ImDiffusion: Imputed Diffusion Models for Multivariate Time Series Anomaly Detection

Yuhang Chen[*]
Peking University
2101210553@pku.edu.cn

Chaoyun Zhang[†]
Microsoft
chaoyun.zhang@microsoft.com

Minghua Ma
Yudong Liu
Microsoft
minghuama@microsoft.com
bahuangliuhe@pku.edu.cn

Ruomeng Ding[*]
Georgia Institute of Technology
rmding@gatech.edu

Bowen Li
Tsinghua University
libowen.ne@gmail.com

Shilin He
Microsoft
shilin.he@microsoft.com

Saravan Rajmohan
Microsoft 365
saravar@microsoft.com

Qingwei Lin
Dongmei Zhang
Microsoft
{qlin,dongmeiz}@microsoft.com

ABSTRACT

Anomaly detection in multivariate time series data is of paramount importance for large-scale systems. However, accurately detecting anomalies in such data poses significant challenges due to the need for precise data modeling capability. Existing forecasting and reconstruction-based methods struggle to address these challenges effectively. To overcome these limitations, we propose a novel anomaly detection framework named ImDiffusion, which combines time series imputation and diffusion models to achieve accurate and robust anomaly detection. The imputation-based approach employed by ImDiffusion leverages the information from neighboring values in the time series, enabling precise modeling of temporal and inter-correlated dependencies, reducing uncertainty in the data, thereby enhancing the robustness of the anomaly detection process. ImDiffusion further leverages diffusion models as time series imputers to accurately capture complex dependencies. We leverage the step-by-step denoised outputs generated during the inference process to serve as valuable signals for anomaly prediction, resulting in improved accuracy and robustness of the detection process.

We evaluate the performance of ImDiffusion via extensive experiments on benchmark datasets. The results demonstrate that our proposed framework significantly outperforms state-of-the-art approaches in terms of detection accuracy and timeliness. ImDiffusion is further integrated into the real production system in Microsoft and observes a remarkable 11.4% increase in detection F1 score compared to the legacy approach. To the best of our knowledge, ImDiffusion represents a pioneering approach that combines imputation-based techniques with time series anomaly detection, while introducing the novel use of diffusion models to the field.

PVLDB Reference Format:
Yuhang Chen, Chaoyun Zhang, Minghua Ma, Yudong Liu, Ruomeng Ding, Bowen Li, Shilin He, Saravan Rajmohan, Qingwei Lin, and Dongmei Zhang. ImDiffusion: Imputed Diffusion Models for Multivariate Time Series Anomaly Detection. PVLDB, 17(3): 359 - 372, 2023.
doi:10.14778/3632093.3632101

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/17000cyh/IMDiffusion.git.

1 INTRODUCTION

The efficient operation of large-scale systems or entities heavily relies on the generation and analysis of extensive and high-dimensional time series data. These data serve as a vital source of information for continuous monitoring and ensuring the optimal functioning of these systems. However, within these systems, various abnormal events may occur, resulting in deviations from the expected downstream performance of numerous applications [4, 31, 60]. These anomalous events can encompass a broad spectrum of issues, including production faults [12, 44], delivery bottlenecks [28], system defects [74, 76], or irregular heart rhythms [37]. When different time series dimensions are combined, they form a multivariate time series (MTS). The detection of anomalies in MTS data has emerged as a critical task across diverse domains. Industries spanning manufacturing, finance, and healthcare monitoring, have recognized the importance of anomaly detection in maintaining operational efficiency and minimizing disruptions [29, 60], and the field of MTS anomaly detection has garnered significant attention from both academia and industry [2, 5, 7, 9, 43].

However, achieving accurate anomaly detection on MTS data is not straightforward, as it necessitates precise modeling of time series data [4, 47, 78]. The complexity of modern large-scale systems introduces additional challenges, as their performance is monitored by multiple sensors, generating heterogeneous time series data that encompasses multidimensional, intricate, and interrelated temporal information [38, 46]. Modeling complex correlations like these requires a high level of capability from the model. Furthermore, time series data often displays significant variability [45], leading to increased levels of uncertainty. This variability can sometimes result in erroneous identification of anomalies. This adds complexity

[*] This work was completed during their internship at Microsoft Research Asia.
[†] Corresponding author.

Proceedings of the VLDB Endowment, Vol. 17, No. 3 ISSN 2150-8097.
doi:10.14778/3632093.3632101

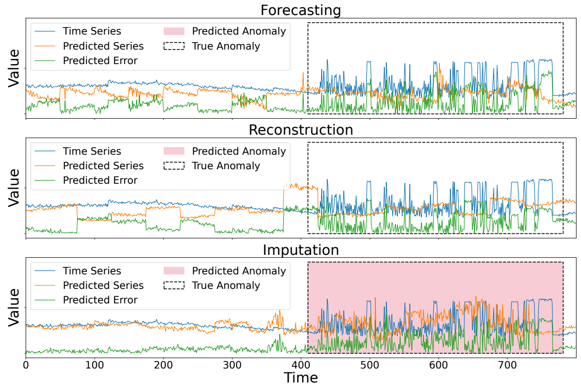

Figure 1: Examples of reconstruction, forecasting and imputation modeling of time series for anomaly detection.

to the anomaly detection process, as the detector must effectively differentiate between stochastic anomalies and other variations to achieve robust detection performance [66, 87].

The aforementioned challenges have spurred the emergence of numerous self-supervised learning solutions aimed at automating anomaly detection. Recent methods can be classified into various categories [60], where forecasting [51, 88] and reconstruction-based [46, 70, 85] approaches have been most widely employed. The former leverages past information to predict future values in the time series and utilizes the prediction error as an indicator for anomaly detection. However, future time series values can exhibit high levels of uncertainty and variability, making them inherently challenging to accurately predict in complex real-world systems. Relying solely on forecasting-based methods may have a detrimental impact on anomaly detection performance [38, 48]. On the other hand, reconstruction-based methods encode entire time sequences into an embedding space. Anomaly labels are then inferred based on the reconstruction error. Since these approaches operate and need to reconstruct the entire time series, their performance heavily relies on the capabilities of the reconstruction model [46]. In cases where the original data exhibit heterogeneity, complexity, and interdependencies, reconstruction-based methods may encounter challenges in achieving low overall reconstruction error and variance [3, 36]. As a result, the anomaly detection performance of such approaches may be sub-optimal. Given these considerations, there is a clear need to rethink and enhance forecasting and reconstruction approaches to achieve accurate and robust anomaly detection.

To address these challenges and overcome the limitations of existing approaches, we propose a novel anomaly detector named ImDiffusion. This detector combines the use of time series imputation [18] and diffusion models [24] to achieve accurate and robust anomaly detection. ImDiffusion employs dedicated grating data masking into the time series data, creating unobserved data points. It then utilizes diffusion models to accurately model the MTS and impute the missing values caused by the data masking. The imputation error is subsequently used as an indicator to determine the anomalies. The imputation-based approach employed by ImDiffusion offers distinct advantages over forecasting and reconstruction methods. Firstly, it leverages neighboring values in the time series as additional conditional information, enabling a more accurate modeling of the temporal and inter-correlated dependencies present in MTS. Secondly, the reference information from neighboring values helps to reduce uncertainty in predictions, and thereby enhancing the robustness of the detection process. Fig. 1 presents an example in which forecasting, reconstruction, and imputation methods are employed to predict a time series using diffusion models. The forecasting method employs a 50-step MTS for observation and predicts the subsequent 50-step MTS. The reconstruction method recovers the entire 100-step MTS. Meanwhile, the imputation method is carried out using the grating data masking. Observe that while all approaches yield comparable errors during the outlier period, the imputation approach achieves a lower error within the normal range due to its superior MTS modeling ability. This attribute enables it to establish a more distinct decision boundary for anomaly identification. As a result, only the imputation method successfully identifies the period of anomaly. We therefore employ time series imputation for accurate self-supervised modeling of time series, which forms the foundation of our proposed ImDiffusion.

To enhance the performance of anomaly detection, ImDiffusion leverages the exceptional unsupervised modeling capability of diffusion models [24] for imputation. Diffusion models have demonstrated superior performance in unsupervised image generation, surpassing traditional generative models such as GANs [21] and VAEs [33]. They have also been successfully applied to model complex temporal and inter-metric dependencies in MTS, showcasing remarkable abilities in forecasting [58] and imputation [68]. We employ a dedicated diffusion model as the time series imputer, replacing traditional forecasting and reconstruction models. This brings several advantages to anomaly detection, namely *(i)* it enables better modeling of complex correlations within MTS data; *(ii)* it allows for stochastic modeling of time series through the noise/denoising processes involved in the imputation; *(iii)* the step-by-step outputs generated during the imputation inference serve as additional signals for determining the anomaly labels in an ensemble manner. These unique advantages of diffusion models enable precise capturing of the complex dependencies and inherent stochasticity present in time series data, and further enhance the robustness of anomaly detection through ensembling techniques.

By integrating imputation and diffusion models, our proposed ImDiffusion achieves exceptional accuracy and timeliness in anomaly detection for both offline and online evaluation in real production. Overall, this paper presents the following contributions:

- We introduce ImDiffusion, a novel framework based on the imputed diffusion model, which accurately captures the inherent dependency and stochasticity of MTS data, leading to precise and robust anomaly detection.
- We develop a grating masking strategy to create missing values in the data for imputation. This strategy enhances the decision boundary between normal and abnormal data, resulting in improved anomaly detection performance.
- ImDiffusion leverages the step-by-step denoised outputs of the diffusion model's unique inference process as additional signals for anomaly prediction in an ensemble voting manner. This approach further enhances inference accuracy and robustness.
- We conduct extensive experiments comparing ImDiffusion with 10 state-of-the-art anomaly detection baselines on 6 datasets.

Results show that ImDiffusion significantly outperforms other approaches in terms of both detection accuracy and timeliness.

- We integrate ImDiffusion in the real production of the Microsoft email delivery microservice system. The framework exhibits an 11.4% higher detection accuracy compared to the legacy online approach, which significantly improves the system's reliability.

To the best of our knowledge, ImDiffusion is the pioneering approach that combines imputation-based techniques with MTS anomaly detection, and it pushes the methodology boundaries by first applying diffusion models to this field.

2 RELATED WORK

2.1 Time Series Anomaly Detection

Time series anomaly detection is an important problem that has received significant attention [8, 52, 53, 60, 67, 69, 74, 84]. Approaches for this area can be categorized into five main classes based on the underlying detection method [60]. These categories include: *(i)* forecasting methods (*e.g.*, [51, 88]), which predict future values to identify anomalies; *(ii)* reconstruction methods (*e.g.*, [11, 70, 85]), which reconstruct the time series and identify anomalies based on the reconstruction error; *(iii)* encoding methods (*e.g.*, [6]), which encode the time series into a different representation and detect anomalies using this encoding; *(iv)* distance methods (*e.g.*, [9, 10]), which measure the dissimilarity between time series and identify anomalies based on the distance; *(v)* distribution methods (*e.g.*, [20, 25]), which model the distribution of the time series data and detect anomalies based on deviations from the expected distribution; and *(vi)* isolation tree methods (*e.g.*, [13, 41]), which use tree-based structures to isolate anomalies.

Among the various approaches explored in the literature, forecasting and reconstruction methods have gained significant popularity due to their reported effectiveness. For instance, Omnianomaly [66] employs a combination of GRU and VAE to learn robust representations of time series. It also utilizes the Peaks-Over-Threshold (POT) method to dynamically select appropriate thresholds for anomaly detection. MTAD-GAT [88] incorporates a graph-attention network to capture both feature and temporal correlations within time series data. By combining forecasting and reconstruction models, it achieves improved anomaly detection performance. MAD-GAN [36] takes advantage of the discriminator's loss in a GAN as an additional indicator for detecting anomalies. More recently, TranAD [70] introduces attention mechanisms in transformer models and incorporates adversarial training to jointly enhance the accuracy of anomaly detection.

