Evaluating Mamba Model Performance on Time-Series Datasets with Prior Knowledge Integration

Tanya Dora Hanne Raum Prof. Joschka Boedecker

Abstract

Mamba, a recently proposed state-space model originally developed for language modeling, has demonstrated strong potential in handling long input sequences with near-linear computational complexity. In this study, we explore its application in environmental forecasting by integrating domain-specific prior knowledge derived from a semi-empirical model of boreal forest processes (PRELES). We propose and evaluate several strategies for incorporating this prior knowledge: either by directly feeding it as additional input features or by embedding it into the model through an attention mechanism that allows selective focus on relevant signals. The objective of this study is to assess whether such knowledge integration improves Mamba's ability to generalize, capture long-term dependencies, and enhance its performance in time series forecasting tasks.

Introduction

Recent advances in deep learning have significantly improved time series forecasting by enabling models to capture complex patterns over time. One such development is the Selective State Space Model, **Mamba** (Gu & Dao, 2024), which was originally introduced for natural language processing tasks. Despite its origins, Mamba has shown strong potential in time series applications due to its ability to efficiently handle long sequences while maintaining high predictive performance (Wang et al., 2024).

In practical forecasting scenarios, **prior knowledge** - such as known relationships from physical models, spatio-temporal dependencies or ecological processes, can serve as an important source of information. In this study, the focus is on the utilization of prior knowledge derived from the **PRELES** model (Peltoniemi et al., 2015), a semi-empirical ecosystem model that simulates **Gross primary production (GPP)** and **Evapotranspiration (ET)** in boreal forests.

In view of the evidence that prior knowledge has been shown to enhance performance in other advanced models, the present study focuses on assessing whether the specific prior knowledge in question similarly benefits the Mamba model. In this study, we explore this question by evaluating different ways of incorporating prior knowledge into the Mamba framework. We compare the model's performance with and without prior knowledge, and assess whether more informed integration - such as through an attention mechanism - can improve its generalization and accuracy.

The following contributions are summarized:

- We integrate prior knowledge in S-Mamba, a Mamba-based model for Time Series Forecasting (TSF) (Wang et al., 2024).This prior knowledge is based on domain-specific insights from a semi-empirical model of boreal forest ecosystems (Peltoniemi et al., 2015), which captures established relationships between gross primary production (GPP), evapotranspiration (ET).
- The effectiveness of incorporating prior knowledge in the Mamba Time Series Forecasting model is evaluated and compared for both in-distribution (ID) and out-of-distribution (OOD) tasks.

Background

Time Series Forecasting

Time series forecasting plays a crucial role in various sectors, such as finance (He et al., 2023) and healthcare (Jung et al., 2021). Unlike other types of data, time series data is inherently sequential and involves complex temporal dependencies, making it more difficult to accurately model and predict. The evolution of these models has been marked by a transition from statistical methodologies to deep learning techniques (Chen et al., 2023).

State Space Model

State space models are used for the purpose of describing the state representations and predicting the subsequent states, based on certain inputs (Gu et al., 2021b; Smith et al., 2022). First-order differential equations are utilized to represent the evolution of the system's internal state, and another set to

describe the relationship between latent states and output sequences, so that input sequences $\mathbf{x}(t) \in \mathbb{R}^D$ can be mapped to output sequences $\mathbf{y}(t) \in \mathbb{R}^N$ in through latent states $\mathbf{h}(t) \in \mathbb{R}^N$, as shown in 1:

$$\mathbf{h}'(t) = A\mathbf{h}(t) + B\mathbf{x}(t), \quad \mathbf{y}(t) = C\mathbf{h}(t) \tag{1}$$

where $A \in \mathbb{R}^{N\times N}$ and $B, C \in \mathbb{R}^{N\times D}$ are learnable matrices. Subsequently, the continuous sequence is discretized by a step size Δ. The discretized state space model is represented as 2:

$$\mathbf{h}_t = \bar{A}\mathbf{h}_{t-1} + \bar{B}\mathbf{x}_t, \quad \mathbf{y}_t = C\mathbf{h}_t \tag{2}$$

where

$$\bar{A} = \exp(\Delta A)$$
$$\bar{B} = (\Delta A)^{-1}(\exp(\Delta A) - I) \cdot \Delta B$$

As the model transitions from the continuous form (Δ, A, B, C) to the discrete form (A, B, C), the calculation can be efficiently executed through a linear recursive approach (Gu et al., 2021b). The Structured State Space Model (S4), a variant of the State Space Model (SSM), was proposed by (Gu et al., 2021a). Using HiPPO (High-order Polynomial Projection Operators) framework (Gu et al., 2020) for initialization, adding structure to the state matrix A, thereby improving the model's ability to capture long-range dependencies.

Selective State Space Model (Mamba)

Mamba (Gu & Dao, 2024) extends S4 by incorporating a data-dependent selection mechanism and integrating hardware-aware parallel algorithms within its recurrent computation loop.

Algorithm 1 The Process of Mamba Block

Input: $\mathbf{X} \in \mathbb{R}^{B\times V\times D}$
Output: $\mathbf{Y} \in \mathbb{R}^{B\times V\times D}$
$\mathbf{x}, \mathbf{z} \in \mathbb{R}^{B\times V\times ED} \leftarrow \text{Linear}(\mathbf{U})$ *// Linear projection*
$\mathbf{x}' \in \mathbb{R}^{B\times V\times ED} \leftarrow \text{SiLU}(\text{Conv1D}(\mathbf{x}))$
$\mathbf{A} \in \mathbb{R}^{D\times N} \leftarrow \text{Parameter}$ *// Structured state matrix*
$\mathbf{B}, \mathbf{C} \in \mathbb{R}^{B\times V\times N} \leftarrow \text{Linear}(\mathbf{x}'), \text{Linear}(\mathbf{x}')$
$\mathbf{\Delta} \in \mathbb{R}^{B\times V\times D} \leftarrow \text{Softplus}(\text{Parameter} + \text{Broadcast}(\text{Linear}(\mathbf{x}')))$
$\mathbf{A}, \mathbf{B} \in \mathbb{R}^{B\times V\times D\times N} \leftarrow \text{discretize}(\mathbf{\Delta}, \mathbf{A}, \mathbf{B})$ *//Input-dependent parameters and discretization*
$\mathbf{y} \in \mathbb{R}^{B\times V\times ED} \leftarrow \text{SelectiveSSM}(\mathbf{A}, \mathbf{B}, \mathbf{C})(\mathbf{x}')$
$\mathbf{y}' \in \mathbb{R}^{B\times V\times ED} \leftarrow \mathbf{y} \otimes \text{SiLU}(\mathbf{z})$
$\mathbf{Y} \in \mathbb{R}^{B\times V\times D} \leftarrow \text{Linear}(\mathbf{y}')$ *// Linear projection*

As illustrated in Algorithm 1 (Wang et al., 2024), the input sequence X is first projected into two streams, x and z, via linear transformation. The stream x undergoes 1D convolution followed by a SiLU activation to produce $\mathbf{x}'$, which is then used to selectively generate the state-space model (SSM) parameters A, B, and C. An input-conditioned step size Δ, computed via a softplus activation, enables the discretization of the SSM for adaptive state transitions. This selective, input-dependent mechanism allows Mamba to focus on relevant temporal information while attenuating noise, enhancing its ability to capture long-range dependencies. Additionally, its parallel-friendly architecture ensures computational scalability, making Mamba both accurate and efficient for long-sequence modeling.

