

Learning the Language of Flow: A Self-Supervised FlowLanguage Framework for Airflow Prediction

Woohyoung Lee¹[0000-0002-0215-6848], Wenpeng Yin¹[0000-0002-3154-1139] and Wangda Zuo¹[0000-0003-2102-5592]

¹ Pennsylvania State University, University Park, PA, USA
wangda.zuo@psu.edu

Abstract. Computational fluid dynamics (CFD) can be utilized in advanced design to establish energy efficiency and decarbonization strategies for buildings. However, the high computational cost of CFD simulations creates a bottleneck in the rapid evaluation of design strategies. Recently, deep learning (DL)-based surrogate models have gained significant attention due to their capacity to accelerate flow field prediction. However, conventional DL-based surrogate models require extensive high-quality labeled datasets for training. This data dependency is a major limitation, as generating new labeled data for every scenario requires substantial resources. To address this limitation, this study develops FlowLanguage framework, which is designed to learn flow physics using the open-source large language model. The FlowLanguage consists of two primary stages. The first stage involves self-supervised pre-training to internalize the grammar of physics, which captures spatial continuity and causality of flow. In the subsequent stage, supervised fine-tuning transfers this internalized grammar, enabling accurate predictions with high data efficiency. To evaluate the framework, flow field was predicted for a 2D isothermal lid-driven cavity flow. The results show the enhancement achieved through transfer learning from self-supervised pre-training, characterized by high data efficiency with limited labeled training data. These findings indicate that transferable representations enable effective airflow modeling even with minimal labeled supervision. This framework serves as a smart design tool, facilitating the digital transformation and performance optimization required for the built environment.

Keywords: Deep learning, Computational fluid dynamics, Large language model, Surrogate model, Field prediction

1 Introduction

Indoor airflow analysis plays a crucial role in energy-efficient building design, including ventilation strategies and building energy optimization [1]. Computational fluid dynamics (CFD) provides essential design information, such as localized temperature and velocity fields [2]. However, CFD entails high computational costs as it provides results through the iterative numerical solution of the Navier-Stokes equations [3]. To address

these challenges, deep learning (DL)-based CFD surrogate models have recently emerged as a promising paradigm for fluid dynamics modeling. By establishing an efficient trade-off between reasonable predictive accuracy and rapid prediction time, DL-based surrogate models have gained significant attention from researchers [4]. Specifically, Zhang et al. (2025) [5] and Faulkner et al. (2023) [6] demonstrated the potential of replacing CFD simulation with convolutional neural network (CNN)- and generative adversarial network (GAN)-based surrogate models, achieving prediction speeds 2,282 and 75,000 times faster, respectively, while maintaining comparable accuracy. However, these conventional supervised DL approaches require extensive high-quality labeled datasets for training [7]. This data dependency poses a significant limitation, as substantial computational resources are required to generate new labeled data for every specific scenario.

Although many researchers have proposed various methods to overcome data dependency and enhance generalization performance [8, 9], a significant limitation remains as achieving high accuracy still necessitates vast amounts of data. To address these challenges, this study proposes FlowLanguage, a novel two-stage training framework that integrates the large language models (LLMs) into fluid dynamics modeling. Beyond merely applying pretrained models, FlowLanguage systematically adopts the LLM paradigm by bifurcating the training process into a self-supervised pre-training stage followed by a supervised fine-tuning stage. This approach establishes a robust foundation for flow field prediction characterized by high data efficiency.

The remainder of this paper is organized as follows. Section 2 introduces the FlowLanguage framework and the two-stage training methodology. Section 3 describes the selection of a 2D case study in a representative indoor environment to evaluate the performance of FlowLanguage. Section 4 provides and analyzes the results of the case study. Finally, Section 5 concludes the study and proposes directions for future research.

