Deep Regression Network with Prediction Confidence in Time Series Application for Asset Health Estimation

Hao Huang, Arun Subramanian, Abhinav Saxena, Nurali Virani, Naresh Iyer
GE Global Research, Niskayuna, NY, USA
hao.huang1@ge.com, arun.subramanian1@ge.com, asaxena@ge.com, nurali.virani@ge.com, iyerna@ge.com

ABSTRACT

Numerous endeavors have been directed towards developing detection, monitoring, and prediction strategies for asset health estimation systems. Classic machine learning models have leveraged the use of physics-informed features derived from domain knowledge. However, this approach can be labor-intensive and constrained by the quality of features generated from existing knowledge. Furthermore, users often grapple with the challenge of determining the reliability of algorithmic predictions, particularly when the true prediction error remains undisclosed. In this study, we present a deep learning-based regression network that not only provides prediction values but also supplies confidence scores for asset health estimation, specifically in scenarios involving short intermittent transient time series. By doing so, our approach alleviates the need for cumbersome manual feature engineering. Through an in-depth experimental analysis, we showcase the model's proficiency in generating accurate predictions when confronted with short intermittent transient multivariate time series as input data. Notably, our model furnishes a confidence score for each prediction, exhibiting a robust negative correlation with the actual prediction error. Our experiments unveil that by setting an acceptance threshold for the confidence score, our model attains an average improvement of 20% in prediction quality with a coverage rate of 90%.

1. INTRODUCTION

The advancement of sensor technology has led to the accumulation of extensive data for monitoring physical assets, fostering a growing demand to ascertain the health status of assets across diverse industries. Accurate asset health estimation stands as a pivotal factor facilitating predictive maintenance strategies, which, in turn, bolster productivity, curtail maintenance expenses, and mitigate safety hazards (Liao & Ahn, 2016; Ellefsen, Æsøy, Ushakov, & Zhang, 2019). Typically, the determination of asset health involves framing the challenge as a regression or time series prediction problem, wherein the model endeavors to deduce continuous values that reflect the condition of the unit (Dong et al., 2010).

The domain of Artificial Intelligence (AI), particularly the realm of deep learning models, has already demonstrated its adaptability and triumphs in various aspects of asset health estimation (Zhang et al., 2019; Ellefsen et al., 2019; Rezaeianjouybari & Shang, 2020; Fink et al., 2020; Yucesan, Dourado, & Viana, 2021). Nevertheless, this progress is not exempt from challenges: trained models can yield erroneous predictions, especially when grappling with noisy datasets or extrapolating beyond familiar scenarios (de Bie, Lucic, & Haned, 2021). The conundrum persists regarding the methodology for identifying flawed algorithmic predictions in real-world production contexts, where the precise error in individual predictions often remains concealed (de Bie et al., 2021). Hence, the inclusion of prediction confidence estimates becomes imperative to heighten reliance and effectiveness of AI models in the sphere of asset health management. However, methods for gauging the reliability of predictions emanating from regression models, particularly when confronted with input of multivariate time series sensor data, have been relatively unexplored.

To bridge this existing gap, we present an innovative framework termed **D**eep **T**ime Series **R**egression with **C**onfidence Score (***DTRC***). Tailored to the context of multivariate time series, particularly short intermittent transient data harnessed from system sensors, *DTRC* not only furnishes projected target values but also furnishes a corresponding confidence score for each algorithmically derived prediction. This dual output aids users in discerning the reliability of our algorithmic predictions within production environments where the veritable error remains enigmatic.

In essence, our proposed *DTRC* encompasses an end-to-end deep learning regression approach. Central to this approach is our exploration of nonlinear strategies to extract enlightening embeddings from multivariate time series sensor data. Moreover, we capitalize on the distribution of the embedding space to gauge the dependability of individual predictions. The confidence score is derived from the density of the neighborhood

within the embedding space, operating under the premise that a dense neighborhood bolsters prediction credibility, whereas a sparse neighborhood engenders diminished confidence in the prediction. To heighten the interconnectedness within the embedding neighborhood, we construct normalized graphs using random walk techniques, applying these graphs to convolve the prediction space. This design choice renders each prediction pertinent not only to the input time series but also reliant on the neighbors inhabiting the embedding space. Distinctive attributes of our approach in contrast to standard deep time series regression models include the following:

1. **Pioneering Solution**: To our knowledge, this marks the inaugural end-to-end deep learning framework that furnishes a confidence score via the embedding space within the realm of time series regression for assessing asset health conditions.
2. **Novel Loss Function**: We introduce an original loss function that encompasses both prediction residuals and regularization within the embedding space. This loss function is crafted to steer samples within a proximate neighborhood in the embedding space towards generating prediction values that are closely aligned.
3. **Empirical Superiority**: We substantiate our proposed approach's efficacy through empirical evidence, showcasing its superiority over prevalent deep learning baselines in terms of both prediction accuracy and confidence estimation. Additionally, by instituting an acceptance threshold for the confidence score, our model yields an average enhancement of over 20% in prediction quality with a coverage rate of 90%.

