ASME Journal of Computing and Information Science in Engineering
Online journal at:
https://asmedigitalcollection.asme.org/computingengineering

Federated Learning on Distributed and Encrypted Data for Smart Manufacturing

Timothy Kuo
Department of Industrial and Manufacturing Engineering,
The Pennsylvania State University,
University Park, PA 16802
e-mail: tjk5789@psu.edu

Hui Yang[1]
Department of Industrial and Manufacturing Engineering,
The Pennsylvania State University,
University Park, PA 16802
e-mail: huiyang@psu.edu

Industry 4.0 drives exponential growth in the amount of operational data collected in factories. These data are commonly distributed and stored in different business units or cooperative companies. Such data-rich environments increase the likelihood of cyber attacks, privacy breaches, and security violations. Also, this poses significant challenges on analytical computing on sensitive data that are distributed among different business units. To fill this gap, this article presents a novel privacy-preserving framework to enable federated learning on siloed and encrypted data for smart manufacturing. Specifically, we leverage fully homomorphic encryption (FHE) to allow for computation on ciphertexts and generate encrypted results that, when decrypted, match the results of mathematical operations performed on the plaintexts. Multilayer encryption and privacy protection reduce the likelihood of data breaches while maintaining the prediction performance of analytical models. Experimental results in real-world case studies show that the proposed framework yields superior performance to reduce the risk of cyber attacks and harness siloed data for smart manufacturing. [DOI: 10.1115/1.4065571]

Keywords: data privacy, fully homomorphic encryption, federated learning, sustainable manufacturing, cyber physical security for factories, cybermanufacturing, data-driven engineering, engineering informatics

1 Introduction

The last decade has witnessed a significant increase in the development and deployment of industry 4.0 technologies. There are more than 6 billion connected devices proactively asking for support in 2018 [1]. Pervasive sensing in the manufacturing industry has resulted in a significant surge in data generation and accumulation. According to a report from International Data Corporation, the manufacturing sector accounted for the highest share of data in 2018, amounting to approximately 3584 Exabytes, and the manufacturing industry is expected to produce as much as about 22.5 Zettabytes of data by the year 2025 [2]. Commonly, these data are distributed and stored in different business units or cooperative companies due to the interconnected collaboration in the global supply chain. By leveraging analytical techniques, organizations can make more informed decisions based on data such as manufacturing sharing economy and resource planning [3]. In other words, data-driven intelligence is indispensable for the realization of smart manufacturing and has become a key enabler to increase manufacturing competitiveness [4].

However, distributed data-rich environments increase the risk of cyber attacks, privacy breaches, and security violations. It is reported that more than 90% of organizations, such as those in the manufacturing, healthcare, or transportation industries, have experienced at least one major cyber attack in the past 2 years [5]. In the manufacturing sector, data breaches often lead to severe implications. For example, confidential designs, robust process parameters, or proprietary manufacturing settings are considered as core assets to gain competitive advantages by a company. If such data are breached, it will undermine a company's innovative strategies and bring significant risks to expose sensitive information to competitors. As a result, competitors can use a smaller investment to replicate advanced manufacturing processes or innovative product designs. Consequently, such breaches represent not just a loss of intellectual property but also a significant setback to a company's competitive position and future growth prospects.

Traditionally, sensing data are collected from different business units and then gathered at a centralized location to develop analytical models. Data ownerships are segregated. Data collected by different business units maintain a distinctive level of data privacy, governed by their respective data management policies. Many stakeholders hesitate and/or decline to communicate raw data for centralized analytics. This reluctance stems from concerns that when large amounts of data are integrated from every independent data owner, the likelihood of privacy breaches also increases. Therefore, a new question emerges, i.e., how to perform analytical computing on siloed datasets from segregated ownerships while maintaining data privacy?

This article presents a novel privacy-preserving framework that enables federated learning on encrypted data stored and distributed in different locations for smart manufacturing. This research includes two key components: (1) *Distributed learning*: It is preferable to maintain distributed storage while developing an analytical model. In the event of a privacy breach at one location, this allows

[1]Corresponding author.

Manuscript received January 19, 2024; final manuscript received May 16, 2024; published online May 31, 2024. Assoc. Editor: Yan Wang.

for the protection of data stored in other locations. Therefore, distributed learning enables businesses to protect the privacy of their data and reduce the risk of data theft. (2) *Computation on encrypted data*: Encryption is a crucial step in enhancing the protection of data privacy and reducing the risk of unauthorized access and theft of sensitive information. Specifically, we leverage fully homomorphic encryption (FHE) to facilitate computations on encrypted data, yielding encrypted results that, upon decryption, align with the results obtained from mathematical operations performed on the unencrypted data.

First, we introduce FHE to protect data privacy and empower computations on encrypted data. Second, we design a federated learning framework on encrypted data so that each factory does not need to communicate raw data with others. Third, the proposed methodology is evaluated and validated with a real-world case study to predict the energy consumption of the machining process. Experimental results show that the proposed privacy-preserving framework increases the protection of data privacy while maintaining the performance of predictive models.

The rest of the article is organized as follows. Section 2 reviews the research background, while the proposed privacy-preserving framework, including federated learning, FHE method, and inference of predictive model on encrypted data, is presented in Sec. 3. Section 4 discusses the real-world case study. Section 5 presents the experimental results. Finally, the concluding remarks are presented in Sec. 6.

2 Research Background

2.1 Data Privacy. Modern manufacturing enterprises are now characterized by complex integration of equipment, processes, and facilities, all of which generate vast amounts of data. This is largely due to the widespread adoption of sensing technologies. Digital twin [6], parallel computing, and network analytics [7] are increasingly employed to derive data-driven intelligence, which promises to revolutionize decision-making processes. In fact, data become a key asset in boosting the competitiveness of manufacturing operations. For example, Yang et al. leveraged a large amount of sensing data to design the new six-sigma quality control of additive manufacturing [8]. Kan and Yang developed a new dynamic network approach for image-guided monitoring of ultraprecision machining and biomanufacturing processes [9]. Nonetheless, most of the existing works focus more on a centralized way to analyze the data for manufacturing intelligence. Although the centralized approach provides convenient access to a shared pool of data, it brings forth significant concerns regarding data privacy, hindering the involved parties to share raw data.

Therefore, data breaches have become a major concern. For example, according to Verizon's 2018 report, the number of security incidents and data breaches exceeded 53,000 and 2,000, respectively. Notably, the manufacturing sector contributed 536 security incidents and 71 data breaches to this total [10]. In addition, as per 2016 California Data Breach Report, Target, a prominent retailer ranked eighth in the United States, experienced a breach of credit card data in 2013 through a third-party vendor. This breach led to the unauthorized disclosure of credit card information of 7.5 million customers [11]. Governments gradually consider data privacy as an important issue, such as California Consumer Privacy Act [12] in the United States and General Data Protection Regulation [13] in the European Union. Moreover, the protection of data privacy at edge devices in the industrial Internet of things has been viewed as a key challenge in recent years [14]. Recently, a variety of innovative privacy-preserving methods have been developed. For example, Krall et al. designed a new mosaic gradient perturbation approach to preserve the privacy of predictive models and demonstrated the effectiveness of privacy protection against model inversion attacks with healthcare datasets and case studies [15,16]. Lee et al. designed a new neuron perturbation approach to preserve the privacy of neural network models that are often employed in data analytics to improve the smartness of manufacturing systems [17]. Hu et al. presented a privacy-preserving analytical model that predicts the energy consumption of machines toward the realization of a smart and sustainable manufacturing system [18].

Currently, smart manufacturing tends to face significant challenges regarding data privacy. Conventional analytical models mainly rely on centralized aggregation and storage of data, which increase the likelihood of data breaches. Therefore, business units often have privacy concerns pertinent to the share of raw data, posing challenges to achieve collective decision-making. Although privacy-preserving methods fueled increasing interest in manufacturing, few, if any, previous investigations have considered the analytical computing on encrypted data. There are different encryption techniques available in the state of the art, e.g., advanced encryption standard (AES). Nonetheless, traditional encryption techniques do not support computing directly on encrypted data, but rather require the encrypted data to be decrypted first before analytical computing. As such, the level of privacy protection becomes weaker after decryption.

2.2 Homomorphic Encryption. Homomorphic encryption (HE) is a cryptographic scheme that allows the evaluation of an arbitrary arithmetic circuit on encrypted data without decryption. This concept was proposed in 1978 [19]. In the early stages, homomorphic cryptography only supports addition or multiplication. In 2009, Gentry put forth the first fully homomorphic encryption approach, which was based on the concept of ideal lattices [20]. However, Cao et al. showed that the scheme presented by Gentry is not efficient and cannot accommodate decimal operations [21].

Therefore, in 2017, Cheon et al. presented an approximate homomorphic encryption method known as CKKS [22]. This method greatly increased the computing effectiveness of floating-point values, attaining homomorphic encryption's optimal efficiency in analytical applications. Moreover, Cheon et al. enhanced the efficiency of the CKKS scheme by utilizing the residual framework as an optimization tool [23]. Their implementation exhibited performance benefits, achieving speed-up improvements of 17.3, 6.4, and 8.3 times for encryption, constant multiplication, and homomorphic multiplication, respectively. In the field of manufacturing, Krall et al. proposed an innovative distributed cryptosystem that integrates Paillier cryptography with the alternating direction method of multipliers for distributed learning and analytics on encrypted data, which is demonstrated with an experimental study for manufacturing resource planning [24].