S-Mamba Block

An adaptation of the Mamba framework, the S-Mamba (Wang et al., 2024) specifically designed for time series forecasting. S-Mamba is designed to efficiently model both temporal dependencies and inter-variate correlations in multivariate time series data. The S-Mamba model takes a sequence $U_{\text{in}} \in \mathbb{R}^{B\times L\times V}$ as input, where B denotes the batch size, L the input sequence length, and V the number of variates, and produces a forecasted sequence $U_{\text{out}} \in \mathbb{R}^{B\times V\times D}$, where D denotes the hidden dimension. Its architecture comprises the following components illustrated by Figure 1 (Wang et al., 2024):

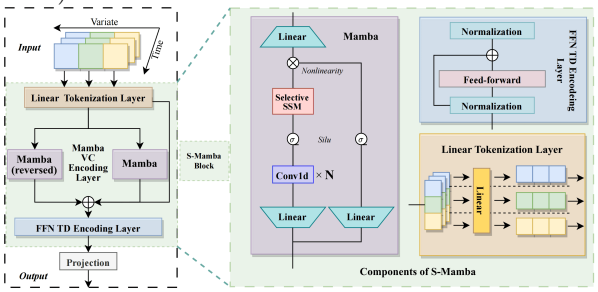


Figure 1: Illustration of the S-Mamba architecture.

S-Mamba model begins by transposing the input time series data to align the variables along the token dimension. A linear projection is then applied to map each variable into a hidden embedding space of dimension D, resulting in token embeddings where each token corresponds to a different variable.

Next, the token embeddings are passed through a Mamba Intervariate Correlation (VC) Encoding Layer. This layer uses two parallel Mamba blocks, one operating in the forward direction and the other in the backward direction. The outputs of these bidirectional Mamba blocks are fused using element-wise addition. A residual connection is then added to maintain information stability and facilitate learning.

The fused output is passed to a Feed-Forward Network (FFN) layer. This layer applies Layer Normalization, followed by a position-wise FFN to model interactions independently across dimensions. Another Layer Normalization is applied to stabilize the output further.

Finally, the resulting embeddings are projected to match the desired prediction horizon T. A final transpose operation is performed to align the dimensions appropriately, producing the model's output.

This architectural difference makes S-Mamba particularly effective for multivariate time series fore-

casting tasks, as it emphasizes capturing the relationships between different variables before modeling their temporal dynamics.

Related Work

Worth of Prior Knowledge

In the context of time series forecasting, incorporating prior knowledge has been shown to enhance model performance, especially in scenarios with limited or noisy data. Embedding domain-specific insights can guide models to focus on relevant features and improve predictive accuracy. Feature augmentation using domain knowledge has been successfully applied in financial and healthcare forecasting tasks (Fang & Lin, 2020).

Moreover, prior knowledge can inform attention mechanisms, enabling models to focus on critical time steps or variables—an approach that has improved forecasts in medical domains (Yu et al., 2024). Building on these findings, the present study explores whether similar benefits can be achieved in environmental forecasting by integrating domain-specific prior knowledge from models like PRELES into Mamba-based architectures.

Inspiration from ST-MambaSync

The model is partially inspired by the ST-MambaSync model (Shao et al., 2025), which demonstrated that combining Mamba with Transformer-based attention improves forecasting performance in traffic-related time series tasks. In contrast, we focused solely on Mamba due to its computational efficiency. Rather than using Transformer layers, we investigate whether attention mechanisms guided by ecological knowledge—such as outputs from the PRELES model—can effectively support Mamba in embedding prior knowledge. This allows us to assess if domain-specific insights can enhance model interpretability and performance without the overhead of Transformer architectures.

Method

Integration of Prior Knowledge

We incorporate domain knowledge from the PRELES (PREdict Light-use Efficiency and Soil moisture stress) model (Peltoniemi et al., 2015), a semi-empirical eco-physiological model designed to estimate daily gross primary production (GPP) and evapotranspiration (ET) in forest ecosystems. PRELES uses environmental variables such as photosynthetically active radiation (PAR), air temperature (Tair), vapor pressure deficit (VPD), soil moisture, and CO_2 concentration to compute these estimates.

In our approach, we use the PRELES-simulated outputs— `GPP_pred` and `ET_pred`. These values serve as a form of structured prior knowledge to S-Mamba. These features are intended to inform the model about established ecological dependencies and serve as a guide for learning more robust and generalizable temporal representations. To evaluate their contribution, we compare different architectural variants of model with and without Prior Knowledge.

Experimental Setup

The present study set out to evaluate the impact of integrating domain-specific prior knowledge into Mamba-based time series forecasting. To this end, a series of experiments was conducted using the proposed S-Mamba model, followed by an analysis of forecasting performance across both in-distribution (ID) and out-of-distribution (OOD) scenarios. In-distribution (ID) refers to scenarios where the model is evaluated on data similar to the training set, while out-of-distribution (OOD) refers to cases where the model is tested on data from a different sites.The experiments are conducted on a dataset comprising daily environmental measurements collected from multiple boreal forest sites. Each record includes meteorological variables such as `PAR, Tair, VPD, Precipitation, fAPAR, and atmospheric` CO_2, along with the target variables: gross primary production (GPP) and evapotranspiration (ET). Additionally, two simulated estimates—`GPP_pred` and `ET_pred`—derived from the PRELES model are included as prior knowledge.

To evaluate the effect of integrating domain-specific prior knowledge into Mamba-based forecasting models, we designed six experimental configurations. These experiments vary in how prior knowledge is introduced into the learning process, either as direct inputs or via attention mechanisms. Additionally, Experiments 1, 2, 3 and 4 are run both **with and without** prior knowledge to serve as a baseline comparison.

We evaluate following strategies for integrating this prior knowledge into the S-Mamba framework:

Experiment 1 & 2: Direct Feature Integration

- **Approach:** Prior knowledge is appended as additional input features alongside the standard variables.
- **Architecture:**
  - **Experiment 1:** Uses a feature embedding layer - which projects the input and optional time features into the model dimension, followed by dropout and then an encoder stack based on Mamba blocks (see Figure 2).
  - **Experiment 2:** Replaces the embedding with a simple linear projection of the input features directly to the model

dimension, omitting time encoding and dropout, before passing through the same Mamba-based encoder (see Figure 3).

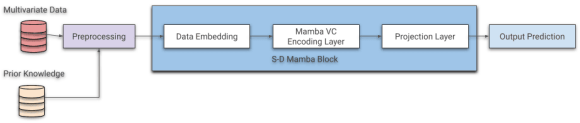


Figure 2: Illustration of Experiment 1.

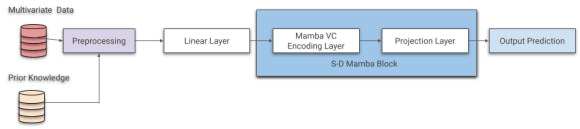


Figure 3: Illustration of Experiment 2.

Experiment 3 & 4: Attention over All Input Features

- **Approach:** A multi-head attention mechanism is applied across **all input features**, including prior knowledge, to allow the model to weigh feature importance contextually.

- **Architecture:**
  - **Experiment 3:** Applies multi-head attention **after** the embedding layer. This allows the model to first transform raw input into the model space before computing attention weights (see Figure 4).
  - **Experiment 4:** Applies attention **before** the embedding layer. Raw input features are passed directly into the attention module. The attention output is then projected and embedded before being passed to the encoder (see Figure 5).

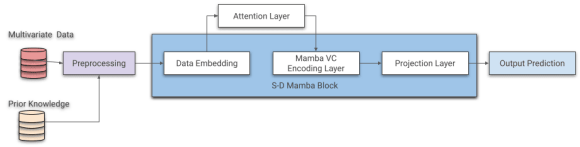


Figure 4: Illustration of Experiment 3.

