PROCEEDINGS, 46th Workshop on Geothermal Reservoir Engineering
Stanford University, Stanford, California, February 15-17, 2021
SGP-TR-218

Deep Learning for Prediction and Fault Detection in Geothermal Operations

Yingxiang Liu, (USC), Wei Ling (USC), Robert Young (USC), Trenton T. Cladouhos (Cyrq Energy Inc.),
Jalal Zia (Cyrq Energy Inc.) and Behnam Jafarpour (USC)

Ming Hsieh Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, California, USA
yingxian@usc.edu

Keywords: Deep learning, predictive analytics, dynamic neural networks, geothermal power plants

ABSTRACT

Automation, control, and real-time surveillance of geothermal power plants require robust and accurate predictive models. However, building physics-based models is fraught with several challenges, including the complexity and uncertainty in describing the processes and technically involved and computationally demanding workflows for model calibration, uncertainty quantification and optimization. Advances in cost-effective monitoring data acquisition systems have motivated the development of data-driven predictive analytics as a flexible, efficient, and easy-to-deploy alternative for real-time application. We present a dynamic neural network model for performance prediction, control, and real-time monitoring of geothermal power plants. Measurements collected from geothermal power plants are often high dimensional, highly cross-correlated, and auto-correlated. In addition, the measurements are often influenced by disturbances and subsequent adjustments of operational settings. Therefore, we present a neural network architecture using with a low-dimensional latent space representation to learn the dynamical behavior of the power plant system under variable control settings. Our developed model consists of three parts: (i) a sliding-window encoder for capturing data correlations to enable latent-space representation, (ii) fully connected neural network layers to describe the evolution of the latent states to make predictions while accounting for the effects of control changes, and (iii) a decoder that maps the prediction results from the low-dimensional latent space back onto the original data space. The resulting dynamic neural network architecture is then trained using historical monitoring and performance data from the power plant to extract the underlying dynamics in the data and use that to generate predictions. Once trained, the model can be used to predict important time-varying responses of the power plant based on observed measurements and referenced control settings. The prediction results can be used for applications such as model predictive control and fault detection. We apply our proposed neural network model to field data from a binary cycle geothermal power plant to demonstrate its prediction and fault (abnormal event) detection performance.

1. INTRODUCTION

Accurate predictions of geothermal power plants' dynamics play an important role in the control, optimization, and real-time surveillance of the power plant. Physics-based simulation models have traditionally been applied to predict the behavior of geothermal power plants. However, they are typically too complex to construct and involve many uncertain parameters since the surface power plants are built based on a precise off-design model specific to the characteristic of the field and ambient conditions (Manente et al., 2013; Wang et al., 2014). Traditional regression approaches such as autoregressive moving average (ARMA) and autoregressive integrated moving average (ARIMA) show disadvantages when solving complex nonlinear problems (Du et al. 2018). With recent advances in machine learning, deep neural network (DNN), convolutional neural network (CNN), and recurrent neural network (RNN) have been used, with help from domain knowledge, to represent the nonlinearity in system dynamics. Arslan et al. (2011) and Tugcu et al. (2017) utilized the artificial neural network (ANN) to optimize the surface process by involving the economic parameter. Haklidir et al. (2020) suggested the DNN provides better performance over the traditional machine learning models in predicting the geothermal reservoir temperature. Shi et al. (2021) proposed a Long Short-Term Memory (LSTM) and Multi-Layer Perceptron (MLP) combinational neural network to forecast the geothermal productivity with constraint conditions. Currently, most applications of deep learning methods in geothermal power systems focus on subsurface and steady-state surface processes. This study presents a dynamic neural network architecture that can learn the dynamics of geothermal power plants from multivariate time series data. By learning the dynamics influenced by the operation adjustments and environmental changes, the model is capable of making multi-step ahead predictions for a given set of control trajectories.

2. METHODOLOGY

We present a neural network architecture suitable to the characteristics of data collected from geothermal power plants and applications related to power plant operations. Due to the nature of physics involved in the power generation process, time-series data collected from geothermal power plants are cross-correlated and auto-correlated. Therefore, a desired property of the dynamic neural network model is the ability to capture the dependencies among different variables and their variations in time. Other desired properties for applications such as model predictive control and fault detection are parsimony and interpretability. However, neural networks are often referred to as 'black-box' models since they approximate a function without providing insights into the function's structure. Some insights about how the predictions are generated may be achieved through domain-specific adaptations and designs to enhance the dynamic network model's interpretability and simplicity.

Figure 1 shows the schematic of the dynamic neural network architecture, which consists of three components: an encoder, a latent-space dynamical model, and a decoder. In this diagram, information and dynamics flow from left to right. First, the encoder brings the past k measurements into a reduced-dimension latent space. This step is designed to exploit the correlations among the variables to enable

dimensionality reduction and latent-space representation. Next, a dynamical model processes the encoded latent states and additional (control) input to make a 1-step ahead prediction in latent space. The dynamical model represents the latent states' evolution and the effects of input changes. In the end, the predictions in the latent space are mapped back to the original data space using a decoder structure. For further predictions in time, the same dynamical model is used to process the information stored in previous latent states and new inputs. In addition, the same decoder is used to take the latent state prediction back to the original space.

