Interpretable Industrial Soft Sensor Design Based on Informer and SHAP

Liang Cao * **Xiaolu Ji** ** **Yankai Cao** * **Yi Luo** *** **Yixiu Wang** *
Lim C. Siang **** **Jin Li** **** **R. Bhushan Gopaluni** *

* *Department of Chemical and Biological Engineering, University of British Columbia, Vancouver, BC, V6T 1Z3, Canada (e-mail: bhushan.gopaluni@ubc.ca)*
** *Department of Statistics, University of British Columbia, Vancouver, BC, V6T 1Z4, Canada*
*** *Research Institute of Mine Big Data, Chinese Institute of Coal Science, 10013, Beijing, China*
**** *Department of Process Control Engineering, Burnaby Refinery, Burnaby, BC, V5C 1L7, Canada*

Abstract: Deep learning models have been widely employed in various domains, yet they have certain limitations when it comes to industrial process applications. The two main challenges are their inability to effectively handle long-sequence predictions and the complexity of their internal structure, which makes it difficult to explain the output of the model. This work aims to build accurate and interpretable soft sensors for industrial processes. The Informer model is used to build accurate soft sensors due to its proficiency in long sequences. Additionally, an interpretable machine learning algorithm, SHapley Additive exPlanations (SHAP), is used to infer the global and local contributions of each feature to the predictions. The effectiveness of the proposed algorithms is validated on real industrial fluid catalytic cracker unit data, and the results show that the Informer model has higher accuracy and better long-sequence data prediction ability. Furthermore, the SHAP analysis enhances the model's utility by providing clear insights into the influence of individual features on the predictions, thereby increasing its transparency and trustworthiness in industrial settings.

Keywords: Soft Sensors, Informer,SHAP, Time series forecasting, Interpretability.

1. INTRODUCTION

With stringent requirements for product quality and cost, the complexity of automation of industrial processes is constantly increasing. As the scale of plants grows, it is urgent to improve the safety and stability of the process, as any accidental situation in industrial processes can lead to disastrous consequences, such as serious casualties, economic losses, and environmental pollution. Therefore, in industrial processes, it is vital to monitor critical variables that are closely related to process safety and economic benefits. These critical variables are called quality variables. However, some quality variables are difficult or costly to measure in real-time, posing a significant challenge for real-time process monitoring. To overcome this challenge, soft sensor technology has been introduced(Yu et al., 2020; Qin, 2014). The basic idea of a soft sensor is to select easily measurable process variables to construct a mathematical relationship that can estimate the values of quality variables.

Modern industrial processes are often highly complex, characterized by multilevel, high-dimensional, strong coupling, and high nonlinearity. Therefore, the search for soft sensor models capable of accurately describing and predicting these intricate industrial processes has always been a focal point in both the industrial and academic sectors. During the last decades, with the accumulation of industrial data(Fan et al., 2014; Qin, 2014), the enhancement of computational capabilities(Zhu et al., 2021), and the advancement of machine learning theories(Gopaluni et al., 2020), soft sensor models have achieved significant breakthroughs.

With the richness of process data and the rapid development of machine learning techniques, data-driven soft sensor technologies are increasingly favored. Although soft sensor models have great potential and value in industrial applications, they still face significant challenges, particularly in the areas of time-series prediction and model interpretability.

Soft sensors for time-series data have become a crucial tool in industrial process management, primarily due to their ability to forecast future conditions and trends in critical quality variables. By providing detailed insights into potential future states of the system, these sensors empower engineers to gauge and fine-tune production processes preemptively. This proactive approach facilitates not only process optimization but also a deeper understanding of production status and reliability assessment. With the booming development of big data and computing power, deep learning-based models have been widely used in various fields. Compared with traditional statistical models and machine learning models, deep learning models can extract patterns and relationships in vast datasets and often have more accurate results.

While deep learning models have revolutionized time-series forecasting in industrial applications, their effectiveness varies due to the complex nature of modern industrial processes (LeCun, 2015). Recurrent Neural Networks (RNNs) introduced the concept of memory to neural networks, enhancing their ability to process sequences. However, RNNs struggle with long-term dependencies and are hindered by slow training times. Long Short-Term Memory (LSTM) networks addressed this by incorporating an additional state in the RNN structure, enabling them to capture long-term dependencies more effectively. Despite this improvement, LSTMs can be computationally demanding, especially when dealing with large-scale, rapidly arriving, and long-sequence data. The Transformer model offers a solution to some of these challenges with its self-attention mechanism and the ability to process sequences in parallel. This design improves efficiency in the handling of complex data structures. However, transformers tend to underperform in capturing long-term dependencies because of the fixed size of their attention window.