2.2 Diffusion Model

Recently, diffusion models [40, 77] have garnered increasing attention in the field of AI generated content [57, 59]. While their potential in the domain of time series modeling and anomaly detection is relatively new, researchers have begun to explore their application in these areas. For instance, CSDI [68] utilizes a probabilistic diffusion model for time series imputation, outperforming deterministic baselines. TimeGrad [58] applies diffusion models in an autoregressive manner to generate future time sequences for forecasting. This approach achieves good performance in extrapolating into the future while maintaining computational tractability. Additionally, diffusion models have been employed in time series generation. In [39], diffusion models are used as score-based generative models to synthesize time-series data, resulting in superior generation quality and diversity compared to baseline approaches.

Diffusion models have also been explored for image anomaly detection. In [75], denoising diffusion implicit models [65] are combined with classifier guidance to identify anomalous regions in medical images. This produces highly detailed anomaly maps without the need for a complex training procedure. Similarly, in [56], diffusion models are used to eliminate bias and mitigate accumulated prediction errors, thereby enhancing anomaly segmentation in CT data. The DiffusionAD [86] formulates anomaly detection as a "noise-to-norm" paradigm, requiring only one diffusion reverse process step to achieve satisfactory performance in image anomaly detection. This significantly improves the inference efficiency.

3 PRELIMINARY

3.1 Multivariate Time Series Anomaly Detection

We consider a collection of MTS denoted as $\mathcal{X}$, which encompasses measurements recorded from timestamp 1 to L. Specifically:

$$\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots \mathbf{x}_L\}, \tag{1}$$

where $\mathbf{x}_l \in \mathbb{R}^K$ represents an K-dimensional vector at time l, *i.e.*, $\mathbf{x}_l = \{x_l^1, x_l^2, \cdots x_l^K\}$. The objective of MTS anomaly detection is to determine whether an observation $\mathbf{x}_l$ is anomalous or not. By employing $y_l \in \{0, 1\}$ to indicate the presence of an anomaly (with 0 denoting no anomaly and 1 denoting an anomaly), the goal transforms into predicting a sequence of anomaly labels for each timestamp, namely $Y = \{y_1, y_2, \cdots, y_L\}$.

3.2 Time Series Imputation

ImDiffusion leverages the prediction error resulting from the imputation [68] of intentionally masked values within a time series to infer the anomaly labels. The mask is denoted as $\mathcal{M} = \{m_{l\in 1:L, k\in 1:K}\} \in \{0, 1\}$, where $m = 1$ indicates that x_l^k is observed, while 0 signifies that it is missing. The mask $\mathcal{M}$ possesses the same dimensionality as the time series $\mathcal{X}$, *i.e.*, $\mathcal{M} \in \mathbb{R}^{T\times K}$. The application of the mask $\mathcal{M}$ to the original time series $\mathcal{X}$ yields a new partially observed time series $\mathcal{X}^{\mathcal{M}}$, which can be expressed as:

$$\mathcal{X}^{\mathcal{M}} = \mathcal{X} \odot \mathcal{M}. \tag{2}$$

Here, the symbol $\odot$ represents the Hadamard product. Let $\mathcal{X}^{\mathcal{M}_0}$ represent the masked value where $m_{l,k} = 0$, and $\mathcal{X}^{\mathcal{M}_1}$ represent the observed values where $m_{l,k} = 1$, the objective of the imputation process is to estimate the missing values in $\mathcal{X}^{\mathcal{M}}$, *i.e.*, $p(\mathcal{X}^{\mathcal{M}_0} \mid \mathcal{X}^{\mathcal{M}_1})$. Interpolation [30, 35, 62] and forecasting [79, 80, 90], can be considered as instances of time series imputation.

3.3 Denoising Diffusion Model

Our ImDiffusion is based on the diffusion models [64], a well-known generative model that draws inspiration from non-equilibrium thermodynamics. Diffusion models follow a two-step process for data generation. Firstly, it introduces noise to the input incrementally, akin to a forward process. Secondly, it learns to generate new samples by progressively removing the noise from a sample noise vector, thereby resembling a reverse process. During the forward process, Gaussian noise is incrementally added to the initial input sample $\mathcal{X}_0$ over T steps. Mathematically, this can be represented as $q(\mathcal{X}_{1:T} \mid \mathcal{X}_0) := \prod_{t=1}^{T} q(\mathcal{X}_t \mid \mathcal{X}_{t-1})$, where

$$q(\mathcal{X}_t \mid \mathcal{X}_{t-1}) := \mathcal{N}(\mathcal{X}_t; \sqrt{1-\beta_t}\mathcal{X}_{t-1}, \beta_t \mathbf{I}). \tag{3}$$

Here, β is a positive noise level constant that can either be learned or predefined. The forward process is parameterized as a Markov chain, as the values of $\mathcal{X}_t$ solely depend on $\mathcal{X}_{t-1}$. The final step, $\mathcal{X}_T$, is fully corrupted and becomes random noise. Its distribution can be expressed in closed form as $q(\mathcal{X}_T \mid \mathcal{X}_0) = \mathcal{N}(\mathcal{X}_T; \sqrt{\alpha_t}\mathcal{X}_0, (1-\alpha_t)\mathbf{I})$, where $\tilde{\alpha}_t := 1 - \beta_t$ and $\alpha_t := \prod_{i=1}^{t} \tilde{\alpha}_i$. Next, $\mathcal{X}_T$ can be represented as $\mathcal{X}_T = \sqrt{\alpha_T}\mathcal{X}_0 + (1-\alpha_T)\epsilon$, where $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

Conversely, we employ a machine learning model with learnable parameters Θ to denoise $\mathcal{X}_T$ and reconstruct $\mathcal{X}_0$. This is accomplished by iteratively computing the following Gaussian transitions:

$$p_\Theta(\mathcal{X}_{t-1} \mid \mathcal{X}_t) := \mathcal{N}(\mathcal{X}_{t-1}; \mu_\Theta(\mathcal{X}_t, t), \Sigma_\Theta(\mathcal{X}_t, t)\mathbf{I}). \tag{4}$$

As a result, the joint distribution can be expressed as $p_\Theta(\mathcal{X}_{0:T}) = p(\mathcal{X}_T) \prod_{t=1}^{T} p_\Theta(\mathcal{X}_{t-1} \mid \mathcal{X}_t)$.

The Denoising Diffusion Probabilistic Model (DDPM) [24] simplifies the reverse process by adopting a fixed variance, result in:

$$\mu_\Theta(\mathcal{X}_t, t) := \frac{1}{\alpha_t}\left(\mathcal{X}_t - \frac{\beta_t}{\sqrt{1-\alpha_t}}\epsilon_\Theta(\mathcal{X}_t, t)\right), \ \ \Sigma_\Theta(\mathcal{X}_t, t) = \sqrt{\tilde{\beta}_t}. \tag{5}$$

Here $\tilde{\beta}_t = \begin{cases} \frac{1-\alpha_{t-1}}{1-\alpha_t}\beta_t, & t > 1 \\ \beta_1, & t = 1 \end{cases}$, and ϵ_Θ represents a trainable denoising function. By employing Jensen's inequality and the speeding-up parameterization from DDPM, the reverse process can be solved by training a model to optimize the following objective function:

$$\min_\Theta \mathcal{L}(\Theta) := \min_\Theta \mathbb{E}_{\mathcal{X}_0 \sim q(\mathcal{X}_0), \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), t} \mid\mid \epsilon - \epsilon_\Theta(\mathcal{X}_t, t) \mid\mid^2, \tag{6}$$

where $\mathcal{X}_t = \sqrt{\alpha_t}\mathcal{X}_0 + (1-\alpha_t)\epsilon$, $\mathcal{X}_0$ is the complete data sample that is unaffected by diffusion process noise, and $q(\mathcal{X}_0)$ is its distribution [24]. The denoising function ϵ_Θ is responsible for estimating the noise added to the corrupted input $\mathcal{X}_t$. Once trained, given an arbitrary noise vector, we can generate a new sample by progressively denoising using $\mathcal{X}_t$ and obtain a final complete sample.

4 THE DESIGN OF IMDIFFUSION

ImDiffusion relies on time series imputation and utilizes the imputed error as a signal for anomaly detection. The imputation process is carried out in a self-supervised learning manner, where we intentionally introduce masks to the MTS, creating missing values that need to be imputed. We then train a diffusion model using ImTransformer designed specifically for imputation and subsequent anomaly detection tasks. During the inference phase, we leverage the intermediate output of the ImTransformer at different denoising steps t as additional information to collectively determine the anomaly label. This ensemble approach enhances the accuracy and robustness of ImDiffusion, further improving its performance.

4.1 Imputed Diffusion Models

Time series anomaly detection often relies on the construction of prediction models that accurately capture the distribution of normal data. These models are expected to exhibit higher prediction errors when anomalies occur, thereby serving as indicators and providing a decision boundary for detecting anomalies. Two commonly used types of prediction models for anomaly detection are *(i)* reconstruction models, which encode the entire time series into a representation that can be reconstructed using a decoder; *(ii)* forecasting models, which aim to predict future values of the time series based on historical observations [27]. However, both types of prediction models have their limitations in terms of their capacity for time series modeling. When applying the diffusion model, the reconstruction method involves corrupting the entire MTS into a complete noise vector for reconstruction. However, this introduces a significant level of uncertainty, particularly when there is a lack of conditional information. Similarly, forecasting models face challenges in accurately predicting future values, especially in the presence of anomalies, further contributing to the uncertainty.

To overcome the limitations of traditional reconstruction [48, 66, 85] and forecasting models [32, 51, 88], we propose the use of time series imputation as the underlying prediction model for anomaly detection in ImDiffusion. We further enhance the imputation capacity of the model by incorporating state-of-the-art diffusion models. This approach offers several advantages. Firstly, it enables enhanced estimation of the data distribution by leveraging the availability of unmasked data values. This leads to improved understanding of the underlying data distribution. Secondly, the imputation-based prediction process stabilizes the inference of the diffusion model, resulting in reduced variance in its predictions. This increased stability enhances the reliability of the model's predictions. Lastly, incorporating imputation-based prediction improves the overall accuracy and robustness of subsequent anomaly detection. By combining diffusion models for time series imputation, ImDiffusion achieves accurate modeling of time series data, resulting in superior performance in anomaly detection tasks.

We begin by introducing the use of score-based diffusion models for MTS data imputation [68]. There are two main categories of diffusion models employed for time series imputation, distinguished by the type of input information they utilized as follows:

- **Conditioned Diffusion Models**: These models estimate the masked values conditioned on the observed data, specifically $p(\mathcal{X}^{\mathcal{M}_0} \mid \mathcal{X}^{\mathcal{M}_1})$. In this case, the observed values $\mathcal{X}^{\mathcal{M}_1}$ are not corrupted by noise and are directly provided as input in the reverse process.
- **Unconditional Diffusion Models**: For the unconditional version [68], both masked and unmasked values are corrupted by noise in the forward process. Instead of directly providing the observed data, it retains the ground-truth noise added to the unmasked values as reference inputs. This leads to the estimation of $p(\mathcal{X}^{\mathcal{M}_0} \mid \epsilon_{1:T}^{\mathcal{M}_1})$, where $\epsilon_{1:T}^{\mathcal{M}_1}$ represents the noise sequence added to the unmasked values $\mathcal{X}^{\mathcal{M}_1}$ during the forward process.

Conditional diffusion models generally outperform unconditional diffusion models in the task of imputation, resulting in lower overall prediction errors [68]. This is because conditional models benefit from the direct inclusion of ground-truth unmasked data as input, which serves as reliable references for neighboring values. However, it is important to recognize the distinction between the objectives of imputation and anomaly detection. While imputation

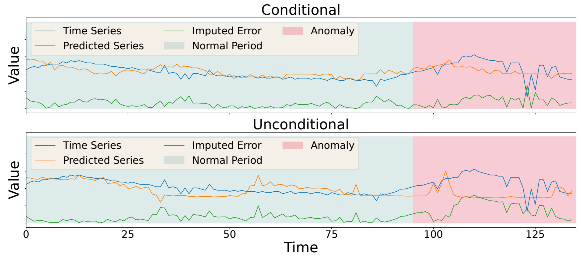

Figure 2: Example cases of conditional/unconditional diffusion models for time series anomaly detection.

aims to minimize the error between predictions and ground truth for all data points, anomaly detection requires a clear boundary between normal and abnormal points, achieved by minimizing imputation errors only for normal data and maximizing errors for anomaly points. During the inference phase, when anomaly points happen to be unmasked and used as inputs for the prediction model, the prediction error for neighboring anomaly points is also reduced. Consequently, the prediction error becomes indistinguishable between normal and abnormal points, compromising the effectiveness of subsequent anomaly detection. The existence of unmasked anomaly points during inference blurs the clear boundary in the prediction error that is vital for accurate anomaly detection.

