AI for stock market prediction: Using LLMs for TimeSeries Predictions

Project by: Jana Nikolovska
Supervised by: Giacomo Frisoni, MSc
Prof. Gianluca Moro, PhD

ALMA MATER STUDITORIUM - University of Bologna

May, 2025

UNDERSTANDING THE PROBLEM: STOCK PREDICTION

Challenge: Predicting stock prices involves analyzing time-series data
Importance: Investment strategies and financial forecasting.
Idea: Use LLMs for forecasting by treating the task as a time-series problem, where the model predicts the next value based on historical data.

Why LLMs Struggle with Time-Series Prediction

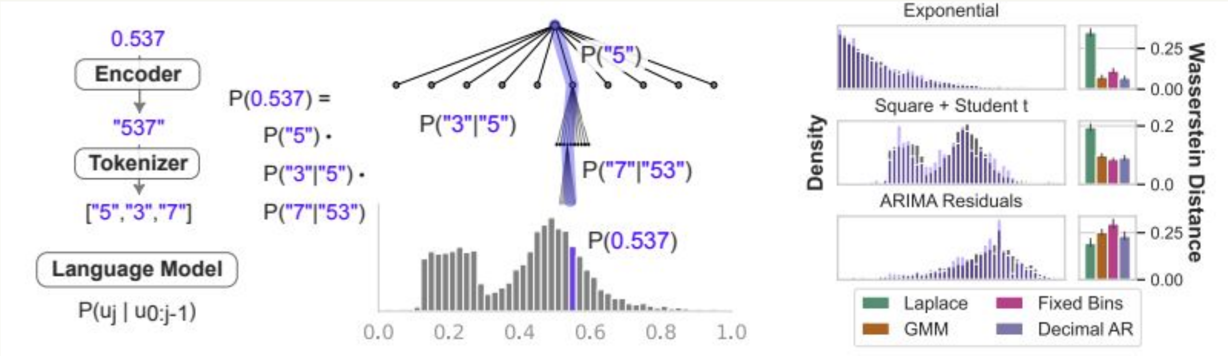
- LLMs are **optimized for sequential text, not numerical time-series data**.
- Time-series data has **dependencies across time**, which LLMs **struggle to capture** without explicit temporal modeling

Proposed Solution:

Gruver, N., LaRocca, J., & Yang, H. (2023). **LLMTime: Leveraging large language models for time series forecasting**. *arXiv preprint arXiv:2310.07820*. https://arxiv.org/pdf/2310.07820

The paper introduces methods for adapting LLMs to handle time-series data, addressing the temporal structure.

LLMTime: Leveraging large language models for time series forecasting



- treats time series forecasting as a sequence prediction task by encoding numerical data as strings of digits
  - encodes numbers as individual digits separated by spaces
  - to prevent large numbers from consuming excessive token space, values are rescaled so that a specific percentile of the data falls within a desired range

PIPELINE

Dataset

- **Load and filter data** for specific dates
- **Resample the data to a daily frequency**, to address the missing data caused by non-trading days (interpolation)
- **Analyze series from dataset**
- **Split dataset**

Running predictions

- Select models: **Linear Regression, ARIMA, GPT3, GPT4**
- **Autoregressive Prediction**
  - **Modifications for Linear Regression** autoregressive simulations
- Dealing with **Probabilistic Prediction** for ARIMA, GPT3, GPT4

Evaluation

- **MAPE**
- **Trading Protocol**
  - **Gain, ROI** (Return on Investment)
- **Visualization** of results
  - **Daily Gain, Cumulative Gain**
- **Averaging out across multiple test datasets**

Data Analysis

- Description of the data: **missing values**, **statistical metrics** (such as count, mean, standard deviation, min, max, and percentiles
- **Correlation matrix**
- Separate analysis of Open and Close series:
  - **Graph** visualization of the data
  - **Autocorrelation** and **Partial Autocorrelation**
  - **Anomaly detection**
  - Calculated **smoothed moving averages** to enhance trend visualization
  - **Histogram of daily returns**
  - **Rolling Volatility**
  - **Seasonal decomposition**

Autoregressive Models

- **ARIMA and GPT models are both autoregressive approaches**.
- To fairly compare traditional ML models with autoregressive models,a custom function that **imitates autoregressiveness** was implemented
- Unlike ARIMA and GPT, **traditional models are trained using a lagged dataset** — where each input consists of a fixed number of past values.
- **The number of lags is configurable**

Predictions Pipeline

- **Linear Regression**:
  - Create lagged dataset, as features with latest value as label (rolling window with fixed predefined value)
  - Fit model and predict
  - Do evaluations
- **Autoregressive models**:
  - Problem: for a sequence of values predict the next value
  - Repeat for both Open and Closed
  - Do evaluations
- **Run for multiple train/testset combination, average for evaluations**

Probabilistic Prediction

- Each model output gives **30 samples**, interpreted as a distribution (estimate a normal distribution)
- We use the **mean** of this distribution as the predicted value
- The **standard deviation** is used to measure **risk and uncertainty** in decisions.

Technologies and Libraries

Link to Github Public Repository

Thank You

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA