Group 19: Cryptocurrency Price Forecasting

Shriyan, Hugo, Marcus

Table of contents

A Digital Currency in Which Transactions Are Verified and Records Maintained By a Decentralized System Using Cryptography.

Machine Learning Problem Statement

Price Prediction Using Historical Correlations

01

Objective 1

Develop a Graph-Based Model to accurately predict the future prices of various cryptocurrencies

02

Objective 2

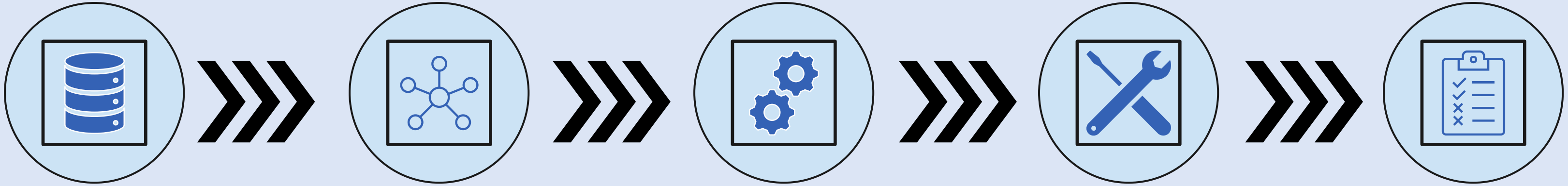
Use multiple features in the Graph-based model and add meaningful features to enhance performance

03

Objective 3

Implement baseline Graph models to compare our Model with.

Methodology Overview



Data Collection

Graph Construction

Feature Engineering

Model Implementation

Training and Evaluation

Data Processing

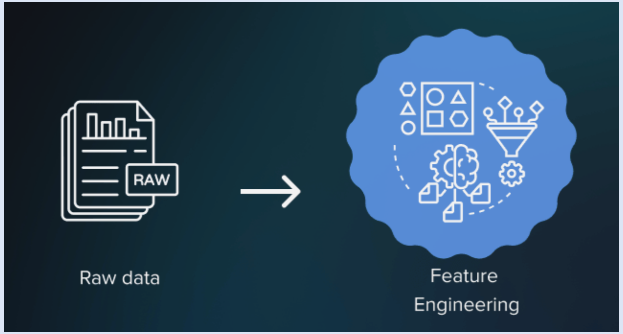
Binance API

fetch_data.py

Item Example

```
1499040000000,      // Kline open time
"0.01634790",       // Open price
"0.80000000",       // High price
"0.01575800",       // Low price
"0.01577100",       // Close price
"148976.11427815",  // Volume
"2434.19055334",    // Quote asset volume
308,                // Number of trades
```

data_loader.py



Add New Features

main.py

Select Parameters

Feature Engineering

Features Added:

- Volume Quote Ratio
- Buy Sell Volume Ratio
- Buy Sell Quote Ratio
- Log Returns
- Stochastic Oscillator indicating momentum.
- Volume-Weighted Average Price
- Parkinson's volatility
- Garman-Klass volatility
- Average size of Trades
- Average Trade Quote Size

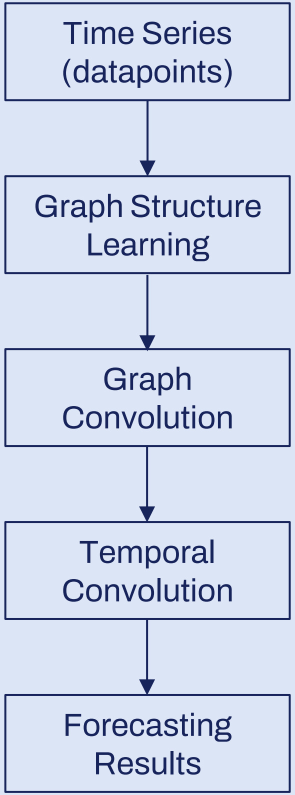
$$GKHV = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}\left(ln\frac{h_i}{l_i}\right)^2 - \frac{1}{N}\sum_{i=1}^{N}(2ln2-1)\left(ln\frac{c_i}{o_i}\right)^2}$$

$$ParkinsonHV = \sqrt{\frac{1}{4Nln2}\sum_{i=1}^{N}\left(ln\frac{h_i}{l_i}\right)^2}$$

These features include liquidity metrics, price movement indicators, momentum indicators, volatility estimates, and trade behavior insights – all factors that can influence future price of a currency

What is MTGNN?