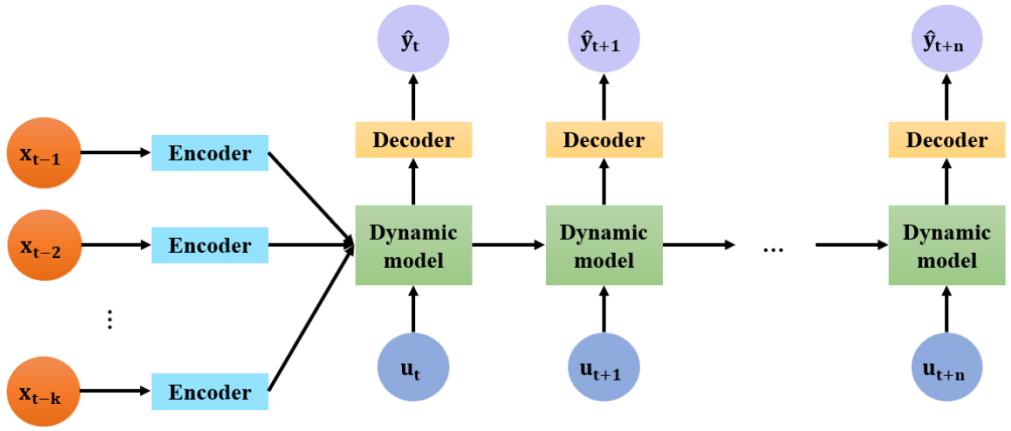

Figure 1: Dynamic neural network model structure.

To specify a dynamic neural network model, three parameters need to be determined: the number of past samples used, the latent space dimension, and the prediction horizon. The details of a dynamic neural network model are illustrated with an example in Figure 2. In this example, a neural network model is used to maps three past samples into a three-dimensional latent space for a 2-steps ahead prediction. The encoder represented by a fully connected layer is used to bring the past 3 samples into a latent space. The encoder can be treated as a function that maps an n-dimensional sample vector $\boldsymbol{x}$ to a reduced dimension h by linearly combining the original variables. The 3 encoded samples are then stacked as a $3 * h$ dimensional vector, which can be presented as

$$[\boldsymbol{z}_{k-3}, \boldsymbol{z}_{k-2}, \boldsymbol{z}_{k-1}] = [f_{encoder}(\boldsymbol{x}_{k-3}), f_{encoder}(\boldsymbol{x}_{k-2}), f_{encoder}(\boldsymbol{x}_{k-1})],$$

where $\boldsymbol{z}_{k-1}$ represents the latent state at time $k-1$. One step ahead prediction of $\hat{\boldsymbol{z}}_k$ is then performed in the latent space using the dynamical model by operating on the stacked vector $[\boldsymbol{z}_{k-3}, \boldsymbol{z}_{k-2}, \boldsymbol{z}_{k-1}]$ and the input $\boldsymbol{u}_k$. The dynamic model is represented using 4 fully connected layers along with an elementwise addition. The first 2 layers represent the evolution of the latent states, which take in the stacked vector $[\boldsymbol{z}_{k-3}, \boldsymbol{z}_{k-2}, \boldsymbol{z}_{k-1}]$ and output a h dimensional vector. The remaining 2 layers are used to map the input $\boldsymbol{u}_k$ onto the h-dimensional latent space. The effect of inputs on the latent state is represented as an elementwise addition of the layers' outputs. As a result, the prediction of $\hat{\boldsymbol{z}}_k$ can be expressed as

$$\hat{\boldsymbol{z}}_k = g([\boldsymbol{z}_{k-3}, \boldsymbol{z}_{k-2}, \boldsymbol{z}_{k-1}]) + f_{control}(\boldsymbol{u}_k).$$

After $\hat{\boldsymbol{z}}_k$ is calculated from the dynamical model, a fully connected layer is used as a decoder to map the predictions in latent space back to the original data space. The prediction of $\hat{\boldsymbol{x}}_k$ can be written as

$$\hat{\boldsymbol{x}}_k = f_{decoder}(\hat{\boldsymbol{z}}_k)$$

The same dynamic model and decoder are used to evolve the latent state's dynamics and to make further predictions in time. To this end, in addition to the input $\boldsymbol{u}_{k+1}$, the most recent latent states, that is, the predicted value $\hat{\boldsymbol{z}}_k$ and two previously encoded vectors $\boldsymbol{z}_{k-2}, \boldsymbol{z}_{k-1}$ are used to make the prediction for the next time step. Hence, the prediction $\hat{\boldsymbol{z}}_{k+1}$ is the output of the function $g([\boldsymbol{z}_{k-2}, \boldsymbol{z}_{k-1}, \hat{\boldsymbol{z}}_k]) + f_{control}(\boldsymbol{u}_{k+1})$, and the prediction $\hat{\boldsymbol{x}}_{k+1}$ is the output of the decoder $f_{decoder}(\hat{\boldsymbol{z}}_{k+1})$.