In order to tackle the issue of long-term dependencies, Informer was proposed (Zhou et al., 2021). This model is specifically designed to be efficient and accurate in forecasting long-sequence time-series. It is able to do this by using mechanisms such as ProbSparse self-attention and distilling layer, which reduce the computational complexity. Informer is especially effective when dealing with large-scale time-series data and is capable of making predictions about future values based on past data. It is more efficient than the transformer and LSTM in terms of speed and has excellent performance when dealing with long-sequence data, making it suitable for industrial processes. In this paper, we first use Informer to construct a soft sensor.

Although deep learning models have achieved good results in many fields, little attention has been paid to explaining their predictions. These models have a common problem: The internal structure is very complex and difficult for humans to understand. The output of the model is also difficult to explain, making its application in some areas related to life safety or important decision-making very risky. Due to the risk-sensitive nature of industrial processes, the reliability and stability of soft sensors are necessary for industrial applications. Interpreting soft sensor predictions can increase the reliability and stability of soft sensors.

Interpretable machine learning is a popular field for current and future machine learning research. In the design of soft sensors, interpretability refers to the transparency and ease of understanding of the model (Murdoch et al., 2019; Molnar, 2020; Du et al., 2019). A highly interpretable soft sensor allows users to understand how the model makes predictions based on input data, which input variables significantly impact the prediction results, and how these variables interact with each other. The main methods of explainable machine learning include LIME (Local Interpretable Model-Agnostic Explanations)(Ribeiro et al., 2016), SHAP (SHapley Additive exPlanations)(Lundberg and Lee, 2017), Counterfactual Explanations, Explainable Neural Networks, and Self-explanatory Machine Learning, etc. Among these methods, SHAP is one of the most commonly used representative approaches.

SHAP is a post-hoc interpretability method that employs perturbation tests and is applicable for both local and global explanations. Inspired by game theory, particularly the concept of Shapley valuesShapley (1951). Specifically, it involves perturbing the inputs to the model to understand how combinations of different features affect the prediction. Through this method, SHAP assigns a quantified importance score to each feature, indicating its importance in the model decision-making process. This approach not only provides explanations for individual predictions, but also assists in analyzing the overall importance of features in the model.

This work aims to establish robust and interpretable industrial soft sensors based on Informer and SHAP. The remainder of this article is organized as follows. In Section 2, detailed explanations of Informer and SHAP are given. In Section 3, novel robust and interpretable inferential sensors are proposed, with detailed implementation procedures and analysis. Section 4 presents a case study on real commercial fluid catalytic cracker (FCC) unit data to verify the effectiveness of the proposed method. Section 5 closes the paper with a summary.

2. METHOD

2.1 Informer

The Informer model addresses two critical challenges in long-term sequence forecasting: efficiently managing long-term dependencies and reducing both computational and memory requirements. This model is specifically designed to handle the complexities of extended sequences, ensuring that important temporal relationships are captured without overwhelming computational resources. The model's effectiveness stems from its innovative architecture, which we will explore in detail in the following sections.

The ProbSparse Self-Attention Mechanism Attention in machine learning is a mechanism that allows a model to focus on specific parts of an input, allowing it to better understand the data and make more accurate predictions. Attention works by assigning weights to different parts of the input, allowing the model to focus on the most important parts of the data. Traditional attention mechanisms were initially designed for specific tasks, such as sequence-to-sequence models in machine translation. While traditional attention is effective in mapping relationships between elements of two different sequences, it is not inherently designed to capture the relationships between elements within a single sequence.

