LLMPatchTST: LLM-Enhanced Time Series Forecasting for Smart Farming

*Note: Sub-titles are not captured in Xplore and should not be used

***Abstract*—Time series forecasting is vital for optimizing smart farming operations, yet traditional methods often falter with complex agricultural data. This paper proposes a novel hybrid model integrating a time series uni-model with a Large Language Model (LLM) via prompt embedding, enabling multimodal forecasting - LLMPatchTST. Tested on a real-world dataset from a smart farm in Damyang, South Korea, our approach surpasses baselines like TCN, Transformer, ARIMA and PatchTST in MSE and MAE metrics. Beyond improved accuracy, this integration fosters advanced reasoning and automation in smart farming. Our results highlight LLMs' potential to revolutionize agricultural data analysis and decision-making.**

Keywords—smart farm, time series forecasting, LLM

I. Introduction (*Heading 1*)

In smart farming, sensor data is vital for monitoring environmental conditions and crop growth. Time series forecasting, using historical data, predicts factors like temperature and humidity to optimize cultivation strategies. AI, including machine learning and deep learning, has advanced this field but faces challenges with complex time series data. Large Language Models, originally designed for natural language processing, show promise in improving forecasting accuracy by handling contextual information. This paper introduces a hybrid model that combines a time series model with an LLM, tested on a dataset from a real smart farm. Experimental results indicate it outperforms baseline models. The paper is organized into sections on methodology, experiments, and conclusion.

II. Method

1) *Task Formulation*

Time series forecasting can be mathematically formulated as follows: Given a multivariate time series:

$$x = \{x_1, x_2, \dots, x_T\}, x_t \in \mathbb{R}^d$$

where: x_t is the observation at time step t, d is the dimensionality of each observation (i.e., number of variables), T is the length of the observed history, the objective is to predict the future values:

$$\hat{x} = \{\hat{x}_{T+1}, \hat{x}_{T+2}, \dots, \hat{x}_{T+H}\}, \hat{x}_t \in \mathbb{R}^d$$

where: H is the forecasting horizon (i.e., how many steps ahead to predict), $\hat{x}_t$ is the predicted value at time t.

2) *Proposed Model*

Our proposed model integrates a time series uni-model with an LLM, utilizing prompt embedding to convert time series data into text, thus enabling a multimodal approach. The model comprises two primary branches: the time series branch and the text branch, which are fused to produce the final forecast (as show in figure 1)

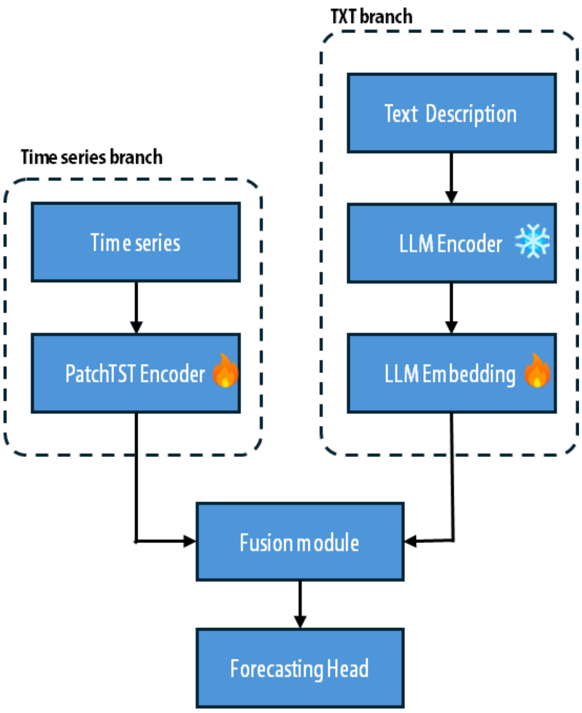

Fig. 1: Architecture proposed

a) Text Branch

The text branch processes time series data transformed into textual descriptions via a prompt template. For instance, a sequence might be represented as: "*The temperature at time step 1 is 25.0, at time step 2 is 25.5, ..., at time step T is 26.0.*" This text is then embedded into feature vectors using an LLM, specifically GPT-2 in this study. Due to the small dataset size and hardware constraints [1], we freeze the LLM's weights during training to reduce computational overhead. A subsequent layer aligns the dimensions of these text feature vectors into a common vector space for fusion.