However, very little has been done to integrate FHE with federated learning for privacy-preserving analytics in current manufacturing practices. Most of the existing federated learning techniques focus more on the distributed learning and model update [25,26], but are less concerned about data privacy and analytical computing on encrypted data. There is an urgent need to investigate FHE-enabled federated learning in the context of smart manufacturing for the protection of data privacy.

3 Research Methodology

As shown in Fig. 1, this article consists of four key components to develop the proposed privacy-preserving framework for federated learning in smart manufacturing. (1) *Segregated data ownership*: We separate the data ownership among independent entities to build a privacy boundary during collaboration in the context of smart manufacturing. (2) *Computation on encrypted data*: The sensitive data are designed to be encrypted by FHE to simultaneously enhance data privacy protection and enable computational operations on the encrypted data. (3) *Federated learning*: The further development of the analytical model is decentralized to eliminate the need of centralized data storage and model learning. (4) *Privacy protection*: A privacy-preserving framework is designed to integrate FHE and federated learning and mitigate the probability of data breaches.

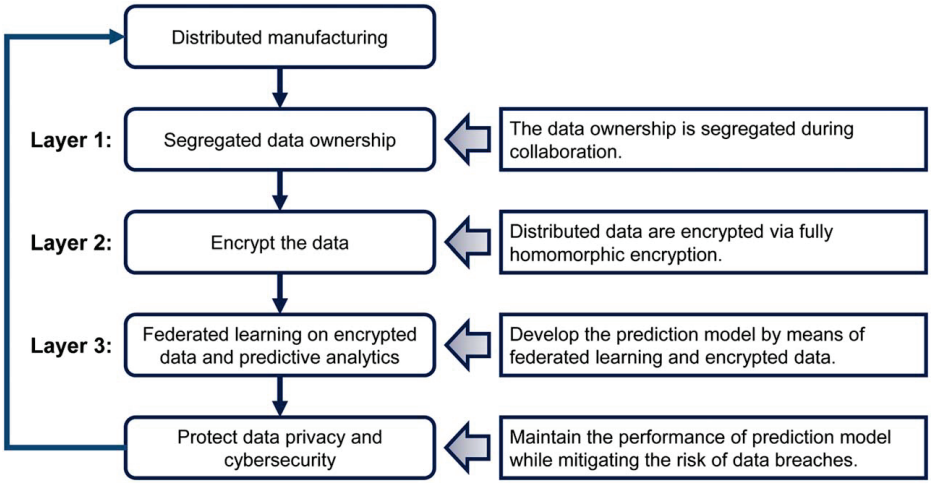

Fig. 1 The proposed privacy-preserving framework

3.1 Manufacturing Data Ownership

3.1.1 Distributed Manufacturing. A finished product is composed of countless parts from various manufacturers that are located in different geographical areas. For example, the chair of an airplane can be produced in Asia, while the engine is manufactured in America. Each unit specializes in making a specific component. A single manufacturer will not be able to produce all components of a finished product effectively and efficiently. Distributed manufacturing involves producing goods across a network of locations rather than in a single facility. A global supply chain can source raw materials and components from various countries, assemble products in different locations, and then ship finished products to customers worldwide [24]. This offers competitive advantages by leveraging specialized expertise, accessing lower costs, and tapping into various advantages offered by different regions. To make a commercially successful product, therefore, manufacturers begin to cooperate together.

However, the risk of data breaches increases when collaborating and sharing sensitive data across multiple units and/or facilities. These manufacturing facilities are often independent, meaning that the data gathered by manufacturers are owned individually. Sensitive data cannot be communicated with others so as to prevent data breaches. This consideration can lead to a loss of trust between facilities, making collaborative learning and analytics a challenge. If sensitive data are hacked, many business units can be affected. This situation will create a barrier to collaboration and hinder effective information exchange, potentially leading to reduced efficiency and missed opportunities.

Hence, if a federated and privacy-preserving framework is realized among various factories, the performance benefits of smart manufacturing can be fully unleashed. Therefore, there is an urgent need to expand the use of data and advanced operational modeling practices to realize the full economic and social benefits of digitalization. Smart manufacturing solutions are vital for manufacturers of all sizes and for plants participating in various supply chains. Democratization will lead to a harmonized manufacturing IT environment, enabling performance improvements in every part of the enterprise. However, despite the collaborative efforts among manufacturers, the ownership of data is partitioned.

3.1.2 Segregated Versus Aggregated Data Ownership. A manufacturing unit or organization can structure, analyze, and interpret its collected data to extract some useful information for decision-making. These data, which are assumed distinct forms based on the situation, can be categorized into two different ownerships: segregated data ownership and aggregated data ownership.

- Segregated data ownership: It implies that data are owned and controlled by individual entities or organizations in isolation. Each business unit maintains ownership of its data and is responsible for managing its usage, sharing, and security.
- Aggregated data ownership: Multiple business units collectively own and manage a dataset that has been combined from various sources. In this context, ownership might not be concentrated in a single business unit but rather distributed among business units that have provided data for the aggregation.

In the context of the worldwide supply chain network, individual manufacturers presently maintain ownership of distinct datasets in isolation, thereby constituting a segregated data ownership. However, due to concerns about data privacy, many manufacturers are reluctant to communicate their sensitive data during collaboration. Thus, the segregated data ownership calls on the design and development of a privacy-preserving framework for data analytics.

Figure 2 depicts the proposed privacy-preserving framework. In each of K units or factories, sensors are installed in factories' machinery to capture data such as energy usage. These data have been designated for the purpose of supporting the collaborative decision-making. To avoid data breaches, we design three layers of protection mechanisms in the proposed framework. First, data ownership is segregated in the layer 1 protection. Raw and unencrypted data belong to the individual owner. In other words, manufacturing operations and machine settings are physically isolated. Next, the layer 2 protection will encrypt the raw data by FHE. Privacy boundaries are established for data across different factories. Finally, layer 3 protection will drive the model development into each factory. This distributed learning approach eliminates the need to store encrypted data in a centralized location.

3.2 Computation on Encrypted Data. Cryptography is aimed at concealing and ensuring the security of data by encrypting data into a concealed form. The history of encryption can be traced back to ancient cultures such as Egypt, Greece, and Rome, where encryption was utilized for religious and military purposes. One of the earliest recorded forms of encryption is Caesar Box, which dates back to approximately 100 BC. The evolution of cryptography has been ongoing, and cryptography remains an indispensable tool for maintaining data privacy in the modern era.

As shown in Fig. 3(*a*), Julius Caesar gave his field commanders the order to launch an attack at noon, and his adversary could potentially have this information if one of his messages was intercepted. To avoid information leakage, each letter of his message is shifted three letters to the left, encrypting the order as

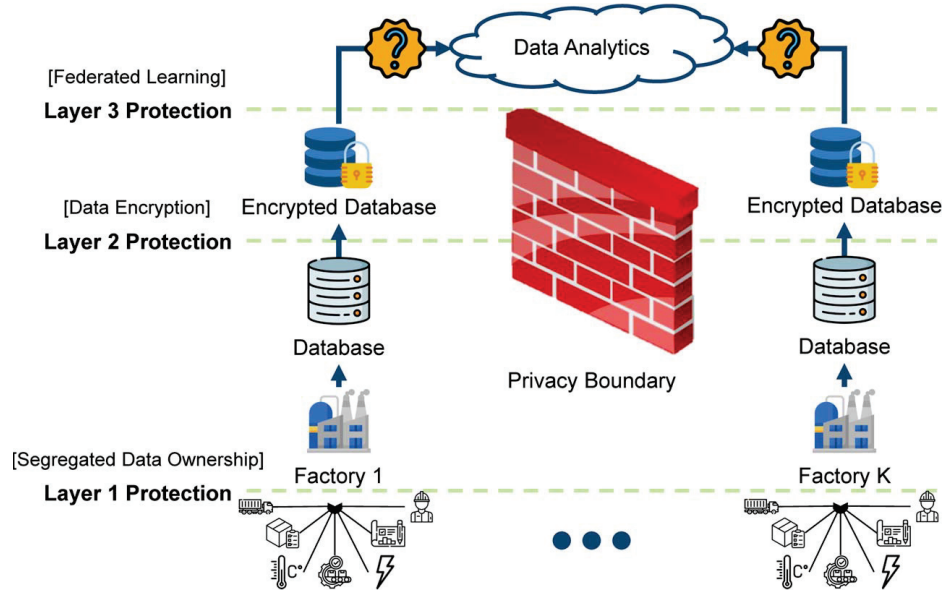

Fig. 2 Multilayer privacy protection of the proposed framework

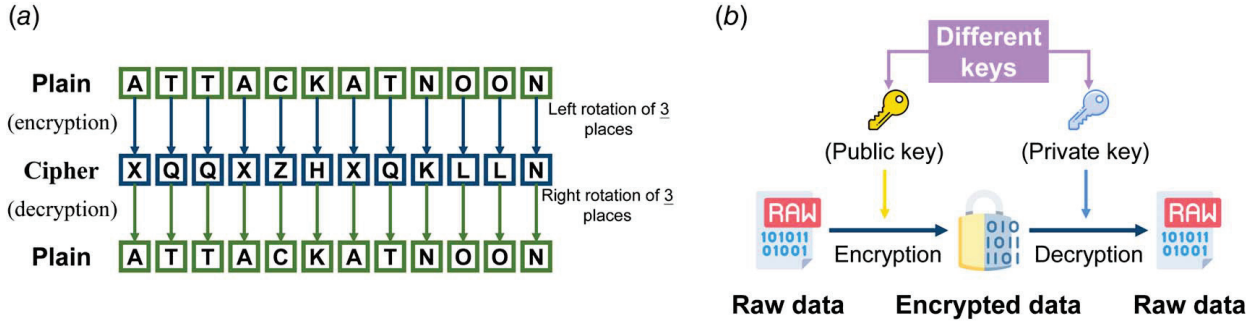

Fig. 3 Comparison between (*a*) symmetric encryption and (*b*) asymmetric encryption

"XQQXZHXQKLLN." This encryption method is currently viewed as symmetric encryption because the same key is used for both encryption and decryption. The plain texts are encrypted by shifting three letters to the left, and the cipher texts are decrypted by shifting three letters to the right. Compared to symmetric encryption, asymmetric encryption is designed to increase data protection. As shown in Fig. 3(*b*), asymmetric encryption has two different keys, namely, the public key and the private key. The public key encrypts the raw data, while the private key decrypts the encrypted data. Notably, the encrypted data cannot be decrypted by the public key.

