Received: 8 December 2021 | Accepted: 13 February 2022

DOI: 10.1002/int.22864

RESEARCH ARTICLE

WILEY

HydroFlow: Towards probabilistic electricity demand prediction using variational autoregressive models and normalizing flows

Fan Zhou[1] | Zhiyuan Wang[1] | Ting Zhong[1] | Goce Trajcevski[2] | Ashfaq Khokhar[2]

[1]School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan, China

[2]Department of Electrical and Computer Engineering, Iowa State University, Ames, Iowa, USA

Correspondence
Ting Zhong, University of Electronic Science and Technology of China, 610054 Chengdu, Sichuan, China.
Email: zhongting@uestc.edu.cn

Funding information
National Natural Science Foundation of China, Grant/Award Numbers: 62072077, 62176043; National Science Foundation SWIFT, Grant/Award Number: 2030249

Abstract

We present HydroFlow, a novel deep generative model for predicting the electricity generation demand of large-scale hydropower stations. HydroFlow uses a latent stochastic recurrent neural network to capture the dependencies in the multivariate time series. It not only utilizes the hidden state of the neural network, but also considers the uncertainty of variables related to natural and social factors. We also introduce an end-to-end approach based on generative flows to approximate the posterior distribution of time series with exact likelihoods. Our model is powerful as adding stochasticity to different factors (e.g., reservoir capacity and water-flow measurements) and thus overcomes the expressiveness limitations of deterministic prediction methods. It also enables trainable latent transformations that can improve the model interpretability. We evaluate HydroFlow on the data collected from the hydropower stations of a large-scale hydropower development company. Experimental results show that our model significantly outperforms the state-of-the-art baseline methods while providing explainable results.

KEYWORDS

generative model, hydropower, intelligent system, multivariate time series, renewable energy

1 | INTRODUCTION

The volume of generated renewable and clean energy such as photovoltaic, wind, and water power has increased rapidly over the last decade.[1,2] Hydropower, which is derived from river basins that provide water slopes to drive turbine generators, is the most significant renewable resources with the lowest carbon footprint per MWh.[3–5] For example, it hit a record 4306 terawatt-hours (TWh) in 2019—the single most enormous contribution from a renewable energy source in history.[*] Moreover, the global weighted average cost of electricity from hydropower projects in 2019 was US $0.047/kWh, making it the lowest-cost source of electricity in many markets. The Covid-19 pandemic has also highlighted hydropower's resilience and critical role in delivering clean, flexible, reliable, and affordable energy, especially in times of crisis, because the plant operations have been less affected due to the degree of automation with modern Industrial Internet of Things (IIoT) facilities and intelligent system.[6–12]

Generally, the grid controls and dispatches the electricity generation demand to hydropower stations ahead of days. However, a hydropower station's power generation is also affected by many dynamic factors, such as river/reservoir inflows, seasonality, abrupt needs, and gross industrial production, the self-usage of electricity, price, and so forth. In addition to hydropower generation, the dams should also regulate the downstream water and maintain water ecology. For example, the inflow in summer cannot be fully exploited to generate electricity because the dam has to release surplus water to guarantee a safe water level and reduce the probability of floods and landslides.[13] Similarly, the hydropower generation in winter is low due to the minimum water level required to ensure the river ecology and the responsibility for irrigation of surrounding farmland.

Since the electricity cannot be stored in large quantities, the power generation of the hydropower enterprises must follow certain dispatching policies.[14] Therefore, large-scale hydropower companies necessitate modeling[15] and forecasting and recommending the power demand, especially in the case of cascaded hydroelectricity plants[16,17]—an example of which is shown in Figure 1.

Within the broader context of Machine Learning (ML) for Internet of Things (IoT),[18] the existing research on renewable energy forecasting focuses on either the power consumption prediction in electricity grid[19–21] or the intermittent resources (e.g., wind, inflow, and solar power) forecasts.[22–30] For example, short-term load prediction has been considered in Reference [21], which models various coevolving time series using neural networks. In Reference [31], the authors

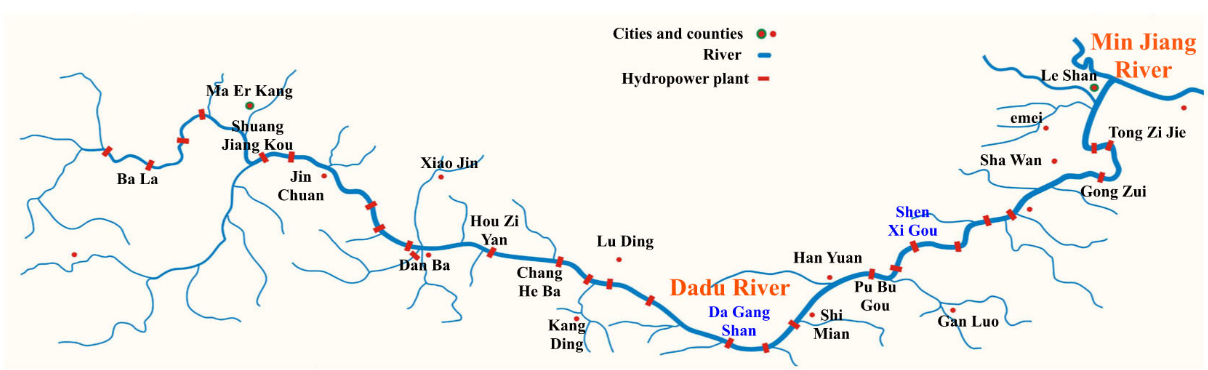


FIGURE 1 A map of the hydropower plant distribution along Dadu River—a tributary of the Yangtze River in China [Color figure can be viewed at wileyonlinelibrary.com]

use optimal reconciliation regression to model the electricity consumption patterns. The renewable generation of wind and photovoltaic (PV) power has been extensively studied in recent years,[22–28,32] which mainly work on various aspects of power probabilistic forecasts,[22] including multiobjective decision-making,[33] dynamic feature engineering,[27] the strategies of balancing various renewable resources,[32] and so forth.

Despite significant progress in modeling and forecasting electricity consumption and the intermittent resources,[34,35] very few efforts have focused on fine-grained hydropower demand prediction. However, the forecasting of hydropower energy demand to be dispatched in the future would improve the support not only of the more economic dam operation, but also river ecology.[36] With the increasing complexity of the power system with the intermittent renewable resources, hydropower demand forecasting may benefit the balance between profit maximization, equipment maintenance, and ecology preservation in the downstream river and dams.[37] For example, hydropower stations could improve the water storage as much as possible within the safe range to maximize the utilization ratio of water resources. Correspondingly, the enterprises could arrange cycle scheduling of water turbines for regular check and repairing of components if less power generation demand is dispatched. Besides, knowing future power generation demand can help hydropower systems make decisions on drainage in the nonflood period, which will better take the rule for river ecology, such as biodiversity conservation and agriculture irrigation.[38,39] Thus, accurate hydropower demand prediction will benefit in enhancing river ecology and developing a power generation plan to reasonably reduce the water loss.

There is a growing need to accurately and efficiently model short-term renewable generation demand uncertainties and minimize the power system's decision risks. However, as common in IoT systems—there are various uncertainty factors stemming from heterogeneous sources[40,41] which pose additional challenges on modeling and forecasting hydroelectric power generation demand. In principle, hydropower data can be large-capacity and real-time data, and its representation is usually composed of multiple time series.[42] In addition to the infrastructure-related factors such as grid demand and grid-connected load, specific natural and social factors also affect hydropower demand prediction, for example, the different on-grid prices of various stations in a company.

In this paper, we propose a novel approach called *HydroFlow* to overcome these challenges. HydroFlow is an innovative probabilistic learning and predicting framework that models the stochastic multivariate time series—which is collected from various IIoT sensors and real-time intelligent platform—and produces latent variables' temporal dependencies. In particular, we design a novel variational latent recurrent neural networks (VL-RNNs) model for learning multivariate hydropower time-series data. It is a latent generative approach preserving both temporal relationships along the time dimension and stochastic processes of the power generation demand. This property allows us to estimate the model uncertainty that is important to support the downstream decision-making regarding dam operation. Our method increases the expressive capacity of the deterministic recurrent neural networks (RNNs) with efficient variational inference. It also enables the estimates of model uncertainty—which is essential to support the downstream decision-making regarding dam operation. To alleviate the inductive bias raised by poor posterior inference, we use a normalizing flow[43]—a Bayesian method transforming a simple prior distribution into the desired target distribution through an invertible neural network—to express the likelihood over the hydropower time series and estimate the density more accurately and flexibly. Furthermore, we consider additional features, including natural (e.g., inflow, temperature, and abandoned water) and social factors (e.g., electricity price) and present an efficient external knowledge fusion network to improve the hydropower generation demand forecast.

In conclusion, the main contributions of this study are fourfold:

- To our knowledge, this is the first study on predicting the hydropower generation demand of large-scale hydroplants. We analyze various observations from a perspective of deep multivariate time-series learning and probabilistic modeling. Our approach opens the door of applying deep generative models on mining industrial power data and may foster enthusiasm for efficient and flexible renewable energy use.
- Moreover, we propose a novel framework to learn the evolution of latent representation with a new designed stochastic RNN and the generative normalizing flow, preserving accurate and robust representations of multivariate time series corresponding to industrial hydropower generation and demand data.
- Finally, we conduct extensive experiments on real-world data sets collected from large-scale hydropower stations. The experimental results show that HydroFlow improves prediction performance over state-of-the-art time-series forecasting approaches while self-explaining the model behavior.
- We successfully deploy our hydropower demand forecasting model on the Hydropower Data Analysis and Data mining platform (HydroDAD) in a large-scale hydropower generation company. The model is continuously optimized with new hydropower data, which combines hydropower demand prediction with various IIoT monitoring data to help make decisions.

2 | RELATED WORK

Modeling multivariate time series is a subject that has attracted researchers from a diverse range of fields. As a specific industrial time series, electricity generation/consumption forecasting has received considerable attention due to its practical value in industry, business, society, and environments.[20,21,31,44] Zhou et al. study short-term load forecasting to improve the reliability of the power grid.[65] The proposed method, called NeuCast, models various loads as coevolving time series using neural networks and captures seasonality of the electricity usage using the external factors, including temperature and precipitation. Pang et al. propose a hierarchical time-series model to forecast the electricity demand through exploring electricity consumption patterns.[31] The basic model is based on the optimal reconciliation regression and can therefore deal with the hierarchical power usage prediction. StreamCast[20] forecasts the power consumption of a location multiple days ahead. The authors propose a temporal linear load model, which characterizes the grid load by modeling the voltage sensitivities.

