LARGE LANGUAGE MODELS FOR TIME SERIES FORECASTING

Hussain Kaide Johar Manasi
Ritesh Patil &
Jinendra Shrimal

Institute For Insight
J.Mack Robinson College of Business
Georgia State University

PROBLEM STATEMENT

Time series forecasting has long been a fundamental pursuit aimed at mitigating uncertainty in various domains. However, traditional forecasting methods often encounter limitations in capturing the complex patterns inherent in time series data. Leveraging Language Models (LLMs), traditionally designed for textual data, presents a promising avenue to enhance the accuracy and effectiveness of time series forecasting. By applying techniques such as prompting, hyperparameter tuning, pooling, and other innovative methodologies, we aim to explore the latent potential of LLMs in numerical applications like time series forecasting. This approach not only bridges the gap between language-centric models and numerical forecasting tasks but also offers a novel perspective on tackling the challenges associated with temporal data analysis. Through this research endeavor, we seek to amplify the benefits derived from LLMs, thereby advancing the state-of-the-art in time series forecasting and empowering decision-makers with more robust and reliable predictive capabilities.

CHALLENGES FOR LLM IN TIME SERIES FORECASTING

•While several surveys offer a broad perspective on large models for time series in general, they do not specifically focus on LLMs or the key challenge of bridging modality gap, which stems from LLMs being originally trained on discrete textual data, in contrast to the continuous numerical nature of time series.
•Prompting-based methods face inefficiencies when applied to numerical data with high precision, as well as multivariate time series, as they necessitate transforming each dimension into separate univariate time series, leading to excessively long inputs. Moreover, they exhibit reduced efficiency for long-term predictions due to the computational demands of generating extended sequences. These methods prove more effective when dealing with simple numerical data intertwined with textual information, such as opening and closing stock prices in financial news articles.
•Existing papers applying Language Models (LLMs) for time series analysis often concentrate on a single modality and task at a time, such as forecasting, classification, or text generation, without accommodating simultaneous multimodal and multitask analysis.
•The analysis is constrained to a univariate model due to the variability in the multivariate dimension obtained by grouping the time series for each dataset.
•To utilize Transformer-based architectures, vector-valued inputs are necessary. However, the frequency of time series in our dataset varies, necessitating the vectorization of univariate data in a manner that accommodates the specific frequencies of the datasets within our corpus.
•The current backbone structures and prompt techniques in language models fail to comprehensively capture the evolution of temporal patterns and the progression of interrelated dynamics over time, crucial aspects for effective time series modeling.
•There is limited research on Language Models (LLMs) for multimodal data incorporating time series.
•Publicly available datasets for time series lack the required scale and volume.
•Characteristics such as frequency, sparsity, trend, seasonality, stationarity, and heteroscedasticity pose distinct challenges for both local and global models.
•Statistical models often assume a linear relationship and stationary time series data, which may not accurately reflect real-world scenarios. Furthermore, these models typically require extensive manual tuning and domain knowledge to select appropriate parameters for a specific forecasting task.

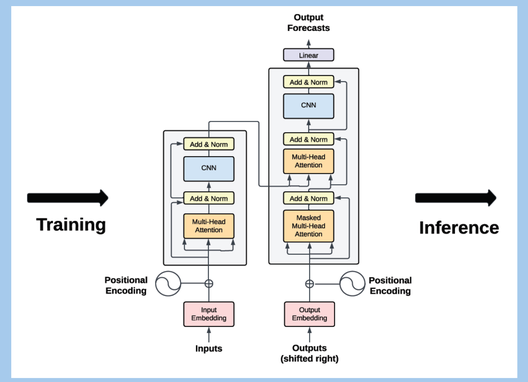

Fig1. TIMEGPT Architecture

POPULAR LLM MODELS FOR TIME SERIES

1.TEMPO (Time sEries proMpt POol)

TEMPO, a novel approach to time series representation learning, leverages two critical inductive biases inherent in the task: the decomposition of complex interactions between trend, seasonal, and residual components, and the utilization of selection-based prompts for distribution adaptation in non-stationary time series. It encompasses two key analytical components focusing on specific time series patterns and obtaining universal insights from past data sequences. Each temporal input is mapped to its corresponding hidden space to construct the time series input embedding of the generative pre-trained transformer (GPT), with TEMPO utilizing a prompt pool to efficiently tune the GPT. Notably, TEMPO achieves substantial improvements in Mean Absolute Error (MAE) compared to state-of-the-art models for time series forecasting, particularly with prediction lengths of 96 and 192. Additionally, the introduction of a new multimodal time series dataset, TETS (Text for Time Series), underscores TEMPO's effectiveness, with significant improvements in SMAPE observed. By combining time-series patching with temporal encoding, TEMPO extracts local semantics, enhancing the historical horizon while reducing redundancy. The incorporation of prompting approaches, coupled with LLM and time series decomposition, significantly enhances accuracy and efficiency in forecasting tasks. Furthermore, the strong generalizability and data efficiency of TEMPO highlight its potential as a foundational time series forecasting model, with SHAP (SHapley Additive exPlanations) values indicating the dominant influence of trend components on predictions. The integration of prompt pool and seasonal trend decomposition provides the model with the ability to recall relevant knowledge from past periods based on input similarity, facilitating robust model optimization. Future directions may involve exploring superior LLMs and further development of foundational models for time series analysis.

