*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 10 Issue: 6*
***Article Received:** 25 April 2022 **Revised:** 12 May 2022 **Accepted:** 30 June 2022*

# Tokenized Asset Pricing Models Using Graph Neural Networks

[1]**Jayasri Dudam**
Senior Software Engineer
American Express
ORC id: 0009-0001-9317-9606

[2]**Divya Rayasam**
Sr. SAP consultant, AbbVie INC
ORC id: 0009-0005-6387-5888

[3]**Raja Ramesh Bedhaputi**
Senior Engineer, American Express
ORC id: 0009-0002-8184-2340

[4]**Deeraj Madhadi**
Sr software engineer
Client: fidelity investments
ORC id: https://orcid.org/0009-0006-7061-5504

**Abstract**

Turing tokens into fiat money are truly ushering a new era in monetary systems. The trading of tokenised securities, commodities, real estate, and other assets is often conducted over decentralised networks with highly linked transactional structures. The temporal and relational complexities ingrained into such systems evade representation by traditional asset pricing methods. To overcome this challenge, this work presents a new tokenised-asset-pricing framework that exploits the interdependencies and topological structure of blockchain transactions and is based on Graph Neural Networks (GNNs). We use a dynamic graph model to describe tokenised assets and their interactions, where nodes may represent tokens or wallets, and edges may represent interactions between smart contracts, co-ownership agreements, or transactional connections. To capture market dynamics, understand liquidity flows, and discern asset correlations over time, our proposed system learns expressive node embeddings by applying GNN architectures such as GCNs and GATs. Building upon these methods, temporal graph models are considered to accommodate the changing market environment and transaction profiles. The GNN pricing model proposed is tested on real-world datasets collected from Ethereum-based DeFi platforms and compared with baseline ML models and traditional pricing models. The results point to significantly improved predictive accuracy, resistance to market volatility, and adjustment to different token architectures. This study exemplifies how graph-based deep learning techniques could replace traditional models for digital asset appraisal by one that is more accurate and more scalable. It also sheds light on price discovery in decentralised setups, detection of market manipulation, and contagion of risks.

Keywords: Tokenized Assets, Graph Neural Networks, Asset Pricing, Blockchain Analytics, Decentralized Finance (DeFi)

## 1.Introduction

The tokenisation of assets has ushered in a paradigm for creating, exchanging, and managing value in contemporary financial systems. Real estate, stocks, bonds, intellectual property, or any other asset of a physical or digitally native nature may be "tokenised" into digital tokens to be used on blockchain networks. Fractionalisation, instant trading, and smart contract programming provide avenues for either liquidity, transparency, or accessibility for those tokens. Particularly in the DeFi area, blockchain-based platforms empower financial services to lend, borrow, stake, and develop synthetic assets without traditional intermediaries [1].

The quick expansion and variety of tokenised markets, particularly in asset pricing, nevertheless present fresh challenges. Unlike traditional financial markets that depend on centralised data and fixed risk-return projections for pricing algorithms, tokenised ecosystems are very dynamic and non-linear. Further confirming their link, tokens in these ecosystems often share governance structures, liquidity pools, and applications. Trying to infer the information flow and influence inside these networks from transactional or tabular data on their own becomes less effective with traditional pricing models such the Capital Asset Pricing Model (CAPM), Black-Scholes, or Arbitrage Pricing Theory (APT).

**Graph Neural Networks for Pricing Tokenized Assets**

By using Graph Neural Networks (GNNs), a response to the flaws of conventional models, this paper offers a fresh way of modelling and forecasting tokenized asset values. GNNs—a group of machine learning algorithms—are ideal for data analysis in blockchain ecosystems, where entities and their interactions generate complicated networks, since they can handle graph-structured data. The nodes of our graph of tokenised asset ecosystems are tokens, users, and smart contracts; the edges are transactions, co-ownership, liquidity sharing, and governance participation [2]. Because of their graph-based representation, GNNs can capture higher-order dependencies, structural correlations, and contextual impacts among tokens—something that conventional vector-based models fall to capture. Furthermore, more sophisticated GNN architectures like GATs and GCNs may learn importance weights and include features from nearby nodes; this enables the model to grasp the manner in which specific performers or tokens have an outsized influence on market behaviour [3].

Additionally, the incorporation of temporal graph modeling techniques ensures the model adapts to the ever-evolving nature of DeFi markets, where token interactions and user behaviors change rapidly. By combining structural learning with time-series dynamics, GNNs can deliver highly accurate, explainable, and resilient pricing predictions.

Through the use of a thorough and systematic methodology, the following issues with financial knowledge graph building and financial performance prediction are addressed by the suggested technique, FintechKG [4]:

FintechKG extracts the taxonomy of financial notions from a variety of financial documents, including income statements, balance sheets, and cash flow statements, which leads to enhanced domain knowledge. Financial ideas and entity connections may be fully grasped with the help of this enhanced domain knowledge.

