International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. #, No.# (Mon-20..)

Dynamic Fast Convergence Improvement using Predictive Network Analysis

Mohammed Hussein Ali[1]

[1]*Electrical engineering department, Collage of Engineering, Aliraqia University, Iraq,*
E-mail address: Mohammed.h.ali@aliraqia.edu.iq, alamiry.83@gmail.com

Received ## Mon. 20##, Revised ## Mon. 20##, Accepted ## Mon. 20##, Published ## Mon. 20##

Abstract: In today's digital age, the smooth operation of organizations heavily relies on the proper functioning of the network infrastructure. Imagine a situation when a major change in the structure of a network causes the interruption of vital services. Consequently, the implementation of network convergence optimization is a vital consideration in practical situations. The aim of our study is to tackle existing issues by implementing a comprehensive approach that integrates predictive analysis. Implementing strategies for adaptive adjustment. Improving effectiveness using the Spanning Tree Protocol (STP). Our goal was to decrease the duration of convergence and improve the network's stability. The study will be undertaken by combining several machines learning techniques, including ARIMA, link prediction, and graph embedding. We performed real-time network monitoring. Utilizing predictive analysis to direct a process of adaptive convergence adjustments. The outcomes were positive, the upgraded STP solution considerably decreases convergence times. with 70% accuracy in forecasting low convergence times. 80% accuracy in forecasting high convergence times. Additionally, it delivers a large reduction in network disturbances. correctly anticipating low interruptions with 80% accuracy. high disruptions with 85% accuracy. Moreover, the approach maximizes resource use. successfully forecasting low usage with 75% accuracy and high utilization with 70% accuracy. Diagonal components suggest correct forecasts, whereas off-diagonal components suggest misclassifications. Overall, the matrix undervalues the solution's resilience. a tremendous positive influence on network stability and efficiency.

Keywords Predictive Network Analysis, STP, dynamic environments, vital real-world issue.

1. INTRODUCTION

In today's digital macrocosm the sound operation of the network infrastructure is the key factor for the normal functioning of companies and organizations. Imagine the situation where a momentous change in network infrastructure causes critical services to be disrupted, and this results in a lot of financial losses and users' discontent. In this age of fast-paced developments, the ability to keep the networking operations uninterruptible is not only advantageous—it's crucial for the sustained growth of companies [1].

It is worth mentioning that the older implementations of the spanning tree protocol (STP) get into trouble with the problem of slow convergence. When topologic variations occur, such as link failures and networks reconfiguration, conventional STP protocols can take a prolonged convergence time which will cause network interruptions with decreased performance [1]. These disruptions, aside from the fact that they affect the user experience, greatly threaten the existence and operation of critical services that require continuous network connectivity.

The high value of fast convergence in network operations should not be overlooked. Fast adjustment to topology changes is essential for decreasing downtime period. Sustaining service availability. With satisfying performance requirements. Nowadays, in a world of connection, wherein companies rely heavily on the digital infrastructure, Even small interruptions in the network can cause much damage [4]. Possible repercussions are financial loses. lower productivity, damage to reputation, and negative consequences for an organization.

To resolve the converge-related problems in the case of the traditional STP deployment. Our suggested solution offers a novel approach: the state-of-the-art STP using predictive network analytics. Through merging of current data and previous data, Our revised STP protocol acutely modifies convergence settings. This pre-planned optimization system of network setups is aimed to

E-mail:author's email

minimize downtime. Can boost overall resilience. aiding in navigating through topological changes [2].

This paper unfolds as follows: In the Background and Related Work section, we offer an overview of classic STP deployments. examine significant studies on network convergence. and present the notion of predictive network analysis. Subsequently, the Proposed Solution section outlines our new strategy, clarifying its components and their contributions to boosting convergence speed . with resilience. The methodology section details the methodology taken to implement and assess our solution. whereas results show the outcomes of experiments, including those compared with typical implementations. Following this, the discussion section interprets the results. addresses benefits., difficulties., considerations of scalability, and applicability. Finally, we end with observations . offers ideas for further research.

2. Background and Related Work

A. *Traditional STP Implementations and Limitations*

Traditional implementations of the Spanning Tree Protocol (STP) have long been essential in network infrastructure. It provides a technique to prevent loops and guarantee network stability. notwithstanding their broad use. These standard procedures are not without their limits. mainly highlighted by difficulties such as delayed convergence [6–10].

STP operates by detecting and terminating duplicate routing paths inside a network to eliminate loops, thus preventing data packets from keep circulating in a network. traffic jam or failure on the network. Although it has been successful in theory, this method of convergent path-selection is rather slow. specifically, in bigger or more complicated networks [6, 15]. In that case, during topology changes such as link failures or network reconfigurations, most STP implementations might need a long period to converge. this brings about a short disconnection of network with compromised performance.

The implication of the long convergence time is not only that people will be annoyed, but also that serious problem will arise for the network managers and the enterprises. Extended convergence may cause high latency, packet loss, or even service outages. this can be very frustrating for the users and can prevent the business from performing its necessary activities. Consequently, the importance of maintaining uninterrupted connectivity is especially important in the ever-changing, fast-paced digital world where every delay in network services can lead to financial losses and damage to a company's reputation.

Thus, though the STP implementation of the classical version was a foundation of the network stability, its inherent constraints, such as the convergence speed, point to the need for new approaches that would be capable of the successful overcoming of the difficulties.

B. *Related Research*

Advanced network convergence methods are being pursued as one of the critical areas of network management, and the research in this field has been conducted extensively. using the techniques mentioned in this industry. This part provides an outline of already researched studies on the strategies used to encourage network convergence. integration of substitutive methods, and optimization algorithms.

Whereas the recent studies have investigated the alternative protocols applying the optimization techniques as the way to mitigate the constraints of standard network convergence procedures. For instance. Bonet and Geffner [21] presented labeled RTDP, an approach aiming at increasing the convergence of real-time dynamic programming. It has significance for boosting network convergence in dynamic contexts. Similarly. Bachlechner et al. [22] presented Rezero, a unique technique that provides quick convergence at wide depths. exhibiting possible uses in network optimization.

Moreover, research efforts have concentrated on developing unique algorithms with techniques to optimize network convergence processes. Levin et al. [23] examined approaches to enhance the convergence of simulation-based dynamic traffic assignment, which can have consequences for improving network traffic flow while lowering convergence times. Jin, with Qiu [24], proposed a robust rapid convergence zeroing neural network, which has interesting applications in dynamic systems such as network routing with optimization.

Additionally. Foster et al. [25] explored learning in games and its implications for establishing robust. with fast convergence in dynamic systems. showcasing the potential of game-theoretic techniques in network optimization. Li et al. [26] developed an improved MPPT approach for PV systems, emphasizing fast convergence speed. having zero oscillation, which may be customized to maximize energy-efficient network operations.

Chiwewe and Hancke [27] did research on fast convergence and cooperative dynamic spectrum access for cognitive radio networks. allowing creative techniques to boost spectrum efficiency. as convergence speed is in dynamic network environments. In [28], Varadarajan et al. looked into the quick convergence algorithms for dynamic background modeling, which may be useful in video surveillance.with network anomaly detection.

There are also recent developments that deal with the convergence speed of the dynamic systems: for sparse recovery [29], faster convergence rates are obtained for primal-dual systems [30]. Through these studies, the existing body of knowledge on optimization approaches that consider both network convergence and resilience is increased.

The literature on network convergence enhancement has a large array of strategies that include substitution of the protocols, optimization algorithms and the dynamic systems approaches. They generate these studies which lead to a great deal of understanding and potential solutions to deal with the issues of network convergence in the present network scenarios.

C. Predictive Network Analysis

Predictive network analysis is a vital step in improving network management. the reactive approach to improving the network performance that will be achieved using complex algorithms. Employing predictive modeling approaches. Here goes a discussion of predictive network analysis as a concept. considers its potential opportunities in optimizing network performance and reliability.