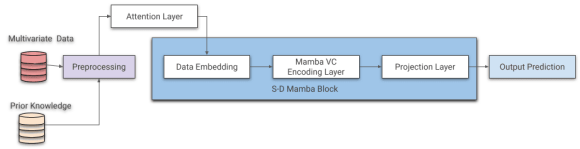


Figure 5: Illustration of Experiment 4.

Experiment 5 & 6: Selective Attention on Prior Knowledge

- **Approach:** Attention is applied **only on the prior knowledge features** (`GPP_pred`, `ET_pred`), allowing the model to selectively weigh them based on temporal context while keeping the rest of the features unchanged. Prior knowledge features are projected to the model dimension using a dedicated linear layer before attention is applied.

- **Architecture:**
  - **Experiment 5:** All features go through embedding layer, but attention is applied only to the projected prior knowledge features. The output is then combined with the other features before passing through the encoder (see Figure 6).
  - **Experiment 6:** The prior knowledge and raw features are projected separately using dedicated layers (`prior_projection` and `input_projection`). Attention is applied only to the prior knowledge, and both representations are then fused and passed directly to the encoder without additional embedding (see Figure 7).

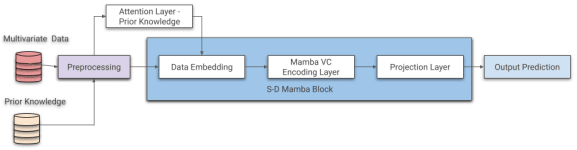


Figure 6: Illustration of Experiment 5.

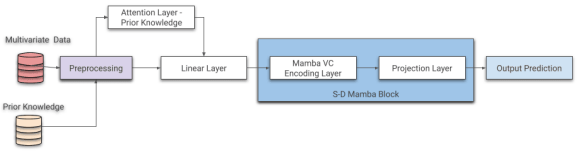


Figure 7: Illustration of Experiment 6.

This diverse set of experiments allows us to systematically explore:

- Whether prior knowledge improves model performance.
- Which integration strategy (feature vs. attention) is most effective.

These integration strategies are evaluated under two experimental scenarios:

- **In-Distribution (ID) Forecasting:** The model is trained and tested on data from the same forest site. This assesses the model's ability to learn and adapt to site-specific patterns and to interpret prior knowledge in a consistent environment.

- **Out-of-Distribution (OOD) Forecasting:** The model is trained on a subset of forest sites and evaluated on entirely unseen regions. This setup tests the generalization capacity of S-Mamba and the extent to which prior knowledge aids in transferring learned representations to novel ecological settings.

Table 1: Dataset Summary

Dataset	Features	Timesteps	Location
results_station_12	10	2558	Sorø,Denmark
results_station_21	10	2557	Hyytiälä,Finland

Optimization and Regularization Strategy

In order to optimize the performance of the model and ensure robust generalization, a carefully designed training pipeline was implemented. This incorporates both effective optimization techniques and regularization strategies. The model has been trained using the AdamW optimizer, a variant of the Adam optimizer that decouples weight decay from the gradient update process. This has been shown to result in superior regularization compared to the standard Adam optimizer. The initial learning rate is set to 1e−4 and is adjusted throughout training using a cosine annealing scheduler, which gradually decreases the learning rate to a minimum of 1e−6, promoting smoother convergence and helping avoid sharp local minima. In order to further prevent overfitting, early stopping with a patience of 10 epochs was employed, allowing the training to be terminated if the validation loss did not improve over a defined number of iterations. The Mean Squared Error (MSE) is utilized as the principal loss function to facilitate the training process, which is commonly used by most TSF models. In addition to standard optimization, we incorporated hyperparameter optimization (HPO) using the `Optuna` framework- TPE (Tree-structured Parzen Estimator) (Akiba et al., 2019). Specifically, we performed a search over `learning rate, drop out rate, batch size, and the number of attention heads` using validation loss as the objective metric. Multiple runs were conducted with different random seeds to ensure robustness and reduce the influence of random initialization. After training, the top 5 models with the lowest validation losses were selected. The convergence rate, which measures how quickly the model's validation loss decreased, was tracked during the training process, and early stopping was applied to prevent overfitting by halting training when no improvement was observed in validation loss over the set patience period. The best models were then evaluated on the test set, and their predictions were averaged to improve generalization. This approach, combining early stopping, convergence monitoring, and averaging predictions from multiple models, ensured that the final model was both well-regularized and robust to unseen data.

Results

In-Distribution Forecasting

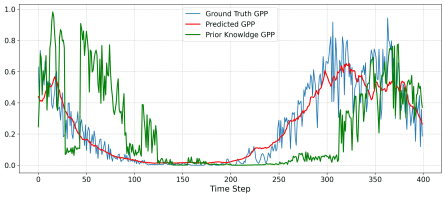
The look back length is set to 64 and the forecast length is set to 7, 30, and 64. Table 2 and Table 3 shows overall performance of our experiments across different prediction lengths. We use the widely adopted MAE, RMSE, MSE (±) STD to evaluate the model's performance. The lower values among each experiment with respect to prediction length are in bold and best value for each prediction length highlighted in red font. Figure 8 and 9 are the comparison of GPP and ET values across all experiments for forecast length of 7-day predictions from the model and stitches them together into a continuous time series. It aligns each prediction with the correct time step, averages overlapping predictions, and creates a complete forecast for the test period.

Table 2: ID forecasting for `results_station_12.csv` performance of GPP.

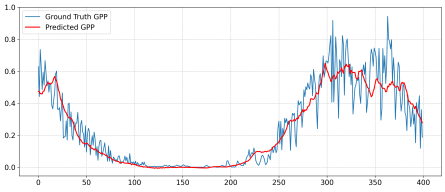
Experiments	Length	Exp 1			Exp 2			Exp 3			Exp 4			Exp 5			Exp 6		
Metrics		MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD
With Prior Knowledge	7	0.0621	0.0933	0.0087 ± 0.019	0.0678	0.0985	0.0097 ± 0.021	0.0616	0.0936	0.0087 ± 0.019	0.0617	0.0888	0.0078 ± 0.017	0.0587	0.0894	0.0080 ± 0.018	0.0718	0.1010	0.0102 ± 0.020
	30	0.0678	0.1008	**0.0101 ± 0.022**	0.0678	0.1018	**0.0103 ± 0.022**	0.0689	0.1022	0.0104 ± 0.023	0.0760	0.1038	0.0107 ± 0.020	0.0649	0.0961	0.0092 ± 0.020	0.0708	0.1056	0.0111 ± 0.023
	64	0.0678	0.1033	**0.0106 ± 0.024**	0.0702	0.1043	**0.0108 ± 0.023**	0.0676	0.1019	0.0103 ± 0.023	0.0710	0.1091	0.0119 ± 0.026	0.0679	0.1021	0.0104 ± 0.023	0.0680	0.1002	0.0100 ± 0.020
Without Prior Knowledge	7	0.0609	0.0943	**0.0083 ± 0.018**	0.0660	0.0964	**0.0093 ± 0.020**	0.0625	0.0909	**0.0082 ± 0.017**	0.0631	0.0912	0.0083 ± 0.017	-	-	-	-	-	-
	30	0.0687	0.1029	0.0105 ± 0.023	0.0714	0.1069	0.0114 ± 0.024	0.0668	0.0995	**0.0091 ± 0.021**	0.0657	0.0959	**0.0092 ± 0.019**	-	-	-	-	-	-
	64	0.0682	0.1032	0.0106 ± 0.024	0.0736	0.1083	0.0117 ± 0.024	0.0671	0.1018	**0.0103 ± 0.023**	0.0716	0.1054	**0.0111 ± 0.024**	-	-	-	-	-	-

Table 3: ID forecasting for `results_station_12.csv` performance of ET.