b) Time Series Branch

The time series branch employs PatchTST as its backbone, a Transformer-based model tailored for time series forecasting. The input time series is segmented into patches—short subsequences—that the Transformer encoder processes to extract temporal and value information, yielding feature vectors.

c) Fusion Module

A fusion module combines the feature vectors from both branches using a Cross-attention mechanism. Here, the Query is derived from the time series branch, while the Key and Value are sourced from the text branch. The fused output is passed through a forecasting layer to generate predictions tailored to the forecasting task.

This hybrid architecture leverages the contextual understanding of LLMs and the temporal feature extraction of PatchTST, enhancing overall predictive performance.

III. Experiments and results

Datasets

We collected a dataset from a real smart farm in Damyang, South Korea, comprising nine variables: CO2, Humidity, Temperature, Sun1, Sun2, Sun3, Sun4, Sun5, and Sun6. Data was recorded in real-time at 15-minute intervals from April 26, 2025, to May 26, 2025. The dataset and code are publicly available at: https://github.com/AISeedHub/LLMPatchTST.

Experimental Settings

Experiments were conducted on a GPU-H100. To mitigate the ‘Drop Last’ issue during testing [Qiu et al., 2024], we set the test batch size to 1. Performance was evaluated using Mean Squared Error (MSE) and Mean Absolute Error (MAE), with an input length of 96 and a forecasting horizon of 96 across all models. The dataset was split into 60% training, 20% validation, and 20% testing.

Baselines

We compared our model against three baselines: Temporal Convolutional Network (TCN) [2], ARIMA and PatchTST [3].

Results

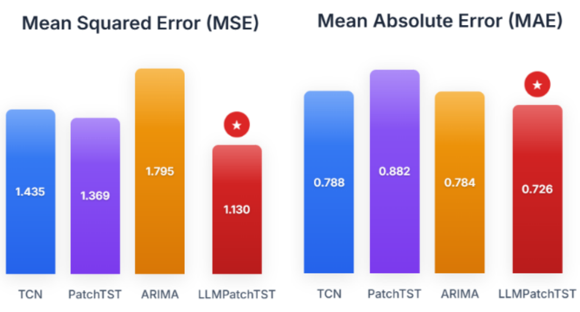

Fig. 2: Time Series Forecasting Model Performance Benchmark on MSE and MAE

Our proposed model achieves the lowest MSE and MAE, indicating superior forecasting accuracy. A visual comparison of predictions, illustrated in a chart Fig 2, further confirms that our model excels, particularly in capturing significant fluctuations, outperforming PatchTST, the strongest baseline.

These results underscore the efficacy of integrating LLMs with time series models, validating the potential of this approach in smart farming applications.

Analysis

These results underscore the efficacy of integrating LLMs with time series models, validating the potential of this approach in smart farming applications.

Conclusion

This paper introduces a hybrid model combining a time series uni-model with an LLM, demonstrating its effectiveness in time series forecasting for smart farming. By transforming time series data into text via prompt embedding and fusing it with PatchTST, we achieve enhanced prediction accuracy, surpassing conventional baselines. This application of LLMs not only advances forecasting capabilities but also unveils broader opportunities in smart farming. Beyond prediction, the contextual strengths of LLMs could extend to reasoning tasks and AI-driven automation, paving the way for future innovations in intelligent agriculture.

References

[1] Ma, Jian, et al. "SplitFrozen: Split Learning with Device-side Model Frozen for Fine-Tuning LLM on Heterogeneous Resource-Constrained Devices." arXiv preprint arXiv:2503.18986 (2025).

[2] Anh, Le Hoang, et al. "Stride-TCN for Energy Consumption Forecasting and Its Optimization." Applied Sciences 12.19 (2022): 9422.

[3] Nie, Yuqi, et al. "A time series is worth 64 words: Long-term forecasting with transformers." arXiv preprint arXiv:2211.14730 (2022)