2 Methodology

The proposed FlowLanguage framework is designed to ensure flow field prediction accuracy while minimizing the dependency on labeled data required in the supervised learning. As shown in Fig. 1, the overall framework consists of three primary phases. First, in the data preprocessing phase, flow field data obtained from OpenFOAM [10] is organized into a cell-wise token structure. This structure integrates the flow field, distance-based context, and boundary condition information corresponding to each token. Subsequently, the entire domain is partitioned into local tiles, which serve as the base units for both the two-stage training and final prediction. In the following phase, the LLM performs two-stage training utilizing these tile-wise tokens. Finally, in the prediction phase, the predicted local tiles are combined to predict the flow field using the trained model.

2.1 Model Structure

Data Preprocessing. FlowLanguage structures the input by representing each cell within the entire domain as an individual token. The original flow fields are transformed into a flow-oriented frame aligned with the position and movement direction of the moving wall; this is then organized alongside pressure as the target flow field variables for each cell. Simultaneously, each cell is assigned role-based information to distinguish boundary locations, as well as spatial context representing its relative position to the moving wall and surrounding boundaries. Additionally, a boundary condition vector—comprising the Reynolds number, moving wall velocity components, and aspect ratio—is constructed at the case level, ensuring that all subsequent tiles share consistent physical condition information. Finally, the entire domain is partitioned into overlapping local tiles, establishing a unified input structure for both the two-stage training process and the prediction.

LLM Physics Engine. This study utilizes Qwen3-1.7B as the backbone, with training conducted via Low-Rank Adaptation (LoRA). The model input consists of cell tokens within local tiles; each token incorporates local coordinates, distance-based spatial context, boundary role information, a boundary condition vector, flow state, and an observation mask. Subsequently, the tokens within each tile are restructured into one-dimensional sequences before being fed into the backbone. In this study, four-way scan orders are employed to facilitate this sequence-based learning. Consequently, the FlowLanguage framework is designed to internalize flow structures at the local tile level using a sequence-based modeling approach.

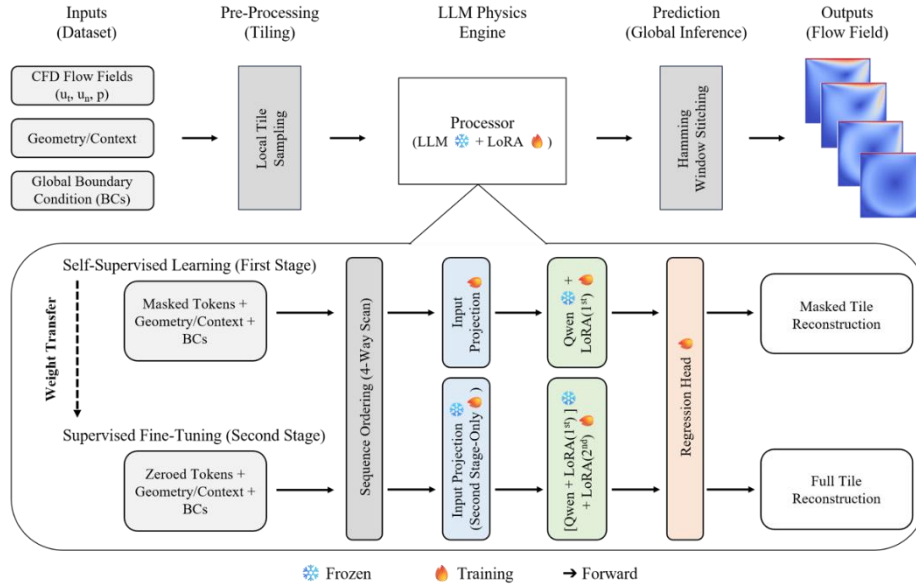


Fig. 1. High-level overview of the proposed FlowLanguage framework.

Prediction. Upon completion of the two-stage training, the flow field is predicted using the trained model. This process involves extracting local tiles from the global domain to perform individual predictions, followed by merging overlapping regions to reconstruct the complete flow field. The current inference stage employs a four-way scan ensemble for tile prediction, and a Hamming window stitching is utilized for the final stitching process.