2. RELATED WORK

Deep learning models have gained substantial traction in asset health estimation and decision-making applications. However, the efficacy of well-trained models can falter during inference due to noise and disparities between training and inference data distributions. Hence, evaluating the reliability and effectiveness of deep learning models before practical deployment has become increasingly critical (Abdar et al., 2021). Thus, the inclusion of not only prediction outputs but also confidence scores for individual predictions within any deep learning model aimed at asset health estimation is strongly desirable.

Within classical uncertainty quantification, two prominent concepts exist: prediction intervals, which forecast the range within which a future individual observation may lie, and confidence intervals, which delineate the likely range of values pertaining to statistical parameters of the data, such as population mean (Finch & Cumming, 2009). However, neither of these concepts straightforwardly addresses the confidence level associated with each individual prediction output in regression problem.

Reliability assessment for individual prediction samples is crucial. Several methods have been introduced to integrate reliability quantification into machine learning models (Virani, Iyer, & Yang, 2020; Bhushan, Yang, Virani, & Iyer, 2020; Iyer, Virani, Yang, & Saxena, 2022). The incorporation of confidence scores in regression for individual predictions is achievable through techniques like bagging, with tree-based approaches such as random forests serving as prominent examples. However, these estimates have exhibited bias. Wager et al. (Wager, Hastie, & Efron, 2014) introduced two procedures that yield more efficient and less biased confidence scores. This was achieved through bias-corrected adaptations of the jackknife-after-bootstrap and infinitesimal jackknife methods. Subsequent simulation studies, introduced by (Brokamp, Rao, Ryan, & Jandarov, 2017), underscored the efficacy of the infinitesimal jackknife estimator in accurately gauging prediction errors, especially when conditional inference trees are employed to construct random forests. Briesemeister et al. (Briesemeister, Rahnenführer, & Kohlbacher, 2012) proposed methods that estimate prediction reliability based on the local characteristics of nearby training data, applicable to both linear and nonlinear regression models with minimal computational overhead. Jiang et al. (Jiang, Kim, Guan, & Gupta, 2018) deemed a prediction trustworthy if it aligned with the training data's behavior, a concept primarily suited for classification but not regression problems. Building on this notion, deBie et al. (de Bie et al., 2021) introduced a confidence measurement for regression models, aiding in assessing prediction trustworthiness in the absence of ground-truth error. More recently, Ghobrial et al. (Ghobrial, Asgari, & Eder, 2023) introduced a methodology for endowing confidence scores in convolutional neural network (CNN) predictions. This metric quantifies prediction trustworthiness by assessing the presence of certain features within CNN-made predictions, serving the dual purpose of measuring trustworthiness and detecting suspicious predictions. Regrettably, none of these methods can be directly transposed onto deep learning regression models for time series data.

In this study, we propose an end-to-end deep learning regression model for industrial asset health estimation, operating on multivariate time series inputs. Significantly, this model offers individual prediction confidence scores. Empirical results underscore the model's capacity to enhance prediction quality by over 20%, with a coverage rate of 90%, through the utilization of confidence scores.

3. PROBLEM SETTING

Multivariate time series data collected by high-frequency sensors are ubiquitous in modern industrial systems. A time series dataset encompassing m variables and ℓ timestamps is denoted as $X \in \mathbb{R}^{n \times m \times \ell}$. Here, n signifies the count of time series instances within the dataset. It's worth noting that, for the sake of notation simplicity, we assume uniform time se-

ries lengths. However, this model's versatility allows for accommodating time series of disparate lengths through techniques like padding or resampling.

Central to the paradigm of time series regression for asset health estimation is the quest to establish an optimal (potentially nonlinear) mapping from historical time series to a continuous regression target. This target typically encapsulates system status or production indicators. Simultaneously, we endeavor to assess prediction errors by furnishing a confidence score for each prediction output. This dual pursuit encompasses the essence of our approach. We formally present our problem setup, encapsulating two primary objectives:

1. **Time Series Regression**: Given a collection of training time series instances, denoted as $\langle x, y \rangle$, where each time series $x = (x_1, x_2, ..., x_\ell)$ is composed of individual timestamps $x_i \in \mathbb{R}^m$, and $y \in \mathbb{R}^1$ represents the associated regression target, we propose a sequence modeling approach. This model is designed to establish a nonlinear mapping from input time series to the regression target, as expressed by the equation:

$$\hat{y} = f(x_1, x_2, ..., x_\ell). \tag{1}$$

2. **Confidence Score Provision**: Beyond prediction, our model's scope extends to generating a confidence score $s \in \mathbb{R}^1$ for each prediction $\hat{y}$. In the context of a dataset X containing n time series instances and an unknown regression target $Y \in \mathbb{R}^n$, the model yields both prediction values $\hat{Y} \in \mathbb{R}^n$ and corresponding confidence scores $S \in \mathbb{R}^n$. Of note, the values in S are intrinsically engineered to exhibit a strong negative correlation with the unknown prediction errors, characterized by the equation:

$$E = |Y - \hat{Y}|. \tag{2}$$

   The ultimate aspiration is that the correlation between E and S approaches the ideal value of -1.

