

# TimeGPT

Leverage Generative AI for Time Series Forecasting in R

Mariana Menchero, Senior Forecaster at Nixtla





# Introducing TimeGPT & nixtlar

## What is nixtlar?

nixtlar is an R package that brings the power of TimeGPT, the first foundation model for time series forecasting, to the R community.

## Why it Matters

With nixtlar, R users gain access to a state-of-the-art forecasting model - no additional training required.

## Get Started

nixtlar is available on CRAN, so it can be easily integrated into your forecasting workflow.



# Let's do a deep dive!



# What is TimeGPT?

## TimeGPT: A Foundation Model for Time Series Forecasting

- TimeGPT is a foundation model developed by Nixtla for time series forecasting and anomaly detection.
- Available in Python through *nixtla*

## Foundation Models Explained

- Foundation models like TimeGPT use *transfer learning*: they are trained on large, diverse datasets and can generalize to different contexts.
- This means they can forecast on unseen datasets without additional training.
- They represent a new paradigm for time series analysis and forecasting.



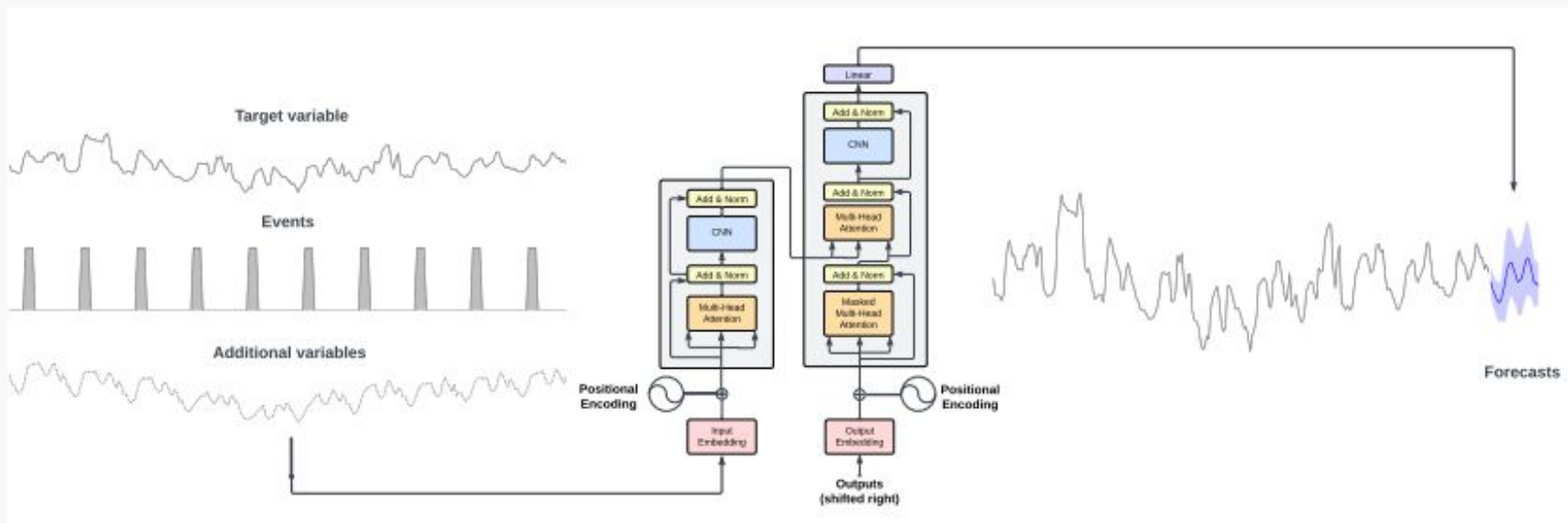
# How TimeGPT works 1/3

TimeGPT is a Transformer-based probabilistic model with self-attention mechanisms.

It takes a window of historical values, like tokens in text, and then predicts what comes next.

## How TimeGPT works 2/3

The TimeGPT architecture consists of an encoder-decoder structure with multiple layers, each with residual connections and layer normalization. A final linear layer maps the decoder's output to the forecasting window dimension.