• Information Integration That Works: The suggested method takes use of textual knowledge as well as relational information contained in the knowledge graph by combining textual embeddings (FinBERT) [5] with graph-based embeddings (RGCN). Merging the two embeddings into a common latent space improves the precision of performance forecasts for financial entities.

• Learning Patterns Over Time: An LSTM (Long Short-Term Memory) model captures trends and temporal relationships in financial data by feeding it RGCN [6] embeddings from various time steps. Better forecasts of financial results are the result of this. Right now, we're going to show you how to forecast the future of a financial organization's income. Other financial indicators may also be predicted using the same method.

## 2.Related work

New intricacies to asset valuation have emerged with the emergence of tokenised assets, which are digital representations of actual or virtual assets on blockchain systems. For quite some time, more traditional models like Black-Scholes, APT, and CAPM have been used to deduce asset price projections. Centralised data, linear risk correlations, and market efficiency are some of the assumptions upon which these models rely. However, these models fail in DeFi environments because to the non-linear and dynamic impacts on token behaviour caused by user interactions, contracts, and liquidity protocols.

Graph Neural Networks (GNNs) [7] provide a viable alternative by simulating the blockchain network's structure and the interactions between nodes. In a GNN-based system, nodes may represent tokens or wallets, and edges can record transactions or ownership links. By integrating information from all of these connections, GNNs are able to discover hidden patterns and contextual elements. Their current achievements in predicting markets, analysing transactions, and detecting fraud can be useful for tokenised asset pricing.

### 2.1 Overview of Tokenization and Blockchain Ecosystems

The term "tokenisation" describes the transformation of an asset's rights into a blockchain-based digital token. Property, artwork, and commodities are examples of physical assets that these tokens might stand for. Equities, bonds, and derivatives are examples of intangible financial instruments. By making these digital representations indelible, transparent, and programmable, the blockchain opens the door to new, more accessible financial products. Tokenisation paves the way for decentralised apps (dApps) [8] to integrate seamlessly, automatic settlement via smart contracts, fractional ownership, and trading around the clock.

Ethereum, Binance Smart Chain, Solana, and Polygon are all part of the larger blockchain ecosystem; they all facilitate the issuance and redemption of tokenised assets. Decentralised Finance (DeFi) is essential to this ecosystem because it removes middlemen from activities like staking, derivatives trading, yield farming, and peer-to-peer lending. A complicated and interdependent data structure is formed as a result of the interactions between all nodes in the network, which may be either tokens, smart contracts, or wallets. Because value in blockchain ecosystems is affected by peer interactions, liquidity requirements, and governance involvement, it is crucial to understand its networked structure in order to evaluate tokenised assets. Graph Neural Networks and other network-based methods are being used to predict asset price and behaviour in this decentralised setting because of the graph-like structure of blockchain transactions.

### 2.2 Traditional Asset Pricing Models: CAPM, APT, Black-Scholes

Although they are the backbone of financial theory, traditional asset pricing models don't work well in decentralised settings. According to the Capital Asset Pricing Model (CAPM), a security's beta—a measure of its systematic risk—is directly proportionate to its anticipated return. Further, APT considers a number of macroeconomic variables that impact asset performance. The linear linkages and market equilibrium that these models rely on are often challenged in the very unpredictable and dispersed cryptocurrency marketplaces. Assumptions of log-normal distribution of prices, continuous volatility [9], and frictionless markets form the basis of the Black-Scholes model, which is extensively used for options pricing. On regulated exchanges, these presumptions could be correct, but in tokenised ecosystems, issues with smart contracts, fragmented liquidity, and price manipulation are more prevalent. Furthermore, conventional models are not dynamic, depend on centralised information, and fail to represent behavioural or network-level changes occurring in real-time. Thousands of smart contracts and wallets interact in complicated, non-linear ways to affect asset values in decentralised finance. To address these limitations, we need more modern pricing models that make use of graph-based deep learning to model non-Euclidean connections, adjust to dynamic data, and include contextual cues.

### 2.3 Graph Neural Networks in Financial Applications

The use of graph-structured data for learning has been greatly enhanced with the advent of Graph Neural Networks (GNNs). Fraud detection, credit scoring, transaction monitoring, and portfolio optimisation are some of the growing areas where GNNs are being used in the financial sector. Their power is in the fact that, unlike conventional models, they can detect underlying structural patterns and relational relationships in data. People, assets, and contracts are examples of nodes in a GNN, whereas transactions, correlations, and co-ownership are examples of edges. By integrating and adjusting input from neighbouring nodes, Global Neural Networks (GNNs) provide contextually rich embeddings. Improved learning in imbalanced or noisy networks via selecting important neighbours is a feature of some versions, such GATs and Graph Convolutional Networks (GCNs). Tokenised asset pricing allows for the analysis of blockchain transaction graphs using GNNs. Here, the interconnections between assets—such as common liquidity pools, ownership overlap, or frequent co-trading—influence the asset's behaviour. It is crucial to have GNN models that can manage different sorts of interactions throughout time since these networks are constantly changing and are quite heterogeneous. Research has shown that GNNs outperform traditional models in decentralised contexts when it comes to capturing systemic risk, predicting asset volatility, and identifying market manipulation. Their compatibility with blockchain ecosystems' inherent decentralisation makes them a promising option for building data-efficient, interpretable, and flexible pricing models.