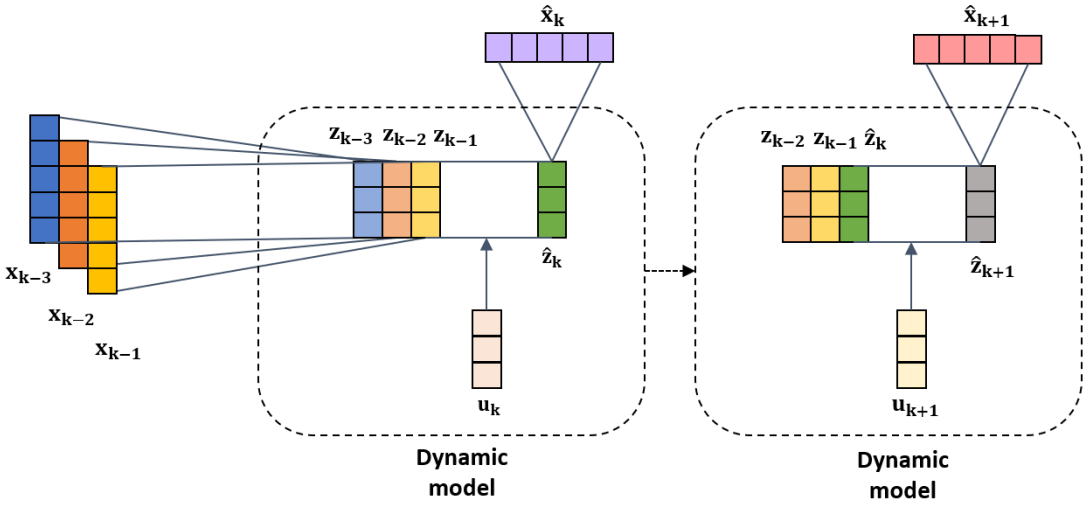

Figure 2: An example of a dynamic neural network model.

3. RESULTS

3.1 Training the dynamic neural network model

Field data from a binary cycle geothermal power plant is used to evaluate the performance of the dynamic neural network model. The datasets used for training the neural network are from a geothermal power plant operated by Cyrq Energy Inc. After examining the datasets, 16 variables related to the primary cycle, secondary cycle, and turbine of the power generation units are used for training. To incorporate the changes in operational settings, three control variables, the brine outlet flow, the IGV set point, and the R134 pump speed, are selected as input variables. The ambient temperature is also included as an input since it closely relates to the power plant's operation and performance.

After removing samples during shutdown periods, data collected during normal operation from three power generation units are normalized and partitioned into a training dataset and a validation dataset. As for the dynamic neural network model, the number of past samples used is selected to be 11. The latent space dimension is selected to be 13, and the prediction horizon is determined to be 48. During training, the mean squared error of the 48-step ahead predictions is minimized by adjusting the weights through backpropagation. The network weights are regularized using L_1-norm minimization. The validation dataset is used to select the best performing model.

3.2 Prediction performance

Once trained, the model is applied to two periods of data that are not in the training and validation datasets. To ensure that the trained model generalizes to different datasets, the two periods of data are selected from two different power generation units. Figure 3 and Figure 4 show the 1-step ahead and 48-steps ahead prediction results of the first period of data. The top 16 subplots correspond to 16 variables, and the bottom 4 subplots are the inputs. When performing the 48-steps ahead predictions, the model uses 11 real measurements from the past and 48 future control inputs (one per each step). After 48 timesteps, the model receives the real measurements from the past and uses the most recent 11 samples to predict forward. As such, the model does not integrate the incoming data in a continuous fashion. Moreover, the model is not retained using the incoming data. The 1-step ahead prediction and 48-steps ahead prediction results of the second period of data are shown in Figure 5 and Figure 6. Compared to the data from the first period, this part of the data is more challenging for the model to perform long-term predictions since the changes in control variables are more complex. In addition, the turbo exhaust pressure and condenser pressure variables are truncated at certain values, which makes them difficult to predict. Overall, from these results, it can be observed that the outputs from the model can follow the general trend in the data closely and show consistent responses to the changes in operating conditions.

To further interpret the model, Figure 7 shows the contribution of latent state dynamics and inputs to the predicted values. For each variable, the blue line represents the prediction contributions related to the latent space internal state dynamics using only the previous latent states without any input control. The red line represents the effect of new input control changes on the prediction. The final prediction is obtained by adding up these two values. The prediction results for the variables that are closely related to ambient temperature changes, such as condenser pressure and preheaters' temperatures, show that the model is able to learn the relationship between the variations in those variables with the changes in temperature. As a result, a large part of the predicted dynamics is represented by the input contributions. When the control variables undergo sudden changes, both latent state dynamics and input controls contribute to the predictions. This indicates that the model is able to make predictions by balancing the contributions of input changes and the latent states.

