Prediction of Unsteady Flow for a 2D Cylinder Using Recurrent Neural Networks

Asmita Basu, Debajyoti Kumar, Pabitra Mitra, and Somnath Roy *Centre for Computational and Data Sciences*
Indian Institute of Technology Kharagpur, India
asmita.basu97, debajyotikumar, pabitra, mechsom@gmail.com

***Abstract*—Recurrent neural network (RNN) methods have gained prominence in forecasting sequential data for computational fluid dynamics (CFD). Predicting unsteady flow is a challenging problem in CFD. In this paper, we present data-driven deep-learning models to predict CFD simulations that forecast a 2D flow past a cylinder at Reynolds number $Re = 100$. Popular RNN models like Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) have been used to forecast turbulent flow using training data generated through CFD simulations. We explored LSTM, GRU, stacked LSTM, and encoder-decoder LSTM models, which demonstrated a high accuracy in predicting future states over consecutive time intervals. The trained models provide a reliable method for extrapolating velocity and pressure values within the temporal domain of the simulation. These results demonstrate the significant potential of RNN methods in CFD simulations.**

***Index Terms*—Unsteady flow, Recurrent Neural networks, GRU, LSTM, Encoder-Decoder, Computational fluid dynamics**

I. Introduction

Turbulence represents a departure from orderly, laminar flow, characterized by fluctuations in velocity, pressure, and density across a wide range of length and time scales. Traditionally, Computational Fluid Dynamics (CFD) has been a popular method for analysing such nonlinear fluid behaviour. It has proven highly beneficial in enhancing the precision of process simulations. However, despite its importance in modelling turbulent flows, it can be computationally expensive.

In recent years, the rapid advancement of machine learning (ML) and deep learning (DL) has emerged as a promising approach to accelerate CFD simulations. Additionally, high-performance computing has significantly helped mitigate the expense and time required for CFD simulations. More recent studies have explored hybrid deep data-driven algorithms. For instance, [1] employed a combination of RNN and GRU (LR-GRU), which exhibited higher accuracy than conventional methods. Their results indicate that the LR-CGRU algorithm can serve as a reliable and efficient alternative for precisely predicting the discharge coefficient of streamlined weirs. In a notable study, [3] proposed an innovative approach using a bidirectional RNN (bi-RNN) model incorporating LSTM units. This model is used to predict in-cylinder flow fields in internal combustion engines. The bidirectional nature of the network allowed it to process sequence data in both forward and backward directions, potentially capturing more comprehensive temporal relationships. The results of their study were particularly promising, demonstrating high accuracy in predicting complex interactions within in-cylinder flow fields.

The prediction of unsteady flows continues to pose a challenge, primarily due to the introduction of the time component, which tends to diminish the reliability of existing methods. [4] proposed a CNN-LSTM model, where LSTM exhibits a high degree of accuracy in predicting the Lift Coefficient C_L across the entire analyzed time range. Subsequently, the CNN, which takes the predicted Lift Coefficient from the LSTM as input, also accurately forecasts the velocity and pressure fields.

The work [5] applied a GRU model to predict the velocity components in the y-direction of this turbulent flow. The GRU, a variation of the Long Short-Term Memory (LSTM) network, is designed to efficiently learn and remember relevant information over extended time sequences. This property makes it particularly suitable for modeling the temporal dynamics inherent in turbulent flows.

The CFD-based LSTM and GRU ML models that were developed by [6] can accurately forecast future time steps using less of the dataset. When comparing the LSTM model to the GRU model and the typical CFD simulation, the LSTM model lowered the time and resources required by more than 50% . This reduction occurred despite having equal hardware configurations and complexity.

In this paper, we have predicted the flow field of flow past 2D circular cylinder at Reynold number 100 using recurrent neural networks. Hence, this paper is organized as follows. The methodology is presented in Section 2, which gives brief explanations of the models and the model framework. In Section 3, the results with diagrams from different data-driven techniques have been shown. Finally, the conclusions are presented in Section 4.

II. Methodology

Deep learning modelsare implemented in Python programming language. In this research, 'sklearn' and 'Keras' packages by 'TensorFlow' backend are used for program development. PARAM Shakti high performance computing system at IIT Kharagpur is used as the implementation hardware.

A. System equations for the fluid flow

In this section, we describe the governing equations describing the fluid flow setup. The continuity and momentum equations describe the behavior of incompressible Newtonian fluids in motion. They can be expressed as follows:

$$\frac{\partial u_i}{\partial x_i} = 0 \tag{1}$$

$$\frac{\partial u_i}{\partial t} + \frac{\partial u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{1}{Re}\frac{\partial^2 u_i}{\partial x_j^2} \tag{2}$$

In this study, the non-dimensional velocity (u_i), pressure (p), and time (t) are denoted as the variables of interest. The Reynolds number (R_e) is also considered in our analysis. To solve Equation (1), we employed the Marker-and-Cell (MAC) method with staggered mesh, which has been proven to be effective. Further details on the flow solver and discretization schemes can be found in the paper by [7].

