Intent-based Automated NFV Management Methods using Large Language Models

Ph.D. Thesis Defense

Jibum Hong

Supervisor: Prof. James Won-Ki Hong

Distributed Processing and Network Management Lab.
Dept. of Computer Science and Engineering
Pohang University of Science and Technology, South Korea

hosewq@postech.ac.kr

December 18, 2025

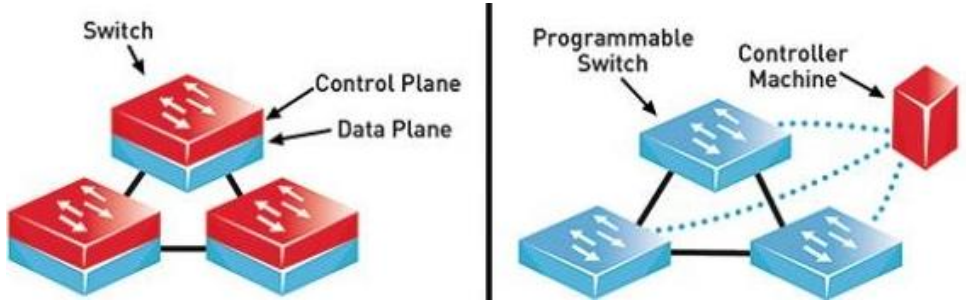
Table of Content

Introduction

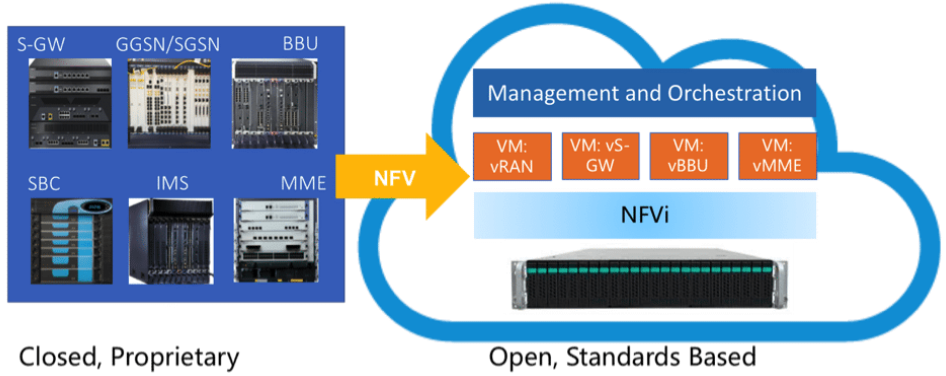
- Introduction
- Problem Statement
- Research Goals and Contributions

Introduction (1/3)

- Network Virtualization
  - Modern network extends beyond physical networks to virtual networks
  - SDN/NFV gives us new ways to design, build, and operate networks
    - Separate control/data planes, centralized management
    - Hardware dedicated, closed network functions → Softwarized Virtual Network Functions (VNFs)
    - Reduce CAPEX/OPEX
  - Recently, telcos and service providers use SDN/NFV to provide their services more efficiently

