Adversarial Vulnerabilities in Large Language Models for Time Series Forecasting

Fuqiang Liu[*1] **Sicong Jiang**[*1] **Luis Miranda-Moreno**[1] **Seongjin Choi**[2] **Lijun Sun**[1]

[1]McGill University [2]University of Minnesota - Twin Cities

Abstract

Large Language Models (LLMs) have recently demonstrated significant potential in time series forecasting, offering impressive capabilities in handling complex temporal data. However, their robustness and reliability in real-world applications remain under-explored, particularly concerning their susceptibility to adversarial attacks. In this paper, we introduce a targeted adversarial attack framework for LLM-based time series forecasting. By employing both gradient-free and black-box optimization methods, we generate minimal yet highly effective perturbations that significantly degrade the forecasting accuracy across multiple datasets and LLM architectures. Our experiments, which include models like LLM-Time with GPT-3.5, GPT-4, LLaMa, and Mistral, TimeGPT, and TimeLLM show that adversarial attacks lead to much more severe performance degradation than random noise, and demonstrate the broad effectiveness of our attacks across different LLMs. The results underscore the critical vulnerabilities of LLMs in time series forecasting, highlighting the need for robust defense mechanisms to ensure their reliable deployment in practical applications. The code repository can be found at Johnson/AdvAttackLLM4TS.

1 INTRODUCTION

Time series forecasting plays a pivotal role in numerous real-world applications, ranging from finance and healthcare to energy management and climate modeling. Accurately predicting temporal patterns in the data is crucial for informed decision-making in these domains (Liu et al., 2022b; Jiang et al., 2024). Recently, Large Language Models (LLMs), originally designed for Natural Language Processing (NLP) tasks, have demonstrated remarkable potential in handling time series forecasting challenges (Gruver et al., 2024; Liu et al., 2024b,a; Tan et al., 2024; Jin et al., 2023a). These models, including BERT (Devlin, 2018), GPT (Brown, 2020; Achiam et al., 2023), LLaMa (Touvron et al., 2023) and their successors, leverage their powerful attention mechanisms and vast pre-training on diverse datasets to capture intricate and non-linear temporal dependencies, making them highly effective for complex forecasting tasks.

LLMs exhibit strong generalization capabilities across various types of time series data. Compared to traditional models like ARIMA (Kalpakis et al., 2001) and Exponential Smoothing (Gardner Jr, 1985), as well as advanced deep learning models such as DNNs (Salinas et al., 2020; Oreshkin et al., 2019; Challu et al., 2023), and Transformer-based architectures (Zhou et al., 2021; Wu et al., 2021; Liu et al., 2024d; Zhou et al., 2022), LLMs excel in modeling long-term dependencies and capturing non-linear patterns within temporal sequences. This has resulted in impressive forecasting accuracy across applications ranging from energy consumption predictions to weather forecasting (Jin et al., 2023b).

However, despite their success, the robustness and reliability of LLMs in real-world forecasting applications remain concerns, particularly their vulnerability to adversarial attacks is under-explored. Adversarial attacks introduce subtle, often imperceptible perturbations to input data, leading to significant and misleading changes in model predictions. While the susceptibility of machine learning models to such attacks has been well-explored in image processing and NLP domains (Xu et al., 2020; Morris et al., 2020; Wei et al., 2018), there is a noticeable gap in research on their impact on LLMs used for time series forecasting.

While adversarial attacks and defenses for deep neural

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258. [*]Co-first Authors

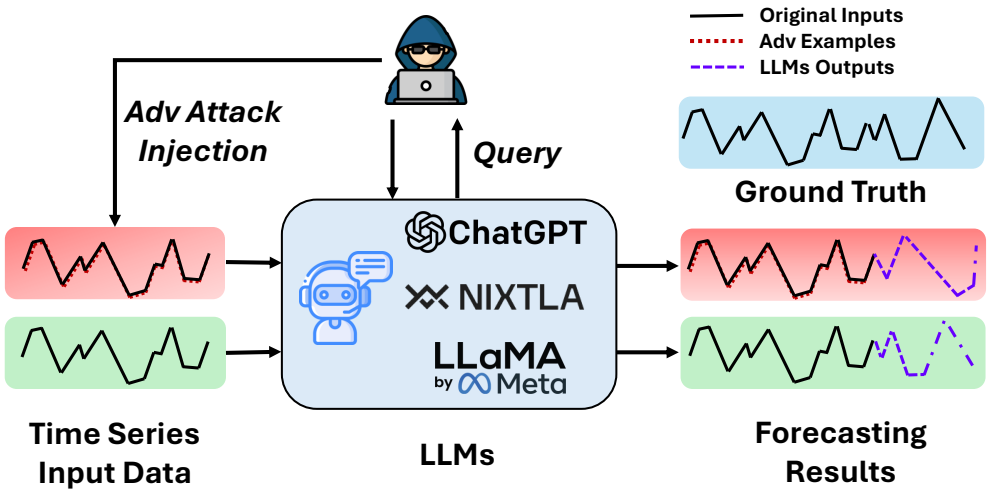


Figure 1: Adversarial Black-box Attack for LLMs in Time Series Forecasting.

networks have been extensively studied across various domains (Madry, 2017), executing adversarial attacks against LLMs in time series forecasting presents two significant challenges. First, to prevent information leakage, we cannot use ground truth values (i.e., future time steps) when attacking forecasting models. Second, LLMs must be treated as strict black-box systems due to the difficulty of accessing their internal workings and parameters.

