A LSTM with Dual-stage Attention Method to Predict Amine Emissions for Carbon Dioxide Capture and Storage

Sai Rajesh Rapelli*, Zhiyuan Chen*, Wei Lu†
* University of Maryland Baltimore County, Baltimore, MD

Email: zhchen@umbc.edu, sairajr1@umbc.edu, wei.lu@raytum-photonics.com
† Raytum Photonics

***Abstract*—To mitigate climate change impacts, carbon capture technologies have been implemented at significant CO2 emission points, such as industrial sites and electric power generation facilities. Solvent-based carbon capture solutions are pivotal in reducing atmospheric CO2 levels and enhancing air quality by capturing harmful pollutants. Amine-based solvents, favored for their efficiency in post-combustion CO2 capture, are susceptible to thermal and oxidative degradation, leading to complex emissions profiles that demand comprehensive management strategies. We develop a Machine Learning model designed to predict future amine emissions in real-time, thereby assisting in the formulation of mitigation strategies required for the operation of capture plants. We conducted an experiment using data from test campaigns run at the Technology Centre Mongstad (TCM). We employed a Long Short-Term Memory (LSTM) autoencoder model with dual-stage attention mechanisms to predict amine emissions using historical data. The results were quite promising: we achieved a mean absolute percentage error ranging from 5.8% to 6.8% percent for the real-time prediction of amine emissions. The results are better than existing approaches using simpler machine learning models as well as the standard LSTM autoencoder model.**

I. Introduction

Climate change mitigation efforts have led to the implementation of carbon capture technologies at major CO2 emission sources, with solvent-based solutions playing a pivotal role. These technologies, particularly those using amine-based solvents, are effective in reducing atmospheric CO2 levels and improving air quality by capturing various pollutants, including sulfur dioxide. The process involves chemical solvents absorbing carbon dioxide from flue gases, with amine-based solutions being especially effective in post-combustion capture by binding directly with carbon dioxide molecules, thus significantly reducing greenhouse gas emissions [1].

However, the deployment of amine-based carbon capture systems introduces challenges related to the dispersion of harmful compounds, such as monoethanolamine (MEA) and nitrosamines (e.g., NDMA). These compounds, if not carefully managed, can pose significant environmental and health risks [2]. Moreover, the concentration of these emissions varies significantly based on environmental factors such as wind speed and the height of emission sources. This underscores the importance of accurate emission predictions and targeted mitigation strategies.

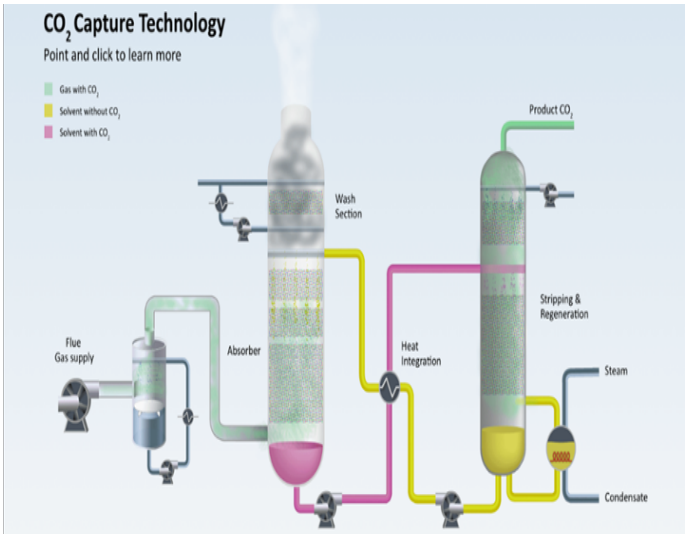

Fig. 1. Diagram of CO2 capture technology

Figure 1 presents a schematic representation of the carbon dioxide capture technology deployed at Technology Centre Mongstad (TCM). It begins with the preparation of industrial flue gas in the flue gas preparation unit, where the gas is cooled, and certain contaminants are removed via water circulation before it enters the CO_2 capture stage. The refined gas, now containing carbon dioxide, is directed to the absorber where it is captured by the solvent as the gas moves upward. A washing section at the absorber's apex ensures the CO_2 capture does not contribute to local environmental pollution. Pure CO_2 is gathered at the stripper's summit and is then dispatched for transportation, storage, or use. The solvent, having been regenerated, is cycled back to the absorber to persist in the CO_2 capture process.

