Critical Evaluation of Time Series Foundation Models in Demand Forecasting

Santosh Kumar Puvvada*
Data Scientist -NTT DATA
Bengaluru, India 560075
santosh.puvvada@nttdata.com

Satyajit Chaudhuri
Data Scientist -NTT DATA
Bengaluru, India 560075
satyajit.chaudhuri@nttdata.com

Abstract

Accurate forecasts are crucial as they enable organizations to make informed decisions about their supply chain. This research aims to benchmark and evaluate the efficiency of various foundation models in time series forecasting especially in the domain of demand forecasting. This research took two demand datasets from recent forecasting competitions and has used traditional statistical, machine learning and deep learning algorithms to forecast demand and compared their forecasting performance with popular foundational models TimeGPT and TimesFM. The evaluation considers both uncertainty and accuracy to establish a credible framework for comparison and benchmarking. This study has shown that TimesFM emerged as the better performing model across MASE & SMAPE and daily, weekly and monthly time granularities. The performance of the foundational models were at par with other traditional models and presented a strong case for wider research and adoption in industrial demand forecasting. Code and data used in the study is available at https://anonymous.4open.science/r/Critical_Evaluation-of_Foundational_Models_in_Demand_Forecasting-BB71/

1 Introduction

Demand forecasting is a critical element of Strategic Planning and Supply Chain Optimization [1]. Thus, improving the forecast accuracy is a significant area of interest for supply chain practitioners and has garnered huge attention in research, particularly among data scientists. Traditionally, researchers relied on Statistical Forecasting (SF), Machine Learning(ML) and Deep Learning(DL) methods to solve this problem. However, success of Foundation Models(FMs) in the field of Natural Language Processing(NLP) in recent years through BERT, GPT etc. has given a new way to look at this age old problem of forecasting. Time Series FMs provide an innovative approach for various tasks such as forecasting, classification, anomaly detection and imputations. This research has used FMs to forecast demand and has compared their performance with SF, ML and DL algorithms.

2 Literature Review

2.1 Traditional approaches for Time Series Forecasting

Time series is a sequence of data points collected or recorded at successive points in time, usually at uniform intervals and many SF, ML, DL and now FM based approaches have been used to forecast them. Statistical algorithms play a vital role in time series forecasting due to their ability to capture underlying patterns such as trends, seasonality and correlations over time. But ML based

*Corresponding author

NeurIPS 2024 Workshop on Time Series in the Age of Large Models.

regression models centered around the tree-based algorithms [2] have gained popularity in last few years. Bagging models like Random Forest (RF) [3] and Boosting models like Gradient Boosting Machine (GBM) [4, 5] have achieved great success and in few studies ML models were found to be more suitable for large scale demand forecasting scenario [6]and hence have been used by the authors in this study as well. Deep Neural Networks (DNNs) have increasingly been employed in multi-horizon forecasting, showing significant performance enhancements compared to traditional time series models [7,8]. Models like N-HiTS [9] are more adapted for time series forecasting due to its multi-rate sampling and hierarchical interpolation, enhancing accuracy, computational efficiency, and the ability to model long-range dependencies effectively. The Temporal Fusion Transformer (TFT) [10] is an attention-based deep learning model for multi-horizon forecasting, handling static covariates, dynamic inputs, and generating interpretable, high-performance predictions across diverse time series datasets.

2.2 Role of Foundational Models in Time Series Forecasting

With the advent of FMs for regression and classification tasks, many models have taken a center stage in forecasting domain also. This include Moirai [11], Time-LLM [12], LLM4TS [13], GPT2 [14], UniTime [15], Lag-Llama [16], TimeGPT [17], Moment [18] and TimesFM [19],TinyTimeMixer [20] and Chronos [21]. All the pre-trained time series models follow different architectures, training methodology and model sizes. TimeGPT and TimesFM are time series based foundation models i.e, they are trained from scratch on time series data. However, Chronos and Moment adapt language model architecture on time series data. With regards to architectures used, TimeGPT and Chronos follow Transformer based encoder-decoder architecture, TimesFM follows Transformer based Decoder only architecture and Moment follows Transformer based Encoder only architecture. In transformer dominated pre-trained models, TinyTimeMixer(TTM) follows an MLP based architecture. However FM implementation comes with a few caveats. TimeGPT requires atleast 36 data points for finetuning on monthly level. Models like Chronos and TimesFM do not support calibration for prediction intervals. Table 1 presents a comparison of some popular FMs.