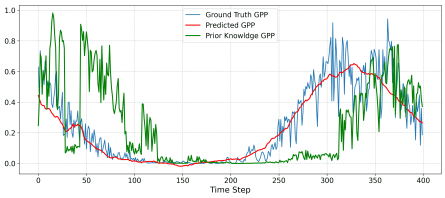
Experiments / Metrics	Length	Exp 1			Exp 2			Exp 3			Exp 4			Exp 5			Exp 6		
		MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD
With Prior Knowledge	**7**	0.0647	0.0932	**0.0087 ± 0.018**	0.0640	0.0903	**0.0081 ± 0.016**	0.0639	0.0939	0.0088 ± 0.016	0.0633	0.0937	**0.0088 ± 0.018**	0.0604	0.0899	0.0080 ± 0.017	0.0677	0.0948	0.0089 ± 0.017
	30	0.0678	0.0975	**0.0095 ± 0.018**	0.0656	0.0965	0.0093 ± 0.020	0.0675	0.0972	0.0094 ± 0.018	0.0841	0.1091	0.0119 ± 0.019	0.0692	0.0984	0.0096 ± 0.019	0.0658	0.0960	0.0092 ± 0.019
	64	0.0673	0.0980	0.0096 ± 0.021	0.0671	0.0980	**0.0096 ± 0.021**	0.0684	0.0985	0.0097 ± 0.020	0.0652	0.0964	0.0093 ± 0.020	0.0674	0.0977	0.0095 ± 0.020	0.0674	0.0980	0.0096 ± 0.020
Without Prior Knowledge	**7**	0.0637	0.0946	0.0089 ± 0.018	0.0633	0.0904	0.0082 ± 0.017	0.0644	0.0925	**0.0085 ± 0.017**	0.0706	0.0981	0.0096 ± 0.018	-	-	-	-	-	-
	30	0.0675	0.0978	0.0095 ± 0.019	0.0663	0.0961	**0.0092 ± 0.020**	0.0656	0.0950	0.0090 ± 0.018	0.0705	0.0997	**0.0099 ± 0.019**	-	-	-	-	-	-
	64	0.0662	0.0966	**0.0093 ± 0.020**	0.0689	0.1006	0.0101 ± 0.022	0.0671	0.0972	**0.0094 ± 0.020**	0.0678	0.0962	0.0092 ± 0.019	-	-	-	-	-	-



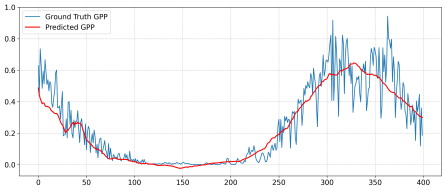
(a) Experiment 1 - With Prior Knowledge



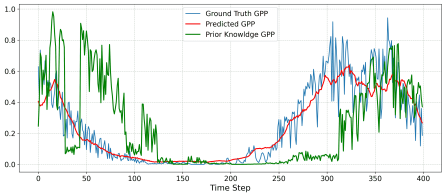
(b) Experiment 1 - Without Prior Knowledge



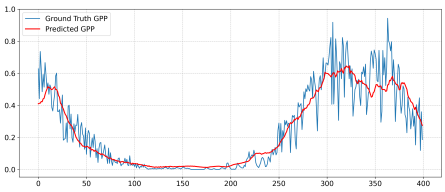
(c) Experiment 2 - With Prior Knowledge



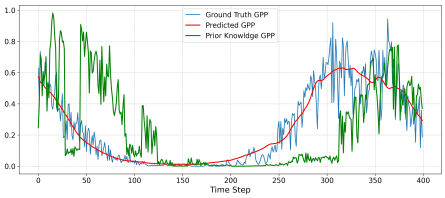
(d) Experiment 2 - Without Prior Knowledge



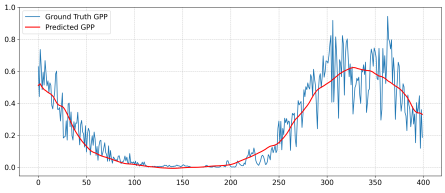
(e) Experiment 3 - With Prior Knowledge



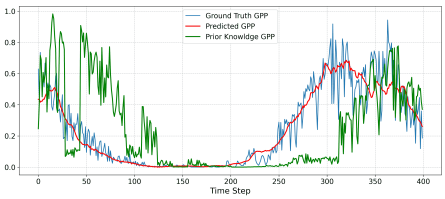
(f) Experiment 3 - Without Prior Knowledge



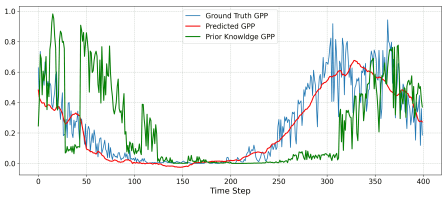
(g) Experiment 4 - With Prior Knowledge



(h) Experiment 4 - Without Prior Knowledge

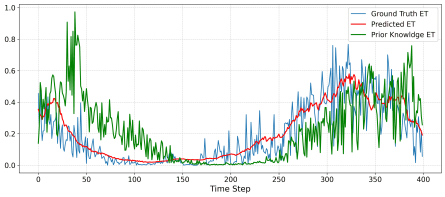


(i) Experiment 5 - With Prior Knowledge

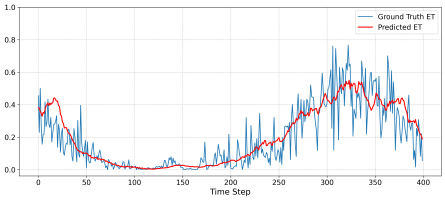


(j) Experiment 6 - With Prior Knowledge

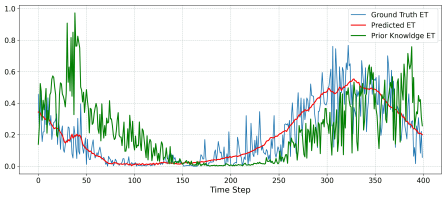
Figure 8: Comparison of GPP with and without prior knowledge for length 7 across all experiments.



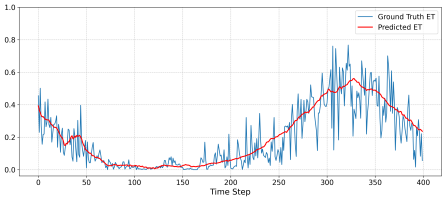
(a) Experiment 1 - With Prior Knowledge



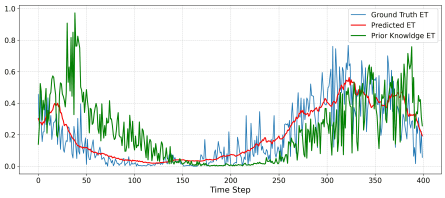
(b) Experiment 1 - Without Prior Knowledge



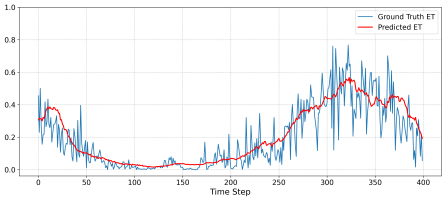
(c) Experiment 2 - With Prior Knowledge



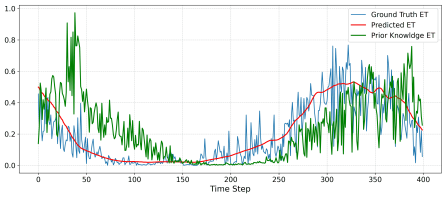
(d) Experiment 2 - Without Prior Knowledge