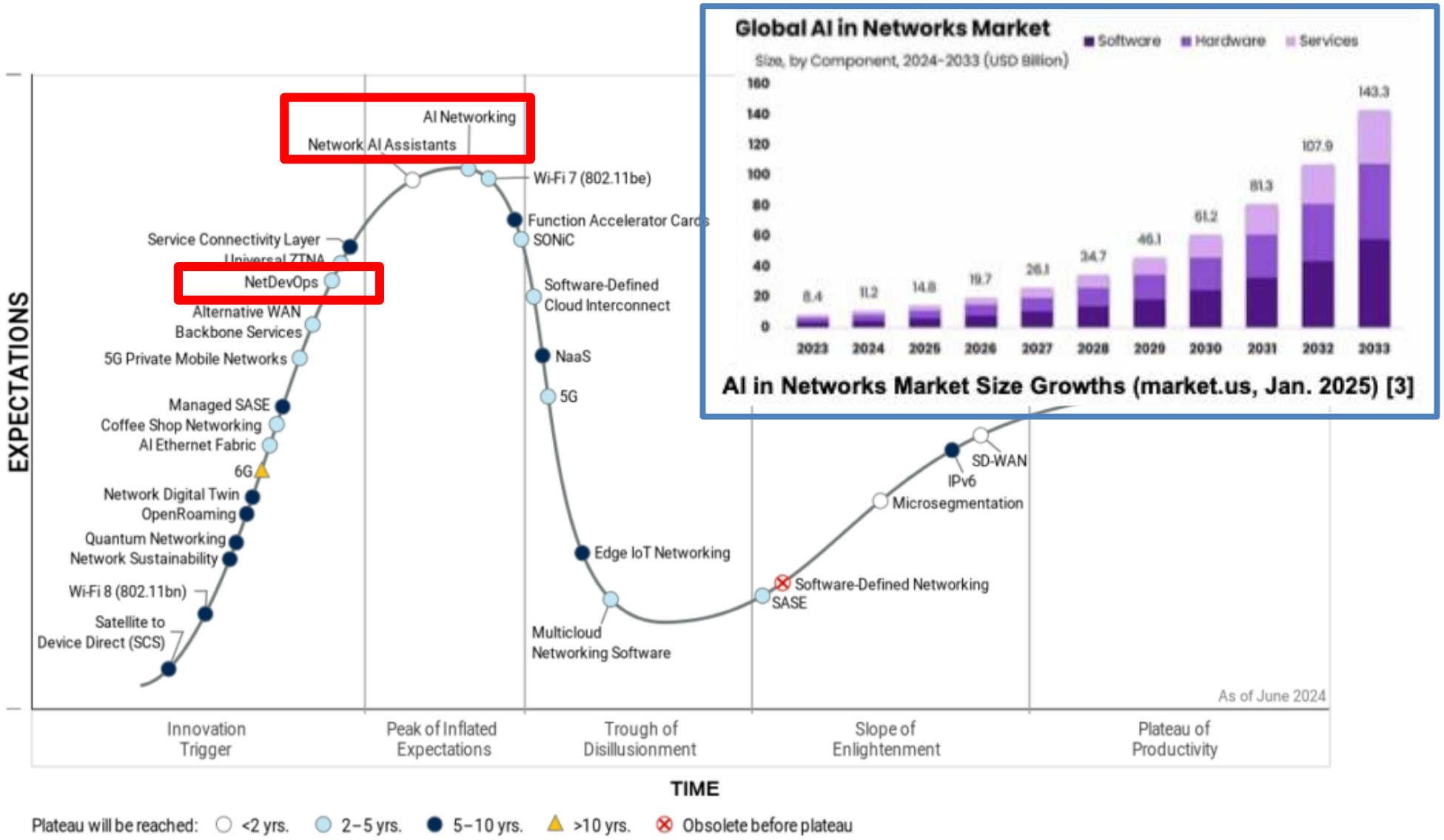


Software-Defined Networking (SDN)



Network Function Virtualization (NFV)

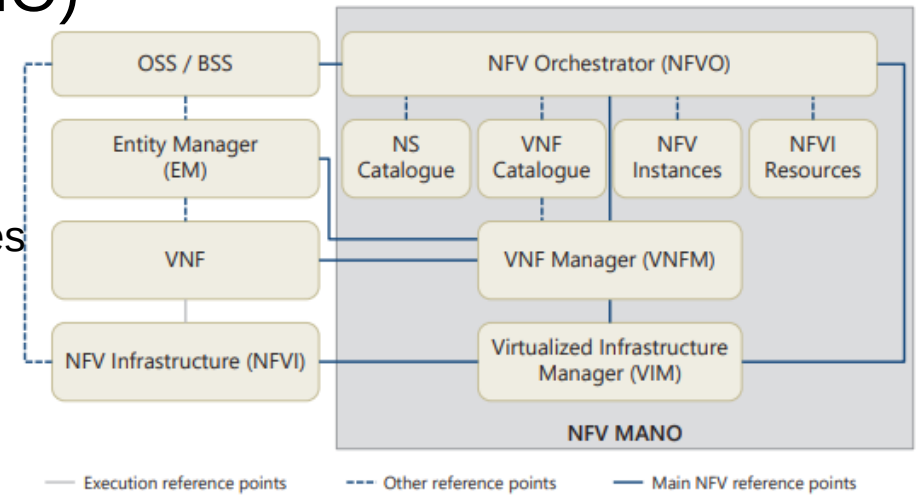
Introduction (2/3)