2.2 First Stage: Self-Supervised Learning

As illustrated in Fig. 2, FlowLanguage consists of two distinct learning stages. The first stage comprises self-supervised learning based on masked reconstruction modeling. In this stage, masked inputs are generated by obscuring a portion of the tokens within each tile; the model is then trained to reconstruct the missing regions by leveraging the surrounding flow structures and boundary context. This study employs a masking ratio ranging from 30-70%, which is designed to enable the model to internalize the capability for local flow pattern restoration.

2.3 Second Stage: Supervised Fine-Tuning

The second stage consists of a full-tile prediction phase performed through supervised fine-tuning (Fig. 2). Unlike the first stage, this phase does not rely on partial masking reconstruction; instead, the model is trained to directly predict the flow field of a tile using inputs from which the initial flow state has been removed. During this process, the model follows a predictive architecture that integrates the flow field, distance-based context, and boundary condition information corresponding to each cell. In essence, the objective of the second stage is to reconstruct the complete flow structure of the local tile based on the provided local boundary and geometric context.

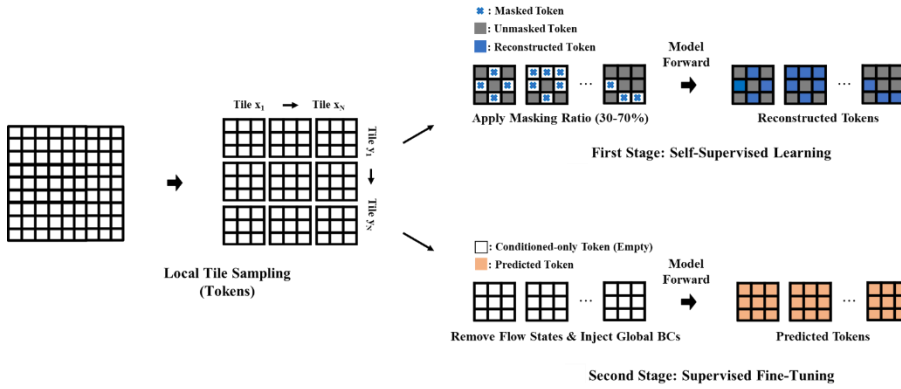


Fig. 2. Illustration of the two-stage learning framework. First stage indicates self-supervised learning process, and second stage indicates supervised fine-tuning.

3 Numerical Simulation Set Up

3.1 Lid-Driven Cavity Description

A two-dimensional (2D) lid-driven cavity flow was selected as the target for analysis. As shown in Fig. 3, the top wall was modeled as a moving wall with a constant velocity of 1.0 m/s, while the remaining three walls were stationary. To examine the variations in flow patterns according to the Reynolds number (Re), Re was utilized as a parameter by adjusting the kinematic viscosity while maintaining a constant lid velocity.

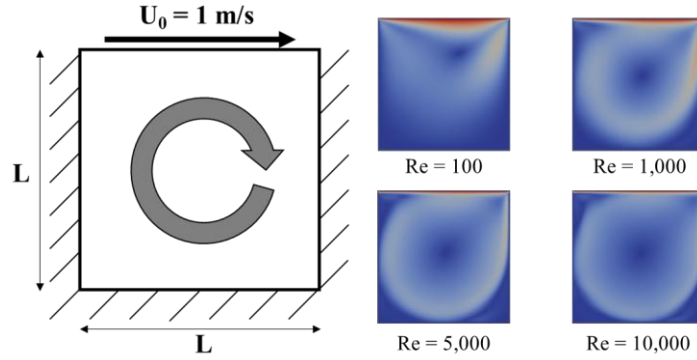


Fig. 3. Diagram of the lid-driven cavity flow and example flow pattern regarding different Re .

3.2 Dataset

In this study, a total of 45 cases with Re values ranging from 100 to 10,000 were constructed to form the total dataset. In the first stage, self-supervised learning was performed using 30 cases. In the subsequent second stage, to evaluate the framework under limited labeled data conditions, the number of labeled cases was configured as $k=2, 3, 5, 7$, and 10 , respectively. The prediction performance was then evaluated on five unseen Re (120, 2,490, 6,300, 7,870, 9,400) values that were excluded from the training.

3.3 Setup and Evaluation

To examine the effectiveness of the first stage, this study compared the FlowLanguage framework, which incorporates both training stages, against a Second Stage-Only baseline. The first stage was applied solely to FlowLanguage, involving 20 epochs of training in 30 cases. In the second stage, both approaches were trained under identical conditions using labeled cases for 25 epochs. Prediction accuracy was assessed using the root mean square error (RMSE) of velocity magnitude between predicted and ground truth fields, as defined in Eq. (1).

$$RMSE = \sqrt{\sum_i^N \frac{(y_i - y_p)^2}{N}} \quad (1)$$

4 Results and Discussion

4.1 Visualization of Flow Fields

Fig. 4 shows a comparison of velocity magnitude fields at $Re=2,490$ for the ground truth, FlowLanguage and the Second Stage-Only. Even with only $k=2$ (Fig. 4a), FlowLanguage predicts a velocity magnitude field consistent with the ground truth. In contrast, the Second Stage-Only exhibits non-physical noise and unstable velocity patterns across the domain. Although increasing the number of labeled cases to $k=10$ (Fig. 4b) substantially improves the Second Stage-Only prediction, FlowLanguage already achieves a comparable level of prediction at $k=2$. This result demonstrates that the representations learned in the first stage are effectively transferred to the second stage, enabling stable prediction with a limited number of labeled cases.

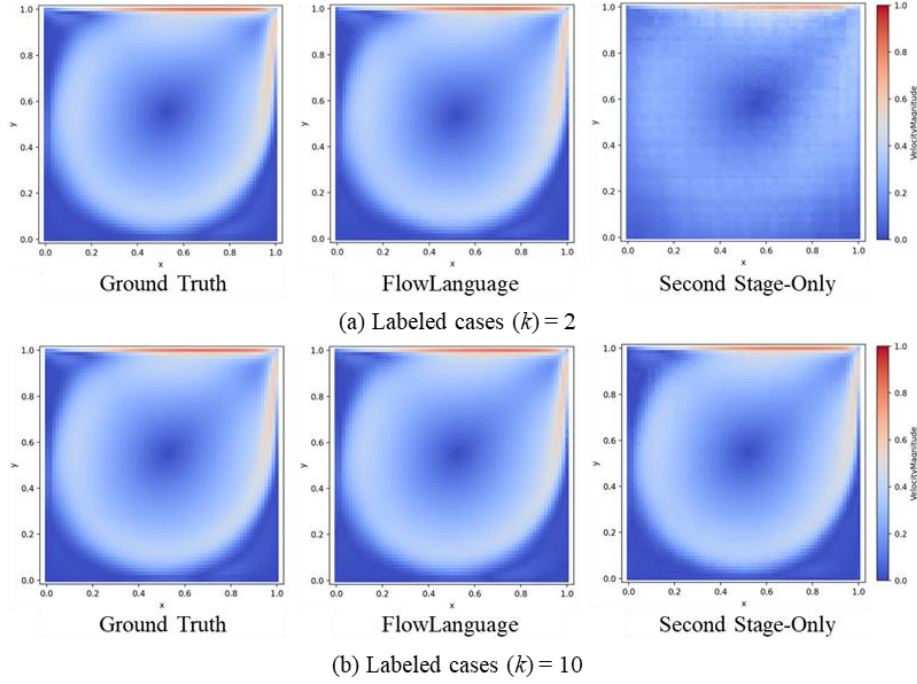


Fig. 4. Comparison of velocity magnitude fields at $Re=2,490$ for the ground truth, FlowLanguage, and the Second Stage-Only: (a) $k=2$ and (b) $k=10$ labeled cases in the second stage.