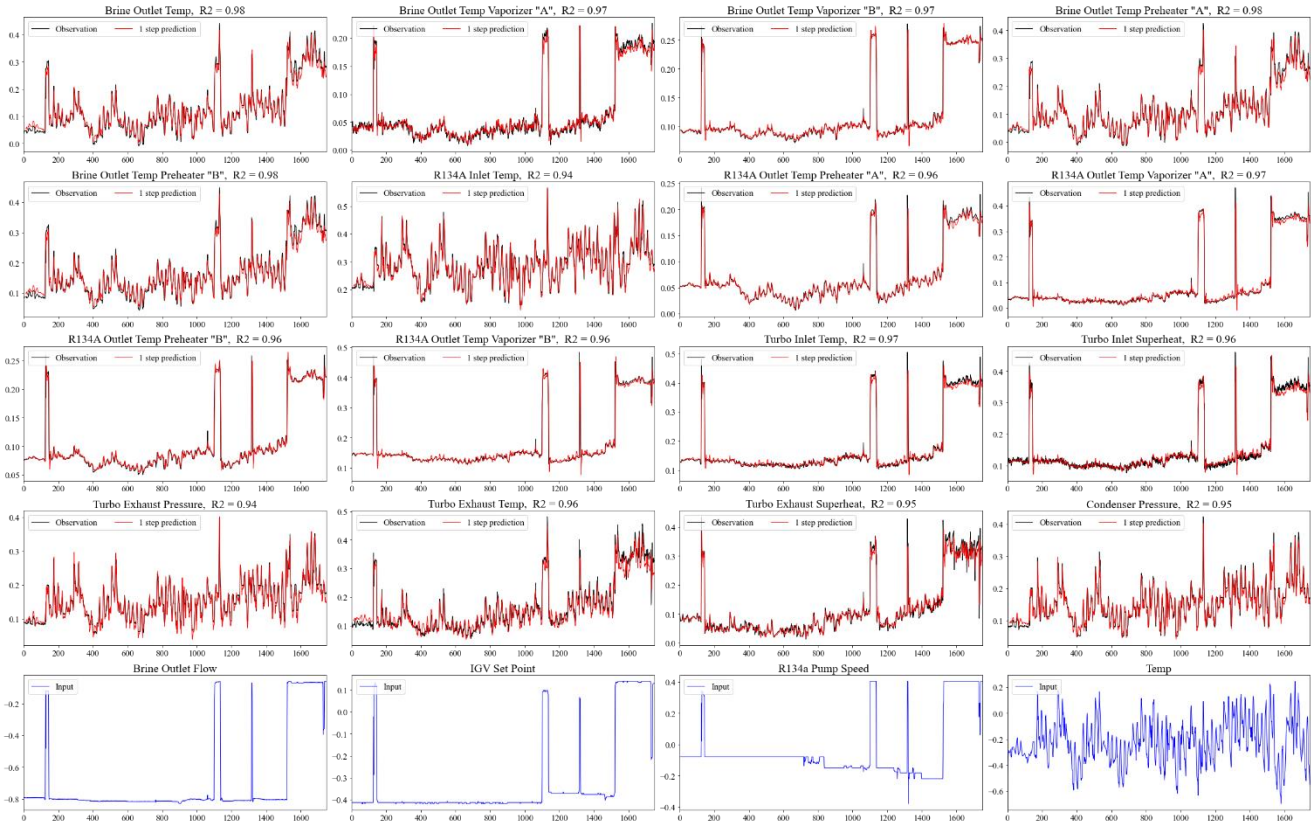

Figure 3: 1-step ahead prediction results of the first period of data.

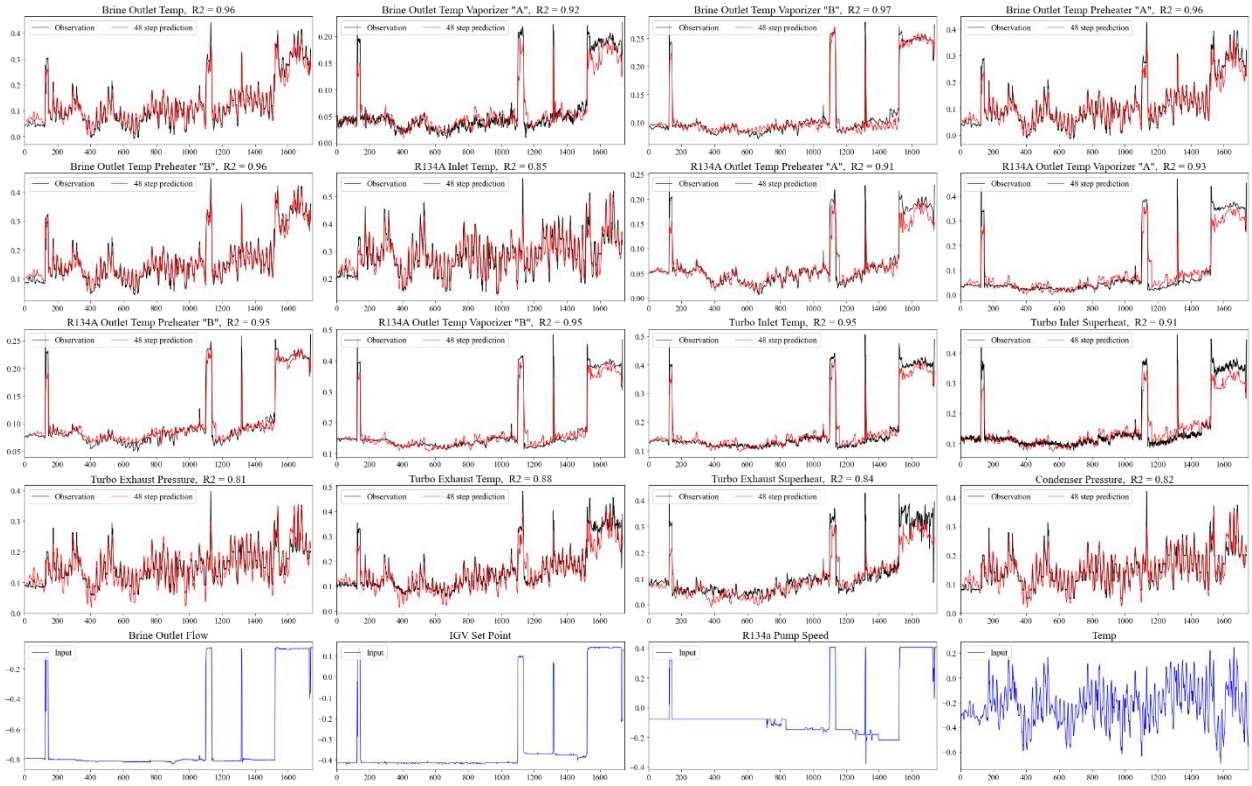

Figure 4: 48-steps ahead prediction results of the first period of data.