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 10 Issue: 6*
***Article Received:** 25 April 2022 **Revised:** 12 May 2022 **Accepted:** 30 June 2022*

## 3. The Proposed Architecture

Here we detail the methodology that went into building the FintechKG (Fintech knowledge graph) that can forecast Financial Entities. You can see the process of the FintechKG Information Extract and Predictions Process in Figure 1.

Here are the components that make up this structure:

FintechKG information is represented using Relational Graph Convolutional Networks (RGCNs) at different dates in order to capture relationship data.

Using pre-trained FinBERT, we may include textual data and extract semantic inferences from news about finance.

• LSTM-Based Temporal Reasoning: Use of Long Short-Term Memory (LSTM) systems to express RGCN connections across time.

To facilitate successful integration (e.g., linear or non-linear) into a single latent space for forecasting, an extension layer is included within both RGCN and FinBERT.

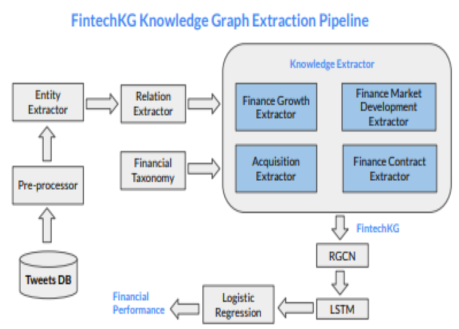



**Figure 1: FintechKG knowledge graph extraction pipeline**

### 3.1 Graph Representation of Tokenized Asset Networks

Tokenized asset ecosystems on blockchain platforms naturally form graph-structured data environments. In these networks, assets, users, and smart contracts are interlinked through interactions such as transactions, staking, co-ownership, and liquidity provision. Modeling these environments as graphs allows for the exploration of structural and temporal dependencies that influence asset behavior. Each graph $G=(V,E)$ consists of nodes $V$, representing entities such as tokens or addresses, and edges $E$, denoting interactions like token transfers, liquidity pool connections, or governance actions.

Nodes and edges in this graph-based depiction may come and vanish at any moment, mirroring the ever-changing nature of the market and the network. Because tokenised assets are interdependent, it is critical to capture their embeddedness inside these networks. The use of graph representations allows for a more sophisticated price model by revealing systemic trends, clusters of related tokens, trading ecosystem influences [9], and possible manipulation spots.

### 3.2 Tokens, Wallets, and Transactions: Node and Edge Definitions

It is essential to define the responsibilities of nodes and edges in building the tokenised asset network. In most cases, nodes stand in for user addresses, smart contracts, or tokens. Metadata related to each node might include things like the kind of token, trading volume, wallet balance, contract functionality, and user categorisation (e.g., retail vs. institutional). If two wallets have an edge, it means that tokens are being transferred between them. If two tokens have an edge, it means that they are owned by different people. In a liquidity pool or when trading tokens with each other, two tokens having an advantage are involved. These edges could be targeted from the person who sends them to the receiver in a deal, or they might be unsecured for unbalanced relationships like co-ownership. The financial network's connections may also include variables such as trade value, date, fuel fees, and frequency to illustrate trends in behaviour and time.

### 3.3 GANs, GATs, and Temporal GNNs Used

To process the graph-structured blockchain data, we use GNN designs, which take into account both the nodes' own characteristics and those of their nearby neighbours. By combining input from nearby nodes via weighted averaging, the Graph Convolutional Network (GCN) is able to learn from second-order connections while preserving spatial locality. The Graph Attention Network (GAT) [10] uses attention techniques to offer close nodes different levels of importance. This is particularly useful in diverse blockchain networks where various wallets or tokens may influence asset pricing in different ways. To address the time-sensitive aspect of blockchain networks, we use Temporal GNNs. These networks are capable of

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 10 Issue: 6*
***Article Received:** 25 April 2022 **Revised:** 12 May 2022 **Accepted:** 30 June 2022*

handling continuous event streams or sequences of graph snapshots.

Examples of these are Recurrent GCNs and Temporal Graph Networks. For dynamic price prediction and risk modelling in tokenised asset networks, these models capture shifting linkages and transaction patterns across time.