However, these conventional encryption methods are limited by their scope and focused primarily on protecting sensitive data from external cyber attacks. In other words, encrypted data cannot be used for computation because mathematical operations on encrypted data do not yield the same results as when performed on unencrypted data. Therefore, we propose data encryption by the FHE method, which facilitates computations on encrypted data.

As shown in Fig. 4, each factory owns its data independently so that there is a privacy boundary among factories. Factory 1's dataset is structured as matrices [**X**, **Y**], where **X** represents the set of independent variables and **Y** is the set of measured performance outcomes in real-world manufacturing process. First, factory 1 uses its public key $\tilde{\mathcal{K}}^{(1)}$ to encrypt the raw data x and y into x_e and y_e by the FHE method, respectively. Notably, these keys are generated and saved on their own local proxy servers. Next, the computation function $F(x_e, y_e) = x_e + y_e$ can be performed directly. When utilizing the private key $\mathcal{K}^{(1)}$ to decrypt the result of $x_e + y_e$, it will be equal to $x + y$. On the other hand, factory k also has a dataset $[\mathbf{x}', \mathbf{y}']$ and plans to do multiplication with encrypted. In the beginning, its public key $\tilde{\mathcal{K}}^{(k)}$ is generated to encrypt the data x' and y' into x'_e and y'_e, respectively. Next, factory k can execute the computation function $F(x'_e, y'_e) = x'_e \cdot y'_e$. In the end, this encrypted result can be decrypted by private key $\mathcal{K}^{(k)}$, and the result will be $x' \cdot y'$.

In this article, we first generate a pair of keys, private key $\mathcal{K}^{(k)}$ and public key $\tilde{\mathcal{K}}^{(k)}$ for each factory k, based on ring learning with error [27]. The cyclotomic polynomial ring R_q constitutes a subset of the cyclotomic polynomials, exhibiting an isomorphism with its roots. The basic settings of FHE algorithm are as follows:

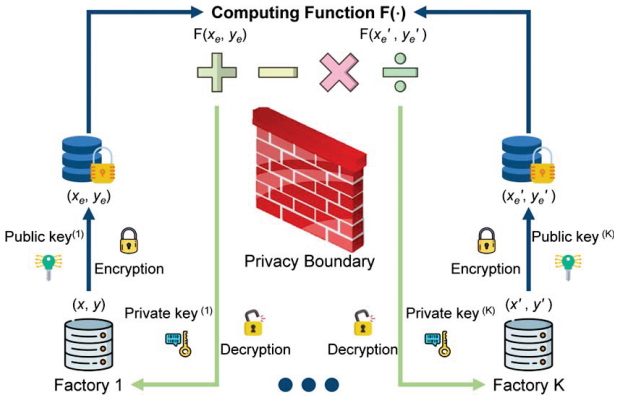

Fig. 4 FHE in smart manufacturing

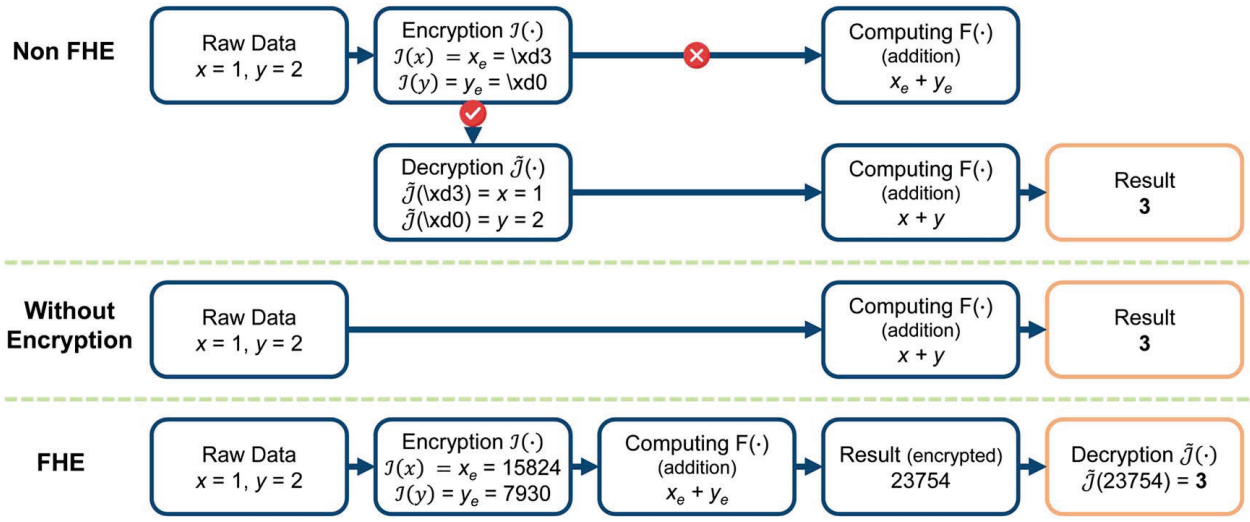

Fig. 5 Comparison among FHE, non-FHE, and without encryption methods

- *Parameters:* $s^{(k)}$ and $a^{(k)}$
- *Private key generation:* $\mathcal{K}^{(k)} = (1, s^{(k)})$, *where* $s^{(k)} \in R_q$.
- *Public key generation:* $\tilde{\mathcal{K}}^{(k)} = (b^{(k)}, a^{(k)})$, where $a^{(k)} \in R_q$, e is error with uniform distribution, and $b^{(k)} = -a^{(k)} \cdot s^{(k)} + e$.
- *Encryption:* This pair of keys allows the factory k to encrypt raw data as follows:

$$x_e = (x, 0) + \tilde{\mathcal{K}}^{(k)} = (x - a^{(k)} \cdot s^{(k)} + e, a^{(k)}) = (c_0, c_1) \quad (1)$$

where x_e represents the encrypted data and x represents the raw data.

- *Decryption:* When the encrypted data are decrypted by factory k, the decryption function is as follows:

$$\tilde{x} = c_0 + c_1 \cdot s^{(k)} = \left(x - a^{(k)} \cdot s^{(k)} + e\right) + a^{(k)} \cdot s^{(k)} = x + e \approx x \quad (2)$$

where $\tilde{x}$ represents the estimated value of x_e. Notably, factory k does not exactly obtain the original data x but x with some noise e. If e is small enough, then the decryption of x_e will be close to the original x. After encryption, the raw data x are hidden in c_0, with a the mask $-a^{(k)} \cdot s^{(k)}$. To remove $a^{(k)} \cdot s^{(k)}$, we can use c_1, which only stores $a^{(k)}$, and combine it with the private key $\mathcal{K}^{(k)}$ to obtain the decryption, which is $x + e$.

The properties of FHE encryption are shown as follows.

- *Additive homomorphism:* Factory k collects x and y, and it plans to do computation with the third party. To avoid data breaches, factory k encrypts the collected data as x_e and y_e as follows:

$$\mathcal{J}(x) = x_e = (c_0^x, c_1^x), \quad \mathcal{J}(y) = y_e = (c_0^y, c_1^y) \quad (3)$$

where $\mathcal{J}(\cdot)$ represents the encryption function. Hence, the addition function is defined as follows:

$$\text{FHE}_{\text{add}} = x_e + y_e = (c_0^x, c_1^x) + (c_0^y, c_1^y) = (c_0^x + c_0^y, c_1^x + c_1^y) \quad (4)$$

When FHE_{add} is decrypted by using $\mathcal{K}^{(k)}$,

$$\begin{aligned} \tilde{\mathcal{J}}(\text{FHE}_{add}) &= c_0^x + c_0^y + (c_1^x + c_1^y) \cdot s^{(k)} \\ &= \left(c_0^x + c_1^x \cdot s^{(k)}\right) + \left(c_0^y + c_1^y \cdot s^{(k)}\right) \\ &= (x + e) + \left(y + e\right) \approx x + y \end{aligned} \quad (5)$$

where $\tilde{\mathcal{J}}(\cdot)$ represents the decryption function.

