SORT do
          for device in sort do
              //判断是否满足排产需求
              If totalOutput>=sumOutput
                Return List;//返回排产列表
              Else If device 的产量+totalOutput<sumOutput &
                   device 的预计完成时间<expectCompleteTime
                   & device not in List
              将 device 的产量累加到 totalOutput;
              将 device 加入到排产列表 List;
              End If
          End for
      End for
      Return “unsuccessful”;
    End while
```

图 3 自定义优先级排产算法流程图

1.3.3 智能排产流程

在智能排产过程中，MFP 模型会根据不同策略预测机台的运行时长，并结合工单信息和机台信息计算每台机台的预测日产量。然后，用户可以自定义排产优先级，该优先级将作为排产过程中的输入参数。图 4 为其中一种机台的排产优先级：空闲机台优先〉占用机台优先〉预计占用时间短的机台优先〉盘头剩余时间短的机台优先。系统首先判断空闲机台能否完成预计的排产需求：若可以，则直接安排空闲机台进行排产；若不可以，则考虑使用占用中的机台进行排产；在考虑占用机台时系统先考虑预计占用时间短的机台能否完成排产；如果不能则再考虑以盘头剩余时间短的机台是否完成排产。占用机台的排产优先级方案，可以根据用户需求自定义输入。如果预计总产量能够达到期望产量，系统将安排机台进行排产；否则，系统会提示当前条件下无法完成排产目标。排产完成后，系统将生成的机台数据和预测结果返回前端，为用户提供详细的排产方案和相关信息。

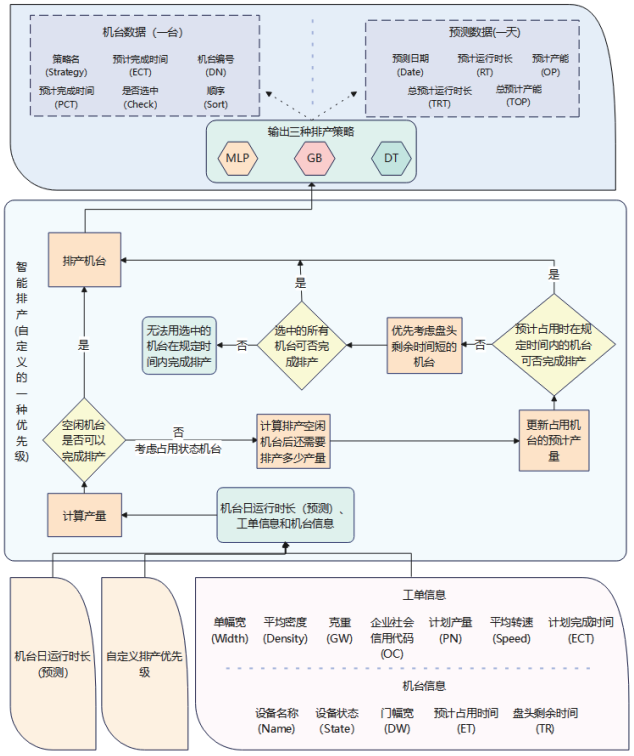

图 4 一种优先级下的智能排产过程示例

2 实验分析

在本节中将详细介绍实验验证分析，首先介绍使用所需的数据集和实验设置，然后在真实数据集上对模型进行实验验证和性能评估。

2.1 数据集

实验数据集来自 3 家真实企业 2023 年 7 月—2024 年 12 月的生产运行产生的机台特征数据，分别为 Nera、Jg 和 Js。2023-07-16—2024-11-14 的数据为训练集，2024-11-15—2024-12-14 的数据为测试集。具体数据量如表 2 所示。

表 2 数据统计表

单位：条

数据集	Nera	Jg	Js
训练集	6 269	14 960	14 395
测试集	730	1 662	1 599