Table 1: Comparison of various popular pre-trained models

Model parameter	TimeGPT	TimesFM	Chronos	Moment	TinyTimeMixer
Parameter size	Not specified	200M	20M to 710M	346M	<= 1 M
Time Frequency	All	All	All	All	Minutely to hourly
Exogenous Variables	Yes	Yes	No	No	Yes
Probabilistic forecasting	Yes	Yes	Yes	No	No

2.3 Limitations in the current benchmarking exercise and contribution of this research

The FMs have shown some promising results but some limitations were noted in the benchmarking process. Performance of FMs are not benchmarked against well-established machine learning models but only against deep learning and other pre trained models [18,20]. Although DL models have a shown a great potential, this is not a recommended approach especially when it has been conclusively proven in M5 forecasting competition that ML models were performing better than DL models [22]. Performance of pretrained models when benchmarked against other FMs, there is limited reasoning as why only those pretrained models were chosen for comparison and others were not [19]. Also, performance of pre-trained models is mostly measured in terms of accuracy while completely ignoring uncertainty [19]. Uncertainty quantification is extremely crucial in decision making, resource allocation and risk management. Easy availability of code and test data to reproduce the results and extend the testing to multiple datasets is not available for most models.

This study aims to address some of these concerns and develop a unified framework for evaluating the foundational models in demand forecasting. This will not only help in benchmarking and research of pretrained models but also time series forecasting in general. As it is tough to benchmark all the pretrained models, this study analyzes two highly popular foundational models, TimeGPT and TimesFM. TimeGPT is the first foundation model specifically designed for time series forecasting and TimesFM is the most liked FM on huggingface and hence they were selected as the worthy representatives of Time Series FMs for this study.

3 Research Methodology

3.1 Dataset

The data used for the study is taken from **Rohlik Orders Forecasting Challenge** [23] for daily time granularity. Rohlik is a leading European e-grocery innovator that is revolutionizing the food retail industry. The study has used the data from 4 warehouses namely Prague_1, Prague_2, Prague_3 and Brno_1. For weekly time granularity, a subset of the original data with 5800 unique ids were selected from **VN1 Forecasting - Accuracy Challenge** dataset [24]. The historical data had mature items with atleast 2 years history and was used to forecast next 13 weeks. The same data with 5800 combinations was then aggregated to a monthly level and was used to forecast next 3 months. Since these competitions were launched post the release of the FMs , it can be supposed that, none of the selected FMs would have been trained on these datasets, giving a fair and impartial way to judge the capability of all models in comparision.

3.2 Algorithms

The statistical forecasting has been carried out by using **AutoARIMA, AutoETS and AutoTBATS** from StatsForecast library. The study also uses Bagging methods like **RF** and Boosting algorithms like **Extreme Gradient Boosting (XGBoost)** and **Light Gradient Boosting (LGBM)** using the MLForecast library. Amongst the neural network architectures the study has chosen **TFT and NHITS** from NeuralForecast library [25] and compared with TimeGPT and TimesFM.

3.3 Evaluation Metrices

This study evaluated the performance of various algorithms from both accuracy and uncertainty perspective. To achieve that, the **Scaled Mean Absolute Percentage Error (SMAPE) and Mean Absolute Scaled Error (MASE)** [26] are used in this research. The scaled errors are independent of the scale of the data and so they can be used to compare the forecasts across data sets with different scales. In this study, we used a scaled version of **Continuous Ranked Probability Score(CRPS)** [27] to evaluate the probabilistic forecast. CRPS averages quantiles for each possible point between 0 & 1. CRPS is an excellent metric for measuring quality of probabilistic forecasts as it balances both sharpness of forecast distribution & coverage of observed values [28].

4 Results and Discussion

4.1 Results

In the daily datasets, the TimesFM model has outperformed the other traditional forecasting algorithms as seen in Table 2. This is closely followed by LGBM showing the relevance of machine learning algorithms in forecasting daily time series datasets. This study also found that the Zero shot and fine tuned TimeGPT models have lagged behind the other algorithms for this time granularity.

In this study, it was observed that TimesFM consistently outperformed other algorithms in both SMAPE and MASE on a weekly granularity, with TFT and NHITS DL algorithms following closely. TimeGPT zeroshot and finetuned also demonstrated superior performance compared to vanilla ML models and traditional statistical algorithms in terms of MASE.

On a monthly granularity, TimesFM excelled in MASE and SMAPE metrics, with DL algorithms performing well. However, the finetuning of TimeGPT was not feasible due to the requirement of at least 36 months of data. The CRPS scores are detailed in Table 3. Although TimesFM showed superior accuracy, its prediction intervals were not well-calibrated for uncertainty. Overall, FMs performed better than most ML and traditional models, advocating for further research. It is worth noting that ML models in this study were not hyperparameter-tuned, implying that with fine-tuning and additional feature engineering, they could potentially surpass FMs, as evidenced by top competition results.

