Macroeconomic Forecasting with Large Language Models

Andrea Carriero[a,b] Davide Pettenuzzo[c] Shubhranshu Shekhar[c]

[a]Queen Mary University of London
[b]University of Bologna
[c]Brandeis University

2024 NBER-NSF Time Series Conference

September 20-21, 2024

Can LLMs predict the future?

03-07-2024 | TECH

LLMs can predict the future as well as—and sometimes better than—humans

A new study suggests that forecasting the future is a task that could well be outsourced to generative AI.

[Photo: Qi Yang/Getty Images]

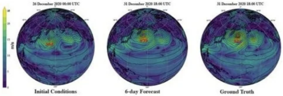
Posted in | News | Atmospheric Robotics

New LLM Offers Accurate Weather Prediction

Download PDF Copy

Reviewed by *Laura Thomson* Jul 11 2024

Scientists at the US Department of Energy's (DOE) Argonne National Laboratory, in close collaboration with researchers Aditya Grover and Tung Nguyen at the University of California, Los Angeles, began developing large artificial intelligence (AI) models for weather forecasting, known as foundation models.



Introduction

- Large Language Models (LLMs) have reshaped natural language processing
  - Have demonstrated proficiency in capturing linguistic nuances and semantic meanings
  - Used routinely for content generation, Information extraction, Sentiment analysis, Code generation and completion, Conversational AI and chatbots
- This paper focuses on a more recent development
  - Time Series Language Models (TSLMs) or Time Series Foundational Models (TSFMs)
  - Large-scale, general-purpose neural networks pre-trained on large amounts of diverse data across various frequencies and domains
  - Main idea: TSLMs build on LLM architecture to uncover intricate nonlinear relationships in time series data

Where we stand right now...

- Several TSLMs have already been productionalized and are publicly available:
  - Time-LLM (January 2024)
  - LagLlama (February 2024)
  - Moirai (Salesforce, March 2024)
  - Chronos (Amazon, March 2024)
  - Tiny Time Mixers (IBM, April 2024)
  - TimesFM (Google, April 2024)
  - Time-GPT (Nixtla, May 2024)[1]

- These models are now being used to accomplish a variety of time series related tasks, ranging from prediction and classification to anomaly detection and data imputation

[1] Use of Time-GPT is done through a proprietary API, with some limitations.

This paper

1. Are TSLMs actually useful for Macro Forecasting?
    - Focus on **zero-shot** forecasting only (working right now on fine-tuning the best performing TSLMs)
2. How do TSLMs fare relative to state-of-the-art macro time series methods?
    - Bayesian Vector Autoregressions
    - Factor models

- Note: challenges in using TSLMs for forecasting
    - "Data leakages"
    - Replicability and interpretability

Results

The setup

- We carried out a pseudo real-time forecasting exercise using the FRED monthly database and an evaluation sample ranging from 1985 to 2023
- We forecast up to 12 months out and focused on point forecast accuracy (*RMSFE*)

Main take-aways

1. Two out of five TSLMs are competitive against the AR(1) benchmark (Moirai and TimesFM)
2. TSLMs seem to perform better when dealing with less persistent series
3. In general, TSLMs show less reliability, prone to occasional unreasonable forecasts
4. Moirai and TimesFM perform generally on par with BVARs and factor models, but are less stable and at times miss the mark significantly

Literature Review

- Quickly expanding literature on using LLMs for time-series tasks
  - **Phase 1:** Chang et al. (2024),Cao et al. (2024),Gruver et al. (2024), Jin et al. (2024), Liu et al. (2024), Sun et al. (2024), Zhou et al. (2023)
  - **Phase 2:** Ekambaram et al. (2024), Jin et al. (2024), Rasul et al. (2024), Das et al. (2024), Woo et al. (2024), Garza and Mergenthaler-Canseco (2023), Ansari et al. (2024)

- Only a few papers thus far applying LLMs to macroeconomics and finance questions
  - Bybee (2023): ChatGPT-3.5 $+$ WSJ articles $\Rightarrow$ Predict financial and macroeconomic variables
  - Chen et al. (2022): BERT, RoBERTa, and OPT $+$ Thomson Reuters Real-time News Feed (RTRS) $\Rightarrow$ Predict firm-level daily returns
  - Kim et al. (2024): ChatGPT4 $+$ financial statements $\Rightarrow$ Predict future earnings
  - Faria-e Castro and Leibovici (2024): Google AI's PaLM $\Rightarrow$ Predict inflation