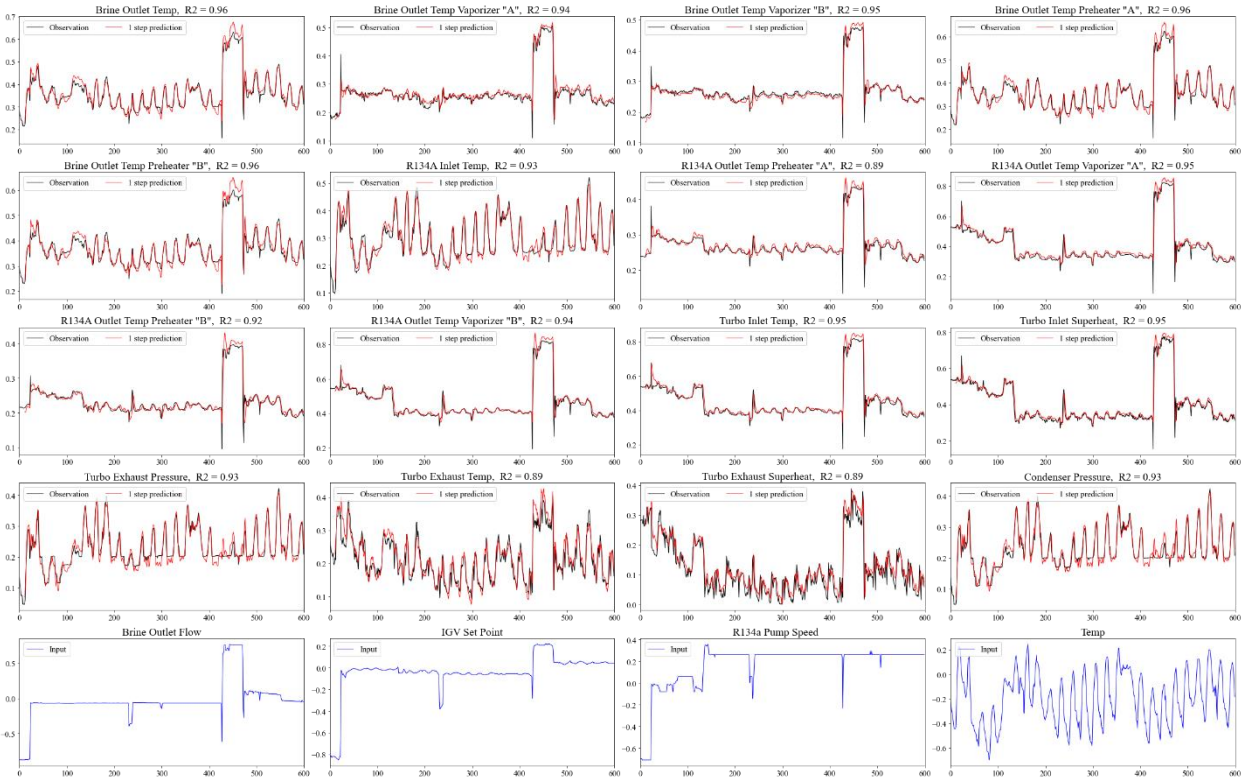

Figure 5: 1-step ahead prediction results of the second period of data.

