Beyond Statistical Analysis: Multimodal Framework for Time Series Forecasting with LLM-Driven Temporal Pattern

Jiahong Xiong[1], **Chengsen Wang**[1], **Haifeng Sun**[1*], **Yuhan Jing**[1], **Qi Qi**[1], **Zirui Zhuang**[1], **Lei Zhang**[2], **Jianxin Liao**[1] and **Jingyu Wang**[1*]
[1]State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications
[2]China Unicom Network Communications Corporation Limited
{xiongjiahong, cswang, hfsun, jingyh, qiqi8266, zhuangzirui}@bupt.edu.cn, zhangl83@chinaunicom.cn,
{liaojx, wangjingyu}@bupt.edu.cn

Abstract

Accurate forecasting of time series is crucial for many applications in the real world. Conventional methods primarily rely on statistical analysis of historical data, often leading to overfitting and failing to account for background information and constraints imposed by external events. Therefore, introducing large language models (LLMs) with robust textual capabilities holds significant potential. However, due to the inherent limitations of LLMs in handling numerical data, they do not exhibit advantages in precise numerical prediction tasks. Therefore, we propose a framework to integrate LLMs with conventional methods synergistically. Rather than directly outputting numerical predictions, we leverage the capabilities of the LLMs to generate textual temporal patterns, thereby fully utilizing their inherent knowledge and reasoning abilities. Additionally, we introduce a memory network designed to decode these textual representations into a format that numerical models can effectively interpret. This approach not only capitalizes on the strengths of the LLM in text processing but also bridges the gap between textual and numerical data, enhancing the overall predictive performance of the model. Our experimental results demonstrate the framework's effectiveness, achieving state-of-the-art performance on various benchmark datasets.

1 Introduction

As a critical task, time series forecasting holds extensive practical application value across various domains such as meteorology, power systems, transportation, and finance [Shao *et al.*, 2022; Choi *et al.*, 2022; Wang *et al.*, 2023; Guo *et al.*, 2023]. In recent years, researchers have proposed a variety of neural network models aimed at capturing detailed features in time series data, including periodicity, scale, fluctuations, and trends [Qiu *et al.*, 2025; Wang *et al.*, 2024a; Nie *et al.*, 2022; Wu *et al.*, 2022; Wu *et al.*, 2021], achieving notable success.

*corresponding author

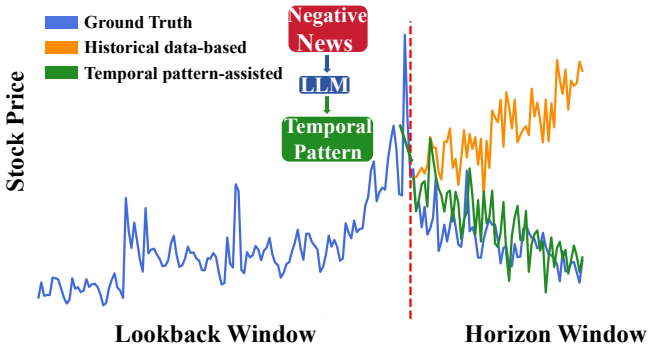

Figure 1: Relying only on historical data leads to poor predictions, while LLM-driven temporal pattern text improves predictions.

However, conventional time series forecasting methods primarily rely on the statistical analysis of historical sequences [Qiu *et al.*, 2024], which exhibit two significant limitations. Firstly, These models are inherently susceptible to overfitting, a tendency underscored by the observation that simple linear models can achieve performance levels comparable to those of intricate network architectures [Xu *et al.*, 2024]. Secondly, the models fail to account for the constraints imposed by contextual information and external events, which are pivotal in real-world predictive tasks. For instance, power demand and traffic flow often exhibit different temporal patterns across various periods, while external events significantly influence financial scenarios, as shown in Figure 1. Such information typically exists in textual form, necessitating models with multimodal modelling capabilities for effective integration and reasoning. It is worth noting that advancements in computer vision (CV) and natural language processing (NLP) have demonstrated that pre-trained large language models (LLMs) exhibit exceptional performance across various downstream tasks, benefiting from their rich knowledge reserves, powerful pattern recognition capabilities, and complex semantic reasoning abilities. Moreover, these models can effectively integrate and utilize knowledge from different modalities for collaborative forecasting in complex scenarios. Therefore, applying powerful LLMs to the field of time series forecasting holds vast research

prospects and application potential.

In the research on integrating LLMs into time series forecasting, one strategy [Xue and Salim, 2023] involves combining numerical data with formatted text and directly utilizing LLMs as the core output for predicting specific numerical values. This approach demonstrates certain performance advantages in zero-shot learning scenarios, effectively leveraging LLMs' text-processing capabilities. However, due to the inherent limitations of LLMs in handling pure numerical data [Wei *et al.*, 2022; Rae *et al.*, 2021], they do not exhibit significant advantages over conventional methods in precise numerical prediction tasks. The second strategy [Jin *et al.*, 2023; Sun *et al.*, 2023] involves using the backbone networks of pre-trained LLMs for feature extraction. Although this method can achieve state-of-the-art forecasting performance in some scenarios, recent studies [Tan *et al.*, 2024] indicate that the predictive performance of LLMs does not significantly surpass and may even fall short of conventional attention-based models. The advantages of such methods primarily rely on the massive network architecture and parameter scale rather than effectively utilizing the text processing capabilities acquired through the pre-training of LLMs. This is because the pre-training of LLMs does not naturally include knowledge and reasoning abilities for pure numerical data [Jin *et al.*, 2023]. Recent advancements in multimodal research [Liu *et al.*, 2024b; Wang *et al.*, 2024c; Wang *et al.*, 2025] have predominantly focused on constructing datasets rather than the more nuanced integration with LLMs. Based on the above analysis, current research methods overly emphasize strategies that use LLMs for pure numerical data processing, which are inefficient. Instead, focusing more on the core strengths of LLMs, namely their text processing capabilities, is the key to effectively integrating LLMs with time series forecasting.