Outline of the talk

Time Series Language Models (TSLMs)

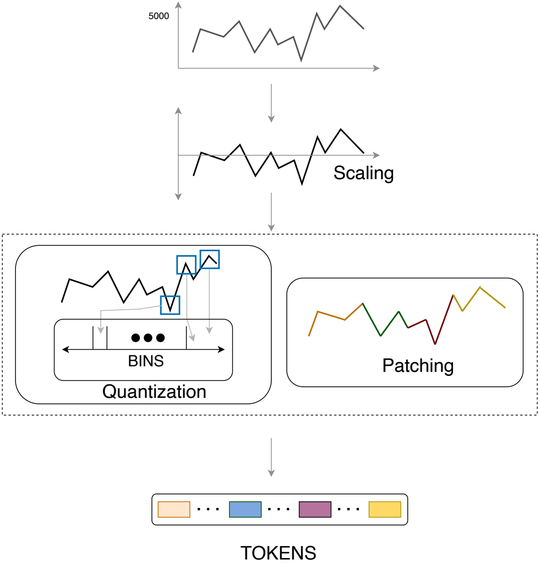
- TSLMs bridge the gap between LLMs original text data training and the numerical nature of time series data
- Main idea:
  1. Train a foundational model on a very large set of time series, $\boldsymbol{X}_{1:T} = (x_{1,1:T}, \ldots, x_{N,1:T})$ to estimate a mapping function f_θ
  2. Use f_θ along with the current and past values of time series of interest $y_{1:T}$, to forecast future values of y,

$$f_\theta \left(y_{T+1:T+h} \middle| y_{1:T}; \boldsymbol{X}_{1:T}, \theta \right).$$

Components of Time Series Language Models

- Building blocks of a TSLM:
  - Tokenization (patching, quantization, and scaling)
  - Time Series Augmentation
  - Model Architecture
  - Pre-training and Fine-tuning

Tokenization



Tokenization: (1) Scaling

- To ensure consistent processing, data going into TSLM are typically re-scaled
- Helps in optimization/learning for deep learning models
- General scaling formula:

  $$\tilde{x}_t = \frac{x_t - M}{S}$$

  $\tilde{x}_t$ is the scaled value, M is a measure of central tendency, and S is a measure of spread.
- Example (LagLlama):
  - $M =$ median of the context window
  - $S =$ inter-quartile range within the context window

Tokenization: (2) Patching

- Patching divides time series into fixed-length segments (patches)
- Patching allows to capture local patterns
- Patch size is a hyper-parameter
- Patches can be overlapping or non-overlapping
- Example: $x_{1:T} = \{4.7, 4.76, 6.8, 7.2, 6.1\}$:
    - Patch size = 3, overlap = 2:
        - $\{4.7, 4.76, 6.8\}$
        - $\{4.76, 6.8, 7.2\}$
        - $\{6.8, 7.2, 6.1\}$

Tokenization: (3) Quantization

- Quantization is used to convert numerical values into discrete tokens
- It divides the time series into a predefined number of bins $\mathbb{B}$
- Each data point assigned a token (a number between 1 and $\mathbb{B}$) based on its bin
- Example for $x_{1:T} = \{4.7, 4.76, 6.8, 7.2, 6.1\}$ with $\mathbb{B} = 4$ and uniform binning:

$$q_t = \begin{cases} 1 & \text{if } 4 \leq x_t < 5 \\ 2 & \text{if } 5 \leq x_t < 6 \\ 3 & \text{if } 6 \leq x_t < 7 \\ 4 & \text{if } 7 \leq x_t < 8 \end{cases}$$

- Resulting tokenized sequence: $q_{1:T} = \{1, 1, 3, 4, 3\}$

Tokenization: Considerations

- Choice of tokenization method depends on the specific TSLM
- Patching:
    - Larger patch sizes for high-frequency time series
    - Smaller patch sizes for low-frequency time series
- Quantization:
    - Uniform quantization: equal-sized bins
    - Data-dependent quantization: adjusts bin sizes based on data distribution
- Scaling:
    - Can be applied at global level, context window level, or patch level