Self-attention, a variant of the attention mechanism, is designed to weigh the importance of different elements within a single sequence. This is crucial in tasks where the relationship between samples in time-series data is significant. Self-attention is crucial in scenarios where understanding the internal dynamics of a single sequence is key. Its ability to consider each element of a sequence in relation to all others allows for a deeper understanding of the data, which is essential in soft sensors. Traditional self-attention computes attention weights using three matrices:

a query matrix $\mathbf{Q}$, a key matrix $\mathbf{K}$, and a value matrix $\mathbf{V}$, as follows:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}, \qquad (1)$$

The input sequence embeddings are transformed into these matrices through linear transformations, specifically $\mathbf{Q} = X\mathbf{W}^Q$ for queries, $\mathbf{K} = X\mathbf{W}^K$ for keys, and $\mathbf{V} = X\mathbf{W}^V$ for values, where X represents the input sequence embedding, and $\mathbf{W}^Q$, $\mathbf{W}^K$, and $\mathbf{W}^V$ are the learnable weight matrices.

The process of self-attention involves calculating the dot product between each query and all keys to generate a matrix of scores. These scores are scaled down by dividing by the square root of the dimension of the key vectors ($\sqrt{d_k}$). Next, the softmax function is applied to these scaled scores to transform the scores into probabilities. These probabilities dictate how much each part of the sequence should be considered when constructing the part of the output corresponding to a given query. Finally, the output of the self-attention layer is computed as a weighted sum of the value matrix $\mathbf{V}$, where the weights are the attention scores.

However, the computational intensity of this mechanism increases significantly with longer sequences due to the quadratic growth in the number of pairwise dot-products. To mitigate this computational burden, the Informer model introduces the ProbSparse Self-Attention mechanism, which intelligently and selectively samples key-query pairs from the sequence. Rather than computing the attention weights for all pairs in $\mathbf{K}$, the ProbSparse mechanism focuses on a subset of these pairs, represented as $\overline{\mathbf{K}}$. The corresponding scores, $\overline{\mathbf{S}}$, are computed as:

$$\overline{\mathbf{S}} = \mathbf{Q}\overline{\mathbf{K}}^\top \qquad (2)$$

A crucial step in ProbSparse Self-Attention is the selection of key-query pairs. This is achieved through a statistical approach where a metric M_{sparse} is used. One approach to compute M_{sparse} for a query $\mathbf{q}_i$ is to measure the discrepancy between the maximum score and the average score across all key interactions. Formally, this can be expressed as follows.

$$M_{\text{sparse}}(\mathbf{q}_i) = \max_j(\text{Score}(\mathbf{q}_i, \mathbf{k}_j)) - \text{average}_j(\text{Score}(\mathbf{q}_i, \mathbf{k}_j))$$

Here, $\max_j$ represents the maximum score for query $\mathbf{q}_i$ across all keys, and average_j represents the average score for the same query across all keys. Once we have computed the M_{sparse} values for all queries, we select the queries with the larger M_{sparse} values. These queries are considered more important in the self-attention mechanism. The selection criterion can be a fixed threshold, a percentile, or a fixed number of top queries based on their M_{sparse} values. This method of calculating M_{sparse} and using it to select the most important queries helps speed up the self-attention mechanism.

Optimizing Memory Usage The distilling layer in the Informer model is a crucial component designed to enhance the model's efficiency, particularly in handling long-sequence data. This layer addresses the challenge of memory usage and computational overhead associated with processing extended sequences. One of the primary functions of the distilling layer is to reduce the length of the sequence. This is achieved without losing critical information, which is essential for accurate forecasting. The

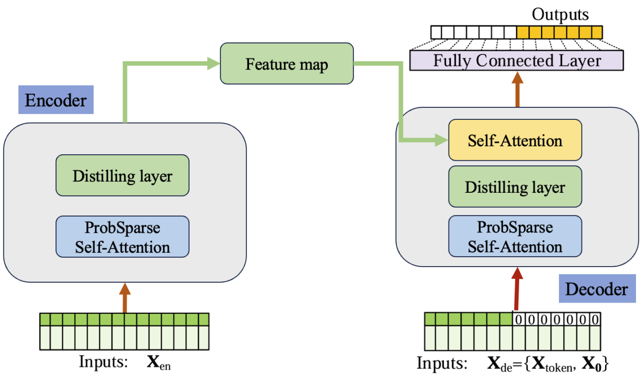

Fig. 1. Informer model overview

distilling layer uses convolutional operations to process the output from the preceding ProbSparse Self-Attention blocks. The convolutional operation is mathematically represented as:

$$\mathbf{X}_{\text{out}} = \text{ELU}\left(\mathbf{W} * \mathbf{X}_{\text{in}} + \mathbf{b}\right) \qquad (3)$$