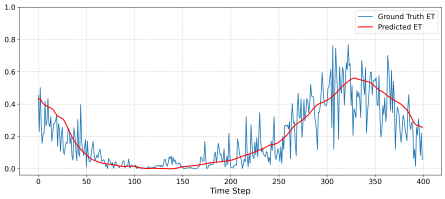
(e) Experiment 3 - With Prior Knowledge



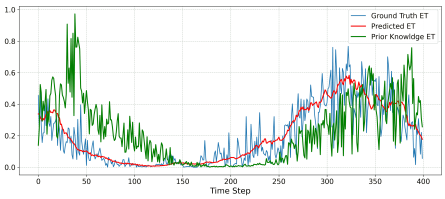
(f) Experiment 3 - Without Prior Knowledge



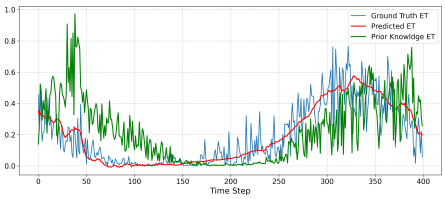
(g) Experiment 4 - With Prior Knowledge



(h) Experiment 4 - Without Prior Knowledge

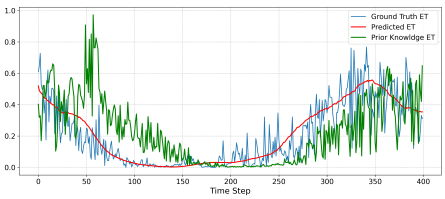


(i) Experiment 5 - With Prior Knowledge

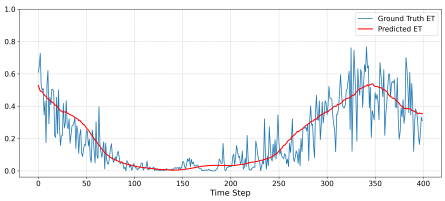


(j) Experiment 6 - With Prior Knowledge

Figure 9: Comparison of ET with and without prior knowledge for length 7 across all experiments.

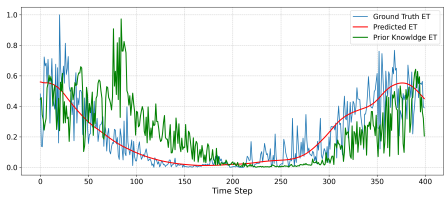


(a) Experiment 3 - With Prior Knowledge

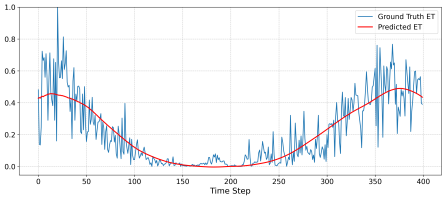


(b) Experiment 3 - Without Prior Knowledge

Figure 10: Comparison of ET for length 30.



(a) Experiment 4 - With Prior Knowledge



(b) Experiment 4 - Without Prior Knowledge

Figure 11: Comparison of ET for length 64.

GPP forecasting

From all experiments in Table 2, Experiment 4 with prior knowledge at 7 length achieved the best MSE of 0.0078 ± 0.017. As the forecast length increases to 30 , the best performance was with Experiment 5 with prior knowledge with MSE of 0.0092 ± 0.020 and at 64 length in Experiment 6 with prior knowledge with MSE of 0.0100 ± 0.020. Figure 8 (g) and (h) shows at time steps 350 to 400 the prior knowledge is helping the model to capture the values. Without prior knowledge, the model shows slightly worse results indicating less stable predictions.

ET forecasting

From Table 3, we observe that Experiment 5 with prior knowledge at 7 days achieved the best MSE of 0.0080 ± 0.017. However, for longer forecast lengths 30 and 64 , the without prior knowledge setup outperformed the with prior knowledge experiments. For the 30-day forecast, Experiment 3 without prior knowledge achieved the best MSE of 0.0090 ± 0.018, and for 64 days, Experiment 4 without prior knowledge performed the best with an MSE of 0.0092 ± 0.019. Figure 10 shows the comparison of experiments 3 at length 30 with and without prior knowledge where at time steps 100 to 250 the prior knowledge is making the prediction worse. Similarly Figure 11 shows the comparison of experiments 4 for length 64 with and without prior knowledge where prior knowledge has minimal influence to support the prediction.

Out-of-Distribution Forecasting

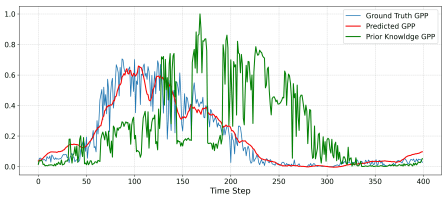
The look back length is set to 64 and the forecast length is set to 7, 30, and 64. Table 4 and Table 5 shows overall performance of our experiments across different prediction lengths. The lower values among each experiment with respect to prediction length are in bold and best value for each prediction length highlighted in red font. Figure 12 and 13 are the comparison of GPP and ET values across all experiments for forecast length of 7.

Table 4: OOD forecasting for train on `results_station_12.csv` and test on `results_station_21.csv` performance for GPP.

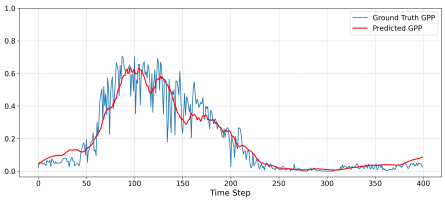
Experiments	Length	Exp 1			Exp 2			Exp 3			Exp 4			Exp 5			Exp 6		
Metrics		MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD
With Prior Knowledge	7	0.0615	0.0887	0.0079 ± 0.016	0.0872	0.1107	0.0122 ± 0.018	0.0626	0.0901	**0.0081 ± 0.015**	0.0840	0.1121	**0.0126 ± 0.020**	0.0723	0.1022	0.0105 ± 0.021	0.1025	0.1338	0.0184 ± 0.027
	30	0.0766	0.1079	0.0117 ± 0.021	0.0837	0.1145	0.0131 ± 0.024	0.0839	0.1187	0.0141 ± 0.026	0.1263	0.1715	**0.0299 ± 0.045**	0.0785	0.1108	0.0123 ± 0.022	0.1002	0.1297	0.0169 ± 0.025
	64	0.0821	0.1153	0.0133 ± 0.025	0.0920	0.1287	**0.0166 ± 0.032**	0.0833	0.1158	**0.0134 ± 0.024**	0.1183	0.1738	0.0308 ± 0.067	0.0822	0.1157	0.0134 ± 0.025	0.1100	0.1494	0.0223 ± 0.037
Without Prior Knowledge	7	0.0648	0.0960	0.0093 ± 0.019	0.0807	0.1076	**0.0116 ± 0.020**	0.0630	0.0919	0.0084 ± 0.016	0.1287	0.1562	0.0245 ± 0.028	-	-	-	-	-	-
	30	0.0774	0.1092	0.0120 ± 0.021	0.0852	0.1123	**0.0126 ± 0.021**	0.0808	0.1138	**0.0130 ± 0.024**	0.1274	0.1812	0.0328 ± 0.056	-	-	-	-	-	-
	64	0.0811	0.1145	0.0131 ± 0.025	0.0935	0.1314	0.0173 ± 0.034	0.0818	0.1166	0.0136 ± 0.025	0.1225	0.1630	**0.0266 ± 0.038**	-	-	-	-	-	-

Table 5: OOD forecasting for train on `results_station_12.csv` and test on `results_station_21.csv` performance for ET.

