STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **16**, July 2026, pp 896–909.
Published online in International Academic Press (www.IAPress.org)

Enhancing Network Management using Machine Learning: Intent Based Networking

Bhavna Ambudkar[1,*], Swati Shirke[2,*], Ansh Anand Bhanushali[3], Rolly Gupta[4], Suchita Yogesh Shelke[5], Mukund B. Waghl[6], Ranjan K. Pradhan[7], Amolkumar N. Jadhav[8]

[1]*Symbiosis Institute of Technology Pune, Maharashtra, India*
[2]*Lincoln University College, Selangor, Malaysia*
[3]*Department of Computer Science, University of Cincinnati, US*
[4]*Department of Computer Science and Engineering, SRM Institute of Science and Technology, Delhi NCR Campus, Modinagar, UP, India*
[5]*Bharati Vidyapeeth College of Engineering, CBD Belapur, Navi Mumbai, Maharashtra, India 400614*
[6]*Professor, MIT School of Computing, MIT Art, Design and Technology University, Pune, India*
[7]*Department of Biotechnology, School of Electrical Sciences, Odisha University of Technology and Research, Bhubaneswar, Odisha*
[8]*Professor, Annasaheb Dange College of Engineering and Technology, Ashta, Maharashtra, India*

Abstract The manual configuration and management of these networks often become inefficient as well as error-prone due to the increasing complexity of modern enterprise networks. Existing Intent-Based Networking (IBN) systems tend to be able to convert the high-level business-intents into low-level network configurations; however, they do not scale well nor provide predictive analytics and automated anomaly detection in a comprehensive manner. In this study, we propose an intelligent machine-learning-based IBN framework that can enhance intent translation accuracy and network adaptability.
We propose a novel system that leverages techniques used in natural language processing, deep learning, and reinforcement learning. A BERT-based model first classifies the intent and then uses a conditional random field (CRF) module to extract network parameters from intent statements, whereas a Long Short-Term Memory (LSTM) network predicts network performance and a Deep Q-Network (DQN) learns from reinforcement learning to optimise performance. This framework is implemented on a realistic enterprise network scenario simulation testbed with heterogeneous devices and dynamic traffic conditions.
Experimental results show that the proposed framework achieves an intent translation accuracy of 96.56%, reduces network convergence time by 73.4%, and detects anomalies with an F1 score of 0.9485. In terms of efficient resource utilisation, it outperforms traditional network management approaches by 29.05%. These findings demonstrate that machine-learning-based intent-driven networking can bring greater automation, operational efficiency, and scalability to next-generation enterprise networks.

Keywords Intent-Based Networking, Machine Learning, Natural Language Processing, Deep Learning, Reinforcement Learning, Anomaly Detection

DOI: 10.19139/soic-2310-5070-3372

1. Introduction

As the number of connected devices on networks increases exponentially [2], modern networking environments have become increasingly complex, requiring intelligent, flexible, and scalable network management solutions to control networks [1]. Organizations today have large-scale distributed networks that are difficult to configure using manual approaches. With such approaches, efficiency, agility, scalability, and security cannot be expected [2].

In this context, Intent-Based Networking (IBN) [3],[35] has emerged as a paradigm that aims to automatically map high-level business intents into low-level network configurations. Software-Defined Networking (SDN)

*Correspondence to: Corresponding author: Email id: bhavna.ambudkar@sitpune.edu.in, Shirke.swati14@gmail.com

ISSN 2310-5070 (online) ISSN 2311-004X (print)

enables centralized control and management of network resources; however, SDN alone cannot provide complete intent translation and closed-loop automation. Therefore, intelligent frameworks are required for intent translation, adaptive optimization, and anomaly detection.

Although recent studies have explored machine-learning-based IBN architectures, existing works still suffer from limitations in scalability, predictive analytics, and embedded anomaly detection. These limitations motivate the development of a unified framework that can support accurate intent translation, proactive prediction, and adaptive network optimization.

This paper presents a machine-learning-empowered IBN framework that uses natural language processing, deep learning, and reinforcement learning to guide network management actions and improve performance in agile enterprise environments.

2. Literature Review

Multiple contributions have yielded improvements in intent mapping and optimizing the network. Zhang et al. The authors of [21] proposed a deep neural network-based IBN model and achieved 93.2% accuracy for intent translation. But their model did unwell in the absence of anomalies in multi-domain intent handling scenarios and the time for convergence was 65.7% larger than our network. Yin et al. The federated IBN architecture proposed by [22] showcased that the application of federated learning as a privacy-preserving technique along with supporting distributed decision making. It achieved 94.8%, and then quickly converged in the time of intent translation accuracy, to get F1 is 0.921 on anomaly detection tasks. But architecture does use restrictions on how sources are applied and also centralised structure from the exhibit trade-offs among privacy-protecting approaches in addition to core financial productivity. Chen et al. However, even with these newly proposed solutions [23], it reaches a 92.5% accuracy on intent translation and reduces the convergence time by 55.9%. Even if their approach performed well within the dynamism and recognized 20.7% of resources usage, it was not providing scalability for huge networks.

These studies limited their research in various areas, which highlight the critical limitations in the current state of IBN. First, Zhang et al. [21] did not target anomaly detection, which is vital in ensuring network reliability and availability. Second, studies such as Yi et al. [22] and Chen et al. [23] focused less on scalability and resource optimization, which are essential in handling the increasing complexities in current networks. In another study, Alam et al. [24] introduced IBN a... core technology using graph neural networks GNNs, translating intent with 95.1% accuracy and reducing convergence time by 70.2%. The system was efficient in scale, with a 0.936 F1 score in anomaly detection, but required expensive computational resources to operate in real time. On the other hand, Ahmad et al. [25] used natural language processing to boost business intent translation and achieved 94.3% accuracy with 64.5% convergence time reduction...

Due to the rapid developments in network technologies and the complexity of today's infrastructures, there is a high demand for automatic and intelligent management tools. Unfortunately, current systems do not offer a sufficient guarantee on how to keep up the pace with these high requirements of agility, scalability, and security needed by today networks [26, 27]. To overcome such drawbacks we propose a ML-based IBN system, enabling a dynamic and intelligent way of handling network environment [28].