- *Multiplicative homomorphism:* On the other hand, factory k has raw data x and plans to perform multiplication by any real number t, which is not a sensitive value and does not need to be encrypted. First, the raw data x are encrypted as (c_0^x, c_1^x). The multiplication function can be defined as follows:

$$\text{FHE}_{mult} = x_e \cdot t = (c_0^x \cdot t, c_1^x \cdot t) \quad (6)$$

Hence, when factory k decrypts FHE_{mult} by using $\mathcal{K}^{(k)}$, the equation will be shown as follows.

$$\begin{aligned} \tilde{\mathcal{J}}(\text{FHE}_{\text{mult}}) &= t \cdot c_0^x + t \cdot c_1^x \cdot s^{(k)} \\ &= t \cdot (c_0^x + c_1^x \cdot s^{(k)}) \\ &= t(x + e) = t \cdot x + t \cdot e \approx tx \end{aligned} \quad (7)$$

Moreover, if both data, x and y, in the multiplication function are sensitive, the factory k needs to encrypt both of them as x_e and y_e. Hence, the multiplication function is defined as follows.

$$\begin{aligned} \tilde{\mathcal{J}}(\text{FHE}_{\text{mult}}) &= \tilde{\mathcal{J}}(x_e) \cdot \tilde{\mathcal{J}}(y_e) \\ &= (c_0^x + c_1^x \cdot s^{(k)}) \cdot (c_0^y + c_1^y \cdot s^{(k)}) \\ &= c_0^x \cdot c_0^y + (c_0^x \cdot c_1^y + c_1^x \cdot c_0^y) s^{(k)} + c_1^x \cdot c_1^y \cdot s^{(k)^2} \\ &= d_0 + d_1 \cdot s^{(k)} + d_2 \cdot s^{(k)^2} \end{aligned} \quad (8)$$

where d_0 is $(c_0^x \cdot c_0^y)$, d_1 is $(c_0^x \cdot c_1^y + c_1^x \cdot c_0^y)$, and d_2 is $(c_1^x \cdot c_1^y)$. Notably, $\tilde{\mathcal{J}}(\cdot)$ typically comprises a pair of polynomials; however, in the case of $\tilde{\mathcal{J}}(\text{FHE}_{\text{mult}})$, it is now evident that three polynomials are involved. Therefore, relinearization $\text{ReLin}(\cdot)$ [23], as depicted in Eq. (9), ensures that the dimension of $\tilde{\mathcal{J}}(\text{FHE}_{\text{mult}})$ remains constrained within 2.

$$(d_0', d_1') = \text{ReLin}(\text{FHE}_{\text{mult}}) \quad (9)$$

where

$$d_0' + d_1' \cdot s^{(k)} = d_0 + d_1 \cdot s^{(k)} + d_2 \cdot s^{(k)^2} \quad (10)$$

Finally, during the decryption of the relinearized result, it is observed that

$$\tilde{\mathcal{J}}(\text{ReLin}(\text{FHE}_{\text{mult}})) \approx x \cdot y \quad (11)$$

Figure 5 illustrates that the FHE method exhibits better performance compared to both nonencryption methods and non-FHE

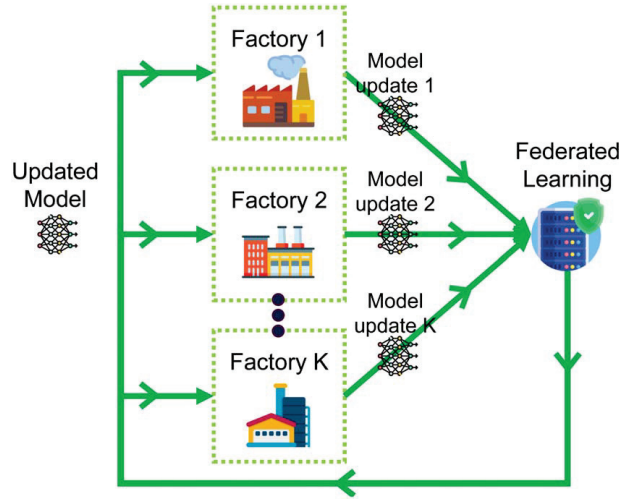

Fig. 6 The architecture of federated learning

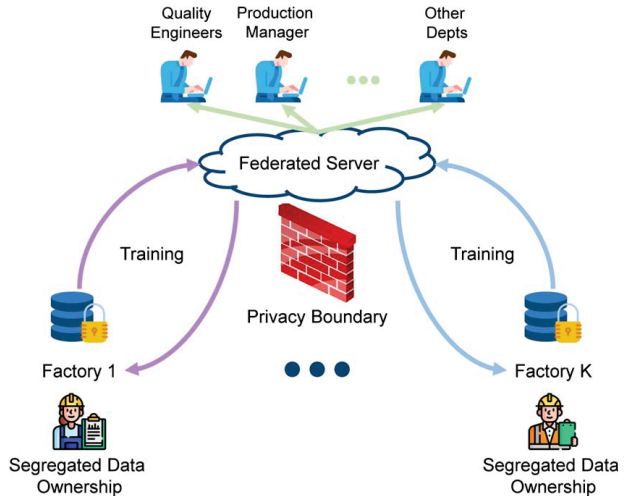

Fig. 7 Federated learning in smart manufacturing

methods, particularly in terms of data privacy and computational capabilities. For example, raw data $x = 1$, $y = 2$ can be directly added, and the factory can get the result as 3. However, the risk is high when communicating raw data. If the factory starts to protect and encrypt sensitive data as x_e and y_e with non-FHE methods, like AES encryption, an issue arises because the computation cannot be realized on AES encrypted data. $\widetilde{\mathcal{J}}(x_e + y_e)$ will not have the same result of 3. Hence, factory needs to decrypt x_e and y_e for computation again. In contrast, if data are encrypted by the FHE method, the computation can be accomplished on encrypted data. When the encrypted result 23754 is decrypted, the value is 3, which is the same as 1+2. But raw data are not exposed. Encryption methods provide greater data privacy protection than without encryption. Further, the FHE method has the capability to allow for computation on encrypted data.

3.3 Federated Learning and Predictive Analytics. Smart manufacturing is established by leveraging large, diverse, and high-quality data. To reduce the likelihood of data breaches, federated learning enables a model to be trained across multiple factories, each holding its respective data, without exchanging them. As shown in Fig. 6, federated learning represents a decentralized methodology for developing analytical models. It eliminates the need to centralize sensitive data from independent data owners on global servers. Instead, independent data owners perform the development separately only with their own data. The insights derived from the context of each independent model's development will be gained collaboratively in the form of a consensus model.

This work proposes a federated learning framework in the context of smart manufacturing. As shown in Fig. 7, K manufacturing factories operate collaboratively, and each individual factory, denoted as factory k, independently owns its data, $\mathcal{D}^{(k)} = \{(\mathbf{x}_i^{(k)}, y_i^{(k)}) | i = 1, \ldots, n^{(k)}\}$. In this context, $\mathbf{x}_i^{(k)}$ represents the ith input vector for factory k, $y_i^{(k)}$ denotes the ith corresponding output, and $n^{(k)}$ signifies the quantity of observations within factory k. Individual factories only store their data and utilize it for the development of analytical models. Due to their manufacturing conditions, capacities, and product orders among distinct factories, different datasets are collected and used for model development. A consensus model is then tasked with integrating these insights drawn from factories.

The FHE method is specifically designed for the execution of arithmetic computations on encrypted data. In the setting of multi-entity cooperation, this study demonstrates the federated learning framework with the use of Bayesian LR as an example. The raw data from each unit are encrypted as follows:

$$\mathcal{J}(\mathbf{x}^{(k)}) = \mathbf{x}_e^{(k)}, \quad \mathcal{J}(y^{(k)}) = y_e^{(k)} \tag{12}$$

$$f(\mathbf{x}_e^{(k)}) = \mathbf{x}_e^{(k)\top}\mathbf{w}^{(t-1)}, \quad y_e^{(k)} = f(\mathbf{x}_e^{(k)}) + \varepsilon \; k = 1, 2, \ldots, K \tag{13}$$

where $\mathbf{x}_e^{(k)}$ represents the encrypted input vector from the kth factory, $\mathbf{w}^{(t-1)}$ is the parameter vector that has been updated for $t-1$ times, $y_e^{(k)}$ is the observed output value from the kth factory, and ε is the Gaussian noise.

Based on the Bayes' theorem, posterior is defined in Eq. (14):

$$p(\mathbf{w}^{(t)}|\mathbf{y}_e^{(k)}, \mathbf{X}_e^{(k)}) = \frac{p(\mathbf{y}_e^{(k)}|\mathbf{X}_e^{(k)}, \mathbf{w}^{(t-1)})p(\mathbf{w}^{(t-1)})}{p(\mathbf{y}_e^{(k)}|\mathbf{X}_e^{(k)})} \tag{14}$$

Therefore, the likelihood of observed data becomes

$$\begin{aligned} p(\mathbf{y}_e^{(k)}|\mathbf{X}_e^{(k)}, \mathbf{w}^{(t-1)}) &= \prod_{i=1}^{I^{(k)}} p(y_{ie}^{(k)}|\mathbf{x}_{ie}^{(k)}, \mathbf{w}^{(t-1)}) \\ &= \prod_{i=1}^{I^{(k)}} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(y_{ie}^{(k)} - \mathbf{x}_{ie}^{(k)\top}\mathbf{w}^{(t-1)}\right)^2}{2\sigma^2}\right) \\ &= \frac{1}{(2\pi\sigma^2)^{\frac{I}{2}}} \exp\left(-\frac{1}{2\sigma^2}\left|\mathbf{y}_e^{(k)} - \mathbf{X}_e^{(k)\top}\mathbf{w}^{(t-1)}\right|^2\right) \\ &= N\left(\mathbf{X}_e^{(k)\top}\mathbf{w}^{(t-1)}, \sigma^2 I\right) \end{aligned} \tag{15}$$

where $I^{(k)}$ represents the number of data factory k has, and $[\mathbf{x}_{ie}^{(k)}, y_{ie}^{(k)}]$ is ith data point of factory k. The marginal likelihood is dependent on the parameters and given by

$$p(\mathbf{y}_e^{(k)}|\mathbf{X}_e^{(k)}) = \int p(\mathbf{y}_e^{(k)}|\mathbf{X}_e^{(k)}, \mathbf{w}^{(t-1)})p(\mathbf{w}^{(t-1)})\, d\mathbf{w} \tag{16}$$

The process of federated learning allows each individual factory to separately and continually develop the predictive model without the need for data centralization. The model is pushed to each factory, as opposed to sharing data with others while building the predictive model. This learning process contributes to democratized manufacturing by enabling factories cooperate with others.