In this paper, we address this gap by proposing a gradient-free black-box attack that transforms the output of LLM-based forecasting models into a random walk, while investigating the vulnerabilities of large language models in time series forecasting. As depicted in Figure 1, we demonstrate that even minimal attack perturbations can cause substantial deviations in LLMs' predictions. We evaluate three forms of LLM applications for time series forecasting, encompassing six sub-models as well as two non-LLM models, across five datasets from various real-world domains. Our findings reveal that LLMs, despite their advanced architectures, are indeed susceptible to adversarial manipulations in time series domain, leading to unstable and inaccurate forecasts. This underscores the urgent need to develop more robust LLMs that can withstand such attacks, ensuring their reliability in real-world applications.

In conclusion, this study contributes to the ongoing discourse on the robustness of LLMs by revealing their vulnerabilities to adversarial attacks in time series forecasting. Our findings underscore the critical need to address these vulnerabilities to ensure that LLM-based forecasting models are not only accurate but also resilient, thereby enhancing their practical utility in high-stakes applications.

2 RELATED WORK

2.1 Adversarial Attacks on Forecasting

Adversarial attacks in time series forecasting have emerged as a crucial area of research, exposing vulnerabilities in forecasting models. Unlike adversarial studies in static domains, such as object recognition or time series classification, adversarial attacks on time series forecasting cannot leverage ground truth data for perturbation generation due to the risk of information leakage (Liu et al., 2022a). To address this challenge, surrogate techniques have been adopted (Liu et al., 2021), which bypass the need for labels, as is done in traditional adversarial attack methods like the Fast Gradient Sign Method (Goodfellow et al., 2014). Several studies have treated forecasting models as white-box systems to investigate the effects of adversarial attacks on commonly used models in time series forecasting, such as ARIMA, LSTMs, and Transformer-based models (Liu et al., 2022c, 2023). These studies demonstrate that even small perturbations can severely impact these models, resulting in inaccurate forecasts. However, evaluating the vulnerability of LLM-based forecasting presents a significant challenge, as internal access is typically restricted, requiring these models to be treated as black-box systems.

2.2 Adversarial Attacks on LLMs

Adversarial attacks on LLMs have gained increasing attention, focusing on how slight manipulations can significantly alter their outputs. These attacks are often classified into prompt-based attacks, token-level manipulations, gradient-based attacks, and embedding

perturbations.

- Jailbreak Prompting (Yu et al., 2024; Wei et al., 2024): Crafted prompts that bypass LLM guardrails, inducing unintended or harmful outputs by exploiting unconventional phrasing.
- Prompt Injection (Greshake et al., 2023; Xue et al., 2024; Liu et al., 2024c): Adversarial instructions embedded into benign prompts to manipulate LLM responses, highlighting their vulnerability to prompt manipulation.
- Gradient Attacks (Madry, 2017; Guo et al., 2021): Using internal model parameters, attackers apply gradient-based methods to perturb inputs, significantly altering outputs with minimal changes.
- Embedding Perturbations (Schwinn et al., 2024; Singh et al., 2024): Subtle changes to input embeddings disrupt LLM's internal representations, leading to erroneous outputs with minimal visible input alterations.

While extensive research has been conducted on attacks against LLMs at various levels, most of these focus on text-based manipulations. However, there's a significant gap in understanding how LLMs perform in non-textual tasks, particularly time series forecasting. In language tasks, attacks typically manipulate static text inputs, such as words or prompts, to exploit the LLM's understanding and induce specific outputs. However, time series forecasting involves dynamic, evolving data points, requiring attackers to introduce perturbations that maintain the sequence's natural flow and coherence.

3 MANIPULATING LLM-BASED TIME SERIES FORECASTING

3.1 Formulations of LLM-based Forecasting

LLMs have shown promising performance in time series forecasting by leveraging their ability to perform next-token prediction, a technique originally developed for text-based tasks (Gruver et al., 2024; Jin et al., 2023a). A typical LLM-based time series forecasting model, denoted as $f(\cdot)$, consists of two primary components: an embedding or tokenization module that encodes the time series data into a sequence of tokens, and a pre-trained LLM that autoregressively predicts the subsequent tokens. The embedding module translates the raw time series into a format suitable for the LLM, while the LLM captures the temporal dependencies and generates predictions based on its learned representations.

Let $\mathbf{X}_t \in \mathbb{R}^d$ denote d-dimensional time series at time t, where $x_{i,t} = [\mathbf{X}_t]_i$ represents the observation of the i-th component of the time series. Given a sequence of recent T historical observations $\mathbf{X}_{t-T+1:t}$, a forecasting model, $f(\cdot)$, is employed to predict the future values for the subsequent τ time steps. The prediction is formulated as:

$$\hat{\mathbf{Y}}_{t+1:t+\tau} = f\left(\mathbf{X}_{t-T+1:t}\right), \tag{1}$$

where $\hat{\mathbf{Y}}_{t+1:t+\tau}$ denotes the predicted future values and $\mathbf{Y}_{t+1:t+\tau}$ represents the corresponding ground truth values. It is important to note that the prediction horizon is typically less than or equal to the historical horizon, i.e., $\tau \leq T$.

3.2 Threat Model

Our objective is to deceive an LLM-based time series forecasting model into producing anomalous outputs that deviate significantly from both its normal predictions and the corresponding ground truth, through the introduction of imperceptible perturbations. This adversarial attack problem can be framed as an optimization task as follows:

$$\begin{aligned} \max_{\boldsymbol{\rho}_{t-T+1:t}} \quad & \mathcal{L}\left(f\left(\mathbf{X}_{t-T+1:t} + \boldsymbol{\rho}_{t-T+1:t}\right), \mathbf{Y}_{t+1:t+\tau}\right) \\ \text{s.t.} \quad & \left\|\rho_i\right\|_p \leq \epsilon, i \in [t-T+1, t], \end{aligned} \tag{2}$$

where $\mathbf{X}_{t-T+1:t}$ denotes the clean input, $\mathbf{Y}_{t+1:t+\tau}$ denotes the true future values, and $\boldsymbol{\rho}_{t-T+1:t}$ denotes the adversarial perturbations.

The loss function $\mathcal{L}$ quantifies the discrepancy between the model's output and the ground truth, while ϵ constrains the magnitude of the perturbations under the ℓ_p-norm, ensuring that the adversarial attack remains imperceptible.

Since the true future values $\mathbf{Y}_{t+1:t+\tau}$ are typically inaccessible in practical time series forecasting, they are replaced with the predicted values $\hat{\mathbf{Y}}_{t+1:t+\tau}$ generated by the forecasting model. Consequently, Eq. 2 is reformulated as

$$\begin{aligned} \max_{\boldsymbol{\rho}_{t-T+1:t}} \quad & \mathcal{L}\left(f\left(\mathbf{X}_{t-T+1:t} + \boldsymbol{\rho}_{t-T+1:t}\right), \hat{\mathbf{Y}}_{t+1:t+\tau}\right) \\ \text{s.t.} \quad & \left\|\rho_i\right\|_p \leq \epsilon, i \in [t-T+1, t]. \end{aligned} \tag{3}$$

In practical applications, accessing the full set of detailed parameters of an LLM is typically infeasible, which forces the attacker to treat the target model as a black-box system. Additionally, acquiring the complete training dataset is impractical, meaning the attacker lacks access to this information as well. The attacker's capabilities can be summarized as follows:

- **no access to the training data**,
- **no access to internal information of the LLM-based forecasting model**,
- **no access to ground truth**,

- **the ability to query the target model**.

The threat model for adversarial attacks against LLM-based time series forecasting underscores the complexity of this task. Unlike attacks on LLMs in static applications, the attacker here cannot leverage labels for crafting attacks. Furthermore, compared to attacks on non-LLM forecasting models, the internal details of LLMs are strictly inaccessible, prohibiting the use of white-box attack techniques. This restriction highlights the increased challenge of developing effective adversarial attacks in this context.

4 DIRECTIONAL GRADIENT APPROXIMATION

Since the attacker has no access to the internal parameters of the LLM, it is not feasible to compute gradients and use them to solve the optimization problem presented in Eq. 3. This results in a gradient-free optimization problem. To address this, we propose a gradient-free optimization approach, referred to as targeted attack with **D**irectional **G**radient **A**pproximation (DGA), aimed at generating perturbations that can effectively deceive LLM-based time series forecasting models.

We first adjust our objective to focus on misleading the forecasting model into producing outputs that closely resemble an anomalous sequence, rather than simply deviating from its normal predictions. Accordingly, the optimization problem in Eq. 3 is reformulated as

$$\begin{aligned} \min_{\boldsymbol{\rho}_{t-T+1:t}} \quad & \mathcal{L}\left(f\left(\mathbf{X}_{t-T+1:t}+\boldsymbol{\rho}_{t-T+1:t}\right), \mathcal{Y}\right) \\ \text{s.t.} \quad & \left\|\rho_i\right\|_p \leq \epsilon, i \in [t-T+1, t], \end{aligned} \tag{4}$$

where $\mathcal{Y}$ represents the targeted anomalous time series.

Supposing $\boldsymbol{\theta}_{t-T+1:t}$ denote a random small signal, the gradient, $\boldsymbol{g}_{t-T+1:t}$, which approximates the direction from the normal output to the targeted anomalous output, can be expressed as

$$\boldsymbol{g}_{t-T+1:t} = \frac{\mathcal{L}\left(\mathcal{Y}-f\left(\mathbf{X}_{t-T+1:t}+\boldsymbol{\theta}_{t-T+1:t}\right)\right)-\mathcal{L}\left(\mathcal{Y}-f\left(\mathbf{X}_{t-T+1:t}\right)\right)}{\boldsymbol{\theta}_{t-T+1:t}}. \tag{5}$$

Supposing ℓ_1-norm is applied in Eq. 4, the magnitude of the perturbation is strictly constrained to be imperceptible. The perturbation, $\boldsymbol{\rho}_{t-T+1:t}$, can be computed from the approximated gradient, and the temporary adversarial example, $\mathbf{X'}_{t-T+1:t}$, is generated as

$$\begin{aligned} \mathbf{X'}_{t-T+1:t} &= \mathbf{X}_{t-T+1:t}+\boldsymbol{\rho}_{t-T+1:t} \\ &= \mathbf{X}_{t-T+1:t}+\epsilon \cdot \operatorname{sign}\left(\boldsymbol{g}_{t-T+1:t}\right), \end{aligned} \tag{6}$$

where $\operatorname{sign}(\cdot)$ denotes the signum function.

A time series forecasting model that produces Gaussian White Noise (GWN) as its output is considered to generate an anomalous prediction. Consequently, GWN can be utilized as the target sequence in Eq. 6, formulated as $\mathcal{Y} \sim \mathcal{N}(\mu, \sigma)$, where μ and σ represent the mean and the standard deviation, respectively. Empirically, the mean and standard deviation of the input data can be used to generate GWN. This results in a situation where a temporally correlated time series is misleadingly predicted as independent and identically distributed (i.i.d.) noise. This approach highlights the model's inability to preserve temporal correlations when subjected to adversarial perturbations, thereby reinforcing the effectiveness of the adversarial attack.