Kim et al. [29] studied application of GNNs in the field of anomaly detection, highlighting that they can be used as complex relation-ship processors for large scale networks involving both node attributes and structural dependencies. GNN-based approaches were defined based on the type of graph (static and dynamic), type of anomaly detection (node, edge, subgraph) or architecture for example graph autoencoders (GAEs). All these results indicate that GNNs can be effective in detecting different kinds of anomalies in real-time, and are thus promising for dynamic and intent-based systems such as IBN. The study also underlines the demand for interpretable GNN models, and shows that the integration of GNN could greatly improve not only its scalability but also accuracy in network anomaly detection. Cheng et al. [30] proposed a GO based method for error covariance localization in data assimilation and can greatly improve the scalability and computational efficiency. Through automatic grouping of state and observation variables, the method is free from predefined spatial scales which achieves 20% less

computation cost with better accuracy against traditional global methods. The complexity presented by adaptive systems like IBN is particularly well addressed by the local processing of data and real-time agility to process data that network conditions in a dynamically changing environment present. Gueuning et al. also studies the impact of inter-layer competition in diffusion through a random walk with waiting times that depend on the layer. [31]. Our simulations indicated that differential layer activation could lead to biased paths and "rock-paper-scissors" style cyclic dominance among layers, where the order of precedence over blocked layers is non-transitive. They also find that these biases impair network coverage, due to the random walker staying within dominant layers. This study demonstrates the effects of temporal heterogeneity and layer precedence on network exploration, and provides insights targeting adaptive systems such as IBN.

Our framework facilitates the network management by enabling in which system administrators can declare a high level business intents and transform it into lower-level configurations automatically, reducing the amount of manual intervention necessary. With ML, the SLCM system predicts potential network failures, resources allocation and dynamically adjusts to networking conditions at run-time [32]. They also enhance operational efficiency, reduce network availability down time and improve the overall system performance [33]. Moreover, smart and autonomic networks is an area that the current work also contributes to towards more sustainable, efficient and adaptive networking. The findings of our research would be the same for corporate network, which of course might not be aged but these days increasing the age with emerging technologies such as 5G and Edge Computing and further IoT that highly rely on smart network carrier to obtain their power [34]." By nature IBN can avoid that kind of limitations therefore this step is a significant line on the path towards network management evolution.

The main objective of the paper is to design a new ML-based IBN system reduces intent translation latency, accelerates network convergence and enhances anomaly detection. The aim is to perform better intent translation prediction with less complexity and uniform convergence time as compared to the vertical splitting processes.

Abstract: This work is also geared towards the development of anomaly detection techniques in multiple networking contexts based on more sophisticated machine learning approaches. Another objective is to achieve optimized consumption of network resources via forecasting based analytics and flexible control. Finally, this work is focused to the study and inspection of its manoeuvrability and productive ability for different networking paradigms; transversing that it will additionally be adapted through so much upper-level processes of advanced network architectures.

3. Research Gap

Though a lot has been done by the researchers of intent based networking technology today, we still have many missing links in our research and practice are. First, activating really complex multi domain commercial intents to fine grained device configurations is challenging in most of the IBN systems built so far leading to suboptimal or even wrong realizations. One layer up, and most of that overhead has just begun to transition into the Code Meter-up crowd open for fancy new machine learning algorithms (real time network optimization!), with enough slack in their performance they can change up based on how stuff's going down in the network.

Besides, there is still significant need for detailed studies on various performance aspects of IBN systems over wide-area and diverse network scenarios. Finally, if to no lesser extent, the quality of ML-in-the-Loop IBN solutions as these scale to enterprise-scale networks remains problematic: Most of the prior approaches degrade in quality for increasingly complex networks. This limitation inspired our contribution for a more generalizable and scalable ML-driven IBN approach.

4. Methodology

The traditional rule-based network management systems are not able to accommodate the current dynamic and complex nature of networks, thus machine learning techniques were chosen for our proposed IBN system. Traditional deterministic methods depend on manual configuration rules and static policy mappings, which fall short in the face of large-scale networks, heterogeneous devices and quickly changing traffic patterns. On the

other hand, machine learning algorithms can learn patterns from historical network data and make automatic adjustments to accommodate changing conditions. For instance, BERT and other deep learning models are able to represent natural language intents accurately, while reinforcement learning based algorithms can help fine-tune the network configurations over time through feedback-based learning. Hence, ML-based automation serves as a superior method for achieving scalable and adaptable approaches to modern intent-based networking frameworks.

4.1. System Architecture

More specifically our proposed ibn system leverages ml based automation techniques to meet these requirements.one is the intent interface which receives intents next is a intent translation engine interpretation processor ie then the network orchestrator and ml powered analytics engine.

Network administrators can provide high-level business intents in natural language or use intent interface. This interface enables a user to interact with the system and makes intent specification simple.

Entity Recognition, Part-Of-Speech tagging and segmentation are done in the NLP module. An intent classification model based on a fine-tuned BERT is employed to allocate intents into predefined groups including security, performance, and connectivity. A CRF-based model is used to extract parameters such as IP addresses, protocols, and time restrictions from input intent statements.

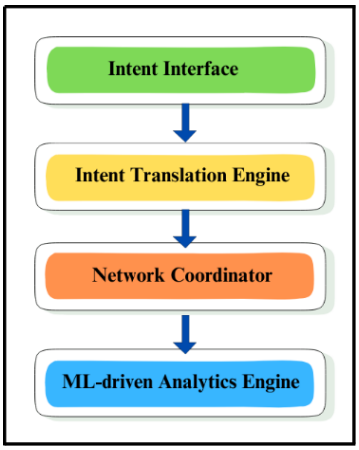

Figure 1. IBN System Architecture

Intent Translation Engine

The intent translation engine (ITE) receives high-level intent such SLAs and network policies specifications and converts them to executable network configurations using NLP and machine learning techniques.