The posterior of the model update for factory k can also be written as follows:

$$p\left(\mathbf{w}^{(t)}|\mathbf{X}_e^{(k)}, \mathbf{y}_e^{(k)}\right) \propto \exp\left(-\frac{1}{2\sigma^2}\left(\mathbf{y}_e^{(k)} - \mathbf{X}_e^{(k)\top}\mathbf{w}^{(t-1)}\right)^\top\left(y_e^{(k)} - \mathbf{X}_e^{(k)\top}\mathbf{w}^{(t-1)}\right)\right)\exp\left(-\frac{1}{2}\mathbf{w}^{(t-1)\top}\Sigma_p^{-1}\mathbf{w}^{(t-1)}\right)$$
$$\propto \exp\left(-\frac{1}{2}\left(\mathbf{w}^{(t-1)} - \bar{\mathbf{w}}\right)^\top\left(\frac{1}{\sigma^2}\mathbf{X}_e^{(k)}\mathbf{X}_e^{(k)\top} + \Sigma_p^{-1}\right)\left(\mathbf{w}^{(t-1)} - \bar{\mathbf{w}}\right)\right) \tag{17}$$

where $\bar{\mathbf{w}} = \sigma^{-2}\left(\sigma^{-2}\mathbf{X}_e^{(k)}\mathbf{X}_e^{(k)\top} + \sum_p^{-1}\right)^{-1}\mathbf{X}_e\mathbf{y}_e$. The posterior distribution follows the Gaussian distribution, with an average $\bar{\mathbf{w}}$ and covariance matrix A^{-1}, shown as follows:

$$p\left(\mathbf{w}^{(k)}|\mathbf{X}_e^{(k)}, \mathbf{y}_e^{(k)}\right) \sim N\left(\bar{\mathbf{w}} = \frac{1}{\sigma^2}A^{-1}\mathbf{X}_e^{(k)}\mathbf{y}_e^{(k)}, A^{-1}\right) \tag{18}$$

where $A = \sigma^{-2}\mathbf{X}_e^{(k)}\mathbf{X}_e^{(k)\top} + \Sigma_p^{-1}$. The updated parameters, $\mathbf{w}^{(t)}$, will be computed for the kth factory with the encrypted data. In other words, $\mathbf{w}^{(t-1)}$, will be updated as $\mathbf{w}^{(t)}$ after learning with data from factory k. Factory k will announce the updated model to every cooperated factory. Hence, factory $k+1$ can utilize the new parameters, $\mathbf{w}^{(t)}$, from the previous factory's updating as the prior to update the parameters to $\mathbf{w}^{(t+1)}$. This federated learning facilitates collaborative computing among multiple units or factories and does not require to communicate the sensitive data with others.

The algorithm of the proposed privacy-preserving framework is described as follows:

Algorithm 1 Federated Learning in the Proposed Privacy-Preserving Framework

Input: $[X^{(k)}, Y^{(k)}]$
  Define the prior $w^{(0)}$
  Create a pair of public key $\tilde{K}^{(k)}$ and private key $K^{(k)}$ for every factory
  The input data undergo encryption before storage
  for iteration $t \in [1, 2, 3, \ldots, T]$ **do**
    Randomly assign factory k to update the model
    Calculate the posterior from encrypted data
    Compute the new parameters of the predictive model
    Factory k announces the updated model to other factories
    t = t+1
  end for

Each factory k has its pair of public key $\tilde{\mathcal{K}}^{(k)}$ and private key $\mathcal{K}^{(k)}$. Factory k encrypts the raw data by means of its public key. When the raw data are collected by sensors, these data will be encrypted. Second, the factory will store these encrypted data. Once the model is pushed to the factory, this factory will utilize the encrypted data as the training data to update the model. Finally, the updated model will be then announced to every factory.

4 Real-World Case Study

In recent decades, society and the natural ecosystem have experienced significant impacts due to climate change. Anticipated effects include heightened intensity and frequency of extreme weather events, prolonged climate shifts across extensive areas, and the melting of ice caps resulting in rising sea levels. The aim to mitigate these impacts involves limiting global carbon dioxide atmospheric concentrations to 450 ppm by the year 2050 [28]. Mitigating carbon dioxide is especially challenging, mainly due to its strong connection with energy generation and consumption processes. Both carbon emission and climate change have driven a growing emphasis on devising strategies to optimize energy consumption, monitoring, and scheduling. Therefore, a predictive model of energy consumption is urgently needed. In this case study, five distinct facilities, designated as factories 1–5, possess independent ownership and control over their respective datasets. If they collaboratively work toward the development of a predictive model for energy management, the concern of data privacy emerges.

Currently, energy consumption data are collected from multiple computer numerical control (CNC) turning machines by sensors in machine shops and sent through Industrial Ethernet to the data storage server within factories. Our previous studies have focused on the aggregation of all datasets into a centralized location for data-driven energy modeling and real-time analysis of energy efficiency [29]. In this investigation, we assume segregated data ownership in each factory. In other words, energy consumption data are collected in real time when parts are processed by CNC machines in each factory. It is common that there are often different numbers of parts and different types of materials processed by a variety of machines in each factory. Therefore, factories 1–5 own unique data assets that are critical to collective decision-making and energy prediction. In fact, factories 1–5 have different number of data points in this case study, which are 500, 200, 800, 450, and 100. Energy consumption data are assumed to be collected sequentially as a part finished processing in each factory. The predictive model will be iteratively updated in a federated way when new data become available from a factory. The energy consumption of machining process is considered as an important aspect within the realm of energy management. Seven features pertinent to energy consumption of machining process are logged in this case study. These features can be categorized into three groups as follows:

- *Product specification*: Product specifications play a crucial role in determining energy consumption, which consist of three important variables as follows: diameter (x_1), material (x_2), and tensile strength (x_3).
- *Machining parameter*: Machining parameters, namely, feed rate (x_4) and cutting depth (x_5), lead to varying levels of energy requirements during the machining process.
- *Workstation energy*: Workstations need varying levels of energy during the machining process, including idle energy (x_6) and air cutting energy (x_7).

This work investigates the proposed framework in terms of data privacy protection and prediction performance. First, four predictive models are compared in this case study, namely, LR, predictor value transformation model (TM-P), target value transformation model (TM-T), and double-sided transformation model (DTM). The optimal predictive model will be implemented in the proposed privacy-preserving framework. Next, this study evaluates and validates the performance of the proposed framework by taking into account privacy protection, predictive results of encrypted data, and the necessity of communicating data.

- Linear regression model

$$y_i^{(k)} = \beta_0 + \sum_{j=1}^{J}\beta_j \cdot x_{ij}^{(k)} + \varepsilon_i \tag{19}$$

- Predictor value transformation model

$$\begin{cases} e^{y_i^{(k)}} = \beta_0 \prod_{j=1}^{J} x_{ij}^{(k)\beta_j} + \varepsilon_i & , \text{if } \lambda = 0 \\ y_i^{(k)} = \beta_0 + \sum_{j=1}^{J}\beta_j \cdot \frac{x_{ij}^{(k)\lambda} - 1}{\lambda} + \varepsilon_i & , \text{if } \lambda \neq 0 \end{cases} \tag{20}$$

- Target value transformation model

$$\begin{cases} y_i^{(k)} = \beta_0 \prod_{j=1}^{J} e^{\beta_j x_{ij}^{(k)}} + \varepsilon_i & , \text{if } \lambda = 0 \\ \frac{y_i^{(k)\lambda} - 1}{\lambda} = \beta_0 + \sum_{j=1}^{J} \beta_j \cdot x_{ij}^{(k)} + \varepsilon_i & , \text{if } \lambda \neq 0 \end{cases} \tag{21}$$

- Double-sided transformation model

$$\begin{cases} y_i^{(k)} = \beta_0 \prod_{j=1}^{J} x_{ij}^{(k)\beta_j} + \varepsilon_i & , \text{if } \lambda = 0 \\ \frac{y_i^{(k)\lambda} - 1}{\lambda} = \sum_{j=1}^{J} \beta_j \cdot \frac{x_{ij}^{(k)\lambda} - 1}{\lambda} + \varepsilon_i & , \text{if } \lambda \neq 0 \end{cases} \tag{22}$$

where $x_{ij}^{(k)}$ is the ith input variable of the jth attribute value from the kth factory, β_j is the coefficient of the jth input variable, β_0 is the intercept of model, ε_i is the ith noise term, $y_i^{(k)}$ is the ith output variable from the kth factory, and λ is the parameter of Box-Cox transformation.