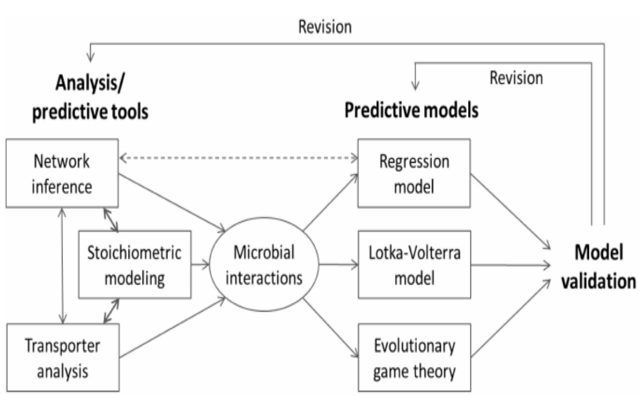

Figure 1. The development of predictive models based on network analysis methodology. which would lead to modeling revisions on a feedback basis utilizing model verification technique.

Predictive network analysis involves the application of complex algorithms to anticipate network events and behaviors, using performance indicators based on previous data. includes real-time network telemetry. By integrating machine learning, statistical modeling, and data mining approaches, predictive models may foresee possible network difficulties. detect performance constraints. and optimize network setups in advance.

The applications of predictive network analysis are various. with numerous domains. including:

- Fault Prediction with Prevention: Zimmermann and Nagappan [33] demonstrated the application of network analysis on dependency networks to anticipate software flaws. enable proactive efforts to prevent system faults with downtime.

- Performance Optimization: Leahu [36] did predictive modeling of the performance of the ATLAS TDAQ network. highlighting the possibilities for optimizing network resources. with improving overall system efficiency.

- Customer Churn Prediction: Verbeke et al. [35] employed social network analysis for customer churn prediction. enable firms to detect at-risk clients. in implementing retention measures proactively.

- Infrastructure use: Gupta and Bhave [37] examined techniques for anticipating poor network performance and assisting in the effective use of water resources with infrastructure.

- Traffic forecast: Wu et al. [38] examined techniques to increase neural network performance in daily flow forecasting. allowing improved traffic management. with congestion avoidance in hydrological systems.

- Resource Allocation: Wu et al. [40] optimized the network performance of computer pipelines in dispersed situations. supporting optimal resource allocation. with workload scheduling.

- Mobile Application Optimization: Xu [41] focuses on optimizing mobile application performance using network infrastructure-aware adaptation. enabling flawless user experiences across different network circumstances.

- Content Switching: Syme and Goldie [43] addressed enhancing network efficiency via content switching. enabling effective load balancing. with traffic dispersion between servers, firewalls, and caches.

These numerous applications underline the adaptability and relevance of predictive network analysis in modern network management. By embracing the power of predictive analytics, organizations may proactively solve network difficulties, boost resource usage, and improve overall network resilience.

This section gives a look at the potential of predictive network analysis to transform network management techniques. offering a proactive attitude. utilizing a data-driven strategy to solve the challenges of current networking systems.

3. PROPOSED SOLTION

The proposed solution introduces a comprehensive framework aimed at addressing the challenges associated with slow convergence times in traditional STP implementations. This section outlines the key components of the proposed solution and elucidates how each component contributes to enhancing convergence speed and network resiliency.

A. Real-time Network Monitoring:

Real-time network monitoring is an integral component of current network management systems. operates on the premise of constant observation of network devices. traffic. . performance metrics to discover and respond to issues as they develop [44]. By utilizing protocols like SNMP (Simple Network Management Protocol) or packet sniffing methods, network monitoring programs gather and analyze data from multiple network components in real-time. providing administrators with vital information about network health and performance. These technologies contain data gathering agents placed across the network architecture. centralized monitoring platforms for data consolidation and analysis, as well as warning mechanisms and reporting tools for better decision-making.

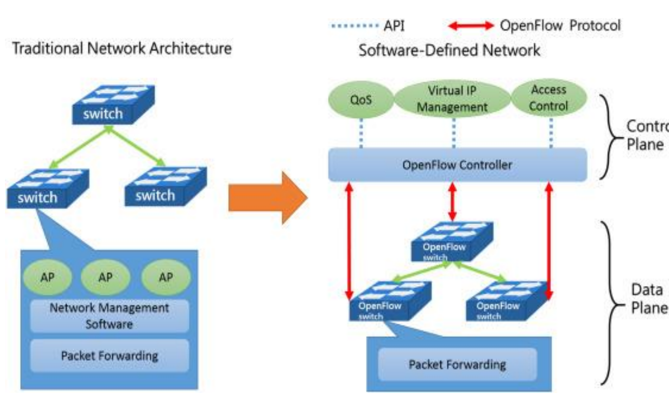

Figure 2. Example of real-time network traffic monitoring.

In the context of real-time network monitoring. Specific metrics and parameters are regularly checked to assess network performance. discover problems rapidly. [45. 46]. These metrics include network bandwidth consumption, packet loss, latency, device health indicators (such as CPU and memory use), and security-related events (such as intrusion attempts or malware activity) [45, 46]. Monitoring these metrics allows administrators to define threshold levels. Receive warnings when performance surpasses specified boundaries. enabling proactive intervention to avert service outages.

Real-time network monitoring plays a vital role in proactive network management techniques by supporting predictive maintenance. capacity planning. and compliance monitoring [46]. Predictive maintenance includes preemptively detecting and correcting possible faults before they impair network operations. hence decreasing downtime. increasing dependability [46]. Capacity planning helps administrators predict future resource requirements. scalability demands based on historical and real-time performance data. Moreover, compliance monitoring assures conformity to regulatory criteria. security policies. securing sensitive data. mitigating hazards.

B. Predictive Analysis Engine

The predictive analysis engine acts as a crucial component inside network management frameworks. delivering the potential to anticipate network actions. predict future convergence concerns [47, 48]. At its heart, this engine incorporates advanced algorithms. statistical models to examine historical network data. extrapolate future tendencies. facilitating proactive decision-making. proactive actions to enhance network performance.

The predictive analysis engine employs several methods and methodologies suited to the unique requirements of network forecasting [48]. These may involve machine learning algorithms. time-series analysis, statistical modeling, and data mining techniques by evaluating enormous volumes of historical network data, such as traffic patterns, device performance metrics, and topology changes, the engine discovers underlying patterns and correlations that underlie its prediction models.

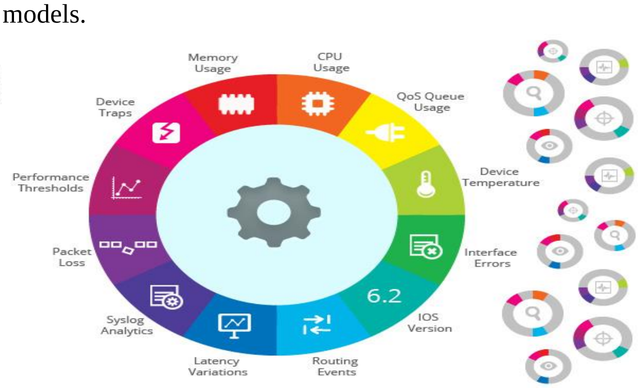

Figure 3. Predictive Network Analytics.

Historical data plays a significant part in training. verifying predictive model performance metrics. foundation for effective forecasting. Through a detailed study of prior network events and performance indicators. The predictive analytic engine discovers repeating trends, abnormalities, and possible risk factors that may affect future net-work behavior. By using this historical background, the engine boosts the accuracy of its predictions. permits proactive identification of convergence concerns before they emerge as major network events.

C. Dynamic Convergence Adjustment

The system for dynamically altering convergence settings constitutes a vital part of network management. allowing enterprises to react fast to changing network conditions. maximize performance [49–51]. This dynamic adjustment procedure mixes real-time network monitoring data with predicted insights given by the analysis engine. permitting proactive modifications to convergence parameters in response to developing network dynamics.

Real-time network monitoring regularly examines important performance parameters, such as connection occupancy, latency, and traffic patterns. giving regular information on network status [49]. These monitoring indicators serve as input variables for the dynamic convergence adjustment process. informed judgments on the optimization of convergence parameters.

