International Journal of Scientific Research in Engineering & Technology
Volume6, Issue4 (July-August 2026), PP: 109-112.
www.ijsreat.com ISSN No: 2583-1240

LLM Powered Network Configuration Assistant

Dr Nuthan A C[1], Shruthi V[2], Tejash Kumar N[3]
[1,2,3] Department of Electronics and Communication Engineering, G Madegowda Institute of Technology Bharathinagara, Karnataka, India.

To Cite this Article: *Dr Nuthan A C[1], Shruthi V[2], Tejash Kumar N[3],"LLM Powered Network Configuration Assistant", International Journal of Scientific Research in Engineering & Technology, Volume 06, Issue 04, July-August 2026, PP:109-112.*

Abstract: *Recent advances in Large Language Models (LLMs) have created new opportunities for automating network management tasks. This paper presents an LLM-based intelligent assistant that translates natural language user requests into vendor-specific network configuration commands. The proposed system integrates prompt engineering, an LLM inference engine, validation mechanisms, and network automation frameworks to generate accurate configurations for platforms such as Cisco and Juniper. Implemented using Python with network automation tools, the assistant supports automated configuration generation and iterative refinement based on user feedback. Experimental evaluation demonstrates that advanced LLMs, particularly GPT-4, produce accurate and context-aware configurations while reducing manual effort and configuration errors. The study also discusses practical challenges, including prompt quality, model hallucinations, security considerations, and API costs. The proposed approach highlights the potential of generative AI to simplify network administration and provides a foundation for future research on context-aware, multi-vendor intelligent network automation.*

Keywords: *Large Language Models (LLMs), Network Automation, Generative AI, Natural Language Processing (NLP), Intent-Based Networking (IBN), Software-Defined Networking (SDN), GPT-4, LLaMA2, Network Configuration, Prompt Engineering.*

I. INTRODUCTION

The rapid growth of cloud computing, IoT, SDN, NFV, and 5G/6G technologies has significantly increased the complexity of modern communication networks. Traditional network configuration methods rely on vendor-specific command-line interfaces and manual scripting, making network management time-consuming, error-prone, and dependent on expert knowledge.Network automation tools such as Ansible and Netmiko have improved operational efficiency but still require scripting expertise. More recently, Intent-Based Networking (IBN) and Large Language Models (LLMs) have enabled users to express networking requirements in natural language, allowing intelligent systems to generate device-specific configurations automatically.This paper presents an **LLM-Powered Network Configuration Assistant** that translates natural language user requests into validated network configuration commands. The proposed framework combines prompt engineering, LLM inference, configuration validation, and network automation to simplify network management. Experimental results demonstrate that advanced LLMs, particularly GPT-4, generate accurate and context-aware configurations while reducing manual effort and configuration errors. The study also discusses current challenges and highlights future directions for intelligent, scalable, and autonomous network management.

Scope of the project

The proposed **LLM-Powered Network Configuration Assistant** aims to simplify network configuration by translating natural language user requests into vendor-specific configuration commands for routers, switches, and firewalls. The system supports common networking tasks such as IP addressing, VLAN configuration, routing protocols, and access control policies.A validation module verifies generated configurations for syntax and logical correctness before deployment, reducing configuration errors and improving reliability. The framework also provides command explanations and user guidance, making it suitable for both network professionals and learners.The system is designed to be scalable and extensible, with support for integration into existing network automation frameworks and future multi-vendor environments. However, the current implementation is limited to controlled or simulated network environments, while real-time production deployment and advanced autonomous network management remain areas for future work.

Problem Statement

Modern networks are becoming increasingly complex, making manual network configuration difficult, time-consuming, and prone to errors. Existing automation tools reduce repetitive tasks but still require scripting expertise and vendor-specific knowledge. This project addresses these challenges by developing an **LLM-Powered Network Configuration Assistant** that converts natural language user requests into accurate, vendor-specific network configurations. The proposed system aims to improve automation, reduce configuration errors, and simplify network management.

II.FUNCTIONAL REQUIREMENTS AND SYSTEM ANALYSIS

Functional Requirements

The proposed LLM-Powered Network Configuration Assistant accepts natural language network requests, identifies user intent, and generates vendor-specific configuration commands. The system validates the generated configurations before deployment and supports multiple network devices and vendors. It also utilizes network context to improve configuration accuracy and reliability.

Non-Functional Requirements

The system is designed to provide high configuration accuracy, low response latency, reliability, security against malicious inputs, scalability for large network environments, and a user-friendly interface with clear feedback and logging mechanisms.

Workflow Diagram

The workflow begins with a user submitting a natural language request. The system identifies the intended network operation, generates the required configuration commands, validates them for correctness, deploys the validated configuration to network devices, and finally provides execution feedback to the user.