Hype Cycle for Enterprise Networking (Gartner, June 2024) [2]

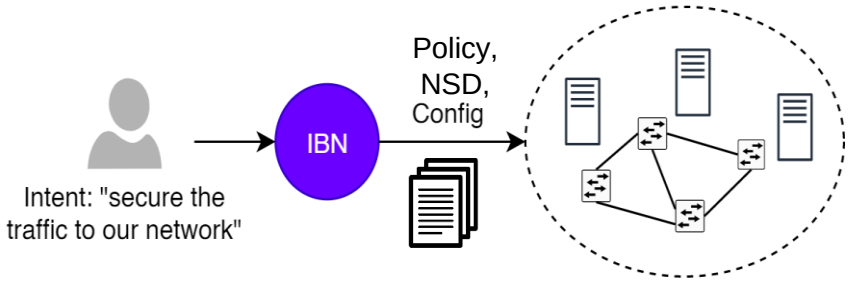
Introduction (3/3)

- NFV Management and Orchestration (NFV-MANO)
  - Provide networks and services in NFV environments
  - Manage physical/virtual networks
    - Resource components: network resource, VNF instance/services
    - Management components: NFV Orchestrator, VNF Manager, Virtualized Infrastructure Manager, etc.



ETSI NFV-MANO architecture [1]

- Intent-based Networking (IBN)
  - Use high-level intent to operate networks and services
    - Specify user's management objectives
  - Understand and interpret user's intent
  - Translate natural language intent and generate management policies and low-level configurations



Concept of Intent-based Networking

Problem Statement

- Needs for AI-driven Network Management
  - Challenges in AI-driven Network Management
    - Most of studies focus on specific management function on FCAPS
      - FCAPS: Fault, Configuration, Accounting, Performance, Security Management
      - e.g., AI-based anomaly detection, resource optimization, etc.
    - Mainly use AI models for specific functions based on generated policy by administrator

      → Need to generate network and service management policies based on intent (IBN)

- Limitations of legacy IBN approaches
  - Traditional rule-based approaches focus on intent parsing → low accuracy and scalability
  - NLP-based approaches
    - Legacy NLP approaches → Still hard to generalization, context understanding, generation accuracy
    - Recent studies still have focused on simple intent translation using LLMs

      → Need policy generation and intent fulfillment, not just abstraction by intent translation

Research Goals & Contributions

- Intent-based Automated Network Management Methods using LLMs in NFV environment
  - LLM-based Intent translation and management policy generation
    - Translate user intent and generate network and service management policy
      - Improve translation accuracy and policy generation accuracy using LLMs
    - Adjust LLM optimization techniques to maximize LLM inferences for network management domain
      - In-context learning with prompt engineering, RAG, feedback mechanism, etc.
    - Case study on management scenarios
      - VNF/service management tasks such as deployment, Service Function Chaining (SFC), auto-scaling, security, etc.
  - Policy validation and intent fulfillment framework
    - Validate generated management policy before deploying to real NFV environment by control plane
      - Utilize the verification and regeneration closed-loop with syntax and semantic validation
    - Evaluate intent fulfillment in real NFV environment
      - Implement the LLM-based pipelines to deploy the generated policy
      - Verify the actual accuracy to identify the user's intent is reflected well in network operations