In this paper, we focus on the development of a machine learning model designed to predict future amine emissions in

978-1-6654-3902-2/21$31.00 ©2021 IEEE

real-time, which can assist in the formulation of mitigation strategies required for the operation of capture plants. Machine learning, particularly deep learning techniques, has shown great promise in various fields for predictive analytics due to its ability to model complex nonlinear relationships and learn from large datasets [3].

We have made the following contributions:

1) We propose a deep learning technique for accurate prediction of amine emissions. Our solution uses a Long Short-Term Memory (LSTM) autoencoder with dual-stage attention. The autoencoder layers compress high dimensional time series data to a lower dimension latent space. The dual-stage attention mechanisms let the model focus on important input features as well as important dimensions in the latent space.
2) We conducted an experiment using data from test campaigns run at the Technology Centre Mongstad (TCM) to evaluate our method. The results were quite promising: we achieved a mean absolute percentage error ranging from 5.8% to 6.8% for the real-time prediction of amine emissions. The results are better than existing approaches using simpler machine learning models.

The rest of the paper is organized as follows. Section II reviews the related work. Section III describes our method. Section IV evaluates the model performance, comparing various machine learning models. Section V concludes the paper.

II. Related Work

Jablonka et. al. [4] treated the problem of predicting amine emission as a time series prediction problem and used gradient-boosted decision tree and Convolutional Neural Network (CNN). However, both gradient-boosted tree and CNN are not the most appropriate methods for time series data [5].

In this paper, we employed a Long Short Term Memory (LSTM) autoencoder model with dual-stage attention mechanisms to predict amine emissions using historical data. LSTM autoencoder learns a more concise data representation through an autoencoder, which transforms input data into a lower-dimensional embedding. Dual-stage attention was first proposed in [6]. It allows the model to focus on important input features and important embeddings, leading to better prediction results.

Other studies have investigated the use of CNNs, LSTMs, and attention mechanisms to increase prediction accuracy and model efficiency in a variety of applications [7]–[10]. Abbasimehr and Paki [11] compared the hybrid model against several standard and hybrid time series forecasting techniques. The proposed method involves using LSTM layers to capture short-term and long-term dependencies in the data, while the multi-head attention mechanism focuses on the most important features. It finally concludes that integrating attention mechanisms with LSTM models offers significant improvements in time series forecasting accuracy.

Banna et al. discussed a machine learning approach for earthquake prediction [12]. The authors added an attention layer to the LSTM architecture to improve the model's prediction accuracy for earthquake occurrence.

Recently, Li et al used a similar technique in the financial business to predict stock prices [13]. They used neural networks to increase the accuracy of financial time-series predictions, with a focus on forecasting the next trading day's closing price. It also used a attention-based LSTM model (AT-LSTM) that blends Long Short-Term Memory (LSTM) networks with attention mechanism.

Overall, LSTM models, especially when integrated with attention mechanisms, represent an effective approach to time series forecasting than previous methods such as CNNs and gradient-boosted decision trees.

III. Methodology

Section III-A describes the pre-processing step of our method. Section III-B presents the deep learning methods. We also discuss complexity of proposed method in Section III-C.

A. Pre-processing

We used data from the Technology Center Mongstad [14]. This dataset contains data from a test campaign conducted between November 1, 2020, and November 15, 2020. For this test, a portion of the product CO_2 was recycled to the Combined Heat and Power (CHP) flue gas inlet stream in a controlled manner to maintain the incoming CO_2 concentration at 5% by volume, dry basis. A process flow diagram, presented in Figure 2, shows the high-level equipment contained within the amine plant, along with key existing instrumentation.