5 EXPERIMENTS

5.1 Datasets

To evaluate the proposed DGA and gain a further understanding of the vulnerability of LLM-based forecasting, We conducted experiments using five widely recognized real-world datasets that cover a broad range of time series forecasting tasks:

- **ETTh1 and ETTh2 (Electricity Transformer Temperature Hourly) (Zhou et al., 2021)**: These datasets consist of two years of hourly recorded data from electricity transformers, capturing temperature and power consumption variables.
- **IstanbulTraffic (Gruver et al., 2024)**: This dataset contains hourly measurements of road traffic volumes across different sensors. It captures temporal dependencies related to traffic patterns, which are dynamic and fluctuating time series.
- **Weather (Wu et al., 2023)**: This dataset comprises meteorological data, including variables such as temperature, humidity, and wind speed, recorded hourly. It provides a challenging forecasting task due to the inherent variability and complexity of weather patterns.
- **Exchange (Lai et al., 2018)**: This dataset consists of daily exchange rates from eight foreign countries—Australia, the United Kingdom, Canada, Switzerland, China, Japan, New Zealand, and Singapore—covering the period from 1990 to 2016.

These diverse datasets allow us to evaluate the adversarial robustness of LLMs across different types of temporal dynamics and forecasting challenges. In our experiments, 50% of the data is used for training, while the remaining data is split evenly: 25% for validation and 25% for testing. We use a 96-step historical time window as input to the forecasting model, which pre-

dicts the subsequent 48-step future values. It should be noted that the attacker does not access either the training or validation part.

5.2 Target Models

To assess the impact of adversarial attacks on LLMs for time series forecasting, we selected three state-of-the-art LLM-based forecasting models as baselines, which together represent three common forms of LLM application for time series tasks:

- **TimeGPT** (Garza and Mergenthaler-Canseco, 2023): A large model specifically pre-trained with a vast amount of time series data. It uses advanced attention mechanisms and temporal encoding to capture complex patterns in sequential data, making it a leading LLM designed explicitly for time series forecasting. Its pre-training, which is conducted from scratch using vast amounts of time series data, allows it to serve as a robust and versatile tool for a wide range of time-dependent applications.
- **LLMTime** (Gruver et al., 2024): This model treats time series forecasting as a next-token prediction task, using LLM architectures like GPT and LLaMa. By converting time series data into numerical sequences, LLMTime enables these models to apply their sequence prediction strengths to time series. To test the robustness of our adversarial attacks, we experimented with base models including GPT-3.5, GPT-4, LLaMa, and Mistral, assessing their resilience when adapted from natural language processing to time series forecasting.
- **TimeLLM** (Jin et al., 2023a): This model presents a novel approach for time series forecasting by adapting LLMs by reprogramming input time series into textual representations that are more compatible with LLMs, allowing the models to perform time series forecasting tasks without altering their pre-trained structures. The key innovation is the Prompt-as-Prefix (PaP) technique, which augments input context to guide the LLM in transforming reprogrammed data into accurate forecasts.
- **TimeNet** (Wu et al., 2023) and **iTransformer** (Liu et al., 2024d)are two supervised forecasting models that leverage the attention-based architecture of transformers, enabling them to capture long-term dependencies in time series data effectively. These models provide strong performance in time series forecasting tasks, and in this study, they are used in the performance and robustness comparison between LLM-based forecasting models with non-LLM models.

Both TimeGPT and LLMTime operate as zero-shot forecasters, making predictions for each time series without prior exposure to the dataset. In contrast, TimeLLM is fine-tuned using 10% of each dataset for every task. On the other hand, the two non-LLM forecasters, TimeNet and iTransformer, are trained using the full available training data for each dataset.

While we selected only three baseline LLM-based models for this study, the setup encompasses the primary approaches to LLM-based time series forecasting: pre-training a large model specifically for time series data (e.g., TimeGPT), leveraging well-developed general-purpose language models (e.g., LLMTime), and fine-tuning language models from other domains for time series forecasting (e.g., TimeLLM). This comprehensive selection provides a representative overview of the key strategies in adapting LLMs for time series tasks.

5.3 Experimental Procedures

We designed a series of experiments to evaluate the vulnerability of the baseline LLM models to adversarial attacks. For each model and dataset combination, we conducted the following procedures: (i) we applied targeted perturbations to the input data, carefully maintaining the overall structure of the original time series while subtly altering the data to mislead the LLMs' forecasting predictions; (ii) we introduced GWN with the same perturbation intensity; (iii) forecasting accuracy was measured using Mean Absolute Error (MAE) and Mean Squared Error (MSE), which allowed us to quantify the performance degradation caused by adversarial attacks compared to Gaussian noise.

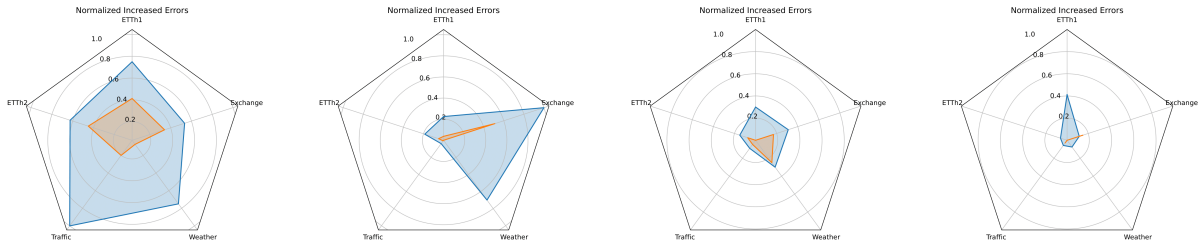
5.4 Overall Comparison

As shown in Table 1, the experimental results demonstrate that the designed adversarial attacks significantly degraded forecasting performance across all datasets, as indicated by increased MSE and MAE values. Compared to GWN of the same perturbation intensity, our attacks had a much more detrimental effect on the models' predictions. For TimeGPT, which is pre-trained with large-scale time series data, the adversarial attack led to a sharp rise in forecasting errors, demonstrating that even models specifically built for time series forecasting are vulnerable. For LLMTime, which includes GPT-3.5, GPT-4, LLaMa, and Mistral as base models, the adversarial attack was even more pronounced.