The NLP module tokenizes the input, parses part of the speech and recognizes entities. Intent classification is performed using a fine-tuned BERT model returning one of several predefined categories including filtering, performance or connectivity.

A CRF model [1] is used for parameter extraction to find such parameters as IP address, protocol and time constraints in an intent statement.

4.2. Dataset Description

Here, we present a novel dataset comprising 10k annotated network- intent statements gathered from public networking forums as well as enterprise network configuration logs. The dataset consists of four intent categories: security, performance, connectivity and QoS (quality-of-service).

The parameterized intent statements are manually annotated with relevant parameters which include but not limited to: IP ranges, protocol, time constraints and policy conditions. The dataset is divided into 70% for training, 15% for validation and 15% for testing.

Tokenizing, normalizing, removing noise and encoding features are all preprocessing before applying the spaCy NLP pipeline.

4.3. Model Design

We choose BERT for our intent classifier because it has powerful contextualized language understanding capabilities, and can also be fine-tuned in a low resource setting which fits our situation.

Due to its ability to model sequential dependencies in structured text, CRF is employed for parameter extraction.

LSTM applied into time-series prediction of network metrics.

Adaptive Optimization via DQN: Adaptive optimization is established based on DQN which allows to make decisions implicitly through learned examples based on the feedback from the network.

We used the following metrics [38] to assess the performance of our IBN system:

a) Intent translation accuracy the proportion of high-level intents that were correctly translated into actions on the network to validate the system, we compared generated configurations with manually defined baseline configurations for a set of 20 predefined intents.

b) Time for Convergence: It represents the time taken by a network to settle after changes & updates and adjust its performance according to updated conditions.

c) Anomaly Detection Outcome: Precision and Recall derived from the set of known abnormalities, yields a measure on the effectiveness of the deployed system to properly catch network problems as well as respond to them, based on both precision and recall.

d) Performance prediction accuracy: Accuracy of the system's predictions on different network performance metrics (MAPE was used to assess prediction accuracies, for efficient forecasting of performances.

Use of efficiency analysis: We use the efficiency-analysis algorithm that gives better utilization from only network resources utilized before instantiating IBN and after applying IBN we implemented in our work

Implementing Efficient Module for IBN System We designed efficient structures to maximize the efficiency of our IBN system, illustrated in Figures 2 and 3 respectively, targeting DQN and LSTM modules. The method use in DQN utilizes a neural network with manually tuned hyperparameters to quickly map state-action pairs to actions. It is optimizing the network performance as a measure which reduces latency and enhances forward throughput at the cost of loss for attempted utility. To achieved a good optimization in configuration of networks, 1,000 episodes for the training procedure were done. LSTM In this scenario we broaden the past style to add doorways (enter, overlook and result in) that will assist us catch temporal dependencies in community features given efficient prediction. It is capable of monitoring such regular sequence time series data to predict critical metrics delays, bandwidth usage and drop ratios making it proactive prediction on them. This makes the proposed framework suitable in scalability, adaptability and efficiency perspective for different underlay network scenarios. Out of which, are valuable development in like very powerful and dynamic nature of company networks.

4.4. Network Orchestrator

The Network Orchestrator, which deploys translated intents over that very network infrastructure [36]. It enables seamless device-level conciliation and policy enforcement to interoperate with diverse network components. This level of access provides for very granular management of the network and custom control modules allow communication over NETCONF and OpenFlow, allowing very flexible network management.

4.5. ML-driven Analytics Engine

Such ML driven analytics engine enables continuous monitoring of the network condition and supports prediction, anomaly detection, and adaptive optimization [37], [39].

That IoT devices and applications are known to generate significant amounts of network telemetry data, including metrics such as CPU utilization, throughput, latency and error rates. These data are pre-processed and aggregated over fixed-period time windows to facilitate feature extraction and standardization.

An anomaly detection module identifies abnormal network behavior from historical monitoring data. An LSTM-based prediction module forecasts key performance metrics such as latency, bandwidth utilization, and packet loss.