Time Series Augmentation

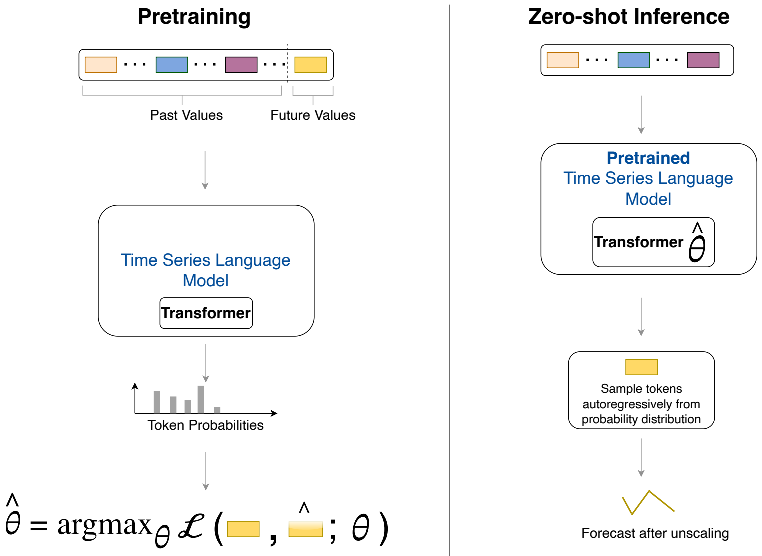
- Helps mitigate scarcity of time series data
- Used to generate more diverse training data
- Techniques include:
    - Convex combinations of existing time series
    - Combinations of ARMA processes, seasonal patterns, trends
    - Combining frequency spectrum of sequences

Model Architecture

- Transformer-based architecture is the most common
- Uses self-attention mechanisms to capture long-range dependencies
- Types:
  - Encoder-decoder (e.g., Chronos)
  - Decoder-only (e.g., TimesFM)
  - Encoder-only (e.g., Moirai)
- Some models use non-transformer architectures (e.g., TTM uses MLP)

Figure

Pre-training and Inference



Pre-training and Fine-tuning

- Pre-training
  - Feeds large amount of time series data to learn general patterns
  - Uses self-supervised techniques
  - Example: LagLlama maximizes likelihood of observed future values
- Fine-tuning
  - Adapts pre-trained model to specific tasks
  - Updates existing knowledge to handle task-specific data
  - Can improve performance compared to zero-shot predictions

TSLM Training Details

Model	Release date	Training datasets (domains)	Size	Multivariate
LagLlama	Feb 2024	Traffic, Uber TLC, Electricity, London Smart Meters, Solar power, Wind farms, KDD Cup 2018, Sunspot, Beijing Air quality, Air Quality UC Irvine Repository, Huawei cloud, Econ/Fin*	352M tokens	No
Moirai	Mar 2024	Energy, Transport, Climate, CloudOps, Web, Sales, Nature, Econ/Fin*, Healthcare	27B obs.	Yes
TTM	April 2024	Electricity, Web traffic, Solar power, Wind farms, Energy consumption, KDD Cup 2018, Sunspot, Australian weather, US births, Bitcoin, Econ/Fin*	1B obs.	Yes
Time-GPT	May 2024	Finance, economics, Demographics, Healthcare, Weather, IoT sensor data, Energy, Web traffic, Sales, Transport, and Banking	100B obs.	No
TimesFM	May 2024	Google Trends, Wiki Page views, M4 Competion, Electricity and the Traffic data, Weather data, Synthetic Time Series Data	100B obs.	No

Table: Training datasets for the various TSLMs considered in this paper. * indicates that the training data include the Monash forecast repository, and therefore includes a large part of the FRED-MD dataset.

1. Bayesian VAR with Natural Conjugate Priors

- Collect all N time series of interest in $y_t = (y_{1t}, ..., y_{Nt})$ and write a VAR(p) model as:

$$y_t = \Phi_c + \Phi_1 y_{t-1} + \Phi_2 y_{t-2} + ... + \Phi_p y_{t-p} + \varepsilon_t; \ \varepsilon_t \sim i.i.d.N(0, \Sigma)$$

- Natural conjugate N-IW prior $+$ Minnesota-style layout

$$\Phi|\Sigma \sim N(\Phi_0, \Sigma \otimes \Omega_0), \ \Sigma \sim IW(S_0, v_0)$$

- Augment prior with "sum of coefficient" and "single unit root" to to avoid the deterministic components to "take over"
- Conjugacy and Kronecker structure in the priors keep computations manageable even in large systems, and yield marginal likelihood in closed form $\Rightarrow$ Exploit this result to optimize prior hyperparameters