In this equation, $\mathbf{X}_{\text{in}}$ is the input to the distilling layer, $\mathbf{W}$ represents the convolutional kernel, $\mathbf{b}$ is the bias term, and $\mathbf{X}_{\text{out}}$ is the output after convolution. The ELU (Exponential Linear Unit) function is applied for non-linear activation. The ELU function is defined as follows:

$$ELU(x) = \begin{cases} x & \text{if } x > 0 \\ \alpha(e^x - 1) & \text{if } x \leq 0 \end{cases} \qquad (4)$$

Here, x is the input to the activation function, and α is a hyperparameter. Following the convolutional operation, the distilling layer applies a max-pooling operation. Max-pooling helps in further reducing the dimensionality of the data, which is crucial for managing large sequences. It selects the maximum value from each subregion of the convolution output. The operation is represented as follows:

$$\mathbf{X}_{\text{pooled}} = \text{MaxPooling}\left(\mathbf{X}_{\text{out}}\right) \qquad (5)$$

This step results in a compressed representation of the input sequence, retaining the most significant features while reducing its length. By reducing the length of the input sequence, the distilling layer significantly decreases the computational complexity and memory requirements, making the model more efficient. The addition of the distilling layer makes the Informer more scalable, capable of handling very long sequences which are common in time-series data. This makes the Informer particularly suitable for complex time-series forecasting tasks.

Figure 1 illustrates the Informer model, which is designed for long-sequence forecasting tasks. The encoder takes in lengthy input sequences and uses the ProbSparse self-attention mechanism and distilling layer to extract and compress important information, reducing the size of the network. The encoder's output, a combined feature map, combines information from all layers, providing a comprehensive context to the decoder. The decoder has both of those layers, but between them there is a self-attention layer that helps the decoder focus on relevant parts of the input sentence. The output of the decoder is generated through a fully connected layer that converts the output of self-attention into the final prediction value.

2.2 SHAP (Shapley Additive exPlanations)

There is a trade-off between model prediction accuracy and model interpretability. It's evident that the model with high accuracy such as the deep learning models typically with lower interpretability. These models with lower interpretability often be said to be " black-box" models that bring challenges when attempting to explain their prediction.

It is essential to comprehend these black boxes, particularly in industrial applications, where forecasting is closely linked to safety and financial gains. SHAP is a method used to interpret predictions made by machine learning models. It provides a consistent approach for evaluating the effect of each feature on a prediction, taking into account both the feature's value and its interactions with other features. SHAP values are based on the concept of Shapley values in cooperative game theory, which guarantees a fair distribution of a coalition's contribution among its members. The Shapley value of the feature x is expressed as follows:

$$\phi_x(f) = \sum_{S' \subseteq p \setminus x} w_x(S') \left[f\left(S' \cup \{x\}\right) - f(S') \right] \tag{6}$$

The complex model Informer is represented by f, and $\phi_x(\bullet)$ is the Shapley value of feature x under f. The number of input features is denoted by p, and S' is a subset of these features. The union of S' and the feature x is represented by $S' \cup \{x\}$, which combines the elements of both S' and $\{x\}$ to form a new set. The weight of S' is defined as $\frac{|S'|!(p-|S'|-1)!}{p!}$, where $|S'|$ is the number of elements in the subset S'. The denominator $p!$ stands for all possible feature combinations, while the numerator $|S'|!(p-|S'|-1)!$ is the number of times $S' \cup \{x\}$ appears in all $p!$ combinations. The expected marginal contribution of the feature x in one combination is given by $f(S' \cup \{x\}) - f(S')$.

The Shapley value can be expressed as a linear additive feature model, which is the definition of SHAP. This is represented as follows:

$$g(z') = \phi_0 + \sum_{j=1}^{p} \phi_j z'_j \tag{7}$$

where ϕ_0 is the base prediction without any input information, usually the mean of the output in the training data, and ϕ_j is the distributed contribution for feature j. $z' \in \{0, 1\}$ is the subset features vector, with 1 indicating that the corresponding feature is present and 0 indicating that it is absent. Computing the Shapley values is an NP-hard problem. For deep learning models such as Informer, the traditional way to compute the Shap value can be computationally challenging. To address this, DeepSHAP, a faster algorithm, is used to compute SHAP values for the Informer-based soft sensor. DeepShap is an extension of SHAP designed to handle the calculation of the deep learning model's shap value.