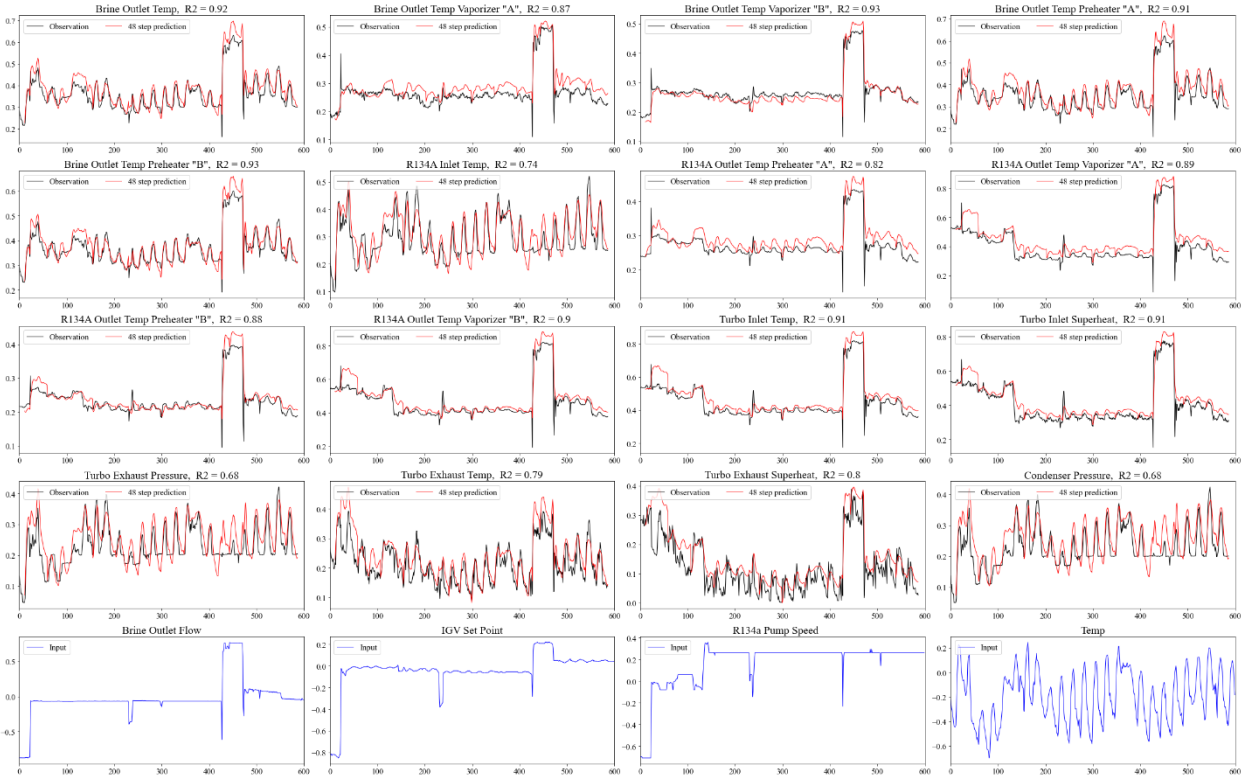

Figure 6: 48-steps ahead prediction results of the second period of data.

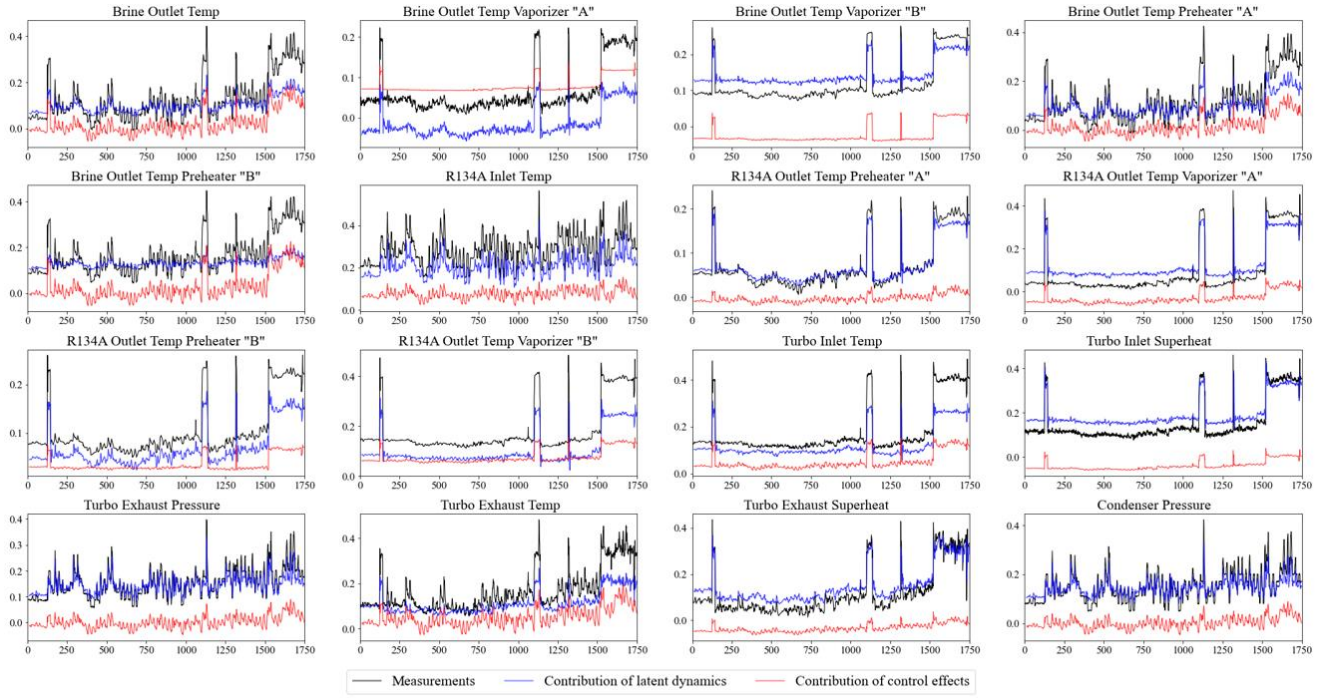

Figure 7: Contributions of latent dynamics and inputs changes to the prediction results.