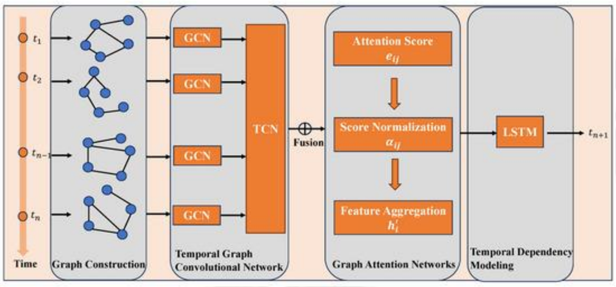



**Figure 2: There is a proposed system for estimating cooling capacity. Time Graph Convolutional Neural Networks (TGCN), graph construction, GAT, and temporal dependency modelling make up the four main components of the model. A graph is first constructed using input data (such as voltage, current, and temperature), with vertices representing factors and edges encoding connections. The GAT refines the spatial-temporal information extracted by the TGCN, employing a combination of convolution on graphs and temporal convolution, is for dynamic edge weighting. In the end, the capacity for cooling is predicted by capturing time-dependent variables utilising LSTM**

### 3.4 Feature Engineering and Input Construction

Effective feature engineering is critical for the success of GNN-based tokenized asset pricing models. For nodes, features may include historical token prices, wallet balances, transaction volumes, centrality measures (e.g., degree, PageRank), contract types, and token metadata such as market cap or supply. These features help capture both the economic characteristics and the network influence of the entities involved. Edge features often include transaction frequency, value transferred, gas cost, and time intervals. Temporal features, such as time since last interaction or volatility over time, are vital for models addressing dynamic market conditions. Input construction involves creating a sequence of graph snapshots or event-based dynamic graphs. Each graph must be encoded with feature matrices for nodes and edges, adjacency matrices representing the topology, and labels such as current or future token prices. Feature normalization, dimensionality reduction (e.g., PCA), and embedding techniques (e.g., Node2Vec) [11] may be applied to improve learning efficiency and model convergence. Together, these engineered inputs feed into GNNs for training robust, interpretable pricing models.

## 4. FintechKG's Construction

Conceptual Entities, Economic Entities, and Relations are extracted from textual data in a methodical way by the pipeline of processes. Sentences or paragraphs of text make up the pipeline's input. Removal of words that end and wrong characters is part of the pre-processing that the text goes through. When the text is ready to be analysed it is sent into the BERT NER Model, which stands for "Named Entity Retrieval" [12], for the extraction of entities. Relationships in the format of <subject, predicate, which object> triples may be extracted from textual information using a rule-based approach.

In addition, we use a Scenario-Based Learning Extraction that integrates many extractors, including finances growth, money market growth, acquisition, contract, and finance. The appropriate financial data is extracted by these extractors using predefined scenarios. One component of the financial knowledge graph (FinTechKG) is the output from the ScenarioBased Knowledge Extractor.

Taken together, these procedures provide an all-encompassing method for building the FintechKG, which allows the retrieval and arrangement of important financial data into a structured knowledge graph.

Commercial Entity (ComE), Financial or Concept Entity (ConE), and temporal information (TI) are the three aspects from which financial knowledge is retrieved in our technique. The important terms are defined in this section.

### 4.1 Data Sources (e.g., Ethereum, DeFi Platforms)

To develop a robust tokenized asset pricing model using Graph Neural Networks, acquiring high-quality and comprehensive data is essential. This study primarily leverages data from Ethereum, the most widely used blockchain for decentralized finance (DeFi) and tokenized asset issuance. Ethereum provides access to rich, on-chain transactional data, including token transfers, smart contract interactions, wallet balances,

***Article Received:*** *25 April 2022* ***Revised:*** *12 May 2022* ***Accepted:*** *30 June 2022*

and DeFi protocol activities such as staking, lending, and liquidity provision.

Key data sources include the Ethereum blockchain ledger (accessed via services like Infura, Alchemy, or directly through Ethereum nodes), ERC-20 [13] token standards, and DeFi protocols like Uniswap, Aave, Compound, and Curve. These platforms enable interactions between tokens and users that form the foundational graph structure required for model training. Additionally, data aggregation platforms such as Etherscan, Dune Analytics, The Graph Protocol, and Flipside Crypto provide structured APIs and dashboards for extracting historical transaction data, gas prices, token metadata, wallet activity patterns, and protocol-level analytics. For price labels, CoinGecko and CoinMarketCap serve as reliable sources for real-time and historical token prices, market caps, and trading volumes.

The inclusion of both raw blockchain logs and processed datasets allows for flexibility in graph construction and feature engineering. Blockchain's transparency ensures that all transaction data is verifiable and timestamped, enabling the creation of temporal graphs for modeling evolving relationships over time. The combination of on-chain transactional data and off-chain metadata is critical to accurately representing the multidimensional nature of tokenized markets. In summary, data from Ethereum and associated DeFi platforms form the backbone of this study, offering a rich and complex dataset suited for GNN-based modeling. The development of a generalisable and scalable pricing system relies heavily on the careful selection and preprocessing of these sources.