本文实验使用 sklearn 框架实现，以下所有实验都在同一实验环境下进行。使用 Ubuntu20.04 操作系统，编程语言为 Python3.9，GPU 为 NVDevice_NumIA RTX3090。

2.2 评价指标

如下预测机台运行时长的评价指标。由于厂家规定预测的运行时长在 ±1 800 s 内就算合规，故提出以下评价指标，其公式为：

$$S(y_i,\hat{y}_i)=\begin{cases}1 & \text{if } |y_i-\hat{y}_i|\le 1800\\ 0 & \text{otherwise}\end{cases} \tag{17}$$

$$\text{Accuracy} = \frac{\sum_{i=1}^{n} S(y_i, \hat{y}_i)}{n} \quad (18)$$

2.3 实验结果

本文提出的算法与其他基线模型的对比结果如表 3 所示。

表 3 实验结果

模型	Nera	Jg	Js
MLP	0.842 953	0.567 393	0.422 612
GB	0.902 013	0.625 074	0.535 067
DT	0.872 483	0.546 204	0.446 191
MFP(Ours)	**0.953 020**	**0.788 111**	**0.734 583**

通过表 3 可以看出，MFP 模型在 3 个数据集上的准确率明显高于其他单一模型的预测结果。经过扰动处理后的数据在送入模型进行预测后，能够显著提高预测准确率，超出了其他 3 种单一模型的表现。

然而，在 Jg 和 Js 数据集上，模型的预测效果却相对较差。进一步分析发现，由于后两家企业对于这些机台特征数据采集不规范，导致这两个数据集包含了大量的干扰数据，最终影响了模型的性能。

2.4 消融实验

表 4 展示了 MFP 模型对于波动模拟模块消融实验的结果。

从表 4 中可以观察到，在去除 FIM 模块后，模型的表现有所下降。这一现象可以归因于引入的随机波动模块对机台特征的增强作用。具体来说，该模块通过对特征进行随机扰动，有效地增强了模型对机台特征的捕捉能力，从而改善了模型的学习过程和泛化能力。这种增强作用通过模拟不同的工作环境和条件变化，使得模型能够在面对多样化的数据时，保持更强的鲁棒性和准确性，从而显著提高了模型在预测任务中的表现。

-FIM 为消融掉波动模拟模块，加粗数据表示最优结果，下划线表示次优结果。

表 4 消融后实验结果

模型	Nera	Jg	Js
MLP-FIM	0.857 534	0.567 393	0.425 635
GB-FIM	0.745 205	0.539 729	0.536 276
DT-FIM	0.749 315	0.508 534	0.451 632
MFP-FIM	0.942 466	0.762 802	0.718 259