3. PROPOSED METHOD

This work introduces a novel approach to developing robust and interpretable soft sensors for industrial applications using the Informer model and SHAP for feature importance analysis. Figure 2 shows the flowchart of the proposed method. The core of this method lies in leveraging the Informer's ability to handle long-sequence data and the interpretability afforded by SHAP to provide insights into the model's predictions.

The first part of the proposed method utilizes the Informer, a transformer-based model adept at processing long sequences with reduced computational overhead. Unlike traditional transformer models, the Informer incorporates ProbSparse self-attention, which selectively focuses on the most informative parts of the input data, thereby enhancing efficiency. Additionally, the distilling layers within the Informer act to compress the sequence length without losing critical temporal information. For industrial soft sensors, which often rely on large amounts of historical data to predict future states, the Informer's architecture is particularly beneficial. It guarantees that the model can accurately represent the fundamental behavior of the process over a long time, something that traditional models such as RNNs and LSTMs may not be able to do.

The complex encoder and decoder structure of the Informer model makes it impossible to explain the prediction by the model itself. The second part of the method involves the use of SHAP values to interpret the predictions of the Informer model. By decomposing the output of the Informer into contributions from each input feature, SHAP provides a granular understanding of how each feature influences the model's predictions. This is particularly important in industrial settings, where explanations of predictions can inform critical decisions. SHAP also allows for the assessment of the model's reliability, ensuring that its predictions are based on sensible data-driven insights rather than spurious correlations.

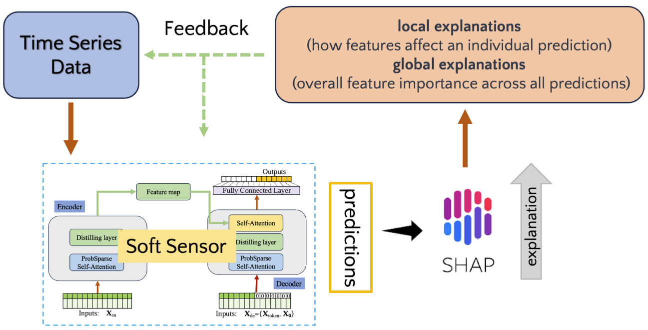

Fig. 2. Flowchart of interpretable Informer soft sensor model

The integration of the Informer model with SHAP-based interpretability represents a significant advancement in soft sensor technology. The method enables real-time monitoring and forecasting of quality variables in industrial processes, facilitating early warning and proactive intervention strategies. The ability to interpret the model's predictions ensures that operators can trust and act upon the insights provided, leading to better decision making and improved process control. The combination of Informer's predictive power and SHAP's interpretability creates a powerful tool for modern industrial applications, balancing the need for both accuracy and understanding in complex systems.

4. CASE STUDY

In the case study, we demonstrate the application of the Informer model coupled with SHAP analysis in a practical industrial setting. The study is conducted using data from a Fluid Catalytic Cracking (FCC) unit at the Parkland Refinery in Burnaby, British Columbia. The FCC unit, a critical component in refining operations, converts heavy hydrocarbons into lighter compounds that form the basis for various petroleum products (Su et al., 2022). FCC unit consists of three main parts, namely the reactor, the regenerator and the fractionator, which can be seen in Figure 3. The complex interactions within the FCC process, characterized by its multilevel systems and nonlinear dynamics, make it an ideal candidate for applying advanced soft sensor models like the Informer.

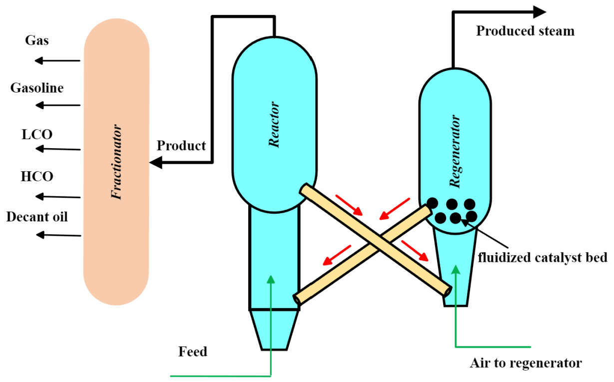

Fig. 3. A flow diagram of a Fluid Catalytic Cracking unit

Specifically, the model is trained to predict the distillation temperature, which is a significant quality variable in the FCC process. The input features are selected based on process knowledge and include readily measurable variables that influence the distillation temperature. A dataset comprising 58,719 samples collected between January 2019 and November 2022 is used. The dataset is partitioned, with 80% allocated for training and the remaining 20% for testing.