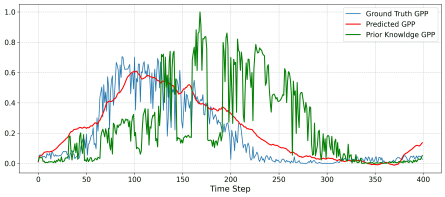
Experiments	Length	Exp 1			Exp 2			Exp 3			Exp 4			Exp 5			Exp 6		
Metrics		MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD	MAE	RMSE	MSE (±) STD
With Prior Knowledge	7	0.0619	0.0892	**0.0080 ± 0.017**	0.0871	0.1166	0.0136 ± 0.025	0.0635	0.0897	0.0080 ± 0.017	0.0804	0.1110	**0.0126 ± 0.022**	0.0655	0.0898	0.0081 ± 0.015	0.1031	0.1338	0.0189 ± 0.029
	30	0.0775	0.1092	0.0120 ± 0.023	0.0866	0.1163	0.0136 ± 0.025	0.0769	0.1107	0.0123 ± 0.024	0.1182	0.1550	**0.0242 ± 0.036**	0.0725	0.1034	0.0107 ± 0.021	0.0991	0.1303	0.0170 ± 0.029
	64	0.0763	0.1086	0.0119 ± 0.024	0.0850	0.1164	**0.0136 ± 0.027**	0.0802	0.1122	0.0126 ± 0.024	0.1136	0.1593	0.0262 ± 0.048	0.0773	0.1084	0.0118 ± 0.023	0.0931	0.1279	0.0164 ± 0.030
Without Prior Knowledge	7	0.0644	0.0922	0.0086 ± 0.018	0.0824	0.1144	**0.0132 ± 0.026**	0.0619	0.0879	0.0077 ± 0.016	0.1188	0.1455	0.0214 ± 0.030	-	-	-	-	-	-
	30	0.0732	0.1037	**0.0108 ± 0.021**	0.0876	0.1151	**0.0133 ± 0.024**	0.0693	0.0990	**0.0098 ± 0.020**	0.1171	0.1664	0.0277 ± 0.046	-	-	-	-	-	-
	64	0.0740	0.1049	0.0111 ± 0.022	0.0886	0.1224	0.0150 ± 0.031	0.0730	0.1063	**0.0113 ± 0.023**	0.1144	0.1522	**0.0232 ± 0.035**	-	-	-	-	-	-



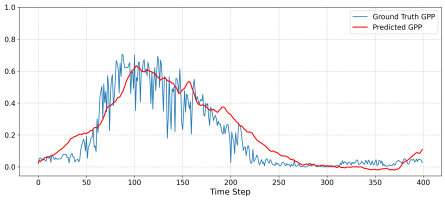
(a) Experiment 1 - With Prior Knowledge



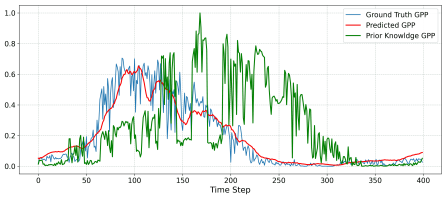
(b) Experiment 1 - Without Prior Knowledge



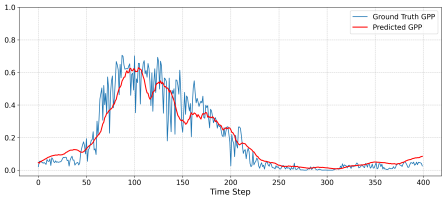
(c) Experiment 2 - With Prior Knowledge



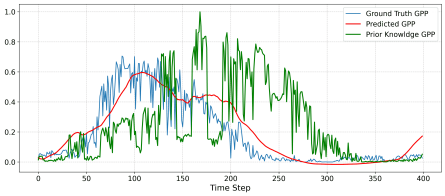
(d) Experiment 2 - Without Prior Knowledge



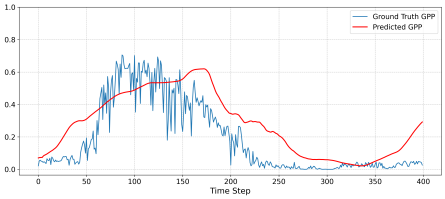
(e) Experiment 3 - With Prior Knowledge



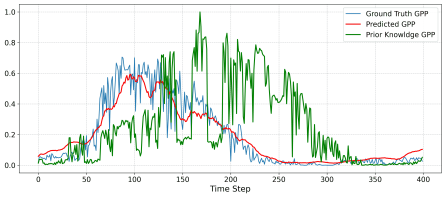
(f) Experiment 3 - Without Prior Knowledge



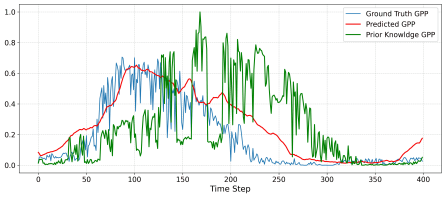
(g) Experiment 4 - With Prior Knowledge



(h) Experiment 4 - Without Prior Knowledge

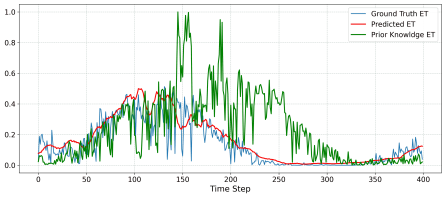


(i) Experiment 5 - With Prior Knowledge

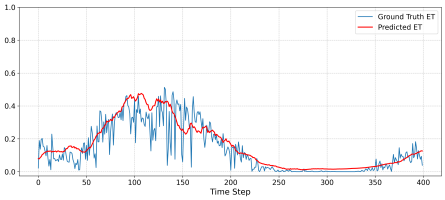


(j) Experiment 6 - With Prior Knowledge

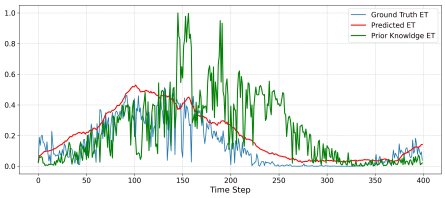
Figure 12: Comparison of GPP with and without prior knowledge for length 7 across all experiments.



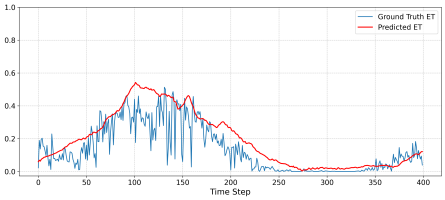
(a) Experiment 1 - With Prior Knowledge



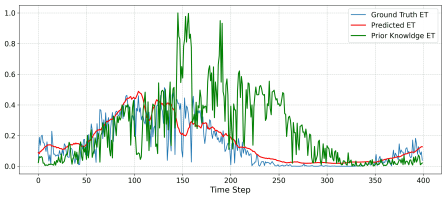
(b) Experiment 1 - Without Prior Knowledge



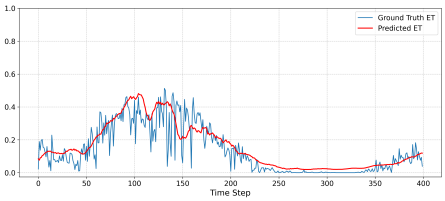
(c) Experiment 2 - With Prior Knowledge



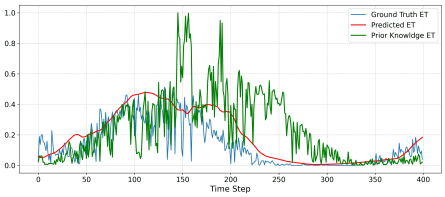
(d) Experiment 2 - Without Prior Knowledge



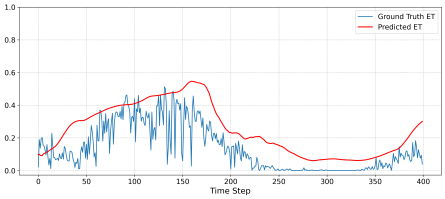
(e) Experiment 3 - With Prior Knowledge



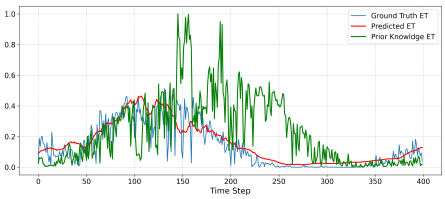
(f) Experiment 3 - Without Prior Knowledge