MFP(Ours)	**0.953 020**	**0.788 111**	**0.734 583**

2.5 案例分析

本文选取历史数据均值预测法（对于需要预测运行时长的机台，选取其在前两个月的历史数据取平均，作为运行时长预测数据，简称均值法）、DT 模型、GB 模型和 MLP 模型，这四种方法作为 MFP 模型预测效果的对比基线。选取 2024 年 11 月 15 日—12 月 14 日这一个月时间的数据作为测试集来进行实验。从训练数据集的 26 个机台中随机挑选了 3 个机台，绘制出了 3 个机台设备在这一个月时间内，上述 5 种方法得到的预测运行时长与真实运行时长的对比图，如图 5~7 所示。MFP 模型预测结果与实际情况基本吻合，预测结果的准确率远远高于其他 4 种方法。以设备 ESN_SC000246_dev101 为例，计算上述 5 种方法的预测结果与真实值之间的最大偏离比例。

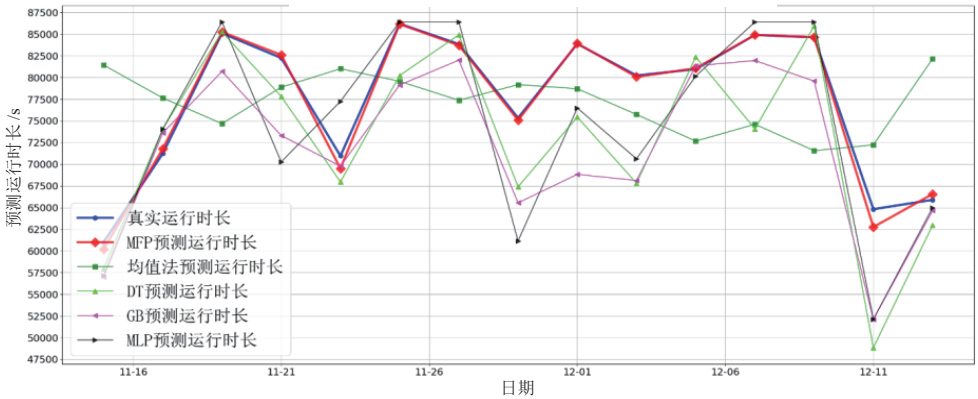

图 5 设备 ESN_SC000227_dev101 在不同模型预测下的效果对比图

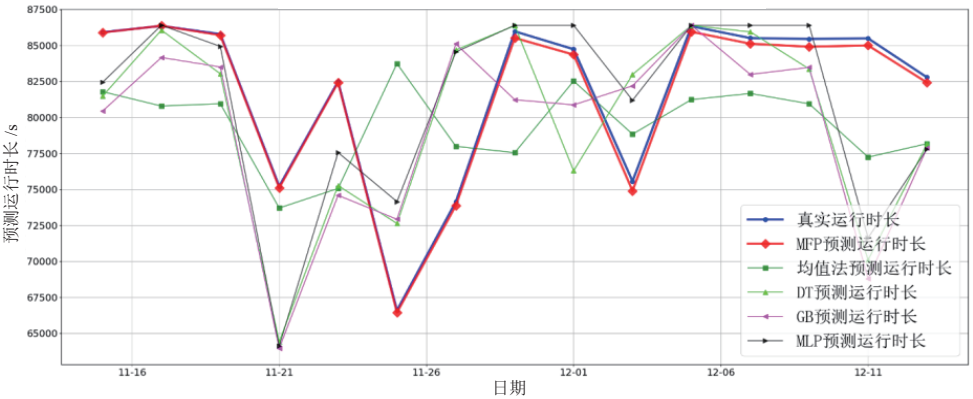

图 6 设备 ESN_SC000236_dev101 在不同模型预测下的效果对比图

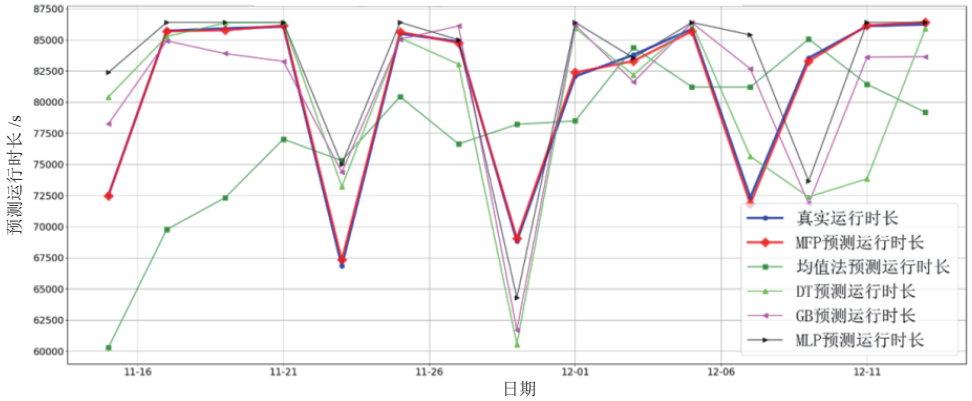

图 7 设备 ESN_SC000246_dev101 在不同模型预测下的效果对比图

在 12 月 8 日，MFP、均值法、DT 模型、GB 模型、MLP 模型的最大偏离比例分别为 2.55%、25.02%、20.80%、23.45%、23.44%。可见，MFP 模型的最大偏离比例非常小，远远优于其他方法。此外，可以观察到均值法由于只是单纯地对历史运行时长进行均值处理，很难准确测试设备在实际生产中真实运行时长。虽然 DT 模型、GB 模型和 MLP 模型也能在一定程度上预测实际生产中真实运行时长，但是这 3 个模型在某些天的预测值远远偏离于真实运行时长，这种对预测结果极端的不稳定性可能会对实际生产造成重大误差。这表明简单地选择某一个模型作为预测结果，并不能满足实际的生产需求。而将这 3 种方法有效融合的 MFP 模型可以规避这种风险，在满足高准确率的同时，还能保持较高的稳定性，能够很好地满足实际生产要求。

综上所述，MFP 模型预测方法能够更加真实地模拟机台设备运行时长在实际生产中的波动，即能够更加有效、准确地预测出机台的运行时长，相较于传统的均值预测法，MFP 模型的预测性能要显著优于传统的方法。

3 结论与展望