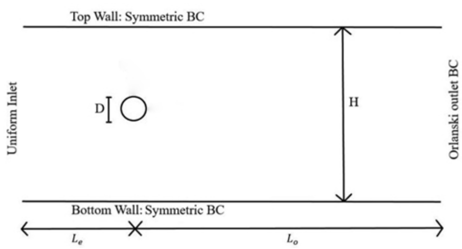

Fig. 1. Illustration of the domain with the solid body

B. Computational domain and boundary conditions

The body's diameter is represented by D. The inlet is positioned at L_e = 10D, while the outlet is at L_e = 30D. The total computational domain height is H = 40D. A uniform flow (U = 1) is maintained at the inlet. The outlet follows the Orlanski boundary condition, while the top and bottom boundaries have symmetric conditions. The body's surface enforces a no-slip condition (Fig.1). For the pressure Poisson equation, Dirichlet boundary conditions are applied at the outlet, with Neumann conditions elsewhere. A 40D x 40D computational domain was chosen to ensure domain independence, based on the time-averaged total drag coefficient (C_D). The domain is vertically divided into three blocks using a multiblock framework, with the solid body positioned in the middle block (Fig.1).

C. Recurrent neural networks

In the perspective of machine learning, the fluctuating flow rates can be treated as time series data. Forecasting models grounded in time series data have found extensive application in various domains to tackle prediction challenges associated with temporal patterns. The various RNN models were employed for forecasting future flow rates through training of historical CFD data.

1) GRU: The Gated Recurrent Unit (GRU) represents a variant of the Recurrent Neural Network (RNN) which, under specific circumstances, exhibits advantages compared to the long short-term memory (LSTM) model. The GRU cell comprises a hidden state, reset gate, and update gate. The reset gate allows control over the retention of previous hidden states, while the update gate facilitates understanding of how much of the new hidden state is derived from the previous one. The reset gate captures short-term dependencies within sequences, while the update gate handles long-term dependencies.

2) LSTM: LSTM's primary innovation lies in its capacity to selectively store, update, and recall information across lengthy sequences, rendering it highly effective for tasks centered on sequential data. The architecture of an LSTM network consists of memory cells, input gates forget gates, and output gates. Memory cells function as long-term storage, input gates manage the influx of new information into the memory cells, forget gates oversee the removal of obsolete information, and output gates dictate the output based on the prevailing state of the memory cells. LSTM tends to demonstrate higher accuracy, particularly when dealing with datasets featuring longer sequences.

3) Encoder-Decoder-LSTM: An autoencoder is a neural network for unsupervised learning, comprising an encoder and decoder. It learns a compact data representation by minimizing the reconstruction error between input and output. In our model, LSTM cells serve as both encoder and decoder. The encoder processes the input sequence, compressing information into internal state vectors (hidden-state and cell-state vectors in LSTM). The decoder, also using an LSTM cell, initializes its states with the encoder's final states. From this starting point, the decoder generates the output sequence, leveraging the information captured by the encoder.

D. Proposed methodology

In this work, we trained four deep learning algorithms on time series data consisting of velocity components in the V_x and V_y direction, and pressure values obtained using CFD simulations. The inputs include velocity components in the x and y direction, pressure at time instances. The output consists of the values for the same variable for the future time steps.

Single layers of GRU and LSTM were used for the GRU, LSTM, and Encoder-Decoder LSTM models. The stacked LSTM model consists of three LSTM layers. GRU and LSTM models use 64 hidden neurons, while the stacked LSTM and Encoder-Decoder LSTM models use 50 neurons each. The models used "ReLU" as its activation function. The models are optimized using Adaptive Moment Estimation (Adam) with a learning rate set to $1e-5$, which offers a fast rate of convergence, and a broad range of parameter values, and is computationally efficient. To avoid overfitting, an early termination code is included as well during model training. The training process halts when validation loss stabilizes.