Figure 2 demonstrates the robustness comparison between LLM-based forecasting models (LLMTime with GPT-4 and TimeGPT) and non-LLM models (iTransformer and TimeNet) under the proposed black-box adversarial attack, DGA. The larger blue areas in the radar charts for the LLM-based models indicate that they experience significantly higher increases in errors,

Table 1: Results for univariate time series forecasting with a consistent input length of 96 and an output length of 48 across all models and datasets. A lower MSE or MAE indicates better prediction performance. The perturbation scale is set to 2% of the mean value of each dataset. Bold text highlights the worst performance for each dataset and model combination.

Models	LLMTime w/ GPT-3.5		LLMTime w/ GPT-4		LLMTime w/ LLaMa 2		LLMTime w/ Mistral		Time-LLM w/ GPT-2		TimeGPT (2024)		iTransformer (2024)		TimesNet (2023)	
Metrcis	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	0.073	0.213	0.071	0.202	0.086	0.244	0.097	0.274	0.089	0.202	0.059	0.192	0.071	0.218	0.073	0.202
w/ GWN	0.077	0.219	0.076	0.213	0.087	0.237	0.094	0.291	**0.102**	0.231	0.059	0.193	0.072	0.216	0.074	0.202
w/ DGA	**0.085**	**0.249**	**0.083**	**0.232**	**0.091**	**0.251**	**0.098**	**0.295**	0.099	**0.248**	**0.060**	**0.198**	**0.075**	**0.226**	**0.081**	**0.213**
ETTh2	0.263	0.372	0.155	0.267	0.237	0.373	0.277	0.492	0.238	0.361	0.161	0.297	0.171	0.296	0.166	0.316
w/ GWN	0.263	0.342	0.175	0.303	0.231	**0.429**	0.346	0.505	0.235	0.355	0.160	0.301	**0.181**	0.302	0.166	0.316
w/ DGA	**0.275**	**0.408**	**0.201**	**0.327**	**0.257**	0.425	**0.356**	**0.554**	**0.302**	**0.441**	**0.171**	**0.312**	0.179	**0.308**	**0.169**	**0.321**
IstanbulTraffic	0.837	0.844	0.805	0.779	0.891	1.005	0.826	0.973	0.995	1.013	1.890	1.201	1.081	0.995	1.095	1.022
w/ GWN	0.882	0.908	0.883	0.864	0.917	1.063	1.054	1.031	1.123	1.221	1.848	1.204	**1.103**	1.015	1.103	1.035
w/ DGA	**0.955**	**1.073**	**1.417**	**1.214**	**0.994**	**1.083**	**1.744**	**1.217**	**1.161**	**1.328**	**1.918**	**1.218**	1.097	**1.034**	**1.155**	**1.047**
Weather	0.005	0.051	0.004	0.048	0.008	0.072	0.006	0.057	0.004	0.034	0.004	0.043	0.005	0.053	0.003	0.042
w/ GWN	0.005	0.053	0.005	0.051	0.008	0.074	**0.007**	**0.066**	0.004	0.033	0.004	0.043	**0.006**	0.063	0.003	0.042
w/ DGA	**0.006**	**0.063**	**0.006**	**0.061**	**0.009**	**0.079**	**0.007**	0.062	**0.005**	**0.052**	**0.006**	**0.071**	**0.006**	**0.065**	**0.004**	**0.045**
Exchange	0.038	0.146	0.040	0.152	0.043	0.167	0.151	0.274	0.056	0.188	0.256	0.368	0.034	0.151	0.056	0.184
w/ GWN	0.042	0.179	0.046	0.182	0.050	0.185	0.160	0.298	0.059	0.194	0.329	0.413	0.044	0.166	**0.065**	**0.195**
w/ DGA	**0.058**	**0.224**	**0.068**	**0.199**	**0.069**	**0.213**	**0.219**	**0.303**	**0.077**	**0.256**	**0.578**	**0.556**	**0.049**	**0.178**	0.062	0.194



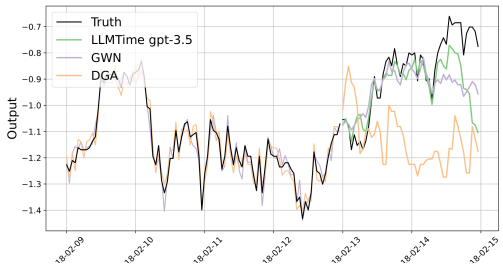
(a) LLMTime w/ GPT-4 (b) TimeGPT (c) iTransformer (d) TimeNet

Figure 2: Robustness comparison between LLM-based forecasting models and lighter models. These figures highlight each model's relative robustness across various datasets. The blue and orange shaded areas represent the normalized increase in MAE for each model under the influence of DGA and GWN perturbations, respectively. A larger shaded area indicates greater vulnerability to perturbations.

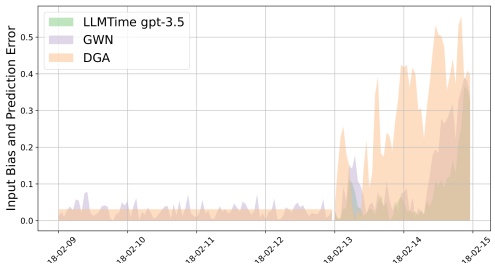
across all datasets (ETTh1, ETTh2, Exchange, Traffic, and Weather). In contrast, the non-LLM models, iTransformer and TimeNet, exhibit much smaller error increases, suggesting that they are more robust to adversarial attacks. This analysis highlights that LLM-based models are generally less resilient than non-LLM models, making them more vulnerable to adversarial manipulations in time series forecasting.