5 Experimental Results

5.1 Descriptive Analysis. The fluctuations in energy consumption can be attributed to manufacturing conditions (e.g., materials, processes, and product specifications). The correlation coefficient helps identify features that exhibit a statistical relationship with energy consumption. Here, we derive the visualization results of correlation coefficients to screen and identify meaningful features for energy consumption prediction. As shown in Fig. 8, diameter, feed rate, cutting depth, idle energy, and air cutting energy exhibit high correlation coefficients with energy consumption, having values of 0.63, 0.47, 0.45, 0.77, and 0.40, respectively. All of them are higher than 0.4. In other words, visualization results reveal that these input variables are sensitive to the variations of energy consumption.

However, it is important to note that datasets of factories 1–5 vary due to differences in manufacturing situations, including product orders and capabilities. For instance, as shown in Fig. 9, the distribution of data differs between idle energy and energy consumption across five factories. Therefore, data from factories 1–5 have different correlation coefficients in relation to energy consumption. As shown in Fig. 10, feed rate, cutting depth, and cutting energy exhibit varying correlation coefficients across different factories. Note that cooperation involves the integration of data among five factories. Comparing factory 5 and cooperation, it is observed that correlation coefficients of feed rate and cutting energy from factory 5 are lower than cooperation. These results of factory 5 suggest that neither feed rate nor cutting energy has a significant influence. Therefore, the predictive model constructed independently by factory 5 will not make these two features as substantial as they should be and statistically defined as insignificant features. On the contrary, factory 5 exhibits a higher correlation coefficient of 0.65 between cutting rate and energy consumption when compared to the cooperative approach. Thus, if factory 5 constructs the predictive model, the cutting rate will lead to a 33% increase in its importance for energy predictions. Overall, through collaborative efforts, communicating data enables understanding of actual situation and enhances the collective capability to construct a predictive model.

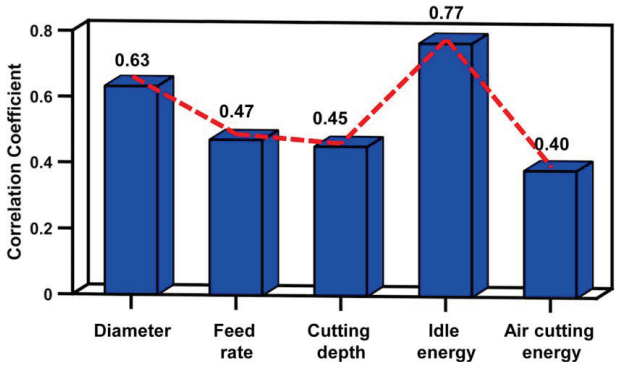

Fig. 8 Correlation coefficients between process features (i.e., diameter, feed rate, cutting depth, idle energy, and air cutting energy) and energy consumption

5.2 Performance Comparison of Models (Without Encryption). The Box-Cox transformation, shown as Eq. (23), leverages λ to transfer nonnormal data into a normal distribution. The choice of λ impacts prediction performance. As shown in Table 1, we conduct the performance comparison among three predictive models with different levels of λ, including −0.5, 0, and 0.5. First, R^2 values of TM-P vary from 77.4%, 81.3%, and 82.3% when λ is varied among three levels. TM-P yields the highest F-statistic value of 1,237 when λ is 0.5. If we look at Akaike information criterion (AIC), TM-P with $\lambda = 0.5$ has the lowest AIC of 3011, second by $\lambda = 0$ with an AIC of 3022. TM-P with $\lambda = -0.5$ has the highest AIC of 3,060. Note that AIC is a technique based on in-sample fit that estimates the likelihood of a model to predict or estimate future values. Lower AIC value indicates a better fit. Similarly, Bayesian information criterion (BIC) has the lowest value when λ is 0.5. BIC is another criterion for model selection that measures the trade-off between model fit and complexity and lower BIC values indicate a better fit. Therefore, the optimal λ for TM-P is 0.5. Second, TM-T attains the highest values for R^2, adj. R^2, and F-statistic when λ is set to 0.5. However, TM-T has the lowest AIC and BIC at $\lambda = 0$. It may be noted that the difference in R^2, adj. R^2, and F-statistic between $\lambda = 0$ and 0.5 is much smaller than that of AIC and BIC. In other words, the optimal value for λ that maximizes TM-T's prediction performance is determined to be 0.5. Finally, experiential results of DTM show that adj. R^2 values are varying from 83.2%, 86%, and 84.9% when the λ is varied among these three levels. The AIC and BIC value for $\lambda = 0$ are −1,639 and −1,635, respectively, which are notably lower than those for $\lambda =$ −0.5 and 0.5. Hence, this comparative analysis underscores that employing the Box-Cox transformation with $\lambda = 0$ has a better prediction performance for DTM.

$$f(y) = \begin{cases} \frac{y_i^{\lambda} - 1}{\lambda}, & \text{if } \lambda \neq 0 \\ \ln(y_i), & \text{if } \lambda = 0 \end{cases} \tag{23}$$

Then, this study compares four predictive models using five different evaluation metrics, namely, R^2, adj. R^2, F-statistic, AIC, and BIC. As shown in Fig. 11(*a*), LR and TM-P return very close R^2 and adj. R^2 values, which are around 82% for both. TM-T exhibits the lowest R-squared and adj. R^2 values, which are 69.4% and 69.3%, respectively. If we look at DTM, it is not hard to find that DTM outperforms the other three predictive models with values of 86.1% and 86%. Figure 11(*b*) illustrates F-statistic values of four predictive models. The superiority of DTM is underscored by the significantly highest F-statistic value of 1757, compared to 1208 for LR, 1320 for TM-P, and 646 for TM-T. As shown in Figs. 11(*c*) and 11(*d*), DTM still results in a better prediction performance. In both cases, DTM outperforms LR, TM-P, and TM-T with the lowest AIC and BIC values, which are −1632 and −1587, respectively.

We further perform normal Q-Q plots and distribution of residual as descriptive graphical tools for model diagnosis of DTM. Note that Fig. 12(*a*) shows the normal Q-Q plot approximately follows a straight line and Fig. 12(*b*) shows the distribution of the residual is normal. Hence, DTM with $\lambda = 0$ stands as the optimal predictive model among four predictive models for forecasting energy consumption in this case study.

5.3 Fully Homomorphic Encryption-Enabled Federated Learning and Modeling. This study first evaluates and validates

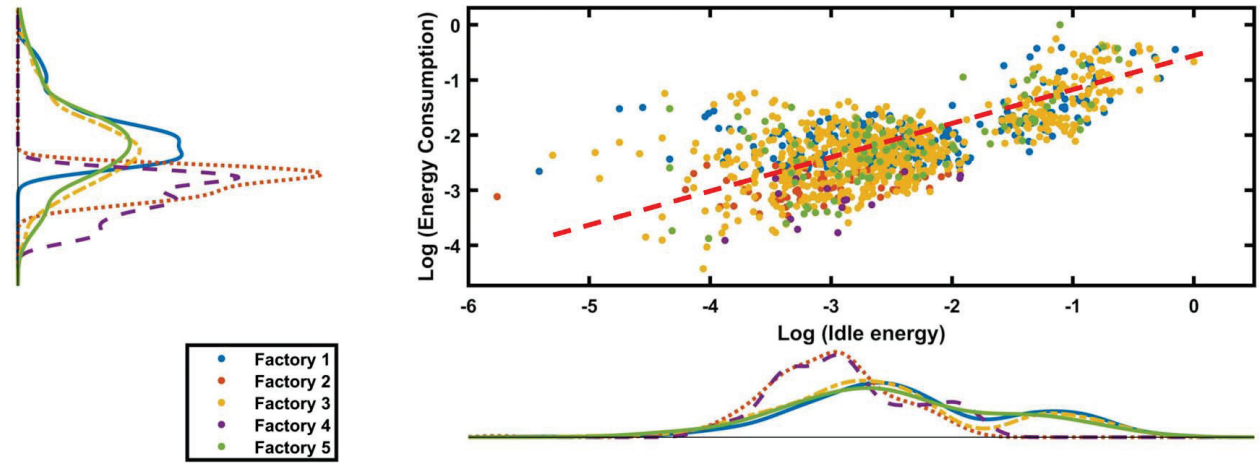

Fig. 9 The variations of scatter plots for idle energy and energy consumption among five factories

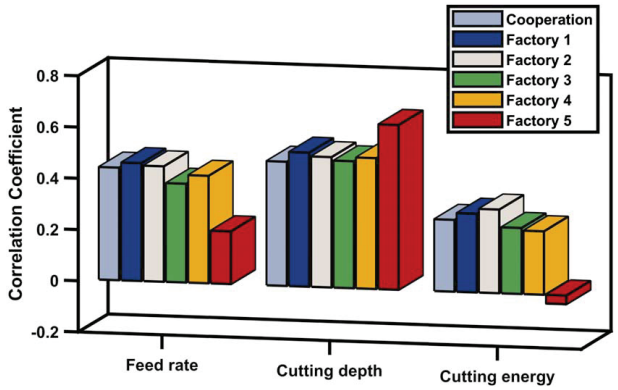

Fig. 10 Comparison of correlation coefficients across five factories and cooperation

data encryption and computational operations of FHE method with regard to their influence on prediction performance. Note that this study compares the performances between the model with FHE and the model without encryption. Experimental results show that both models exhibit similar performance in terms of R^2, adj. R^2, AIC, and BIC. We further measure the absolute differences among these four metrics, which are 1×10^{-4}, 3×10^{-4}, 1.9×10^{-3}, and 1.6×10^{-3}, respectively. The performance variation between encrypted and non-encrypted models is trivial. Therefore, the proposed privacy-preserving framework does not lead to a reduction in the prediction performance while adding the encryption mechanism to protect data.