With the increasing penetration of stochastic renewable generation, recent studies try to model and predict possible future renewable electricity.[22–27,32] Uncertainty modeling techniques for power probabilistic forecasts were studied in Reference [23], where a Gaussian copula approach was proposed to model the short-term wind power. Complementary, data reduction was used in Reference [22] to generate representative power penetration and load levels and solve the multiobjective decision-making problem based on information entropy and the analytic hierarchy process. Forecasting the PV output power at several time horizons using several ML techniques such as least square support vector machine and neural networks was studied in Reference [33]. An efficient dynamic power generation prediction method based on

Gibbs sampling was presented in Reference [27]. The generated renewable power scenarios are drawn from the joint distribution, capturing multiple renewable power plants' statistical features. Keeping renewable electricity production in balance with the actual demand has been studied in Reference [32], which introduced a complementary imbalance forecasting framework that can help the system determine future intermittent renewable production.

As for hydropower prediction research, prior works mainly concentrate on the inflows forecasting and power generation forecasting. For example, Zhong et al.[45] studied the influence of climate change on future hydropower generation, which is limited to natural factors without considering the social and commercial impact on power generation. Learnt hydropower time series using RNN was proposed in Reference [46], which only considers the sequential dependencies among power generation data. Combining social and natural factors to predict reservoir inflows and power generation's evolution using RNN was tackled by References [30,46]. However, the proposed models are built based on solving ordinary differential equations (ODEs), making the model hard to train due to the inaccurate and unstable numerical ODE solvers.

In addition, some recent works studied the issues w.r.t. hydropower generation demand, which, however, mainly focus on long-term and real-time hydropower generation dispatch optimizations from the grid's perspective. For example, Zhang et al.[47] analyze the characteristics of hydropower to determine the optimal working position and capacity of each power plant in the daily load curve. Lion swarm optimization algorithm to optimize the dispatch of cascade hydropower stations was presented in Reference [48]. The interactions between power generation and degree of ecological flow satisfaction under different operation modes to enhance the stability of river ecosystems was considered in Reference [49].

Notwithstanding the progress enabled by the above works, we note that there are few works paying attention to predicting short-term dispatches of hydropower generation demand, which has significant benefit for scheduling electricity generation and operating dams from the hydropower station's perspective.

Recent advances in deep neural networks have inspired many works that improve time-series prediction with various deep learning techniques,[50] among which autoregressive models such as RNN and its variants[51] are the main building blocks. However, RNNs are deterministic models that cannot efficiently capture the uncertainty associated with multivariate time series, for example, the inflow/outflow of reservoir and rainfall measurements for hydropower dams.

Nowadays, deep generative models are applied to time-series modeling and forecasting tasks and achieve important advances.[52,53] Thus, to overcome the above issues, we introduce generative methods by proposing a VL-RNN model to capture the stochasticity of power generation and enhance the robustness of learning multivariate time-series data. Besides, normalizing flow[43] is explored to improve posterior inference through an invertible density transformation. Specifically, we propose a continuous variational inference approach based on free-form Jacobian of reversible dynamics[54] to fit the dynamic systems of hydropower, where the latent variables can be composed with each other to approximate more complex distributions beyond mixture Gaussian.

In addition to modeling the electrical power as in previous neural approaches, we also distill the knowledge from external factors, the dependencies between latent variables and, most importantly, the uncertainty of hidden states. In this spirit, we take the initiatives to model industrial time series and external data fusion in a profoundly generative learning way. Rather than merely improving prediction results, our model is more robust on data representation and can interpret the learned latent space.

3 | PRELIMINARIES

We consider a portfolio of multivariate hydropower observations $\mathbf{X}$. Each observation $\mathbf{x}_t$ at time t includes electricity $\mathbf{v}_t$ and water flow $\mathbf{w}_t$. The power, defined as $\mathbf{v}_t = \{v_t^1, v_t^2, v_t^3\}$, includes total power demand, power generation, grid-connected power and auxiliary power. The water flow, defined as $\mathbf{w}_t = \{w_t^1, w_t^2, w_t^3\}$, consists of water inflow, water outflow, and generation flow. Therefore, each observation $\mathbf{x}_t \in \mathbb{R}^P$ at time t is a vector with a dimension of $P = 7$.

Several external factors are considered including: (1) the temporal factor τ of holiday effect, which consists of three categories, that is, HourOfDay, DayOfWeek, and WeekdayOrNot; (2) natural factors including water flow and temperature, which are uncertain due to inaccurate measurement and prediction; and (3) the electricity price of individual plants that can be considered as a commercial/social factor. We use $\mathbf{e}$ to represent these external factors. *Hydropower generation forecasting.* Let $\mathbf{X}_{t-1} = \{\mathbf{x}_{t-N}, \mathbf{x}_{t-N+1}, ..., \mathbf{x}_{t-1}\} \in \mathbb{R}^{N \times P}$ be the N historical observations, and $\mathbf{e}_t$ be the external factors at time t, the task is to learn a model $\mathcal{F}$ from historical data $\mathbf{X}_{t-1}$ to predict the future power $\hat{\mathbf{v}}_t$:

$$\hat{\mathbf{v}}_t = \mathcal{F}(\mathbf{X}_{t-1} | \mathbf{e}_t; \Omega), \tag{1}$$

where Ω are the set of learnable parameters.

4 | METHODOLOGY

We now provide a formal definition of the problem and present the details of our methodology.

4.1 | Problem definition

We consider a portfolio of multivariate hydropower observations $\mathbf{X}$. At each time t, we have the observation $\mathbf{x}_t$ consisting of electricity $\mathbf{v}_t$ and water flow $\mathbf{w}_t$ measurement values. The generated power data are composed of four different time series, that is, $\mathbf{v}_t = \{v_t^1, v_t^2, v_t^3, v_t^4\}$, which are the: *total power demand*, *power generation*, *grid-connected power*, and *auxiliary power*, respectively. The water flow, denoted as $\mathbf{w}_t = \{w_t^1, w_t^2, w_t^3\}$, consists of water inflow, water outflow, and generation flow. That is, each observation $\mathbf{x}_t \in \mathbb{R}^P$ at time t is a vector with a dimension of $P = 7$.

Several external factors are considered including: (1) the temporal factor τ of holiday effect, which consists of three categories, that is, HourOfDay, DayOfWeek, and WeekdayOrNot; (2) natural factors including water flow and temperature, which are uncertain due to inaccurate measurement and prediction; and (3) the electricity price of individual plants that can be considered as a commercial/social factor. We use $\mathbf{e}$ to represent these external factors. In this study, we study the following problem: *Hydropower generation demand forecasting.* Let $\mathbf{X}_{t-1} = \{\mathbf{x}_{t-N}, \mathbf{x}_{t-N+1}, ..., \mathbf{x}_{t-1}\} \in \mathbb{R}^{N \times P}$ be the N historical observations, and $\mathbf{e}_t$ be the external factors at time t, the task is to learn a model $\mathcal{F}$ from historical data $\mathbf{X}_{t-1}$ to predict the future power $\hat{\mathbf{v}}_t$:

$$\hat{\mathbf{v}}_t = \mathcal{F}(\mathbf{X}_{t-1} | \mathbf{e}_t; \Theta), \tag{2}$$

where Θ is the set of learnable parameters.

4.2 | System overview

Our generative model-based hydropower forecasting method enables probabilistic time-series learning by incorporating stochastic variables into RNNs. In particular, our HydroFlow model consists of three core components: a stochastic time-series encoder–decoder network, the external attributes embedding network, and the feature fusion and prediction network.

Figure 2 illustrates the overall architecture of HydroFlow. First, HydroFlow learns the latent representations of multivariate time-series data (cf. the left part of Figure 2) in an end-to-end manner by introducing a stochastic RNN method, called VL-RNN. VL-RNN captures long and short temporal dependencies among sequential observations with the gated mechanism of gated recurrent unit (GRU) while embedding the stochastic variables to enable variational inference. Rather than merely relying on predefined diagonal Gaussian as previous autoregressive variational models, we borrow the idea from generative flows to better estimate latent variables' distribution. Next, our model combines external knowledge (e.g., sale price, temperature, and holiday effect) using an external information embedding network (EIEN). Finally, the multiple representations learned from both time-series data and categorical data—including external knowledge $\mathbf{a}_t$ and stochastic latent variables $\mathbf{z}_t$—are fed into a fusion network to make the prediction. We will delve deeper into the three components in the subsequent sections.

4.3 | Stochastic encoder–decoder network

RNNs have proven to be effective for time-series modeling and learning. RNN autoencoders are usually used for learning latent representations of time series, which can provide useful information for downstream tasks, including future value prediction, outlier detection, and classification.[50]