Network Topology Diagram

The proposed architecture consists of an LLM Assistant connected to network devices through routers and switches. The assistant generates configuration commands, which are applied to routers, switches, and connected end devices, enabling automated and centralized network management.

Data Flow Diagram

The data flow starts with a natural language request from the user. The LLM processes the request, generates device-specific configurations, stores or retrieves relevant information from the configuration database, and deploys the validated configuration to the target network devices.

Use Case Diagram

The Network Administrator interacts with the system to generate, validate, deploy, and monitor network configurations. The assistant automates the complete configuration workflow, reducing manual effort while improving configuration accuracy, consistency, and operational efficiency.

Procedure methodology

Data Collection and Preparation

The proposed system utilizes network configuration guides, vendor documentation, RFCs, sample network topologies, and policy documents as data sources. Synthetic configuration datasets generated using GPT-4 can be employed to augment limited training data, improving the performance of LLM-based network configuration models.

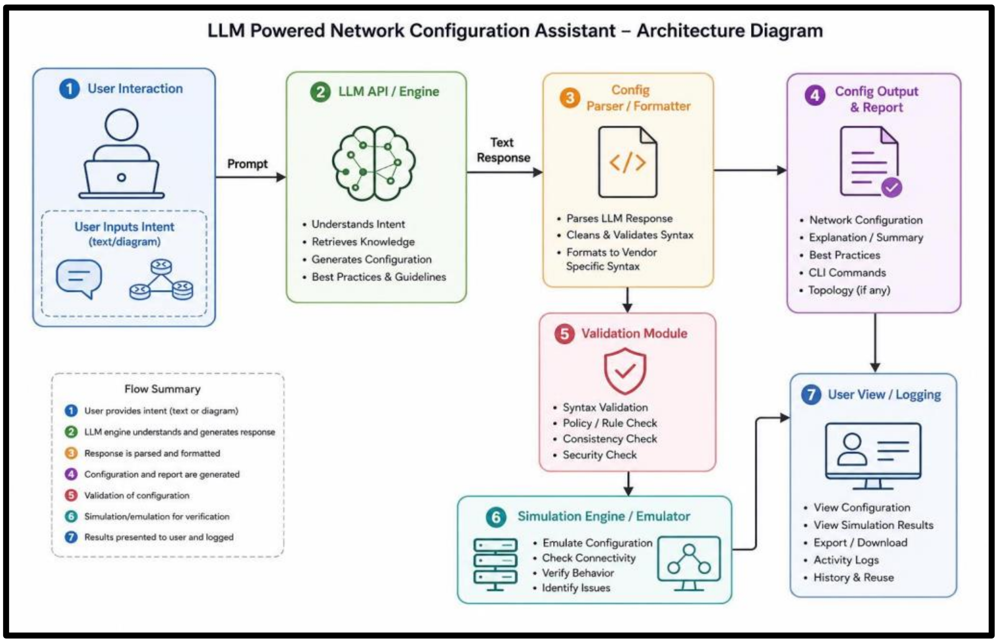


Fig: Methodology

LLM Selection and Fine-Tuning

Several LLMs, including GPT-4, GPT-3.5, Llama-2, Mistral, Falcon, and domain-specific models, are evaluated for network configuration tasks. Model selection is based on configuration accuracy, inference time, and deployment cost. Fine-tuning open-source models on network configuration datasets can further enhance performance and domain adaptability.

Model	Params	Vendor	Architecture	Notable Features
GPT-4/GPT-4o	~500B	OpenAI	Decoder- only	Highest accuracy for Config tasks 【12†L219- L227】
ChatGPT-3.5	6B*	OpenAI	Decoder- only	Conversational, slower, less accurate than GPT- 4 【12†L219-L227】
Llama-2 (7B/13B)	7B/13B	Meta	Decoder- only	Open, can be fine-tuned on domain data
Mistral-7B/13B	7B/13B	Mistral AI	Decoder- only	Compact, high reasoning capability
Cisco NX-AI F16	1B	HuggingFace (Cisco)	Decoder	Fine-tuned on Cisco configs, smaller model 【6†L303-L310】
Falcon-40B	40B	TII	Decoder- only	Code-oriented, could handle script generation
* GPT-3.5-Turbo is ~6B effective (via distillation).				

Prompt Engineering Strategy

The system employs structured prompt engineering to improve configuration generation. User requests are combined with network context, device information, and policy constraints to generate accurate vendor-specific configurations. Prompt optimization and contextual information improve the reliability and consistency of the generated outputs.

Evaluation Metrics

The proposed framework is evaluated using configuration accuracy, semantic correctness, execution time, token usage, policy compliance, and functional verification in simulated network environments. These metrics assess the effectiveness, efficiency, and reliability of the generated network configurations across different LLMs.

III.RESULT

The proposed prototype successfully generated vendor-specific network configurations from natural language inputs. The generated configurations were validated and tested in a GNS3 environment, where routing and connectivity were successfully established.