3.3 Abnormal event detection for power plant data

Assuming that the trained dynamic neural network has a good predictive performance, the resulting prediction errors are expected to be small, and monitoring indices related to the prediction errors can be used to detect abnormal dynamics. As a result, any abnormal changes in new data will result in large prediction errors and can be detected using the monitoring indices. The neural network model's ability to capture the dynamics implies that most of the autocorrelation in the data is removed, and the prediction errors are likely to be static and uncorrelated. With this assumption, a widely used monitoring index based on multivariate analysis in Statistical Process Monitoring (SPM) can be applied. The Squared Prediction Error (SPE) Statistic is commonly used in combination with the Principal Component Analysis (PCA). The SPE index is defined as $\mathrm{SPE} = \|\tilde{\mathbf{x}}\|^2$ where $\tilde{\mathbf{x}}$ is the prediction error of a given sample. The sample is considered abnormal if its SPE index exceeds a specific control limit, which is calculated from the fitting results of training and validation datasets, that is

$$\mathrm{SPE} \leq \delta_\alpha^2$$

where δ_α^2 is the upper limit of SPE index with a confidence level α in χ^2 distribution, i.e.,

$$\delta_\alpha^2 = g\chi_{h;\alpha}^2 \quad \text{where} \quad g = \frac{\theta_2}{\theta_1} \quad , \quad h = \frac{\theta_1^2}{\theta_2}$$

The parameter θ_i is defined as $\theta_i = \sum_{j=l+1}^{m} \lambda_j^i$ in which l is the number of retained principal components. In our case, l equals 0.

The trained model is applied to selected data that is not in the training and validation datasets. The monitoring indices for the prediction errors and one control variable (IGV setpoint) are plotted in Figure 8. The horizontal red line in the top plot is the control limit calculated based on the model's prediction errors from training and validation data. The plot corresponding to the control variable IGV setpoint on the bottom is used to identify the period in which the power generation unit is in operation. When the setpoint is 0 the unit is shut down. If the unit is shut down, the corresponding monitoring indices are not valid for determining faults. Therefore, in the plot of SPE indices, the period in which the plant is shutdown is marked using gray color. By excluding the indices related to the shutdown period, it can be observed that most of the monitoring indices are below the control limit, which means the prediction results are close to the actual measurements, and there is no fault. However, a few samples are above the control limit and therefore indicate abnormal dynamics around 05/09/2020. The detected abnormal dynamics could result from faults in the system or just transient behaviors due to sudden control changes. Further analysis and validation with field fault logs are needed to verify these results.

Fault detection of chemical plant benchmark datasets

We test the performance of our fault detection algorithm with the proposed neural network architecture by applying it to a widely used benchmark dataset for process monitoring and fault detection, known as the TEP datasets. The benchmark dataset allows us to test the model's ability to capture normal dynamics and detect abnormal behavior. The TEP datasets consist of one normal dataset and several faulty datasets containing different types of faults. In our example, four types of faults are selected due to their similarity to the types of

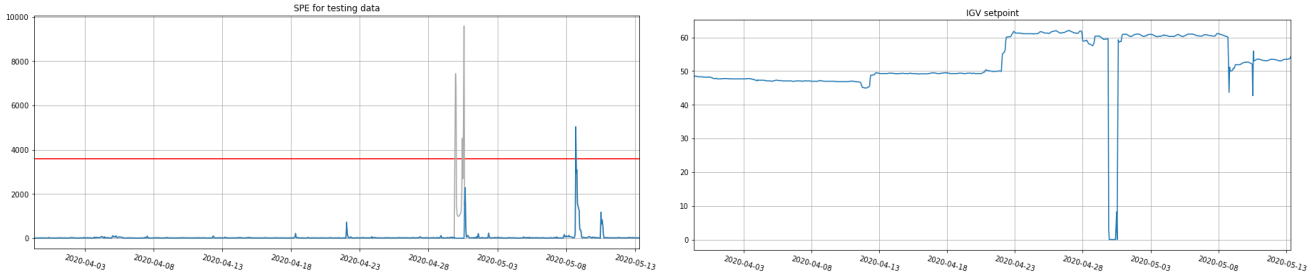

Figure 8: Monitoring indices and control changes.

faults that could occur in geothermal power plants. These include Fault 7: a step change of header pressure loss-reduced availability, Fault 11: random variation of reactor cooling water inlet temp, Fault 12: random variation of condenser cooling water inlet temp, and Fault 13: slow drift of reaction kinetics. To detect these faults, the dynamic neural network architecture is trained using the normal (fault-free) dataset, and a control limit is also established. The neural network model is then applied to the faulty datasets. The fault detection results for five realizations of each fault are plotted in Figure 9, where the vertical red line indicates when the faults occur, and the horizontal red line is the calculated control limit. It can be observed that for all four fault types, the SPE indices change dramatically and grow above the control limit after faults are introduced starting from the 480th data point. This result demonstrates the learning capability of the dynamic neural network model and the effectiveness of our proposed fault detection procedure.