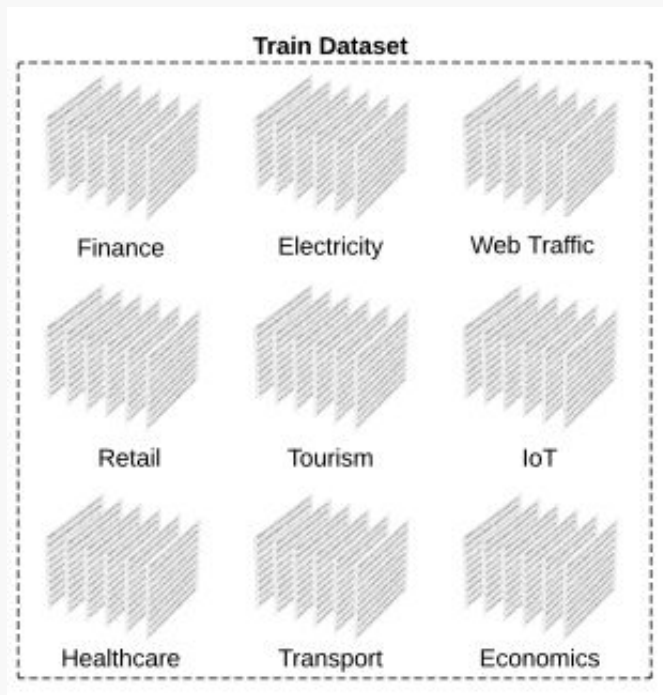


# How TimeGPT works 3/3

TimeGPT was trained on a huge collection of publicly available time series data, with over 100 billion observations, from various domains such as retail, finance, energy, demographics, and healthcare.

This dataset includes time series with different seasonalities and trends, as well as varying levels of noise and numbers of unusual observations or outliers.

This diversity helps TimeGPT handle a wide range of scenarios and forecast unseen time series accurately without additional training.





# Foundation Models Arena

TimeGPT was the first foundation model for time series forecasting, and after its release, other models followed, including:

- Lag-Llama
- TimesFM
- Tiny Time Mixers
- Moirai
- Chronos
- Moment

What do they all have in common? None are available in R...





# Enter nixtlar

Making TimeGPT easily accessible to the R community



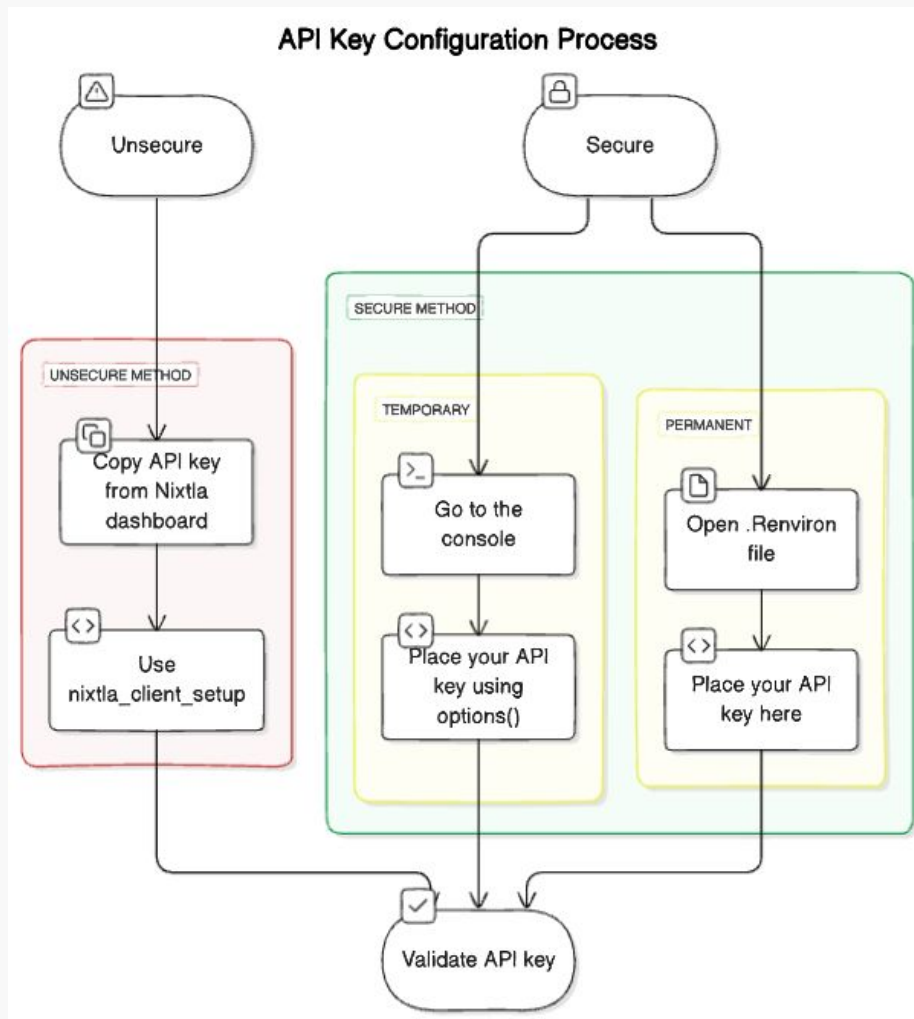
# nixtlar in Action - Retail Forecasting