2. Bayesian VAR with Asymmetric Conjugate Priors

- Natural conjugate prior rules out cross-variable shrinkage
- Chan (2022) extended this setup to allows for asymmetry in the prior while maintaining conjugacy. Starting point is the BVAR in its structural form:

$$Ay_t = b + B_1 y_{t-1} + B_2 y_{t-2} + \ldots + B_p y_{t-p} + u_t; \quad u_t \sim i.i.d.\ N(0, D)$$

  where A is a lower triangular matrix and D is diagonal. This allows for estimation recursively, one equation at a time

- Prior assumes that all the BVAR parameters are a priori independent across equations
- As in the previous case, prior is augmented with "sum of coefficient" and "single unit root" to to avoid the deterministic components to "take over"

3. Factor Model

- Factor models are another class of models that has repeatedly shown to be well suited for macroeconomic forecasting

- We proceeed in two steps:

  1. A set of static factors is estimated from the whole cross section of available data using PCA + EM (Stock and Watson, 2002) to handle missing values

  2. Use the extracted first factor to augment an autoregression of the i-th series with its lag values, i.e.:

$$y_{i,t} = \alpha_h + \beta_h(L)\widehat{f}_{1,t-h} + \gamma_h(L)y_{i,t-h} + \varepsilon_{i,t}$$

- Direct h-step ahead forecast are given by

$$\widehat{y}_{i,t+h} = \widehat{\alpha}_h + \widehat{\beta}_h(L)\widehat{f}_{1,t} + \widehat{\gamma}_h(L)y_{i,t}$$

Empirical Application

- US monthly macro time series spanning January 1959 to December 2023
- Data source: Federal Reserve Economic Data Monthly Dataset (FRED-MD) at https://fred.stlouisfed.org.
- FRED-MD covers 120+ key macroeconomic variables (output, prices, interest rates, etc.)
- Use data from January 1960 to December 1984 for initial parameter estimates
- Forecast from January 1985 to December 2019 (2023) using both pre-trained TSLMs and econometrics models
- BVARs and factor models estimated using an expanding window approach and predictive simulation for $h = 1$ to 12 months ahead
- Model sizes: Medium (19 variables), Large (39 variables), X-large (120 variables)

Variables in Medium VAR

Abbreviation	Description	Transformation
PAYEMS	All Employees: Total nonfarm	5
INDPRO	IP Index	5
FEDFUNDS	Effective Federal Funds Rate	1
UNRATE	Civilian Unemployment Rate	1
RPI	Real personal income	5
DPCERA3M086SBEA	Real PCE	5
CMRMTSPLx	Real Manu. and TradeIndustries Sales	5
CUMFNS	Capacity Utilization: Manufacturing	1
CES0600000007	Avg Weekly Hours: Goods-Producing	1
HOUST	Housing Starts, Total	4
S&P 500	S&P's Common Stock Price Index: Composite	5
T1YFFM	1-Year Treasury C Minus FEDFUNDS	1
T10YFFM	10-Year Treasury C Minus FEDFUNDS	1
BAAFFM	Moodys Baa Corporate Bond Minus FEDFUNDS	1
EXUSUKx	U.S.-UK Foreign Exchange Rate	5
WPSFD49207	PPI: Final Demand: Finished Goods	5
PPICMM	PPI: Metals and metal products	5
PCEPI	Personal Consumption Expenditures	5
CES0600000008	Avg Hourly Earnings: Goods-Producing	6

Table: Variables in the Medium model. (1) no transformation, (2) Δx_t, (5) $\Delta \log(x_t)$, (6) $\Delta^2 \log(x_t)$ with Δ^i indicating i th differences.