2. LAG LLAMA

The model demonstrates impressive zero-shot prediction capabilities on unseen "out-of-distribution" time-series datasets, surpassing supervised baselines. Trained on a vast array of time-series datasets, the transformer model's performance is evaluated on an unseen "out-of-distribution" dataset suitable for scaling law analyses of time series foundation models. Lag-Llama outperforms or compares favorably to supervised baselines when tested zero-shot on unseen time series datasets. The primary objective of this work is to apply the foundation model approach to time-series data and explore the extent of transfer achievable across a wide range of time-series domains. The only covariates employed in this model are derived from the target values, particularly lag features, constructed from appropriate lag indices for quarterly, monthly, weekly, daily, hourly, and second-level frequencies corresponding to those in the corpus of time series data. An alternative approach to vectorizing a univariate series involves using potentially overlapped patches or segments of a certain size and stride, albeit this may lead to causally mixed vectors. Stratified sampling, where datasets are weighed by the total temporal time points, is employed to prevent over/under sampling. The training set comprises a total of 305,443 individual time series, with a batch size (B) of 100 and a learning rate (α) of 10−3. Training halts if the validation loss does not improve for 50 validation epochs, each consisting of 100 windows. The Continuous Ranked Probability Score (CRPS) metric is utilized for evaluation. Lag-Llama outperforms the baseline on the Traffic dataset after approximately 106 parameters and achieves stable zero-shot performance with various hyperparameter configurations between 106 and 107. As the model size increases, its performance improves and stabilizes across hyperparameter specifications. Further plan involves first ablating various architectural design and model selection choices, assessing the model's zero-shot performance across other datasets in a leave-one-out fashion to obtain zero-shot performance results across a spectrum of datasets for insightful analyses. Additionally, fine-tuning the pretrained models on the training splits of downstream datasets will be conducted to assess few-shot and many-shot finetuning performance across datasets. Finally, scaling both the model size and the amounts of diverse time-series training data will be explored while comparing scaling laws of this and other candidate architectures for time-series foundation models.

3. TIMEGPT

TimeGPT, the pioneering foundation model for time series, implements conformal prediction machinery for uncertainty quantification post point-forecasting emission, distinguishing itself by its direct focus on probabilistic forecasting. In evaluating the pre-trained model against established statistical, machine learning, and deep learning methods, TimeGPT showcases superior performance, efficiency, and simplicity, presenting an exciting opportunity to democratize access to precise predictions and reduce uncertainty in forecasting. This innovation represents a paradigm shift towards a more accessible and accurate forecasting practice with reduced computational complexity. TimeGPT, not based on existing large language models (LLMs), features an architecture specialized in handling time series data, trained specifically to minimize forecasting errors. Trained on over 100 billion data points spanning various domains, TimeGPT outperforms a comprehensive collection of battle-tested statistical models and state-of-the-art deep learning approaches, consistently ranking among the top performers across different frequencies. Internal tests reveal TimeGPT's impressive GPU inference speed of 0.6 milliseconds per series, significantly outperforming global models like LGBM, LSTM, and NHITS, which demonstrate a much longer average inference time per series. The contextual relevance of Transformers varies with dataset sizes, indicating their increasing benefit with larger datasets. Such insights offer practical guidance for model selection tailored to specific tasks, particularly in scenarios with constraints on dataset availability or computational resources. Looking ahead, there's a call for Informed forecasting, integrating knowledge about underlying processes like physical laws, economic principles, or medical facts, while also prompting further examination of assumptions surrounding the taxonomy of time series.