To address this issue, we employ unconditional imputed diffusion models, which utilize the forward noise $\epsilon_{1:T}^{\mathcal{M}_1}$ as a reference for unmasked data input, rather than directly feeding the data values. By using the forward noise, we avoid explicitly revealing the exact values, even when anomaly points are unmasked. However, the forward noise still provides indirect information about the unmasked data, serving as a weak hint for the model. The unmasked data can be perfectly recovered in the reverse process by subtracting the noise from the observed values step-by-step. The lower subplot in Fig. 2 demonstrates the application of an unconditional diffusion model. A notable distinction from the conditional model (upper subplot) is the substantial difference in imputed error between normal and abnormal data points. This significant gap in imputed error values provides a distinct boundary for the thresholding approach, which improves the anomaly detection performance.

We denote the noise added to the unmasked input from step $t-1$ to t as $\epsilon_t^{\mathcal{M}_1}$. Note that $\epsilon_t^{\mathcal{M}_1}$ is drawn from the same Gaussian distribution in Eq. (3), and serves as the reference for unmasked data in the reverse inference process. Similar to Eq. (4), the unconditional imputed diffusion models estimate the masked values in a reverse denoising fashion but condition on the $\epsilon_t^{\mathcal{M}_1}$ as additional input. Traditional diffusion models lack the capability to incorporate conditional information $\epsilon_t^{\mathcal{M}_1}$ during the denoising process. Consequently, an enhancement is required in order to extend the estimation in Eq. (5) to accommodate conditional information. This can be achieved by modifying the estimation as follows:

$$\mu_\Theta\left(\mathcal{X}_t^{\mathcal{M}_0}, t \mid \epsilon_t^{\mathcal{M}_1}\right) = \mu\left(\mathcal{X}_t^{\mathcal{M}_0}, t, \epsilon_\Theta\left(\mathcal{X}_t^{\mathcal{M}_0}, t \mid \epsilon_t^{\mathcal{M}_1}\right)\right), \tag{7}$$

$$\Sigma_\Theta\left(\mathcal{X}_t^{\mathcal{M}_0}, t \mid \epsilon_t^{\mathcal{M}_1}\right) = \Sigma\left(\mathcal{X}_t^{\mathcal{M}_0}, t\right). \tag{8}$$

By utilizing the denoising function ϵ_Θ and the forward noise for unmasked value $\epsilon_t^{\mathcal{M}_1}$, we can leverage the reverse denoising process of imputed diffusion models to infer the masked values $\mathcal{X}^{\mathcal{M}_0}$. This is accomplished by sampling from the distribution of $\mathcal{X}_t^{\mathcal{M}_0}$. In contrast to anomaly detection methods based on reconstruction and forecasting, the integration of additional information offers valuable signals that assist the diffusion model in generating more reliable predictions. This leads to a reduction in output randomness and variance, while maintaining the confidentiality of abnormal data values. Consequently, it enhances the performance and robustness of subsequent anomaly detection.

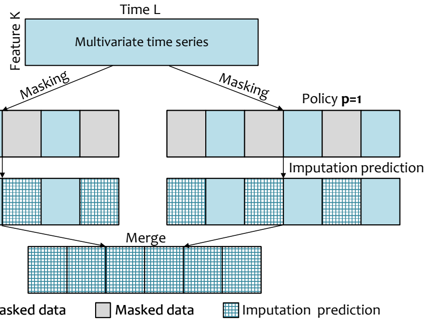

Figure 3: An illustration of the the grating masking and the imputation process under this strategy.

4.2 Design of Data Masking

The ImDiffusion approach leverages deliberate masking, using a mask $\mathcal{M}$ applied to the time series data, to create unobserved points that require imputation. The choice of the masking strategy plays a crucial role in determining the performance of anomaly detection. In this paper, we compare two masking strategies:

- **Random strategy**: This strategy randomly masks data values in the raw time series with a 50% probability [68]. It provides a straightforward and simple masking technique.
- **Grating strategy**: The grating strategy masks the data at equal intervals along the time dimension, as illustrated in Fig. 3. The raw time series is divided into several windows, with masked and unmasked windows appearing in a staggered manner.

For the grating strategy depicted in Fig. 3, two different mask policies indexed by $p \in \{0, 1\}$ are applied to the same time series, resulting in two imputation instances. These two masks are mutually complementary, ensuring that the masked values in mask $p = 0$ are unmasked in mask $p = 1$, and vice versa. This guarantees that all data points are imputed by the ImDiffusion approach, enabling the generation of prediction error signals for anomaly detection. After performing imputation on each masked series individually, the imputation results are merged through simple concatenation. During training and inference, the masking index p is provided to the model, indicating the masking policy applied to reduce ambiguity. This leads to an additional conditional term p on Eq. (4) and (7), while the estimation of Σ_Θ in Eq. (8) remains unchanged, *i.e.*,

$$p_\Theta\left(\mathcal{X}_{t-1}^{\mathcal{M}_0} \mid \mathcal{X}_t^{\mathcal{M}_0}, \epsilon_t^{\mathcal{M}_1}\right) := \mathcal{N}\left(\mathcal{X}_{t-1}^{\mathcal{M}_0}; \mu_\Theta\left(\mathcal{X}_t^{\mathcal{M}_0}, t \mid \epsilon_t^{\mathcal{M}_1}, p\right), \Sigma_\Theta\left(\mathcal{X}_t^{\mathcal{M}_0}, t \mid \epsilon_t^{\mathcal{M}_1}, p\right)\mathbf{I}\right), \tag{9}$$

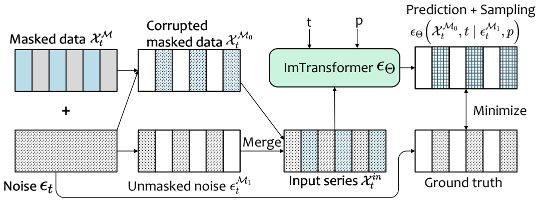
Prediction + Sampling
$\epsilon_\Theta\left(\mathcal{X}_t^{\mathcal{M}_0}, t \mid \epsilon_t^{\mathcal{M}_1}, p\right)$
t
p
Masked data $\mathcal{X}_t^{\mathcal{M}}$
Corrupted masked data $\mathcal{X}_t^{\mathcal{M}_0}$
ImTransformer ϵ_Θ
+
Minimize
Merge
Noise ϵ_t
Unmasked noise $\epsilon_t^{\mathcal{M}_1}$
Input series $\mathcal{X}_t^{in}$
Ground truth

Figure 4: The training process of ImDiffusion.

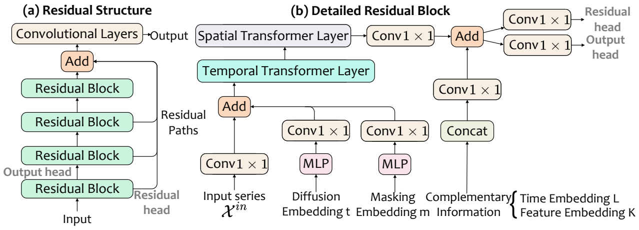

Figure 5: The ImTransformer architecture, with (a) the residual structure; and (b) the details of a residual block.

$$\mu_\Theta\left(\mathcal{X}_t^{\mathcal{M}_0}, t \mid \epsilon_t^{\mathcal{M}_1}\right) = \mu\left(\mathcal{X}_t^{\mathcal{M}_0}, t, \epsilon_\Theta\left(\mathcal{X}_t^{\mathcal{M}_0}, t \mid \epsilon_t^{\mathcal{M}_1}, p\right)\right). \tag{10}$$

The grating strategy introduces a unique characteristic to the imputation, as it can be considered a “partially” reconstruction task. This approach offers several advantages: Firstly, it provides additional information that aids in modeling the time series more effectively. By incorporating the partially reconstructed data, the model gains a better understanding of the underlying patterns and correlations. Secondly, the utilization of the grating strategy allows for a partial glimpse into the future values of the time series within the masked window, akin to forecasting techniques. This enables to improve timeliness in detecting anomalies, as it provides insights into the potential future trajectory of the time series data.

4.3 Training Process of ImDiffusion

The training process of ImDiffusion is illustrated in Fig. 4. Starting with a given data sample time series $\mathcal{X}$, we generate two masked samples $\mathcal{X}^{\mathcal{M}}$ using the grating masking strategy discussed in Sec. 4.2. These masked samples are gradually corrupted by introducing Gaussian noise ϵ_t, resulting in $\mathcal{X}_t^{\mathcal{M}}$. Our objective is to train a model μ_Θ that can effectively denoise $\mathcal{X}_t^{\mathcal{M}}$ and impute the masked values in a step-by-step manner. Given that ImDiffusion employs an unconditional diffusion model, the input series $\mathcal{X}_t^{in}$ is divided into two halves, *i.e.*, $\mathcal{X}^{in} = \{\mathcal{X}_t^{\mathcal{M}_0}, \epsilon_t^{\mathcal{M}_1}\}$. One half contains the corrupted data within the masked regions, denoted as $\mathcal{X}_t^{\mathcal{M}_0}$, while the other half represents the ground truth forward noise applied to the unmasked regions, denoted as $\epsilon_t^{\mathcal{M}_1}$, serving as a reference. These two data sources are concatenated to form the input $\mathcal{X}_t^{in}$, which is then fed into the dedicated transformer-based model ImTransformer for denoising inference, *i.e.*, $\epsilon_\Theta\left(\mathcal{X}_t^{\mathcal{M}_0}, t \mid \epsilon_t^{\mathcal{M}_1}, p\right)$.

The unconditional imputed diffusion models utilize the same parameterization as Eq. (5), with the only difference lying in the form of μ_Θ, which takes additional inputs of unmasked forward noise $\epsilon_t^{\mathcal{M}_1}$ and mask index p. We follow the standard training process of DDPM, beginning with sampling Gaussian noise as masked data at step T, *i.e.*, $\mathcal{X}_T = \sqrt{\alpha_T}\mathcal{X}_0 + (1-\alpha_T)\epsilon$, and optimizing ϵ_Θ by minimizing the following loss function:

$$\min_\Theta \mathcal{L}(\Theta) := \min_\Theta \mathbb{E}_{\mathcal{X}_0 \sim q(\mathcal{X}_0), \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), t} ||\epsilon - \epsilon_\Theta(\mathcal{X}_t^{\mathcal{M}_0}, t \mid \epsilon_t^{\mathcal{M}_1}, p)||^2. \tag{11}$$

Once trained, we can utilize the diffusion model to infer the masked values given a random Gaussian noise $\mathcal{X}_T^{\mathcal{M}_0}$, as well as the forward noise sequence added to the unmasked data $\epsilon_{1:T}^{\mathcal{M}_1}$.

4.4 Imputation with ImTransformer

Drawing inspiration from the studies conducted in [22, 34, 68], which employ hierarchical structures of transformers [71] to capture temporal correlations and interactions among variables, we introduce ImTransformer, a specialized architecture designed for MTS imputation, as illustrated in Fig. 5. It comprises a series of stacked residual blocks, with each containing dedicated components that process the feature and temporal dimensions separately.

The ImTransformer model incorporates four distinct groups of input data: *(i)* the input time series $\mathcal{X}_t^{in}$, *(ii)* diffusion embedding that encodes information related to the current diffusion step t, *(iii)* masking embedding that encodes the masking group p of the current data, and *(iv)* complementary information that embeds the dimensional information of time l and feature k. Each of these groups of data is individually processed by convolutional and/or multilayer perceptron layers to ensure a consistent dimensionality. The embeddings of the inputs are then combined into a single tensor and further processed by a temporal and a spatial transformer layer.