As illustrated in Figure 3, the attack caused a clear divergence between the forecasted values and the true time series, with all different variants of LLMTime exhibiting larger deviations compared to GWN. LLMTime with GPT-3.5 in particular, showed significant susceptibility, with their errors increasing substantially under adversarial conditions.

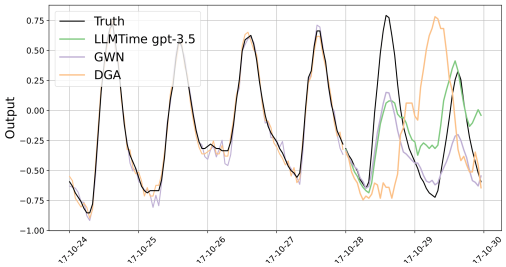
Across all models and datasets, the adversarial perturbations introduced significantly greater disruptions than GWN, clearly impacting the predictions and demonstrating the precision of the attack in destabilizing LLM-based forecasting. The magnitude of the degradation in predictive accuracy highlights the effectiveness of the proposed DGA. These findings emphasize the urgent need for robust defensive strategies to safeguard LLM-based forecasting models against adversarial threats. The current vulnerability of these models presents a significant challenge for real-world applications, particularly in high-stakes domains such as financial forecasting, energy demand prediction, and intelligent transportation systems. Without adequate defenses, adversarial attacks could lead to erroneous predictions, resulting in potential financial losses, inefficient resource allocations, or compromised safety in critical infrastructure.



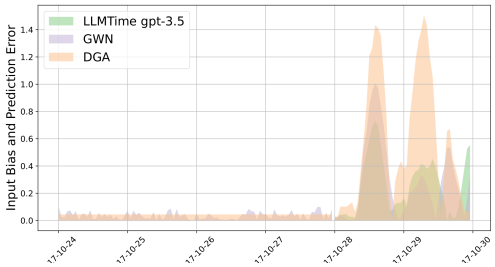
(a) ETTh1, forecast: 02/13-02/15



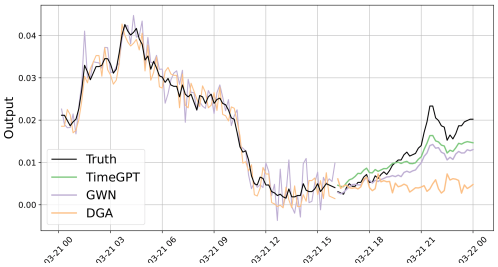
(b) ETTh1, input bias and prediction error



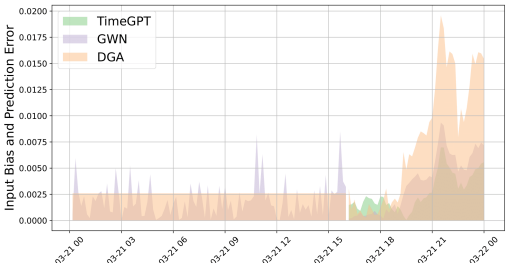
(c) ETTh2, forecast: 10/28-10/30



(d) ETTh2, input bias and prediction error

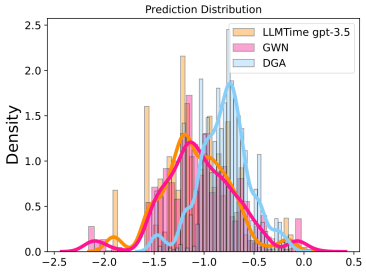


(e) weather, forecast: 3/21/16pm-3/22/12am

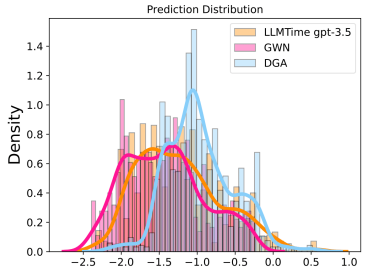


(f) weather, input bias and prediction error

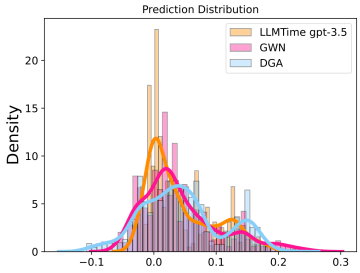
Figure 3: (a) Inputs and predictions from LLMTime (using GPT-3.5) on the ETTh1 dataset; (b) Input bias and prediction errors corresponding to (a); (c) Inputs and predictions from LLMTime (using GPT-3.5) on the ETTh2 dataset; (d) Input bias and prediction errors corresponding to (c); (e) Inputs and predictions from TimeGPT on the weather dataset; (f) Input bias and prediction errors corresponding to (e). This figure highlights the greater disruption caused by DGA compared to GWN, showing significant deviations from the ground truth.