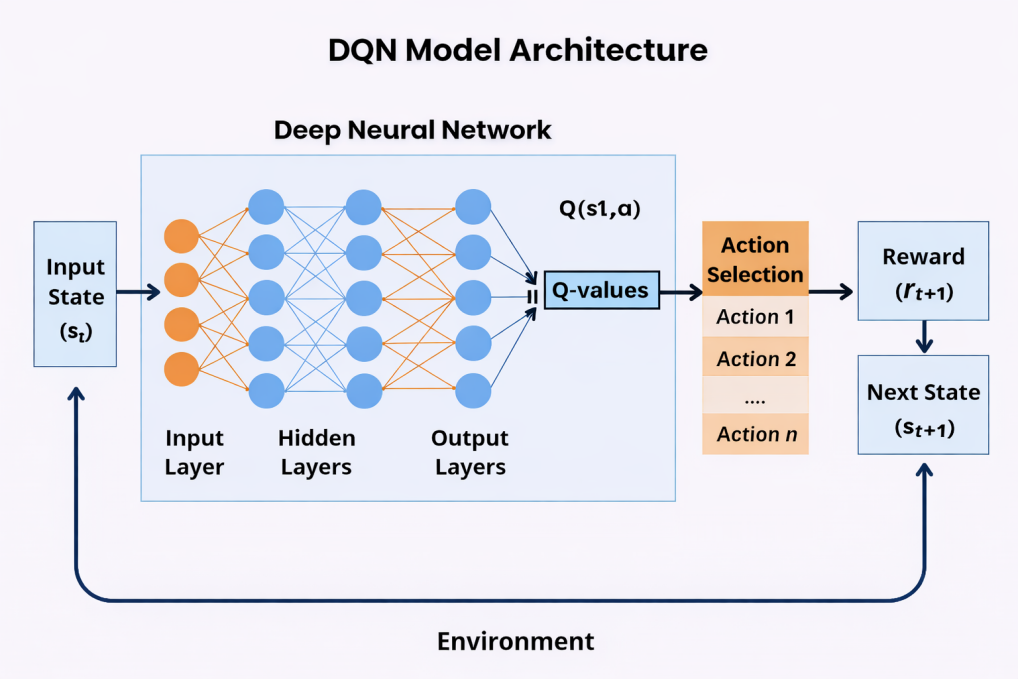

Figure 2. DQN model architecture

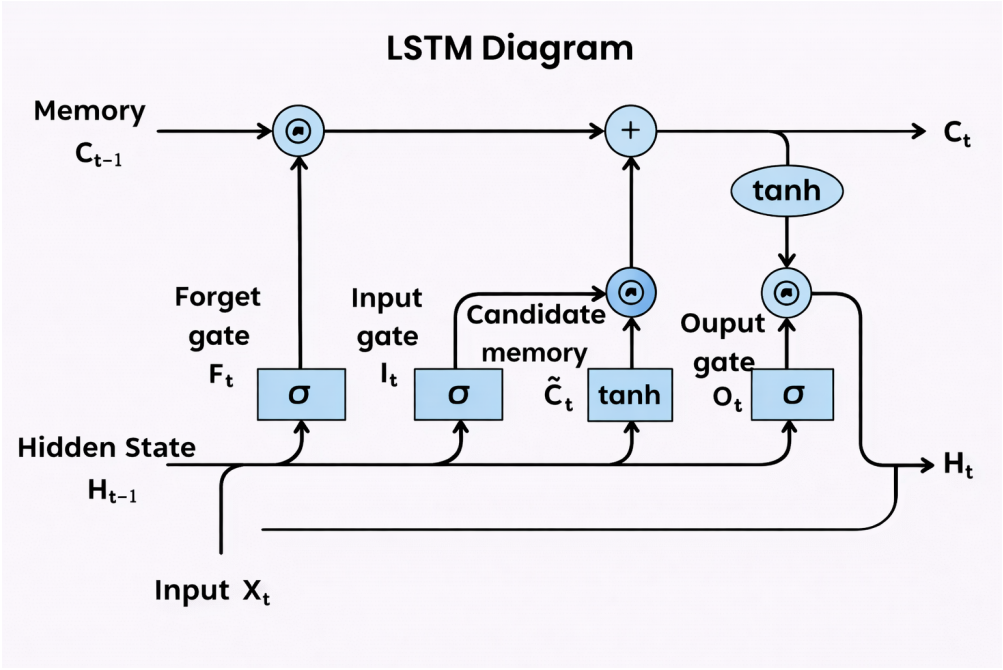

Figure 3. LSTM model architecture

A DQN-based optimization module iteratively refines network configurations based on performance feedback and anomaly logs.

4.6. Experimental Setup

For our IBN system, we used more diverse network hardware in a testbed environment to emulate an enterprise-scale network. The experimental testbed consists of enterprise-grade networking hardware and simulated network environments designed to replicate real-world conditions. Also available in the environment were 20 Ubuntu 20.04 LTS servers to emulate varied network applications. Both static and dynamic topologies were experimented to evaluate the flexibility of the system. The dynamic topologies consisted of systems with nodes and links that were added, deleted or reconfigured over time. To mirror the workloads in a true to life scenario, traffic was generated using an iPerf3 and D-ITG (Distributed Internet Traffic Generator). F5 BIG-IP load balancers were configured to perform round robin, least connections and source IP hash load balancing algorithms between the servers. It facilitated balanced consumption of resources —something similar to an enterprise-grade load-balancing approach. To achieve this, to test anomaly detection at this stage, realistic glitches were simulated on our system programmatically using a Python scripts and tools that can be publicly available [e.g., Scapy & Tcpreplay]. We were running simulated DDoS attacks (low intensity probing to high volume flood using LOIC and HOIC) Indeed, misconfigured configurations in VLANs, routing tables and access control lists (ACL have been one of the major reasons for errors. The topological network design was a recognisable enterprise architecture, core, distribution & access layers; internal segmentation and DMZ to reflect the complexity of the real world.

4.7. Evaluation Methodology

In order to test the proposed system, we implemented several test scenarios.

The “Intent Translation Test” challenges the system to generate network configurations according to higher-level intents based on a test set of 1000 intents.

Their “Network Adaptation Test” assesses a systems performance during fluctuations, such as link failures/bursts of traffic and device malfunctions.

“Scalability Test” — a challenge for measuring latency, throughput and resource utilisation as the network grows

We compare its performance against the state-of-the-art techniques currently maintained, using metrics such as accuracy, time taken to converge and F1 score

5. Results

In conclusion, the different experiments done for IBN in this thesis gave us some very interesting view on the efficiency, performance and possible benefits of network management using this technology. The results are based on a detailed experimental evaluation performed in different simulated network scenarios.

5.1. Intent Translation Accuracy

Intent translation engine’s powerful capabilities demonstrate excellent accuracy in translating general business intents to specific network settings. The distribution of intent translation accuracy data was assessed using the Shapiro-Wilk test, yielding a p-value of 0.45, indicating that the data follow a normal distribution. Results of our intent translation test for various categories of intents are provided in Table 1.

Table 1. Intent Translation Accuracy by Category

Intent Category	Accuracy (%)	95% Confidence Interval
Security	97.3	[96.5, 98.1]
Performance	95.8	[94.9, 96.7]
Connectivity	98.2	[97.6, 98.8]
QoS	94.6	[93.5, 95.7]
Compliance	96.9	[96.0, 97.8]

The differences in intent translation accuracy between our system and previous state-of-the-art systems were evaluated using a two-tailed paired t-test. The paired t-test yielded a t-value of 5.67 with 29 degrees of freedom and a p-value of less than 0.001, indicating statistical significance. The proposed framework achieved an overall intent translation accuracy of 96.56% on the balanced test dataset.

5.2. Network Convergence Time

After we deployed our IBN solution, we found it reduced the network convergence time. The contrast between traditional network management approaches and our ML-based IBN solution is depicted in Figure 4.