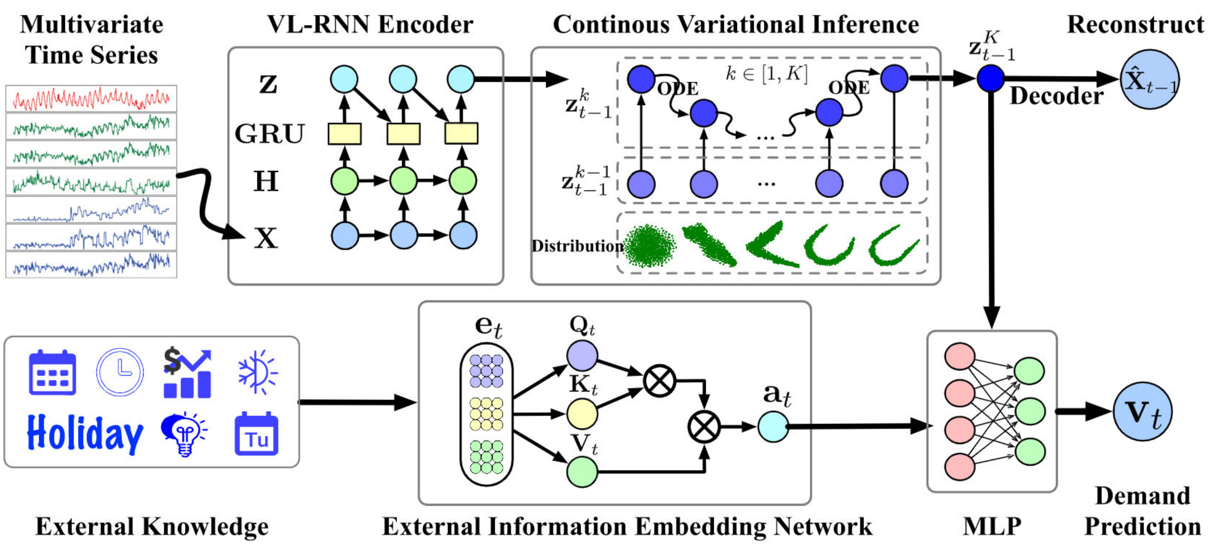


FIGURE 2 An overview of our model for hydroelectricity generation demand prediction. VL-RNN and continuous variational inference: a latent RNN model for learning stochastic dependencies and variational uncertainty inference. External knowledge includes temperature, time, weekday, electricity sale price, and so forth. We design an external information embedding network to analyze them. GRU, gated recurrent unit; MLP, multiple layer perceptron; ODE, ordinary differential equation; RNN, recurrent neural network; VL-RNN, variational latent recurrent neural network [Color figure can be viewed at wileyonlinelibrary.com]

However, RNNs are deterministic models that cannot handle the stochasticity inherent in time series, which causes the existing methods to fail to fully exploit latent temporal dependencies between pairs of variables. Here we present a novel hydropower inference network that explicitly modifies the RNN to consider the uncertainty of RNN hidden states. In addition, we equip our model with the normalizing flow technique that allows us to approximate more complex posterior, beyond diagonal Gaussian, with flexible density transformations.

4.3.1 | Variational latent RNN

Our VL-RNN behaves as the encoder and decoder networks in HydroFlow. It takes multivariate time series as input and restores the input as the objective. The main idea of VL-RNN is illustrated in the bottom left of Figure 2. Typical RNN-based variational autoencoder (VAE) models[55] usually parameterize the latent variables $\mathbf{z}$ on the last hidden state $\mathbf{h}$ of RNN, which may not well encode the temporal dependencies into latent codes $\mathbf{z}$. In contrast, VL-RNN specifically considers the inherent relationships among latent variables $\mathbf{Z}_{t-1}$ up to the previous step, while maintaining the hidden states of RNN cells. Therefore, our model learns the representations conditioned not only on the time-series observations but also on some latent random variables.

Figure 3 illustrates the workflow of VL-RNN in one time step. Specifically, we use GRU as the basic RNN cells. At current time step $t-1$, the input of the GRU is the observed time-series value $\mathbf{x}_{t-1}$ and the mean and variance of previous hidden states $(\boldsymbol{\mu}^{\mathbf{h}}_{t-2}, \boldsymbol{\sigma}^{\mathbf{h}}_{t-2})$. The GRU cell first outputs the mean and variance of the observation $(\boldsymbol{\mu}^{\mathbf{h}}_{t-1}, \boldsymbol{\sigma}^{\mathbf{h}}_{t-1})$, followed by the hidden state $\mathbf{h}_{t-1}$. Subsequently, another GRU cell would take $(\boldsymbol{\mu}^{\mathbf{z}}_{t-2}, \boldsymbol{\sigma}^{\mathbf{z}}_{t-2})$ (the stochasticity of observations obtained at previous time step $t-2$) and current hidden state $\mathbf{h}_{t-1}$ as inputs to generate the latent variables $(\boldsymbol{\mu}^{\mathbf{z}}_{t-1}, \boldsymbol{\sigma}^{\mathbf{z}}_{t-1})$ at $t-1$, which are considered as the mean and variance of a certain distribution. Then, we use the *reparameterization trick*[56] to compute the latent variables: $\mathbf{z}_{t-1} = \boldsymbol{\mu}^{\mathbf{z}}_{t-1} + \boldsymbol{\sigma}^{\mathbf{z}}_{t-1} * \epsilon$, where ϵ are samples from a standard Gaussian $\epsilon \sim \mathcal{N}(0, I)$.

Algorithm 1: The pseudo-code of VL-RNN.

Input: multivariate hydropower data: $\mathbf{X}_{t-1}$.
Output: latent variables $\mathbf{Z}_{t-1}$ and hidden states $\mathbf{H}_{t-1}$.

Initialize $\mu^{\mathbf{h}}_{t-N-1} = \mathbf{0}, \sigma^{\mathbf{h}}_{t-N-1} = \mathbf{0}, \mu^{\mathbf{z}}_{t-N-1} = \mathbf{0}, \sigma^{\mathbf{z}}_{t-N-1} = I$;
foreach i *in* $[t-N, \cdots, t-1]$ **do**
    $\mu^{\mathbf{h}}_i, \sigma^{\mathbf{h}}_i = \text{GRUCell}\left(\mu^{\mathbf{h}}_{i-1}, \sigma^{\mathbf{h}}_{i-1}, \mathbf{x}_i\right)$;
    $\mathbf{h}_i = \mu^{\mathbf{h}}_i$;
    $\mu^{\mathbf{z}}_i, \sigma^{\mathbf{z}}_i = \text{GRUCell}\left(\mu^{\mathbf{z}}_{i-1}, \sigma^{\mathbf{z}}_{i-1}, \mathbf{h}_i\right)$;
    $\mathbf{z}_i = \text{Reparameterize}\left(\mu^{\mathbf{z}}_i, \sigma^{\mathbf{z}}_i\right)$;
    Save hidden state $\mathbf{h}_i$ and latent variable $\mathbf{z}_i$;
end
Concatenate all hidden states $\mathbf{h}_{t-N:t-1}$ to $\mathbf{H}_{t-1}$;
Concatenate all latent variables $\mathbf{z}_{t-N:t-1}$ to $\mathbf{Z}_{t-1}$;
return $\mathbf{Z}_{t-1}$ and $\mathbf{H}_{t-1}$.

VL-RNN realizes the variational latent representation learning based on both latent variables and hidden states of observations. Since the latent variables in VL-RNN are also

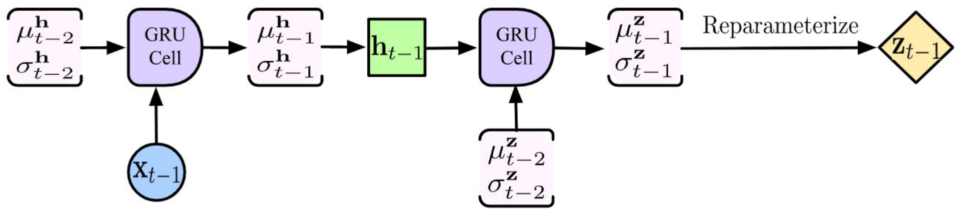


FIGURE 3 Zoom-in of one-time step computation in VL-RNN. GRU, gated recurrent unit; VL-RNN, variational latent recurrent neural network [Color figure can be viewed at wileyonlinelibrary.com]

conditioned on hidden states, it can be considered as a hierarchical RNN architecture that summarizes high-level time-series patterns and is expected to capture statistical dependencies among observations. Once the latent variables $\mathbf{z}_{t-1}$ are available, we can directly sample $\mathbf{z}_{t-1}$ from the learned posterior distribution $q_\phi(\mathbf{z}_{t-1}|\mathbf{X}_{t-1}, \mathbf{z}_{t-2})$ (ϕ denotes the parameters), and concatenate all hidden states $\mathbf{h}_{t-N:t-1}$ and all latent variables $\mathbf{z}_{t-N:t-1}$ for downstream high-level feature fusion. Algorithm 1 summarizes the computations in VL-RNN.

4.3.2 | Inference with normalizing flow

The majority of RNN-based autoencoders model their variational distributions $q_\phi(\mathbf{z}_{t-1}|\mathbf{X}_{t-1}, \mathbf{z}_{t-2})$ as a unimodal distribution via isotropic Gaussians with diagonal covariance. This approach is computationally convenient since it permits a closed-form solution for computing the Kullback–Leibler (KL) term and facilitates end-to-end gradient-based optimization via the reparametrization trick. However, it may pose strong regulations on the latent variables from VL-RNN, which is usually non-Gaussian.

To remedy this issue, we propose to employ more flexible posterior distributions in our HydroFlow model. Inspired by the normalizing flows,[43] we present a way of computing more flexible posterior to approximate the variational distribution. The basic idea is to draw a simple (e.g., Gaussian) base density $p(z_t^0)$, and apply K invertible deterministic transformation functions β (also known as *flows*) on the base density:

$$\mathbf{z}_t^K = \beta^K(\cdots \beta^2(\beta^1(\mathbf{z}_t^0))), \tag{3}$$

whose probability distribution $p(\mathbf{z}_t^K)$ can, theoretically, approximate *any* complete posterior.[43]

There are many forms of normalizing flows that have been proposed in recent years, such as linear flows, radial and planar flows, coupling flows, autoregressive flows, residual flows, and continuous normalizing flows (see Reference [57] for a comprehensive review). In our HydroFlow, we propose a continuous-form normalizing flow based on free-form Jacobian of reversible dynamics[54] as the transformation module. Its calculation uses relatively cheap trace operations and, as a result, is able to achieve more flexible estimation with unrestricted transformation functions. For the transformation functions in Equation (3), the change of variables is as

$$\mathbf{z}_t^{k+1} = \beta^k\left(\mathbf{z}_t^k\right) = \mathbf{z}_t^k + \int_k^{k+1} f\left(\mathbf{z}_t^{k'}, k'; \theta\right) dk', \tag{4}$$

$$\frac{\partial \log p\left(\mathbf{z}_t^k\right)}{\partial k} = -\operatorname{Tr}\left(\frac{\partial f}{\partial \mathbf{z}_t^k}\right), \tag{5}$$

$$\log p\left(\mathbf{z}_t^{k+1}\right) = \log p\left(\mathbf{z}_t^k\right) - \int_k^{k+1} \operatorname{Tr}\left(\frac{\partial f}{\partial \mathbf{z}_t^{k'}}\right) dk', \tag{6}$$

where $k \in K$ is the index of transformation layers and $\operatorname{Tr}(\cdot)$ is the trace function of the matrix. Equation (5) is also known as the instantaneous change of variables formula.[58] $f(\mathbf{z}_t^{k}{}', k'; \theta) = d\mathbf{z}_t^{k}{}'/dk'$, namely, the ODE solver, is a differential equation describing the continuous transformation of $\mathbf{z}_t^{k}{}'$ parameterized by θ, and is estimated by a neural network with fully connected layers. In general, computing the trace in Equation (5) incurs a square of matrix dimension complexity. Here, for efficiency, we introduce Hutchinson's trace estimator[59] to estimate this term by taking a double product of that matrix with a noise vector ϵ:

$$\operatorname{Tr}\left(\frac{\partial f}{\partial \mathbf{z}_t^k}\right) = \epsilon^{\mathrm{T}} \frac{\partial f}{\partial \mathbf{z}_t^k} \epsilon, \quad \epsilon \sim \mathcal{N}(0, I), \tag{7}$$

which allows us to obtain the unbiased trace estimation with a linear time complexity. Combining Equations (3) and (7), we can encourage the model to learn the probabilistic distribution with more flexible and accurate form in the latent space that can better reflect the characteristics and trends of hydrological time-series patterns.