It is well known that FHE encryption is computationally heavy, albeit providing a high level of data privacy protection. Thus, we have further compared computational times between the model with FHE and the model without encryption.

- *Model learning:* Due to the computational complexity, we have performed the experiments on 70 processors of a high-performance computing cluster, each equipped with 48 cores. The computational speeds for model learning with factory 1 are different, approximately 5894.13 ms for the model with FHE versus <1 ms if without encryption. This comparison highlights the inherent complexity and resource intensity that are needed for the purpose of data protection.
- *Privacy protection:* Nonetheless, the importance of data privacy cannot be overstated, particularly in the manufacturing industry, where the confidentiality of data is not just a preference but a necessity. When the risk of data exposure carries significant operational and reputational risks, the privacy-preserving framework, despite its higher computational requirements, is imperative. This calls upon a strategic investment in data protection, acknowledging that the value of protecting sensitive data far outweighs the increased resource allocation for computational processes.

Consequently, this study shows the resource implications of different data protection strategies and affirms the importance of privacy preservation in the manufacturing industry. In addition, there are different approaches (i.e., either hardware or algorithmic designs) to further improve the computational efficiency of FHE within federated learning frameworks. For example, new algorithms can be designed to streamline the processing of encrypted data. Large-scale parallelization (e.g., MapReduce architectures and hardware acceleration) can also be leveraged to reduce the computational time.

We further conduct a performance comparison among FHE with the federated learning model and independent models for factories 1–5. Notably, factories 1–5 relied solely on their respective data

Table 1 Performance comparison of different λ for TM-P, TM-T, and DTM

	TM-P			TM-T			DTM		
λ Value	−0.5	0	0.5	−0.5	0	0.5	−0.5	0	0.5
R^2	77.4%	81.3%	82.3%	69.4%	77.2%	81.5%	83.3%	86.1%	85%
Adj. R^2	77.3%	81.2%	82.2%	69.3%	77.1%	81.4%	83.2%	86%	84.9%
F-statistic	971.9	1237	1320	646	960	1251	1,418	1757	1611
AIC	3060	3022	3011	−1519	−643.3	1439	−1632	−1639	1397
BIC	3064	3026	3015	−1514	−598	1443	−1587	−1635	1401

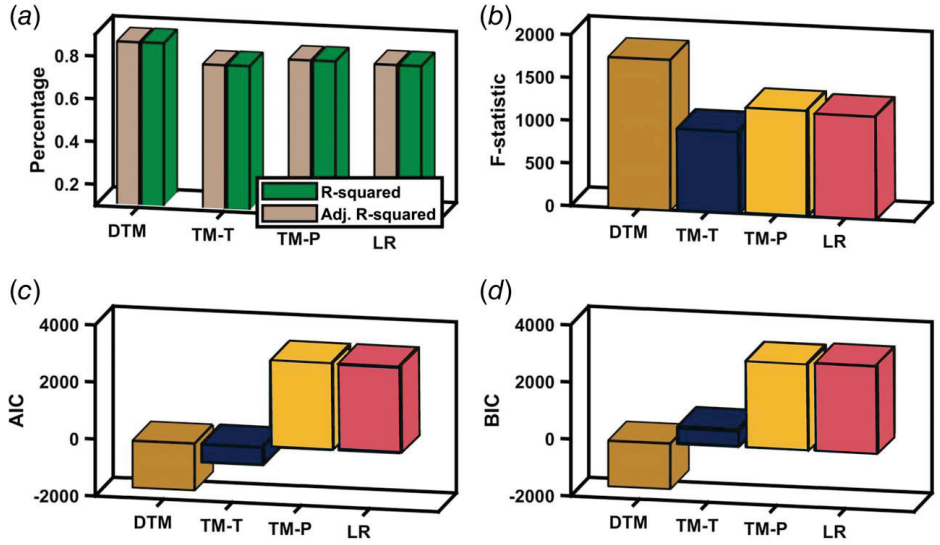

Fig. 11 Performance comparison among four predictive models under different metrics of (*a*) R^2 and adj. R^2, (*b*) F-statistic, (*c*) AIC, and (*d*) BIC

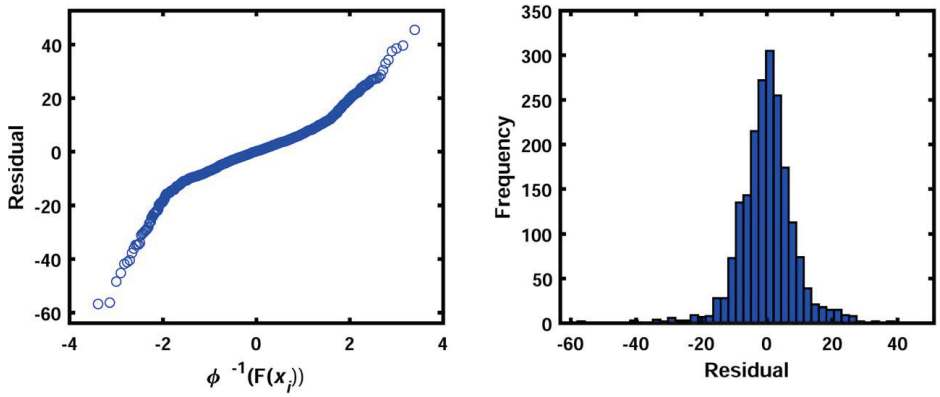

Fig. 12 The residual diagnosis of DTM model

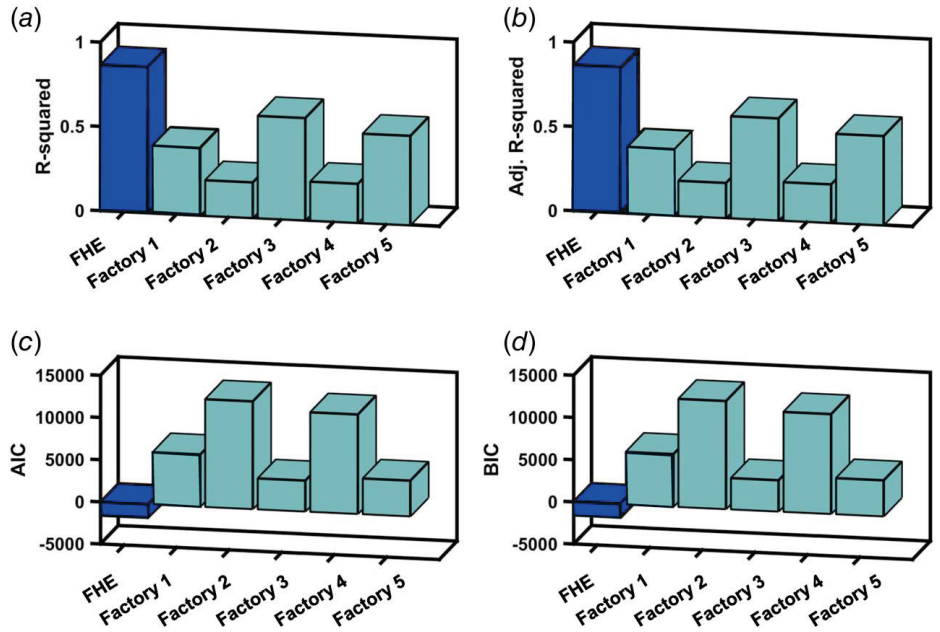

Fig. 13 Performance comparison among FHE with federated learning model and independent models from five factories under different metrics of (*a*) R^2, (*b*) adj. R^2, (c) AIC, and (*d*) BIC

for the independent development of predictive models. Statistical comparison of predictive results measured in terms of R^2 and adj. R^2 values are summarized in Figs. 13(*a*) and 13(*b*). The model with FHE and federated learning achieves the highest R^2 value of 0.86 compared to factories 1–5. By adopting this model, R^2 value is improved by at least 43.3 %. Similarly, the FHE with the federated learning model is found to be better than the others in terms of having maximum adj. R^2 values. Furthermore, in Figs. 13(*c*) and 13(*d*), FHE with the federated learning model also yields a better performance in terms of AIC and BIC. AIC is −1,633, significantly lower than that of factories 1–5. Similarly, BIC is −1,594.53, also lower than that of factories 1–5, which are 6,205, 12,769, 3,674, 11,792, and 4,241, respectively. Therefore, these results indicate that FHE with the federated learning model achieves superior prediction performance.

5.4 Privacy Analysis. In this article, our primary focus encompasses two distinct aspects of cybersecurity: the preservation of privacy against external adversaries and the resistance to attacks.

(1) Privacy preservation from external adversaries:
The current paradigm of smart manufacturing poses a vulnerability wherein sensitive data is at risk of exposure. This vulnerability arises from the centralization of data storage within the computing platform, consolidating all data for utilization in the analytical process. In contrast, through the encryption of sensitive data and the adoption of a decentralized approach to both learning processes and data storage, the proposed framework mitigates the risk of sensitive data exposure.

(2) Resistance to attacks:
In order to mitigate cyber attacks, including offline attack and encryption attack, the proposed framework is designed to reduce the risk of the likelihood of data breaches. First, dictionary and brute force attacks are common in the context of offline attacks. A dictionary attack involves systematically entering every word in a dictionary as a private key, while a brute force attack uses trial and error to guess private key. The proposed framework is designed to enhance the resistance of these attacks by means of randomly generating the pair of public and private keys and increasing the key length such that the probability of key guessing is close to 0. Second, for the encryption attack, the pairs of public and private keys in the proposed privacy-preserving framework are not the same for different factories. The use of nonfixed keys makes attackers difficult to get the sensitive data under the known plaintext attack model and chosen ciphertext attack model.