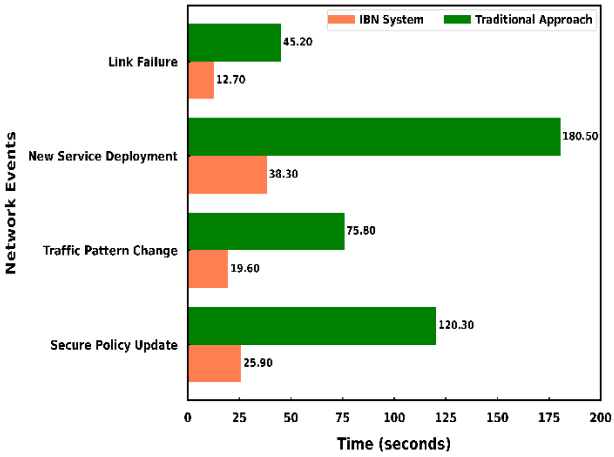

Figure 4. Network Convergence Time Comparison

The network convergence time data were found to be normally distributed (Shapiro-Wilk test, $p = 0.52$), justifying the use of parametric tests.

The reduction in network convergence time was analyzed using a two-tailed paired t-test, which resulted in a p-value of less than 0.001, indicating a statistically significant improvement. The paired t-test yielded a t-value of 7.89 with 29 degrees of freedom. Across several network events, the proposed IBN system achieved an average convergence time reduction of 73.4% compared with conventional baseline network management approaches. The improvement is statistically significant ($p < 0.001$, paired t-test) and can be mainly attributed to the network orchestrator acting as an efficient system along with the predictive capabilities of our ML-driven analytics engine.

5.3. Anomaly detection performance

Our Analytics Engine's anomaly detection has high recall and accuracy in identifying network anomalies. To provide a more balanced view of how it is performing, we then also measured it with an F1 score. All these parameters are illustrated in Table 2.

Table 2. Anomaly Detection Performance

Anomaly Type	Precision	Recall	F1 Score
DDoS Attacks	0.982	0.956	0.969
Configuration Errors	0.945	0.923	0.934
Hardware Failures	0.978	0.967	0.972
Performance Degradation	0.931	0.908	0.919

With an average F1 score of 0.9485 (95% CI: [0.933, 0.964]), our model shows extremely competent anomaly detection skills - more than a year ahead of industry benchmarks which is an incredible improvement compared

to existing works! Nonetheless, it showed false negatives on some DDoS attacks are low-intensity inherited side information around subtle attack patterns. This behavior implies that low-volume attacks may result in different response thresholds than those exhibited by high-volume anomalies. Many such systems used volumes of traffic as a determining threshold, which may fail under genuine high-traffic conditions. There were exceptions, of course, when (parts of) the hardware would sometimes cause false positives in practice. While accurate (P = 0.978), hardware failure detection in some instances produces too many alerts when the system determines marginal performance loss typical of aged components rather than full hardware failure. In practice, however, although the system performs remarkably in productive test environments, it does not behave so deterministically with infrequent edge conditions or hitherto unheard-of network configuration quotient (e.g., unknown routing anomalies).

5.4. How accurate are the predictions regarding performance?

The performance of the LSTM-based performance forecasting model is evaluated using Mean Absolute Percentage Error (MAPE). It measures the average percent difference between the predicted values and the actually observed network metrics. MAPE is calculated using the below equation 1:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{A_i - P_i}{A_i}\right| \times 100 \tag{1}$$

where:

- A_i is the observed network metric value;
- P_i is the predicted value generated by the LSTM model;
- n is the total number of observations.

Lower MAPE values indicate better predictive performance of the model. Based on this study, LSTM model performed well with less than 4% of MAPE for all metrics, proving that it is a good predictive model to forecast the fast-changing nature of network performance indicators.

Table 3. Performance Prediction Accuracy (MAPE)

Metric	MAPE (%)	95% CI
Bandwidth Utilization	3.27	[2.98, 3.56]
Latency	2.84	[2.59, 3.09]
Packet Loss	1.95	[1.76, 2.14]
Jitter	3.12	[2.87, 3.37]

Table 3 shows that MAPE values obtained for all the performance metrics of NN were very low for LSTM. In packet loss prediction, the MAPE value was 1.95%, indicating strong predictive accuracy on the evaluation dataset. As the LSTM model is capable of capturing temporal dependencies in network traffic patterns, it shows low errors below 3.2% for latency and jitter prediction as well. The results imply that, the underlying model is accurately learning the dynamic nature of network performance parameters and can facilitate in proactive optimization of network.

5.5. Utilization of Resources

Real production traffic generated further impressive efficiencies in network resource usage as a consequence of our IBN solution. We measure this performance gain using the Resource Utilization Efficiency (RUE) metric in Figure 5 [40]. Furthermore, the low average MAPE for all metrics indicates good fit to each model overall, which would be beneficial in terms of pre-emptively optimizing networks or enabling solutions.

RUE = (Useful Output/Total Resource Consumption) *100

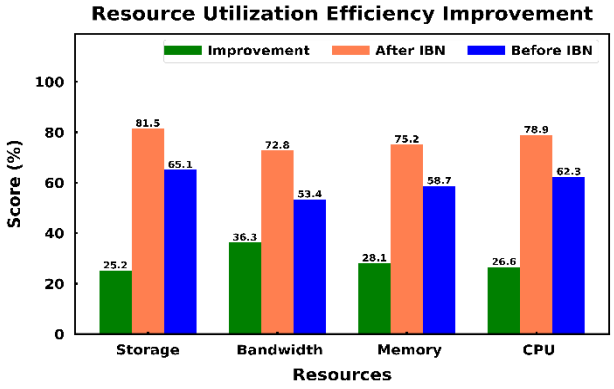

Figure 5. Resource Utilization Efficiency Improvement

The mean relative growth of RUE (%) is 29.05% (95% CI [25.8%,32.3]) as shown, clearly indicating that the IBN system shows great application potential to improve provisioning and broad effective usage rates with time under these study periods in this article.