4. MODEL ARCHITECTURE

In this paper, we present an innovative temporal regression model that leverages multivariate time series as input to accomplish two main objectives: 1) predict the target variable, and 2) generate a confidence score for each prediction. We denote this framework as **D**eep **T**ime Series **R**egression with **C**onfidence Score, abbreviated as ***DTRC***. The architectural layout of our model is depicted in Figure 1. It comprises two core components: a temporal convolutional network (*TCN*) and a neighbor convolutional network (*NCN*).

The *TCN* is responsible for processing the input time series data and generating an embedding vector for each respective time series. Subsequently, the *NCN* steps in to convolve neighborhood information present within the embedding space, culminating in the derivation of the final prediction. This dual-component structure, combining *TCN* and *NCN*, underpins the learning framework of our proposed model, *DTRC*.

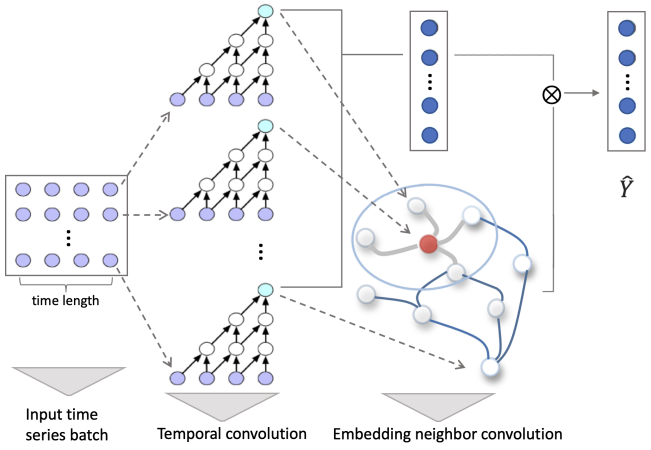


Figure 1. Illustration of our *DTRC* model framework. It encompasses a temporal convolutional network and a neighbor convolutional network. The former transforms time series data into embedding vectors, while the latter convolves these embeddings using a random-walk normalized matrix to yield the ultimate regression targets.

4.1. Temporal Convolutional Network

The initial component of our model is a temporal convolutional network (*TCN*) that adeptly extracts non-linearly transformed features from the input time series.

The concept of *TCN* was originally introduced in (Oord et al., 2016) and has gained prominence across diverse sequence modeling tasks (Bai, Kolter, & Koltun, 2018; Franceschi, Dieuleveut, & Jaggi, 2019). A noteworthy distinction from recurrent neural networks is that *TCN* does not employ a recursive structure, thereby mitigating the challenges of gradient vanishing (Bai et al., 2018). To cater to extended temporal dependencies and facilitate the acquisition of high-level features, *TCN* often integrates multiple levels. In our specific implementation (depicted in Figure 2), we adopt a multi-level *TCN* architecture for processing the input time series. In this configuration, the input for each subsequent level stems from the output of the preceding level. The culmination of this arrangement results in the last *TCN* level yielding an embedding vector that encapsulates comprehensive high-level insights spanning the entire input time series.

Illustrating the mechanics in greater detail, each *TCN* level within our design encompasses five core stages: a temporal (*1D*) convolutional filter, Batch Normalization, a Rectified Linear Unit (ReLU), a dropout operation, and another ReLU applied to the summation of newly derived features and the original input. This structural arrangement integrates

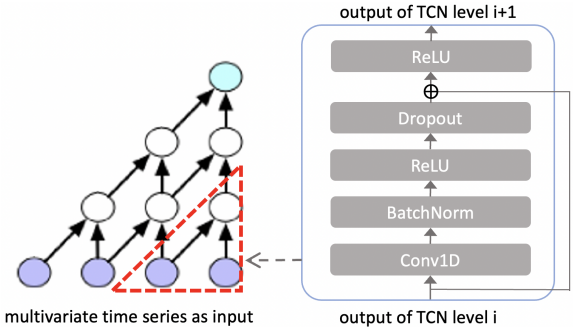


Figure 2. Module 1 encompasses a temporal convolutional network with skip connections. Its primary goal is to extract an embedding vector from each input time series.

skip connections (or residual connections) (He, Zhang, Ren, & Sun, 2016) to address performance degradation and ensure the reusability of shallow features (Wang, Cao, Wang, & Zaiane, 2022; Cao et al., 2023).