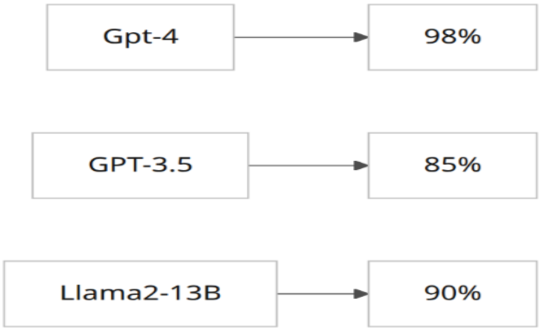
Performance evaluation showed that **GPT-4** achieved the highest configuration accuracy (**98%**) and syntax correctness, outperforming GPT-3.5, Llama-2, and Mistral. Fine-tuning improved the performance of open-source models. Among the automation frameworks evaluated, **Nornir with Netmiko** provided the best balance of flexibility and scalability for multi-device configuration deployment.

The system demonstrated low response latency and reliable configuration generation in simulated environments. Feedback from network engineers indicated that the assistant significantly reduced manual configuration effort while emphasizing the need for expert validation in complex networking scenarios. Although GPT-4 incurred API costs, it delivered superior performance compared to open-source alternatives. Overall, the results demonstrate the effectiveness of LLMs in intelligent network configuration automation.

Tool	Type	Language	Features	Use in Project
Netmiko	SSH library	Python	Simplifies SSH to many vendors 【43†L175-L184】	Used for low-level CLI tasks.
NAPALM	Abstraction	Python	Unified API for vendor net 【43†L185-L193】	Planned for config retrieval (future).
Nornir	Framework	Python	Multi-threaded, Python- native 【43†L199-L204】	Main engine for parallel pushes.
Ansible	Declarative	YAML	Agentless, many network modules 【43†L193-L200】	Considered for potential deployment.
Paramiko	SSH library	Python	Basic SSH (no device logic) 【43†L175-L184】	Used indirectly via Netmiko.

Table no 1: Network Automation Tools Comparison

Table



IV.CONCLUSION

The proposed **LLM-Powered Network Configuration Assistant** can be applied in enterprise networks, cloud data centers, and telecommunication systems to automate network configuration, improve consistency, and reduce manual effort. It also supports cybersecurity by assisting with firewall, ACL, and security policy configuration.The system is beneficial for educational institutions as a learning tool for networking concepts and for small and medium enterprises (SMEs) that lack experienced network administrators. It can be integrated with DevOps and network automation frameworks to support automated deployment and configuration management.Furthermore, the assistant aids in network troubleshooting, multi-vendor network management, and intelligent AIOps environments, contributing toward autonomous, scalable, and self-managing communication networks.

REFERENCES

1. N. Tu, S. Nam, and J. W.-K. Hong, “Intent-Based Network Configuration Using Large Language Models,” International Journal of Network Management, 2025.
2. S. Chakraborty, N. Chitta, and R. Sundaresan, “Automation of Network Configuration Generation Using Large Language Models,” ResearchGate Preprint, 2024.
3. C. Wang et al., “NetConfEval: Can LLMs Facilitate Network Configuration?” IEEE Conference on Network Management, 2024.
4. O. G. Lira, M. Caicedo, and N. L. S. da Fonseca, “Large Language Models for Zero- Touch Network Configuration Management,” IEEE Communications Magazine, 2024.
5. F. Li, H. Lang, J. Zhang, J. Shen, and X. Wang, “PreConfig: A Pretrained Model for Automating Network Configuration,” arXiv preprint arXiv:2403.09369, 2024.
6. S. K. Mani et al., “Enhancing Network Management Using Code Generated by Large Language Models,” arXiv preprint arXiv:2308.06261, 2023.
7. A.Abane, A. Battou, and M. Merzouki, “An Adaptable AI Assistant for Network Management,” NIST Publication, 2024.
8. B. Ifland et al., “GeNet: A Multimodal LLM-Based Co-Pilot for Network Topology and Configuration,” IEEE Networking Conference, 2024.
9. K. Aykurt, A. Blenk, and W. Kellerer, “NETLLMBENCH: A Benchmark Framework for Large Language Models in Network Configuration Tasks,” IEEE Transactions on Network and Service Management, 2024.
10. F. Liu, B. Farkiani, and P. Crowley, “Large Language Models for Computer Networking Operations and Management: A Survey on Applications, Key Techniques, and Opportunities,” Computer Networks Journal, 2025.
11. I. Protogeros, R. Asadli, B. Hoffman, and L. Vanbever, “Benchmarking LLM-Driven Network Configuration Repair,” arXiv preprint, 2026.
12. “Survey on LLM-Based Agents for Network Operations and AIOps,” Journal of AI in Networking Systems, 2026.