It is configured to process input sequences comprising 24 time steps, using these data to predict the subsequent 6 time steps. This capability is crucial for capturing the long-term temporal dependencies integral to the FCC process. The model is composed of two encoder layers and a single decoder layer, with an embedding depth of 512 and eight attention heads. The depth of the feedforward network is set at 2048 with a dropout rate of 0.05 to avoid overfitting. The sequence configuration is designed such that the encoder ingests a 24-time step sequence and the decoder, using an additional 12-time step sequence, forecasts the next 6-time steps, thereby providing a comprehensive view of the system's future state. Figure 4 shows the detailed prediction performance of the Informer soft sensor on the test data.

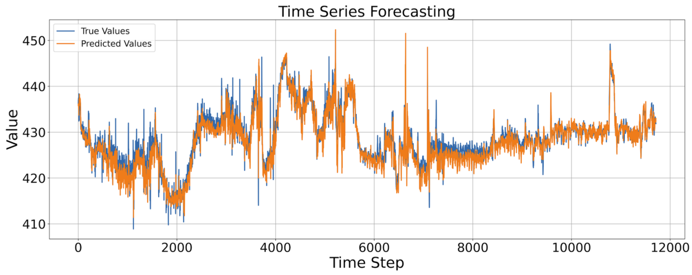

Fig. 4. The performance of Informer soft sensor on test data

Table 1. Comparison of different soft sensors on test data

	RMSE	R^2
Informer	1.9204	0.8829
Lasso Regression	3.6371	0.5790
Elastic Net	3.6171	0.5836
Huber Regressor	4.9286	0.2269
Bayesian Ridge	4.1901	0.4412
Ridge Regression	4.1907	0.4411
Linear Regression	4.1908	0.4410
Least Angle Regression	5.2178	0.1335
Extra Tree Regressor	5.0910	0.1751
Gradient Boosting Regressor	5.5064	0.0350
AdaBoost Regressor	5.3472	0.0900
Random Forest Regressor	5.9515	-0.1273
Passive Aggressive Regressor	4.2592	0.4226
Dummy Regressor	6.1233	-0.1933
LSTM	7.0044	-0.5614
Orthogonal Matching Pursuit	7.2656	-0.6801
Decision Tree Regressor	10.5976	-2.5744
K Neighbors Regressor	14.8899	-6.0561

Table 1 presents a comparative performance analysis of different soft sensors applied to test data from an FCC unit. It is evident from the results that the Informer model significantly outperforms traditional statistical and machine learning models, with an RMSE of 1.9204 and an R^2 value of 0.8829, indicating high prediction accuracy and a strong correlation with the actual data. This underscores the Informer's advanced capability in handling complex, long-term dependencies in time-series forecasting. In contrast, the LSTM model, which is also designed for time series data, exhibits a lower performance (RMSE of 7.0044 and an R^2 value of -0.5614), highlighting its limitations in capturing long-term dependencies.

We now have a soft sensor with outstanding performance, however, the Informer model itself has a complicated structure, making it hard to comprehend the inference process of the result from within the model. Generally, only the predicted value is provided, and the model is not interpretable at present. By incorporating SHAP to improve the interpretability of the model after the model is trained, and to extract the implicit information learned by the model, industry workers can understand the contribution of each feature in the forecasting process.