Related Work

- Background
- Related Work

Background

- Applying LLMs to Network Management Domain
  - How do LLM understand network domain knowledge?
    - Use pre-trained knowledge and new/updated domain-specific data
- Domain Adaptation Techniques
  - Prompt Engineering
    - Design the prompt to align with specific task
  - Retrieval-Augmented Generation (RAG)
    - Retrieve the additional documents from external knowledge bases
  - Feedback mechanism
    - Align outputs using self-feedback or human feedback

Related Work

- Related studies for IBN approaches using LLM

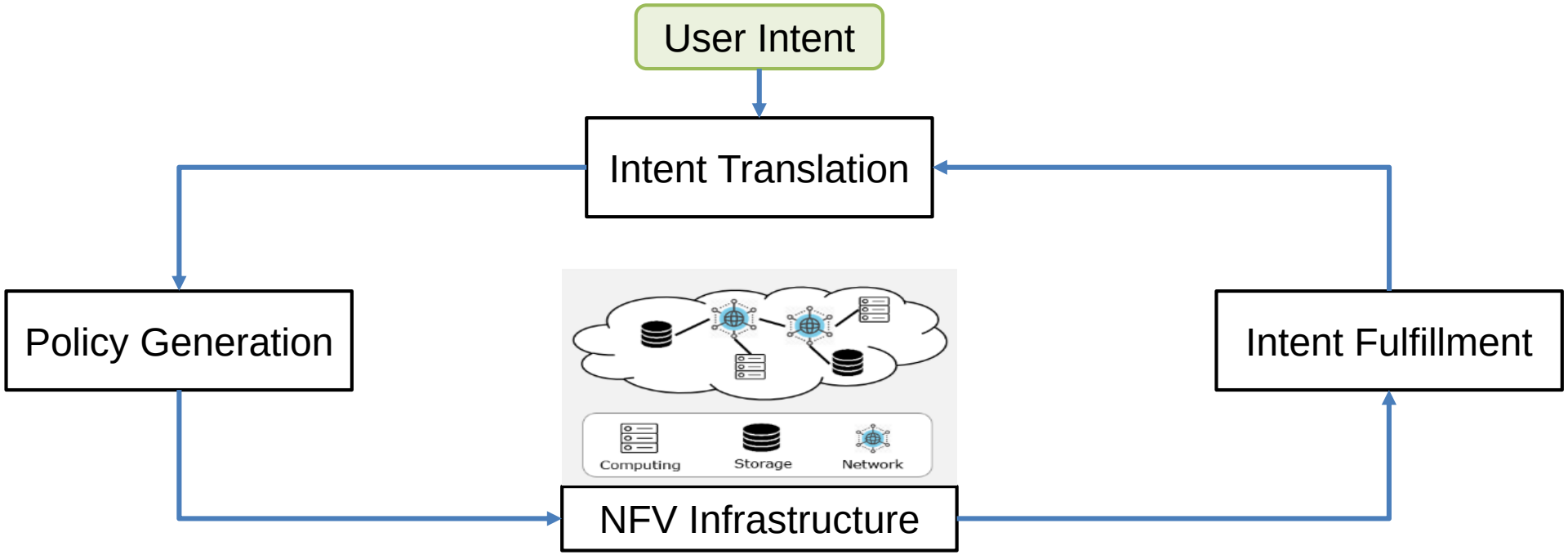
Reference	Proposed	Target Output	Target Environment	Models/Techniques
Dzeparoska et al. [6] (2024)	Intent translation and policy generation	Policy for instance deployment (create, delete)	NFV environment (OpenStack)	GPT-3.5 / Prompt engineering
Mekrache et al. [7] (2024)	Intent translation and NSD generation	NSD for service deployment (create, delete)	5G testbed	GPT-3.5 / Prompt engineering
Wang et al. [8] (2024)	Intent management architecture design	Management policy	Overall networks	(not implemented)
Kou et al. [9] (2025)	Intent abstraction and detect collision	Abstracted template (python script), intent collision detection	Simulation (Python NetworkX)	GPT-3.5, GPT-4, GPT-4o / not specified
Tu et al. [10] (2025)	Intent translation for network configuration	Management template (JSON) for device/NFV configuration	Network devices and NFV environment (OpenStack)	LlaMA, GPT / Prompt engineering, fine-tuning
Nam et al. [11] (2025)	MOP-based automated VNF installation & configuration	VNF instance creation and installation workflows	NFV environment (Kubernetes)	Open-source LLMs, GPT / Prompt engineering, RAG
Ours	**Intent management (intent translation, policy generation, intent fulfillment)**	**NFV Management policy (deployment, SFC, auto-scaling, security/network configuration)**	**NFV environment (Kubernetes)**	**Open-source LLMs, GPT/ Prompt engineering, RAG, feedback mechanism**