本文针对工业互联网背景下个性化定制生产面临的动态排产挑战，提出了一种基于多模型融合的机台时长预测算法（MFP）并将其应用于智能排产方案推荐系统。研究结果表明，MFP 模型通过精确预测机台运行时长，为智能排产模块提供了可靠的数据支持。通过综合考虑预测的时长、工单信息及设备资源状态，排产算法能够动态生成高效的排产方案，显著提高设备利用率和生产效率。实验结果表明，与传统方法相比，本文提出的 MFP 模型在基态时长预测精度上表现出显著的提升，基于此的智能排产系统成功应对了企业复杂的生产环境及动态资源约束，实现了生产周期的缩短和成本的降低。这一成果不仅验证了工业互联网技术在智能排产领域的应用价值，也为制造业向个性化、智能化转型提供了切实可行的技术方案。

尽管研究取得了一定成果，但在实际应用中仍存在一些不足。未来的工作可以围绕模型的进一步优化展开，探索更为复杂的特征提取方法和自适应调整机制，以增强算法对变化环境的响应能力。同时，考虑到不同制造场景的个性化需求，开发针对特定行业或产品的定制化排产方案也将是一个重要的研究方向。

参考文献：

[1] 丁婧伊，金嘉晖，杨丰赫，等．基于云边协作的工业互联网排产方法：以钢铁热轧生产为例 [J]. 电子学报，2024, 52(9): 2988-2999.

[2] 王纪章，刘哲，高志恒，等．基于数字孪生的拖拉机混流装配智能排产仿真评价 [J]. 农业机械学报 ,2024,55(4):385-393.

[3] 丁京，丁小进，方晏．车间智能排产模型与算法研究 [J]. 模具制造，2024, 24(4):252-254.

[4] 朱简，翟敏．基于线性回归模型的基础教育装备资源需求重要支撑数据预测研究 [J]. 中国现代教育装备，2024(22): 68-72.

[5] 侯文涛，柴鑫杰，赵晓乐．基于支持向量回归的噪声激励下混沌系统的预测 [J]. 运城学院学报 ,2023,41(6):47-52.

[6] 齐鹏远，姚锡文，刘清华，等．基于多元线性回归模型的生物质与烟煤混燃灰熔融特征温度预测 [J]. 农业工程学报，2024, 40(15): 174-182.

[7] 吴静依，林瑜，蔺轲，等．基于机器学习的重症监护室超长入住时长预测 [J]. 北京大学学报（医学版），2021, 53(6): 1163-1170.

[8] 付传志，张煜，马杰．基于 AHP-PSO-RF 的三峡枢纽通行船舶待闸时长预测方法 [J]. 武汉理工大学学报，2023, 45(2): 27-34.

[9] 耿晓斌，程云章，钟鸣，等．基于随机森林的血糖变异预测 ICU 监护时长研究 [J]. 软件导刊 ,2021,20(1):51-57.

[10] 刘俊文，谢劭峰，钟雁琴，等．基于 MLP 神经网络的中国南方地区多因子 PWV 预测模型 [J]. 中国科技论文，2024, 19(1): 99-107.

[11] 周峰，吴剑．多目标优化算法在制造业排产中的研究进展 [J]. 系统工程理论与实践，2022, 42(7): 1638-1649.

【作者简介】

孙平（1972—），女，福建福州人，本科，高级工程师，研究方向：工业互联网、知识图谱工业应用、标识解析应用。

（收稿日期：2025-04-03　修回日期：2025-09-08）