The temporal transformer plays a crucial role in capturing the temporal dependencies within the time series [90]. It enables the dynamic weighting of feature values at different time steps and takes into account the masked status of features. The attention mechanism employed in the temporal transformer provides the necessary flexibility for this purpose. Additionally, a 1-layer spatial transformer is employed to capture the interdependencies between different variables at each time step. This spatial transformer allows for adaptive weighting and facilitates interaction between variables. The output of the spatial transformer is combined with the complementary information, creating a residual head for skip connection, as illustrated in Fig. 5. Both the spatial and temporal transformers play crucial roles in the imputation and anomaly detection tasks, as the feature and temporal dimensions may contribute differently to the predictions [70], which can be learned by the attention mechanism [71]. The incorporation of a residual structure [23] further enhances the model capacity by facilitating gradient propagation.

4.5 Ensemble Anomaly Inference

Traditional anomaly detection models typically rely on a single signal, *i.e.*, the prediction error, to determine the anomaly label for testing data. However, relying solely on one signal can lead to unrobust predictions, as the prediction error can be subjective to stochasticity and affected by various random factors. The presence of anomalous data within the training set further raises concerns about the robustness requirement. To address this limitation, we leverage the unique advantage of diffusion models. Unlike traditional models that provide a single-shot prediction, imputed diffusion models progressively denoise the masked data over T steps.

Algorithm 1 The ensemble inference process of ImDiffusion.

```
Inputs:
    Masked data input series $\mathcal{X}_t^{in}$, masking tensors $\mathcal{M}$, a trained denoising model $\epsilon_\Theta$, the forward ground truth noise on unmasked region $\epsilon_{1:T}^{\mathcal{M}_1}$, total denosing step $T$.
Initialise:
    Initial noise vector $\mathcal{X}_T$.
for $t = T$ to 1 do
    Construct two input series $\mathcal{X}_t^{in} = \{\mathcal{X}_t^{\mathcal{M}_0}, \epsilon_t^{\mathcal{M}_1}\}$ with masking $\mathcal{M}$.
    Predicting $\mu_\Theta$, $\Sigma_\Theta$ using the denosing model $\epsilon_\Theta$.
    Sampling using equation (9) and obtain predicted $\mathcal{X}_{t-1}$.
    Compute prediction error $E_t = ||\mathcal{X} - \mathcal{X}_{t-1}||^2$.
end for
for $t = T$ to 1 do
    Computing the anomaly prediction label $Y_t$ using Eq. (12).
end for
Aggregating the voted anomaly prediction $\mathcal{V}_l = \sum_{t=1}^{T} y_{t,l}$.
Computing the final anomaly prediction $y_l = \mathbb{1}(\mathcal{V}_l > \xi)$.
```

This results in at least T intermediate outputs, each having the same dimension as the original time series, which is not available in traditional models. Although these intermediate outputs are not fully denoised, they converge towards the same imputation objective and offer different perspectives on the time series modeling. By appropriately utilizing these outputs, we can uncover the step-by-step reasoning of ImDiffusion and utilize them as additional signals to enhance the robustness and accuracy of anomaly detection.

ImDiffusion utilizes the prediction error at each denoising step t, denoted as $\mathcal{E} = \{E_1, E_2, \cdots, E_T\}$, as input and ensembles them using a function $f(\mathcal{E})$ to determine the final anomaly labels. E_t denotes the prediction error tensor for the imputed output at denoising step t, and it has the same dimension as the original time series $\mathcal{X}$. The ensemble anomaly inference algorithm is presented in Algorithm 1, and Fig. 6 provides an illustration of the process. At each denoising step, ImDiffusion generates a prediction using the denoising model ϵ_Θ and computes the prediction error E_t with respect to the ground truth time series $\mathcal{X}$. The set $\mathcal{E}$ collects the prediction errors at each denoising step, and an ensemble function $f(\mathcal{E})$ is employed to leverage the all-step errors to obtain the final voting signal $\mathcal{V}$ for determining the anomaly label, *i.e.*, $\mathcal{V} = f(\mathcal{E})$.

The design of the ensemble function. ImDiffusion utilizes a voting ensemble mechanism [16] to strengthen the overall anomaly detection process by aggregating anomaly predictions from each denoising step. At each denoising step t, the anomaly prediction label Y_t is determined using the following equation:

$$Y_t = \mathbb{1}(E_t \geq \tau_t), \text{ where } \tau_t = \frac{\sum E_T}{\sum E_t} \cdot \tau_T. \tag{12}$$

Here, τ_T represents the upper percentile of imputed errors at the final denoising step T. The rationale behind this design is to utilize the imputed error at the last step as a baseline and use it as an indicator of imputation quality. The rescaling ratio $\frac{\sum E_T}{\sum E_t}$ measures the imputation quality at each step t. If the ratio is small, it indicates poor imputation quality, and therefore the upper percentile of imputed errors for determining the anomaly label is reduced. In this case, only the label for the timestamp with the highest imputed error and high confidence is retained. Conversely, if the ratio is small, it suggests good imputation quality, and the error threshold for anomaly detection is relaxed. This dynamic adjustment of the threshold allows for adaptability based on the quality of imputation.

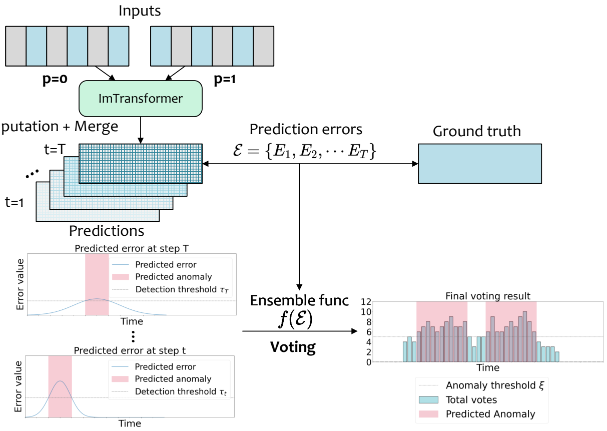

Figure 6: The ensemble anomaly inference of ImDiffusion.

Using Eq. (12), we derive the step-wise anomaly predictions $Y_t = \{y_{t,1}, \cdots, y_{t,L}\}$, where $y_{t,l} = 1$ indicates that the data at time step l is predicted as an anomaly using the imputation at diffusion step t, and $y_{t,l} = 0$ otherwise. To determine the final anomaly prediction at each time step l, we employ a voting mechanism. The voting signal $\mathcal{V}_l$ represents the total number of anomaly votes received at time step l, given by $\mathcal{V}_l = \sum_{t=1}^{T} y_{t,l}$. If a time step receives more than ξ votes as an anomaly across all denoising steps, it is marked as a final anomaly, denoted as $y_l = \mathbb{1}(\mathcal{V}_l > \xi)$. To optimize inference efficiency and ensure correctness, we sample every 3 steps from the last 30 denoising steps for the voting process. This voting mechanism strengthens the ImDiffusion framework by utilizing the intermediate imputed outputs as additional signals. *This is unique to diffusion models, as they generate predictions progressively.*

5 OFFLINE EVALUATION

We conducted a comprehensive offline evaluation of the ImDiffusion for MTS anomaly detection. The evaluation aimed to address the following research questions (RQs):

- **RQ1:** How does ImDiffusion perform compare to state-of-the-art methods in MTS anomaly detection?
- **RQ2:** How effective are each specific design in ImDiffusion?
- **RQ3:** What insights can be gained from each mechanism employed in ImDiffusion?

Implementation. The ImDiffusion framework is implemented using the PyTorch framework [55] and trained on a GPU cluster comprising multiple NVIDIA RTX 1080ti, 2080ti, and 3090 accelerators. The detection thresholds τ for the MSL dataset vary across different subsets, while a fixed value of 0.02 is employed for the other datasets. The voting threshold ξ is dataset-dependent and is specified in the provided code link. As for the baseline models, their hyperparameters and detection thresholds are set based on the information provided in their respective original papers. In cases where these details were not explicitly mentioned, a grid search was conducted to determine the optimal values.

5.1 Datasets, Baselines & Evaluation Metrics

We test the performance of ImDiffusion using 6 publicly available MTS anomaly detection datasets, namely SMD [66], PSM [1], MSL [27], SMAP [27], SWaT [50] and GCP [46]. In order to ensure the completeness of the experiment, we trained and evaluated the ImDiffusion on all subsets of the aforementioned dataset, rather than selectively choosing non-trivial sequences as done in [70]. This may lead to differences in the evaluation metrics compared to the results reported in the original paper.

We evaluate the performance of ImDiffusion by comparing it with 10 state-of-the-art MTS anomaly detection models: *(i)* Isolation forest (IForest) [42] separates the anomaly data point with others for detection. *(ii)* BeatGAN [89] utilizes generative adversarial networks (GANs) [21, 82] to reconstruct time series and detect anomalies. *(iii)* LSTM-AD [49] employs LSTM [26, 79] to forecast future values and uses the prediction error as an indicator of anomalies. *(iv)* InterFusion [38] captures the interaction between temporal information and features to effectively identify inter-metric anomalies. *(v)* OmniAnomaly [66] combines GRU [14] and VAE [54] to learn robust representations of time series and utilizes the Peaks-Over-Threshold (POT) [63] method for threshold selection. *(vi)* GDN [15] introduces graph neural networks into anomaly detection and leverages meta-learning methods to combine old and new knowledge for anomaly identification. *(vii)* MAD-GAN [36] employs GANs [21, 82] to recognize anomalies by reconstructing testing samples from the latent space. *(viii)* MTAD-GAT [88] utilizes Graph Attention Network (GAT) [72] to model MTS and incorporates forecasting-based and reconstruction-based models to improve representation learning [81]. *(ix)* MSCRED [85] uses ConvLSTM networks [61, 80, 83] to capture correlations among MTS and operates as an anomaly detector. *(x)* TranAD [70] leverages transformer models to perform anomaly inference by considering the broader temporal trends in the data. We compare ImDiffusion with these models for evaluation.

In line with previous studies [38, 70, 88], we evaluate the anomaly detection accuracy of both the baseline models and our proposed ImDiffusion using precision, recall, and F1 score. Note that we conducted 6 independent runs for each baseline model and ImDiffusion, and report the average performance. Additionally, we provide the standard deviation of the F1 score (F1-std) in the 6 runs to assess the stability and robustness of all methods examined in this investigation. We also utilize the R-AUC-ROC evaluation metric introduced in [52] to provide a threshold-independent accuracy assessment tailored to range-based anomalies. This metric mitigates the bias introduced by threshold selections and offers a different perspective on the performance of anomaly detection methods by using continuous buffer regions. Further, we utilize the Average Sequence Detection Delay (ADD) metric proposed in [17] to evaluate the speed and timeliness of anomaly detection provided by each approach. The ADD metric is defined as follows:

$$\mathrm{ADD} = \frac{1}{S}\sum_{i=1}^{S}(\mathcal{T}_i - \varrho_i), \tag{13}$$

where ϱ_i represents the start time of anomalous event i, $\mathcal{T}_i \geq \varrho_i$ denotes the corresponding detection delay time by the anomaly detector, and S indicates the total number of anomalous events. A smaller value of ADD indicates a more timely detection of anomalies, which is crucial in real-world detection scenarios.

5.2 Anomaly Detection Performance (RQ1)

5.2.1 Accuracy Performance.

We first present the precision, recall, F1 and R-AUC-PR performance of ImDiffusion and the baseline methods in Table 1 for each of the six datasets considered in this study. Please note that all the results presented in the table are the average values obtained from 6 individual runs, which allows us to assess the robustness of each detector. Additionally, the F1-std. (standard deviation) provides an indication of the variability of the F1 scores across these runs. represents the standard deviation across the 6 runs. Additionally, Table 2 displays the average performance across all six datasets. Notably, ImDiffusion overall demonstrates exceptional performance in terms of all evaluation metrics, namely precision (92.98%), recall (93.01%), and F1 score (92.84%) and R-AUC-PR (29.86%). It achieves the highest average scores across six datasets, surpassing the performance of the other baseline methods. In particular, ImDiffusion exhibits at least a 2.4% increase in precision, a 4.67% increase in recall, a 3.97% increase in F1 score and a 4.85% increase in and R-AUC-PR compared to the other baselines. These results demonstrate the effectiveness of the imputation approach and diffusion models employed in ImDiffusion.