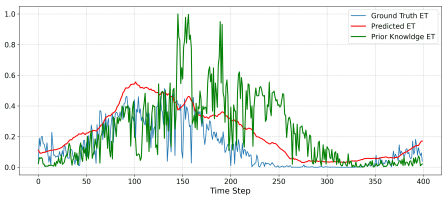
(g) Experiment 4 - With Prior Knowledge



(h) Experiment 4 - Without Prior Knowledge

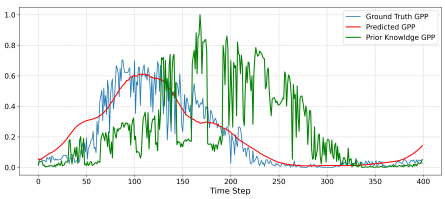


(i) Experiment 5 - With Prior Knowledge

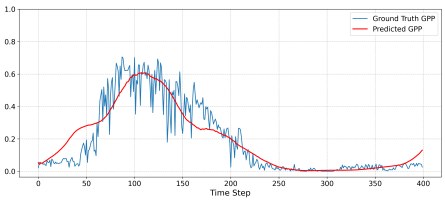


(j) Experiment 6 - With Prior Knowledge

Figure 13: Comparison of ET with and without prior knowledge for length 7 across all experiments.

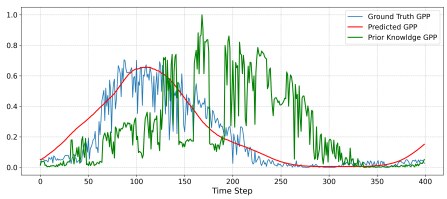


(a) Experiment 1 - With Prior Knowledge

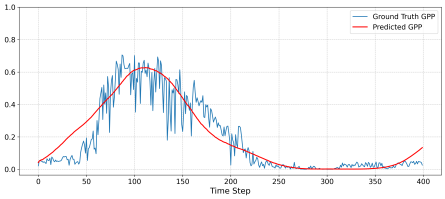


(b) Experiment 1 - Without Prior Knowledge

Figure 14: Comparison of GPP for length 30.

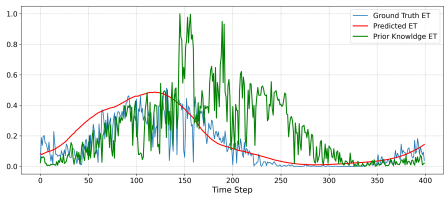


(a) Experiment 1 - With Prior Knowledge

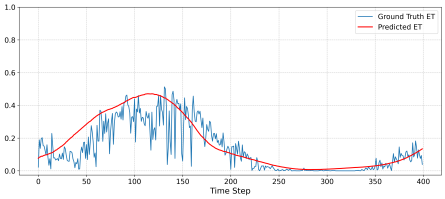


(b) Experiment 1 - Without Prior Knowledge

Figure 15: Comparison of GPP for length 64.



(a) Experiment 1 - With Prior Knowledge



(b) Experiment 1 - Without Prior Knowledge

Figure 16: Comparison of ET for length 64.

GPP forecasting

From Table 4, for short prediction length 7, Experiments 1 perform best with prior knowledge, achieving the lowest MSE. At length 30, Experiments 1 again benefit from prior knowledge, while Experiment 5 performs competitively. At prediction length 64, Experiment 1 without prior knowledge performs comparably to Experiment 1, 3 and 5 with prior knowledge. Figure 14 and 15 shows best performance Experiment comparison of length 30 and 64 respectively where at length 30 for time steps around 150 to 200 prior knowledge is guiding the prediction towards GPP values whereas for much longer length of 64 it prior knowledge has minimal effect on supporting GPP predictions.

ET Forecasting

From Table 5, for short prediction length 7, models without prior knowledge on Experiments 3 achieve the best performance. However, Experiment 1,3,5 with prior knowledge yields competitive performance at this length. For medium horizons (30), Experiments 5 tend to perform better with prior knowledge and at prediction length 64, Experiment 1 without prior knowledge is the top performer with the lowest MSE. Figure 13 (a) and (b) and 16 shows best performance experiment 1 for length 7 and 64 respectively where prior knowledge has minimal influence to support the prediction.

Discussion

Overall performance of model

The results presented for both In-Distribution (ID) and Out-of-Distribution (OOD) tasks show the influence of prior knowledge and attention mechanisms on GPP (Gross Primary Production) and ET (Evapotranspiration) forecasting.GPP predictions were more accurate than ET predictions, as GPP is influenced by fewer, more predictable factors like light, temperature, and leaf area. In contrast, ET is impacted by more complex factors such as soil moisture, atmospheric conditions, and plant characteristics, leading to higher uncertainty.

Impact of Prior Knowledge

The results of the experiments showed mixed outcomes with prior knowledge integration into the Mamba framework. It is important to contextualize the impact of prior knowledge by recognizing the inherent approximation error present in the prior itself. The error values ahead are `GPP` and `GPP_pred` MSE 0.0138 $\pm$ 0.12 , `ET` and `ET_pred` MSE 0.0102 $\pm$ 0.10. These error levels illustrate that the prior knowledge—while grounded in domain expertise—is itself an imperfect estimate. This explains why prior knowledge integration improved short-term GPP forecasting, where the model could still effectively leverage structural ecological cues. However, for ET, the smaller but still notable deviation from true values may have introduced conflicting signals, especially in longer horizons, leading to less consistent performance or even degradation. These findings suggest that the effectiveness of prior knowledge integration is context-dependent, offering advantages in some cases and no improvement or worse performance in others.

Attention Mechanisms vs. Direct Integration

We explored two strategies for integrating prior knowledge: direct feature integration and attention mechanisms. While attention mechanisms improved GPP prediction for in-distribution tasks, their impact was less favorable for out-of-distribution tasks, sometimes even worsening performance. For ET prediction, however, attention mechanisms were more favorable than direct integration, as they allowed the model to focus on rele-

vant temporal and environmental features, improving its ability to handle the complexity and variability of ET. Similarly, direct integration showed mixed results, enhancing short-term GPP forecasting for out-of-distribution tasks but not consistently improving long-term predictions.

Performance Comparison: In-Distribution vs. Out-of-Distribution

Prior knowledge integration provided more benefits for in-distribution tasks, enhancing the model's generalization in familiar conditions. However, for out-of-distribution tasks with unseen data, it did not consistently improve performance and, in some cases, even worsened results. This suggests that prior knowledge is more effective in known scenarios than in novel environments.

Effectiveness Across Time Horizons

Prior knowledge improved short-term forecasting - 7-day predictions, stabilizing the model's predictions. However, as the forecast length increased to 30- and 64-day horizons, its benefits diminished, and simpler models without prior knowledge sometimes outperformed more complex ones in most cases.

Conclusion and Key Takeaways

This study demonstrated that integrating prior knowledge into the Mamba model for time series forecasting yielded mixed results. While it improved short-term predictions, particularly for GPP, its benefits diminished for longer horizons and in out-of-distribution tasks. Attention mechanisms generally provided better results for ET forecasting. These findings suggest that prior knowledge can enhance model performance in certain scenarios, but its effectiveness is highly context-dependent.

Future Work

To improve the effectiveness of prior knowledge integration, future research should focus on refining the quality and relevance of the prior knowledge used. This report presents preliminary experiments exploring the impact of incorporating prior knowledge across various forecasting architectures for both in-distribution and out-of-distribution tasks, with an emphasis on GPP and ET predictions. Expanding the study to include diverse datasets and more complex environmental conditions will further optimize the integration process and enhance its generalization across different domains.