As shown in Fig. 2, the proposed privacy-preserving framework overcomes the catastrophic consequences of data breaches even if a hacker successfully attacks the database. First, when there is a data breach in layer 1 protection for one factory, only the raw data of that specific factory are exposed. Data belonging to other factories remain protected due to segregated data ownership. Second, a hacker can only access encrypted data in layer 2 protection. In other words, without the correct pair of keys, hackers cannot decipher the real data from its encrypted form. Finally, if data are exposed in layer 3 protection, an attacker can only obtain encrypted data and parameters of analytical models from one factory because the model development is distributed. Therefore, the proposed framework mitigates the risk of data breaches when multiple independent data owners collaborate in developing an analytical model.

6 Conclusions

Democratized manufacturing leads to a rise in collaboration among business units. Communicating great amounts of data within the collaboration to build analytical models enhances the performance of manufacturing. However, these data can be processed and then transformed into sensitive information, which can pertain to every aspect of the manufacturing system. This situation makes business units face some new risks, such as data breaches, which can disrupt the trust among multiple cooperative units, factories, or organizations. Therefore, the development of a privacy-preserving framework is urgently needed to protect the sustainability and resilience of smart manufacturing.

In this article, we present a novel privacy-preserving framework that enables federated learning on encrypted data stored and distributed in different locations for smart manufacturing. Due to the mechanism of federated learning and FHE method, the proposed privacy-preserving framework does not require storing sensitive data from different business units in a centralized location. Moreover, the learning process is decentralized and based on encrypted data. The proposed privacy-preserving framework is evaluated and validated with real-world data. We compare the proposed privacy-preserving model with the traditional analytical model in terms of the level of privacy protection and prediction performance. Experimental results show that the proposed framework can significantly reduce the likelihood of data breaches, consequently fostering collaboration among different business units. The collaboration will enhance the model performance, enabling manufacturers to refine their strategic plans to save time and costs. In addition, the proposed framework ensures that the model performance remains comparable when employing encryption methods for analytical model. Overall, the proposed framework shows strong potential to promote smart manufacturing while preserving data privacy.

In addition, it is important to note that malleability may pose further challenges to the practical implementation of FHE. This is still an open question in the cryptography community. Future research can be performed to investigate various antimalleability techniques in real-world manufacturing case studies, evaluating their impact on data privacy, computational efficiency, and user experience. This will help balance the trade-offs between data privacy and performance, moving closer to the deployment of FHE in practical applications. Furthermore, differential privacy represents an alternative strategy that ensures that one's participation in a dataset, or lack thereof, will not be disclosed. Future research can investigate the integration of differential privacy with the proposed privacy-preserving framework, thereby improving the multilayer integration for the protection of data privacy.

Acknowledgment

This article is based upon research studies supported by the U.S. National Science Foundation (Award No. IIS-2302834). Any opinions, findings and conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the U.S. National Science Foundation.

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The datasets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request.

References

[1] Oztemel, E., and Gursev, S., 2020, "Literature Review of Industry 4.0 and Related Technologies," J. Intel. Manufact., **31**, pp. 127–182.
[2] Rydning, D., Reinsel, J., and Gantz, J., 2018, "The Digitization of the World From Edge to Core," https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf, Accessed January 15, 2024.

[3] Yang, H., Chen, R., and Kumara, S., 2021, "Stable Matching of Customers and Manufacturers for Sharing Economy of Additive Manufacturing," J. Manuf. Syst., **61**, pp. 288–299.
[4] Kusiak, A., 2017, "Smart Manufacturing Must Embrace Big Data," Nature, **544**(7648), pp. 23–25.
[5] Xiong, H., Mei, Q., and Zhao, Y., 2020, "Efficient and Provably Secure Certificateless Parallel Key-Insulated Signature Without Pairing for IIoT Environments," IEEE Syst. J., **14**(1), pp. 310–320.
[6] Lee, H., and Yang, H., 2023, "Digital Twinning and Optimization of Manufacturing Process Flows," J. Manuf. Sci. Eng., **145**(11), p. 111008.
[7] Kan, C., Yang, H., and Kumara, S., 2018, "Parallel Computing and Network Analytics for Fast Industrial Internet-of-Things (IIoT) Machine Information Processing and Condition Monitoring," J. Manuf. Syst., **46**, pp. 282–293.
[8] Yang, H., Rao, P., Simpson, T., Lu, Y., Witherell, P., Nassar, A. R., Reutzel, E., and Kumara, S., 2021, "Six-Sigma Quality Management of Additive Manufacturing," Proc. IEEE, **109**(4), pp. 347–376.
[9] Kan, C., and Yang, H., 2017, "Dynamic Network Monitoring and Control of in Situ Image Profiles From Ultraprecision Machining and Biomanufacturing Processes," Q. Reliabil. Eng. Inter., **33**(8), pp. 2003–2022.
[10] Widup, S., Spitler, M., Hylender, D., and Bassett, G., 2018, "Verizon Data Breach Investigations Report," https://enterprise.verizon.com/resources/reports/DBIR_2018_Report_execsummary.pdf, Accessed January 15, 2024.
[11] Harris, K. D., and General, A., 2016, "California Data Breach Report 2012-2015," https://oag.ca.gov/sites/all/files/agweb/pdfs/dbr/2016-data-breach-report.pdf, Accessed January 15, 2024.
[12] De la Torre, L., 2018, "A Guide to the California Consumer Privacy Act of 2018," https://ssrn.com/abstract=3275571, Accessed January 15, 2024.
[13] European Parliament and Council of European Union, 2018, "General Data Protection Regulation," https://gdpr-info.eu
[14] Yang, H., Kumara, S., Bukkapatnam, S. T., and Tsung, F., 2019, "The Internet of Things for Smart Manufacturing: A Review," IISE Trans., **51**(11), pp. 1190–1216.
[15] Krall, A., Finke, D., and Yang, H., 2021, "Mosaic Privacy-Preserving Mechanisms for Healthcare Analytics," IEEE J. Biomed. Health Inform., **25**(6), pp. 2184–2192.
[16] Krall, A., Finke, D., and Yang, H., 2020, "Gradient Mechanism to Preserve Differential Privacy and Deter Against Model Inversion Attacks in Healthcare Analytics," 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Montreal, QC, Canada, July 20–24, pp. 5714–5717.
[17] Lee, H., Finke, D., and Yang, H., 2024, "Privacy-Preserving Neural Networks for Smart Manufacturing," J. Comput. Inf. Sci. Eng., **24**(7), p. 071002.
[18] Hu, Q., Chen, R., Yang, H., and Kumara, S., 2020, "Privacy-Preserving Data Mining for Smart Manufacturing," Smart Sustainable Manufact. Syst., **4**(2), pp. 99–120.
[19] Rivest, R. L., Adleman, L., and Dertouzos, M. L., 1978, "On Data Banks and Privacy Homomorphisms," Foundat. Secure Comput., **4**(11), pp. 169–180.
[20] Gentry, C., 2009, "Fully Homomorphic Encryption Using Ideal Lattices," Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing, Bethesda, MD, May 31–June 2, pp. 169–178.
[21] Cao, Z., Liu, L., and Li, Y., 2018, "Ruminations on Fully Homomorphic Encryption in Client-Server Computing Scenario," Inter. J. Electron. Inform. Eng., **8**(1), pp. 32–39.
[22] Cheon, J. H., Kim, A., Kim, M., and Song, Y., 2017, "Homomorphic Encryption for Arithmetic of Approximate Numbers," *Advances in Cryptology – ASIACRYPT 2017*, T. Takagi and T. Peyrin, eds., Springer, Cham, Switzerland, pp. 409–437.
[23] Cheon, J. H., Han, K., Kim, A., Kim, M., and Song, Y., 2019, "A Full RNS Variant of Approximate Homomorphic Encryption," *Selected Areas in Cryptography—SAC 2018*, C. Cid and M. J. Jacobsen Jr., eds., Springer International Publishing, Cham, Switzerland, pp. 347–368.
[24] Krall, A., Finke, D., and Yang, H., 2024, "Distributed Cryptosystem for Service-Oriented Smart Manufacturing," IISE Transac., pp. 1—14.
[25] Chung, S., and Al Kontar, R., 2024, "Federated Condition Monitoring Signal Prediction With Improved Generalization," IEEE Trans. Reliab., **73**(1), pp. 438–450.
[26] Sun, W., Lei, S., Wang, L., Liu, Z., and Zhang, Y., 2021, "Adaptive Federated Learning and Digital Twin for Industrial Internet of Things," IEEE Trans. Indus. Inform., **17**(8), pp. 5605–5614.
[27] Lyubashevsky, V., Peikert, C., and Regev, O., 2013, "On Ideal Lattices and Learning With Errors Over Rings," J. ACM, **60**(6), pp. 1–35.
[28] IPCC, 2001, "Climate Change 2001: Mitigation: Contribution of Working Group III to the Third Assessment Report of the Intergovernmental Panel on Climate Change," https://www.ipcc.ch/site/assets/uploads/2018/03/WGII_TAR_full_report-2.pdf, Accessed January 15, 2024.
[29] Wang, Q., and Yang, H., 2020, "Sensor-Based Recurrence Analysis of Energy Efficiency in Machining Processes," IEEE Access, **8**, pp. 18326–18336.