(a) ETTh1, LLMTime w/GPT-3.5

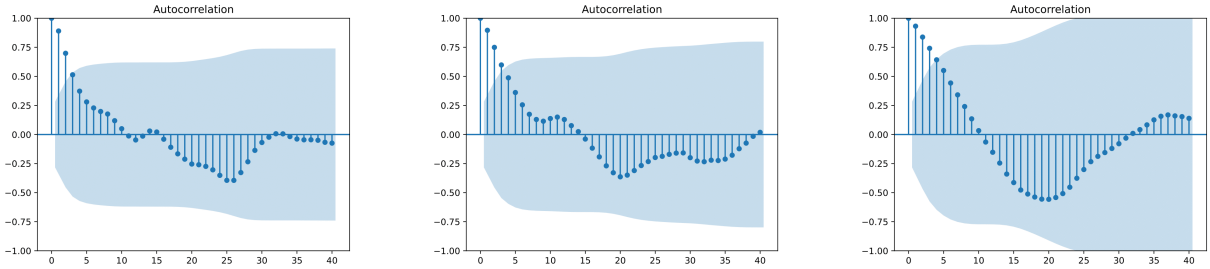


(b) ETTh2, LLMTime w/GPT-3.5



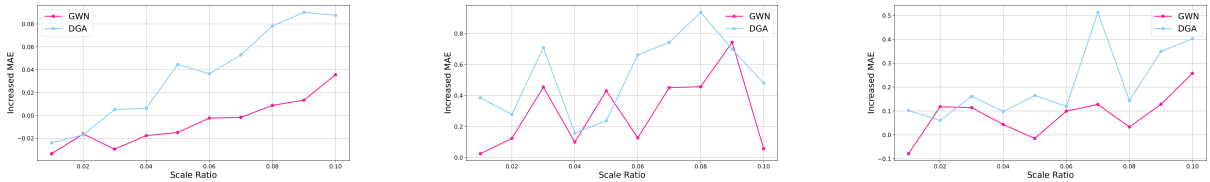
(c) Weather, LLMTime w/GPT-4

Figure 4: Prediction distribution comparison for LLMTime (using GPT-3.5, GPT-4) across different datasets under clean input, GWN, and DGA.



(a) Clean Input (b) Input with GWN (c) Input with DGA

Figure 5: Autocorrelation function curve comparison on ETTh2 by LLMTime using GPT-3.5



(a) Weather, LLMTime w/GPT-4 (b) Traffic, LLMTime w/Mistral (c) Exchange, TimeGPT

Figure 6: Hyperparameter study on the effects of different scale ratios under GWN and DGA.

5.5 Interpretation Study

Figure 4 illustrates the distribution shift in predictions caused by targeted perturbations on the LLM-based forecasting model. The proposed DGA method is designed to mislead the forecasting model, causing its predictions to resemble a random walk. As depicted in Figure 4, the "blue" shaded area, representing the perturbed prediction distribution, deviates significantly from the original "yellow" distribution and approaches a normal distribution. This shift underscores how subtle, well-crafted perturbations can manipulate the model into producing inaccurate forecasts. The effect of DGA-induced perturbations is pronounced when examining the prediction distributions, where errors are much more severe compared to the minor disruptions caused by GWN. These findings suggest that LLM-based forecasting models are highly susceptible to adversarial attacks that exploit the model's inherent vulnerabilities.

Additionally, the autocorrelation function (ACF) analysis provides further evidence of the detrimental impact of these adversarial attacks. Normally, LLMs demonstrate a strong ability to capture the temporal dependencies within time series data, maintaining coherent relationships between consecutive data points. However, as illustrated in Figure 5, when subjected to adversarial perturbations, these temporal dependencies break down, resulting in forecasts that no longer reflect the true underlying trends of the data. The disrupted autocorrelation patterns clearly illustrate the model's difficulty in preserving the natural flow of time series data under attack. In contrast, the addition of Gaussian noise, though introducing some fluctuations, does not cause the same level of disruption, maintaining a closer relationship to the clean data.

5.6 Hyperparameter Study

We systematically analyze the impact of varying scale ratios on model performance under both GWN and DGA adversarial perturbations. The vertical axis in Figure 6 represents the increase in MAE, serving as a measure of the extent to which each attack degrades the precision of the forecast. This experiment is conducted across three distinct datasets using three LLM-based forecasting models.

As illustrated in Figure 6, the DGA consistently induces a more pronounced increase in MAE compared to GWN as the scale ratio increases. This trend highlights the greater effectiveness of the DGA in destabilizing model predictions, as it is specifically designed to exploit model vulnerabilities rather than introduce random noise. To achieve a balance between imperceptibility and attack effectiveness, we determine that an optimal perturbation scale can be set at 2% of the mean value of the given dataset. This choice ensures that the adversarial perturbation remains subtle enough to

evade detection while still significantly impairing the forecasting model's performance.

5.7 Discussion of Mitigation Methods

Challenges of Adversarial Training in LLM4TS: Adversarial training effectively mitigates attacks but poses challenges for LLM-based time series forecasting due to the high computational costs of pretraining on large datasets. Its iterative nature further escalates costs, as it requires generating adversarial examples and optimizing against them during training. Retrofitting pre-trained models like TimeGPT with adversarial defenses is similarly impractical, often demanding months of computation and significant pipeline modifications.

Alternative Mitigation Strategies for LLM4TS: To overcome these challenges, preprocessing-based methods provide a practical alternative. Filter-based defenses reform time series data before forecasting, while machine learning-based anomaly detection identifies and filters adversarial inputs. These computationally efficient approaches are particularly effective against black-box adversarial attacks, offering a promising defense for LLM4TS models. Future work will focus on refining and evaluating these strategies.

6 CONCLUSION

In this study, we demonstrated the significant vulnerabilities of LLM-based models for time series forecasting to adversarial attacks. Through a comprehensive evaluation of TimeGPT and LLMTime (with GPT-3.5, GPT-4, LLaMa, and Mistral as base models), we found that targeted adversarial perturbations, generated using Directional Gradient Approximation (DGA), caused substantial increases in prediction errors. These attacks were far more damaging than Gaussian White Noise (GWN) of similar intensity, highlighting the precision and effectiveness of the adversarial strategy.

The experimental results revealed that both large, pretrained models like TimeGPT and fine-tuned models such as LLMTime are highly susceptible to adversarial manipulation. The proposed attack can significantly degrade model performance across various datasets. This poses serious challenges for the deployment of LLMs in real-world time series applications, where reliability is critical.

Our findings emphasize the need for future research to focus on developing robust defense mechanisms to mitigate adversarial threats and enhance the resilience of LLM-based time series forecasting models. Without such protections, these models remain vulnerable to attacks that could undermine their practical utility in high-stakes environments.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

Cristian Challu, Kin G Olivares, Boris N Oreshkin, Federico Garza Ramirez, Max Mergenthaler Canseco, and Artur Dubrawski. Nhits: Neural hierarchical interpolation for time series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 6989–6997, 2023.