4.2. Neighbor Convolutional Network

The second component of our model involves convolving each time series embedding with its neighboring embeddings in the embedding space to generate the final prediction. This component is realized through a neighbor convolutional network (*NCN*). This module receives inputs from the preceding *TCN*, consisting of a batch of embedding vectors denoted as $V \in \mathbb{R}^{b\times d}$, where b signifies the number of time series instances within the batch, and d corresponds to the hidden dimensions derived from the final layer of the *TCN*.

As depicted in Figure 3, the *NCN* module employs a dual-path design. The upper path involves a series of fully connected layers aimed at projecting the time series embeddings $V \in \mathbb{R}^{b\times d}$ into the target space, denoted as $\tilde{Y} \in \mathbb{R}^{b\times 1}$. This resultant $\tilde{Y}$ is referred to as the unsmoothed prediction.

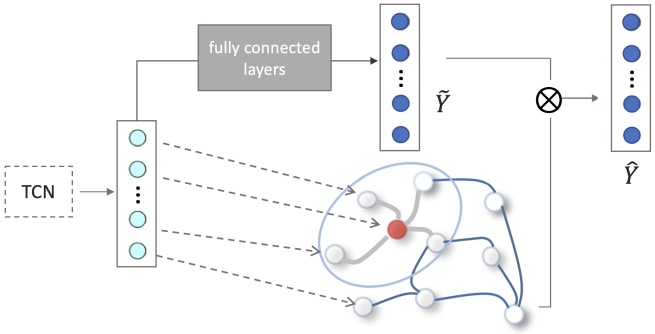


Figure 3. Module 2 involves a neighbor convolutional network designed to convolve each prediction with the neighboring embeddings and subsequently generate the ultimate prediction.

In the lower path, a Gaussian kernel is applied to the batch of embedding vectors to construct a similarity graph matrix $A \in \mathbb{R}^{b\times b}$. Precisely, each edge in this graph is determined by the Gaussian kernel computed on the l_1 distance (Manhattan distance) between embeddings:

$$A_{ij} = exp(-|V_i - V_j|_1/2\sigma^2). \tag{3}$$

However, the effectiveness of Equation (3) hinges on the scaling parameter σ. To enhance robustness, we adopt a neighbor adaptive scale (Zelnik-Manor & Perona, 2004; Yang, Gong, & U, 2011) that accounts for the local structure around each embedding point. The local scaling σ for each embedding is defined as the mean distance from the embedding to all other points.

Subsequently, random walk normalization is employed to normalize matrix A:

$$W = D^{-1}A, \tag{4}$$

where D represents the degree matrix of A (diagonal matrix of vertex degrees). Matrix W serves as the transition matrix for random walk on graph matrix A.

The normalized transition matrix W is then multiplied with the unsmoothed prediction $\tilde{Y}$ to yield the final prediction, which is smoothed by neighborhood information. This can be mathematically expressed as:

$$\hat{Y} = W\tilde{Y}. \tag{5}$$

This process can be conceptualized as a one-step diffusion map: if we consider the unsmoothed prediction $\tilde{Y}$ as $Y(t_0)$, a preliminary prediction of final targets at time t_0, performing a one-step diffusion with random walk yields $Y(t_1) = WY(t_0)$. This $Y(t_1)$ is our ultimate prediction $\hat{Y}$.

4.3. Loss function and Regularization

The final residual is derived using the following equation:

$$loss = \|Y - \hat{Y}\| + \|U_V/d - U_Y\|. \tag{6}$$

In this equation, U_V and U_Y represent the l_1 distance (Manhattan distance) matrix of the time series embedding V and the ground truth target Y, respectively. Here, d corresponds to the number of dimensions in the embedding space.

Equation (6) comprises two distinct components:

1. The first part encapsulates the Mean Squared Error (*MSE*) between the actual ground truth values Y and the predicted values $\hat{Y}$.
2. The second part constitutes a regularization term applied to the embedding space. This regularization enforces congruence in the geometric relationships within the embedding space and those present in the target space.

Together, these components amalgamate to formulate the

complete loss function, which serves as the foundation for guiding the model's learning process.

4.4. Confidence Score

Our confidence score for a prediction is quantified by evaluating the neighborhood density of the corresponding embedding within the (training) embedding space. Specifically, when denoting the embedding of a time series instance from the inference phase as V_i, we measure the inverse of the summation of distances to its k nearest neighbors (*KNN*) within the training embedding set:

$$S_i = 1/\sum_{j \in \text{KNN}_i} |V_i - V_j|_1. \tag{7}$$

Here, V_j are selected from the training embedding vectors through *KNN* neighborhood based on the Manhattan distance.