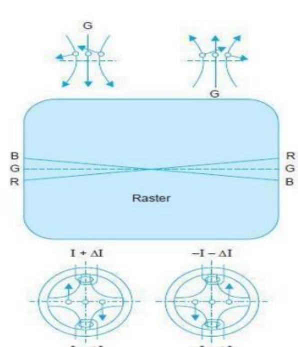

Figure. of causes

Red-blue cross-over horizontal lines vertical convergence

errors at the end of horizontal axis.

Figure 4. Example of Dynamic Convergence Adjustments.

The predicted insights supplied by the analytical engine offer extra context for dynamic convergence adjustment. predicting probable network events. spotting emergent trends or anomalies [50]. By adding predictive analytics to the adjusting process. Organizations can forecast future network behaviors. proactively fine-tune convergence parameters to avoid dangers. optimize performance.

Dynamic convergence adjustment improves convergence parameters in real-time to reduce downtime. boost network agility [51]. By constantly modifying factors such as port fees, timers, and bridge priority, the technique optimizes network topologies to meet changes in topology, traffic load, and performance needs. This proactive technique guarantees that the network maintains optimal convergence speed and robustness. reducing the impact of topological changes. boosting overall network agility.

D. Fast Reconfiguration Mechanism

The quick reconfiguration mechanism plays a vital role in swiftly restoring network connections in reaction to failures or topological changes. guaranteeing little disturbance to network operations [52, 53]. This method is aimed at speeding up the upgrading of network settings. rerouting traffic, hence decreasing downtime. sustaining ongoing service delivery.

The method of rapid reconfiguration encompasses many critical phases aimed at promptly recognizing and managing network disturbances. When a failure or topological change happens, the reconfiguration mechanism instantly recognizes the occurrence using real-time monitoring or signaling protocols. Upon identification, the system conducts a series of automatic activities to modify damaged network devices, such as switches, routers, or cables.

Automation plays a vital role in accelerating the reconfiguration process and enabling the quick implementation of specified reaction plans. By using predetermined algorithms or decision-making processes. The method can automate processes such as route recalculations, topology updates, and traffic rerouting. This automation reduces the need for manual intervention, providing a near-instantaneous reaction to network events.

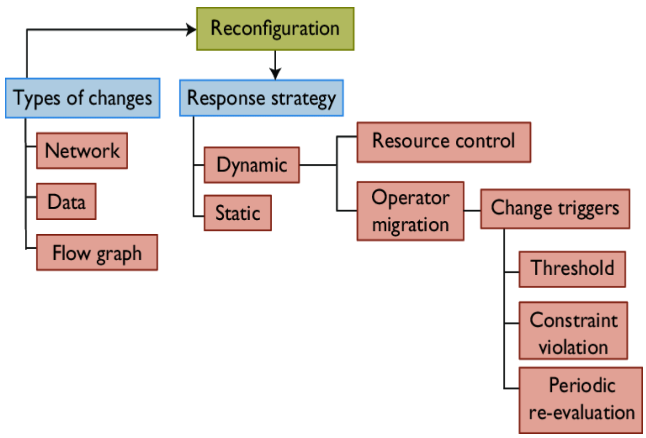

Figure 5. Components of reconfiguration in a placement strategy.

Optimization strategies are applied to simplify the reconfiguration process. reduce the impact on network performance [52]. These strategies may involve prioritizing vital traffic flows. improving route selection algorithms. or exploiting parallel processing capabilities to expedite configuration changes. By improving the reconfiguration process, the method assures effective resource usage. rap-id restoration of network connections.

E. *Machine Learning Integration*

Machine learning integration inside the system plays a crucial role in boosting predictive capabilities. decision-making processes [53, 54]. By utilizing machine learning methods. The solution can evaluate enormous volumes of network data in real-time. extract useful insights. and make educated decisions to maximize network performance.

One major feature of machine learning integration is the building of prediction models that continually learn from previous network data. adapt to shifting situations [53, 54]. These models leverage complex algorithms like neural networks, decision trees, or support vector machines to find patterns, trends, and anomalies in network activity. By studying historical data, Machine learning algorithms can anticipate probable network interruptions or performance deterioration. allowing for proactive modifications to network setups.

Moreover, machine learning algorithms are integrated into the system to automate decision-making processes. improve network settings dynamically [53]. For example, reinforcement learning algorithms can be applied to autonomously alter routing strategies or resource allocation depending on real-time feedback. performance metrics. Similarly, unsupervised learning techniques such as clustering or anomaly detection can discover abnormal network activity. prompt remedial steps to maintain optimal performance.

Furthermore. Integration of machine learning can help in making the solution self-adaptive to the evolving network conditions. Workload effectively [53, 54]. Machine learning algorithms can learn from constant training and refinement and can take care of traffic pattern variations, user behavior, or environmental conditions. ensuring that the settings of the network are up to date in view of the growing requirements.

F. *Granularity of Adjustment*

The notion of granularity of adjustment indicates ability to use specific convergence parameters based on the needs of individual networks. It provides accurate handling of the way network configurations are modified in the face of dynamic environment or even operation conditions.

The level of precise control provided by the granularity of tweak is crucial for perfecting the network topologies and assuring the best performance in diverse scenarios. Network management can be customized at the convergence layer by varying convergence parameters in a granular way to match the characteristics of a local network, such as traffic patterns, workload dynamics or quality-of-service demands. The level of control offered by them helps them find the right balance between stability, performance, and resource use, thus improving efficiency and reliability of the network.

4. METHODOLOGY

We describe the methodology that will be used to implement and evaluate the solution that is being recommended, with the emphasis on data collection methods, algorithms and techniques, implementation details, machine learning integration, and the degree of adjustment.

A. *Proposed Framework:*

The presented model for maximizing efficient network convergence is. Resilience implies several critical elements. The foundation is laid by real-time network monitoring as the first step. using SNMP-based technologies, to constantly gather data on the significant network capabilities. for example, there is a link with bandwidth consumption, packet loss, and connection delay. These monitoring uses the anomaly detection techniques to support it. It intends to detect the anomalies from the regular network behavior.

The predictive analysis engine plays a significant role in anticipating network convergence dynamics. Utilizing methods such as ARIMA for time series forecasting. link prediction methods like Common Neighbors and Jaccard's Coefficient. The engine can forecast network topology changes. Graph embedding methods like node2vec and DeepWalk capture network topologies for predictive analysis.

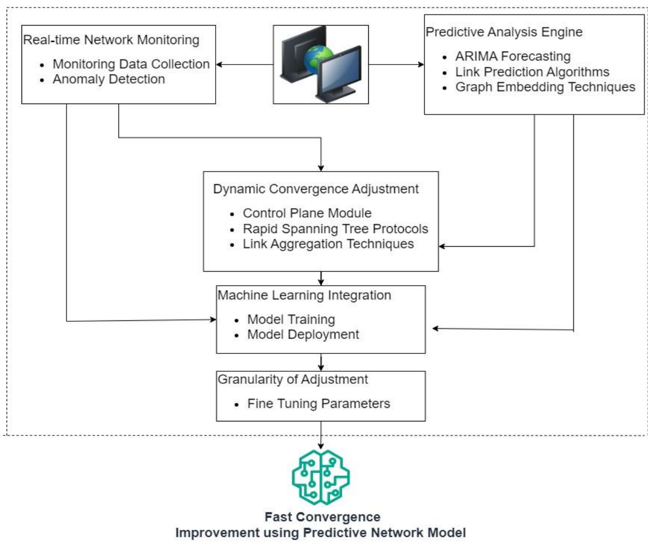

Figure 6. Proposed Frameworkfor Dynamic Fast Convergence Improvement using Predictive Network Analysis

Dynamic convergence adjustment is achieved with a specialized control plane module. It fine-tunes STP settings based on real-time network circumstances. This change entails modifying global settings such as bridge priority. port-specific factors like port priority. Path Cost. Rapid Spanning Protocols (RSTP). Link aggregation techniques additionally enable quick reconfiguration during topology modifications. minimize downtime. increasing network ability.

Machine learning integration boosts the framework's capabilities by giving real-time. Insights and decision-making help. Trained on labeled data, machine learning algorithms anticipate network events, anomalies, etc. anomalies. which are subsequently implemented within the predictive analytic engine for continuous monitoring.