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

Everette S Gardner Jr. Exponential smoothing: The state of the art. *Journal of forecasting*, 4(1):1–28, 1985.

Azul Garza and Max Mergenthaler-Canseco. Timegpt-1. *arXiv preprint arXiv:2310.03589*, 2023.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *arXiv preprint arXiv:1412.6572*, 2014.

Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. Not what you've signed up for: Compromising real-world llm-integrated applications with indirect prompt injection. In *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security*, pages 79–90, 2023.

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.

Chuan Guo, Alexandre Sablayrolles, Hervé Jégou, and Douwe Kiela. Gradient-based adversarial attacks against text transformers. *arXiv preprint arXiv:2104.13733*, 2021.

Yushan Jiang, Zijie Pan, Xikun Zhang, Sahil Garg, Anderson Schneider, Yuriy Nevmyvaka, and Dongjin Song. Empowering time series analysis with large language models: A survey. *arXiv preprint arXiv:2402.03182*, 2024.

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*, 2023a.

Ming Jin, Qingsong Wen, Yuxuan Liang, Chaoli Zhang, Siqiao Xue, Xue Wang, James Zhang, Yi Wang, Haifeng Chen, Xiaoli Li, et al. Large models for time series and spatio-temporal data: A survey and outlook. *arXiv preprint arXiv:2310.10196*, 2023b.

Konstantinos Kalpakis, Dhiral Gada, and Vasundhara Puttagunta. Distance measures for effective clustering of arima time-series. In *Proceedings 2001 IEEE international conference on data mining*, pages 273–280. IEEE, 2001.

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.

Chenxi Liu, Qianxiong Xu, Hao Miao, Sun Yang, Lingzheng Zhang, Cheng Long, Ziyue Li, and Rui Zhao. Timecma: Towards llm-empowered time series forecasting via cross-modality alignment. *arXiv preprint arXiv:2406.01638*, 2024a.

Fan Liu, Hao Liu, and Wenzhao Jiang. Practical adversarial attacks on spatiotemporal traffic forecasting models. *Advances in Neural Information Processing Systems*, 35:19035–19047, 2022a.

Fuqiang Liu, Luis Miranda-Moreno, and Lijun Sun. Spatially focused attack against spatiotemporal graph neural networks. *arXiv preprint arXiv:2109.04608*, 2021.

Fuqiang Liu, Jiawei Wang, Jingbo Tian, Dingyi Zhuang, Luis Miranda-Moreno, and Lijun Sun. A universal framework of spatiotemporal bias block for long-term traffic forecasting. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):19064–19075, 2022b.

Fuqiang Liu, Jingbo Tian, Luis Miranda-Moreno, and Lijun Sun. Adversarial danger identification on temporally dynamic graphs. *IEEE Transactions on Neural Networks and Learning Systems*, 35(4):4744–4755, 2023.

Linbo Liu, Youngsuk Park, Trong Nghia Hoang, Hilaf Hasson, and Jun Huan. Robust multivariate time-series forecasting: Adversarial attacks and defense mechanisms. *arXiv preprint arXiv:2207.09572*, 2022c.

Peiyuan Liu, Hang Guo, Tao Dai, Naiqi Li, Jigang Bao, Xudong Ren, Yong Jiang, and Shu-Tao Xia. Taming pre-trained llms for generalised time series forecasting via cross-modal knowledge distillation. *arXiv preprint arXiv:2403.07300*, 2024b.

Xiaogeng Liu, Zhiyuan Yu, Yizhe Zhang, Ning Zhang, and Chaowei Xiao. Automatic and universal prompt injection attacks against large language models. *arXiv preprint arXiv:2403.04957*, 2024c.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. *International Conference on Learning Representations*, 2024d.

Aleksander Madry. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

John X Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. *arXiv preprint arXiv:2005.05909*, 2020.

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.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International journal of forecasting*, 36(3):1181–1191, 2020.

Leo Schwinn, David Dobre, Sophie Xhonneux, Gauthier Gidel, and Stephan Gunnemann. Soft prompt threats: Attacking safety alignment and unlearning in open-source llms through the embedding space. *arXiv preprint arXiv:2402.09063*, 2024.

Ayush Singh, Navpreet Singh, and Shubham Vatsal. Robustness of llms to perturbations in text. *arXiv preprint arXiv:2407.08989*, 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.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. Jailbroken: How does llm safety training fail? *Advances in Neural Information Processing Systems*, 36, 2024.

Xingxing Wei, Siyuan Liang, Ning Chen, and Xiaochun Cao. Transferable adversarial attacks for image and video object detection. *arXiv preprint arXiv:1811.12641*, 2018.

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.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. *International Conference on Learning Representations*, 2023.

Han Xu, Yao Ma, Hao-Chen Liu, Debayan Deb, Hui Liu, Ji-Liang Tang, and Anil K Jain. Adversarial attacks and defenses in images, graphs and text: A review. *International journal of automation and computing*, 17:151–178, 2020.

Jiaqi Xue, Mengxin Zheng, Ting Hua, Yilin Shen, Yepeng Liu, Ladislau Bölöni, and Qian Lou. Trojllm: A black-box trojan prompt attack on large language models. *Advances in Neural Information Processing Systems*, 36, 2024.

Zhiyuan Yu, Xiaogeng Liu, Shunning Liang, Zach Cameron, Chaowei Xiao, and Ning Zhang. Don't listen to me: Understanding and exploring jailbreak prompts of large language models. *arXiv preprint arXiv:2403.17336*, 2024.

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.

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.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Not Applicable]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Not Applicable]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Not Applicable]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]