Furthermore, despite that diffusion models require sampling at every denoising step, introducing randomness, the F1-std (0.0083) calculated from 6 independent runs remains relatively small compared to other baselines, ranking second lowest among all approaches. This indicates the remarkable robustness of ImDiffusion. It can be attributed to two key design elements in ImDiffusion: *(i)* the imputation methods leverage neighboring information for self-supervised modeling, reducing prediction uncertainty, and *(ii)* the dedicated ensemble mechanism aggregates votes for step-wise anomaly inference, further reducing prediction variance. We provide a more detailed ablation study in Sec. 5.3.1 and 5.3.2.

Upon closer examination of the dataset-specific performance in Table 1, we observe that ImDiffusion achieves the highest F1 score in 5 out of the 6 datasets. The exception is the MSL dataset, where TranAD outperforms ImDiffusion. This can be attributed to the fact that it is specifically designed to capture the internal correlations across different dimensions, which are the prominent characteristics of the MSL dataset. A plausible solution to reinforce ImDiffusion is to explicitly model these dependencies through hierarchical inter-metric embedding, as employed in InterFusion [38]. However, ImDiffusion also takes a different approach by leveraging the exceptional self-supervised learning ability of diffusion models and the spatial transformer in ImTransformer to capture correlations and provide a more general solution across various datasets. This enables ImDiffusion to achieve competitive performance in most datasets and surpass other baselines. Furthermore, we observe that ImDiffusion also achieves the highest R-AUC-PR in 4 out of the 6 datasets. This highlights the robustness of ImDiffusion to threshold selection and its consistent ability to deliver accurate predictions in detecting range anomalies.

Table 1: The Precision (P), Recall (R), F1 and R-AUC-ROC of all anomaly detectors on benchmark datasets. The average values of P, R, F1 and R-AUC-ROC were calculated from 6 individual runs, while F1-std. is the standard deviation across the 6 runs.

Method	SMD					PSM					SWaT				
	P	R	F1	F1-std.	R-AUC-PR	P	R	F1	F1-std.	R-AUC-PR	P	R	F1	F1-std.	R-AUC-PR
IForest	0.2030	0.2130	0.1799	0.0138	0.0257	0.6630	0.4919	0.5641	0.0070	0.2058	0.9764	0.6650	0.7907	0.0020	0.0685
BeatGAN	0.9013	0.8894	0.8797	0.0058	0.3200	0.9204	0.8767	0.8975	0.0178	0.3453	0.9606	0.7020	0.8107	0.0022	0.3215
LSTM-AD	0.3361	0.3229	0.2639	0.0123	0.0399	0.9050	0.7707	0.8313	0.0036	0.2561	**0.9925**	0.6737	0.8026	0.0013	0.3118
InterFusion	0.8815	0.9071	0.8772	0.0226	0.3012	0.9533	0.9128	0.9326	0.0036	0.1896	0.8683	0.8530	0.8600	0.0309	0.1477
OmniAnomaly	0.8751	0.9052	0.8775	0.0083	0.2525	0.9551	0.8859	0.9191	0.0060	0.3718	0.9749	0.7500	0.8470	0.0271	**0.3722**
GDN	0.8460	0.7862	0.7865	0.0109	0.1637	0.8750	0.8385	0.8564	**0.0000**	0.3230	0.1311	0.0585	0.0808	**0.0009**	0.1318
MAD-GAN	0.8851	0.9045	0.8803	0.0384	0.2295	0.8596	0.8838	0.8698	0.0339	0.4416	0.7918	0.5423	0.6385	0.3048	0.2633
MTAD-GAT	0.8836	0.8330	0.8463	0.0316	0.3006	0.8763	0.8725	0.8744	0.0000	0.4116	0.8468	0.8224	0.8344	0.0067	0.3196
MSCRED	0.8567	0.9038	0.8426	**0.0002**	0.2601	0.9555	0.6857	0.7965	0.0102	0.3846	0.4823	0.4065	0.4407	0.3408	0.1668
TranAD	0.8906	0.8982	0.8785	0.0023	0.2941	0.9506	0.8951	0.9220	0.0045	0.3994	0.7025	0.7266	0.6886	0.1089	0.1670
ImDiffusion	**0.9520**	**0.9509**	**0.9488**	0.0039	**0.3821**	**0.9811**	**0.9753**	**0.9781**	0.0072	**0.4711**	0.8988	**0.8465**	**0.8709**	0.0124	0.1939

Method	SMAP					MSL					GCP				
	P	R	F1	F1-std.	R-AUC-PR	P	R	F1	F1-std.	R-AUC-PR	P	R	F1	F1-std.	R-AUC-PR
IForest	0.2886	0.7671	0.4163	0.0026	0.1096	0.6059	0.5328	0.5334	0.0309	0.0942	0.8055	0.7385	0.7370	0.0120	0.1558
BeatGAN	0.8915	0.6781	0.7663	0.0162	**0.1303**	0.7782	0.8512	0.8102	0.0342	0.1421	0.9865	0.9630	0.9717	0.0074	0.2414
LSTM-AD	0.7841	0.5630	0.6533	0.0382	0.1099	0.7330	0.5745	0.6378	0.1473	0.1066	0.9591	0.9575	0.9553	0.0013	0.2610
InterFusion	0.8788	0.7704	0.8204	0.0077	0.1457	0.7688	**0.9464**	0.8442	0.0330	0.1083	0.9361	0.9720	0.9092	0.0005	0.2846
OmniAnomaly	0.8407	**0.9674**	0.8995	0.0078	0.0978	0.8321	0.8125	0.8221	0.0121	0.1290	0.9572	0.9796	0.9668	0.0027	0.2029
GDN	0.9689	0.5401	0.6936	0.0037	0.0961	0.8668	0.8072	0.8360	0.0004	0.1295	0.9648	0.9628	0.9589	0.0011	0.2096
MAD-GAN	0.9547	0.5474	0.6952	0.0013	0.0990	0.7047	0.7841	0.7423	**0.0000**	0.1301	0.9766	0.9558	0.9605	0.0055	0.1867
MTAD-GAT	**0.9718**	0.5259	0.6824	**0.0012**	0.1083	0.7321	0.7616	0.7432	0.0200	0.1278	0.9490	0.9523	0.9461	0.0047	0.2210
MSCRED	0.4107	0.8604	0.2712	0.0625	0.1042	0.5008	0.6088	0.4899	0.0788	0.1090	0.9754	0.9735	0.9712	**0.0006**	0.2068
TranAD	0.8224	0.8502	0.8360	0.0090	0.1077	**0.8951**	0.9297	**0.9115**	0.0051	0.1057	0.9472	0.9812	0.9631	0.0030	0.2026
ImDiffusion	0.8771	0.9618	**0.9175**	0.0095	0.1105	0.8930	0.8638	0.8779	0.0152	**0.2381**	**0.9771**	**0.9825**	**0.9774**	0.0014	**0.3957**

Table 2: P, R, F1, F1-std and R-AUC-ROC performance of all anomaly detectors averaged over six benchmark datasets.

Method	P	R	F1	F1-std.	R-AUC-PR
IForest	0.5904	0.5680	0.5369	0.0114	0.1099
BeatGAN	0.9064	0.8267	0.8560	0.0139	0.2501
LSTM-AD	0.7850	0.6437	0.6907	0.0340	0.1809
InterFusion	0.8811	0.8936	0.8739	0.0164	0.1962
OmniAnomaly	0.9058	0.8834	0.8887	0.0107	0.2377
GDN	0.7754	0.6656	0.7020	**0.0028**	0.1756
MAD-GAN	0.8621	0.7697	0.7978	0.0640	0.2250
MTAD-GAT	0.8766	0.7946	0.8211	0.0107	0.2481
MSCRED	0.6969	0.7398	0.6353	0.0822	0.2053
TranAD	0.8681	0.8802	0.8666	0.0221	0.2128
ImDiffusion	**0.9298**	**0.9301**	**0.9284**	0.0083	**0.2986**

Table 3: The ADD (mean±std.) performance comparison for all approaches. Results are averaged on 6 runs.

Method	SMD	PSM	SMAP	MSL	SWaT	GCP	Average
IsolationForest	90 ± 1	191 ± 17	394 ± 93	123 ± 28	539 ± 20	203 ± 3	257 ± 27
BeatGAN	38 ± 2	166 ± 11	345 ± 23	68 ± 24	607 ± 6	130 ± 13	226 ± 13
LSTM-AD	87 ± 1	224 ± 54	541 ± 51	115 ± 29	627 ± 4	107 ± 1	284 ± 23
InterFusion	**22 ± 2**	40 ± 10	423 ± 4	**32 ± 15**	454 ± 141	141 ± 1	185 ± 29
OmniAnomaly	26 ± 1	121 ± 11	116 ± 38	93 ± 2	550 ± 48	131 ± 6	173 ± 18
GDN	38 ± 1	148 ± 0	402 ± 4	106 ± 2	1478 ± 0	125 ± 0	383 ± 1
MAD-GAN	59 ± 57	122 ± 2	404 ± 20	88 ± 0	926 ± 337	157 ± 0	293 ± 69
MTAD-GAT	90 ± 100	182 ± 0	542 ± 2	96 ± 17	482 ± 80	145 ± 0	256 ± 33
MSCRED	32 ± 0	218 ± 35	622 ± 48	109 ± 30	1065 ± 339	145 ± 3	365 ± 76
TranAD	25 ± 0	127 ± 4	291 ± 2	56 ± 12	657 ± 246	104 ± 11	210 ± 46
ImDiffusion	24 ± 1	**28 ± 1**	**98 ± 31**	46 ± 4	**350 ± 43**	**75 ± 1**	**104 ± 14**

However, in the SWaT and SMAP datasets, we observe a notable reduction in precision for ImDiffusion compared to several baselines. This can be attributed to a slight overfitting exhibited by ImDiffusion on these specific datasets, which leads to increased errors in normal data. Consequently, applying a fixed error threshold results in the identification of more false anomalies, thereby compromising precision. A potential solution could involve the implementation of dynamic thresholding approaches [27] to achieve a better balance between precision and recall. Moreover, mitigating overfitting can be achieved by reducing the complexity of the ImTransformer. These considerations are reserved for future work.

Notably, the performance improvements achieved by ImDiffusion are particularly remarkable in the SMD and PSM datasets, where it outperforms other baselines by at least 6.8% and 5.9% in terms of F1 score and 6.21% and 2.19% in terms of R-AUC-PR, respectively. These two datasets exhibit small distribution deviations between anomalous and normal data [70], and ImDiffusion's unconditional imputation design effectively amplifies the gap in imputed error between normal and abnormal data, contributing to its superior performance. Furthermore, ImDiffusion consistently outperforms other baselines with low F1-std. in the SWaT, SMAP, and GCP datasets, which demonstrates its remarkable robustness. Interestingly, we observe that all approaches demonstrate comparatively lower performance in the SwaT dataset. This can be attributed to the intricate and diverse MTS patterns present in the SWaT dataset, underscored by the dataset's expansive training set size and high dimensionality (51). This leads to challenges in accurate modeling, consequently resulting in inferior anomaly detection performance.