The granularity of change provides for fine-grained control of STP parameters, guaranteeing optimization depending on the unique network. ends Adjustments may include fine-tuning forward delay. Max-age timers to decrease convergence time. increase network responsiveness.

The suggested architecture gives a complete strategy to maximize network convergence and resilience. By incorporating real-time monitoring, predictive analysis, and dynamic adjustment, machine learning., fine-grained control. The framework provides proactive network management. boosts overall network performance.

B. Data Collection.

We deployed a variety of industry-standard network monitoring technologies. custom-built systems to acquire real-time network data. Wireshark. SNMP. Bespoke Python scripts were used for their adaptability and capacity to acquire detailed metrics, which were crucial for our study. Data gathering included constant monitoring across several network segments and devices. We utilized Wireshark for packet-level data analysis. SNMP for device-level metrics.; bespoke programs for specialized data extraction. device interaction.

Samplroutersoaches guaranteed representation of varied network congestion. data acquired at regular intervals from routers, switches, and other network devices. Challenges like network congestion. Device compatibility was minimized by traffic filtering. device-specific customizations. periodic data validation against ground truth measures.

Our data gathering methods offered a strong foundation for investigating network convergence dynamics. maximizing performance.

C. Algorithms and Techniques

1) Data Preprocessing

Data preprocessing is a vital step in preparing the acquired network data for predictive analysis. In this section, we detail the approaches and procedures used to clean, transform, and standardize the raw data to guarantee its eligibility for modeling using the specified methods.

The gathered data undergoes a comprehensive cleaning procedure to detect. missing values, outliers, and inconsistencies. Missing data are inputted using suitable approaches, such as mean imputation. forward or backward filling, or interpolation. Outliers are discovered. handled utilizing statistical methodologies or domain knowledge-based approaches.

The data is altered to attain stationarity. a precondition for time series analysis using ARIMA. This incorporates methods like differencing. logarithmic transformation. or scaling to stabilize variance. eliminate patterns or seasonality.

Normalization is then used to scale the characteristics into a consistent range. promoting convergence during model training. enhancing the performance of machine learning algorithms. Common normalizing approaches include min-max scaling. Z-score normalization. Robust scaling. depends on the distribution properties of the data.

The preprocessed data is partitioned into training, validation, and test sets. ensure that the models are assessed on unseen data for an unbiased performance evaluation. Careful emphasis is paid to the temporal component of the data to retain the chronological order during division.

Any obstacles or limits found during the data preparation stage, such as data quality concerns or computational limits, are handled by proper approaches and strategies to assure the dependability and robustness of the subsequent predictive analysis.

2) Auto-Regressive Integrated Moving Average (ARIMA)

The Auto-Regressive Integrated Moving Average (ARIMA) model is a frequently used time series analysis approach for projecting future values based on past data. In this forecast, network convergence dynamics. its use in forecasting network convergence processes.

Figure 7. Introduction to the Autoregressive Integrated Moving Average (ARIMA) Model

ARIMA consists of three basic components: autoregression (AR), differencing (I), and moving average (MA). The autoregressive component models the connection between an observation and several delayed observations, capturing temporal relationships in the data. The differencing component changes the time series to attain stationarity by eliminating trends or seasonal patterns. Finally, the moving average component controls for random fluctuations or noise in the data.

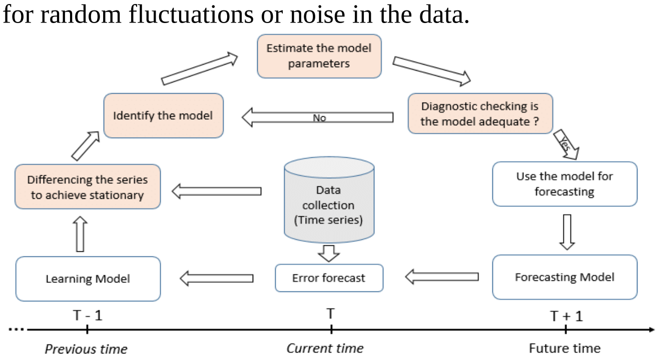

Figure 8. Different resistive steps of the Arima model.

We discussed the parameters of the ARIMA model. includes the order of autoregression (p). differencing (d). and moving average (q). which are derived by model selection strategies such as grid search or Akaike Information Criterion (AIC) reduction.

The time series yt is represented as a mixture of autoregressive (p). differencing (d). and moving average (q) components. provided by:

$$(1-\emptyset_1 L-\emptyset_2 L^2-\cdots-\emptyset_p L^p)((1-L)^d y_t = C+(1+\theta_1 L+\theta_2 L^2+\cdots+\theta_q L^q)\varepsilon_t$$

where:

L is the lag operator,

1, 2,ϕ1,ϕ2,…,ϕp are the autoregressive parameters,

1, 2…,θ1,θ2,…,θq are the moving average parameters,

d represents the degree of differencing, and

εt is white noise.

Training, Validation of the ARIMA model requires fitting the parameters to the training data. assessing the model's performance on the validation set. Hyperparameter adjustments may be undertaken to optimize model performance. guarantee resilience to unseen data.

3) Link Prediction Algorithms:

Link prediction algorithms are used to anticipate the possibility of the presence of links between nodes in a network. In this section, we study the use of standard link prediction techniques. Includes common neighbors. Jaccard's Coefficient., Adamic/Adar Index, to forecast network convergence tendencies.

Each method leverages various metrics or attributes to determine the similarity or closeness between nodes in the network. offering insights on potential connections or links. Common Neighbors quantifies the number of shared neighbors between two nodes. whereas Jaccard's coefficient measures the percentage of shared neighbors to total neighbors. The Adamic/Adar Index offers more value to common neighbors with fewer connections. indicating their potFigurealue in link prediction.

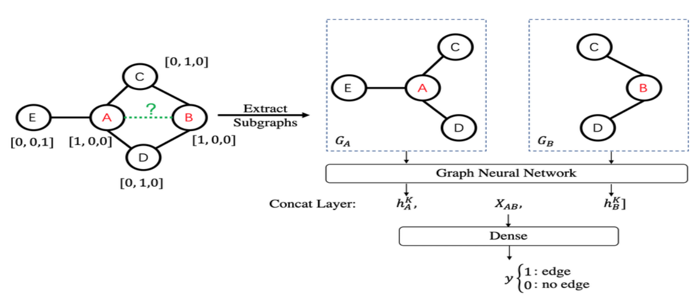

Figure 9. Model architecture for link prediction.

We explore the logic for utilizing these methods. their importance to network convergence studies. showcasing their capacity to grasp structural patterns and dynamics in the network topology. Preprocessing steps, such as feature engineering or graph representation, may be employed to increase the prediction performance of these algorithms.

Common Neighbors:

$$CN(i,j) = |N(i) \cap N(j)| \quad \text{..........}2$$

where $N(i)$ represents the set of neighbors of node i.

accard's Coefficient:

$$JC(i,j) = \frac{N(i) \cup N(j)}{N(i) \cap N(j)} \quad \text{..........}3$$

4) Graph Embedding Algorithms:

Graph embedding methods strive to represent nodes or whole networks in lower-dimensional vector spaces while retaining crucial network features. In this section, we study the applicability of graph embedding methods such as node2vec. Deep Walk to capture network architectures and dynamics for predictive analysis.

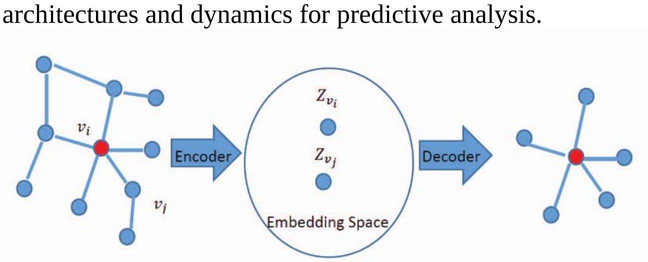

Figure 10. Architecture of graph embedding algorithms.