References

Akiba, Takuya et al. (2019). "Optuna: A Next-generation Hyperparameter Optimization Framework". In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 2019, pp. 2623–2631. DOI: 10.1145/3292500.3330701. URL: https://doi.org/10.1145/3292500.3330701.

Chen, Zhenfeng et al. (2023). "Long sequence time-series forecasting with deep learning: A survey". In: *Information Fusion* 97, p. 101819. ISSN: 1566-2535. DOI: 10.1016/j.inffus.2023.101819. URL: https://doi.org/10.1016/j.inffus.2023.101819.

Fang, Jie & Lin, Jianwu (2020). "Prior knowledge distillation based on financial time series". In: *arXiv preprint arXiv:2006.09247*. URL: https://arxiv.org/abs/2006.09247.

Gu, A. & Dao, T. (2024). "Mamba: Linear-time sequence modeling with selective state spaces". In: *First Conference on Language Modeling*. [Preprint]. URL: https://doi.org/10.48550/arXiv.2312.00752.

Gu, Albert & Goel, Kushal & Ré, Christopher (2021a). "Efficiently modeling long sequences with structured state spaces". In: *arXiv preprint arXiv:2111.00396*. [Preprint]. URL: https://arxiv.org/abs/2111.00396.

Gu, Albert et al. (2020). "HiPPO: Recurrent memory with optimal polynomial projections". In: *arXiv preprint arXiv:2008.07669*. [Preprint]. URL: https://arxiv.org/abs/2008.07669.

Gu, Albert et al. (2021b). "Combining recurrent, convolutional, and continuous-time models with linear state-space layers". In: *Advances in Neural Information Processing Systems (NeurIPS)*. URL: https://doi.org/10.48550/arXiv.2110.13985.

He, K. et al. (2023). "Financial time series forecasting with the deep learning ensemble model". In: *Mathematics* 11.4, p. 1054. DOI: 10.3390/math11041054.

Jung, S. et al. (2021). "Self-attention-based deep learning network for regional influenza forecasting". In: *IEEE Journal of Biomedical and Health Informatics* 25.12, pp. 4554–4563. DOI: 10.1109/JBHI.2021.3093897.

Peltoniemi, Mikko et al. (2015). "A semi-empirical model of boreal-forest gross primary production, evapotranspiration, and soil water—Calibration and sensitivity analysis". In: *Boreal Environment Research* 20, pp. 151–169. URL: https://jukuri.luke.fi/bitstream/handle/10024/485901/peltoniemi.pdf.

Shao, Zhiqi et al. (2025). "ST-MambaSync: Complement the power of Mamba and Transformer fusion for less computational cost in spatial–temporal traffic forecasting". In: *Information Fusion*. DOI: 10.1016/j.inffus.2024.102872.

Smith, J.T. & Warrington, A. & Linderman, S.W. (2022). "Simplified state space layers for sequence modeling". In: *arXiv preprint arXiv:2208.04933*. URL: https://arxiv.org/abs/2208.04933.

Wang, Z. et al. (2024). "Is Mamba effective for time series forecasting?" In: *arXiv preprint arXiv:2403.11144*. [Preprint]. URL: https://arxiv.org/abs/2403.11144.

Yu, Xue & Yang, Zhi & Wang, Xia, et al. (2024). "A prior-knowledge-guided dynamic attention mechanism to predict nocturnal hypoglycemic events in type 1 diabetes". In: *BMC Medical Informatics and Decision Making* 24.1, p. 378. DOI: 10.1186/s12911-024-02761-3. URL: https://doi.org/10.1186/s12911-024-02761-3.

Appendix

Dataset Overview. Table 1 summarizes the characteristics of the datasets used. We focused on two stations—Station 12 and Station 21—due to their consistent availability of key eco-hydrological variables, and to enable comparative cross-station forecasting. Both datasets were preprocessed to handle missing values using forward fill followed by interpolation. All features were normalized using MinMax scaling. Only samples with full-length sequences were retained, and no categorical variables were included. The use of only two stations in this study allowed controlled experimentation for evaluating generalization across varying eco-climatic regimes, serving as a foundational step before scaling to multi-station or regional-level models.

Attention Mechanism. We used multi-head self-attention with 4 heads across all attention-based experiments (Exp. 3–6). The key, query, and value matrices were learned from the projected feature representations with a dimensionality equal to the model's hidden size ($d_{\text{model}} = 512$). Each head independently performed scaled dot-product attention, and outputs were concatenated and linearly transformed. This mechanism enables the model to attend to both temporal and feature-level dependencies adaptively.

Full hyperparameter search space and best configuration found by Optuna. **Search Space:** `num_heads=[1, 2, 4, 8]`, `d_model=[128, 256, 512]`, `dropout=[0.1, 0.2, 0.3]`, `batch_size=[16, 32, 64]`

Best Trial: `num_heads=4, d_model=512, dropout=0.1, batch_size=32`

Table A1 and A2 shows the convergence rate and early stopping epochs for each experiment and prediction length.

Table A1: In-distribution: Convergence rate and early stopping epochs for each experiment and prediction length.

Experiments Metrics	Length	Exp 1		Exp 2		Exp 3		Exp 4		Exp 5		Exp 6	
		Conv Rate	Epochs	Conv Rate	Epochs	Conv Rate	Epochs	Conv Rate	Epochs	Conv Rate	Epochs	Conv Rate	Epochs
With Prior Knowledge	7	-0.001190	31.00	-0.003549	20.80	-0.000969	29.00	-0.005666	31.20	-0.001150	30.20	-0.002533	17.80
	30	-0.001430	22.20	-0.003368	20.60	-0.001192	28.20	-0.007373	22.00	-0.001352	23.20	-0.001554	29.80
	64	-0.000991	26.20	-0.002682	23.20	-0.000944	24.60	-0.004635	40.00	-0.000986	28.60	-0.001136	28.40
Without Prior Knowledge	7	-0.001115	29.20	-0.002629	21.20	-0.000896	32.00	-0.006917	24.00	-	-	-	-
	30	-0.001357	24.20	-0.003009	20.20	-0.000952	26.60	-0.009596	17.40	-	-	-	-
	64	-0.001007	25.20	-0.003022	22.60	-0.000858	25.20	-0.010254	24.40	-	-	-	-

Table A2: Out of distribution: Convergence rate and early stopping epochs for each experiment and prediction length.

Experiments Metrics	Length	Exp 1		Exp 2		Exp 3		Exp 4		Exp 5		Exp 6	
		Conv Rate	Epochs	Conv Rate	Epochs	Conv Rate	Epochs	Conv Rate	Epochs	Conv Rate	Epochs	Conv Rate	Epochs
With Prior Knowledge	7	-0.001283	26.40	-0.002752	19.60	-0.001052	23.60	-0.003896	37.20	-0.001112	27.60	-0.001356	17.60
	30	-0.000836	23.00	-0.003185	20.40	-0.000580	26.40	-0.003605	35.40	-0.001068	23.60	-0.001594	24.80
	64	-0.000869	27.00	-0.003320	22.40	-0.000772	27.40	-0.004287	39.20	-0.000854	25.60	-0.001323	29.60
Without Prior Knowledge	7	-0.001201	28.00	-0.002217	21.40	-0.001010	24.60	-0.004756	24.00	-	-	-	-
	30	-0.000984	22.60	-0.002601	21.20	-0.000673	27.40	-0.005991	22.60	-	-	-	-
	64	-0.000841	26.20	-0.003132	22.20	-0.000757	30.60	-0.007782	23.80	-	-	-	-