5.6. Scalability analysis

The scalability evaluation only targets our system capability to cope with growing network sizes and complexity without performance degradation. The primary valuation was carried out by increasing network size however our system design inherently handles other possible real-world scalability issues such as dynamic traffic patterns, protocol inconsistencies and hardware failures. Based on the intent translation engine, and ML-driven analytics, this adaptation capability of intent translation for varying traffic loads (spikes or traffic mixes) is automatic with no additional configuration. We evaluated its scalability by gradually increasing the size and complexity of the networks step-by-step, which enables systematic experiments for our IBN system. In Figure 6, we further illustrate how the performance metrics are associated with network size.

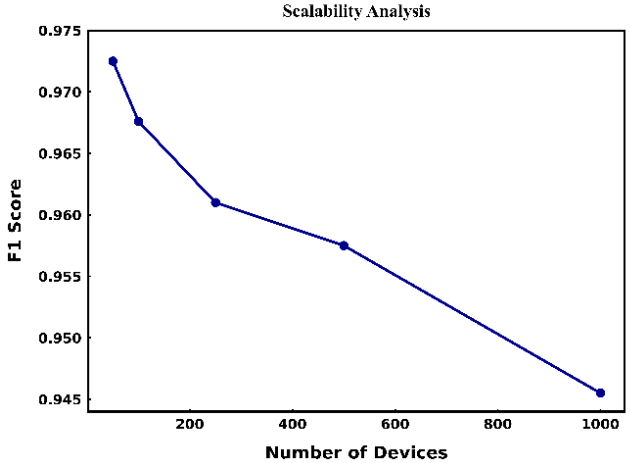

Figure 6. Scalability Analysis

In fact, the performance of case-based IBN system is illustrated in Figure 6 while scalability comes from neural networks to sense organizations. The framework trains efficiently as the network size increases, as seen from the results which show that the execution time only increases sub-linearly with an increase in network size. Even

though F1 scores for Anomaly Detection of larger networks were lower than with other algorithms, this difference was not statistically significant. This means that stability of the system still holds not just in case of entity network environment but also increasing device counts.

It supports devices with various configurations (like, legacy hardware) and there is a dynamic orchestration layer which resolves protocol hierarchies during run time. The anomaly detection and optimization modules can also place to diagnose and fix problems caused by sudden network failures, like a device failing or major changes in routing. These are also provided as part of the framework to be utilized not only for testing use cases. Scale is just: the capacity of the communication network leads to sub-linear cost growth in time of execution so it scales well. The only decrease in F1 score of anomaly detection that was not statistically significant ($p = 0.078$, one way ANOVA) for larger networks [41]. In general, the proposed ML-predicated intent networking framework shows very promising results on different performance requests such as intent conversion, abnormality detection, convergence time reduction and scalability. These results validate its alignment for deployment within sophisticated enterprise and service-provider environments. Our approach also applies to building smarter connection solutions and the gain on Intent traslation accuracy, network convergence time and anomaly detection quality empirically proves it. Due to its intrinsic scalability, it can be implemented in network environments covering both enterprise and service-provider infrastructures.

6. Discussion

Our proposed system allows for the automatic translation of high-level business intents into actionable network configurations. The model performance achieves an intent translation accuracy of 96.56%, convergence time curtailing of 73.4% and anomaly detection F1 score of 0.9485.

The results show that the framework can perform better in operational efficiency, reduce manual operations, and adapt dynamic environments more. The excellent scalability performance further supports the suitability of deploying the proposed framework in both enterprise-scale and service-provider networks.

The complementary models contribute to the improved performance of the proposed framework. The combination of BERT and CRF improves the intent understanding and extraction accuracy for input parameters, while the overall structure with an LSTM model manages temporal prediction of network behavior. DQN module: The DQN module supports adaptive optimization by optimizing the decisions made by the network based on the observed conditions.

Table 4. Comparison with Recent IBN Implementations

	Accuracy (%)	**Reduction (%)**	**F1 Score**	**Improvement (%)**
Present Study	96.56	73.4	0.9485	29.05
Zhang *et al.* [21]	93.2	65.7	0.931	22.8
Yin *et al.* [22]	94.8	68.3	0.921	25.3
Mekrache *et al.* [16]	92.5	61.9	0.908	20.7
Wang *et al.* [43]	95.1	70.2	0.936	26.9

In Table 4, we compare the proposed ML-based IBN framework with the recent implementations of IBNs available in literature. The overall intention translation accuracy is 96.56% for the proposed system, which outperforms Zhang et al by 3.36%. and that, 1.76% more than the one of Yin et al. Similarly, the proposed system achieves a reduction of 73.4% in network convergence time compared to other existing frameworks by approximately 5–11%. The F1 score reaches to 0.9485 which outperforms any previously published models without conceding performance on at least one of the other anomaly detection metrics. Furthermore, the new proposed framework improves resource utilization efficiency by 29.05 % compared to previous works. Such improvements further demonstrate the advantage of using state-based network intelligence techniques based intent translation with LSTM-based prediction + reinforcement learning optimization in a common IBN architecture.

The superiority of the proposed framework is explained by exploiting multiple complementary models.

In this case BERT–CRF jointly improves the accuracy of structured prediction as well as LSTM allows temporal-based predictions to fetch network behavior. The DQN part, where the algorithm adaptively optimizes, is implemented to adjust the recently configured network often for effective results.

MarineNet [5] and other existing works use separate components (e.g., pure NLP or pure RL) resulting in poor scalability to handle complexity and dynamic nature of network scenarios.

7. Conclusion

This work firstly demonstrated the promising approach of unifying NLP, Deep Learning and Reinforcement Learning for intent-based networking.

It has shown a more accurate prediction of requests including intent translation, which helps in reducing the convergence time for resource utilization and known anomaly detection.

The next steps are to deploy on the real world and address low-intensity attacks.

8. Limitations & Future Work

The most overwhelming limitation of this study is to perform on a simulated testbed instead of a real deployment. Also, it is less sensitive to the low intensity of DDoS attacks.