Drawing on the aforementioned observations, we propose a novel perspective on integrating LLMs with time series forecasting, which is grounded in the forecasting of conventional methods complemented by the time patterns forecasted by LLMs based on their textual capabilities. This approach aims to construct a forecasting framework that not only leverages the strengths of LLMs in text processing but also delivers precise numerical prediction results. Considering that LLMs inherently function as general-purpose pattern recognition machines [Mirchandani *et al.*, 2023] and to exploit their knowledge and reasoning abilities in text fully, we employ a sentence-to-sentence approach without requiring the LLMs to output specific numerical data. Instead, we request the LLMs to provide time patterns within a specific future time frame, which are used as auxiliary information in the forecasting backbone. This design allows LLMs to focus on processing text data, which they excel at, while the coarser-grained time pattern text predictions also help mitigate model overfitting issues. However, due to the high-dimensional nature of text encoding, influenced by syntax, sentence structure, context, semantic roles, and other factors [Piantadosi, 2023], directly inputting them into the forecasting backbone may adversely affect subsequent model learning. Therefore, we propose an innovative memory network to decode text into effects that numerical models can understand. Our contributions are summarized below.

- **Propose a Novel Perspective on Combining LLMs:** We innovatively propose a multimodal forecasting framework that centers on numerical prediction using non-large model methods while incorporating time pattern text prediction from large models as auxiliary information. This design fully leverages LLMs' strengths in text processing and achieves precise numerical prediction, offering a new solution for time series forecasting tasks.
- **Design an Innovative Memory Network Module:** We introduce a novel memory network module for fine-grained constraint management of text embeddings. This design ensures the diversity of text embeddings while avoiding excessive dispersion. Additionally, our method successfully integrates multimodal data, further improving the performance of time series forecasting tasks.
- **Validate the Effectiveness Through Extensive Experiments:** We conduct experiments on multiple widely used time series forecasting datasets, and the results demonstrate that our framework achieves state-of-the-art performance. Furthermore, we validated the model's capability in multimodal data processing for text-assisted prediction on several newly proposed multimodal datasets.

2 Related Work

Deep Learning Models for Time Series Forecasting. Deep Learning Models for Time Series Forecasting. Recently, deep learning models with meticulous architectures have emerged as promising methods for time series forecasting. Among them, the RNN-based model [Zhao *et al.*, 2017; Lai *et al.*, 2018] and CNN-based [Hewage *et al.*, 2020; Luo and Wang, 2024] model, respectively, uses recurrent connections and convolutional layers to capture temporal patterns. Nevertheless, due to their excellent performance in modeling long sequences, Transformer-based models have gained more widespread recognition. LogSparse [Li *et al.*, 2019], Informer [Zhou *et al.*, 2021] and Reformer [Kitaev *et al.*, 2020] mainly concern with complexity optimization in modeling long sequences. Autoformer [Wu *et al.*, 2021] and FEDformer [Zhou *et al.*, 2022] introduce a time decomposition module to capture features. PatchTST [Nie *et al.*, 2022] and Crossformer [Zhang and Yan, 2023] consider the effects of temporal segmentation and multi-dimension. However, architectural frameworks based on linear layers [Zeng *et al.*, 2023; Wang *et al.*, 2024b; Oreshkin *et al.*, 2019; Xu *et al.*, 2023] have demonstrated commendable performance approximating that of state-of-the-art complex models, indicating that current methods based on statistical analysis have reached a saturation point [Xu *et al.*, 2024]. It is necessary to incorporate the textual ability to circumvent the model's sole reliance on historical numerical data and to harness external information.

LLM-based Forecasters. Inspired by LLMs' strong pattern recognition and inference capabilities on complex token sequences, exploring how to effectively transfer knowledge from these powerful pre-trained large language models to the

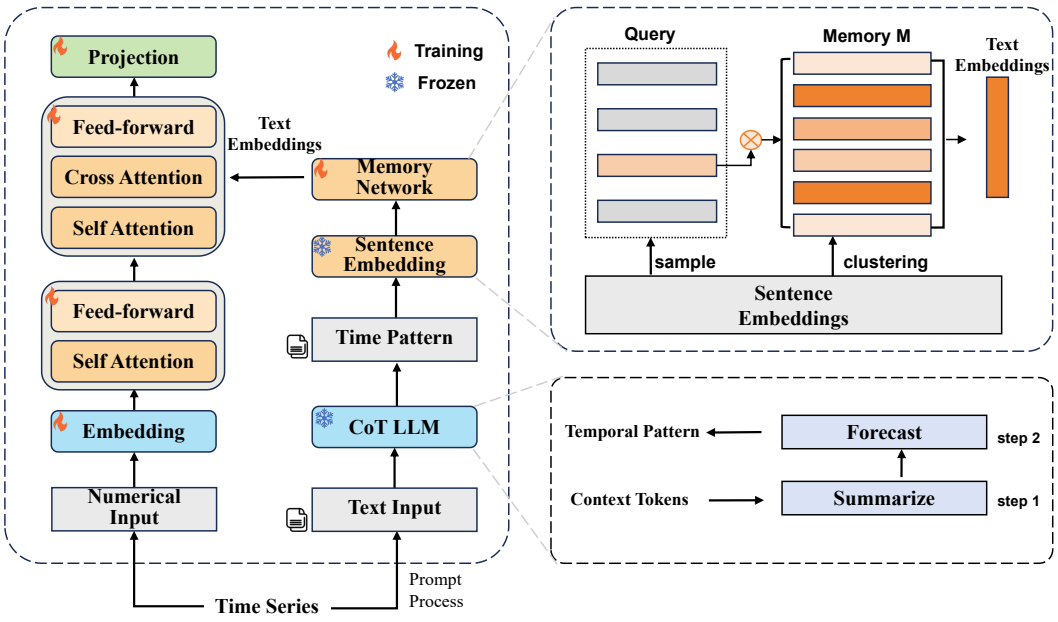

Figure 2: Framework overview, including numerical branch and textual branch.