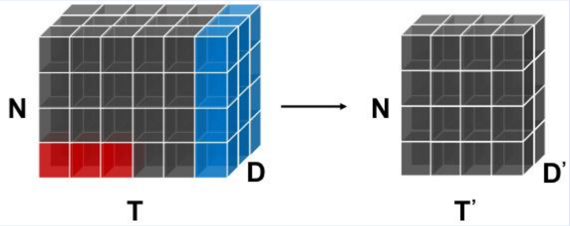
General Framework



Graph Learning

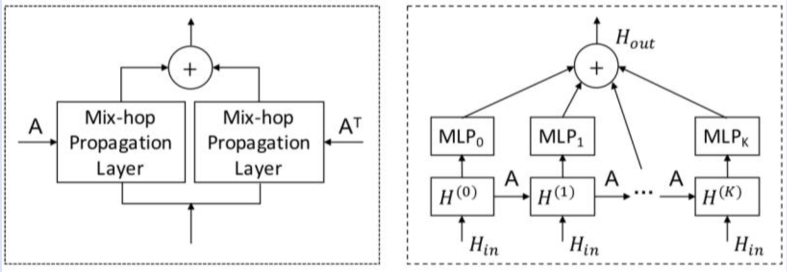
$$\mathbf{M}_1 = tanh(\alpha \mathbf{E}_1 \Theta_1)$$
$$\mathbf{M}_2 = tanh(\alpha \mathbf{E}_2 \Theta_2)$$
$$\mathbf{A} = ReLU(tanh(\alpha(\mathbf{M}_1 \mathbf{M}_2^T - \mathbf{M}_2 \mathbf{M}_1^T)))$$
$$for\ i = 1, 2, \cdots, N$$
$$\mathbf{idx} = argtopk(\mathbf{A}[i, :])$$
$$\mathbf{A}[i, -\mathbf{idx}] = 0,$$