Our approach to measuring the confidence score is rooted in the following intuition: When an input time series yields a dependable prediction, it's because the corresponding embedding in the *TCN* space has a dense neighborhood that effectively supports its prediction. Consequently, such instances receive higher confidence scores. Conversely, if the time series embedding possesses a sparse neighborhood, it indicates that the model hasn't encountered similar time series during the training phase. As a result, its prediction is deemed less reliable.

In practice, we normalize the confidence scores to a range of [0, 1] using the minimum and maximum confidence scores from the training set.

5. EXPERIMENT

Dataset. We assessed our regression model using a simulated dataset called the Fine Motion Control Rod Drive (*FMCRD*) dataset (GE-Research & University of Tennessee, 2023). This dataset captures inputs, outputs, and annotations from a Simulink-based servomotor simulator that replicates intermittent servomotor operation, particularly the FMCRD mechanism in nuclear reactors. Intermittent drives like FMCRD experience wear and damage due to various factors, impacting rotor shaft movement differently than continuous machinery. This dataset fills the gap for data and algorithms in this scenario, shedding light on transient operational patterns and distinct degradation modes. We simulate cumulative damage by introducing an opposing load on the motor shaft, influencing observable signals. The dataset characterizes degradation assessment through load variations sampled from a mixture of Gaussians with four modes.

The core objective of this assessment was to predict the load value, a scalar metric, based on an input time series that comprises 7 channels: motor torque, rotor speed, actual rod position, the position delta, and stator current of phase A, B and C. The target load values span a range from 3 to 5012.

The dataset is divided into two segments: a training set and a testing set. The training set encompasses 600 runs, each encompassing 5 transitions (time series instances) characterized by varying lengths that range from 400 to 800 timestamps. Similarly, the testing set consists of an equivalent number of runs and transitions, also with varied lengths ranging from 400 to 800 timestamps. To ensure uniformity, we employed resampling techniques to transform all time series instances into a standardized length of 200 timestamps.

Evaluation Metrics. To gauge prediction accuracy, we employ the Mean Absolute Error (*MAE*) metric:

$$\textit{MAE} = \frac{1}{n}\sum_{i=1}^{n} |\hat{Y}_i - Y_i|. \tag{8}$$

Here, $\hat{Y}$ and Y represent the predicted and actual load values, and n signifies the total number of time series instances. The range of *MAE* can span from 0 to ∞, with smaller values indicating higher prediction quality.

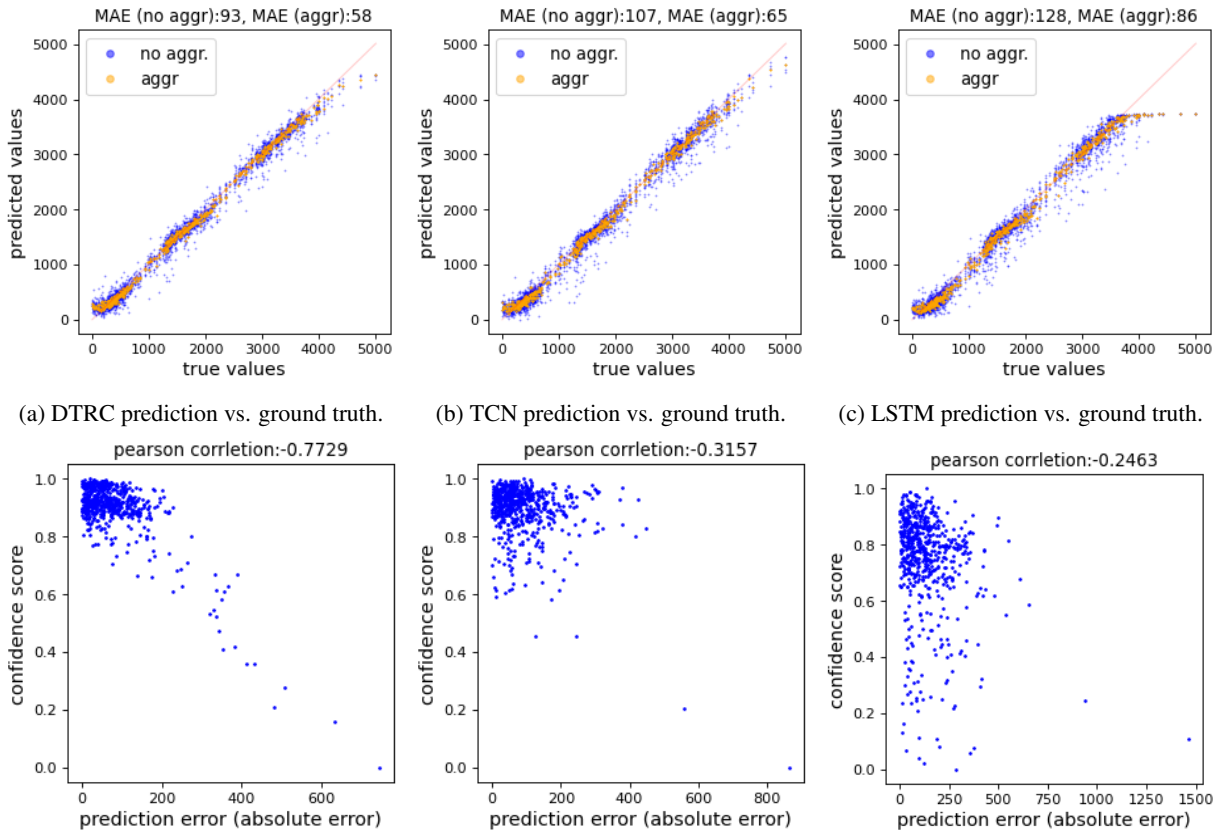
For evaluating the performance of our confidence scores in relation to prediction errors, we employ the Pearson Correlation:

$$\textit{Pearson Correlation} = \frac{\sum(S_i - \bar{S})(E_i - \bar{E})}{\sqrt{\sum(S_i - \bar{S})^2 \sum(E_i - \bar{E})^2}}. \tag{9}$$

In this equation, S and E represent the confidence score (calculated using Equation (7)) and prediction error (computed using Equation (2)), respectively. The symbols $\bar{*}$ denote average values. The range of Pearson Correlation lies between -1 and 1, with a value close to -1 indicating that the confidence scores effectively estimate the unknown prediction error.

Model Architecture and Baselines. The configuration of our implemented *DTRC* structure for the *FMCRD* experiment is outlined as follows: The *TCN* module comprises three levels, each with an output channel count of 100. The kernel sizes are set at 10 with a stride of 2. Towards the culmination of the *TCN* module, we incorporate a max-pooling layer along the temporal axis. The fully connected chain within Module 2 consists of 2 layers, the first layer generating 100 channels, and the second layer resulting in a single channel corresponding to the target dimension. We set $k = 20$ in Equation (7) to define the neighborhood size for quantifying the confidence score.

Our evaluation involves a comparison between our *DTRC* model and two widely-used deep learning baselines: the temporal convolutional network (*TCN*) and long short-term memory networks (*LSTM*). The *TCN* baseline shares the same structural framework as the initial segment of our *DTRC*, with the addition of an extra layer (transitioning from



(a) DTRC prediction vs. ground truth. (b) TCN prediction vs. ground truth. (c) LSTM prediction vs. ground truth.

(d) DTRC confidence scores vs. errors. (e) TCN confidence scores vs. errors. (f) LSTM confidence scores vs. errors.

Figure 4. Regression performance and confidence scores demonstrated by various algorithms. Notably, our *DTRC* exhibits superior prediction quality alongside more precise confidence scores that effectively estimate prediction errors.

100 to 1 channel) at the end, responsible for projecting time series embeddings into the target space. Similarly, for the *LSTM* baseline, we employ an additional layer after three levels of *LSTM*, each with an output channel size of 100. Both baselines also employ the same method as our *DTRC* to compute confidence scores based on their time series embeddings.

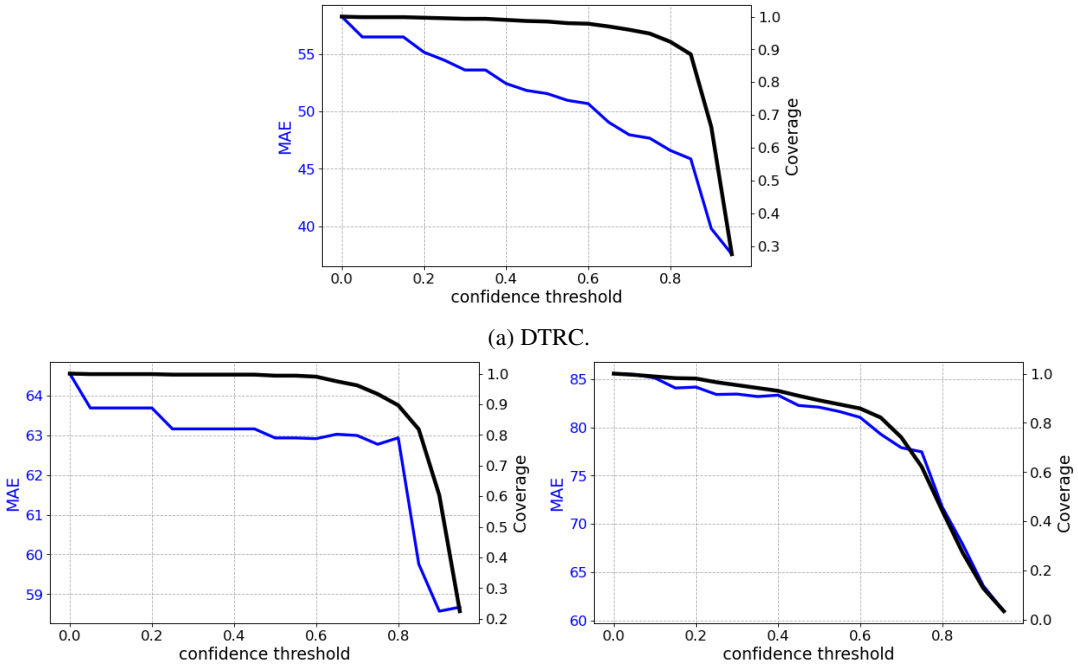
Analysis of Prediction Results. Figure 4a illustrates the prediction performance of our *DTRC* model on the testing set as compared to the ground truth. The blue points correspond to the raw predictions, whereas the orange points represent post-processed predictions achieved by applying aggregation (averaging) on time series transitions that belong to the same run. Our aggregated predictions showcase an *MAE* of 58, signifying an improvement of over 37% compared to the raw predictions.