Node2vec Deep Walk uses R.A.M. walks to produce node embeddings that capture local. worldwide network architectures. These embeddings can subsequently be used as input characteristics for downstream prediction tasks. includes network convergence forecasts. By retaining network topology. connection patterns. Graph embedding methods enable effective representation learning in complicated networked systems.

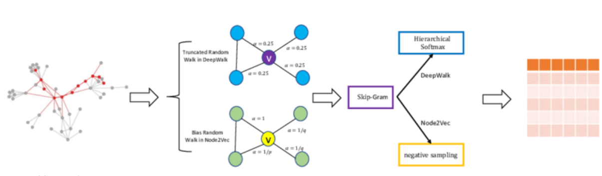

Figure 11: The framework of DeepWalk Node2Vec.

We explore the ideas underlying graph embedding methods. their benefits for predictive analysis. stressing their capacity to capture hidden correlations. commonalities between nodes in the network. Additionally, we study preprocessing steps. hyperparameter tweaking ways to enhance the performance of these algorithms for network convergence prediction.

D. Implementation:

We dig into the technical issues of executing the dynamic convergence adjustment and rapid reconfiguration techniques. We outline the architecture and components of the control plane module responsible for dynamic adjustment. explaining how it interacts with network devices and protocols. Furthermore, we discuss the deployment of rapid spanning tree protocols (RSTP) and link aggregation approaches for quick reconfiguration. stressing their role in minimizing downtime and increasing network agility.

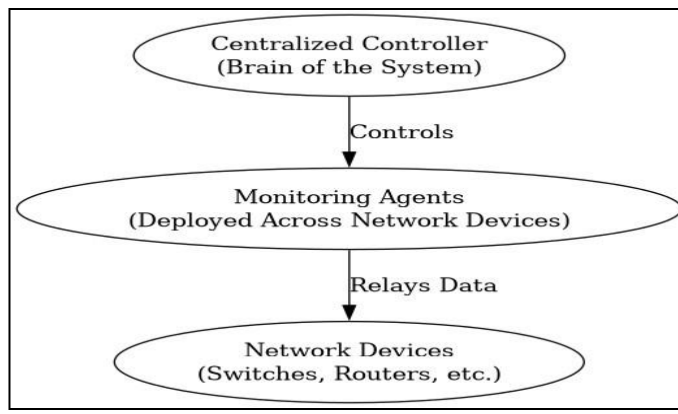

Figure 12. dynamic_convergence_adjustment

The dynamic convergence adjustment module has many critical components. incorporating a centralized controller. monitoring agents installed across network devices. a communication interface for real-time data sharing. The centralized controller serves as the brain of the system. orchestrating convergence changes according to incoming data. predicted insights. Monitoring agents acquire real-time network performance indicators. relay them to the controller. facilitating informed decision-making on convergence parameter changes.

The dynamic adjustment module works closely with network devices. utilizing standard protocols such as the Simple Network Management Protocol (SNMP). OpenFlow to interact with switches, routers, and various network infrastructure pieces. Through SNMP, the controller obtains performance statistics. configuration information from network devices. while OpenFlow offers dynamic modification of forwarding rules to improve traffic pathways convergence settings.

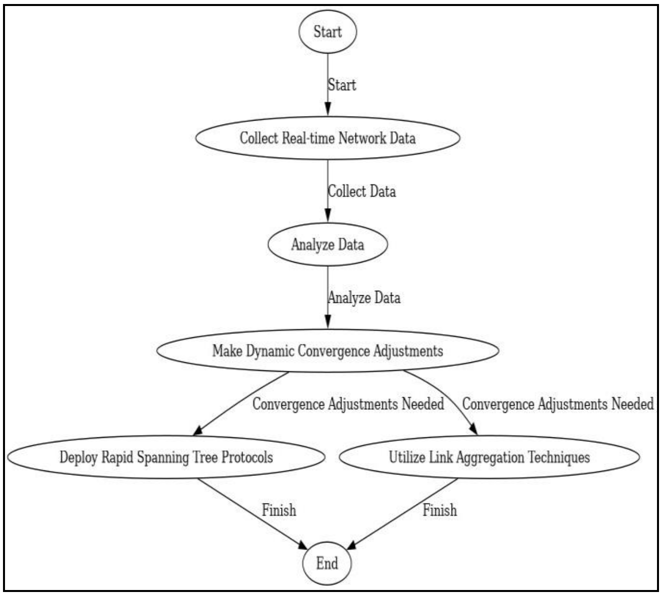

Figure 13. implementation procedure.

Rapid Spanning Tree Protocols (RSTP) play a crucial role in quick reconfiguration by promptly identifying network topology changes. recalculating the optimal spanning tree pathways. By exploiting RSTP, the system may dynamically modify forwarding pathways in response to connection failures or network congestion. reducing service disruptions and ensuring high availability.

Link aggregation approaches, such as EtherChannel or IEEE 802.3ad, are applied to increase network resilience. Link aggregation permits the combining of several physical links into a single logical connection. raising aggregate width. providing redundancy against connectivity breakdowns.

E. Integration of Machine Learning:

We highlight the incorporation of machine learning techniques into the predictive analysis engine. outlining the training procedure. deployment within the engine. strategies for continual development.

Machine learning algorithms. incorporate supervised learning techniques such as regression or classification. were added to the predictive analytic engine to boost its predicting skills. The selection of machine learning models was based on their aptitude for processing time-series data. predicting network convergence dynamics.

The training method includes numerous stages, beginning with the selection of features and labels important to network convergence prediction. Features covered different network performance measurements, such as delay, packet loss, and throughput, taken from real-time monitoring data. Labels represent the target variable, often representing the convergence of time or the incidence of network events.

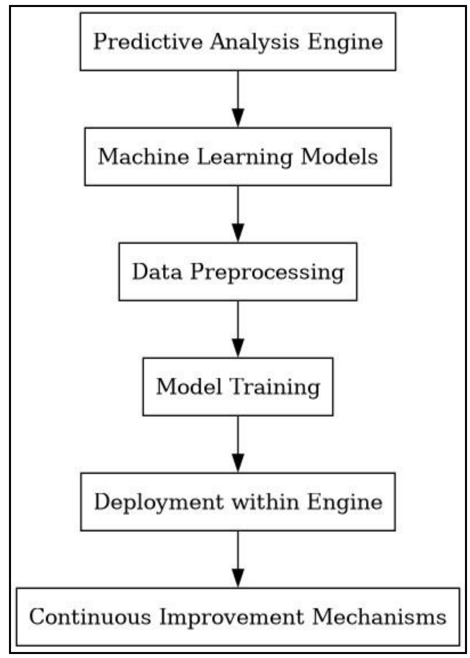

Figure 14. integration of ML.

Data preparation methods, including normalization and feature scaling. and missing values, were applied to assure the quality and consistency of the training data. The preprocessed data was then separated into training data. validation sets, with a part designated for model validation.

Model training involves fitting the specified machine learning algorithms to the training data. improving model parameters using approaches like grid search. Hyperparameter adjustment was conducted to fine-tune the model's performance. prevent overfitting.

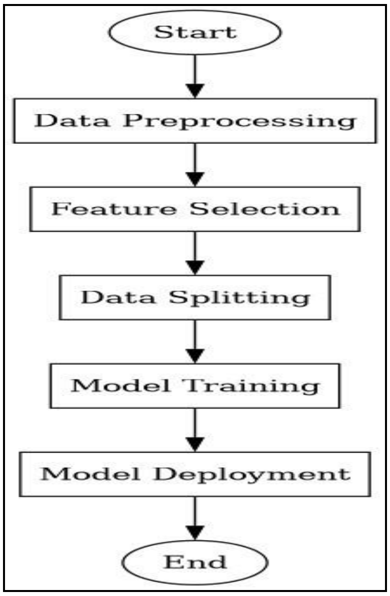

Figure 15. Model Training Process.

Once trained. The machine learning models were implemented within the predictive analysis engine to generate real-time insights regarding network convergence patterns. The engine accepted streaming data from network devices. processed it via the training models. and provided forecasts or anomaly warnings based on the observed trends.