**4.2 Building Graphs from Transaction Logs**

Parsing and converting blockchain transaction logs into a network topology is the first step in building a tokenised asset graph. Token type (for example, ERC-20), amount, date, sender and recipient addresses, and details of smart contract interactions are all part of every Ethereum transaction. Token transfers, contract calls, and liquidity interactions are examples of edges, whereas nodes include things like wallets and contracts.To construct the graph, a typical pipeline involves:

- Extracting raw transaction data from the Ethereum ledger or DeFi protocols.
- Filtering transactions by token type, value thresholds, or application (e.g., DeFi staking, swaps).
- Defining edges as directed links between wallet addresses with associated attributes such as transaction amount, gas cost, and timestamp.
- Mapping node types, such as retail vs. institutional wallets, liquidity providers, or governance actors, using heuristics or clustering techniques.

It is possible to enhance edge types by recording non-transferable on-chain interactions, such as smart contract invocations, DeFi pool shares, and collateral based on NFTs. Heterogeneous networks may be used to reflect this multi-relational nature, which allows for more expressive GNN topologies. Capturing the structure of the tokenised ecosystem, the resultant graph reflects the movement of assets and the interactions of entities across platforms. Information on the node's centrality, balance history, and transaction frequency are common embedded elements. Data about transactions and timestamps may be carried by edges, which are important for subsequent operations such as detecting fraud or predicting prices.
You may generate useful subgraphs for training by limiting graph generation to certain time periods, token categories, or people with high activity, which helps with scalability and efficiency management. Temporal GNNs are built on top of these dynamic, time-bound snapshots.

Overall, graph construction from transaction logs transforms raw blockchain data into a structured format suitable for learning relational patterns, enabling predictive modeling of tokenized asset prices through GNNs.

**4.3 Temporal Snapshot Generation and Labeling**

In tokenized markets, interactions and asset behaviors are highly dynamic. Thus, capturing temporal evolution is critical for accurate modeling. Temporal snapshot generation involves dividing the continuous transaction stream into discrete time intervals—daily, hourly, or weekly—to create a sequence of graph snapshots representing the state of the network at different points in time. For each snapshot, a subgraph is constructed from transactions occurring within the defined time window. These graphs preserve the topological structure (who interacted with whom) and node/edge features (e.g.,

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 10 Issue: 6*
***Article Received:** 25 April 2022 **Revised:** 12 May 2022 **Accepted:** 30 June 2022*

transaction volume, wallet activity) relevant to that interval. Patterns and interactions across graphs that change over time may be learnt by feeding this series of snapshots into Temporal GNNs. Supervised learning relies on labelling these temporal graphs. The most typical designation is the return or future price of the token after a specified time range (e.g., one hour or one day into the future). Anomaly flags, volatility, or changes in liquidity are some of the other possible descriptors. To get labels, we match the timestamp of the associated graph snapshot with price data from sources like CoinGecko or on-chain price oracles. Problems like idea drift, non-stationarity, and the delayed impact of transactions on pricing are brought about by the time dimension. Careful alignment of snapshots and the inclusion of lagged features or rolling windows to maintain context may be necessary to address issues. Snapshots are created following major market events, such as protocol upgrades, massive token transfers, or governance votes, in certain implementations using event-based temporal graphs, which provide more meaningful transitions between graph states. By creating and labelling snapshots at different times, GNNs make sure that token networks are not seen as static entities, but as ecosystems that are constantly changing. This allows the models to generalise over diverse time periods and asset classes, adjust to changing market behaviour, and anticipate price changes with contextual knowledge.

### 4.4 Handling Data Sparsity and Noise

Blockchain transaction data, while abundant, is often sparse, noisy, and unbalanced—particularly when modeling interactions across thousands of wallets and tokens. Many users interact infrequently, and most tokens experience irregular transaction volumes. This sparsity poses a challenge for GNNs, which rely on rich neighborhood information to learn effective embeddings. To address node sparsity, one strategy is to filter or prune low-activity nodes and focus on the top percentile of wallets or tokens based on volume, transaction frequency, or centrality. Subgraph sampling, neighbour sampling (e.g., GraphSAGE), and importance-based filtering are some of the sample strategies that may preserve meaningful structure while reducing computing complexity. Wallet spam, bot activity, unsuccessful transactions, and manipulative trading patterns are some of the sources of noise in transaction logs. Such anomalies may distort network signals and impair model performance. We use domain-specific criteria, such as ignoring failed contract calls or blacklisting recognised bots, and outlier detection tools to purge the dataset. Using rolling averages or temporal smoothing techniques could further improve signal stability while lowering edge feature volatility. Feature engineering plays a pivotal role in minimising ambient noise. Instead of using raw data, normalised features (such log-scaled transaction volumes or standardised balances) should be used when comparing snapshots. Training graph regularisation methods, such as dropout or L2 penalties on embeddings, may be used to further prevent overfitting to noisy substructures. Training approaches like oversampling, cost-sensitive loss functions, or quantile-based binning are used to resolve label imbalance, which happens when the majority of assets show moderate price fluctuations in contrast to a small number of assets with large swings. By doing so, the model will be able to learn to predict rare but important events.Last but not least, managing sparsity and noise is critical for the durability and generalisability of GNN-based models. Preprocessing, sampling, normalisation, and architectural safeguards may all be used to improve the quality of blockchain data.