5.2.2 Timeliness Performance. In Table 3, we present the ADD (mean±std.) performance comparison on all datasets over 6 runs,

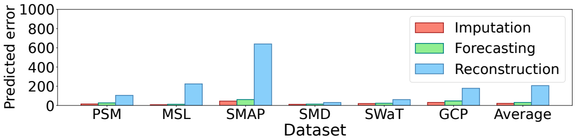

Figure 7: Predicted error of imputation, forecasting and reconstruction approaches on all datasets.

along with their average values. Observe that ImDiffusion demonstrates remarkable performance in this aspect as well. Overall, ImDiffusion achieves the lowest average ADD values (104) with low variance, surpassing other baselines by at least 39.9%. Upon closer examination of dataset-specific performance, it is observed that ImDiffusion consistently outperforms other baselines in 4 out of 6 datasets. This indicates that ImDiffusion is highly sensitive to abnormal points and can capture them at the earliest detection timing. This superior performance can be attributed to the grating masking design employed by ImDiffusion, which enables to partially envision the future values of the time series in the masked regions. A more detailed ablation analysis on the masking strategy is presented in Sec. 5.3.4. The ADD metric holds significant importance in industrial practice, as early detection of anomalies allows for prompt mitigation of failures [19], potentially preventing more severe consequences. The advantage of ImDiffusion in achieving faster anomaly detection makes it a suitable choice for real-world deployment in systems with high reliability requirements.

5.3 Ablation Analysis (RQ2, RQ3)

Next, we conduct a comprehensive ablation analysis to evaluate the effectiveness of each design choice in ImDiffusion, shedding light on how these design choices contribute to enhancing the anomaly detection performance. The aggregated results specific to each dataset are presented in Table 4, while Table 5 showcases the average results across all datasets. The reported results are the average of 6 independent runs. Note that in the tables, "ImDiffusion" *represents the combination of the following designs: Imputation, Ensembling, Unconditional, Grating Masking and full ImTransformer.*

5.3.1 Imputation vs. Forecasting vs. Reconstruction. First, we compare the anomaly detection performance of different MTS modeling approaches, namely imputation, forecasting, and reconstruction. In the case of forecasting and reconstruction, we adopt the same configuration as ImDiffusion, with the only distinction being the forecasting method that predicts future values given historical observations, while the reconstruction method corrupts all values with noise vectors and reconstructs them. Overall, the ImDiffusion framework, which utilizes the imputation method, achieves the highest performance in terms of accuracy and timeliness on average, outperforming other MTS modeling methods by at least 3.18% on F1 score, 1.78% on R-AUC-PR and 26.2% ADD. In addition, we observe that forecasting outperforms reconstruction, indicating that incorporating historical information leads to improved performance of the self-supervised model. Moreover, as shown in Table 4, ImDiffusion achieves the highest F1 score and ADD in 5 out of 6 datasets, and the highest R-AUC-PR on 4 out of 6 datasets, highlighting the accuracy and robustness of the imputation approach.

The performance improvement can be attributed to the superior self-supervised modeling quality achieved through the imputation approach. Fig. 7 illustrates the predicted error of each modeling approach, along with their average values. A lower prediction error signifies a more accurate modeling of MTS, which in turn enhances the performance of anomaly detection. Notably, the imputation approach consistently exhibits the lowest predicted error across all datasets, significantly outperforming the forecasting and reconstruction approaches. These results indicate the imputation approach's superior self-supervised modeling capability. They further validate that enhancing the self-supervised modeling ability contributes to the anomaly detection performance for MTS data.

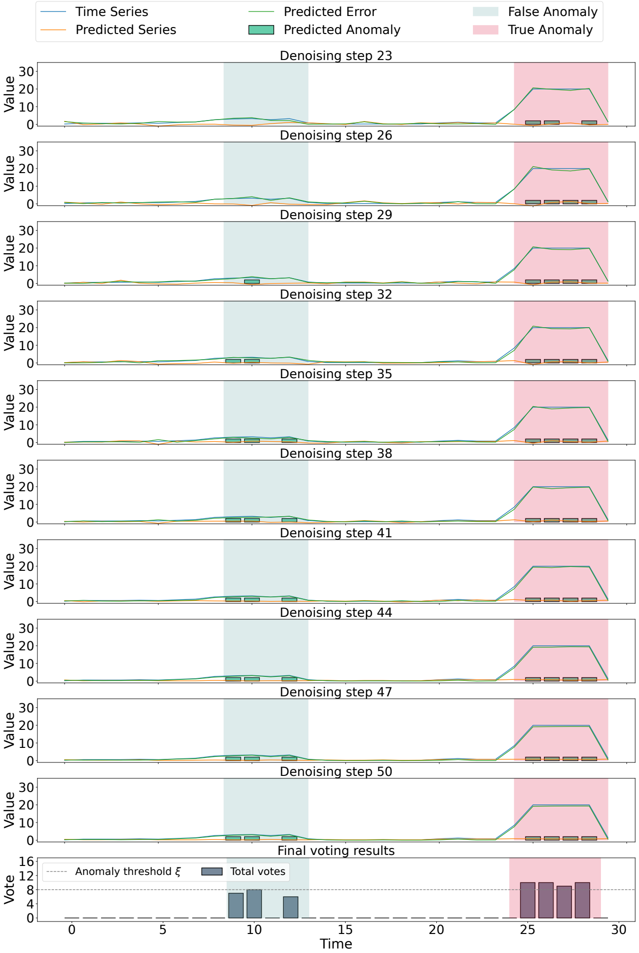

Figure 8: An example of the ensemble inference, showcasing the step-wise prediction and the final voting mechanism.

5.3.2 Ensembling vs. Non-ensembling. Next, we investigate the impact of the ensembling voting mechanism on the anomaly detection performance. The non-ensembling approach solely relies on the final denoised results and applies thresholding on the imputed error for anomaly detection. In comparison, ImDiffusion on average achieves a 0.73% higher F1 score, 6.06% higher R-AUC-PR and a lower 35.8% ADD compared to the non-ensembling approaches. This suggests that the utilization of the ensembling approach enhances both the accuracy and timeliness performance of anomaly detection, particularly for ranged anomalies. Furthermore, ImDiffusion consistently outperforms its counterpart across all datasets.

Table 4: Performance comparison on 6 benchmark datasets for all ablation analysis considered in this paper.

Method	SMD					PSM					SWaT				
	P	R	F1	R-AUC-PR	ADD	P	R	F1	R-AUC-PR	ADD	P	R	F1	R-AUC-PR	ADD
ImDiffusion	0.952	0.951	0.949	0.382	23.7	0.981	0.975	0.978	0.471	28.4	0.899	0.846	0.871	0.194	350.4
Forecasting	0.892	0.918	0.896	0.268	22.6	0.974	0.868	0.914	0.411	96.2	0.895	0.792	0.839	0.290	451.7
Reconstruction	0.641	0.796	0.682	0.106	14.1	0.891	0.898	0.894	0.291	58.3	1.000	0.657	0.793	0.564	663.1
Non-ensemble	0.934	0.953	0.941	0.230	24.5	0.975	0.974	0.974	0.390	30.6	0.898	0.831	0.861	0.248	430.5
Conditional	0.955	0.951	0.951	0.395	22.9	0.977	0.962	0.969	0.425	38.4	0.932	0.849	0.888	0.217	307.0
Random Mask	0.953	0.946	0.946	0.106	23.1	0.976	0.977	0.977	0.291	23.7	0.906	0.872	0.889	0.113	295.2
w/o spatial transformer	0.951	0.947	0.946	0.337	22.2	0.980	0.963	0.971	0.467	31.2	0.940	0.895	0.867	0.353	405.0
w/o temporal transformer	0.899	0.892	0.884	0.281	28.6	0.964	0.973	0.969	0.391	25.8	0.951	0.787	0.861	0.396	375.5

Method	SMAP					MSL					GCP				
	P	R	F1	R-AUC-PR	ADD	P	R	F1	R-AUC-PR	ADD	P	R	F1	R-AUC-PR	ADD
ImDiffusion	0.877	0.962	0.917	0.110	98.4	0.89	0.864	0.878	0.238	46.3	0.977	0.983	0.977	0.396	75.9
Forecasting	0.872	0.946	0.907	0.113	130.6	0.873	0.818	0.843	0.222	64.3	0.979	0.984	0.980	0.381	78.1
Reconstruction	0.879	0.978	0.926	0.120	72.6	0.783	0.662	0.707	0.168	105.3	0.942	0.968	0.952	0.281	59.9
Non-ensemble	0.871	0.963	0.915	0.127	99.0	0.879	0.862	0.870	0.187	57.8	0.956	0.981	0.966	0.246	86.3
Conditional	0.851	0.740	0.787	0.106	287.4	0.872	0.865	0.868	0.271	52.4	0.979	0.978	0.976	0.402	81.7
Random Mask	0.920	0.908	0.913	0.180	293.5	0.888	0.896	0.892	0.168	49.1	0.975	0.980	0.975	0.406	78.8
w/o spatial transformer	0.816	0.579	0.677	0.107	360.0	0.865	0.889	0.876	0.243	48.3	0.981	0.925	0.938	0.462	99.8
w/o temporal transformer	0.873	0.964	0.916	0.110	108.1	0.873	0.863	0.867	0.168	43.3	0.976	0.998	0.986	0.398	67.4

Table 5: Average results over all datasets of ablation analysis.

Method	P	R	F1	R-AUC-PR	ADD
ImDiffusion	0.9298	0.9301	0.9284	0.2986	104
Forecasting	0.9139	0.8876	0.8966	0.2808	141
Reconstruction	0.8559	0.8266	0.8256	0.2550	162
Non-ensemble	0.9187	0.9273	0.9211	0.2380	121
Conditional	0.9278	0.8910	0.9066	0.3026	132
Random Mask	0.9363	0.9298	0.9318	0.2107	127
w/o spatial transformer	0.9224	0.8514	0.8794	0.3280	161
w/o temporal transformer	0.9229	0.9131	0.9139	0.2910	108

Upon closer examination of Table 5, we observe that ImDiffusion exhibits a greater advantage in terms of precision over recall. This indicates that the anomalies detected through ensembling are more likely to be true anomalies, thereby reducing the false positive rate.

Fig. 8 illustrates an example on the SMD dataset, demonstrating how the ensembling mechanism improves the anomaly detection performance. The first 10 subplots depict the time series, imputation prediction, predicted error, and anomaly prediction for each of the 10 denoising steps used in the ensembling voting. The final subplot showcases the aggregated voting results for anomalies. Several key insights can be derived from the figure. Firstly, the imputation results progressively improve with each denoising step, aligning with our expectations as diffusion models perform step-by-step imputation. Secondly, relying solely on the final step can lead to false positive predictions (blue shaded area). However, the ensembling voting mechanism plays a crucial role in correcting these false positives. From step 23 to 32, the false positive data receives fewer votes compared to the true positive region (red shaded area), causing them to fall below the final voting threshold (8 votes) and be eliminated from the ensemble's predicted anomalies. This case study provides a clear illustration of how ensembling can enhance accuracy and robustness.

5.3.3 *Unconditional vs. Conditional Diffusion Models.* We now shift our focus to evaluating the effectiveness of the design of unconditional diffusion models described in Sec. 4.1. In Table 5, we observe that ImDiffusion, which utilizes unconditional diffusion models, achieves superior accuracy and timeliness performance compared to its conditional counterpart, with a 2.1% higher F1 score and a 21.1% lower ADD. This gain is particularly pronounced in the SMAP dataset, which comprises shorter time sequences. The R-AUC-PR values obtained from both approaches are comparable, with the conditional method exhibiting a slight advantage.

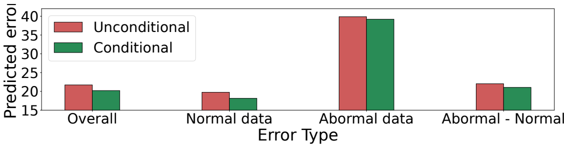

Figure 9: Predicted error of normal/abnormal data comparison on conditional/unconditional diffusion models.

The improvement can be attributed to the unconditional approach, which expands the predicted error between normal and abnormal data. Fig. 9 showcases the overall predicted error, error on normal data, error on abnormal data, and the difference (abnormal data - normal data) averaged for all datasets. Note that larger difference in predicted error between abnormal and normal data generally indicates a more distinct classification boundary for thresholding approaches. Notably, the unconditional approach generally yields higher overall predicted error. This aligns with our expectations, as the conditional approach makes predictions based on the ground truth time series values, providing more direct and focused guidance compared to the unconditional approach, which predicts solely based on the forward noise in the unmasked regions. However, the error difference between abnormal and normal data is amplified by the unconditional method, as illustrated in the figure. This further confirms the effectiveness of the unconditional approach, as it establishes a clearer error decision boundary for thresholding, enabling better discrimination of anomalies.