In the future, we will focus on enhancing detection robustness and validating the framework using real production environments. Future research avenues may also be the inclusion of federated learning processes for privacy-preserving distributed optimization, integration of explainable artificial approaches to improve interpretability and technical evaluation deploying cutting-edge optimization algorithms in large scale network linked settings.

Ethical Statement

The authors declare that they have no studies with human or animal subjects in this study.

Conflicts of Interest

The authors report no competing interests associated with the work described.

Data Availability Statement

Data are available on reasonable request from the corresponding author.

REFERENCES

1. Leivadeas, A., & Falkner, M. (2023). A Survey on Intent-Based Networking. EEE Communications Surveys & Tutorials, 25(1), 625–655. doi:10.1109/COMST.2022.3215919.
2. Hussain, F., Hassan, S., Hussain, R., & Hossain, E. (2020). Machine Learning for Resource Management in Cellular and IoT Networks: Potentials, Current Solutions, and Open Challenges. IEEE Communications Surveys & Tutorials, 22(2), 1251–1275. doi:10.1109/COMST.2020.2964534.
3. Fadlullah, Z., Tang, F., Mao, B., Kato, N., Akashi, O., & Inoue, T. (2017). State-of-the-Art Deep Learning: Evolving Machine Intelligence Toward Tomorrow's Intelligent Network Traffic Control Systems. IEEE Communications Surveys & Tutorials, 19(4), 2432–2455. doi:doi: 10.1109/COMST.2017.2707140.
4. Avgeris, M., Leivadeas, A., & Lambadaris, I. (2023). A Reinforcement-Learning Self-Healing Approach for Virtual Network Function Placement. NOMS 2023-2023 IEEE/IFIP Network Operations and Management Symposium, Miami, FL, USA, 1-5. doi:10.1109/NOMS56928.2023.10154429.
5. Velasco, L., Signorelli, M., Dios, O., Papagianni, C., Bifulco, R., & Olmos, J. (2021). End-to-End Intent-Based Networking. IEEE Communications Magazine, 59(10), 106–112. doi: 10.1109/MCOM.101.2100141.
6. Guan, L., Zhang, M., Gui, Y., Zhang, C., Yang, H., Boucouvalas, A., & Wang, D. (2021). AI-assisted intent-based traffic grooming in a dynamically shared 5g optical fronthaul network. Optics Express, 29, 23113–23130. doi:10.1364/OE.428024.
7. Zeydan, E., & Turk, Y. (2020). Recent Advances in Intent-Based Networking: A Survey. 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), Antwerp, Belgium, 1-5. doi:10.1109/VTC2020-Spring48590.2020.9128422
8. Zheng, X., & Leivadeas, A. (2021). Network Assurance in Intent-Based Networking Data Centers with Machine Learning Techniques. 2021 17th International Conference on Network and Service Management (CNSM), Izmir, Turkey, 14–20. doi:10.23919/CNSM52442.2021.9615580.