How GC and TC works



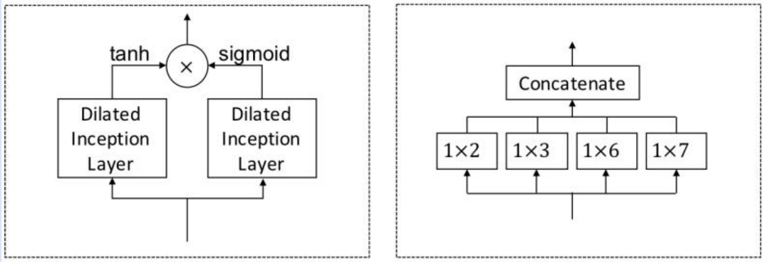
Red = TC filtering, Blue = GC filtering

Graph Convolution



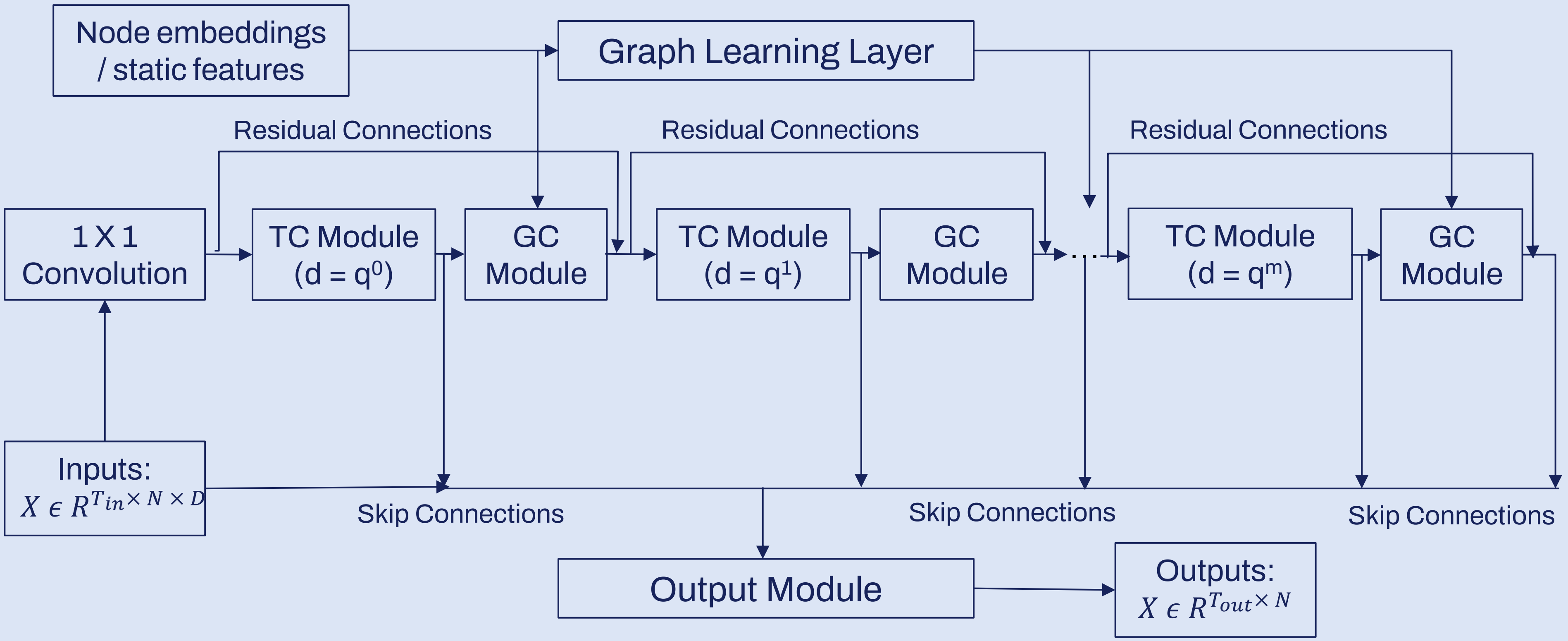
(a) GC module **(b) Mix-hop propagation layer**

Temporal Convolution



(a) TC module **(b) Dilated inception layer**

Model Architecture – Adapted from Paper



Our Contributions

Attention Layer

Noise Removal

Resolve Data Leakage

Multiple Versions
With Multiple Features

Baseline Models

Experiment Setup

- Learning Rate: 0.001
- Epochs: 16
- Optimizer: AdamW
- Batch Size: 16
- Ablation Study: Multiple features, Attention layer, Noise removal

Date Range: 2021/10/01 - 2024/10/01

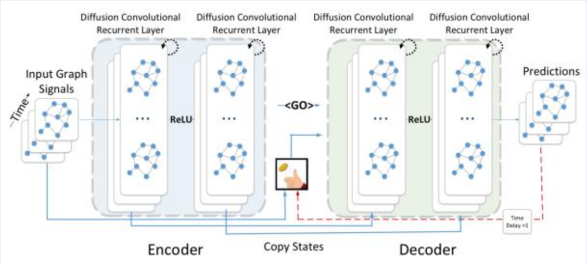
Timeframe of Training Set: 1 Hour (27019 Samples per Currency) and 1 Day (1108 Samples per Currency)

Train-Test Split: First 60% of the dataset for training, Next 20% for validation, last 20% for testing

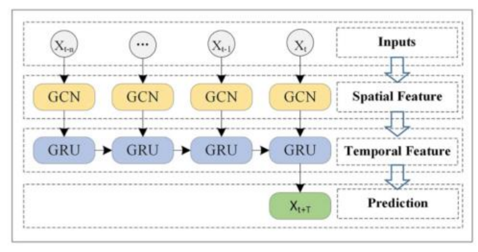
For Baseline Models, we initialized a Graph and Adjacency Matrix based on the correlation of currencies in 2024

Baseline Models

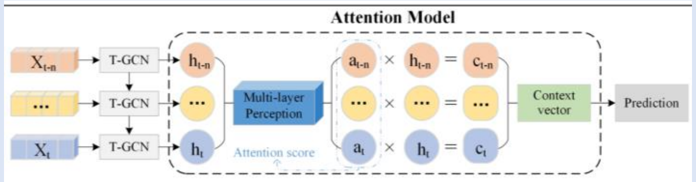
DCRNN (Diffusion Convolutional Recurrent Neural Network)



T-GCN (Temporal Graph Convolutional Network)



A3T-GCN (Attention 3 T-GCN)



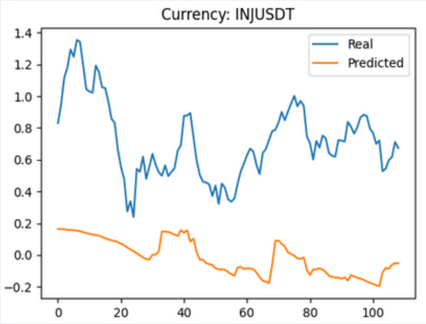
Results – Baseline Models

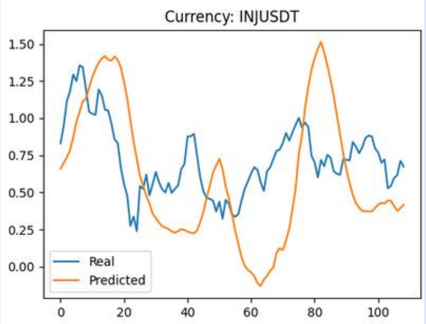
Run details	Test Error	Test RMSE	Test RAE	Test CORR
T-CGN_1d	1458.4494	0.5929	0.6219	0.7289
A3T-GCN_1d	884.310	0.3931	0.5392	0.8241
DCRNN_1d	2472.27	0.8964	0.8674	0.3595
T-CGN_1h	7871.092	**0.2928**	**0.1772**	0.9360
A3T-GCN_1h	8497.078	0.3042	0.2328	**0.9417**
DCRNN_1h	51904.086	0.7519	0.8107	0.5655

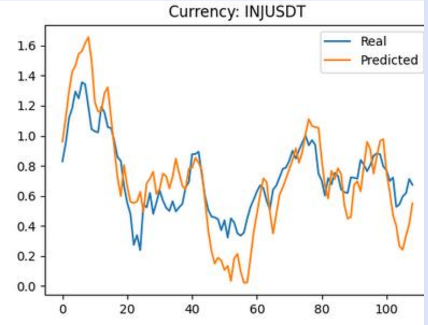
Results - Baseline Models Visualization

DCRNN | **A3T-GCN** | **T-GCN**

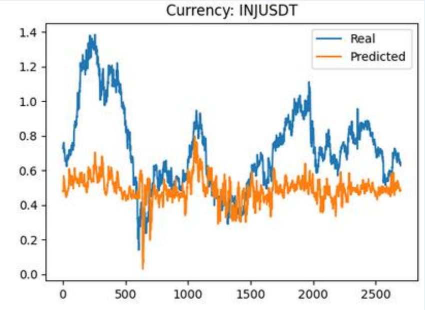
Daily

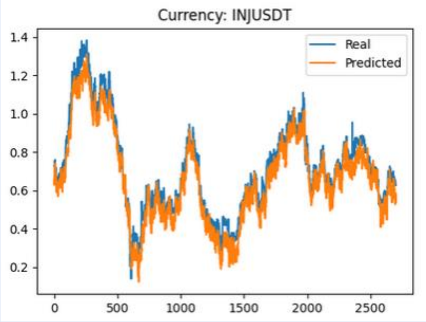


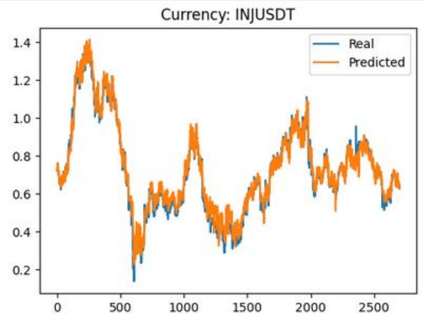




Hourly





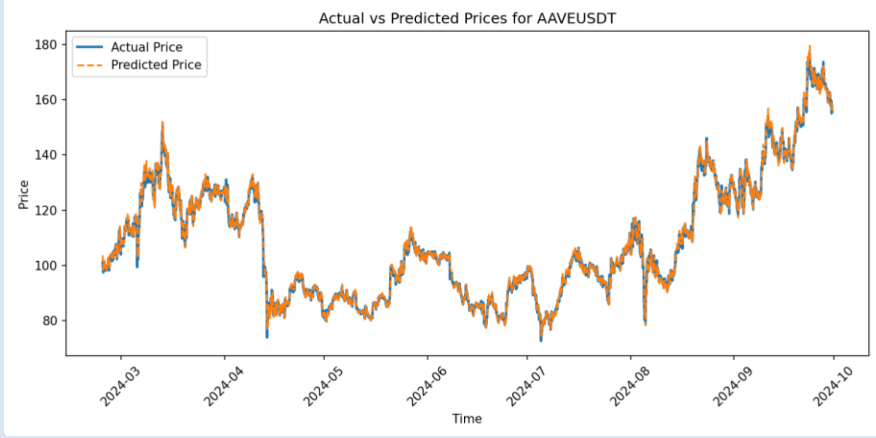


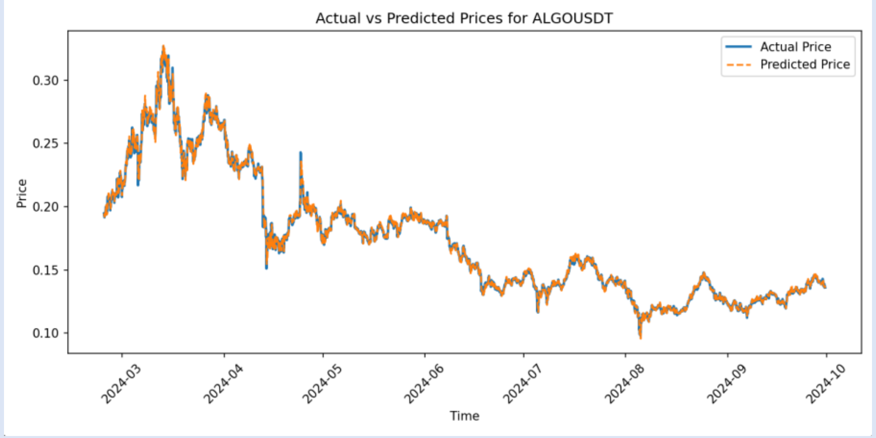
Results - MTGNN

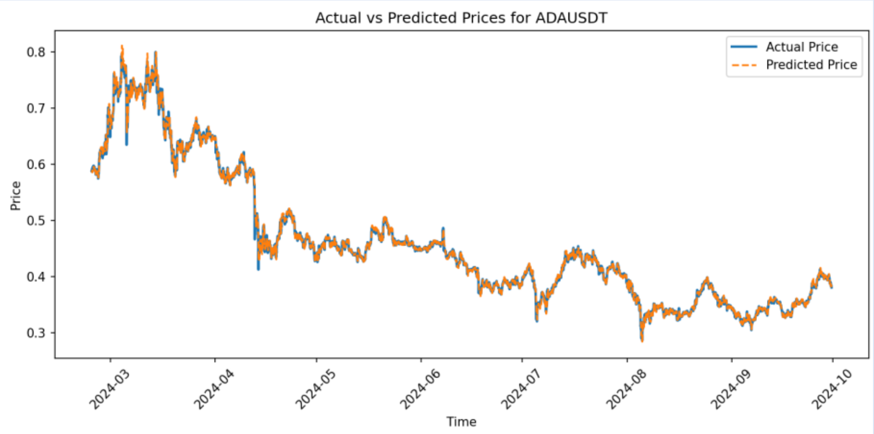
Run details	Test Error	Test RMSE	Test RAE	Test CORR
old_all_features_1h	686.432431	0.00242951	0.00021545	0.9950386
new_all_features_1h_attention	872.8712	0.00273965	0.00024256	0.99425787
new_all_features_1h	1293.12702	0.00333457	0.00029693	0.9919641
old_one_feature_1h	5966.82006	0.00716503	0.00288709	0.9979799
new_one_feature_1h	18007.9753	0.0124474	0.00514457	0.9946672
new_one_feature_1h_attention	20948.873	0.01342145	0.00552297	0.99392205
new_one_feature_1h_ma_attention	69975.6649	0.02452973	0.00665949	0.9873395
old_one_feature_1h_ma	73606.7123	0.02515811	0.00682768	0.9869195
new_one_feature_1h_ma	76468.436	0.0256425	0.00692622	0.9862451
old_one_feature_1d	1322001.6	0.10677032	0.04745535	0.7394306
new_one_feature_1d	1344669.62	0.10768181	0.05016395	0.667452

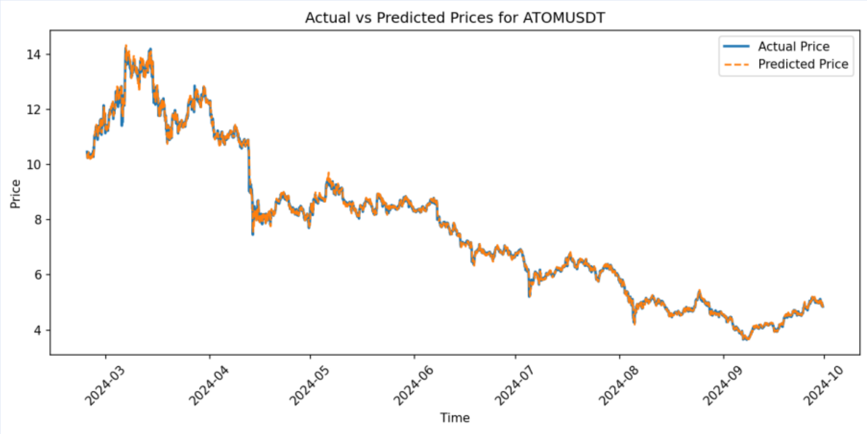
old = Original implementation, new = pytorch_geometric implementation, one_feature = Include close prices only

Results – Best MTGNN Visualization









Insights - Compare

- MTGNN has comparable performance on daily dataset without Adjacency Matrix to baseline models, meaning it was able to learn the correlation and dependencies between cryptocurrencies on its own.

- Performance is consistent across various features when using hourly data. Dataset size and shorter timeframe is the most important factor in determining prediction accuracy.

- MTGNN performed better than baseline models in every aspect, several reasons could be that the adjacency matrix it learnt captures interdependencies better than initializing a matrix. Additionally, the modules such as the mix-hop propagation contributes to overall performance

Limitations, Future Scope and Conclusion

- Even though difference is low, the value of Bitcoin is so high that difference is 500. This can lead to inaccuracy.

- While we used moving averages to train our hourly models to capture long-term dependencies, in the future, we plan on using more sophisticated ensemble methods such as training a meta-model using the prediction results of models in varying timeframes.

- Therefore, our MTGNN model performs better on various metrics.