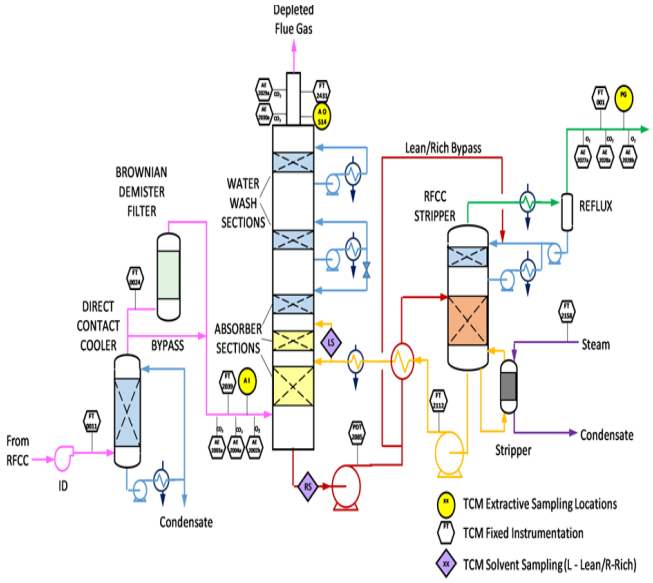

Fig. 2. Diagram of CO2 capture technology

This dataset records flue gas inlet properties, system performance, solvent circulation parameters, depleted flue gas composition, amine emissions, and water wash parameters.

The data is sampled every 10 minutes and includes 31 plant parameters and 4 amine emission measures, with 2 measures

each for AMP and Piperazine (two types of amine emission). AMP (2-amino-2-methyl-1-propanol) and Piperazine (an organic compound that consists of a six-membered ring containing two opposing nitrogen atoms) are solvent commonly used in CO2 capture and they can generate emission during the process.

In the pre-processing step, we converted the data into a time series using a window size of 25, meaning that we used data from the past 100 minutes to predict the emission data 10 minutes later. We selected a varying number of features using Pearson correlation, specifically, features with the highest absolute values of Pearson correlations with the emissions were selected. This data was then converted into a time series format, where each data sample consists of $10m$ dimensions, with m being the number of features. Each input data record includes 10 time steps, with m readings at each time step, where the first $m-1$ values are the selected features and the last value is a historical emission reading. The output is the AMP or Piperazine emission in the next 10 minutes (or 100 minutes after the initial reading in the input).

Output - AMPFTIR	
8610-TT-2041	Flue Gas temperature
8611-FT-2045	**Lean Solvent Flow**
8610-ai-2030a	co2 ir high
8610-AI-2036A	co2 gc
8610-AI-2036D	n2 gc
8610-AI-2026G	ammonia
8610-AI-2026K	formaldehyde
8610-AI-2026L	acetaldehyde
8610-TT-2102	**Upper WW Water Temp In**

Output - Piperazine FTIR	
8610-AT-0044	RFCC DCC pH
8610-TT-2035	Depleted FG Outlet Temp
8611-TT-2119	**Lean solvent temp in absorber**
8611-dt-2143	**Lean solvent Density**
8611-FT-2045	**Lean Solvent Flow**
8610-AI-2026L	acetaldehyde
8610-FIC-2373	Lower Water Wash Water Flow
8610-AI-2070AN	Piperazine IMR-MS
8610-TT-2558	**Upper WW Water temp out**

Fig. 3. Selected Features

Figure 3 displays the 10 features selected based on Pearson correlation that are from highest to lowest values, for AMP and Piperazine. Those in bold text are control parameters, e.g., flue gas temperature, lead solvent flow, and upper WW (Wastewater) Water temperature In. The remaining features are measure of depleted flue gas composition.