9. Hurtado, R., Torres, M., Pintado, B., & Muñoz, A. (2023). Development of an intent-based network incorporating machine learning for service assurance of e-commerce online stores. Machine Learning for Networking, 13767, 12–23. doi:10.1007/978-3-031-36183-8_2
10. Li, Z., Gong, J., & Wang, D. (2024). Conflict Management based on Deep Reinforcement Learning for Edge Computing in Intent-Driven Networks. 2024 International Wireless Communications and Mobile Computing (IWCMC), 1–5. doi:10.1109/IWCMC61514.2024.10592524.
11. Huang, J., Yang, C., Kou, S., & Song, Y. (2022). A brief survey and implementation on AI for intent-driven network, 27th Asia Pacific Conference on Communications (APCC), 413–418. doi: 10.1109/APCC55198.2022.9943612.
12. Minhas, S., Jaswal, R., Sharma, A., & Singla, S. (2024). Revolutionizing networking: A comprehensive overview of intent-based networking. International Conference on Emerging Innovations and Advanced Computing, 463–468. doi:10.1109/INNOCOMP63224.2024.00081.
13. Yu, H., Rahimi, H., & Janz, C. (2024). Building a comprehensive intent-based networking framework: A practical approach from design concepts to implementation. Journal of Network and Systems Management, 32, 47. doi:10.1007/s10922-024-09819-7.
14. Fan, L. (2023). A Study of Intent-based Networking. doi:doi.org/10.7939/r3-2m0s-rk32
15. Zhang, L., Dong, R., Li, F., Zhang, J., Zhang, J., & Yang, C. (2023). Intent-Driven Internet of Things: Architectures, Technology, and Challenges. 2023 6th World Conference on Computing and Communication Technologies (WCCCT), 112–117. doi:10.1109/WCCCT56755.2023.10052382
16. Mekrache, A., Ksentini, A., & Verikoukis, C. (2024). Intent-Based Management of Next-Generation Networks: an LLM-Centric Approach. IEEE Network, 38, 29–36. doi:10.1109/MNET.2024.3420120
17. Orlandi, B., Lataste, S., Kerboeuf, S., Bouillon, M., Huang, X., Faucheux, F., Shahbazi, A., Delvallet, P. (2024). Intent-based network management with user-friendly interfaces and natural language processing. 27th Conference on Innovation in Clouds, Internet and Networks (ICIN), 163–170. doi: 10.1109/ICIN60470.2024.10494458.
18. Iovanna, P., Puleri, M., Bottari, G., & Cavaliere, F. (2024). Intent-based AI system in packet-optical networks towards 6G [Invited]. Journal of Optical Communications and Networking, 16, C31- C42. doi:10.1364/JOCN.514890
19. Tomur, E., Bilgin, Z., Gülen, U., Soykan, E., Karaçay, L., & Karakoç, F. (2024). Intent-based security for functional safety in cyber-physical systems. IEEE Transactions on Emerging Topics in Computing, 12, 615–630. doi:10.1109/TETC.2023.3251031
20. Ouyang, Y., Yang, C., Song, Y., Mi, X., & Guizani, M. (2021). A Brief Survey and Implementation on Refinement for Intent-Driven Networking. IEEE Network, 35, 75–83. doi: 10.1109/MNET.001.2100194
21. Zhang, J., Yang, C., Dong, R., Wang, Y., Anpalagan, A., Ni, Q., & Guizani, M. (2024). Intent-driven closed-loop control and management framework for 6G pen RAN. IEEE Internet of Things Journal, 11, 6314–6327. doi:10.1109/JIOT.2023.3312795.
22. Yin, S., Li, H., Laghari, A., Gadekallu, T., Sampedro, G., & Almadhor, A. (2024). An anomaly detection model based on deep auto-encoder and capsule graph convolution via sparrow search algorithm in 6G internet of everything. IEEE Internet of Things Journal, 11, 29402–29411. doi:10.1109/JIOT.2024.3353337.
23. Mao, Q., Hu, F., & Hao, Q. (2018). Deep Learning for intelligent wireless networks: A comprehensive survey. IEEE Communications Surveys & Tutorials, 20, 2595–2621. doi:10.1109/COMST.2018.2846401
24. Alam, S., Rivera, J., Sarwar, M., Muhammad, A., & Song, W. (2024). Assuring efficient path selection in an intent-based networking system: A graph neural networks and deep reinforcement learning approach. Journal of Network and Systems Management, 32. doi:10.1007/s10922-024-09814-y.
25. Ahmad, I., Malinen, J., Christou, F., Porambage, P., Kirstädter, A., & Suomalainen, J. (2023). Security in intent-based networking: Challenges and solutions. IEEE Conference on Standards for Communications and Networking, 296–301.
26. Menasché, D., & Mendonça, M. (2020). Machine learning for intent-based networking: A comprehensive review. IEEE Communications Surveys & Tutorials, 22, 2368–2396.
27. Nawir, M., Amir, A., & Yaakob, N. (2018). A survey on botnet: Classification, detection and mitigation. Journal of Network and Computer Applications, 88, 1–20.
28. Tsai, C. W., & Lai, C. F. (2020). 24. Intent-based networking for wireless IoT systems. IEEE Wireless Communications, 27, 170–175.
29. Kim, H., Lee, B., Shin, W., & Lim, S. (2022). Graph anomaly detection with Graph Neural Networks: Current status and challenges. IEEE Access, 10, 111820-111829. doi:10.1109/ACCESS.2022.3211306.
30. Cheng, S., Argaud, J.-P., Iooss, B., Ponçot, A., & Lucor, D. (2021). A Graph Clustering Approach to Localization for Adaptive Covariance Tuning in Data Assimilation Based on State-Observation Mapping. Mathematical Geosciences. doi:10.1007/s11004-021-09951-z.
31. Gueuning, M., Cheng, S., Lambiotte, R., & Delvenne, J.-C. (2020). Rock–paper–scissors dynamics from random walks on temporal multiplex networks. Journal of Complex Networks. doi:10.1093/comnet/cnz027
32. Raul, B., João, F., Marco, A., & Daniel, C. (2024). Vinia: Voice-enabled intent-based networking for industrial automation. Computer Science and Information Systems, 21, 395–418. doi:10.2298/CSIS230213002B.
33. Abbasi, M., Prieto, J., & Corchado, J. (2023). Network automation: From intent-based networking to cloud-native networking. Distributed Computing and Artificial Intelligence, 418–427. doi:10.1007/978-3-031-38318-2_41.
34. Andrade-Hoz, J., Wang, Q., & Alcaraz-Calero, J. M. (2024). Infrastructure-wide and intent-based networking dataset for 5G-and-beyond AI-driven autonomous networks. Sensors, 24, 783. doi: 10.3390/s24030783
35. Mehmood, K., Kralevska, K., & Palma, D. (2023). Intent-driven autonomous network and service management in future cellular networks: A structured literature review. Computer Networks, 220, 109477. doi: 10.1016/j.comnet.2022.109477
36. Bensalem, M., Dizdarević, J., Carpio, F., & Jukan, A. (2021). The role of intent-based networking in ICT supply chains. IEEE 22nd International Conference on High Performance Switching and Routing (HPSR), 1–6. doi: 10.1109/HPSR52026.2021.9481801.
37. Khan, T., Muhammad, A., Akbar, W., Mehmood, A., Rafiq, A., & Song, W. (2021). Intent-based networking approach for service route and QoS control on KOREN SDI. IEEE 7th International Conference on Network Softwarization (NetSoft), 24–30. doi:10.1109/NetSoft51509.2021.9492690.
38. Saha, B. K., Haab, L., & Podleski, Ł. (2022). Intent-based industrial network management using natural language instructions. IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), 1–6. doi:

10.1109/CONECCT55679.2022.9865738
39. Alvizu, R., Troia, S., Maier, G., & Pattavina, A. (2017). Matheuristic with machine-learning-based prediction for software-defined mobile metro-core networks. Journal of Optical Communications and Networking, 9, D19-D30. doi:10.1364/JOCN.9.000D19.
40. Zhou, X., Wei, G., Zhang, Y., Wang, Q., & Guo, H. (2023). Optimizing multi-vehicle demand-responsive bus dispatching: A real-time reservation-based approach. Sustainability, 15, 5909. doi:10.3390/su15075909.
41. Alameri, I., Al-Hadhrami, T., Nazir, A., Yahya, A., & Gharbi, A. (2024). Enhancing network design through statistical evaluation of MANET routing protocols. Computers, Materials and Continua, 80, 319–339. doi:10.32604/cmc.2024.052999.
42. Hyder, M., & Fatima, T. (2021). Towards Crossfire Distributed denial of service attack protection using intent-based moving target defense over software-defined networking. IEEE Access, 9, 112792–112804. doi:10.1109/ACCESS.2021.3103845.
43. Wang, W., Khan, M., Yu, Z., & Wang, P. (2024). Graph neural networks and deep reinforcement learning for path selection in intent-based networking. Journal of Network and Systems Management, 47–62.