III. RESULT

The error metrics, mean squared error (MSE) and root mean squared error (RMSE), are used to evaluate the performance of each deep learning method. These metrics are calculated by comparing predicted and actual flow rates over the testing

TABLE I
MODEL PERFORMANCE PARAMETERS

Model	Trn. Iterations	RMSE	MSE	# Parameters
GRU	1000	0.03460	$1.97x10^{-3}$	14197
LSTM	1000	0.01956	$3.80x10^{-4}$	18485
Stacked-LSTM	1000	0.03844	$1.47x10^{-3}$	51855
Encoder-Decoder LSTM	1000	0.0250	$6.25x10^{-4}$	11455

period, providing a measure of the model's predictive accuracy. The metrics are defined as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{y}_i)^2 \tag{4}$$

The simulation data has been taken from a 16-second flow. For the models, 9 seconds for training, 2 seconds for validation, and 5 seconds for testing, have been used. The dataset has x-axis from $0.4 - max$ and y-axis to $10 - max$. For each second, there are 35000 data points. For each of these points, we have the 2D velocities V_x and V_y, pressure and time. The predictions have been forecasted over two future time steps, such as 186.999 and 187.999 for a current time of 185.999. It is evident that the predictive capability of a single-layer LSTM surpasses that of a double-layer LSTM, such as the stacked-LSTM model, even when employing the same learning rate (0.00001). Notably, the accuracy of the double-layer LSTM model remains suboptimal, even when varying the learning rate from 0.001 to 0.00001. GRU converged the fastest, but LSTM was better at predicting RMSE values. The Encoder-Decoder LSTM had the best RMSE among all the models, although it had a slower convergence rate than the single-layer LSTM and GRU models. A comparison of the performances of the models are given in Table I.