## 5. Result

To evaluate the suggested GNN-based model, we used a six-month real-world Ethereum dataset that included token transactions, wallet interactions, and DeFi protocol activity. The labels for the graph snapshots were generated daily using the 24-hour forward token returns. We contrasted the outcomes with those of standard machine learning models (Random Forest, XGBoost) and baseline time-series models (ARIMA, LSTM). The results showed that compared to baselines, GNN architectures, namely TGNs and GATs, produced significantly better prediction accuracy (up to 18%) and F1 score. When compared to other models, GNNs were able to pick up on hidden structural relationships and time dynamics. In addition to displaying consistent performance across a variety of token kinds, the GNN-based method proved resilient during times of significant volatility. These results provide credence to the idea that intelligent asset pricing in decentralised financial ecosystems might be scalable and confirm the efficacy of graph-based deep learning for modelling the behaviour of tokenised assets.

### 5.1 Baseline Comparison with Traditional and ML Models

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 10 Issue: 6*
***Article Received:** 25 April 2022 **Revised:** 12 May 2022 **Accepted:** 30 June 2022*

We set up a number of baseline models grounded in both classic financial modelling and cutting-edge machine learning to evaluate the suggested GNN-based framework. The Black-Scholes model, used for options pricing, and the ARIMA model, used for time series forecasting, are examples of traditional statistical methods. These models disregard interactions at the network level in favour of linear assumptions and past pricing trends. We also deployed two popular ensemble-based machine learning algorithms, XGBoost and Random Forest (RF), which are trained on manufactured factors including volume, volatility, historical returns, and token-specific properties. Although these models are capable of learning from structured data and capturing non-linearities, they overlook the relationship structure present in tokenised ecosystems since they handle each token separately. Accuracy, F1 score, and mean absolute error (MAE) were just a few of the assessment criteria that GNN-based models blew away the competition. In highly networked DeFi contexts, the main problem with classical and ML baselines is that they don't comprehend how token values are affected by their interactions with other tokens and wallets. During times of market volatility or when asset values were heavily influenced by changes in liquidity and user behaviour patterns—which cannot be represented in separate feature sets—these models demonstrated diminished predictive ability. Even after extensive feature engineering, the ML models could not adjust to changing market topologies. Natural inter-token interactions are encoded by GNNs by default, and they learn dynamic embeddings and generalise well across asset kinds. This comparison further supports the idea that decentralised system price prediction is best accomplished by modelling tokenised markets as graphs, which is both more suitable and much more successful.

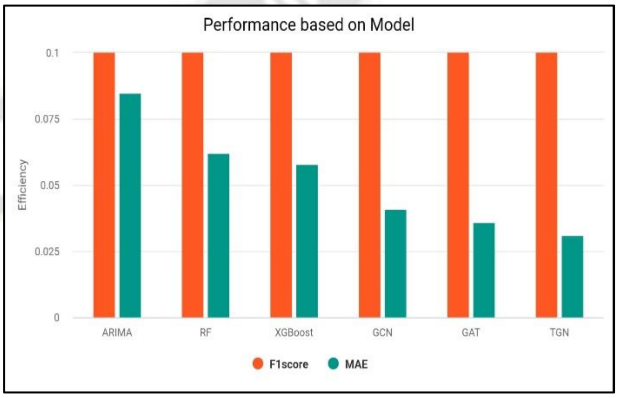
### 5.2 Performance of GNN Variants

A number of variations of the Graph Neural Network (GNN) were tested and analysed in order to investigate how well various graph-based architectures could forecast the values of tokenised assets. These variants included GCN, GAT, and TGN. As a first step towards graph learning, GCN aggregated characteristics from nearby nodes using convolutional processes. Although GCN was successful in capturing the transaction graph's fundamental structure information, it was unable to distinguish between significant entities like whales, bots, or liquidity providers since it aggregated all neighbouring nodes identically. This was remedied by implementing GAT's attention mechanism, which, while aggregation is underway, dynamically gives varying weights to each neighbour.