5.3.4 *Grating Masking vs. Random Masking.* We compare the performance of the grating masking and random masking designs as introduced in Sec. 4.2. Interestingly, on average, the two approaches achieve comparable F1 scores, with random masking slightly outperforming grating masking by 0.4%. The accuracy performance of the two masking designs is also quite similar across datasets. However, it is worth noting that the grating masking design consistently outperforms the random masking design in terms of R-AUC-PR. On average, the grating masking design achieves an 8.79% higher score compared to the random masking design, outperforming its counterpart on 4 out of 6 datasets. This result suggests that ImDiffusion exhibits higher accuracy in detecting ranged anomalies. This is particularly relevant in real-world scenarios where such ranged anomalies occur frequently. In addition, we observe that grating masking exhibits a significant advantage in terms of ADD, with a gain of 18.4%. This can be attributed to the value envisioning property in the windowed masked regions of the grating design, which is absent in random masking. Therefore, grating masking is more suitable for industrial applications where timely detection of anomalies is crucial to ensure system reliability.

5.3.5 Components of ImTransformer. Finally, we conduct an ablation analysis to evaluate the impact of removing individual components of ImTransformer, specifically the spatial and temporal transformers, on anomaly detection performance. As summarized in Table 5, the removal of either the spatial or temporal transformer results in a decrease in F1 and ADD performance compared to the complete ImDiffusion model, albeit to varying degrees. Notably, ImDiffusion without the spatial transformer exhibits poorer performance than when removing the temporal transformer, emphasizing the importance of capturing inter-metric correlations in MTS anomaly detection. This is particularly evident in the SMAP dataset, which also exhibits strong interrelations between metrics, as shown in Table 4. Employing the spatial transformer significantly improve the anomaly detection performance on it.

Conversely, the removal of the temporal transformer also leads to a decline in performance across all metrics. As shown in Table 4, the most substantial impact is observed in the SMD dataset, with a significant performance drop compared to other datasets. This emphasizes the importance of accurately modeling and weighting the time dimension in the SMD dataset, where temporal correlations play a crucial role. Consequently, both the spatial and temporal transformers play pivotal roles in enhancing ImDiffusion's anomaly detection performance.

6 PRODUCTION IMPACT AND EFFICIENCY

The proposed ImDiffusion has been integrated as a critical component within a large-scale email delivery microservice system at Microsoft. This system consists of more than 600 microservices distributed across 100 datacenters worldwide, generating billions of trace data points on a daily basis [73]. ImDiffusion serves as a latency monitor for email delivery, for detecting any delay regression in each microservice, which may indicate the occurrence of an incident. The online latency data for each microservice are sampled at a frequency of every 30 seconds. In order to assess the performance of ImDiffusion, we deployed it online and operated over a period of 4 months. We compared the results obtained by ImDiffusion with a legacy deep learning-based MTS anomaly detector, which has been in operation for years.

Table 6: Online performance of ImDiffusion in production compared to the legacy detector.

Improvement					Inference efficiency
P	R	F1	R-AUC-PR	ADD	[points/second]
9.0%	12.7%	11.4%	14.4%	30.2%	5.8

Table 6 presents the online improvements achieved by ImDiffusion compared to the legacy detector over a period of 4 months[1]. The evaluation of efficiency was conducted on containers equipped with Intel(R) Xeon(R) CPU E5-2640 v4 processors featuring 10 cores. Observe that the replacement of the legacy detector with ImDiffusion has resulted in significant enhancements in anomaly detection accuracy and timeliness, as evidenced by the substantial improvements across all evaluation metrics. Specifically, compared to the previous online solution, ImDiffusion exhibits a performance improvement of 11.4% in terms of F1 score, 14.4% improvement in terms of R-AUC-PR, and 30.2% reduction on ADD. Despite the requirement of multiple inferences to obtain the final results, the online efficiency of ImDiffusion remains well within an acceptable range. Considering that the latency data are sampled every 30 seconds, performing inference at a rate of 5.8 data points per second is more than sufficient to meet the online requirements.

The reliability assessment of a cloud system encompasses two aspects: *(i)* detection accuracy and *(ii)* detection timeliness of anomalies or incidents [19]. The notable performance improvements achieved by ImDiffusion have made a significant impact on the Microsoft email delivery system from the above perspectives, as they have led to considerable time savings in incident detection (TTD), reduced the number of false alarms triggered by the legacy approach, and ultimately enhanced the system's reliability.

7 CONCLUSION

This paper presents ImDiffusion, a novel framework that combines time series imputation and diffusion models to achieve accurate and robust anomaly detection in MTS data. By integrating the imputation method with a grating masking strategy, the proposed approach facilitates more precise self-supervised modeling of the intricate temporal and interweaving correlations that are characteristic of MTS data, which in turn enhances the performance of anomaly detection. Moreover, ImDiffusion employs dedicated diffusion models for imputation, effectively capturing the stochastic nature of time series data. The framework also leverages multi-step denoising outputs unique to diffusion models to construct an ensemble voting mechanism, further enhancing the accuracy and robustness of anomaly detection. Notably, ImDiffusion is the first to employ time series imputation for anomaly detection and to utilize diffusion models in this context. Extensive experiments on public datasets demonstrate that ImDiffusion outperforms state-of-the-art baselines in terms of accuracy and timeliness. Importantly, ImDiffusion has been deployed in real production environments within Microsoft's email delivery system, serving as a core latency anomaly detector and significantly improving system reliability.

[1] To comply with confidentiality requirements, the actual numbers for all metrics are omitted, and only the relative improvements are reported.

REFERENCES

[1] Ahmed Abdulaal, Zhuanghua Liu, and Tomer Lancewicki. 2021. Practical approach to asynchronous multivariate time series anomaly detection and localization. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*. 2485–2494.

[2] Sarah Alnegheimish, Dongyu Liu, Carles Sala, Laure Berti-Equille, and Kalyan Veeramachaneni. 2022. Sintel: A machine learning framework to extract insights from signals. In *Proceedings of the 2022 International Conference on Management of Data*. 1855–1865.

[3] Julien Audibert, Pietro Michiardi, Frédéric Guyard, Sébastien Marti, and Maria A Zuluaga. 2020. USAD: Unsupervised anomaly detection on multivariate time series. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 3395–3404.

[4] Ane Blázquez-García, Angel Conde, Usue Mori, and Jose A Lozano. 2021. A review on outlier/anomaly detection in time series data. *ACM Computing Surveys (CSUR)* 54, 3 (2021), 1–33.

[5] Paul Boniol, Michele Linardi, Federico Roncallo, and Themis Palpanas. 2020. Automated anomaly detection in large sequences. In *2020 IEEE 36th international conference on data engineering (ICDE)*. IEEE, 1834–1837.

[6] Paul Boniol and Themis Palpanas. [n. d.]. Series2Graph: Graph-based Subsequence Anomaly Detection for Time Series. *Proceedings of the VLDB Endowment* 13, 11 ([n. d.]).

[7] Paul Boniol, Themis Palpanas, Mohammed Meftah, and Emmanuel Remy. 2020. Graphan: Graph-based subsequence anomaly detection. *Proceedings of the VLDB Endowment* 13, 12 (2020), 2941–2944.

[8] Paul Boniol, John Paparrizos, Yuhao Kang, Themis Palpanas, Ruey S Tsay, Aaron J Elmore, and Michael J Franklin. 2022. Theseus: navigating the labyrinth of time-series anomaly detection. *Proceedings of the VLDB Endowment* 15, 12 (2022), 3702–3705.

[9] Paul Boniol, John Paparrizos, Themis Palpanas, and Michael J Franklin. 2021. SAND: streaming subsequence anomaly detection. *Proceedings of the VLDB Endowment* 14, 10 (2021), 1717–1729.

[10] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. 2000. LOF: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*. 93–104.

[11] David Campos, Tung Kieu, Chenjuan Guo, Feiteng Huang, Kai Zheng, Bin Yang, and Christian S Jensen. 2022. Unsupervised Time Series Outlier Detection with Diversity-Driven Convolutional Ensembles–Extended Version. (2022).

[12] Yinfang Chen, Huaibing Xie, Minghua Ma, Yu Kang, Xin Gao, Liu Shi, Yunjie Cao, Xuedong Gao, Hao Fan, Ming Wen, Jun Zeng, Supriyo Ghosh, Xuchao Zhang, Chaoyun Zhang, et al. 2023. Empowering Practical Root Cause Analysis by Large Language Models for Cloud Incidents. *arXiv preprint arXiv:2305.15778* (2023).

[13] Zhangyu Cheng, Chengming Zou, and Jianwei Dong. 2019. Outlier detection using isolation forest and local outlier factor. In *Proceedings of the conference on research in adaptive and convergent systems*. 161–168.

[14] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* (2014).

[15] Ailin Deng and Bryan Hooi. 2021. Graph neural network-based anomaly detection in multivariate time series. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 4027–4035.

[16] Thomas G Dietterich. 2000. Ensemble methods in machine learning. In *Multiple Classifier Systems: First International Workshop, MCS 2000 Cagliari, Italy, June 21–23, 2000 Proceedings 1*. Springer, 1–15.

[17] Keval Doshi, Shatha Abudalou, and Yasin Yilmaz. 2022. Reward Once, Penalize Once: Rectifying Time Series Anomaly Detection. In *2022 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.

[18] Chenguang Fang and Chen Wang. 2020. Time series data imputation: A survey on deep learning approaches. *arXiv preprint arXiv:2011.11347* (2020).

[19] Supriyo Ghosh, Manish Shetty, Chetan Bansal, and Suman Nath. 2022. How to fight production incidents? an empirical study on a large-scale cloud service. In *Proceedings of the 13th Symposium on Cloud Computing*. 126–141.

[20] Markus Goldstein and Andreas Dengel. 2012. Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm. *KI-2012: poster and demo track* 1 (2012), 59–63.

[21] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

[22] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. 2021. Transformer in transformer. *Advances in Neural Information Processing Systems* 34 (2021), 15908–15919.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[24] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems* 33 (2020), 6840–6851.

[25] Jordan Hochenbaum, Owen S Vallis, and Arun Kejariwal. 2017. Automatic anomaly detection in the cloud via statistical learning. *arXiv preprint arXiv:1704.07706* (2017).

[26] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[27] Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. 2018. Detecting spacecraft anomalies using LSTMs and non-parametric dynamic thresholding. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 387–395.

[28] Olumuyiwa Ibidunmoye, Francisco Hernández-Rodriguez, and Erik Elmroth. 2015. Performance anomaly detection and bottleneck identification. *ACM Computing Surveys (CSUR)* 48, 1 (2015), 1–35.

[29] Vincent Jacob, Fei Song, Arnaud Stiegler, Bijan Rad, Yanlei Diao, and Nesime Tatbul. 2021. Exathlon: a benchmark for explainable anomaly detection over time series. *Proceedings of the VLDB Endowment* 14, 11 (2021), 2613–2626.

[30] Sheo Yon Jhin, Jaehoon Lee, Minju Jo, Seungji Kook, Jinsung Jeon, Jihyeon Hyeong, Jayoung Kim, and Noseong Park. 2022. Exit: Extrapolation and interpolation-based neural controlled differential equations for time-series classification and forecasting. In *Proceedings of the ACM Web Conference 2022*. 3102–3112.

[31] Pengxiang Jin, Shenglin Zhang, Minghua Ma, Haozhe Li, Yu Kang, Liqun Li, Yudong Liu, Bo Qiao, Chaoyun Zhang, Pu Zhao, et al. 2023. Assess and Summarize: Improve Outage Understanding with Large Language Models. In *Proceedings of the Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE)*.

[32] Chunggyeom Kim, Jinhyuk Lee, Raehyun Kim, Youngbin Park, and Jaewoo Kang. 2018. DeepNAP: Deep neural anomaly pre-detection in a semiconductor fab. *Information Sciences* 457 (2018), 1–11.