Design

- Thesis Scope
- Overall Designs
- Intent Translation
- Policy Generation

Thesis Scope

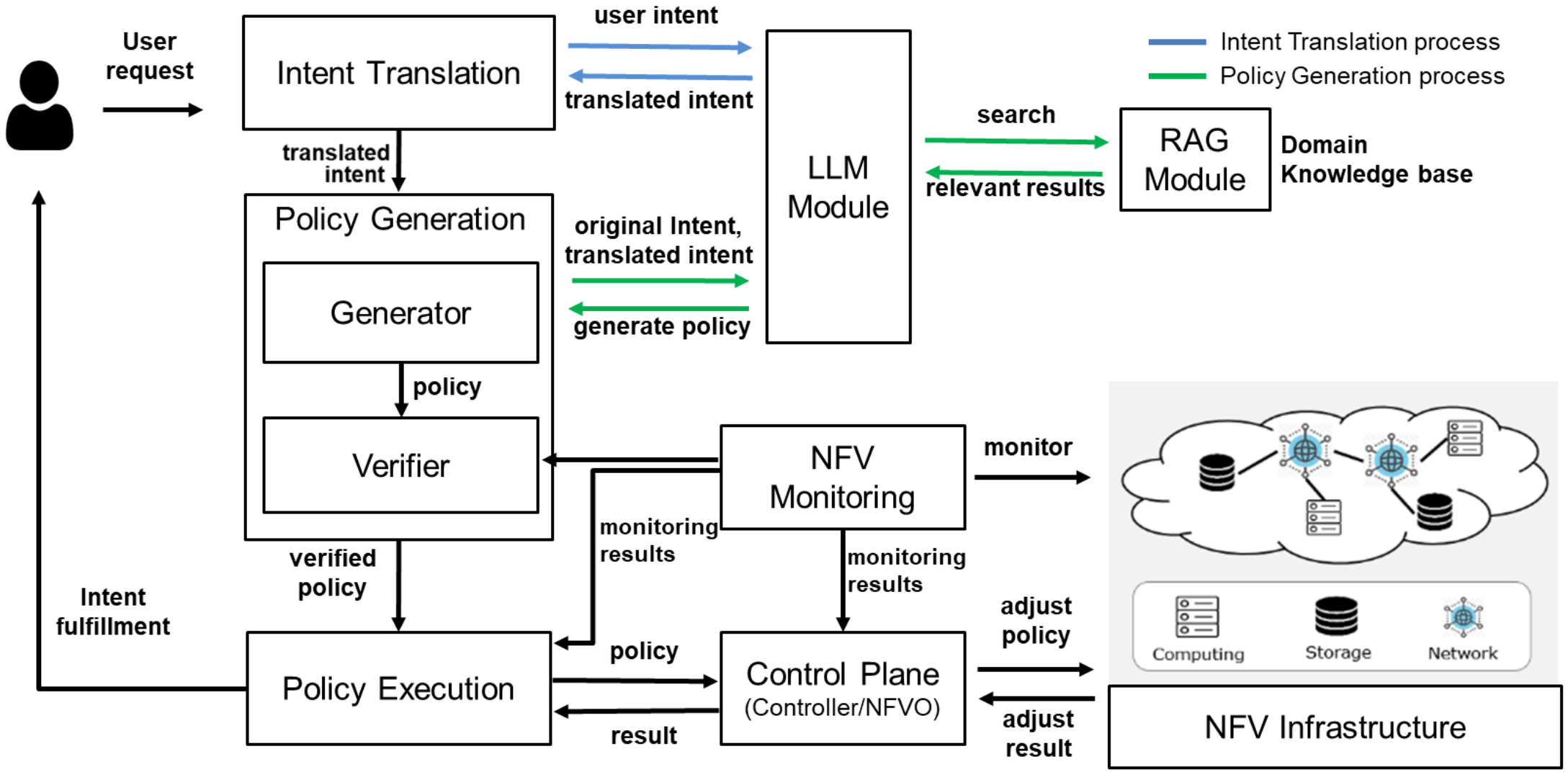
- Intent-based Automated NFV Management Methods using LLMs
  - Intent Translation
  - Policy Generation
  - Intent Fulfillment