time series domain becomes a growing tendency. Some researchers [Gruver *et al.*, 2024] use frozen LLMs for zero-shot learning after processing the raw numerical data. Promptcast [Xue and Salim, 2023] transforms the numerical input and output into prompts in a sentence-to-sentence manner, applying language models for forecasting purposes directly. Time-LLM [Jin *et al.*, 2023] and TEST [Sun *et al.*, 2023] align time series embedding with textual embedding to optimize the utilization of the capabilities of LLMs. One Fits All [Zhou *et al.*, 2023] and TEMPO [Cao *et al.*, 2023] freeze some components of LLMs and fine-tune the selected components to enhance their performance in time series analysis tasks. Nevertheless, recent studies [Tan *et al.*, 2024] have indicated that the performance of prediction methods based on pre-trained large language models is mainly comparable to, or even inferior to, that of traditional baseline transformer-based deep learning models. Works such as Time-MMD [Liu *et al.*, 2024b; Wang *et al.*, 2024c] have predominantly concentrated on constructing high-quality multimodal datasets, representing a relatively nascent stage in developing LLMs. We contend that current studies are overly fixated on enabling LLMs to perform numerical predictions, thereby neglecting the true forte of LLMs: their prowess in text processing.

3 Preliminaries

Problem Definition. We first define the concept of time series forecasting. Time series forecasting involves predicting the future numerical sequence within a certain period based on past time series data. We can represent a segment of past time series data as input using vectors $\mathbf{X}_T = \{\mathbf{x}_{t-T+1:t}\} \in \mathbb{R}^T$, where x_t denotes the observed value of the historical time series at time point t; the objective of time series forecasting is to predict future observations $\mathbf{X}_M = \{\mathbf{x}_{t+1:t+M}\} \in \mathbb{R}^M$, where M represents the number of time steps into the future to be predicted.

Prompt-Based Temporal Pattern Forecasting. The prompt-based temporal pattern forecasting task extends the conventional time series forecasting task. Its primary aim is to circumvent LLMs' direct processing of numerical data while harnessing their extensive knowledge and sophisticated reasoning abilities in textual data. The transformation of traditional numerically oriented forecasting tasks from input-output numerical pairs to a sentence-to-sentence format is necessitated through an elaborate prompting process [Xue and Salim, 2023]. In detail, this involves the conversion of input numerical sequences $\mathbf{X}_T$, along with temporal information, into descriptive natural language sentences to create input prompts. Concurrently, the output numerical values are reinterpreted as textual descriptions of temporal patterns delineated by specific time intervals.

4 Method

4.1 Framework Overview

The overall framework comprises four primary modules: prompt-based temporal pattern forecasting, temporal pattern representation, time series representation, and cross-modal auxiliary representation, as shown in Figure 2.

The prompt-based temporal pattern forecasting module incorporates a two-step chain-of-thought prompt process. In the first step, the LLMs summarize the current input temporal patterns, while in the second step, it forecasts future temporal patterns. The temporal pattern representation section initially utilizes a frozen LLM encoder to encode the previously obtained temporal pattern text. Simultaneously, the text embeddings are aligned with the input time series data based on timestamps to generate the initial text embeddings. The memory network is then utilized to constrain the feature

representation of the temporal pattern text input. The time series representation module consists of a normalization layer, a patch embedding layer, and a time series encoder. The cross-modal auxiliary representation module also employs cross-modal alignment to aggregate the learned representations of time series numerical values and text temporal patterns to extract compelling future features from temporal patterns. Finally, a linear decoder layer is used to project the cross-modal fused temporal representations onto the prediction length for forecasting. In the subsequent sections, we will elaborate on the technical details of each module.

4.2 Prompt-Based Temporal Pattern Forecasting

Due to LLMs' limitations in handling numerical tasks and fully leveraging their knowledge and reasoning capabilities developed from pre-training on extensive text datasets, we have transformed conventional numerical prediction tasks into sentence-to-sentence temporal pattern forecasting. Rather than demanding specific numerical forecasting from the LLMs, we require them to provide temporal patterns within discrete time intervals, which aligns with recent research findings that position LLMs as general-purpose pattern recognition machines [Mirchandani *et al.*, 2023]. To enhance the LLMs' proficiency in handling complex arithmetic, common sense, and symbolic reasoning tasks and to emphasize a chained reasoning process, we have adopted a structured chain-of-thought [Wei *et al.*, 2022] approach with a two-step prompt-based prediction process.

In the context of time series forecasting tasks, it is typically necessary to first analyze the given historical sequence to distill trends, seasonality, and other temporal patterns. Additionally, the geographical, dimensional, and domain-specific information associated with the historical sequence is paramount for analyzing time series. However, these aspects are often overlooked or challenging to integrate into the forecasting process in previous work. To address this issue, we have designed the first step of the thinking chain prompt, which transforms numerical values devoid of additional context into statements encompassing background information. This prompt requires the LLMs to analyze the temporal patterns of the historical sequence, leveraging their knowledge and reasoning capabilities to conduct a comprehensive analysis that integrates multifaceted information.