Measuring Predictive Accuracy

- Measure the accuracy of the h-step-ahead point forecasts for model i and variable j, relative to that from the univariate AR(1), using relative Root MSFEs:

$$RMSFE_{ijh} = \sqrt{\frac{\sum_{\tau=\underline{t}}^{\overline{t}-h} e_{i,j,\tau+h}^2}{\sum_{\tau=\underline{t}}^{\overline{t}-h} e_{bcmk,j,\tau+h}^2}},$$

  where $\underline{t}$ and $\overline{t}$ denote the start and end of the out-of-sample period and model $i \in \{\text{BVAR v1}, \text{BVAR v2}, \text{Factor model}\} \cup \text{TSLMs}$ and the AR(1) model

- For BVARs, point forecasts is computed as the median of predictive densities

- Focus on point forecast accuracy due to complexity in constructing density forecasts with TSLMs

TSLMs Performance Comparison

Distribution of RMSFEs

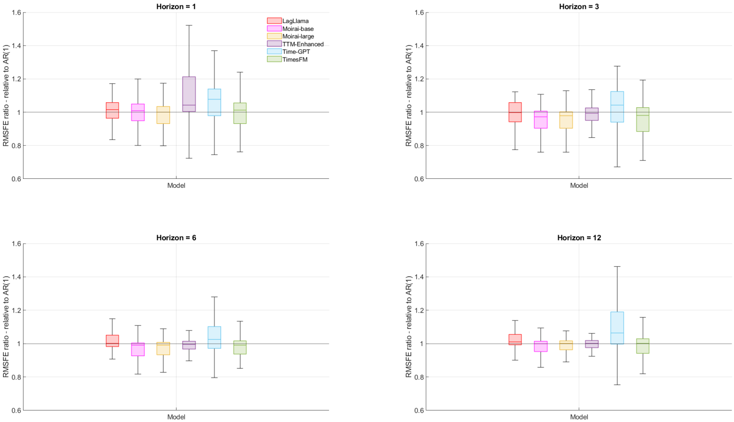


Figure: Distribution of RMSFEs (relative to AR benchmark) for TSLMs

TSLMs Detailed Performance Statistics

RMSFE by model type and forecast horizon

	h=1				h=3			
	Median	**Std**	**Min**	**Max**	**Median**	**Std**	**Min**	**Max**
LagLlama	1.015	1.057	0.726	7.271	0.997	0.787	0.633	4.843
Moirai-base	1.008	0.097	0.704	1.204	0.973	0.100	0.634	1.107
Moirai-large	0.999	0.102	0.703	1.436	0.978	0.099	0.637	1.158
TimesFM	1.014	0.129	0.706	1.482	0.980	0.127	0.635	1.318
TTM-Enhanced	1.044	0.352	0.723	2.959	0.993	0.108	0.718	1.448
Time-GPT	1.077	0.124	0.745	1.531	1.044	0.134	0.672	1.278
	h=6				**h=12**			
	Median	**Std**	**Min**	**Max**	**Median**	**Std**	**Min**	**Max**
LagLlama	1.002	0.461	0.568	3.577	1.009	0.260	0.597	2.431
Moirai-base	0.990	0.093	0.567	1.109	0.999	0.098	0.594	1.168
Moirai-large	0.991	0.096	0.600	1.159	1.001	0.113	0.619	1.324
TimesFM	0.990	0.142	0.593	1.629	1.001	0.158	0.482	1.440
TTM-Enhanced	0.995	0.097	0.643	1.400	1.002	0.198	0.663	2.257
Time-GPT	1.025	0.140	0.600	1.363	1.063	0.169	0.611	1.515

Table: Selected RMSFE statistics by TSLM model type and forecast horizon

Key Takeaways

- Heterogeneity: Significant performance differences among TSLMs
- Top performers: Moirai Large and TimesFM
- Underperformers: TTM-enhanced and Time-GPT
- Outliers: TSLMs can produce extremely inaccurate forecasts in some cases
- Recommendation: Use TSLMs with caution and monitor results carefully