For the observations $\mathbf{X}_t$ before current time t, the latent variables obtained by VL-RNN are denoted as $\mathbf{z}_t$ through seeking parameters θ that maximize the marginal of latent variables:

$$\log p_\theta(\mathbf{X}_t) = \mathcal{L}(\theta, \phi) - \mathrm{KL}\left[q_\phi\left(\mathbf{z}_t^K | \mathbf{X}_t\right) \parallel p_\theta\left(\mathbf{z}_t^K\right)\right], \tag{8}$$

where the first term $\mathcal{L}(\theta, \phi)$ refers to the evidence lower bound (ELBO), and $\mathrm{KL}[\cdot|\cdot]$ is the KL divergence. Following References [56,60], we use amortized variational approximation $q_\phi(\mathbf{z}_\mathbf{t}^\mathbf{K} | \mathbf{X}_\mathbf{t}) \approx p_\theta(\mathbf{z}_t^K)$ to minimize the KL divergence, which equals to maximize the ELBO $\mathcal{L}(\theta, \phi)$ as

$$\begin{aligned}
\mathcal{L}(\theta, \phi) &= \mathbb{E}_{q_\phi} \log\left[\frac{p_\theta\left(\mathbf{X}_t, \mathbf{z}_t^K\right)}{q_\phi\left(\mathbf{z}_t^K \mid \mathbf{X}_t\right)}\right] \\
&= \mathbb{E}_{q_\phi} \log\left[p_\theta\left(\mathbf{X}_t \mid \mathbf{z}_t^K\right) p_\theta\left(\mathbf{z}_t^K\right)\right] - \mathbb{E}_{q_\phi} \log\left[q_\phi\left(\mathbf{z}_t^K \mid \mathbf{X}_t\right)\right] \\
&= \mathbb{E}_{q_\phi} \log\left[p_\theta\left(\mathbf{X}_t \mid \mathbf{z}_t^K\right)\right] + \mathbb{E}_{q_\phi} \log\left[p_\theta\left(\mathbf{z}_t^K\right)\right] \\
&\quad + \mathbb{E}_{q_\phi}\left[\sum_{k=1}^{K} \int_k^{k+1} \epsilon^{\mathrm{T}} \frac{\partial f}{\partial \mathbf{z}_t^k} \epsilon \, dk'\right] - \mathbb{E}_{q_\phi} \log\left[q_\phi\left(\mathbf{z}_t^0 \mid \mathbf{X}_t\right)\right], \quad \epsilon \sim \mathcal{N}(0, I),
\end{aligned} \tag{9}$$

which is improved when the model learns a posterior distribution of latent variables that minimizes the reconstruction loss (the first term). The second term can be directly calculated by Equation (4). The variational prior $q_\phi(\mathbf{z}_t^0)$ is an isometric Gaussian. The last term is the result of applying Equation (6).

In the decoder, we feed $\mathbf{Z}_{t-1}$ as the input to reconstruct the hydropower data $\mathbf{X}_{t-1}$:

$$\hat{\mathbf{X}}_{t-1} = \text{VL-RNN}\left(\mathbf{Z}_{t-1}, \mathbf{z}_{t-1}^{K}\right), \tag{10}$$

where $\hat{\mathbf{X}}_{t-1}$ is the reconstruction data. By applying the above invertible transformations, our model may attain a more appropriate variational posterior distribution $q_\phi(\mathbf{z}_{t-1}^{K}|\mathbf{X}_{t-1})$ that is more instructive for future time step prediction, as will be demonstrated in our experiments.

4.4 | External knowledge embedding

As mentioned earlier, power generation demand is closely related to power consumption and other highly seasonal and periodical values. Moreover, water flow is always associated with hydropower demand or water-level regulation—reflecting the responsibility of hydropower dams to prevent flood and protect water ecology. In addition, socioeconomic attributes such as electricity prices may further potentially affect hydroelectric power generation demand, that is, the higher the electricity price, the higher the profitability of hydroelectric power systems.

To incorporate these aspects of knowledge, we design an external information extraction network, as shown in the bottom of Figure 2. Specifically, we embed categorical features (e.g., HourofDay and DayofWeek) as one-hot vectors and feed all external attributes to a self-attention unit,[61] where the close correlation between various factors can be considered. This process at time t can be described as follows:

$$[\mathbf{Q}_t, \mathbf{K}_t, \mathbf{V}_t] = [w_1, w_2, w_3]\mathbf{e}_t + [b_1, b_2, b_3], \tag{11}$$

$$\mathbf{a}_t = \text{Softmax}\left(\mathbf{Q}_t\mathbf{K}_t^{\mathrm{T}}/\sqrt{d_a}\right)\mathbf{V}_t, \tag{12}$$

where $d_a = 128$ is the dimension of $\mathbf{a}_t$, $\mathbf{Q}_t$, $\mathbf{K}_t$, and $\mathbf{V}_t$ denote the query, key, and value in attention mechanism. This unit outputs 128 dimensions vectors $\mathbf{a}_t$, which would be leveraged for predicting the future hydropower generation demand.

4.5 | Fusion and prediction

Feature fusion: Now we can immediately forecast the power using the representations learned by the above two key components of HydroFlow, including the latent variables $\mathbf{z}_{t-1}^{K}$ determined by both VL-RNN extraction and normalizing flow transformation, and the external knowledge vector $\mathbf{a}_t$. In our implementation, we use a multiple layer perceptron (MLP) to predict the hydropower $\hat{\mathbf{v}}_t$ at time t:

$$\hat{\mathbf{v}}_t = \text{MLP}\left(\left[\mathbf{z}_{t-1}^{K}, \mathbf{a}_t\right]\right). \tag{13}$$

Objective: Clearly, our objective is composed of three parts: (1) minimizing the reconstruction in VL-RNN, (2) minimizing the loss between predicted power and the ground-truth, and (3) maximizing the ELBO $\mathcal{L}(\theta, \phi)$ in Equation (9):

$$\ell(\Theta) = ||\mathbf{X}_{t-1} - \hat{\mathbf{X}}_{t-1}||_2^2 + ||\mathbf{v}_t - \hat{\mathbf{v}}_t||_2^2 - \mathcal{L}(\theta, \phi), \tag{14}$$

where Θ refers to the parameter set.

Algorithm 2: HydroFlow

Input: multivaraiate hydropower data $\mathbf{X}_{t-1}$; external attributes $\mathbf{e}_t$;
Output: predicted hydropower generation demand $\hat{\mathbf{v}}_t$.

```
Initialize all parameters Θ of HydroFlow;
while not converged do
    Obtain Z_{t-1} and H_{t-1} by Algorithm 1;
    Sample z^0_{t-1} = μ^z_{t-1} + σ^z_{t-1} × ϵ, ϵ ~ N(0, I);
    Compute z^K_{t-1} using continuous variational inference by Eq.(4) – Eq.(7);
    Reconstruct X_{t-1} via VL-RNN (Eq.(10));
    Obtain embed external features a_t;
    Predict hydropower generation demand v̂_t with input [z_{t-1}, a_t] by Eq.(13);
    Update Θ by minimizing the objective in Eq.(14).
end
```

4.6 | Model analysis

- *Complexity for generating* **Z** *and* **H**: VL-RNN has two GRU cells in each time step, that is, evolution in h-space and z-space, respectively. Besides, VL-RNN needs to compute the mean and variance of the hidden state and latent variables. Thus the time complexity of VL-RNN is four times that of vanilla GRU.
- *Complexity for near-field (NF) transformation*: The time complexity in NF is linear with the number of hidden units $\mathcal{M}$ and the number of transformations K which, consequently, yields $\mathcal{O}(\mathcal{M}) \times \mathcal{O}(\mathcal{K})$.
- *Complexity for external information extraction*: We utilize a self-attention unit to embed the prior knowledge, the time complexity is linear in dimension d_t and quadratic in the length of time series N. Thus the time complexity of this component is $\mathcal{O}(d_a) \times \mathcal{O}(N^2)$.
- *Complexity for other parts of HydroFlow*: The time and space complexities of MLP are related to the input dimensions of latent variables.

Overall, the computation cost of the proposed model is the same as the typical RNN-based autoencoders. Algorithm 2 depicts the training procedure for HydroFlow.

5 | EXPERIMENTS

In this section, we present the result of our evaluations on real-world data sets and compare our method against the baseline approaches. We also investigate the effectiveness of different components of HydroFlow and present intuitive explanations of the model performance.

5.1 | Experimental settings

Data sets. All evaluations are conducted on data sets collected from the three largest hydropower generation stations affiliated with the Dadu River Hydropower Development Co., that is, Pubugou (PDP) Dam, Dagangshan (DGS) Dam, and Shenxigou (PDS) Dam. Figure 4 illustrates the operational components of the hydropower plants.

- Pubugou (*PDP*) hydropower plant is located in Hanyuan County, which has a total of 3600 MW (6×600 MW) installed capacity.
- Dagangshan (*DGS*) is a new station located in Shimian County. It consists of four 650 MW hydropower generators.
- Shenxigou (*PDS*) is at the downstream of PDP and plays the role of antiregulation station of PDP. It is a smaller plant with a 660 MW capacity.