4.2 Evaluation of Data Efficiency Depending on Labeled Training Data

To evaluate label efficiency in the second stage, the mean RMSE and standard deviation across the five unseen Re values were calculated for different numbers of labeled cases, as shown in Fig. 5. The mean RMSE for FlowLanguage was 0.020 for $k=2$, 0.019 for $k=3$, 0.014 for $k=5$, 0.013 for $k=7$, and 0.012 for $k=10$. This result indicates that

FlowLanguage maintains a relatively low and stable error range as the number of labeled cases varies, while showing gradual improvement as the number of labeled cases increases. In contrast, the mean RMSE for the Second Stage-Only showed a much stronger dependence on the number of labeled cases, decreasing from 0.071 at $k=2$ to 0.053 at $k=3$, 0.029 at $k=5$, 0.020 at $k=7$, and 0.016 at $k=10$. Notably, FlowLanguage consistently achieved a lower mean RMSE than the Second Stage-Only for all values of k . This result indicates that the representations learned in the first stage improve label efficiency and support more accurate predictions with a limited number of labeled cases.

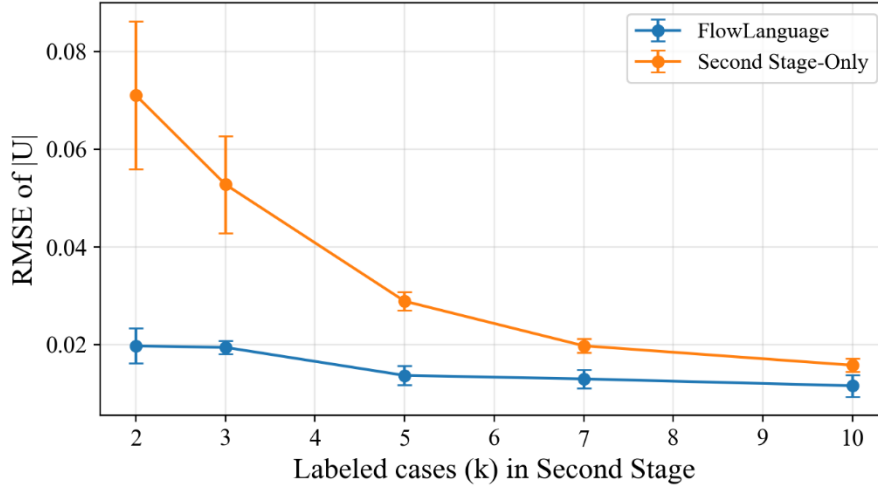


Fig. 5. Mean RMSE of FlowLanguage and the Second Stage-Only across different numbers of labeled cases (k) in the second stage. Error bars indicate the standard deviation across five unseen Re values.

4.3 Evaluation of Prediction Performance Depending on Re

Table 1 shows the RMSE for the five unseen Re values across different numbers of labeled cases. FlowLanguage maintained consistently low RMSE across all of k , with an overall range of 0.009–0.024. In contrast, the Second Stage-Only showed substantially higher RMSE, ranging from 0.050 to 0.096 at $k=2$, and decreasing to 0.014–0.018 at $k=10$. Across all five unseen Re values, FlowLanguage yielded lower RMSE than the Second Stage-Only at the same k . Moreover, from $k=5$ onward, FlowLanguage also outperformed the Second Stage-Only at $k=10$ across all five unseen Re values. This result quantitatively demonstrates that self-supervised pretraining in the first stage improves prediction accuracy and data efficiency during the second stage.

Table 1. RMSE of FlowLanguage and the Second Stage-Only for five unseen Re values across different numbers of labeled cases (k).