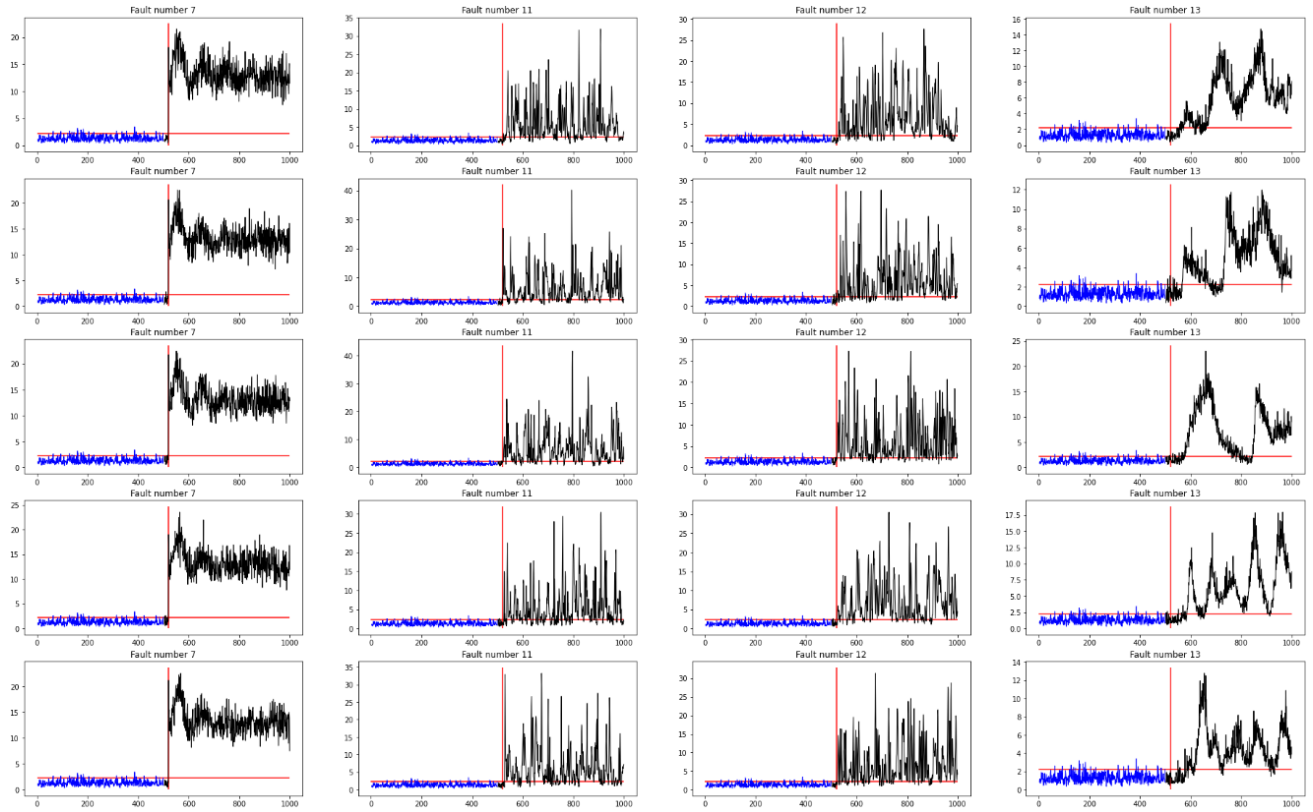

Figure 9: Fault detection results of TEP datasets.

4. CONCLUSION

We present a dynamic neural network architecture for performance prediction and real-time monitoring of geothermal power plants. The architecture is designed using an encoder-decoder model to enable dimension-reduced latent space representation in which the dynamics are learned. The method offers a general framework for designing and training a neural network model to predict geothermal power plants' dynamics and responses of variables due to change in control settings and ambient temperature. Some of the preliminary prediction results are demonstrated using different periods of testing data. The dynamic neural network model is applied to fault detection using a benchmark chemical plant dataset as well as real measurements from a geothermal power plant. In the latter case, the detection results need to be validated using the fault logs from the field. Additional enhancement of the model for long-term prediction and application to model predictive control, for optimization of power plant operations, may be possible with additional data and finetuning of important hyperparameters.

ACKNOWLEDGEMENTS

This material is based upon work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Geothermal Technologies Office award number DE-EE0008765.

REFERENCES

Manente, Giovanni, Andrea Toffolo, Andrea Lazzaretto, and Marco Paci: An Organic Rankine Cycle off-design model for the search of the optimal control strategy. Energy 58 (2013): 97-106.

Wang, Xing, Xiaomin Liu, and Chuhua Zhang.: Parametric optimization and range analysis of Organic Rankine Cycle for binary-cycle geothermal plant. Energy conversion and management 80 (2014): 256-265.

Arslan, Oguz, and Ozge Yetik. "ANN based optimization of supercritical ORC-Binary geothermal power plant: Simav case study." Applied Thermal Engineering 31, no. 17-18 (2011): 3922-3928.

Du, Pei, Jianzhou Wang, Wendong Yang, and Tong Niu.: Multi-step ahead forecasting in electrical power system using a hybrid forecasting system. Renewable Energy 122 (2018): 533-550.

Tugcu, Abtullah, and Oguz Arslan.: Optimization of geothermal energy aided absorption refrigeration system—GAARS: a novel ANN-based approach. Geothermics 65 (2017): 210-221.

Haklidir, Fusun S. Tut, and Mehmet Haklidir.: Prediction of reservoir temperatures using hydrogeochemical data, Western Anatolia geothermal systems (Turkey): a machine learning approach. Natural Resources Research 29, no. 4 (2020): 2333-2346.

Shi, Yu, Xianzhi Song, and Guofeng Song.: Productivity prediction of a multilateral-well geothermal system based on a long short-term memory and multi-layer perceptron combinational neural network. Applied Energy 282 (2021): 116046.