B. Deep Learning Models for Amine Prediction

We explored a number of deep learning models and selected the one that offers the highest accuracy. The list of models includes:

- **LSTM**: This is a popular type of recurrent neural network (RNN) well-suited for time series analysis. Unlike other RNN methods, LSTM avoids the problem of vanishing gradients, where long-term gradients tend toward zero. Such networks typically consist of several LSTM layers, each comprising a different number of LSTM nodes. An LSTM node has three inputs: the data X_t at time t, the cell state C_{t-1} from the previous node, and the hidden state (output) h_{t-1} from the previous node. It generates two outputs: C_t, the current cell state, and h_t, the current hidden state. The last hidden state is the output of the LSTM network [15]. An LSTM cell contains a cell state, which is managed by gates that control the flow of information. These gates, using a sigmoid activation, determine the amount of information added or removed from the cell state through pointwise multiplication [16]. These LSTM layers extract important features from the input data, while the last few densely connected layers are used for prediction.
- **LSTM Autoencoder**: Since our input data has a large number of dimensions, it often makes sense to map the input to lower dimensions (embedding) to extract only important information. The LSTM autoencoder is a popular method for mapping input time series data to lower-dimensional embedding. The data is fed to an encoder with a few LSTM layers and is mapped to an embedding. A decoder, having a reverse structure to the encoder, reconstructs the input from the embedding. A few dense layers are also used to predict future emission products from the embedding.
- **LSTM Autoencoder with Dual-Stage Attention**: Attention is a technique commonly used in deep learning to enhance performance over long sequences. For long sequences, RNNs tend to prioritize more recent data and overlook earlier data. Attention addresses this issue by computing a soft weight (meaning the weight can change depending on the data) for each input unit (e.g., input data at each time step or output from the previous layer). We used a dual-stage attention mechanism as described in [6]. This mechanism adds two attention layers to a regular LSTM autoencoder: one as the first layer (before the first LSTM layer) to assign weights to input features based on the encoder's previous hidden state, and the other is a temporal attention layer placed after the encoder but before the decoder, calculating weights of the embedding (output of the encoder) based on the decoder's previous hidden states.

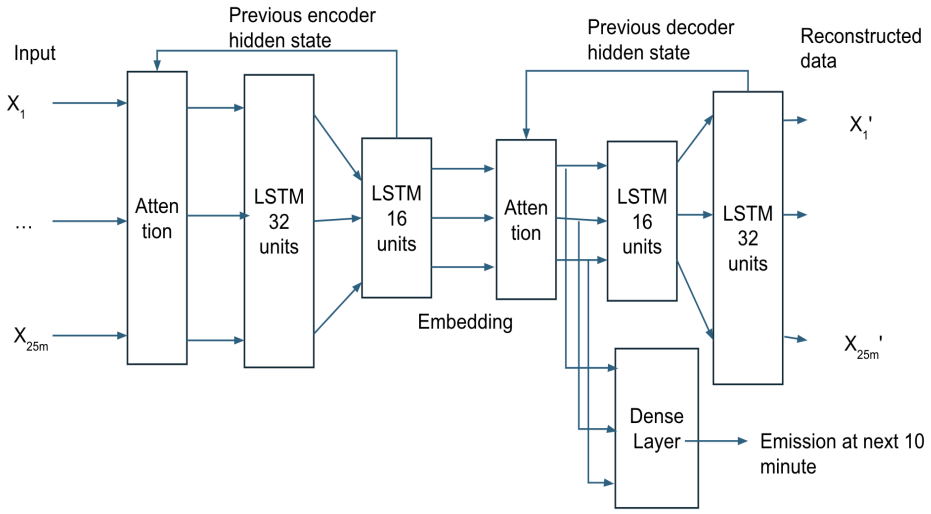

Fig. 4. LSTM Autoencoder with dual-stage attention

In our LSTM autoencoder with dual-attention model, the encoder consists of two LSTM layers: the first with 32 units and the second with 16 units. The output of the encoder is an embedding of the input, which is then passed through two decoder layers to reconstruct the input from the embedding [17]. Additionally, this embedding feeds into a dense layer with 100 units to predict emission outputs. Two attention layers are added, one before the first LSTM layer and the other after the embedding layer but before the first LSTM layer in the decoder. Figure 4 shows the structure of our model.

C. Computation Time and Model Complexity

The complexity of training a deep learning model is in the order of number of parameters in the network times number of training examples times number of training epochs. Since the deep learning models we use mainly contains LSTM layers and dense layers, we will examine number of parameters in each case. A LSTM layer has $4((x+h)h+h)$ parameters, where x is input size, h is output size (also number of LSTM nodes in the layer) [18].

