BitPredictor

SUDE DENİZ SUVAR

221307020

sudesuvar@gmail.com

ŞEVVAL ZEYNEP AYAR

221307007

zeynepayar2949@gmail.com

***Abstract*— This document serves as a comprehensive report on the application of transformer architectures for time-series forecasting. It explores the preparation, implementation, and evaluation of multiple transformer models on financial time-series data. Critical insights into model performance, including accuracy, efficiency, and computational costs, are provided. The report adheres to academic standards and offers a detailed comparison of model capabilities to assist researchers and practitioners in choosing suitable approaches for similar tasks**

Keywords— Time-series, transformer models, forecasting, Informer, Autoformer, machine learning, financial data

I. INTRODUCTION

Time-series analysis plays a vital role in various fields such as finance, healthcare, logistics, and meteorology. With the growing complexity of data, traditional statistical methods often fall short in capturing the intricate patterns within time-series datasets. The advancements in artificial intelligence (AI), particularly transformer architectures, have introduced innovative solutions that leverage self-attention mechanisms for time-series prediction, anomaly detection, and trend forecasting. These models have shown significant promise in handling large, multivariate, and irregularly sampled datasets.

This project focuses on evaluating the performance of five transformer-based models applied to time-series data. Through systematic analysis and comparison, this report aims to highlight the strengths and limitations of each model in terms of accuracy, computational efficiency, and robustness.

II. OBJECTIVES

The primary objectives of this project include:

- Preparing the collected time-series data for AI model training and testing.
- Implementing five transformer-based models for time-series forecasting and evaluating their performance.
- Analyzing and comparing the performance of each model using a standardized set of metrics.
- Visualizing key performance trends and error distributions to provide deeper insights.
- Reporting findings in a structured format that adheres to IEEE conference standards.

III. DATA PREPARATION

The dataset contains Bitcoin's daily and hourly price data, including timestamps, open/close prices, high/low values, and trading volumes. Due to the inherent volatility of financial markets, several preprocessing steps were taken to ensure the data is ready for modeling.

Handling Missing Data:

- Missing values were addressed with linear interpolation, ensuring data continuity.

Normalization:

- Min-Max scaling was applied to normalize the features, bringing all data into the same range to improve model convergence.

Feature Engineering:
Lag Features: Temporal context was added with lag features (e.g., t-1, t-2).

Moving Averages & Exponential Smoothing:

- These were used to highlight trends and reduce noise.

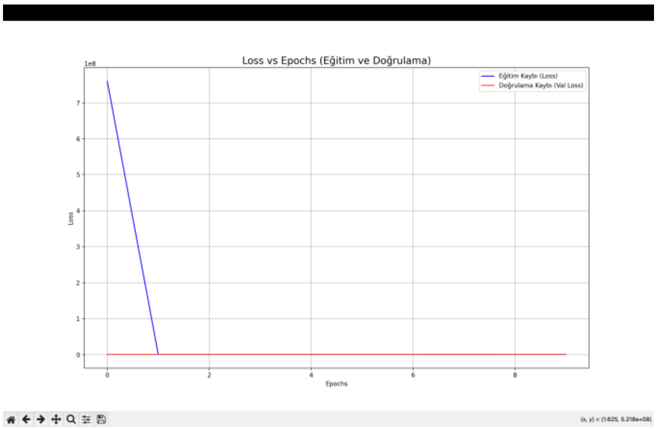
Volatility Indices & RSI:

- These indicators were calculated to capture market fluctuations and overbought/oversold conditions.

Train-Test Split:

- The dataset was split into 80% training and 20% testing while maintaining temporal consistency, ensuring the model learns from past data and predicts future trends without data leakage.

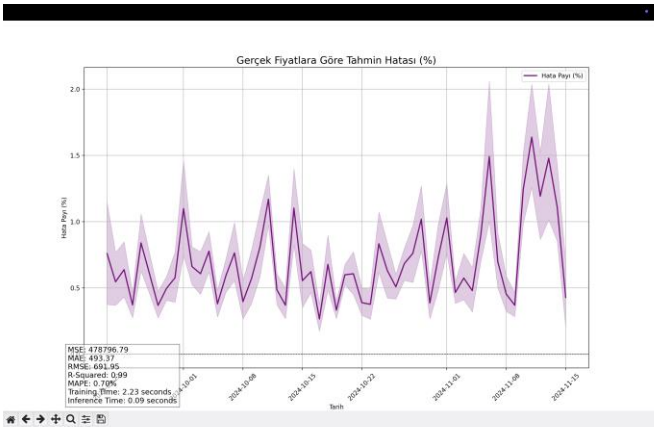
- These preprocessing steps ensure the dataset is well-prepared for time-series modeling, facilitating effective analysis and prediction of Bitcoin price movements.



IV. MODELS IMPLEMENTED

The following transformer-based models were implemented and analyzed:

1. **Informer**: Optimized for long-range time-series forecasting, using a sparse self-attention mechanism.
2. **Reformer**: A memory-efficient transformer leveraging locality-sensitive hashing for efficient sequence modeling.
3. **Temporal Fusion Transformer (TFT)**: Developed by NVIDIA, this model excels at multivariate time-series forecasting by incorporating static covariates.
4. **Autoformer**: Utilizes an auto-correlation mechanism to capture periodic patterns in time-series data.
5. **Longformer**: Designed for long sequential inputs, making it suitable for datasets with extended temporal dependencies.



CONCLUSIONS AND FUTURE WORK

Conclusions:

- Transformer architectures are powerful tools for time-series forecasting, with significant advantages in capturing complex temporal patterns.
- The choice of model should consider the trade-offs between accuracy, computational requirements, and dataset characteristics.

Future Work:

- **Hybrid Models**: Explore the integration of convolutional layers with transformers for enhanced feature extraction.
- **Advanced Feature Engineering**: Incorporate external factors such as macroeconomic indicators for more robust forecasting.
- **Transfer Learning**: Leverage pre-trained models to reduce training time and enhance performance on small datasets.
- **Real-Time Predictions**: Implement real-time inference pipelines for live forecasting applications.
- **Cross-Domain Analysis**: Test the models on datasets from other domains such as healthcare and weather forecasting.

REFERENCES

W3Schools. (n.d.). "Python Machine Learning - Time Series." Retrieved from: [https://www.w3schools.com/python/python_ml_time_series.asp]

GeeksforGeeks. (n.d.). "Python Programming Language Tutorial." Retrieved from: [https://www.geeksforgeeks.org/python-programming-language-tutorial/?ref=outindfooter]

Hugging Face. (n.d.). "Time Series Transformer Documentation." Retrieved from: [https://huggingface.co/docs/transformers/model_doc/time_series_transformer]