Similarly, in Figure 4b, we depict the *MAE* for both raw and aggregated predictions generated by the *TCN* model, and in Figure 4c, we display the corresponding *MAE* results for the *LSTM* model. The following observations can be made:

1. Aggregated predictions consistently outperform raw predictions for all three methods.
2. Our *DTRC* model showcases an *MAE* that is approximately 11% superior to that of the *TCN* model, and around 33% better than the *LSTM* model in terms of *MAE*. This notably highlights the positive impact of the neighbor convolutional network (*NCN*), which constitutes the second part of our proposed model, as well as our novel loss function that bridges prediction and the embedding manifold.

Confidence Score Analysis. In Figure 4d, the depicted graph illustrates the confidence scores and the absolute prediction errors for each prediction generated by our *DTRC* model. Notably, our confidence scores exhibit a strong negative correlation with the absolute prediction errors. The Pearson Correlation coefficient between our confidence scores and prediction errors stands at -0.7729, underscoring the robustness of this relationship. This correlation coefficient is substantially higher than that observed for the *TCN* model (-0.3157) in Figure 4e, as well as the *LSTM* model (-0.2463) in Figure 4f.

Enhancing Prediction Quality Using Confidence Scores. In practical applications, one approach involves setting a



(a) DTRC.

(b) TCN.

(c) LSTM.

Figure 5. Mean Absolute Error (*MAE*) plotted against Coverage for various confidence thresholds. As the confidence threshold increases, the prediction error of our *DTRC* exhibits a noteworthy decline, while maintaining substantial coverage when the threshold remains ≤ 0.85.

threshold on the confidence score and only accepting predictions that surpass this threshold. Figure 5 graphically presents the impact of different confidence thresholds on prediction quality measured by *MAE*. Additionally, the coverage (the proportion of accepted predictions within the entire dataset) is taken into consideration. Our observations are as follows:

1. In comparison to *LSTM*, both our *DTRC* and *TCN* models maintain higher coverage levels when confidence thresholds are set below 0.9. Notably, our *DTRC* even exhibits slightly higher coverage than *TCN*.
2. With increasing confidence thresholds, the prediction error of our *DTRC* model experiences a marked decrease. At a confidence threshold of 0.85, our *DTRC* model achieves an average improvement of 20% in prediction quality, while maintaining 90% coverage. In contrast, both the *TCN* and *LSTM* models display slower decreases in *MAE*. This reaffirms that our confidence scores are highly advantageous in real-world scenarios, effectively enhancing the reliability and predictive performance of our model.

6. CONCLUSION

In this paper, we introduce a novel approach named **D**eep **T**ime Series **R**egression with **C**onfidence Score (***DTRC***). Operating on multivariate time series data, particularly short intermittent transient data collected from system sensors, *DTRC* offers both predicted target values and corresponding confidence scores. This feature empowers users to assess the reliability of algorithmic predictions even when the true error remains elusive. A central aspect of our model involves an embedding neighbor convolution enhanced by random-walk normalized graphs. The confidence score is derived from the density of neighboring time series inputs within the embedding space. Empirical results demonstrate that by applying an acceptance threshold to the confidence score, our model can achieve an average prediction quality improvement of 20% while maintaining 90% coverage.

7. ACKNOWLEDGEMENT

Research funding for this work was provided by the Advanced Research Projects Agency-Energy (ARPA-E), U.S. Department of Energy, under Award Number DE-AR0001290, with the aim of advancing the use of AI to reduce O&M costs for nuclear power plants. The views and opinions expressed by the authors do not necessarily represent those of the United States Government or any of its agencies. This work is part of GE's Humble AI initiative.

REFERENCES

Abdar, M., Pourpanah, F., Hussain, S., Rezazadegan, D., Liu, L., Ghavamzadeh, M., ... others (2021). A review

of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, *76*, 243–297.

Bai, S., Kolter, J. Z., & Koltun, V. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*.

Bhushan, C., Yang, Z., Virani, N., & Iyer, N. (2020). Variational encoder-based reliable classification. In *2020 ieee international conference on image processing (icip)* (pp. 1941–1945).

Briesemeister, S., Rahnenführer, J., & Kohlbacher, O. (2012). No longer confidential: estimating the confidence of individual regression predictions. *PloS one*, *7*(11), e48723.

Brokamp, C., Rao, M., Ryan, P., & Jandarov, R. (2017). A comparison of resampling and recursive partitioning methods in random forest for estimating the asymptotic variance using the infinitesimal jackknife. *stat*, *6*(1), 360–372.

Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., Tian, Q., & Wang, M. (2023). Swin-unet: Unet-like pure transformer for medical image segmentation. In *Computer vision–eccv 2022 workshops: Tel aviv, israel, october 23–27, 2022, proceedings, part iii* (pp. 205–218).

de Bie, K., Lucic, A., & Haned, H. (2021). To trust or not to trust a regressor: Estimating and explaining trustworthiness of regression predictions. *arXiv preprint arXiv:2104.06982*.

Dong, M., et al. (2010). A tutorial on nonlinear time-series data mining in engineering asset health and reliability prediction: concepts, models, and algorithms. *Mathematical Problems in Engineering*, *2010*.

Ellefsen, A. L., Æsøy, V., Ushakov, S., & Zhang, H. (2019). A comprehensive survey of prognostics and health management based on deep learning for autonomous ships. *IEEE Transactions on Reliability*, *68*(2), 720–740.

Finch, S., & Cumming, G. (2009). Putting research in context: Understanding confidence intervals from one or more studies. *Journal of Pediatric Psychology*, *34*(9), 903–916.

Fink, O., Wang, Q., Svensen, M., Dersin, P., Lee, W.-J., & Ducoffe, M. (2020). Potential, challenges and future directions for deep learning in prognostics and health management applications. *Engineering Applications of Artificial Intelligence*, *92*, 103678.

Franceschi, J.-Y., Dieuleveut, A., & Jaggi, M. (2019). Unsupervised scalable representation learning for multivariate time series. *arXiv preprint arXiv:1901.10738*.

GE-Research, & University of Tennessee, U., Knoxville. (2023). *Servomotor-driven ballscrew mechanism degradation data set.* Retrieved from `https://data.phmsociety.org/servomotor_dataset/`

Ghobrial, A., Asgari, H., & Eder, K. (2023). Towards a measure of trustworthiness to evaluate cnns during operation. *arXiv preprint arXiv:2301.08839*.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 770–778).

Iyer, N., Virani, N., Yang, Z., & Saxena, A. (2022). Mixed initiative approach for reliable tagging of maintenance records with machine learning. In *Annual conference of the phm society* (Vol. 14).

Jiang, H., Kim, B., Guan, M., & Gupta, M. (2018). To trust or not to trust a classifier. *Advances in neural information processing systems*, *31*.

Liao, L., & Ahn, H.-i. (2016). Combining deep learning and survival analysis for asset health management. *International Journal of Prognostics and Health Management*, *7*(4).

Oord, A. v. d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., ... Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

Rezaeianjouybari, B., & Shang, Y. (2020). Deep learning for prognostics and health management: State of the art, challenges, and opportunities. *Measurement*, *163*, 107929.

Virani, N., Iyer, N., & Yang, Z. (2020). Justification-based reliability in machine learning. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 34, pp. 6078–6085).

Wager, S., Hastie, T., & Efron, B. (2014). Confidence intervals for random forests: The jackknife and the infinitesimal jackknife. *The Journal of Machine Learning Research*, *15*(1), 1625–1651.

Wang, H., Cao, P., Wang, J., & Zaiane, O. R. (2022). Uctransnet: rethinking the skip connections in u-net from a channel-wise perspective with transformer. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 36, pp. 2441–2449).

Yang, Y., Gong, Z., & U, L. H. (2011). Identifying points of interest by self-tuning clustering. In *Proceedings of the 34th international acm sigir conference* (pp. 883–892).

Yucesan, Y. A., Dourado, A., & Viana, F. A. (2021). A survey of modeling for prognosis and health management of industrial equipment. *Advanced Engineering Informatics*, *50*, 101404.

Zelnik-Manor, L., & Perona, P. (2004). Self-tuning spectral clustering. *Advances in neural information processing systems*, *17*.

Zhang, L., Lin, J., Liu, B., Zhang, Z., Yan, X., & Wei, M. (2019). A review on deep learning applications in prognostics and health management. *Ieee Access*, *7*, 162415–162438.