Table 2: MAPE and SMAPE for the Algorithms across Daily, Weekly and Monthly Time granularities.

Granularity	**Daily**		**Weekly**		**Monthly**	
Metric	**MASE**	**SMAPE**	**MASE**	**SMAPE**	**MASE**	**SMAPE**
Arima	0.0981	11.93%	0.2912	16.25%	0.4359	17.13%
ETS	0.0969	11.58%	0.2582	15.42%	0.3765	17.23%
TBATS	0.0942	11.82%	0.2981	16.37%	0.6998	24.85%
RF	0.0961	11.71%	0.2796	15.34%	0.4687	17.10%
XGB	0.1010	12.04%	0.2702	15.22%	0.5761	18.61%
LGBM	0.0948	11.65%	0.2888	15.51%	0.5849	18.77%
TFT	0.0965	11.77%	0.2081	14.82%	0.3728	17.59%
NHITS	0.0989	12.03%	0.2082	15.50%	0.3692	17.46%
TimeGPT	0.1073	12.64%	0.2510	15.86%	0.3802	16.49%
TimeGPT Finetuned	0.1052	12.47%	0.2509	15.98%	-	-
TimesFM	**0.0933**	**11.50%**	**0.2043**	**14.69%**	**0.3465**	**16.49%**

Table 3: CRPS Score for the Algorithms across Daily, Weekly and Monthly Time granularities.

Granularity	**Daily**	**Weekly**	**Monthly**
Arima	0.0480	0.4078	0.5168
ETS	**0.0469**	0.3980	0.5034
TBATS	0.0583	0.4123	0.4531
RF	0.0905	0.5342	0.4209
XGB	0.0825	0.4903	0.5667
LGBM	0.0641	0.5287	0.4248
TFT	0.0553	0.3910	0.3480
NHITS	0.0714	**0.3856**	**0.3337**
TimeGPT	0.1306	0.8358	0.7862
TimeGPT Finetuned	0.0963	0.7982	-
TimesFM	0.0520	0.4920	0.4210

4.2 Key areas for further work

Going forward, more FMs shall be evaluated, fine-tuned and their performances benchmarked across datasets from different domains that account for all types of time series patterns like seasonality, cyclicity, intermittency etc. Even in demand forecasting, the nature of datasets vary across domains. The patterns in a retail industry data can be very different from a manufacturing industry dataset. Thus the models need to be tested across various domains so as to establish a robust benchmark. Understanding where FMs are working and where they are not will also provide direction for research in foundational model development. Creating ensembles with pretrained models alongside already established models should be explored, as ensembles work better when combining models of diverse nature.

5 Conclusion

This research has used demand forecasting datasets from forecasting competitions to establish a comparative study between the performances of Statistical, ML, DL and FMs across daily, weekly and monthly time horizons. To evaluate the performances of the algorithms, MASE & SMAPE were used as scaled errors are independent of the scale of the data. TimesFM emerged as the best performing algorithm across all time granularities. These were closely followed by the DL & vanilla ML models. TimeGPT has also outperformed the statistical and ML models across some time horizons. Overall, it can be concluded that the foundational models, although being very new members of a forecasters' toolkit, has shown impressive performance and can be used to establish a strong baseline for further research. The FMs can adapt to new data distributions with minimal tuning and do not require manual feature engineering and careful selection of lagged variables unlike ML regressors and thus allow the users to build and deploy forecasting solutions quickly and easily.

References

[1] Panda, S.K. and Mohanty, S.N., 2023. Time series forecasting and modeling of food demand supply chain based on regressors analysis. IEEE Access, 11, pp.42679-42700.

[2] James, G., Witten, D., Hastie, T. and Tibshirani, R., 2013. An introduction to statistical learning (Vol. 112, p. 18). New York: springer.

[3] Breiman, L., 2001. Random forests. Machine learning, 45, pp.5-32.

[4] Friedman, J.H. Greedy function approximation: A gradient boosting machine. Ann. Stat. 2001, 29, 1189–1232.

[5] Friedman, J.H., 2002. Stochastic gradient boosting. Computational statistics and data analysis, 38(4), pp.367-378.

[6] Huber, J. and Stuckenschmidt, H., 2020. Daily retail demand forecasting using machine learning with emphasis on calendric special days. International Journal of Forecasting, 36(4), pp.1420-1438.

[7] Rangapuram, S.S., Seeger, M.W., Gasthaus, J., Stella, L., Wang, Y. and Januschowski, T., 2018. Deep state space models for time series forecasting. Advances in neural information processing systems, 31.