For a dense layer, the number of parameters equals (input size +1) times output size. So if the input data is wm where w is number of time steps and m is number of features, and layer i has n_i neurons.

The first LSTM layer will have $4((wm+n_1)n_1+n_1)$ parameters. The second layer will take n_1 input and output n_2. So it has $4((n_1+n_2)n_2+n_2)$ parameters. We can continue this computation for each layer and then compute a sum of each layer.

In our proposed LSTM autoencoder with dual-stage attention, the total number of parameters is 34033, with most of parameters in the LSTM layers.

When we use the model for prediction, the computation cost is much lower because we don't need to have multiple epochs. So the computational cost is only related to the number of parameters in the model. In addition, when LSTM autoencoder is used, we no longer need to use the decoder layers in the model. This will also reduce the computational time. In general, although the training of prediction models may take significant amount of time (in hours), the prediction can be done in real time (in seconds).

IV. Experimental Results

A. Setup

a) Data set: We used TCM data set as described in Section III-A. We trained two models separately for each machine learning method: one to predict AMP emission and the other to predict Piperazine emission, using 80% of the data for training and the remaining 20% for testing. Features were selected using Pearson correlation. The pre-processing step was described in Section III-A.

b) Machine learning models: We evaluated the following machine learning models in our experiments.

1) LSTM: We used two LSTM layers with 32 and 16 LSTM units each, and a dense layer for prediction.
2) LSTM autoencoder: The encoder also consists of two LSTM layers with 32 and 16 LSTM units each, the decoder has the same two layers but in reverse order, and a dense layer taking encoder output as input for prediction.
3) Autoencoder with dual-stage attention: This is the proposed method, with LSTM autoencoder as described above and two attention layers, one as the first layer before encoder, and the other in between the encoder and decoder.
4) Autoencoder with a single attention layer: This is a variant where we only add an attention layer between encoder and decoder.
5) Gradient boosting tree: This is the method proposed in [4].

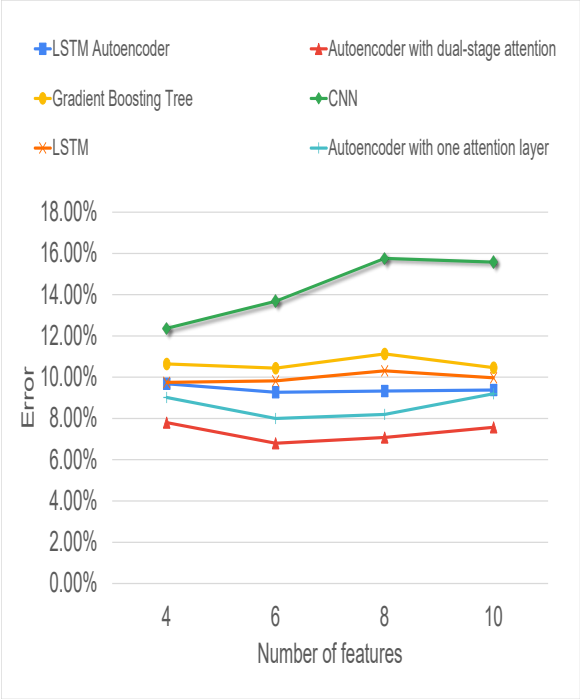

Fig. 5. Prediction of AMP emission using different models.

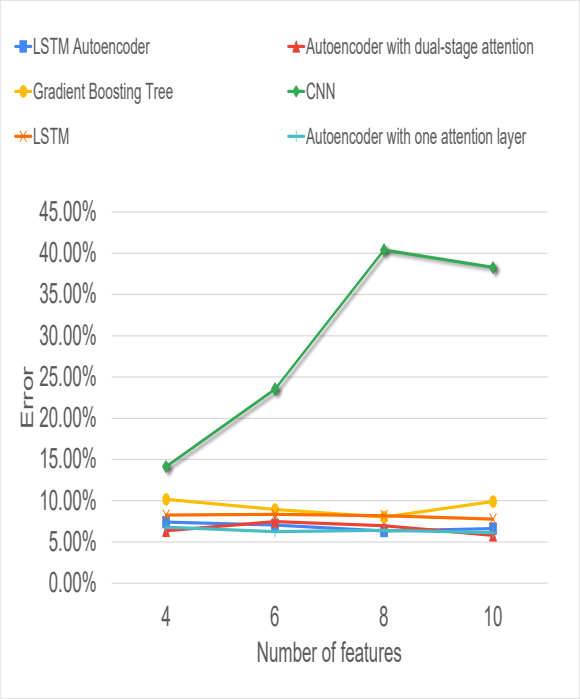

Fig. 6. Prediction of Piperazine emission using different models.