Let's imagine you're the manager of a Corporación Favorita store, a grocery chain in Ecuador. You're responsible for forecasting the sales of all products at your store for the next two weeks. You have daily sales data for each product: for some, you have years' worth of data, while for others, only a few weeks.

To make things interesting, you need forecasts for all products by tomorrow to start ordering inventory.

As an R programmer, you decide to put nixtlar to the test. For comparison, you will also forecast sales using the ARIMA and ETS models from fable, along with Prophet. As a baseline, you will use a seasonal naive forecast.

# Setting Up Your API Key





# nixtlar in Action - Retail Forecasting

- Data: Daily sales for 3,633 products
- Forecast horizon: Next 14 days
- Models: Out-of-the-box implementations for
  - TimeGPT via nixtlar
  - ARIMA, ETS, and Seasonal Naive via fable
  - Prophet via fable.prophet

Model	Relative execution time (seconds)	MAE	MASE
Seasonal Naïve - baseline	1	6.59	1.03
ARIMA	293.45	5.56	<b>0.88</b>
ETS	69.52	5.72	0.89
Prophet	210.43	7.85	1.23
TimeGPT	0.74	5.55	0.99
TimeGPT (long horizon)	1.97	<b>5.25</b>	0.91

Table 1. Results for the Corporación Favorita Store



# Additional features and capabilities

nixtlar provides access to TimeGPT's features and capabilities, such as:

- **Fine-tuning:** Enhance TimeGPT's capabilities by fine-tuning the model on your specific datasets, enabling the model to adapt to the nuances of your unique time series data and improving performance on tailored tasks.
- **Add Exogenous Variables:** Incorporate additional variables that might influence your predictions to enhance forecast accuracy such as promotions or holidays.
- **Custom Loss Function:** Tailor the fine-tuning process with a custom loss function to meet specific performance metrics.
- **Prediction Intervals or quantiles:** Quantify the uncertainty of your predictions with prediction intervals or quantiles.
- **Azure Integration:** You can use TimeGEN-1, a version of TimeGPT optimized for Azure.



But wait, there's more!



# Anomaly Detection and Cross-Validation



# Conclusions

**nixtlar** brings TimeGPT, the first foundation model for time series forecasting, to the R community, making advanced forecasting capabilities accessible and user-friendly.

**Proven Performance:** TimeGPT has demonstrated that it can outperform well-known and widely used forecasting models in R, offering comparable accuracy in a fraction of the time.

**Enhanced Capabilities:** Additional features—such as fine-tuning, exogenous variables, and prediction intervals—are available to further refine and improve forecast accuracy.

Explore nixtlar and see how TimeGPT performs on your own dataset!





Read the documentation:  
<https://nixtla.github.io/nixtlar>

And please give us a star on GitHub! ★