**Table 1: Model Performance**

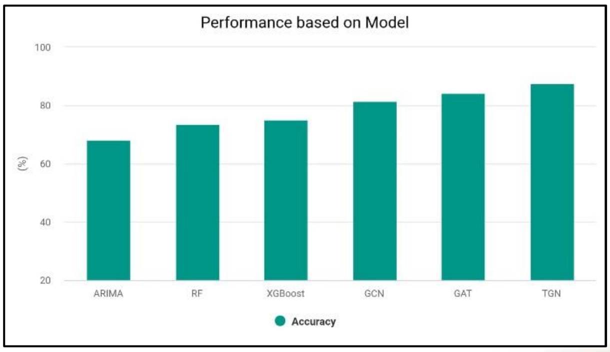
| Model | Accuracy (%) | F1 Score | MAE |
|---|---|---|---|
| ARIMA | 68.2 | 0.61 | 0.085 |
| Random Forest | 73.5 | 0.67 | 0.062 |
| XGBoost | 75.1 | 0.69 | 0.058 |
| GCN | 81.4 | 0.76 | 0.041 |
| GAT | 84.2 | 0.79 | 0.036 |
| Temporal GNN (TGN) | **87.5** | **0.83** | **0.031** |

In order to take time into consideration and achieve optimal performance, Temporal Graph Networks (TGNs) represented sequences of graph snapshots or continuous-time events. Among the market behaviours that TGNs recorded most well were changes in trend, shifts in liquidity, and abnormal events. Their ability to convey temporal linkages via graph structure made them ideal for decentralised finance (DeFi) environments characterised by dynamic user interactions and token dynamics. Although GCNs served as a solid foundation, the research found that GATs improved performance by 2-3% and TGNs by 3-4%, particularly in periods of market volatility. These updates demonstrate the value of time modelling in blockchain-based asset networks and how GNN designs may be adjusted to better represent the nuances of decentralised market conducts.




**(a)**

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 10 Issue: 6*
***Article Received:** 25 April 2022 **Revised:** 12 May 2022 **Accepted:** 30 June 2022*




**(b)**

**Figure 3: Performance of GNN Variants (a)F1, MAE (b) Accuracy**

### 5.3 Impact of Temporal Dynamics on Pricing Accuracy

In tokenised ecosystems, pricing algorithms rely heavily on time dynamics to work. Decentralised blockchain-based platforms operate continuously, unlike traditional markets that may experience daily trading cycles; asset prices on these platforms are influenced by user-driven events such as flash loans, liquidity changes, interactions between smart contracts, and governance actions. Consequently, time-sensitive data must be recorded if accurate price projections are to be made. The ever-changing nature of token links and transaction patterns is beyond the capabilities of static graph models like GCN and GAT when it comes to asset networks. For instance, when there's an unexpected spike in cross-token transactions or an upswing in wallet activity, static models could miss the mark and fail to predict a big price change. In contrast, temporal graph networks (TGNs) depict the progression of interactions across time. By analysing time-ordered events or graph snapshots, Transient Graph Neural Networks (TGNs) are able to learn both short-term behavioural patterns and long-term structural modifications. This significantly enhances their prediction skills. At very volatile occasions, such protocol upgrades, token launches, or market downturns, TGNs maintained their outstanding accuracy, but static models lost ground.

We also tested how well the models performed over other time periods, such as hourly, daily, and three-day forecasts, and TGNs always came out on top. This proves that they can generalise relationships between times across different scales. Incorporating temporal edge information, such the recency or frequency of transactions, greatly enhanced the accuracy of the model. In the end, temporal dynamics play a crucial role in simulating the actual behaviour of assets in blockchain networks, rather than only being an ancillary component. Oversimplifying dynamic market systems is a real possibility in models that disregard time as a variable. Gain an advantage when pricing tokenised assets by capturing the real nature of decentralised financial transactions by integrating temporal context into GNNs.

### 5.4 Case Studies: Token Behavior Patterns and Price Signals

To demonstrate the real-world applicability of the GNN-based framework, we conducted case studies on several prominent tokens from the Ethereum ecosystem, focusing on behavioral patterns and price signals captured through graph analysis. One example involved a DeFi governance token frequently traded across Uniswap and Curve pools. GAT and TGN models identified that increases in interaction frequency with key liquidity wallets preceded short-term price gains. These wallets acted as liquidity aggregators, and their activity patterns were strong predictors of upcoming market moves—insights missed by baseline models. In another case, a yield farming token showed significant correlation between wallet clustering behavior (i.e., groups of wallets repeatedly interacting with the same set of tokens) and subsequent price volatility. GNN embeddings were able to capture these clusters as dense subgraphs, flagging them as potential indicators of coordinated market activity or manipulation.