Comparing TSLMs and Econometric Models

Distribution of RMSFEs

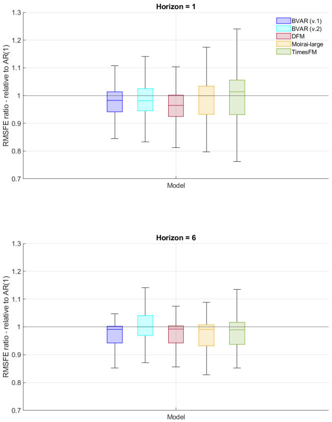


Figure: Distribution of RMSFEs for econometric models and best TSLMs

Comparing TSLMs and Econometric Models

RMSFE by model type and forecast horizon

	h=1				h=3			
	Median	**Std**	**Min**	**Max**	**Median**	**Std**	**Min**	**Max**
BVAR (v.1)	0.983	0.060	0.827	1.158	0.970	0.060	0.767	1.089
BVAR (v.2)	0.982	0.081	0.800	1.315	0.992	0.099	0.732	1.243
Factor model	0.965	0.065	0.766	1.103	0.980	0.102	0.682	1.183
Moirai-large	0.999	0.102	0.703	1.436	0.978	0.099	0.637	1.158
TimesFM	1.014	0.129	0.706	1.482	0.980	0.127	0.635	1.318
	h=6				**h=12**			
	Median	**Std**	**Min**	**Max**	**Median**	**Std**	**Min**	**Max**
BVAR (v.1)	0.991	0.068	0.725	1.106	0.999	0.080	0.671	1.162
BVAR (v.2)	1.000	0.095	0.714	1.222	1.004	0.111	0.611	1.343
Factor model	0.992	0.087	0.602	1.105	0.999	0.089	0.633	1.125
Moirai-large	0.991	0.096	0.600	1.159	1.001	0.113	0.619	1.324
TimesFM	0.990	0.142	0.593	1.629	1.001	0.158	0.482	1.440

Table: Selected RMSFE statistics for econometric models and best performing TSLMs

Key Takeaways

- Both TSLMs and econometric models generally outperform AR benchmark
- Econometric models offer more stable and reliable performance
- TSLMs show higher likelihood of large forecast errors
- BVARs and factor models: left-skewed RMSFE distribution
- TSLMs: more pronounced right tail in RMSFE distribution

Statistical Significance of Forecast Differences

Overview

- Focus: 19 variables from Medium model
- Display *RMSFE* ratios relative to AR(1) benchmark
- Include Diebold-Mariano (DM) t-statistic
    - Serial correlation robust std. errors
    - Harvey et al. (1997) small-sample adjustment
- Evaluation period: January 1985 to December 2019

RMSFE ratios, $h = 1$

	$h=1$									
	BVAR(v.1)		BVAR(v.2)		Factor model		Moirai Large		TimesFM	
PAYEMS	0.84	***	0.83	***	0.87	***	0.83	***	0.82	***
INDPRO	0.90	***	0.93	**	0.97		0.95	**	0.94	**
FEDFUNDS	1.21		0.97		0.87	**	1.00		1.23	
UNRATE	0.88	***	0.84	***	0.89	***	1.04		1.11	
RPI	0.98		0.98		0.99		1.00		1.04	
DPCERA3M086SBEA	1.00		0.99		0.98	*	1.02		1.06	
CMRMTSPLx	0.96	*	0.95	**	0.98		1.01		1.04	
CUMFNS	0.90	*	0.91	**	0.95		1.09		1.48	
CES0600000007	0.90	***	0.91	***	0.91	**	0.93	*	0.94	**
HOUST	0.95	***	0.95	**	0.93	***	1.00		0.92	***
S&P 500	1.04		1.03		1.00		1.03		1.02	
T1YFFM	1.19		1.18		1.06		1.04		1.13	
T10YFFM	1.06		1.00		1.00		1.04		1.16	
BAAFFM	1.02		0.96		0.94		1.01		1.28	
EXUSUKx	1.02		1.03		1.01		1.02		1.01	
WPSFD49207	0.97		1.00		1.03		1.01		1.01	
PPICMM	1.00		1.01		0.99		1.03		1.04	
PCEPI	0.94	***	0.94	**	0.98		0.99		0.95	*
CES0600000008	0.85	***	0.85	***	0.78	***	0.77	***	0.78	***

Table: Differences in accuracy that are statistically significant at 10%, 5%, and 1% levels are denoted by one, two, or three stars, respectively.

RMSFE ratios, $h = 12$

	$h = 12$									
	BVAR(v.1)		BVAR(v.2)		Factor model		Moirai Large		TimesFM	
PAYEMS	1.00		0.99		1.00		1.00		0.99	
INDPRO	1.01		1.01		1.01		1.01		0.99	
FEDFUNDS	0.96		1.01		0.96		1.00		1.08	
UNRATE	0.89	***	0.88	***	0.86	**	0.96		0.89	*
RPI	1.00		1.00		1.00		1.00		1.00	
DPCERA3M086SBEA	1.00		1.00		1.00		1.00		1.00	
CMRMTSPLx	1.01		1.01		1.01		1.02		1.02	
CUMFNS	0.91		0.93		1.06		0.92		1.02	
CES0600000007	0.70	***	0.75	***	0.81	***	0.76	***	0.78	***
HOUST	1.03		1.11		1.02		0.96		0.48	**
S&P 500	1.01		1.01		1.00		1.01		1.05	
T1YFFM	1.25		1.50		1.01		1.11		1.25	
T10YFFM	0.87		1.05		0.90	**	1.03		1.15	
BAAFFM	0.94		1.01		0.96		0.96		1.22	
EXUSUKx	1.00		1.01		1.00		1.03		1.00	
WPSFD49207	1.02		1.05		1.03		1.00		1.03	
PPICMM	1.00		1.00		1.01		1.01		1.03	
PCEPI	0.90	*	0.97		0.90	*	0.82	***	0.83	***
CES0600000008	0.88	***	0.88	***	0.81	***	0.78	***	0.80	***

Table: Differences in accuracy that are statistically significant at 10%, 5%, and 1% levels are denoted by one, two, or three stars, respectively.

Are these differences stable over time?

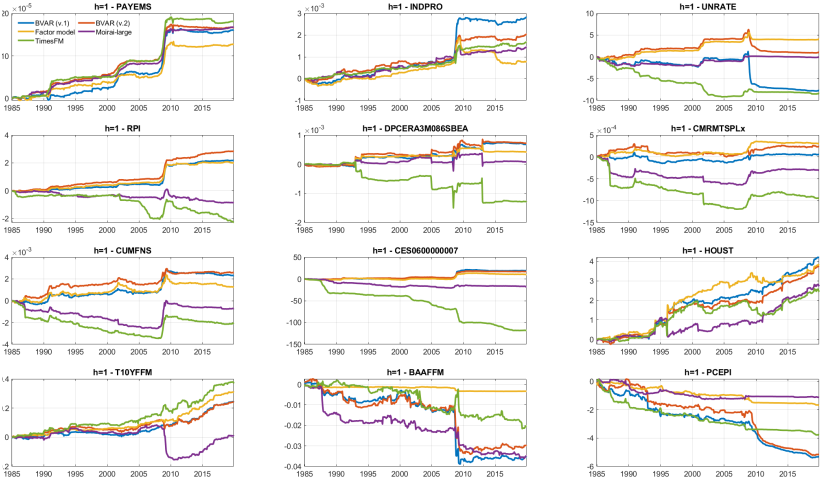


Figure: Cumulative sum of squared error differences for selected variables, $h = 1$. The evaluation sample is Jan 1985 to December 2019.

Conclusions

- We have presented a comparative analysis evaluating the forecasting accuracy of Time Series Language Models against state of the art econometric models using all 120+ series from FRED-MD
- Forecasting performance of a few TSLM (Moirai, TimesFM) is comparable to that of the BVARs and factor models
- BVARs and factor models appear to be generally more reliable, more robust to structural changes and less prone to generate unreasonable forecasts
- Lots of room for further explorations
  - Pre-trained vs. fine-tuned TSLMs
  - Expand to more models (e.g. Amazon Chronos)
  - Look at density forecasts

References I

Ansari, A. F., L. Stella, C. Turkmen, X. Zhang, P. Mercado, H. Shen, O. Shchur, S. S. Rangapuram, S. P. Arango, S. Kapoor, et al. (2024). Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*.

Bybee, L. (2023). Surveying generative ai's economic expectations. *arXiv preprint arXiv:2305.02823*.

Cao, D., F. Jia, S. O. Arik, T. Pfister, Y. Zheng, W. Ye, and Y. Liu (2024). TEMPO: Prompt-based generative pre-trained transformer for time series forecasting. *arXiv preprint arXiv:2310.04948*.

Chan, J. C. C. (2022). Asymmetric conjugate priors for large bayesian vars. *Quantitative Economics 13*(3), 1145–1169.

Chang, C., W.-Y. Wang, W.-C. Peng, and T.-F. Chen (2024). LLM4TS: Aligning pre-trained llms as data-efficient time-series forecasters. *arXiv preprint arXiv:2308.08469*.

Chen, Y., B. T. Kelly, and D. Xiu (2022). Expected returns and large language models. Available at SSRN: https://ssrn.com/abstract=4416687.

References II

Das, A., W. Kong, R. Sen, and Y. Zhou (2024). A decoder-only foundation model for time-series forecasting. In *International Conference on Machine Learning*. PMLR.