Given the limitations of LLMs in handling numerical data, the prompt for the second step of the thinking chain was not designed to require LLMs to output numerical forecasting directly. Moreover, previous approaches often rely solely on the statistical analysis of historical data, which can lead to over-simplification or over-fitting [Xu *et al.*, 2024]. In real-world time series forecasting scenarios, future sequences are typically influenced by many factors, necessitating the integration of various external information sources for enhanced prediction accuracy, such as using climate change data to forecast energy prices or news text to predict stock prices. Therefore, in the second step of the prompt process, LLMs are tasked with predicting future temporal pattern texts based on past patterns, optionally incorporating external information, to achieve more precise forecasting results.

4.3 Temporal Pattern Representation

Following the generation of the prediction text, we employ a frozen pre-trained LLMs text encoder to generate text embeddings. Considering that text embeddings are closely related to various semantic factors, we have designed an innovative memory network module to ensure the stability and diversity of the text embeddings, resulting in high-quality temporal pattern text embeddings.

We utilize a pre-trained large language model backbone network, optimized based on the BERT [Devlin, 2018] architecture [Wang *et al.*, 2020], as a text encoder to project the text input into a temporal feature dimension, as illustrated by the following equation.

$$\boldsymbol{T}_i = \text{SentenceTransformers}(\boldsymbol{R}_i) \tag{1}$$

where $\boldsymbol{R}_i$ represents the original textual input, $\boldsymbol{T}_i \in \mathbb{R}^D$ represents the text features embeddings.

To mitigate the issue of excessive dispersion in text encoding, which could lead to suboptimal learning performance in subsequent models, we introduce a novel memory network module. Firstly, memory items are initialized through K-means clustering, as represented by the following equation.

$$\boldsymbol{M} = \text{K-means}(\boldsymbol{T}) \tag{2}$$

where $\boldsymbol{M} \in \mathbb{R}^{N \times D}$ represents the initial memory items embeddings, and $\boldsymbol{N}$ represents the number of memory items.

We align the text embeddings $\boldsymbol{T}$ with the time series on the timestamp to the text query queue $\boldsymbol{Q} \in \mathbb{R}^{l \times D}$, $\boldsymbol{l}$ represents the length of series. Specifically, we map all timestamps within the same period to identical temporal pattern text embeddings, thereby constructing a text embeddings queue of the same length as the input time series. To obtain higher-quality text embeddings, which would allow the model to better understand key semantic information, we derived the weight matrix $\boldsymbol{W}$ by calculating the similarity between the query queue and the memory items. We obtained the final text embeddings through the weighted output of the memory items based on the weight matrix $\boldsymbol{W}$. Due to the high dimensionality of the text embeddings, in order to further reduce dimensions and extract adequate information, both the query queue and the memory items were processed through an additional linear projection layer.

4.4 Time Series Representation

Due to the superior capability of attention mechanisms in modelling long-term temporal dependencies within long data sequences, our temporal representation module is primarily derived from an adapted version of the vanilla Transformer architecture. Initially, we incorporate normalization layers to stabilize the sequence, thereby mitigating the issue of distribution shift. Within the token encoding layer, a patch-wise segmentation approach [Nie *et al.*, 2022] is adopted, which enriches the tokens with more comprehensive temporal semantic information. This approach also aligns the granularity with temporal pattern embeddings, facilitating more effective information exchange. Ultimately, the self-attention mechanism is employed to extract temporal features.

Owing to the phenomenon of distribution shift frequently encountered in time series data, we initially incorporate a

Method	Ours		Time-LLM		GPT4TS		PatchTST		TimesNet		DLinear		FEDformer		Autoformer		Informer	
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETT	**0.778**	**0.639**	0.897	0.653	0.817	0.640	0.834	0.657	0.854	0.657	0.873	0.666	0.827	0.672	0.950	0.724	1.405	0.878
Electricity	0.357	**0.424**	0.394	0.451	0.352	0.426	0.374	0.438	**0.351**	0.426	0.393	0.457	0.390	0.459	0.569	0.556	0.526	0.538
Traffic	**0.189**	**0.280**	0.247	0.341	**0.189**	0.289	0.193	0.285	0.196	0.292	0.323	0.404	0.201	0.296	0.265	0.365	0.297	0.378
Station	**0.215**	**0.311**	0.386	0.405	0.296	0.359	0.254	0.333	0.305	0.357	0.448	0.452	0.348	0.420	0.569	0.549	0.467	0.466
Weather	**0.526**	**0.466**	0.604	0.505	0.610	0.502	0.634	0.507	0.618	0.511	0.546	0.470	0.606	0.508	0.702	0.576	0.686	0.560
Exchange	0.473	0.470	0.467	0.469	0.493	0.485	0.468	0.470	0.487	0.484	**0.376**	**0.457**	0.557	0.537	0.581	0.551	0.993	0.746
Illness	0.692	**0.647**	0.871	0.777	**0.669**	0.653	0.706	0.654	0.747	0.683	1.340	0.992	0.826	0.731	0.799	0.731	5.321	2.075
1st Count	10		0		2		0		1		2		0		0		0	

Table 1: Univariate time series forecasting performance comparisons.

straightforward reverse instance norm module [Kim *et al.*, 2021]. This module employs a simple approach of normalizing the input time series using mean and variance, which are added back to the output at the last stage. After this normalization process, we replicate the last value of the sequence and append it to the end of the original sequence to ensure accurate patching. The patching process involves dividing the time series X_T into overlapping or non-overlapping patches. Letting the patch length be denoted as P, and the non-overlapping stride between adjacent patches as S, we can obtain a sequence of patches $\boldsymbol{X_p} \in \mathbb{R}^{N \times P}$. Here, N represents the number of patches, where N is calculated as $N = \lceil \frac{L-P}{S} \rceil + 1$.