6) Convolutional neural network (CNN): This is the method proposed in [4]. To make a fair comparison, we used two layers of CNN with 32 and 16 units in each layer followed by a dense layer for prediction.

All models were implemented using Tensorflow 2.17.0 and Keras 3.4.1. All experiments were conducted on a Google Colab free tier node with 900 training epochs. The default configuration for a Colab free node uses a NVIDIA Tesla K80 GPU with 12 GB of VRAM.

c) Metrics: we used mean absolute percentage error as the metric for all models.

B. Results

Figure 5 compares the performance of various models to predict AMP emission with 4, 6, 8, and 10 features. The y-axis is mean absolute percentage error.

The error for LSTM Autoencoder (blue line) starts at around 8.5% error with 4 features, decreases to approximately 7% at 6 features, and then slightly increases to about 7.5% at 10 features. Its performance is slightly better than LSTM (without encoder/decoder).

Our proposed method, LSTM autoencoder with dual-stage attention, consistently outperformed other models. The error for autoencoder with dual-stage attention mechanisms (red line) starts at around 7.5% with 4 features, decreases to approximately 6.8% at 6 features, and remains stable around 7% for 8 and 10 features. The optimal setting is 6 features. This verifies the benefits of using dual-stage attention along with LSTM autoencoder.

LSTM autoencoder with just one attention layer performs better than autoencoder without attention, but is worse than LSTM autoencoder with dual-stage attention. This shows the benefits of using two layers of attention rather than just one layer because the former allows the model to focus on important input features as well as important embedding features.

The result of both Gradient boosting tree and CNN are worse than than the LSTM based models (over 10% error rate). CNN has the worst performance. This is expected as LSTM is known to be more suitable for time series data.

Figure 6 illustrates the performance of these different models in predicting Piperazine emission, measured as mean absolute percentage error. The x-axis ranges from 4 to 10. Since CNN performs much worse for Piperazine than other methods, we also plot the results without CNN in Figure 7.

The results are similar to the results for predicting AMP emission. The proposed model autoencoder with dual-stage attention outperforms the autoencoder and autoencoder with one attention layer with 4 and 10 features, but has slightly higher error with 6 and 8 features. However the lowest error rate is still achieved by the dual-stage attention model at 5.8% using all 10 features.

Autoencoder with one attention layer is the second best method, outperforming autoencoder without attention, but is

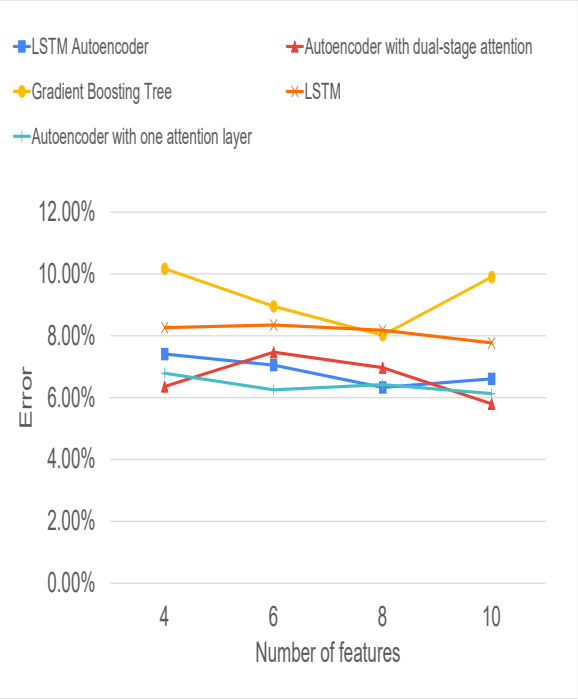

Fig. 7. Piperazine prediction results without CNN.

worse than autoencoder with dual-stage attention in terms of the best performance (using 10 features). This is expected as using dual-stage attention can better focus on features both in input and in latent space, especially when there are many features.