Ekambaram, V., A. Jati, N. H. Nguyen, P. Dayama, C. Reddy, W. M. Gifford, and J. Kalagnanam (2024). Ttms: Fast multi-level tiny time mixers for improved zero-shot and few-shot forecasting of multivariate time series. *arXiv preprint arXiv:2401.03955*.

Faria-e Castro, M. and F. Leibovici (2024). Artificial intelligence and inflation forecasts. Federal Reserve Bank of St. Louis Working Paper Series.

Garza, A. and M. Mergenthaler-Canseco (2023). TimeGPT-1. *arXiv preprint arXiv:2310.0358*.

Gruver, N., M. Finzi, S. Qiu, and A. G. Wilson (2024). Large language models are zero-shot time series forecasters. *arXiv preprint arXiv:2310.07820*.

Harvey, D., S. Leybourne, and P. Newbold (1997). Testing the equality of prediction mean squared errors. *International Journal of Forecasting 13*(2), 281 – 291.

References III

Jin, M., S. Wang, L. Ma, Z. Chu, J. Y. Zhang, X. Shi, P.-Y. Chen, Y. Liang, Y.-F. Li, S. Pan, and Q. Wen (2024). Time-LLM: Time series forecasting by reprogramming large language models. In *International Conference on Learning Representations (ICLR)*.

Jin, M., Y. Zhang, W. Chen, K. Zhang, Y. Liang, B. Yang, J. Wang, S. Pan, and Q. Wen (2024). Position: What can large language models tell us about time series analysis. *arXiv preprint arXiv:2402.02713*.

Kim, A. G., M. Muhn, and V. V. Nikolaev (2024). Financial statement analysis with large language models. Available at SSRN: https://ssrn.com/abstract=4835311.

Liu, P., H. Guo, T. Dai, N. Li, J. Bao, X. Ren, Y. Jiang, and S.-T. Xia (2024). Calf: Aligning llms for time series forecasting via cross-modal fine-tuning. *arXiv preprint arXiv:2403.07300*.

References IV

Rasul, K., A. Ashok, A. R. Williams, H. Ghonia, R. Bhagwatkar, A. Khorasani, M. J. D. Bayazi, G. Adamopoulos, R. Riachi, N. Hassen, M. Biloš, S. Garg, A. Schneider, N. Chapados, A. Drouin, V. Zantedeschi, Y. Nevmyvaka, and I. Rish (2024). Lag-llama: Towards foundation models for probabilistic time series forecasting.

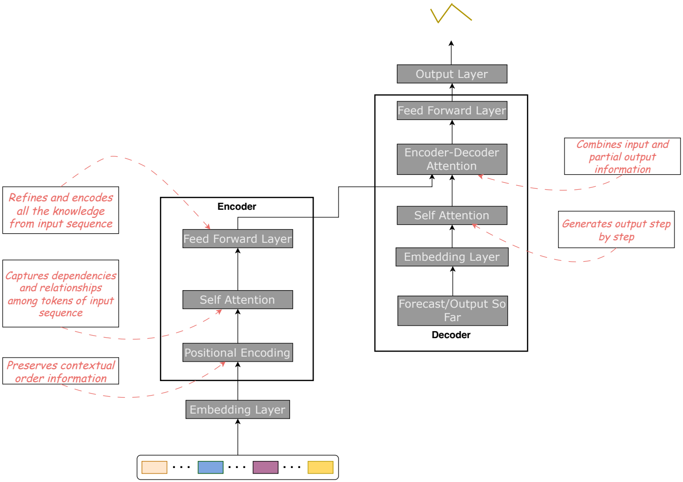
Stock, J. H. and M. W. Watson (2002, April). Macroeconomic Forecasting Using Diffusion Indexes. *Journal of Business & Economic Statistics 20*(2), 147–162.

Sun, C., H. Li, Y. Li, and S. Hong (2024). TEST: Text prototype aligned embedding to activate llm's ability for time series. *arXiv preprint arXiv:2308.08241*.

Woo, G., C. Liu, A. Kumar, C. Xiong, S. Savarese, and D. Sahoo (2024). Unified training of universal time series forecasting transformers. *arXiv preprint arXiv:2402.02592*.

Zhou, T., P. Niu, X. Wang, L. Sun, and R. Jin (2023). One fits all:power general time series analysis by pretrained lm. *arXiv preprint arXiv:2302.11939*.

Model Architecture



back