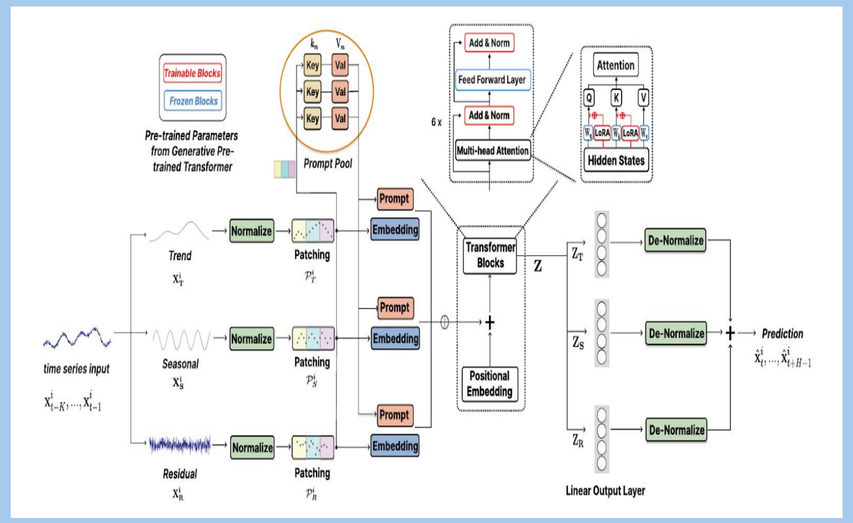

Fig2. Tempo Architecture

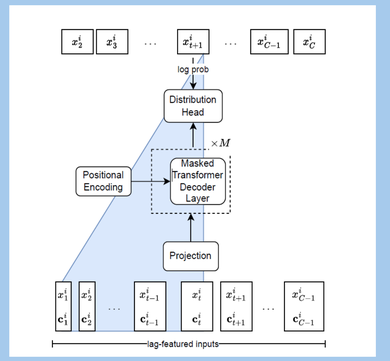

Fig3. LAG LLAMA Architecture

PROBABLE USE CASE

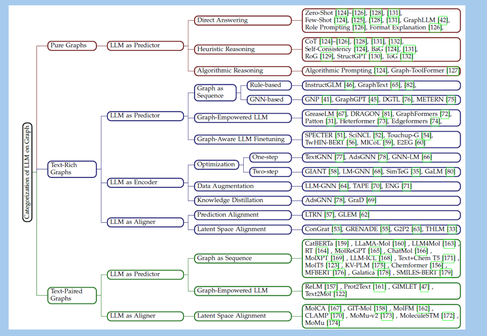

Fig3. Taxonomy of LLM on graph scenarios

One of the major domain that LLMs could step into and assist in tackling various problems is Graph-Based Applications. Initial steps should begin by outlining potential scenarios for the adoption of Language Models (LLMs) on graphs, categorizing them into three main types: pure graphs, text-attributed graphs, and text-paired graphs. It then proceeds to discuss detailed techniques for leveraging LLMs on graphs, including treating LLMs as Predictor, Encoder, or Aligner, depending on their interaction with Graph Neural Networks (GNNs). Various graph reasoning tasks such as inferring connectivity, shortest path determination, subgraph matching, and logical rule induction are highlighted within this framework. Notable tasks include the systematic categorization of graph scenarios, a comprehensive review of language models on graphs, and the proposal of prospective avenues for future exploration. Delving into foundational principles and illustrating examples from diverse domains like academic citation networks and molecular graphs, emphasizes the non-sequential nature of data in complex structures such as graphs. This also addresses challenges faced by Graph Neural Networks (GNNs), such as over-smoothing, over-squashing, interpretability, and bias. This comprehensive examination underscores the importance of integrating language models with graph structures to address real-world challenges and foster future advancements in the field.

In comparing Language Models (LMs) with Graph Transformers, three key distinctions emerge. Firstly, LMs primarily utilize word tokens, whereas Graph Transformers operate with node tokens. Secondly, LMs employ positional encoding based on the sequence of words, while Graph Transformers utilize metrics such as shortest path distance or eigenvalues of the graph Laplacian to encode node positions. Thirdly, the goals of LMs are centered around text encoding and generation, whereas Graph Transformers focus on node or graph encoding. Moving on to evaluation metrics and datasets, the predominant datasets span various domains including academia, e-commerce, social media, and Wikipedia, with tasks like node classification, link prediction, edge classification, regression, and recommendation being common. Evaluation metrics vary accordingly, including accuracy, Macro-F1, and Micro-F1 for classification tasks, Mean Reciprocal Rank (MRR), Normalized Discounted Cumulative Gain (NDCG), and Hit Ratio (Hit) for link prediction and recommendation, and mean absolute error (MAE) or root mean square error (RMSE) for regression tasks. Data splitting options encompass random, source-based, and activity-based methods, along with data balancing techniques. Graph classification often employs the Area Under the Curve (AUC) metric, while regression tasks utilize metrics such as MAE, RMSE, and R2. In text generation evaluation, the Bilingual Evaluation Understudy (BLEU) score is commonly used, while heuristic evaluation methods are adopted for molecule generation, considering factors like validity, novelty, and uniqueness.

In various domains, the application of graphs and Language Models (LLMs) offers transformative solutions. In virtual screening, LLMs assist in sifting through vast libraries of unlabeled applicants, expediting the identification of promising candidates. For molecular generation in drug and material discovery, LLMs aid in generating hypotheses that align with textual descriptions and adhere to chemical constraints. Synthesis planning benefits from graph representations of molecular structures and textual descriptions of reaction conditions, where LLMs suggest synthesis paths or serve as agents for planning tools. E-commerce operations leverage graphs with users as nodes and products as edges, facilitating tasks like item recommendation, bundle recommendation, and product understanding. Similarly, social media analysis employs user-graphs to model interactions, enabling friend recommendation, user analysis, community detection, and personalized response generation. In academia, graphs represent papers as nodes and citation relations as edges, supporting tasks such as paper recommendation, classification, and author identification. Legal domains utilize graphs for clause classification and opinion recommendation. Education graphs model coursework as nodes and relations as edges, enabling knowledge tracing and student performance prediction. Across these diverse domains, the integration of graphs and LLMs fosters efficient problem-solving and decision-making processes, showcasing their potential for advancing various fields.