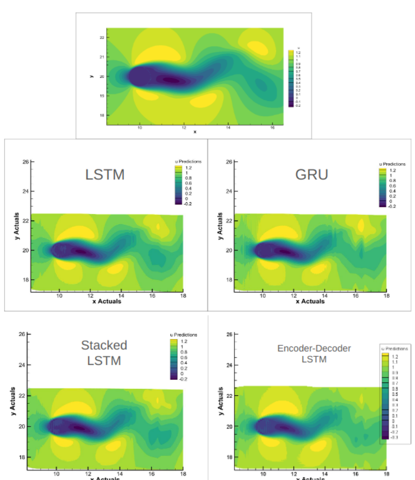

Fig. 2. Velocity in V_x direction at time 186.999 second

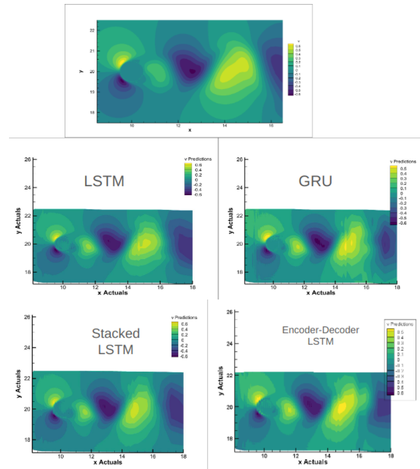

Fig. 3. Velocity in V_y direction at time 186.999 second

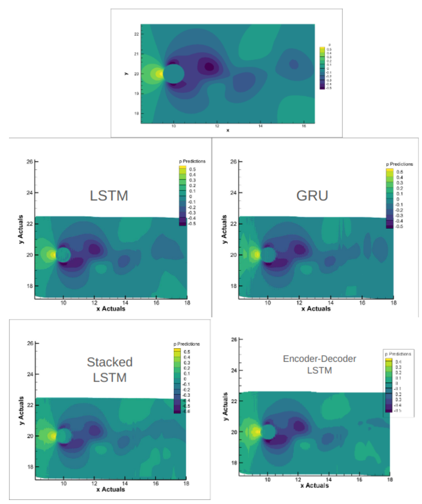

Fig. 4. Pressure at time 186.999 second

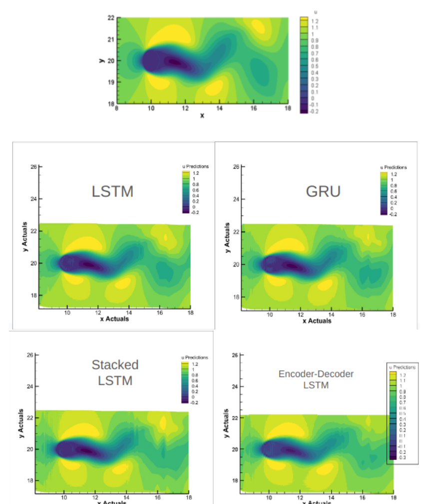

Fig. 5. Velocity in V_x direction at time 187.999 second

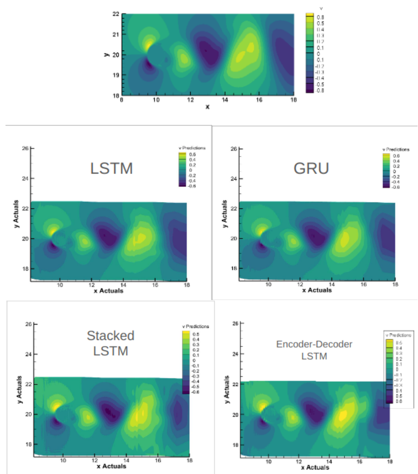

Fig. 6. Velocity in V_y direction at time 187.999 second

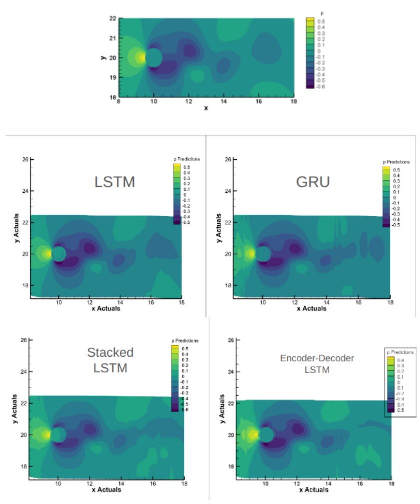

Fig. 7. Pressure at time 187.999 second

IV. Conclusion

Our study explores the potential of deep learning algorithms for the prediction of unsteady flows. Recurrent neural network techniques, which include Long Short-Term Memory (LSTM) networks and Gated Recurrent Unit(GRU), can be applied to predict future values after being trained on historical CFD data. The Encoder-Decoder LSTM model was compared against other Recurrent Neural Network(RNN) models and has been used to predict velocity in the x and y direction, and pressure in the fluid flow. Forecasting models, such as single-layer LSTM, stacked-LSTM, Encoder-Decoder LSTM, and GRU, possess the capability of retaining long-term dependencies. This feature is important for accurately predicting time series data. The models have been trained on historical CFD data that have been generated from simulations.

Predictions by the LSTM and GRU models demonstrated commendable agreement with CFD predictions, requiring significantly less computational effort. The error metrics such as MSE and RMSE have been selected to assess the performance of the models. The LSTM model had the best prediction outcome for RMSE, and Encoder-Decoder LSTM had the least GPU memory utilization while having comparatively better RMSE value than the rest of the models. Overall, both single-layer LSTM and Encoder-Decoder LSTM models did well in performance, without much difference between them. The models captured the prediction of the fluctuating parameters sufficiently. As research in this field continues to advance, LSTM and its variants are likely to play an increasingly important role in predicting and analysing dynamic systems across various scientific and engineering domains.

Acknowledgment

Authors thank the National Supercomputing Mission, MeitY, for financial support.

References

[1] W. Chen, D. Sharifrazi, G. Liang, S. S. Band, K. W. Chau, A. Mosavi, Accurate discharge coefficient prediction of streamlined weirs by coupling linear regression and deep convolutional gated recurrent unit, Engineering Applications of Computational Fluid Mechanics 16 (1) (2022) 965–976.

[2] F. Zhao, Z. Ruan, Z. Yue, D. L. Hung, S. Som, M. Xu, Time-sequenced flow field prediction in an optical spark-ignition direct-injection engine using bidirectional recurrent neural network (bi-rnn) with long short-term memory, Applied Thermal Engineering 173 (2020) 115253.

[3] K. Portal-Porras, U. Fernandez-Gamiz, E. Zulueta, O. Irigaray, R. Garcia-Fernandez, Hybrid lstm+ cnn architecture for unsteady flow prediction, Materials Today Communications 35 (2023) 106281.

[4] R. Hassanian, M. Riedel, L. Bouhlali, The capability of recurrent neural networks to predict turbulence flow via spatiotemporal features, in: 2022 IEEE 10th Jubilee International Conference on Computational Cybernetics and Cyber-Medical Systems (ICCC), IEEE, 2022, pp. 000335–000338.

[5] M. Nadda, K. Singh, S. Roy, A. Yadav, A comparative assessment of cfd based lstm and gru for hydrodynamic predictions of gas-solid fluidized bed, Powder Technology 441 (2024) 119836.

[6] D. Kumar, S. Roy, Simulation of flow over a fixed and oscillating circular cylinder with different surface roughness patterns using a multi-block-multi-mesh framework immersed boundary method, Physics of Fluids 35 (11) (2023).

[7] A. Raj, P. M. Khan, M. I. Alam, A. Prakash, S. Roy, A gpu-accelerated sharp interface immersed boundary method for versatile geometries, Journal of Computational Physics 478 (2023) 111985.

[8] K. Wang, R. B. Gopaluni, J. Chen, Z. Song, Deep learning of complex batch process data and its application on quality prediction, IEEE Transactions on Industrial Informatics 16 (12) (2018) 7233–7242.