Autoencoder method also works better than LSTM and gradient boosting tree. CNN again has much higher error rate (over 10%) than all other methods and its performance is worse when more features are selected.

The error rates for LSTM based methods are also smaller for predicting Piperazine emission than for predicting AMP emission.

It is evident that incorporating attention mechanisms improves the prediction accuracy for both AMP and Piperazine emission prediction. LSTM based models also outperform existing methods such as Gradient Boosting Tree and CNN.

Figure 8 plots real values of AMP and values predicted by LSTM autoencoder with dual-stage attention. Figure 9 plots the same for Piperazine. The red lines indicate predictions, whereas the blue lines display actual numbers.

The results indicate that the predicted values of AMP and Piperazine using our method closely match the actual data. The proposed model effectively captures fluctuations and patterns with good precision.

Our proposed models are also efficient to train due to its relatively small size. Training time was below 30 minutes in all cases using just a Google Colab free tier node. The inference

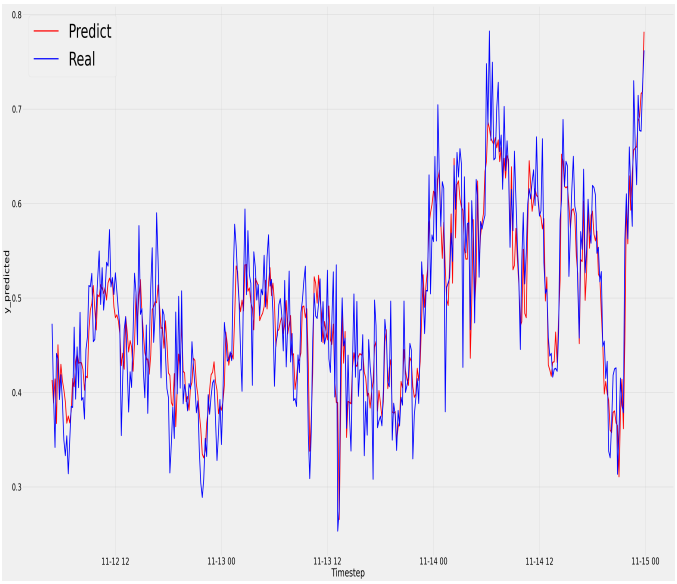

Fig. 8. Predicted vs. Real AMP FTIR in the TCM data set.

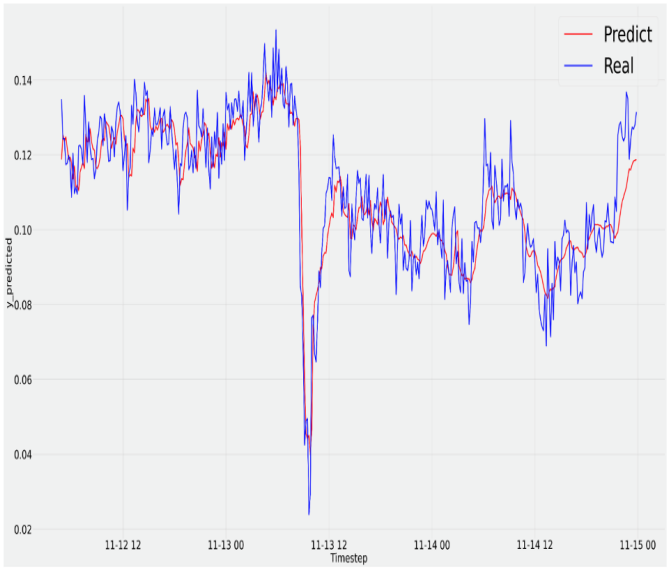

Fig. 9. Predicted vs. Real Piperazine in the TCM data set.

will just take around a second.

V. Conclusion and Future Work

We proposed an accurate amine emission prediction model using LSTM autoencoder with dual-attention. Experiments with real emission data showed promising results. Our method achieved an error rate between 5.8% to 6.8%. This marks a significant improvement over LSTM autoencoder as well as other existing methods.

We will enhance our emission prediction models by leveraging transfer learning techniques to generalize models trained on TCM data across different carbon capture technologies.

REFERENCES

[1] C. Wu, Q. Huang, Z. Xu, A. T. Sipra, N. Gao, L. P. de Souza Vandenberghe, S. Vieira, C. R. Soccol, R. Zhao, S. Deng *et al.*, "A comprehensive review of carbon capture science and technologies," *Carbon Capture Science & Technology*, vol. 11, p. 100178, 2024.

[2] A. Rusin and K. Stolecka-Antczak, "An analysis of hazards caused by emissions of amines from carbon dioxide capture installations," *Polish Journal of Environmental Studies*, vol. 25, 06 2016.

[3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[4] K. M. Jablonka, C. Charalambous, E. Sanchez Fernandez, G. Wiechers, J. Monteiro, P. Moser, B. Smit, and S. Garcia, "Machine learning for industrial processes: Forecasting amine emissions from a carbon capture plant," *Science Advances*, vol. 9, no. 1, p. eadc9576, 2023.

[5] J. A. Miller, M. Aldosari, F. Saeed, N. H. Barna, S. Rana, I. B. Arpinar, and N. Liu, "A survey of deep learning and foundation models for time series forecasting," *arXiv preprint arXiv:2401.13912*, 2024.

[6] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, and G. Cottrell, "A dual-stage attention-based recurrent neural network for time series prediction," *arXiv preprint arXiv:1704.02971*, 2017.

[7] X. Wen and W. Li, "Time series prediction based on LSTM-Attention-LSTM model," *IEEE Access*, vol. 11, pp. 48 322–48 331, 2023.

[8] L. Ma, Y. Zhao, B. Wang, and F. Shen, "A multistep sequence-to-sequence model with attention LSTM neural networks for industrial soft sensor application," *IEEE Sensors Journal*, vol. 23, no. 10, pp. 10 801–10 813, 2023.

[9] Y. Li and J. Yang, "Hydrological time series prediction model based on Attention-LSTM neural network," 09 2019, pp. 21–25.

[10] M. A. R. Suleman and S. Shridevi, "Short-term weather forecasting using spatial feature attention based LSTM model," *IEEE Access*, vol. 10, pp. 82 456–82 468, 2022.

[11] H. Abbasimehr and R. Paki, "Improving time series forecasting using LSTM and attention models," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, pp. 673 – 691, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:234311110

[12] M. H. A. Banna, T. Ghosh, M. J. A. Nahian, K. A. Taher, M. S. Kaiser, M. Mahmud, M. S. Hossain, and K. Andersson, "Attention-based bi-directional long-short term memory network for earthquake prediction," *IEEE Access*, vol. 9, pp. 56 589–56 603, 2021.

[13] H. Li, Y. Shen, and Y. Zhu, "Stock price prediction using attention-based multi-input lstm," in *Asian Conference on Machine Learning*, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:53591822

[14] Department of Energy, "Test data from a recent campaign at technology center mongstad (tcm) using cesar1 solvent," 2023. [Online]. Available: https://netl.doe.gov/sites/default/files/CESAR1-Emissions-Data-TCM-for-SBIR.xlsx

[15] F. Landi, L. Baraldi, M. Cornia, and R. Cucchiara, "Working memory connections for LSTM," *Neural networks : the official journal of the International Neural Network Society*, vol. 144, pp. 334–341, 2021.

[16] S. Sen and A. Raghunathan, "Approximate computing for long short term memory (LSTM) neural networks," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 37, no. 11, pp. 2266–2276, 2018.

[17] S. Tiwari, L. Kumaraswamidhas, C. Gautam, and N. Garg, "An auto-encoder based LSTM model for prediction of ambient noise levels," *Applied Acoustics*, vol. 195, p. 108849, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0003682X22002237

[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.