To generate the input for the Transformer encoder, we initially project the sequence of patches $\boldsymbol{X_p} \in \mathbb{R}^{N \times P}$ into hidden representations through a linear projection $\mathbf{W}_p \in \mathbb{R}^{P \times D}$ combined with positional encoding $\mathbf{W}_{\text{pos}}$, resulting in a sequence of patch representations, $\boldsymbol{X_d^i} = \mathbf{W}_p \boldsymbol{X_p^i} + \mathbf{W}_{\text{pos}}$, which serves as the input token sequence for the Transformer. Subsequently, for each head h of the multi-head attention mechanism, linear projections$\mathbf{W}_q, \mathbf{W}_k, \mathbf{W}_v$ are applied to map the input to the Query ($\mathbf{Q}$), Key ($\mathbf{K}$), and Value ($\mathbf{V}$) matrices. Attention scores are then computed through scaled dot-product operations as illustrated by the following equation and get the final output of the encoder $\mathbf{O}_h^{(i)} \in \mathbb{R}^{N \times D}$.

$$(\mathbf{O}_h)^T = \text{Attention}(\mathbf{Q}_h, \mathbf{K}_h, \mathbf{V}_h) = \text{Softmax}(\frac{\mathbf{Q}_h {\mathbf{K}_h}^T}{\sqrt{D_k}})\mathbf{V}_h \tag{3}$$

4.5 Multimodal Auxiliary Representation

To seamlessly integrate temporal pattern text features with temporal attributes, we have conceptualized a novel multimodal auxiliary representation module. This module is designed to enhance the generation of a more accurate temporal representation by effectively merging disparate data modalities. The core function of this module is to meticulously extract high-quality embeddings from temporal pattern embeddings, leveraging the synergistic power of time series embeddings. This dual-embedding approach ensures that the resulting information is comprehensive and precise. By harmonizing these distinct data sources, the module facilitates a deeper understanding of temporal dynamics, enabling more reliable and insightful forecasts.

Specifically, we employ a distinct cross-attention layer where temporal (numerical) features are used as queries, and temporal pattern (textual) features are employed as keys and values, ensuring that the output dimensions are naturally aligned with the temporal dimensions. Through linear projections $\mathbf{W}_q, \mathbf{W}_k, \mathbf{W}_v$, we obtain the Query ($Q$), Key ($K$), and Value ($V$) matrices separately. Subsequently, the attention mechanism computes the scores, revealing the correlation between the temporal numerical data and the temporal patterns. These computed scores are then utilized to aggregate the information, producing high-quality representation. Utilizing the cross-modal information interaction module, we transfer the high-quality temporal pattern predictive information from robust pre-trained large-scale models to the time series embeddings. By structuring the attention layer in this manner, we maximize the utility of numerical and textual information, leading to a more robust and precise temporal representation.

Finally, we employ a linear projection $\mathbf{W}_o \in \mathbb{R}^{\hat{N} \times M}$ to map the obtained high-quality temporal representations to the desired prediction length where$\hat{\boldsymbol{N}} = \boldsymbol{N} \times \boldsymbol{D}$, followed by the completion of the reverse instance norm module.

5 Experiments

5.1 Experimental Setup

Datasets. The proposed model framework has been rigorously evaluated across eight real-world benchmark datasets, encompassing various time series application domains. The datasets include **ETT** [Zhou *et al.*, 2021], which includes four sub-datasets: ETTh1, ETTh2, ETTm1, and ETTm2; **Traffic**; **Electricity** [Lai *et al.*, 2018]; **Weather**; **ILLness**; **Exchange**; and **Station** [Liu *et al.*, 2020]. We also evaluate our method on multimodal datasets from **Time-MMD** [Liu *et al.*, 2024b] and **From News to Forecast** [Wang *et al.*, 2024c]. The evaluation of these datasets thoroughly examines the model's efficacy in various time series forecasting scenarios.

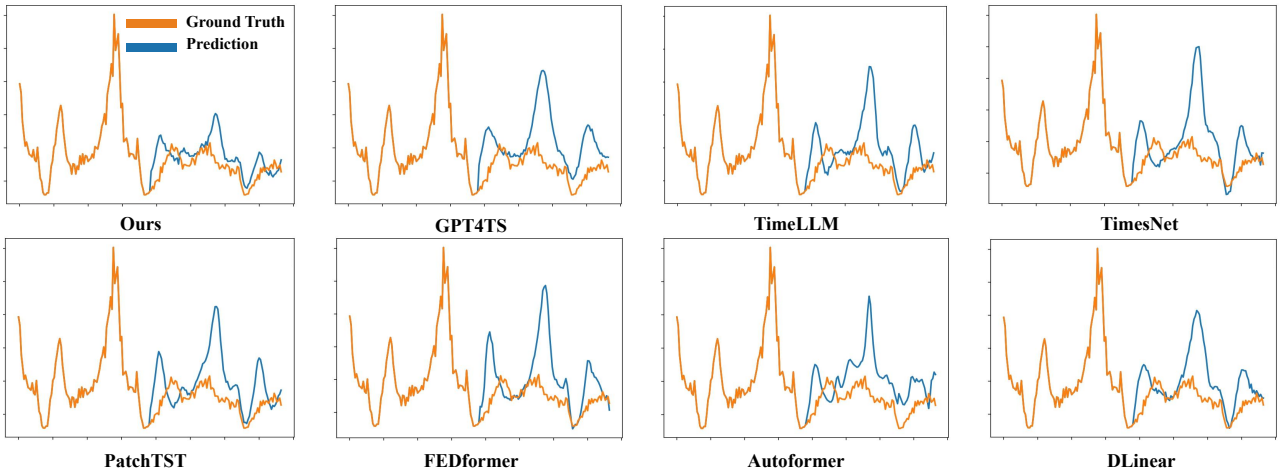

Figure 3: Prediction cases from Station by different models. The input sequence length is set to 96 and the predictive length is set to 96. Blue lines are the ground truths and orange lines are the model predictions.

Baselines. We have selected eight state-of-the-art time series forecasting methods for comparative analysis. This includes transformer-based models such as Informer [Zhou *et al.*, 2021], Autoformer [Wu *et al.*, 2021], FEDformer [Zhou *et al.*, 2022], and PatchTST [Nie *et al.*, 2022]; a linear model, DLinear [Zeng *et al.*, 2023]; a CNN-based model, TimesNet [Wu *et al.*, 2022]; language model-based approachs TimeLLM [Jin *et al.*, 2023] GPT4TS [Zhou *et al.*, 2023].

Experimental Settings. We adhere to consistent settings throughout our experiments to ensure a fair comparison with the experimental setup detailed in Wu et al. [Wu *et al.*, 2022]. The maximum number of epochs is set to 10, incorporating an early stopping mechanism. Specifically, we employ a lookback window of 36 for the illness dataset, while for the remaining datasets, a lookback window of 96 is utilized. We integrated DeepSeek2 [Liu *et al.*, 2024a] for the sentence-to-sentence temporal pattern text predictor. Additionally, for the text encoder, we utilized the encoder component of MinLM [Wang *et al.*, 2020], which remains frozen during the training process. The patch length is strategically selected to be 16, and the stride is set to 8. The training process is facilitated using the AdamW optimizer, with an initial learning rate of 0.0001. All experiments are meticulously conducted on an NVIDIA 3090 24GB GPU.

5.2 Main Results

Table 1 presents the overall univariate time series forecasting result, with an input length of 36 for the illness dataset and 96 for the others; the forecast lengths are 24, 36, 48, and 60 for the illness dataset, and 96, 192, 336, and 720 for the other datasets. The table records the average performance across the four forecast lengths for each dataset. We have bolded the best performance for each row and underlined the second-best performance. As indicated in the table, our model demonstrates significant improvements over the baseline models, achieving the best or second-best performance in 61 out of 70 instances. Our performance surpasses that of the other language model approachs, validating the effectiveness of our integration of LLMs with time series.

Method	Multimodal		Unimodal	
Metric	MSE	MAE	MSE	MAE
AULF	**0.174**	**0.283**	0.194	0.306
Agriculture	**0.147**	**0.247**	0.151	0.249
Climate	**0.326**	**0.425**	0.340	0.433
Economy	**0.205**	**0.358**	0.225	0.375
Energy	**0.215**	**0.332**	0.225	0.339
Environment	**0.299**	**0.399**	0.321	0.406
Health(US)	**1.328**	**0.758**	1.352	0.768
Security	**72.175**	**4.133**	73.115	4.156
SocialGood	**0.881**	**0.414**	0.903	0.425
Traffic	**0.122**	**0.209**	0.125	0.213

Table 2: Multimodal experiments.

Our model's performance across various datasets does not consistently achieve top-tier results, which is a limitation because our framework is constructed upon an attention mechanism network. The inherent constraints of the attention network restrict the framework's performance, particularly in scenarios where specific datasets exhibit characteristics that are more conducive to linear models.

Furthermore, we evaluate our method across ten multimodal datasets, specifically assessing its capacity to enhance predictive accuracy by incorporating external textual information. Integrating external texts resulted in superior performance across all datasets, underscoring our method's adeptness at leveraging textual knowledge and reasoning capabilities to refine prediction accuracy. We provide the average results in the table 2. Notably, we also conducted a comparative analysis using the Time-MMD method [Liu *et al.*, 2024b]. However, a significant performance disparity was observed, potentially due to differences in experimental configurations.

Upon meticulous analysis of the experimental results, we observe that the proposed framework exhibits significant per-

formance advantages on datasets such as stations, which possess pronounced temporal patterns. This edge is attributed to the knowledge and reasoning capabilities of the LLMs, which effectively predict the temporal patterns of future sequences. Nevertheless, when confronted with datasets like ETT, where the temporal patterns are highly complex, the performance of all models is generally low.

5.3 Model Analysis

Result Analysis

The performance of our model is more vividly depicted in the accompanying figure 3, which highlight the inherent limitations of models that exclusively rely on statistical analysis. When confronted with scenarios where the patterns within the forecast window diverge significantly from those observed in the historical window, these models demonstrate a pronounced performance decline. In contrast, our approach, which integrates LLMs, capitalizes on their extensive knowledge and sophisticated reasoning capabilities. By harnessing these attributes, we enable the model to infer potential shifts in future patterns based on temporal information. This inference process not only constrains but also corrects the predictions generated by the numerical model, leading to more precise forecasts and demonstrating the model's robustness in handling abrupt changes or complex time series.

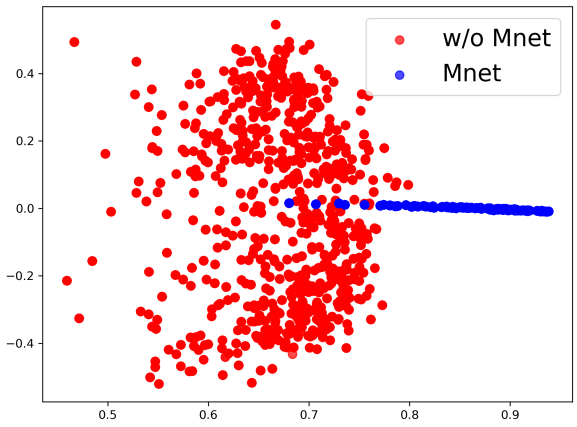

Figure 4: The text embeddings are visualized in a 2D space.