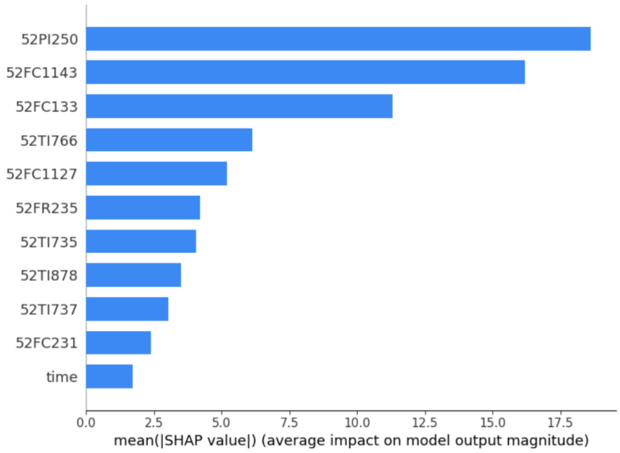

Fig. 5. Feature importance of Informer based on Shap

The SHAP analysis demonstrates that certain features, such as particular temperature and pressure readings, have a more significant effect on the model's predictions. These data confirm the effectiveness of the soft sensor and provide a layer of transparency to its operations. Figure 5 displays a visual representation of the SHAP values, which shows the contribution of each feature to the model's predictions, further strengthening the interpretability of the soft sensor. The SHAP value (contribution) is represented on the X axis. It is evident that for all the data, the most influential feature is feature 52PI250; On the other hand, feature 52FC231 has the smallest SHAP value, so its feature value has the least contribution to the final prediction.

5. CONCLUSION

In summary, this research has successfully demonstrated the implementation and effectiveness of the Informer model, combined with SHAP for better understanding, in industrial settings. The Informer model has been proven to be highly efficient in dealing with large-scale, long-sequence time-series data. The performance of the proposed algorithms was tested on real industrial FCC unit data, and the results showed that Informer models outperformed traditional statistical and machine learning models, with RMSE 1.9204 and R^2 0.8829, indicating a high degree of prediction accuracy and a strong correlation with the actual data. The addition of SHAP analysis enhances the model's utility by providing clear insights into the influence of individual features on the predictions, thereby increasing the model's transparency and trustworthiness in an industrial setting.

ACKNOWLEDGEMENTS

We express our gratitude to Mitacs and Parkland Corporation for their financial support and to our colleagues at Parkland for providing the necessary data and computational resources.

REFERENCES

Du, M., Liu, N., and Hu, X. (2019). Techniques for interpretable machine learning. *Communications of the ACM*, 63(1), 68–77.

Fan, J., Han, F., and Liu, H. (2014). Challenges of Big Data analysis. *National Science Review*, 1(2), 293–314.

Gopaluni, R.B., Tulsyan, A., Chachuat, B., Huang, B., Lee, J.M., Amjad, F., Damarla, S.K., Kim, J.W., and Lawrence, N.P. (2020). Modern machine learning tools for monitoring and control of industrial processes: A survey. *IFAC-PapersOnLine*, 53(2), 218–229. 21st IFAC World Congress.

LeCun, Y., B.Y..H. (2015). Deep learning. *Nature*, 521, 436–444.

Lundberg, S.M. and Lee, S.I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

Molnar, C. (2020). *Interpretable machine learning*. Lulu. com.

Murdoch, W.J., Singh, C., Kumbier, K., Abbasi-Asl, R., and Yu, B. (2019). Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, 116(44), 22071–22080.

Qin, S.J. (2014). Process data analytics in the era of big data. *AIChE Journal*, 60(9), 3092–3100.

Ribeiro, M.T., Singh, S., and Guestrin, C. (2016). "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd KDD International Conference*, 1135–1144. New York, NY, USA.

Shapley, L.S. (1951). Notes on the n-person game—ii: The value of an n-person game.

Su, J., Cao, L., Lee, G., Gopaluni, B., Siang, L.C., Cao, Y., van Dyk, S., Pinchuk, R., and Saddler, J. (2022). Tracking the green coke production when co-processing lipids at a commercial fluid catalytic cracker (fcc): combining isotope 14c and causal discovery analysis. *Sustainable Energy & Fuels*, 6, 5600–5607.

Yu, F., Cao, L., Li, W., Yang, F., and Shang, C. (2020). Feature based causality analysis and its applications in soft sensor modeling. *IFAC-PapersOnLine*, 53(2), 138–143. 21st IFAC World Congress.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., and Zhang, W. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 11106–11115.

Zhu, J., Jiang, M., Peng, G., Yao, L., and Ge, Z. (2021). Scalable soft sensor for nonlinear industrial big data via bagging stochastic variational gaussian processes. *IEEE Transactions on Industrial Electronics*, 68(8), 7594–7602.