We used 2 years of electricity demand data—spanning from January 1, 2017 to December 31, 2018—from the above-mentioned three hydropower stations for evaluation. We split the data in each station into two periods P1 (2017) and P2 (2018). The time interval of all time series is set as 1 h.

The corresponding natural and social factors are also used for training the models. Besides, the holiday effect features (e.g., HourOfDay) are embedded as one-hot vectors during training. Table 1 shows the statistics of the two data sets. We use the first 80% data for training and examine the model performance with the rest data.

Baselines: We compare our HydroFlow against the following baseline approaches, which are typically used for modeling and prediction in time series:

- *Historical Average (HA)* averages a historical time period T to predict the time series in the next time step. T is fixed as 7 in our evaluation.
- *Autoregressive integrated moving average (ARIMA)*[62] is a generalization of an autoregressive moving average (ARMA) model.
- *ARIMA*[62] is a generalization of an ARMA model.

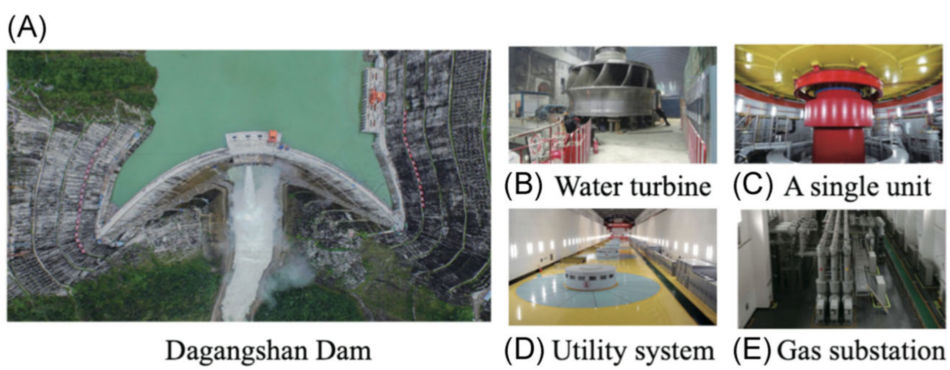


FIGURE 4 (A) A snap of the arch dam in Dagangshan. (B) A picture of the water turbine (650 MW generators) in Dagangshan station. (C) A single electromechanical unit in Pubugou (PDP). (D) The entire utility system of PDP station. (E) The gas-insulated substation in PDP [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 1 Data set description

Data set	PDP	DGS	PDS
Time interval (h)	1	1	1
Power demand (MWh)	[0.0, 3587.8]	[0.0, 2591.2]	[0.0, 668.0]
Water flow			
Water inflow	[0.0, 7020]	[0.0, 4640.0]	[0.0, 11579.0]
Water outflow	[119.0, 5670]	[55.1, 5220.0]	[53.8, 6490.0]
Generation flow	[119.0, 2470]	[55.1, 1730.0]	[55.1, 2480.0]
External Factors			
Temperature (°C)	[−24.4, 21.7]	[−24.4, 21.7]	[−24.4, 21.7]
HourOfDay	[0, 24]	[0, 24]	[0, 24]
DayOfWeek	[1, 7]	[1, 7]	[1, 7]
WeekendOrNot	{0, 1}	{0, 1}	{0, 1}

Note: Private information such as sale price is masked.

Abbreviations: DGS, Dagangshan; PDP, Pubugou; PDS, Shenxigou.

- *Long short-term memory (LSTM)*[51] is an RNN-based model which captures the information of long-short-term dependency and has been widely used for time series forecasting.[50]
- *Bi-GRU*[63] is a bidirectional GRU model that concatenates the forward and backward hidden states for prediction.
- *GRU-VAE*[55] combines GRU and VAE for learning the latent variables of time-series data in an encoder–decoder manner.
- *Bi-GRU-ATT* incorporates attention mechanism into the Bi-GRU network, which can discriminate the vital information and dependency among time series.
- *DeepHydro*[21] captures temporal dependencies in coevolving time series with a conditioned latent RNN. It also introduces neural ODEs to make extrapolation for temporal inference.

Variants of HydroFlow. Because the proposed HydroFlow contains several key components, we derive the following variants to demonstrate the effectiveness of its three major parts:

- *HydroFlow-NF* removes normalizing flow module free-form Jacobian of reversible dynamics (FFJORD) and uses diagonal Gaussian for the latent variables $\mathbf{z}$.
- *HydroFlow-RNN* replaces VL-RNN with deterministic LSTM as the encoder and decoder.
- *HydroFlow-NE* does not account for the external factors and makes predictions based on the multivariate time series only.

Evaluation protocols: Following typical time-series prediction works, we evaluate all methods using three widely used metrics for time-series prediction, that is, Root Mean-Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE).

Hyperparameters: All deep learning algorithms are implemented with PyTorch and are trained with Adam optimizer.[64] The learning rate is initialized as 0.0001 and decays 50% every 20 epochs.

The batch size $\boldsymbol{B} = \mathbf{128}$. For all methods, we run 200 epochs to train the model and then verify the model on testing data. We repeat each experiment 10 times and report the average of results.

5.2 | Evaluation results

5.2.1 | Performance comparison

We report the overall performance of our method compared with the state-of-the-art time-series forecasting models in Table 2. We observe that the proposed HydroFlow significantly outperforms the other algorithms for all three metrics. Our approach's advantage mainly lies in its ability to encode both the observations and latent variables to learn the temporal dependencies among multivariate time series. Linear forecasting models (e.g., ARIMA and its variant) yield inferior results since they are weak in handling nonlinear interactions between different kinds of data and long-range dependencies in multivariate time series. Moreover, simple deep recurrent networks, for example, LSTM, Bi-GRU, Bi-GRU–ATT, do not show comparable performance, because they fail to consider the stochasticity of time-series data. This result also implies that the proper generative process for time-series data modeling helps in learning more meaningful latent representation (Table 3).

5.2.2 | Analysis of HydroFlow components

To validate the effectiveness of each component of our model, we conduct ablation experiments on different HydroFlow variants. The results are depicted in Figure 5. Comparing HydroFlow with HydroFlow-NE, we can see that external factors are crucial for power demand prediction, which shows the necessity of incorporating social and natural knowledge. Next, the advantage of our latent model VL-RNN can be quantified by the improved performance of HydroFlow over HydroFlow-RNN. This result suggests that modeling the temporal dependence among latent variables enhances the model expressiveness. It also reaps notable performance compared with the methods using deterministic RNNs, that is, capturing more informative signals from the stochastic variables. Finally, the difference between HydroFlow and HydroFlow-NF reveals that the model performance is improved considerably by exploiting the continuous normalizing flow FFJORD in posterior approximation.

5.2.3 | Influence of different factors

We now further investigated the influence of various external factors by individually removing them from HydroFlow and present the results in Figure 6. Each dimension of the radar figure indicates the discrepancy between HydroFlow and the model without the corresponding factor—that is, the larger the value, the more significant the factor. As expected, water inflow and outflow are informative for forecasting power demand, since the water derives the hydropower electricity and it is an important standard in dispatching. Furthermore, the holiday effect, temperature, and electricity price are also correlated to hydropower demand, for example, seasonal fluctuation, residential and industrial electricity consumption, and so forth. All of these should be taken into account for fine-grained power demand forecasting.

TABLE 2 Performance comparisons on PDP and DGS over two different time spans

Data sets	PDP						DGS					
Time span	**P1**			**P2**			**P1**			**P2**		
Method	**RMSE**	**MAE**	**MAPE**	**RMSE**	**MAE**	**MAPE**	**RMSE**	**MAE**	**MAPE**	**RMSE**	**MAE**	**MAPE**
HA	684.9	543.5	0.409	744.8	586.5	0.469	491.8	390.1	0.823	533.8	434.9	0.963
ARIMA	338.7	245.3	0.378	378.6	282.5	0.462	242.5	165.4	0.733	278.7	227.6	0.882
SARIMA	321.3	235.6	0.370	366.1	274.3	0.458	239.4	161.8	0.735	273.3	220.5	0.884
LSTM	270.3	209.5	0.346	296.8	224.8	0.442	200.3	143.9	0.631	235.4	182.3	0.839
Bi-GRU	267.6	206.4	0.339	295.6	222.6	0.430	197.8	137.6	0.605	233.2	174.6	0.702
GRU-VAE	268.1	206.9	0.344	295.8	223.1	0.438	195.3	138.6	0.621	232.1	173.2	0.669
Bi-GRU-ATT	266.1	205.5	0.338	294.1	221.9	0.433	193.8	136.7	0.580	232.0	172.2	0.643
DeepHydro	228.1	176.0	0.285	248.4	189.2	0.379	144.1	106.4	0.412	182.2	133.4	0.537
HydroFlow	**225.3**	**173.9**	**0.273**	**246.7**	**187.7**	**0.367**	**143.1**	**105.9**	**0.402**	**181.1**	**132.3**	**0.526**

Note: Best performance is in bold font.

Abbreviations: ARIMA, autoregressive integrated moving average; ATT, attention; Bi-GRU, bidirectional GRU; DGS, Dagangshan; GRU, gated recurrent unit; HA, historical average; LSTM, long short-term memory; MAE, Mean Absolute Error; MAPE, Mean Absolute Percentage Error; PDP, Pubugou; RMSE, Root Mean-Squared Error; SARIMA, seasonal ARIMA; VAE, variational autoencoder.

TABLE 3 Performance comparisons on Shenxigou (PDS) over two different time spans

Data sets	PDS			
Time span	P1		P2	
Method	RMSE	MAPE	RMSE	MAPE
HA	139.3	0.590	139.3	0.446
ARIMA	72.1	0.322	74.8	0.283
SARIMA	71.3	0.318	73.4	0.273
LSTM	51.6	0.208	53.3	0.182
Bi-GRU	50.8	0.201	52.6	0.179
GRU-VAE	50.2	0.201	52.1	0.178
Bi-GRU-ATT	50.1	0.204	51.9	0.177
DeepHydro	35.2	0.169	38.8	0.153
HydroFlow	**33.9**	**0.163**	**37.3**	**0.148**

Note: Best performance is in bold font.

Abbreviations: ARIMA, autoregressive integrated moving average; ATT, attention; Bi-GRU, bidirectional GRU; GRU, gated recurrent unit; HA, historical average; LSTM, long short-term memory; MAPE, Mean Absolute Percentage Error; PDS: Shenxigou; RMSE, Root Mean-Squared Error; SARIMA, seasonal ARIMA; VAE, variational autoencoder.