Ablation Studies

The primary objective of this work is to propose an enhanced structure for integrating LLMs with time series data. As such, the ablation studies are designed to validate the language model branch modules' effectiveness rigorously. Initially, we systematically remove the entire language model module to evaluate the utility of our LLMs auxiliary module in isolation. Subsequently, we focus on validating the memory network's (Mnet) effectiveness within the language model branch. Our ablation results unequivocally demonstrate that the language model module significantly augments model performance, thereby underscoring the value of incorporating LLMs into the forecasting framework.

Additionally, the experiments reveal that the absence of a memory network to constrain text embeddings leads to a degradation in model performance. This finding highlights the critical role of our memory network in preventing overly broad text embeddings while maintaining the necessary diversity. Furthermore, we calculate the dispersion coefficients (ratio of variance to mean) before and after processing, which can measure the degree of data dispersion. Through the memory network, the dispersion coefficient of the feature vectors decreased from **0.023** to **0.008**, proving that our network can suppress overly dispersed embeddings. Figure 4 offers a succinct visualization of the memory network's functionality. We have reduced the dimensionality of the text embeddings using principal component analysis (PCA) and projected them onto a two-dimensional space for visualization. The illustration demonstrates that the memory networks effectively constrain overly scattered text embeddings, allowing the model to understand key semantic information better. The table 3 present a selection of our ablation results.

Method		Ours		w/o LLMs		w/o Mnet	
Metric		MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	**0.720**	**0.605**	0.740	0.607	0.785	0.629
	192	**0.748**	**0.621**	0.800	0.639	0.849	0.666
	336	**0.801**	**0.650**	0.877	0.676	0.825	0.666
	720	**0.842**	**0.680**	0.917	0.704	0.855	0.696
	Avg	**0.778**	**0.639**	0.834	0.657	0.829	0.664
Traffic	96	**0.192**	**0.283**	0.200	0.290	0.198	0.288
	192	**0.184**	**0.274**	0.190	0.281	0.189	0.282
	336	**0.181**	**0.275**	0.183	0.276	0.185	0.280
	720	**0.200**	**0.289**	0.201	0.291	0.208	0.297
	Avg	**0.189**	**0.280**	0.193	0.285	0.195	0.287

Table 3: Ablation of method designs.

6 Conclusion

This work offers a novel perspective on leveraging the strengths of LLMs in time series tasks, playing for strength: capitalizing on the language models' proficiency in text processing while entrusting numerical models with the handling of numerical data. The proposed approach fully utilizes LLMs' knowledge and reasoning abilities by leveraging the predictive capabilities of these language models to generate textual descriptions of future temporal patterns and embedding these texts to assist numerical model predictions. It achieves precise forecasting, thereby enhancing predictive performance. This method effectively mitigates the limitations of conventional models that rely solely on statistical analysis of historical data. Extensive evaluations have confirmed the effectiveness of our model in advancing state-of-the-art predictive performance. It presents a promising direction for integrating LLMs with time series forecasting efforts.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants (62321001, 62471055, U23B2001, 62101064, 62171057, 62201072, 62071067), the High-Quality Development Project of the MIIT(2440STCZB2584), the Ministry of Education and China Mobile Joint Fund (MCM20200202, MCM20180101), the Fundamental Research Funds for the Central Universities (2024PTB-004).

References

[Cao *et al.*, 2023] Defu Cao, Furong Jia, Sercan O Arik, Tomas Pfister, Yixiang Zheng, Wen Ye, and Yan Liu. Tempo: Prompt-based generative pre-trained transformer for time series forecasting. *arXiv preprint arXiv:2310.04948*, 2023.

[Choi *et al.*, 2022] Taesung Choi, Dongkun Lee, Yuchae Jung, and Ho-Jin Choi. Multivariate time-series anomaly detection using seqvae-cnn hybrid model. In *2022 International Conference on Information Networking (ICOIN)*, pages 250–253. IEEE, 2022.

[Devlin, 2018] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[Gruver *et al.*, 2024] Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems*, 36, 2024.

[Guo *et al.*, 2023] Wei Guo, Chang Meng, Enming Yuan, Zhicheng He, Huifeng Guo, Yingxue Zhang, Bo Chen, Yaochen Hu, Ruiming Tang, Xiu Li, et al. Compressed interaction graph based framework for multi-behavior recommendation. In *Proceedings of the ACM Web Conference 2023*, pages 960–970, 2023.

[Hewage *et al.*, 2020] Pradeep Hewage, Ardhendu Behera, Marcello Trovati, Ella Pereira, Morteza Ghahremani, Francesco Palmieri, and Yonghuai Liu. Temporal convolutional neural (tcn) network for an effective weather forecasting using time-series data from the local weather station. *Soft Computing*, 24:16453–16482, 2020.

[Jin *et al.*, 2023] Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*, 2023.

[Kim *et al.*, 2021] Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International Conference on Learning Representations*, 2021.

[Kitaev *et al.*, 2020] Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*, 2020.

[Lai *et al.*, 2018] Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long-and short-term temporal patterns with deep neural networks. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 95–104, 2018.

[Li *et al.*, 2019] Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyou Zhou, Wenhu Chen, Yu-Xiang Wang, and Xifeng Yan. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *Advances in neural information processing systems*, 32, 2019.

[Liu *et al.*, 2020] Lingbo Liu, Jingwen Chen, Hefeng Wu, Jiajie Zhen, Guanbin Li, and Liang Lin. Physical-virtual collaboration modeling for intra-and inter-station metro ridership prediction. *IEEE Transactions on Intelligent Transportation Systems*, 23(4):3377–3391, 2020.

[Liu *et al.*, 2024a] Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*, 2024.