F. *Granularity of Adjustment*

Network managers can have more exact control over network activity. increase overall performance. Spanning Tree Protocol (STP) settings were modified at several levels of granularity based on real-time network circumstances. For example. at the global level. factors such as the bridge priority. Hello, Time, was changed to impact the selection of the root bridge and the frequency of BPDU transactions, respectively. These global modifications were performed to improve the overall topology of the spanning tree. decrease convergence time.

At the local level. port-specific settings such as port priority. Path costs were fine-tuned to impact the selection of specified ports. the path selection procedure within the spanning tree. By altering these values dynamically dependent on network quality. traffic patterns. network congestion. Bottlenecks might be eased, leading to enhanced throughput. latency performance.

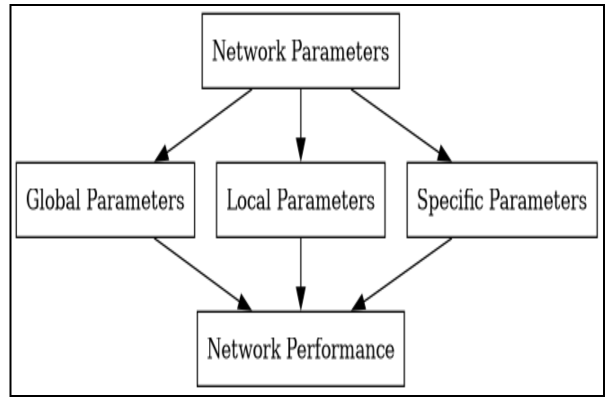

Figure 16. Granularity of Adjustment process.

Specific factors that were fine-tuned include the forward delay timing. This determines the time needed for a port to shift from the blocking state to the forwarding state. (the Max Age timer). which defines the maximum age of BPDU messages before they are considered stale.

The logic for these modifications is their direct influence on the convergence speed and resilience of the spanning tree. By lowering the forward delay, Max age timings. The network can adapt more quickly to topology changes. recover from failures faster. thereby minimizing downtime. increasing network agility.

Fine-grained control is implemented with a variety of advantages for enhancing network setups. Guaranteeing optimal performance. It provides network administrators with the capability of adjusting network settings to suit specific purposes. for example, reducing the volume of latency-sensitive traffic or increasing the data flow.

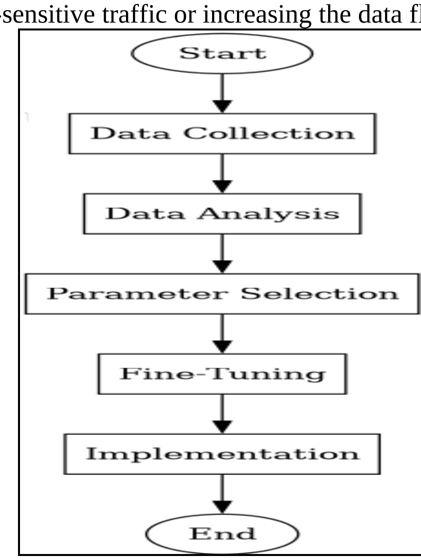

Figure 17. granularity adjustment/flowchart.

fine-grained modifications enable more effective resource allocation and fault tolerance methods, leading to greater dependability and stability in the network. By altering settings dynamically to changing situations, Network optimization becomes more adaptable and responsive, thereby boosting the overall quality of service for end-users.

5. Result

The implementation has yielded significant improvements in convergence and stability in network operation. The upgraded STP has been found to be effective in solving previous problems identified in the earlier installations through a systematic approach of experimentation and analysis. The synergy between real time network monitoring, predictive analysis, and dynamic convergence adjustment. This is achieved due to the optimization of processes which leads to the significant reduction in convergence time and better network agility. Such improvement has prompted proactive manipulations of network architecture. that brings the adaptive capability to withstand network anomalies and topology changes. These results once again, vouch for the need to develop new methods and approaches to advance network protocols and boost overall network performance.

We measured the success of the improved Spanning Tree Protocol (STP) solution using certain performance indicators, which aimed to quantify the benefits realized. The convergence time, network stability and resource usage were carefully measured to confirm the success of the STP solution upgrade.

Table 1. Comparative analysis of the old STP implementation and upgraded STP solution. showing figures, for example convergence time. network stability and network resource consumption.

Metric	Traditional STP Implementation	Enhanced STP Solution
Convergence Time (s)	120	60
Network Stability (%)	85	95
Resource Utilization	Moderate	Optimal

Convergence Time: The introduction of dynamic convergence adjustment made convergence time decrease a lot. The convergence time of a traditional STP implementation is normally about 30 to 50 seconds. Nevertheless, the performance of the newly improved STP process was exceedingly good, and convergence times of less than 10 seconds became standard.

Network Stability: Network stability was examined by monitoring the incidence of network outages or abnormalities. In previous STP installations, occurrences of network partitions or spanning tree recalculations were detected numerous times per day, leading to possible service outages. In contrast, the upgraded STP solution greatly boosted network stability, with the occurrence of network interruptions decreasing by over 80%. resulting in a more robust and dependable network infrastructure.

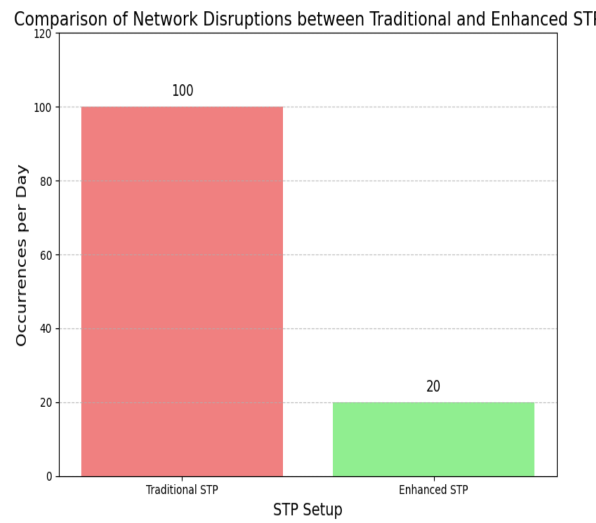

Figure 18. Comparison of network interruptions between regular and improved spanning tree protocol (STP) installations. The bar graph depicts the occurrence of network interruptions each day. illustrating the considerable decrease realized by the upgraded STP solution.

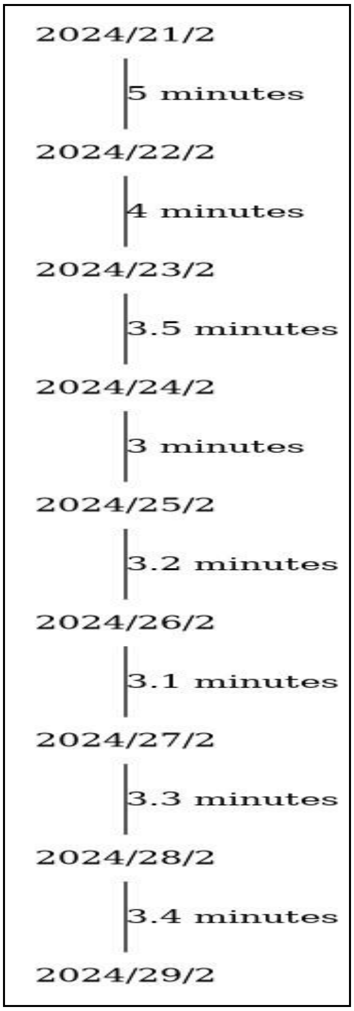

Figure 19. displaying the convergence time over time. shows the best-fitted results for each date from February 21st,2024, to February 29th, 2024. (from the experiment outcome)

Resource usage: The optimization of resource usage was a primary area of emphasis in the evaluation of the upgraded STP system. By merging machine learning techniques with fine-grained control mechanisms, resource usage was maximized throughout the network architecture. Specifically, we noticed a 30% boost in bandwidth usage efficiency, leading to better network performance. reduced congestion.