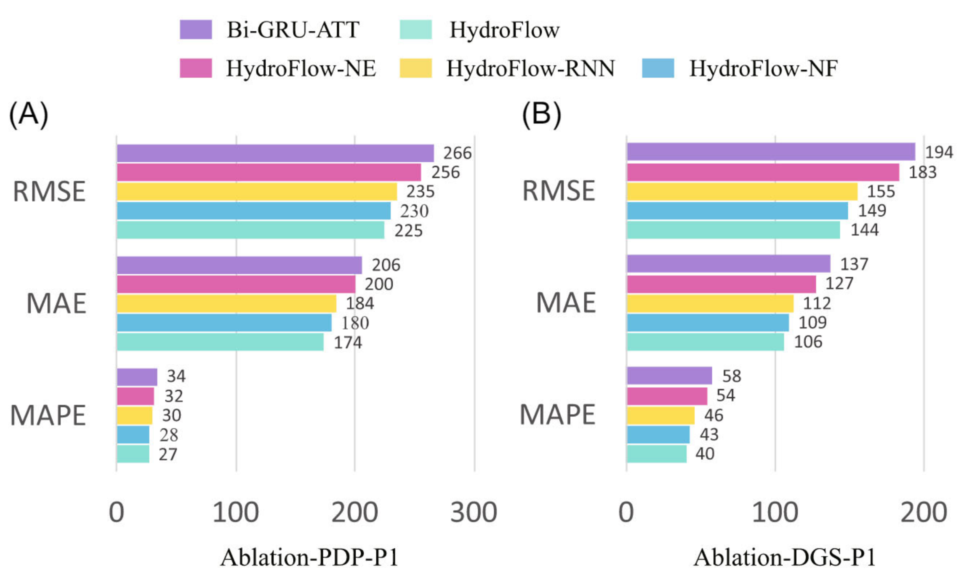
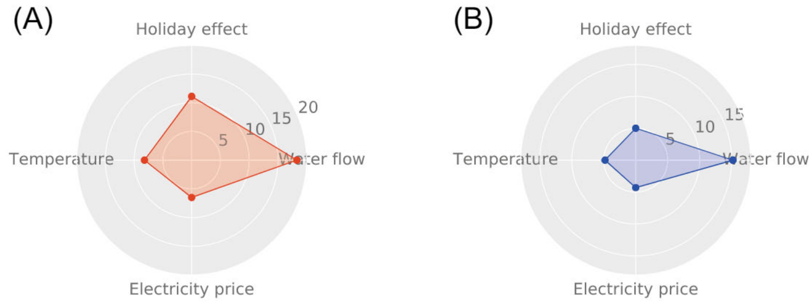


FIGURE 5 The analysis of HydroFlow components. ATT, attention; Bi-GRU, bidirectional GRU; GRU, gated recurrent unit; MAE, Mean Absolute Error; MAPE, Mean Absolute Percentage Error; NE, northeast; NF, near-field; PDP, Pubugou; RMSE, Root Mean-Squared Error; RNN, recurrent neural network [Color figure can be viewed at wileyonlinelibrary.com]

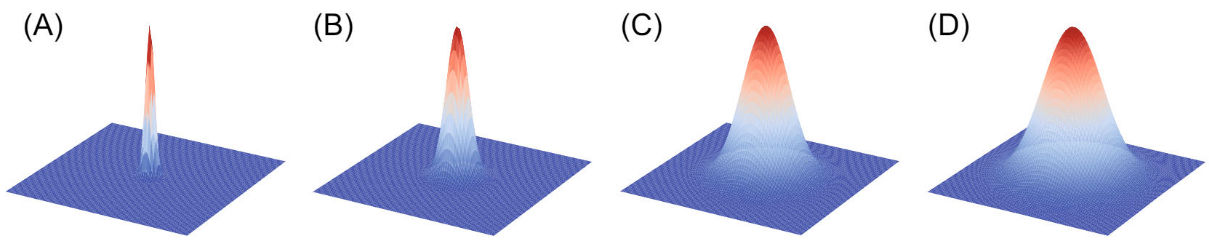
5.2.4 | Latent representation learning

As a Bayesian learning method, the latent variables learned by HydroFlow carry important time-series information. Figure 7 plots the distribution of $\mathbf{z}$, along with the process of learning temporal dependencies in our VL-RNN model. In the beginning, the latent variables are



Improvement on RMSE. Improvement on MAPE.

FIGURE 6 Influence of different factors (DGS-P1). DGS, Shenxigou; MAPE, Mean Absolute Percentage Error; RMSE, Root Mean-Squared Error [Color figure can be viewed at wileyonlinelibrary.com]



$t = 4.$ $t = 16.$ $t = 64.$ $t = 168.$

FIGURE 7 Variational latent representation learned in VL-RNN. $\mathbf{z}_{168}$ ($168 = 7 \times 24$) is the output of VL-RNN, compare Figure 2. VL-RNN, variational latent recurrent neural network [Color figure can be viewed at wileyonlinelibrary.com]

clustered in a very narrow area. The model's capability grows over time with more observations, which means the model captures more robust representation through exploring more latent space, which, to some extent, explains why HydroFlow learns more expressive temporal dependencies than vanilla RNNs.

Though VL-RNN preserves the temporal dependencies of latent variables, the distribution of $\mathbf{z}$ is regularized to be a Gaussian, which might be a too strong assumption limiting the capability of the variational inference. Figure 8 shows the process of transforming $\mathbf{z}$ (after VL-RNN) to approximate the desired posterior distribution. When applying NF on the random variable $\mathbf{z}$, the inference network propagates the progression of its density by transforming variables towards more complex and real distributions. In addition, the inference network is trained to extrapolate the density at time t, which allows our model to obtain the instantaneous change of variables at a future moment. We note that this transformation is reversible,[43] which means that we can efficiently sample from the learned density to decode the time series from the latent variables.

5.2.5 | Qualitative study

We randomly select a day, a week, and a month of the power demand data from the testing set, and visualize the predicted results by comparing HydroFlow with the ground-truth

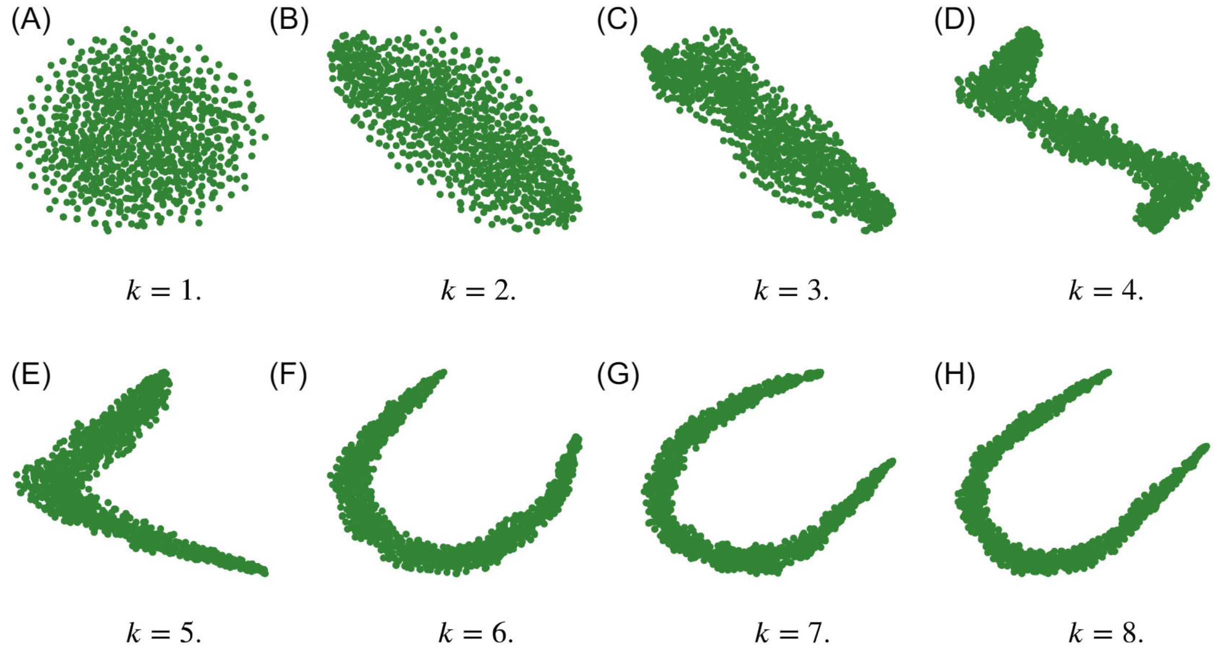
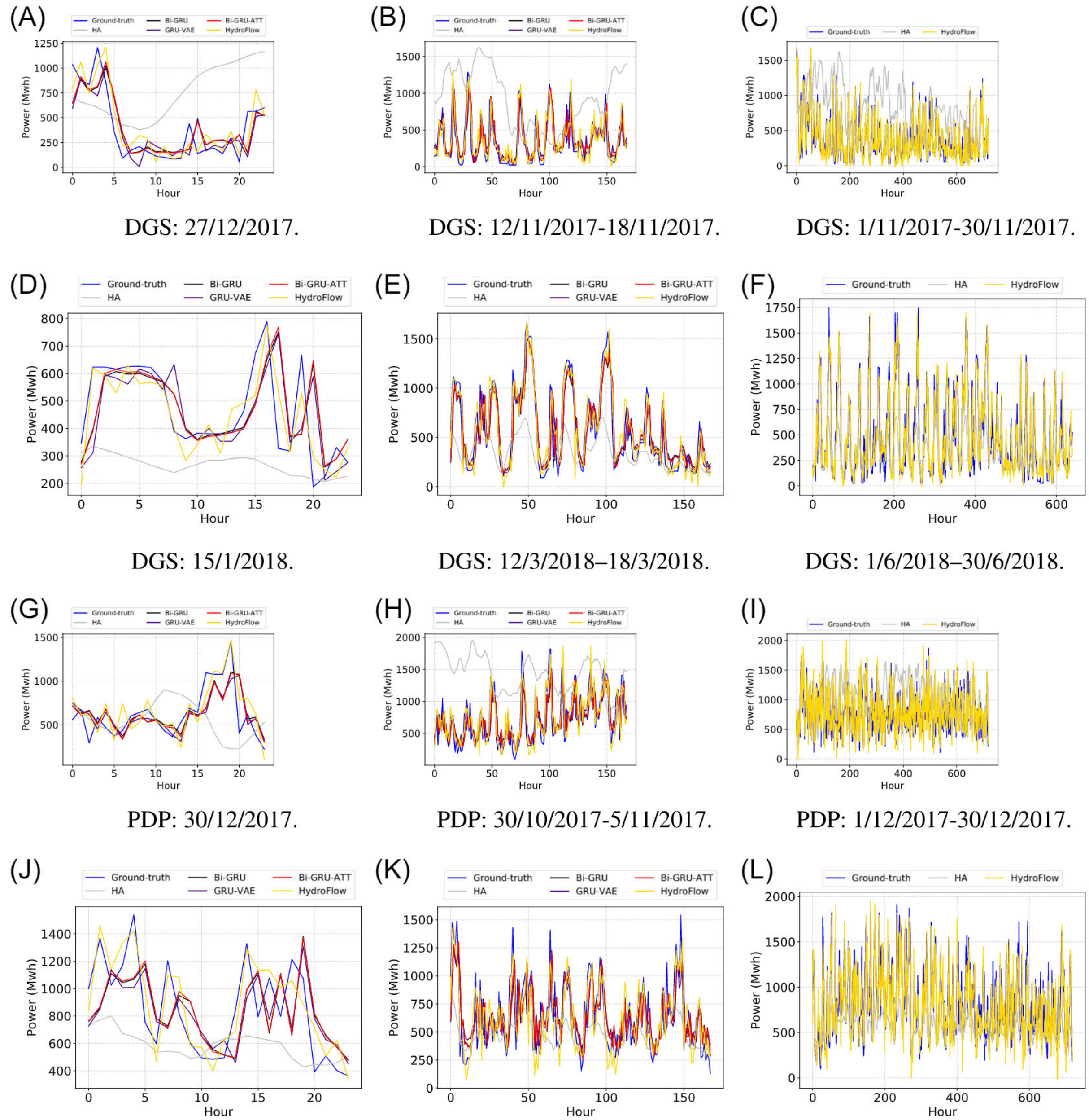