Re	FlowLanguage					Second Stage-Only				
	k=2	k=3	k=5	k=7	k=10	k=2	k=3	k=5	k=7	k=10
120	0.014	0.022	0.013	0.013	0.009	0.050	0.036	0.026	0.021	0.015
2,490	0.024	0.019	0.010	0.010	0.009	0.096	0.066	0.028	0.018	0.014
6,300	0.023	0.019	0.015	0.014	0.013	0.075	0.056	0.030	0.019	0.015
7,870	0.020	0.018	0.014	0.013	0.013	0.069	0.054	0.030	0.019	0.016
9,400	0.019	0.019	0.016	0.015	0.015	0.066	0.052	0.031	0.021	0.018

5 Conclusion

This study proposed FlowLanguage, a flow field prediction framework that combines a Qwen3 backbone with a two-stage training strategy. In the first stage, the model learns local flow patterns and boundary context through self-supervised learning, and these learned representations are transferred to the second stage for supervised fine-tuning. Evaluation on the 2D lid-driven cavity flow showed that FlowLanguage consistently achieved lower RMSE than the Second Stage-Only baseline under the same number of labeled cases, while maintaining stable prediction performance with a limited number of labeled cases. In particular, from $k=5$ onward, FlowLanguage outperformed the Second Stage-Only at $k=10$ across all unseen Re values. These results demonstrate that self-supervised pretraining improves prediction accuracy and label efficiency. Future work will extend FlowLanguage to broader indoor airflow problems and investigate its zero-shot and few-shot generalization under more diverse boundaries and flow conditions.

Acknowledgements

This research was supported by the National Science Foundation, United States under Award Number ECCS-2446782. This research was also supported by Rising Researcher award ICDS_RRYY_027628 and Faculty Upskilling award from Penn State's Institute for Computational and Data Sciences (RRID: SCR_025154).

References

1. Hoyt T, Arens E, Zhang H (2015) Extending air temperature setpoints: Simulated energy savings and design considerations for new and retrofit buildings. *Building and Environment* 88:89-96.
2. Blocken B, Defraeye T, Derome D, Carmeliet J (2009) High-resolution CFD simulations for forced convective heat transfer coefficients at the facade of a low-rise building. *Building and Environment* 44(12):2396-2412.
3. Turek S (1996) A comparative study of time-stepping techniques for the incompressible Navier-Stokes equations: from fully implicit non-linear schemes to semi-implicit projection methods. *International Journal for Numerical Methods in Fluids* 22(10):987-1011.
4. Zhao L, Zhou Q, Li M, Wang Z (2024) Evaluating different CFD surrogate modelling approaches for fast and accurate indoor environment simulation. *Journal of Building Engineering* 95:110221.
5. Zhang W, Zhang C, Zhao Y, Wang Z, Liu Y, Zhou C, Hu Y (2025) Convolutional neural networks-based surrogate model for fast computational fluid dynamics simulations of indoor airflow distribution. *Energy and Buildings* 326:115020.
6. Faulkner CA, Jankowski DS, Castellini Jr JE, Zuo W, Epple P, Sohn MD, Saad W (2023) Fast prediction of indoor airflow distribution inspired by synthetic image generation artificial intelligence. *Building Simulation* 16(7):1219-1238.
7. Mao R, Lan Y, Liang L, Yu T, Mu M, Leng W, Long Z (2025) Rapid CFD prediction based on machine learning surrogate model in built environment: A review. *Fluids* 10(8):193.
8. Zhang W, Wang Z, Wang F, Zhang C, Zhao Y (2026) U-Net-based CFD surrogate modeling method for fast indoor airflow field prediction with enhanced spatial generalization capabilities. *Journal of Building Engineering* 115811.
9. Vandewiel MR, Eneyew DD, Awol AD, Capretz MA, Bitsuamlak GT (2025) Approximating CFD simulations of natural ventilation: A deep surrogate model with spatial attention mechanism. *Journal of Building Engineering* 105:112425.
10. OpenCFD Ltd (2023) OpenFOAM User Guide v2306. <https://www.openfoam.com>, last accessed 2026/03/18.