[Liu *et al.*, 2024b] Haoxin Liu, Shangqing Xu, Zhiyuan Zhao, Lingkai Kong, Harshavardhan Kamarthi, Aditya B Sasanur, Megha Sharma, Jiaming Cui, Qingsong Wen, Chao Zhang, et al. Time-mmd: Multi-domain multimodal dataset for time series analysis. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

[Luo and Wang, 2024] Donghao Luo and Xue Wang. Moderntcn: A modern pure convolution structure for general time series analysis. In *The Twelfth International Conference on Learning Representations*, 2024.

[Mirchandani *et al.*, 2023] Suvir Mirchandani, Fei Xia, Pete Florence, Brian Ichter, Danny Driess, Montserrat Gonzalez Arenas, Kanishka Rao, Dorsa Sadigh, and Andy Zeng. Large language models as general pattern machines. *arXiv preprint arXiv:2307.04721*, 2023.

[Nie *et al.*, 2022] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*, 2022.

[Oreshkin *et al.*, 2019] Boris N Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-beats: Neural basis expansion analysis for interpretable time series forecasting. *arXiv preprint arXiv:1905.10437*, 2019.

[Piantadosi, 2023] Steven Piantadosi. Modern language models refute chomsky's approach to language. *Lingbuzz Preprint, lingbuzz*, 7180, 2023.

[Qiu *et al.*, 2024] Xiangfei Qiu, Jilin Hu, Lekui Zhou, Xingjian Wu, Junyang Du, Buang Zhang, Chenjuan Guo, Aoying Zhou, Christian S. Jensen, Zhenli Sheng, and Bin Yang. Tfb: Towards comprehensive and fair benchmarking of time series forecasting methods. In *Proc. VLDB Endow.*, pages 2363–2377, 2024.

[Qiu *et al.*, 2025] Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. Duet: Dual clustering enhanced multivariate time series forecasting. In *SIGKDD*, pages 1185–1196, 2025.

[Rae *et al.*, 2021] Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*, 2021.

[Shao *et al.*, 2022] Zezhi Shao, Zhao Zhang, Wei Wei, Fei Wang, Yongjun Xu, Xin Cao, and Christian S Jensen. Decoupled dynamic spatial-temporal graph neural network for traffic forecasting. *arXiv preprint arXiv:2206.09112*, 2022.

[Sun *et al.*, 2023] Chenxi Sun, Hongyan Li, Yaliang Li, and Shenda Hong. Test: Text prototype aligned embedding to activate llm's ability for time series. *arXiv preprint arXiv:2308.08241*, 2023.

[Tan *et al.*, 2024] Mingtian Tan, Mike A Merrill, Vinayak Gupta, Tim Althoff, and Thomas Hartvigsen. Are language models actually useful for time series forecasting? *arXiv preprint arXiv:2406.16964*, 2024.

[Wang *et al.*, 2020] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33:5776–5788, 2020.

[Wang *et al.*, 2023] Fei Wang, Di Yao, Yong Li, Tao Sun, and Zhao Zhang. Ai-enhanced spatial-temporal data-mining technology: New chance for next-generation urban computing. *The Innovation*, 4(2), 2023.

[Wang *et al.*, 2024a] Chengsen Wang, Qi Qi, Jingyu Wang, Haifeng Sun, Zirui Zhuang, Jinming Wu, and Jianxin Liao. Rethinking the power of timestamps for robust time series forecasting: A global-local fusion perspective. In *Thirty-eighth Conference on Neural Information Processing Systems*, 2024.

[Wang *et al.*, 2024b] Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y Zhang, and Jun Zhou. Timemixer: Decomposable multiscale mixing for time series forecasting. *arXiv preprint arXiv:2405.14616*, 2024.

[Wang *et al.*, 2024c] Xinlei Wang, Maike Feng, Jing Qiu, Jinjin Gu, and Junhua Zhao. From news to forecast: Integrating event analysis in llm-based time series forecasting with reflection. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[Wang *et al.*, 2025] Chengsen Wang, Qi Qi, Jingyu Wang, Haifeng Sun, Zirui Zhuang, Jinming Wu, Lei Zhang, and Jianxin Liao. Chattime: A unified multimodal time series foundation model bridging numerical and textual data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 12694–12702, 2025.

[Wei *et al.*, 2022] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[Wu *et al.*, 2021] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34:22419–22430, 2021.

[Wu *et al.*, 2022] Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. *arXiv preprint arXiv:2210.02186*, 2022.

[Xu *et al.*, 2023] Zhijian Xu, Ailing Zeng, and Qiang Xu. Fits: Modeling time series with $10k$ parameters. In *The Twelfth International Conference on Learning Representations*, 2023.

[Xu *et al.*, 2024] Zhijian Xu, Yuxuan Bian, Jianyuan Zhong, Xiangyu Wen, and Qiang Xu. Beyond trend and periodicity: Guiding time series forecasting with textual cues. *arXiv preprint arXiv:2405.13522*, 2024.

[Xue and Salim, 2023] Hao Xue and Flora D Salim. Promptcast: A new prompt-based learning paradigm for time series forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 2023.

[Zeng *et al.*, 2023] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 11121–11128, 2023.

[Zhang and Yan, 2023] Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *The eleventh international conference on learning representations*, 2023.

[Zhao *et al.*, 2017] Zheng Zhao, Weihai Chen, Xingming Wu, Peter CY Chen, and Jingmeng Liu. Lstm network: a deep learning approach for short-term traffic forecast. *IET intelligent transport systems*, 11(2):68–75, 2017.

[Zhou *et al.*, 2021] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 11106–11115, 2021.

[Zhou *et al.*, 2022] Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International conference on machine learning*, pages 27268–27286. PMLR, 2022.

[Zhou *et al.*, 2023] Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. One fits all: Power general time series analysis by pretrained lm. *Advances in neural information processing systems*, 36:43322–43355, 2023.