FIGURE 8 Visualization of the transformation of the latent variables (PDP-P1). Note that the number of flow in FFJORD is 8. FFJORD, free-form Jacobian of reversible dynamics; PDP, Pubugou [Color figure can be viewed at wileyonlinelibrary.com]

and the results from strong baselines, as shown in Figure 9. Generally, HydroFlow fits better on spiking changes, mainly due to the ability to leverage the knowledge of external factors, such as electricity price and water inflow/outflow of reservoirs. We respectfully note that the extreme events, for example, the full-load operation and scheduled outage, are still hard to be predicted. A possible solution is to consolidate the power grid dispatch command and multiobjective dispatch problem in cascaded hydropower systems for such event detection, which is beyond the scope of this paper and left for our future work.

5.2.6 | Deployment

Our model has been successfully deployed on the HydroDAD mining platform in the Dadu River Hydropower Development Co. and is continuously optimized with new hydropower data. Figure 10 shows a snapshot of the HydroDAD platform, where both natural factors (e.g., weather, precipitation, and water level) and social factors (electricity market share and sale price) are displayed. Real-time hydropower demand predicted by our HydroFlow is shown in Figure 11. As we can see, the (predicted) power demand (yellow line) is far from reaching the full-load (red line). This phenomenon happens due to the lower water level of the reservoir and relatively less precipitation in the winter—note that the day of this snapshot is December 19, 2019. Our ultimate goal is to minimize the gap between the two lines by optimizing power demand allocation, which could maximize the enterprise's profit.



DGS: 27/12/2017. DGS: 12/11/2017-18/11/2017. DGS: 1/11/2017-30/11/2017.

DGS: 15/1/2018. DGS: 12/3/2018–18/3/2018. DGS: 1/6/2018–30/6/2018.

PDP: 30/12/2017. PDP: 30/10/2017-5/11/2017. PDP: 1/12/2017-30/12/2017.

PDP: 30/1/2018. PDP: 30/4/2018–6/5/2018. PDP: 1/8/2018–30/8/2018.

FIGURE 9 Prediction versus ground-truth. ATT, attention; Bi-GRU, bidirectional GRU; DGS, Shenxigou; GRU, gated recurrent unit; HA, historical average; PDP, Pubugou; VAE, variational autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

6 | CONCLUSION

In this paper, we presented a deep generative model to address the problem of hydropower demand forecasting. We used stochastic RNNs with variational inference to capture the sequential dependencies and uncertainty of the power-related time series. Normalizing flows with flexible density transformers are used to alleviate the agnostic posterior estimation problem in stochastic variational inference. Experimental results on real-world hydropower data

FIGURE 10 User interface of the deployment platform. There exists our hydropower demand prediction results and various IIoT sensors monitoring data for industrial operations and decisions making. IIoT, Industrial Internet of Things; KPI, key performance indicator [Color figure can be viewed at wileyonlinelibrary.com]

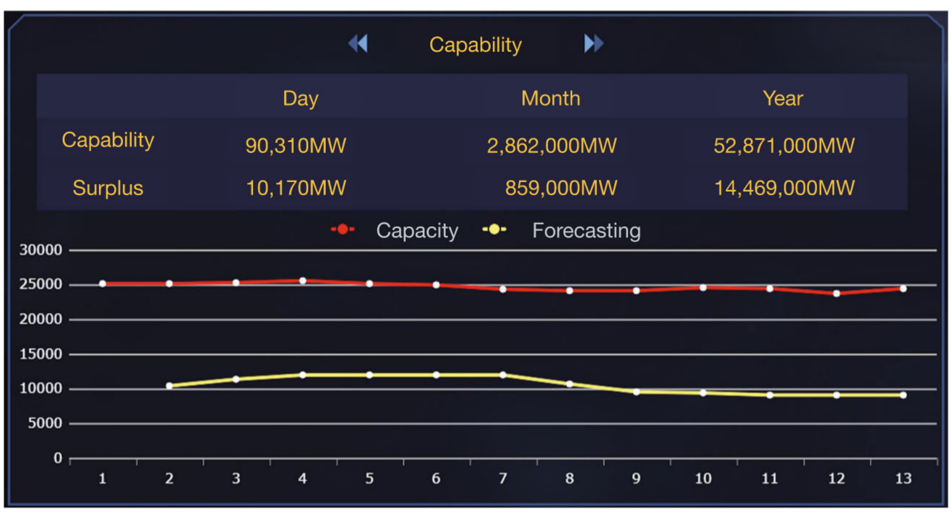


FIGURE 11 Power demand forecasting by HydroFlow on HydroDAD platform. HydroDAD, Hydropower Data Analysis and Data [Color figure can be viewed at wileyonlinelibrary.com]

demonstrate the effectiveness and superiority of the proposed framework in solving the hydropower demand prediction problem.

As our future work, we plan to improve the power forecasting performance by incorporating climate changes and meteorological satellite data. Besides, we are interested in simultaneously predicting the power generation of the cascaded hydroplants in the company, which allows us to optimize the electricity generation distribution and system operation by dynamically coordinating the demand allocation. Another objective of our future work is to accurately forecast the water inflow in the reservoirs and maximize power generation and profit. Lastly, as the increasing wind and PV power generated in our company, it is desirable to determine future electricity power trends and proactively alleviate the imbalances between different energy sources, which could help increase the robustness of the system to better support the smart grid demand.

ACKNOWLEDGMENTS

This study was supported by the National Natural Science Foundation of China (Grant Nos. 62176043 and 62072077) and the NSF SWIFT grant 2030249.

ORCID

Fan Zhou https://orcid.org/0000-0002-8038-8150
Zhiyuan Wang https://orcid.org/0000-0001-7167-9055
Ting Zhong https://orcid.org/0000-0002-8163-3146
Goce Trajcevski https://orcid.org/0000-0002-8839-6278
Ashfaq Khokhar https://orcid.org/0000-0002-6504-8502

ENDNOTE

*https://www.c2es.org/content/renewable-energy/

REFERENCES

1. Rossi B, Chren S. Smart grids data analysis: a systematic mapping study. *IEEE Trans Ind Inf*. 2019;16(6):3619-3639.
2. Gao Y, Ai Q, Wang X, Yousif M. Distributed cooperative economic optimization strategy of a regional energy network based on energy cell-tissue architecture. *IEEE Trans Ind Inf*. 2019;15(9):5182-5193.
3. Kishor N, Singh S. Simulated response of NN based identification and predictive control of hydro plant. *Expert Syst Appl*. 2007;32(1):233-244.
4. Petrik M, Zilberstein S. *Linear Dynamic Programs for Resource Management*. AAAI; 2011.
5. Lin X, Wu J, Mumtaz S, Garg S, Li J, Guizani M. Blockchain-based on-demand computing resource trading in IoV-assisted smart city. *IEEE Trans Emerging Top Comput*. 2021;9(3):1373-1385.
6. Kishor N, Saini R, Singh S. A review on hydropower plant models and control. *Renewable Sustainable Energy Rev*. 2007;11(5):776-796.
7. Zamora R, Srivastava AK. Controls for microgrids with storage: review, challenges, and research needs. *Renewable Sustainable Energy Rev*. 2010;14(7):2009-2018.
8. Hossain MA, Pota HR, Hossain MJ, Blaabjerg F. Evolution of microgrids with converter-interfaced generations: challenges and opportunities. *Int J Electr Power Energy Syst*. 2019;109:160-186.
9. Sharma S, Singh B, Mishra S. Economic operation and quality control in PV-BES-DG-based autonomous system. *IEEE Trans Ind Inf*. 2019;16(4):2346-2356.
10. Du J, Jiang C, Han Z, Zhang H, Mumtaz S, Ren Y. Contract mechanism and performance analysis for data transaction in mobile social networks. *IEEE Trans Network Sci Eng*. 2017;6(2):103-115.
11. Chandra K, Marcano AS, Mumtaz S, Prasad RV, Christiansen HL. Unveiling capacity gains in ultradense networks: using mm-wave NOMA. *IEEE Veh Technol Mag*. 2018;13(2):75-83.
12. Cai G, Fang Y, Wen J, Mumtaz S, Song Y, Frascolla V. Multi-carrier M-ary DCSK system with code index modulation: an efficient solution for chaotic communications. *IEEE J Sel Top Signal Process*. 2019;13(6):1375-1386.
13. Mierla M, Romanescu G, Nichersu I, Grigoras I. Hydrological Risk map for the Danube delta–a case study of floods within the fluvial delta. *IEEE J Sel Top Appl Earth Obs Remote Sens*. 2015;8(1):98-104.
14. Ho MS, Wang Z, Yu Z. *China's Power Generation Dispatch*. https://www.rff.org/publications/reports/chinas-power-generation-dispatch; 2017.
15. Stoll B, Andrade J, Cohen S, Brinkman G, Martinez-Anido CB. *Hydropower Modeling Challenges*. Tech. Rep. NREL/TP-5D00-68231. National Renewable Energy Lab. (NREL), Golden, CO, USA; 2017.
16. Chen J, Zhong PA. A multi-time-scale power prediction model of hydropower station considering multiple uncertainties. *Sci Total Environ*. 2019;677:612-625.