[8] Alaa, A.M. and van der Schaar, M., 2019. Attentive state-space modeling of disease progression. Advances in neural information processing systems, 32.

[9] Challu, C., Olivares, K.G., Oreshkin, B.N., Ramirez, F.G., Canseco, M.M. and Dubrawski, A., 2023, June. Nhits: Neural hierarchical interpolation for time series forecasting. In Proceedings of the AAAI conference on artificial intelligence (Vol. 37, No. 6, pp. 6989-6997).

[10] Lim, B., Arık, S.Ö., Loeff, N. and Pfister, T., 2021. Temporal fusion transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting, 37(4), pp.1748-1764.

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

[12] Jin, M., Wang, S., Ma, L., Chu, Z., Zhang, J.Y., Shi, X., Chen, P.Y., Liang, Y., Li, Y.F., Pan, S. and Wen, Q., 2023. Time-llm: Time series forecasting by reprogramming large language models. arXiv preprint arXiv:2310.01728.

[13] Chang, C., Peng, W.C. and Chen, T.F., 2023. Llm4ts: Two-stage fine-tuning for time-series forecasting with pre-trained llms. arXiv preprint arXiv:2308.08469.

[14] Zhou, T., Niu, P., Sun, L. and Jin, R., 2023. One fits all: Power general time series analysis by pretrained lm. Advances in neural information processing systems, 36, pp.43322-43355.

[15] Liu, X., Hu, J., Li, Y., Diao, S., Liang, Y., Hooi, B. and Zimmermann, R., 2024, May. Unitime: A language-empowered unified model for cross-domain time series forecasting. In Proceedings of the ACM on Web Conference 2024 (pp. 4095-4106).

[16] Rasul, K., Ashok, A., Williams, A.R., Ghonia, H., Bhagwatkar, R., Khorasani, A., Bayazi, M.J.D., Adamopoulos, G., Riachi, R., Hassen, N. and Biloš, M., 2024. Lag-llama: Towards foundation models for probabilistic time series forecasting. Preprint.

[17] Garza, A. and Mergenthaler-Canseco, M., 2023. TimeGPT-1. arXiv preprint arXiv:2310.03589.

[18]Goswami, M., Szafer, K., Choudhry, A., Cai, Y., Li, S. and Dubrawski, A., 2024. Moment: A family of open time-series foundation models. arXiv preprint arXiv:2402.03885.

[19] Das, A., Kong, W., Sen, R. and Zhou, Y., 2023. A decoder-only foundation model for time-series forecasting. arXiv preprint arXiv:2310.10688.

[20] Ekambaram, V., Jati, A., Dayama, P., Mukherjee, S., Nguyen, N.H., Gifford, W.M., Reddy, C. and Kalagnanam, J., 2024. Tiny Time Mixers (TTMs): Fast Pre-trained Models for Enhanced Zero/Few-Shot Forecasting of Multivariate Time Series.CoRR.

[21] Ansari, A.F., Stella, L., Turkmen, C., Zhang, X., Mercado, P., Shen, H., Shchur, O., Rangapuram, S.S., Arango, S.P., Kapoor, S. and Zschiegner, J., 2024. Chronos: Learning the language of time series. arXiv preprint arXiv:2403.07815.

[22] Makridakis, S., Spiliotis, E. and Assimakopoulos, V., 2022. M5 accuracy competition: Results, findings, and conclusions.International Journal of Forecasting 38(4), pp.1346-1364.

[23] Kecera, M. (2024) Rohlik Orders Forecasting Challenge. Kaggle. Available at: https://kaggle.com/competitions/rohlik-orders-forecasting-challenge.

[24] Vandeput, Nicolas. "VN1 Forecasting - Accuracy Challenge." DataSource.ai, DataSource, 2024, https://www.datasource.ai/en/home/data-science-competitions-for-startups/phase-2-vn1-forecasting-accuracy-challenge/description

[25] https://nixtlaverse.nixtla.io/neuralforecast/docs/getting-started/introduction.html

[26] García-Aroca, C., Martínez-Mayoral, M.A., Morales-Socuéllamos, J. and Segura-Heras, J.V., 2024. An algorithm for automatic selection and combination of forecast models. Expert Systems with Applications, 237, p.121636.

[27] Gneiting, T. and Raftery, A.E., 2007. Strictly proper scoring rules, prediction, and estimation. Journal of the American statistical Association, 102(477), pp.359-378.

[28] Rangapuram, S. S., Werner, L. D., Benidis, K., Mercado, P., Gasthaus, J., and Januschowski, T. (2021, July). End-to-end learning of coherent probabilistic forecasts for hierarchical time series. In International Conference on Machine Learning (pp. 8832-8843). PMLR.