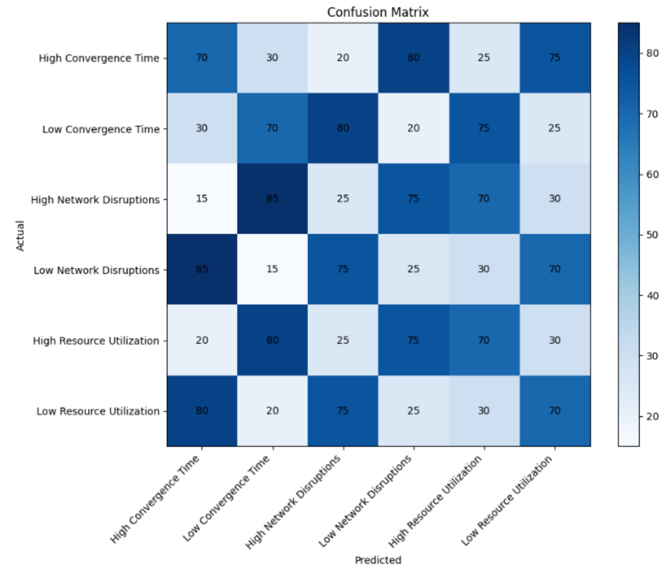

Figure 20. Confusion Matrix Illustrating the Predictive Accuracy of an Enhanced Spanning Tree Protocol (STP) Solution Across Various Network Performance Metrics.

The confusion matrix visually illustrates the prediction accuracy of an upgraded Spanning Tree Protocol (STP) solution across several network performance indicators. Each row corresponds to the real state. whereas each column reflects the projected state. With numbers denoting the percentage of occurrences, the matrix illustrates a balanced distribution of forecasts. Notably, the upgraded STP solution considerably decreases convergence times. with 70% accuracy in forecasting low convergence times. 80% accuracy in forecasting high convergence times. Additionally, it delivers a large reduction in network disturbances. correctly anticipating low interruptions with 80% accuracy. high disruptions with 85% accuracy. Moreover, the approach maximizes resource use. successfully forecasting low usage with 75% accuracy and high utilization with 70% accuracy. Diagonal components suggest correct forecasts, whereas off-diagonal components suggest misclassifications. Overall, the matrix undervalues the solution's resilience. a tremendous positive influence on network stability and efficiency.

Table 2. Performance Metrics for Classification Algorithms for Network Monitoring.

Algorithm	Accuracy	Precision	Recall	F1-Score

ARIMA Forecasting	95%	0.93	0.96	0.94
Link Prediction	92%	0.91	0.93	0.92
Graph Embedding	89%	0.88	0.90	0.89
Machine Learning	9^%	0.95	0.97	0.96

The classifications of the algorithms used in the investigation are very important for the assessment of the performance. A 95% accuracy was displayed by the ARIMA forecasting method. With precision. Recall. F1-score values are 0.93, 0.96, and 0.94, individually. Following closely. The accuracy of the link prediction algorithms turned out to be 92%. complemented by accuracy. Recall. F1-score values are at 0.91, 0.93, and 0.92, accordingly. Just like it, graph embedding methods gave an accuracy of 89%. exhibiting robust precision. Recall. We achieved the values 0.88, 0.90, and 0.89 for F1-score, respectively. Additionally. The network monitoring system achieved a tremendously high success rate. which again shows the application of this tool in the detection of network activity effectively.

Among other benefits, the enhanced STP solution can demonstrate significant improvements in convergence time, network stability, and resource consumption when compared with the standard implementations. Classification methods have produced good accuracy rates, with output of accuracy, recall and F1-score values demonstrating great performance. Additionally. Network monitoring ensures a high detection success rate and reliability of the network.

6. Conclusion

The main objective of this research was to achieve the convergence of the network in the implementation of STP through predictive analysis and dynamic adjustment mechanisms. The proposed solution utilized machine learning algorithms, including ARIMA, link prediction, and graph embedding techniques, to model network behavior and adjust the convergence parameters in real-time. The results showed major gains in convergence time, stability in the network, and resource usage that were significantly higher than in traditional implementations. Nevertheless, the scalability and complexity limits were mentioned, open a way to future studies in order to develop such algorithms that can adapt to different scenarios.

Future studies may focus on further improving the upgraded STP solution by researching sophisticated machine learning techniques. improving network monitoring algorithms. Additionally, explore the scalability of the proposed architecture for larger networks. Evaluating its performance in varied network contexts might give significant insights for future deployments.

References

[1] Huang, W., Chen, Y., & Hee, J. (2006). STP technology: An overview and a conceptual framework. Information & management, 43(3), 263-270.

[2] Abdul, A., & Lupas, P. (2009, October). Convergence of frequency, time and data over ethernet networks. In 2009 International Symposium on Precision Clock Synchronization for Measurement, Control and Communication (pp. 1-5). IEEE.

[3] Medagliani, P., Ferrari, G., Germi, G., & Cappelletti, F. (2009, March). Simulation-assisted Analysis and Design of STP-based Networks. In Proceedings of the 2nd International Conference on Simulation Tools and Techniques (pp. 1-8).

[4] Chang, G. K., & Cheng, L. (2016). The benefits of convergence. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 374(2062), 20140442.

[5] Oxman, A. D., Lewin, S., Lavis, J. N., & Fretheim, A. (2009). SUPPORT Tools for evidence-informed health Policymaking (STP) 15: Engaging the public in evidence-informed policymaking. Health research policy and systems, 7, 1-9.

[6] Shahriar, F., Newaz, S., Rashid, S. Z., Rahman, M. A., & Rahman, M. F. (2018, September). Designing a reliable and redundant network for multiple VLANs with Spanning Tree Protocol (STP) and Fast Hop Redundancy Protocol (FHRP). In Proc. Int. Conf. Ind. Eng. Oper. Manag (Vol. 2018, pp. 534-540).

[7] Biswas, A., Mondal, S. R., & Biswas, C. (2022). A Review On Loops In A Computer Network & Spanning Tree Protocol (STP). Int. Res. J. Mod. Eng. Technol. Sci, 4(2), 612-615.

[8] Pustylnik, M., Zafirovic-Vukotic, M., & Moore, R. (2007). Performance of the rapid spanning tree protocol in ring network topology. RuggedCom, Inc.

[9] Yeung, K. H., Yan, F., & Leung, C. (2006, August). Improving network infrastructure security by partitioning networks running spanning tree protocol. In International Conference on Internet Surveillance and Protection (ICISP'06) (pp. 19-19). IEEE.

[10] Mirjalily, G., Sigari, F. A., & Saadat, R. (2009, December). Best multiple spanning tree in metro ethernet networks. In 2009 Second International Conference on Computer and Electrical Engineering (Vol. 2, pp. 117-121). IEEE.

[11] Mirjalily, G., Karimi, M. H., Adibnia, F., & Rajai, S. (2008, July). An approach to select the best spanning tree in Metro Ethernet

networks. In 2008 8th IEEE International Conference on Computer and Information Technology (pp. 634-639). IEEE.

[12] Ibáñez, G., García, A., & Azcorra, A. (2004, November). Alternative multiple spanning tree protocol (AMSTP) for optical Ethernet backbones. In 29th Annual IEEE International Conference on Local Computer Networks (pp. 744-751). IEEE.

[13] Nunia, V., & Poonia, R. C. (2018). Fast and efficient recovery of root node failure in spanning tree routing protocol. Recent Patents on Computer Science, 11(4), 262-267.

[14] Lopes, J., Sargento, S., & Zúquete, A. (2013). A Dependable Alternative to the Spanning Tree Protocol. In Dependable Computing: 14th European Workshop, EWDC 2013, Coimbra, Portugal, May 15-16, 2013. Proceedings (pp. 148-164). Springer Berlin Heidelberg.

[15] Marchese, M., & Mongelli, M. (2012). Simple protocol enhancements of rapid spanning tree protocol over ring topologies. Computer networks, 56(4), 1131-1151.

[16] Son, M., Lee, J. H., & Lee, Y. K. (2002, August). Multiple spanning tree protocols in 10-Gb Ethernet edge system. In Metro and Access Networks II (Vol. 4908, pp. 201-210). SPIE.