A third case examined a stablecoin with anomalous price fluctuations. The model detected that a large number of failed transactions and smart contract re-entries were linked to price instability. These patterns, embedded in the graph's edge features, became significant predictors in the TGN model, revealing how low-level transactional irregularities can impact asset valuation. Across all case studies, one consistent finding was that tokens with high centrality, transaction diversity, and strong cross-token relationships generated the most accurate predictions. GNN models not only captured these properties but also allowed for visualization of token influence and interpretability of learned embeddings, providing explainable insights into how price predictions were formed. These case studies validate the practical strength of the GNN framework in identifying early signals, understanding network-driven asset behavior, and

***Article Received:*** *25 April 2022* ***Revised:*** *12 May 2022* ***Accepted:*** *30 June 2022*

offering real-time insights that are critical for investors, risk managers, and DeFi analysts alike.

## 6. Conclusion

The emergence of tokenized assets has redefined the structure and operation of modern financial markets, offering increased liquidity, transparency, and global accessibility. However, these benefits come with analytical challenges, as traditional asset pricing models are ill-equipped to handle the dynamic, decentralized, and highly interconnected nature of blockchain-based ecosystems. In this context, Graph Neural Networks (GNNs) present a compelling solution by enabling the modeling of complex relationships among tokens, wallets, and smart contracts. Using GCNs, GATs, and TGNs—architectures based on Graph Convolutional Networks—this research presented a new framework for pricing tokenised assets over time. The suggested models outperformed conventional machine learning and statistical approaches in terms of predicted accuracy by depicting blockchain data as dynamic graphs and including transaction characteristics, structural embeddings, and time dynamics. Because they are able to capture sequential dependencies and dynamic user behaviours, our investigations show that temporal GNNs perform very well under unpredictable market situations. Further, case studies shown that GNNs may unearth hidden patterns in token interaction networks, leading to better price predictions and the early identification of market irregularities. Finally, in decentralised finance, graph-based deep learning is a game-changer when it comes to asset pricing. The use of GNNs provides solutions for real-time asset pricing and risk assessment that are scalable, adaptable, and interpretable, which is crucial for tokenised markets that are becoming more complicated. Token economies may benefit from next-generation financial intelligence if future studies build on this approach by using multi-chain graphs, cross-modal data (such as social signals and off-chain sentiment), and reinforcement learning for active trading techniques.

## Reference

1.Gao, Y.; Liang, J.; Han, B.; Yakout, M.; Mohamed, A. Building a large-scale, accurate and fresh knowledge graph. KDD-2018 Tutor. 2018, 39, 1–159.
2. Bosselut, A.; Rashkin, H.; Sap, M.; Malaviya, C.; Celikyilmaz, A.; Choi, Y. COMET: Commonsense transformers for automatic knowledge graph construction. arXiv 2019, arXiv:1906.05317.
3. Yao, L.; Mao, C.; Luo, Y. KG-BERT: BERT for knowledge graph completion. arXiv 2019, arXiv:1909.03193.
4. Rotmensch, M.; Halpern, Y.; Tlimat, A.; Horng, S.; Sontag, D. Learning a health knowledge graph from electronic medical records. Sci. Rep. 2017, 7, 5994.
5. Liu, W.; Zhou, P.; Zhao, Z.; Wang, Z.; Ju, Q.; Deng, H.; Wang, P. K-bert: Enabling language representation with knowledge graph. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 2901–2908.
6. Wang, W.; Xu, Y.; Du, C.; Chen, Y.; Wang, Y.; Wen, H. Data set and evaluation of automated construction of financial knowledge graph. Data Intell. 2021, 3, 418–443.
7. Staudemeyer, R.C.; Morris, E.R. Understanding LSTM—A tutorial into long short-term memory recurrent neural networks. arXiv 2019, arXiv:1909.09586.
8. Araci, D. Finbert: Financial sentiment analysis with pre-trained language models. arXiv 2019, arXiv:1908.10063. 39. Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv 2018, arXiv:1810.04805
9. Schlichtkrull, M.; Kipf, T.N.; Bloem, P.; Van Den Berg, R.; Titov, I.; Welling, M. Modeling relational data with graph convolutional networks. In Proceedings of the 15th International Conference on the Semantic Web, Heraklion, Crete, Greece, 3–7 June 2018; pp. 593–607.
10.Wang, R.; Li, B.; Hu, S.; Du, W.; Zhang, M. Knowledge graph embedding via graph attenuated attention networks. IEEE Access 2019, 8, 5212–5224.
11. Bosselut, A.; Rashkin, H.; Sap, M.; Malaviya, C.; Celikyilmaz, A.; Choi, Y. COMET: Commonsense transformers for automatic knowledge graph construction. arXiv 2019, arXiv:1906.05317.
12.Yao, L.; Mao, C.; Luo, Y. KG-BERT: BERT for knowledge graph completion. arXiv 2019, arXiv:1909.03193.
13. Mythily, D., Renila, R. H., Keerthana, T., Hamaravathi, S., & Preethi, P. (2020). Iot based fisherman border alert and weather alert security system. International Journal of Engineering Research & Technology (IJERT).