17. Fekete K, Nikolovski S, Klaić Z, Androjić A. Optimal re-dispatching of cascaded hydropower plants using quadratic programming and chance-constrained programming. *Energies*. 2019;12(9):1604.
18. Messaoud S, Bradai A, Bukhari SHR, Quang PTA, Ahmed OB, Atri M. A survey on machine learning in Internet of Things: algorithms, strategies, and applications. *Internet Things*. 2020;12:100314.
19. Song HA, Hooi B, Jereminov M, Pandey A, Pileggi L, Faloutsos C. PowerCast: mining and forecasting power grid sequences. In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer; 2017:606-621.
20. Hooi B, Song HA, Jereminov M, Pileggi L, Faloutsos C. StreamCast: fast and online mining of power grid time sequences. In: *SDM*. SIAM; 2018:531-539.
21. Zhou F, Li L, Zhang K, et al. Forecasting the evolution of hydropower generation. In: *KDD*. ACM; 2020:2861-2870.
22. Ma XY, Sun YZ, Fang HL, Tian Y. Scenario-based multiobjective decision-making of optimal access point for wind power transmission corridor in the load centers. *IEEE Trans Sustainable Energy*. 2012;4(1):229-239.
23. Pinson P, Girard R. Evaluating the quality of scenarios of short-term wind power generation. *Appl Energy*. 2012;96:12-20.
24. Wang Y, Dvorkin Y, Fernandez-Blanco R, Xu B, Qiu T, Kirschen DS. Look-ahead bidding strategy for energy storage. *IEEE Trans Sustainable Energy*. 2017;8(3):1106-1117.
25. Wu W, Peng M. A data mining approach combining k-means clustering with bagging neural network for short-term wind power forecasting. *IEEE Internet Things J*. 2017;4(4):979-986.
26. Wang Y, Shen Y, Mao S, Chen X, Zou H. LASSO and LSTM integrated temporal model for short-term solar intensity forecasting. *IEEE Internet Things J*. 2018;6(2):2933-2944.
27. Tang C, Wang Y, Xu J, Sun Y, Zhang B. Efficient scenario generation of multiple renewable power plants considering spatial and temporal correlations. *Appl Energy*. 2018;221:348-357.
28. Tang N, Mao S, Wang Y, Nelms R. Solar power generation forecasting with a LASSO-based approach. *IEEE Internet Things J*. 2018;5(2):1090-1099.
29. Chen MR, Zeng GQ, Lu KD, Weng J. A two-layer nonlinear combination method for short-term wind speed prediction based on ELM, ENN, and LSTM. *IEEE Internet Things J*. 2019;6(4):6997-7010.
30. Zhou F, Li L. *Forecasting Reservoir Inflow Via Recurrent Neural ODEs*. Vol 35. AAAI; 2021:15025-15032.
31. Pang Y, Yao B, Zhou X, Zhang Y, Xu Y, Tan Z. Hierarchical electricity time series forecasting for integrating consumption patterns analysis and aggregation consistency. In: *IJCAI*; 2018:3506-3512.
32. Salem TS, Kathuria K, Ramampiaro H, Langseth H. *Forecasting Intra-hour Imbalances in Electric Power Systems*. Vol 33. AAAI; 2019:9595-9600.
33. De Giorgi MG, Malvoni M, Congedo PM. Comparison of strategies for multi-step ahead photovoltaic power forecasting models based on hybrid group method of data handling networks and least square support vector machine. *Energy*. 2016;107:360-373.
34. Li LL, Wen SY, Tseng ML, Wang CS. Renewable energy prediction: a novel short-term prediction model of photovoltaic output power. *J Cleaner Prod*. 2019;228:359-375.
35. Yan J, Ouyang T. Advanced wind power prediction based on data-driven error correction. *Energy Convers Manage*. 2019;180:302-311.
36. Yu B, Xu L. Review of ecological compensation in hydropower development. *Renewable Sustainable Energy Rev*. 2016;55:729-738.
37. Wild TB, Reed PM, Loucks DP, Mallen-Cooper M, Jensen ED. Balancing hydropower development and ecological impacts in the Mekong: tradeoffs for Sambor Mega Dam. *J Water Resour Plan Manage*. 2019;145(2):05018019.
38. Xu C, Xu Z, Yang Z. Reservoir operation optimization for balancing hydropower generation and biodiversity conservation in a downstream wetland. *J Cleaner Prod*. 2020;245:118885.
39. Baird I, Manorom K, Phenow A, Gaja-Svasti S. Opening the gates of the Pak Mun Dam: fish migrations, domestic water supply, irrigation projects and politics. *Water Altern*. 2020;13:141-159.
40. Qin Y, Sheng QZ, Falkner NJG, Dustdar S, Wang H, Vasilakos AV. When things matter: a survey on data-centric internet of things. *J Network Comput Appl*. 2016;64:137-153.
41. Ullah I, Youn HY. Intelligent data fusion for smart IoT environment: a survey. *Wireless Pers Commun*. 2020;114(1):409-430.
42. Javed U, Fraz MM, Mahmood I, Shahzad M, Arif O. Forecasting of electricity generation for hydro power plants. In: *International Conference on Smart Communities: Improving Quality of Life Using ICT, IoT and AI (HONET)*. IEEE; 2020:32-36.

43. Rezende D, Mohamed S. Variational inference with normalizing flows. In: ICML. PMLR; 2015:1530-1538.
44. Bedi G, Venayagamoorthy GK, Singh R. Development of an IoT-driven building environment for prediction of electric energy consumption. *IEEE Internet Things J*. 2020;7(6):4912-4921.
45. Zhong W, Guo J, Chen L, Zhou J, Zhang J, Wang D. Future hydropower generation prediction of large-scale reservoirs in the upper Yangtze River Basin under climate change. *J Hydrol*. 2020;588(11):125013.
46. Li L, Yao F, Huang Y, Zhou F. Hydropower generation forecasting via deep neural network. In: *ICISCE*. IEEE; 2019:324-328.
47. Zhang Q, Wangg M, Wang X, Tian S. Mid-long term optimal dispatching method of power system with large-scale wind-photovoltaic-hydro power generation. In: *IEEE Conference on Energy Internet and Energy System Integration* (EI2). IEEE; 2017:1-6.
48. Liu J, Li D, Wu Y, Liu D. Lion swarm optimization algorithm for comparative study with application to optimal dispatch of cascade hydropower stations. *Appl Soft Comput*. 2020;87(1):105974.
49. Zhang H, Chang J, Gao C, et al. Cascade hydropower plants operation considering comprehensive ecological water demands. *Energy Convers Manage*. 2019;180(8):119-133.
50. Han Z, Zhao J, Leung H, Ma KF, Wang W. A review of deep learning models for time series prediction. *IEEE Sens J*. 2019;21(6):7833-7848.
51. Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput*. 1997;9(8):1735-1780.
52. Rasul K, Sheikh A, Schuster I, Bergmann UM, Vollgraf R. Multivariate probabilistic time series forecasting via conditioned normalizing flows. In: *ICLR*; 2021.
53. Ravuri S, Lenc K, Willson M, et al. Skillful precipitation nowcasting using deep generative models of radar. *Nature*. 2021;597:672-677.
54. Grathwohl W, Chen RTQ, Bettencourt J, Sutskever I, Duvenaud D. FFJORD: free-form continuous dynamics for scalable reversible generative models. In: *ICLR*; 2019.
55. Xu H, Chen W, Zhao N, et al. Unsupervised anomaly detection via variational auto-encoder for seasonal KPIs in web applications. In: *WWW*. ACM; 2018:187-196.
56. Kingma DP, Welling M. Auto-encoding variational Bayes. In: *ICLR*; 2014.
57. Papamakarios G, Nalisnick E, Rezende DJ, Mohamed S, Lakshminarayanan B. Normalizing flows for probabilistic modeling and inference. *J Mach Learn Res*. 2021;22(57):1-64.
58. Chen TQ, Rubanova Y, Bettencourt J, Duvenaud S, Zilberstein DK. Neural ordinary differential equations. In: *NeuIPS*; 2018.
59. Hutchinson MF. A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines. *Commun Stat—Simul Comput*. 1989;18(3):1059-1076.
60. Rezende DJ, Mohamed S, Wierstra D. Stochastic backpropagation and approximate inference in deep generative models. In: *ICML*. PMLR; 2014.
61. Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. In: *NIPS*; 2017:5998-6008.
62. Bartholomew DJ. Time series analysis forecasting and control. *J Oper Res Soc*. 1971;22(2):199-201. doi:10.1057/jors.1971.52
63. Chung J, Gulcehre C, Cho K, Bengio Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv*:1412.3555. 2014.
64. Kingma DP, Ba J. A method for stochastic optimization. In: *ICLR*; 2015.
65. Chen P, Liu S, Shi C, Hooi B, Wang B, Cheng X. NeuCast: seasonal neural forecast of power grid time series. In: *IJCAI*; 2018:3315-3321.

How to cite this article: Zhou F, Wang Z, Zhong T, Trajcevski G, Khokhar A. HydroFlow: Towards probabilistic electricity demand prediction using variational autoregressive models and normalizing flows. *Int J Intell Syst*. 2022;37:6833-6856. doi:10.1002/int.22864