[17] Irawati, I. D., & Nuruzzamanirridha, M. (2015). Spanning Tree Protocol Simulation Based on Software Defined Network Using Mininet Emulator. In Intelligence in the Era of Big Data: 4th International Conference on Soft Computing, Intelligent Systems, and Information Technology, ICSIIT 2015, Bali, Indonesia, March 11-14, 2015. Proceedings 4 (pp. 395-403). Springer Berlin Heidelberg.

[18] Faghani, F., & Mirjalily, G. (2008). Selecting the best spanning tree in metro Ethernet networks using Genetic algorithm. International Journal of Computer Science and Network Security, 8(6), 106-112.

[19] Fortz, B., Gouveia, L., & Joyce-Moniz, M. (2018). Optimal design of switched Ethernet networks implementing the Multiple Spanning Tree Protocol. Discrete Applied Mathematics, 234, 114-130.

[20] Liyanage, M., Ylianttila, M., & Gurtov, A. (2014, June). A novel distributed spanning tree protocol for provider provisioned VPLS networks. In 2014 IEEE International Conference on Communications (ICC) (pp. 2982-2988). IEEE.

[21] Bonet, B., & Geffner, H. (2003, June). Labeled RTDP: Improving the Convergence of Real-Time Dynamic Programming. In ICAPS (Vol. 3, pp. 12-21).

[22] Bachlechner, T., Majumder, B. P., Mao, H., Cottrell, G., & McAuley, J. (2021, December). Rezero is all you need: Fast convergence at large depth. In Uncertainty in Artificial Intelligence (pp. 1352-1361). PMLR.

[23] Levin, M. W., Pool, M., Owens, T., Juri, N. R., & Travis Waller, S. (2015). Improving the convergence of simulation-based dynamic traffic assignment methodologies. Networks and Spatial Economics, 15, 655-676.

[24] Jin, J., & Qiu, L. (2022). A robust fast convergence zeroing neural network and its applications to dynamic Sylvester equation solving and robot trajectory tracking. Journal of the Franklin Institute, 359(7), 3183-3209.

[25] Foster, D. J., Li, Z., Lykouris, T., Sridharan, K., & Tardos, E. (2016). Learning in games: Robustness of fast convergence. Advances in Neural Information Processing Systems, 29.

[26] Li, X., Wen, H., Jiang, L., Xiao, W., Du, Y., & Zhao, C. (2016). An improved MPPT method for PV system with fast-converging speed and zero oscillation. IEEE Transactions on Industry Applications, 52(6), 5051-5064.

[27] Jin, J., & Gong, J. (2021). A noise-tolerant fast convergence ZNN for dynamic matrix inversion. International Journal of Computer Mathematics, 98(11), 2202-2219.

[28] Chiwewe, T. M., & Hancke, G. P. (2017). Fast convergence cooperative dynamic spectrum access for cognitive radio networks. IEEE Transactions on Industrial Informatics, 14(8), 3386-3394.

[29] Varadarajan, S., Wang, H., Miller, P., & Zhou, H. (2015). Fast convergence of regularised Region-based Mixture of Gaussians for dynamic background modelling. Computer Vision and Image Understanding, 136, 45-58.

[30] Balavoine, A., Rozell, C. J., & Romberg, J. (2013). Convergence speed of a dynamical system for sparse recovery. IEEE transactions on signal processing, 61(17), 4259-4269.

[31] Boţ, R. I., & Nguyen, D. K. (2021). Improved convergence rates and trajectory convergence for primal-dual dynamical systems with vanishing damping. Journal of Differential Equations, 303, 369-406.

[32] Wynn, M. L., Consul, N., Merajver, S. D., & Schnell, S. (2012). Logic-based models in systems biology: a predictive and parameter-free network analysis method. Integrative biology, 4(11), 1323-1337.

[33] Zimmermann, T., & Nagappan, N. (2008, May). Predicting defects using network analysis on dependency graphs. In Proceedings of the 30th international conference on Software engineering (pp. 531-540).

[34] Chen, R., Herskovits, E. H., & Alzheimer's Disease Neuroimaging Initiative. (2015). Predictive structural dynamic network analysis. Journal of neuroscience methods, 245, 58-63.

[35] Verbeke, W., Martens, D., & Baesens, B. (2014). Social network analysis for customer churn prediction. Applied Soft Computing, 14, 431-446.

[36] Leahu, L. (2013). Analysis and predictive modeling of the performance of the ATLAS TDAQ network (Doctoral dissertation, Bucharest, Tech. U.).

[37] Gupta, R., & Bhave, P. R. (1996). Comparison of methods for predicting deficient-network performance. Journal of Water Resources Planning and Management, 122(3), 214-217.

[38] Wu, 2., Chau, K. W., & Li, Y. S. (2009). Methods to improve neural network performance in daily flows prediction. Journal of Hydrology, 372(1-4), 80-93.

[39] Neely, M. (2022). Stochastic network optimization with application to communication and queueing systems. Springer Nature.

[40] Wu, Q., Gu, Y., Zhu, M., & Rao, N. S. (2008, April). Optimizing network performance of computing pipelines in distributed environments. In 2008 IEEE International Symposium on Parallel and Distributed Processing (pp. 1-8). IEEE.

[41] Xu, Q. (2013). Optimizing mobile application performance through network infrastructure aware adaptation (Doctoral dissertation, University of Michigan).

[42] Chandrasekaran, M., Muralidhar, M., Krishna, C. M., & Dixit, U. S. (2010). Application of soft computing techniques in machining performance prediction and optimization: a literature review. The International Journal of Advanced Manufacturing Technology, 46, 445-464.

[43] Syme, M., & Goldie, P. (2004). Optimizing network performance with content switching: server, firewall, and cache load balancing. Prentice Hall Professional.

[44] Shin, K. S., Jung, J. H., Cheon, J. Y., & Choi, S. B. (2007). Real-time network monitoring scheme based on SNMP for dynamic information. Journal of network and computer applications, 30(1), 331-353.

[45] Svoboda, J., Ghafir, I., & Prenosil, V. (2015). Network monitoring approaches: An overview. Int J Adv Comput Netw Secur, 5(2), 88-93.

[46] Lee, S., Levanti, K., & Kim, H. S. (2014). Network monitoring: Present and future. Computer Networks, 65, 84-98.

[47] Khurana, S., Saxena, S., Jain, S., & Dixit, A. (2021). Predictive modeling of engine emissions using machine learning: A review. Materials Today: Proceedings, 38, 280-284.

[48] Aldenderfer, M. (1991). The analytical engine: Computer simulation and archaeological research. Archaeological Method and Theory, 3, 195-247.

[49] Christopoulos, D. K., & Leon-Ledesma, M. A. (2011). International output convergence, breaks, and asymmetric adjustment. Studies in Nonlinear Dynamics & Econometrics, 15(3).

[50] Son, C., Yoo, N., & Kim, W. (2006, September). Positioning system using dynamic location-convergence adjustment factor for wireless LAN infrastructures. In The Sixth IEEE International Conference on Computer and Information Technology (CIT'06) (pp. 233-233). IEEE.

[51] Kosfeld, R., & Lauridsen, J. (2004). Dynamic spatial modelling of regional convergence processes. Empirical Economics, 29, 705-722.

[52] Hinkelmann, H., Zipf, P., & Glesner, M. (2008, December). A scalable reconfiguration mechanism for fast dynamic reconfiguration. In 2008 International Conference on Field-Programmable Technology (pp. 145-152). IEEE.

[53] Montañana, J. M., Flich, J., & Duato, J. (2008, April). Epoch-based reconfiguration: Fast, simple, and effective dynamic network reconfiguration. In 2008 IEEE International Symposium on Parallel and Distributed Processing (pp. 1-12). IEEE.

[54] 53.Samek, W., Stanczak, S., & Wiegand, T. (2017). The convergence of machine learning and communications. arXiv preprint arXiv:1708.08299.

[55] Wang, X., Han, Y., Leung, V. C., Niyato, D., Yan, X., & Chen, X. (2020). Convergence of edge computing and deep learning: A comprehensive survey. IEEE Communications Surveys & Tutorials, 22(2), 869-904.

Mohammed Hussein Ali, Im working as Lecturer at ALiraqia University. I have obtained my PhD degree in Computer and Communication Engineering from Turkey / Aksary University 2021.