[33] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[34] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. [n. d.]. DiffWave: A Versatile Diffusion Model for Audio Synthesis. In *International Conference on Learning Representations*.

[35] Mathieu Lepot, Jean-Baptiste Aubin, and François HLR Clemens. 2017. Interpolation in time series: An introductive overview of existing methods, their performance criteria and uncertainty assessment. *Water* 9, 10 (2017), 796.

[36] Dan Li, Dacheng Chen, Baihong Jin, Lei Shi, Jonathan Goh, and See-Kiong Ng. 2019. MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks. In *Artificial Neural Networks and Machine Learning–ICANN 2019: Text and Time Series: 28th International Conference on Artificial Neural Networks, Munich, Germany, September 17–19, 2019, Proceedings, Part IV*. Springer, 703–716.

[37] Hongzu Li and Pierre Boulanger. 2020. A survey of heart anomaly detection using ambulatory Electrocardiogram (ECG). *Sensors* 20, 5 (2020), 1461.

[38] Zhihan Li, Youjian Zhao, Jiaqi Han, Ya Su, Rui Jiao, Xidao Wen, and Dan Pei. 2021. Multivariate time series anomaly detection and interpretation using hierarchical inter-metric and temporal embedding. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*. 3220–3230.

[39] Haksoo Lim, Minjung Kim, Sewon Park, and Noseong Park. 2023. Regular Time-series Generation using SGM. *arXiv preprint arXiv:2301.08518* (2023).

[40] Lequan Lin, Zhengkun Li, Ruikun Li, Xuliang Li, and Junbin Gao. 2023. Diffusion Models for Time Series Applications: A Survey. *arXiv preprint arXiv:2305.00624* (2023).

[41] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. 2008. Isolation forest. In *2008 eighth ieee international conference on data mining*. IEEE, 413–422.

[42] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. 2012. Isolation-based anomaly detection. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 6, 1 (2012), 1–39.

[43] Yue Lu, Renjie Wu, Abdullah Mueen, Maria A Zuluaga, and Eamonn Keogh. 2022. Matrix profile XXIV: scaling time series anomaly detection to trillions of datapoints and ultra-fast arriving data streams. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1173–1182.

[44] Minghua Ma, Yudong Liu, Yuang Tong, Haozhe Li, Pu Zhao, Yong Xu, Hongyu Zhang, Shilin He, Lu Wang, Yingnong Dang, et al. 2022. An empirical investigation of missing data handling in cloud node failure prediction. In *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 1453–1464.

[45] Minghua Ma, Zheng Yin, Shenglin Zhang, Sheng Wang, Christopher Zheng, Xinhao Jiang, Hanwen Hu, Cheng Luo, Yilin Li, Nengjun Qiu, et al. 2020. Diagnosing root causes of intermittent slow queries in cloud databases. *Proceedings of the VLDB Endowment* 13, 8 (2020), 1176–1189.

[46] Minghua Ma, Shenglin Zhang, Junjie Chen, Jim Xu, Haozhe Li, Yongliang Lin, Xiaohui Nie, Bo Zhou, Yong Wang, and Dan Pei. 2021. Jump-Starting Multivariate Time Series Anomaly Detection for Online Service Systems. In *2021 USENIX Annual Technical Conference (USENIX ATC 21)*. 413–426.

[47] Minghua Ma, Shenglin Zhang, Dan Pei, Xin Huang, and Hongwei Dai. 2018. Robust and rapid adaption for concept drift in software system anomaly detection. In *2018 IEEE 29th International Symposium on Software Reliability Engineering (ISSRE)*. IEEE, 13–24.

[48] Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, and Gautam Shroff. 2016. LSTM-based encoder-decoder for multi-sensor anomaly detection. *arXiv preprint arXiv:1607.00148* (2016).

[49] Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, Puneet Agarwal, et al. 2015. Long Short Term Memory Networks for Anomaly Detection in Time Series. In *ESANN*, Vol. 2015. 89.

[50] Aditya P Mathur and Nils Ole Tippenhauer. 2016. SWaT: A water treatment testbed for research and training on ICS security. In *2016 international workshop on cyber-physical systems for smart water networks (CySWater)*. IEEE, 31–36.

[51] Mohsin Munir, Shoaib Ahmed Siddiqui, Andreas Dengel, and Sheraz Ahmed. 2018. DeepAnT: A deep learning approach for unsupervised anomaly detection in time series. *Ieee Access* 7 (2018), 1991–2005.

[52] John Paparrizos, Paul Boniol, Themis Palpanas, Ruey S Tsay, Aaron Elmore, and Michael J Franklin. 2022. Volume Under the Surface: A New Accuracy Evaluation Measure for Time-Series Anomaly Detection. *Proceedings of the VLDB Endowment* 15, 11 (2022), 2774–2787.

[53] John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S Tsay, Themis Palpanas, and Michael J Franklin. 2022. TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection. *Proceedings of the VLDB Endowment* 15, 8 (2022), 1697–1711.

[54] Daehyung Park, Yuuna Hoshi, and Charles C Kemp. 2018. A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder. *IEEE Robotics and Automation Letters* 3, 3 (2018), 1544–1551.

[55] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* 32 (2019).

[56] Walter HL Pinaya, Mark S Graham, Robert Gray, Pedro F Da Costa, Petru-Daniel Tudosiu, Paul Wright, Yee H Mah, Andrew D MacKinnon, James T Teo, Rolf Jager, et al. 2022. Fast unsupervised brain anomaly detection and segmentation with diffusion models. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part VIII*. Springer, 705–714.

[57] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125* (2022).

[58] Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. 2021. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *International Conference on Machine Learning*. PMLR, 8857–8868.

[59] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10684–10695.

[60] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. 2022. Anomaly detection in time series: a comprehensive evaluation. *Proceedings of the VLDB Endowment* 15, 9 (2022), 1779–1797.

[61] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. 2015. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems* 28 (2015).

[62] Satya Narayan Shukla and Benjamin Marlin. 2023. Interpolation-Prediction Networks for Irregularly Sampled Time Series. In *International Conference on Learning Representations*.

[63] Alban Siffer, Pierre-Alain Fouque, Alexandre Termier, and Christine Largouet. 2017. Anomaly detection in streams with extreme value theory. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1067–1075.

[64] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*. PMLR, 2256–2265.

[65] Jiaming Song, Chenlin Meng, and Stefano Ermon. [n. d.]. Denoising Diffusion Implicit Models. In *International Conference on Learning Representations*.

[66] Ya Su, Youjian Zhao, Chenhao Niu, Rong Liu, Wei Sun, and Dan Pei. 2019. Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2828–2837.

[67] Emmanouil Sylligardos, Paul Boniol, John Paparrizos, Panos Trahanias, and Themis Palpanas. 2023. Choose Wisely: An Extensive Evaluation of Model Selection for Anomaly Detection in Time Series. *Proceedings of the VLDB Endowment* 17, 1 (2023).

[68] Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. 2021. CSDI: Conditional score-based diffusion models for probabilistic time series imputation. *Advances in Neural Information Processing Systems* 34 (2021), 24804–24816.

[69] Kai Ming Ting, Zongyou Liu, Hang Zhang, and Ye Zhu. 2022. A new distributional treatment for time series and an anomaly detection investigation. *Proceedings of the VLDB Endowment* 15, 11 (2022), 2321–2333.

[70] Shreshth Tuli, Giuliano Casale, and Nicholas R Jennings. 2022. TranAD: deep transformer networks for anomaly detection in multivariate time series data. *Proceedings of the VLDB Endowment* 15, 6 (2022), 1201–1214.

[71] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[72] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. [n. d.]. Graph Attention Networks. In *International Conference on Learning Representations*.

[73] Lu Wang, Chaoyun Zhang, Ruomeng Ding, Yong Xu, Qihang Chen, Wentao Zou, Qingjun Chen, Meng Zhang, Xuedong Gao, Hao Fan, et al. 2023. Root Cause Analysis for Microservice Systems via Hierarchical Reinforcement Learning from Human Feedback. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 5116–5125.

[74] Phillip Wenig, Sebastian Schmidl, and Thorsten Papenbrock. 2022. TimeEval: a benchmarking toolkit for time series anomaly detection algorithms. *Proceedings of the VLDB Endowment* 15, 12 (2022), 3678–3681.

[75] Julia Wolleb, Florentin Bieder, Robin Sandkühler, and Philippe C Cattin. 2022. Diffusion models for medical anomaly detection. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part VIII*. Springer, 35–45.

[76] Xiaohan Yan, Ken Hsieh, Yasitha Liyanage, Minghua Ma, Murali Chintalapati, Qingwei Lin, Yingnong Dang, and Dongmei Zhang. 2023. Aegis: Attribution of Control Plane Change Impact across Layers and Components for Cloud Systems. In *2023 IEEE/ACM 45th International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*. IEEE, 222–233.

[77] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Yingxia Shao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. 2022. Diffusion models: A comprehensive survey of methods and applications. *arXiv preprint arXiv:2209.00796* (2022).

[78] Zhengran Zeng, Yuqun Zhang, Yong Xu, Minghua Ma, Bo Qiao, Wentao Zou, Qingjun Chen, Meng Zhang, Xu Zhang, Hongyu Zhang, et al. 2023. TraceArk: Towards Actionable Performance Anomaly Alerting for Online Service Systems. In *2023 IEEE/ACM 45th International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*. IEEE, 258–269.

[79] Chaoyun Zhang, Marco Fiore, Iain Murray, and Paul Patras. 2021. CloudLSTM: A recurrent neural model for spatiotemporal point-cloud stream forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 10851–10858.

[80] Chaoyun Zhang, Marco Fiore, and Paul Patras. 2019. Multi-service mobile traffic forecasting via convolutional long short-term memories. In *2019 IEEE International Symposium on Measurements & Networking (M&N)*. IEEE, 1–6.

[81] Chaoyun Zhang, Marco Fiore, Cezary Ziemlicki, and Paul Patras. 2020. Microscope: mobile service traffic decomposition for network slicing as a service. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–14.

[82] Chaoyun Zhang, Xi Ouyang, and Paul Patras. 2017. ZipNet-GAN: Inferring fine-grained mobile traffic patterns via a generative adversarial neural network. In *Proceedings of the 13th International Conference on emerging Networking EXperiments and Technologies*. 363–375.

[83] Chaoyun Zhang and Paul Patras. 2018. Long-term mobile traffic forecasting using deep spatio-temporal neural networks. In *Proceedings of the Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*. 231–240.

[84] Chaoyun Zhang, Paul Patras, and Hamed Haddadi. 2019. Deep learning in mobile and wireless networking: A survey. *IEEE Communications surveys & tutorials* 21, 3 (2019), 2224–2287.

[85] Chuxu Zhang, Dongjin Song, Yuncong Chen, Xinyang Feng, Cristian Lumezanu, Wei Cheng, Jingchao Ni, Bo Zong, Haifeng Chen, and Nitesh V Chawla. 2019. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 1409–1416.

[86] Hui Zhang, Zheng Wang, Zuxuan Wu, and Yu-Gang Jiang. 2023. DiffusionAD: Denoising Diffusion for Anomaly Detection. *arXiv preprint arXiv:2303.08730* (2023).

[87] Chenyu Zhao, Minghua Ma, Zhenyu Zhong, Shenglin Zhang, Zhiyuan Tan, Xiao Xiong, LuLu Yu, Jiayi Feng, Yongqian Sun, Yuzhi Zhang, et al. 2023. Robust Multimodal Failure Detection for Microservice Systems. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

[88] Hang Zhao, Yujing Wang, Juanyong Duan, Congrui Huang, Defu Cao, Yunhai Tong, Bixiong Xu, Jing Bai, Jie Tong, and Qi Zhang. 2020. Multivariate time-series anomaly detection via graph attention network. In *2020 IEEE International Conference on Data Mining (ICDM)*. IEEE, 841–850.

[89] Bin Zhou, Shenghua Liu, Bryan Hooi, Xueqi Cheng, and Jing Ye. 2019. BeatGAN: Anomalous Rhythm Detection using Adversarially Generated Time Series.. In *IJCAI*, Vol. 2019. 4433–4439.

[90] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 11106–11115.