Overall Concept of IBN (Thesis Scope)

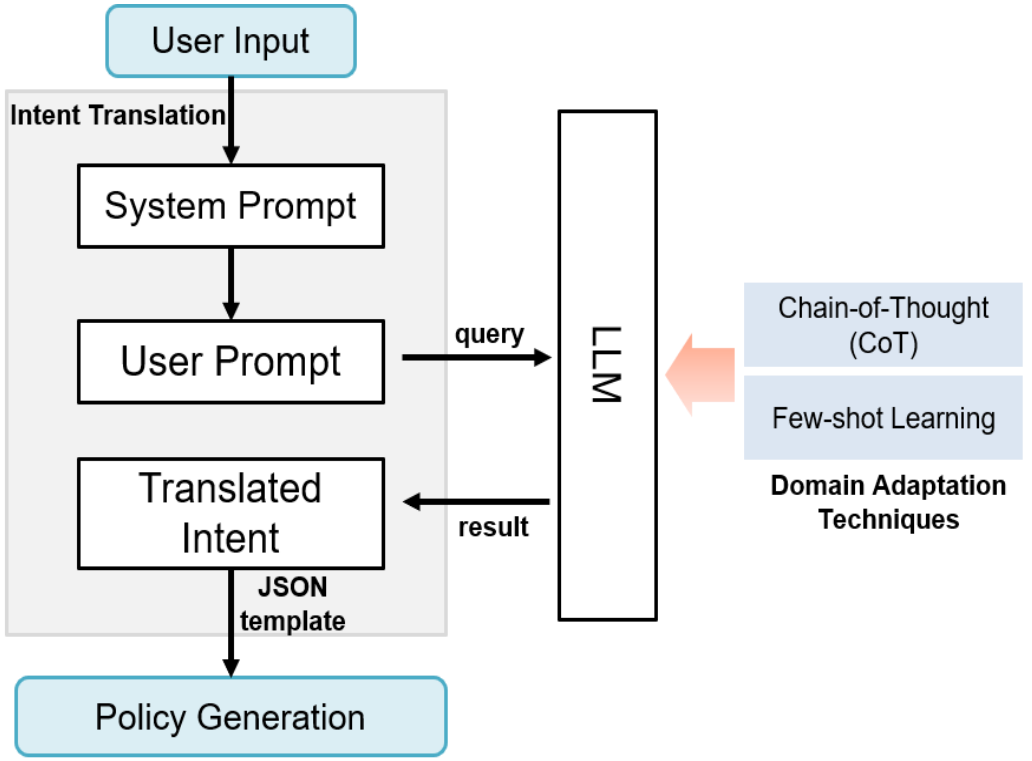
Overall Design

- Intent-based Automated NFV Management Framework



Intent Translation

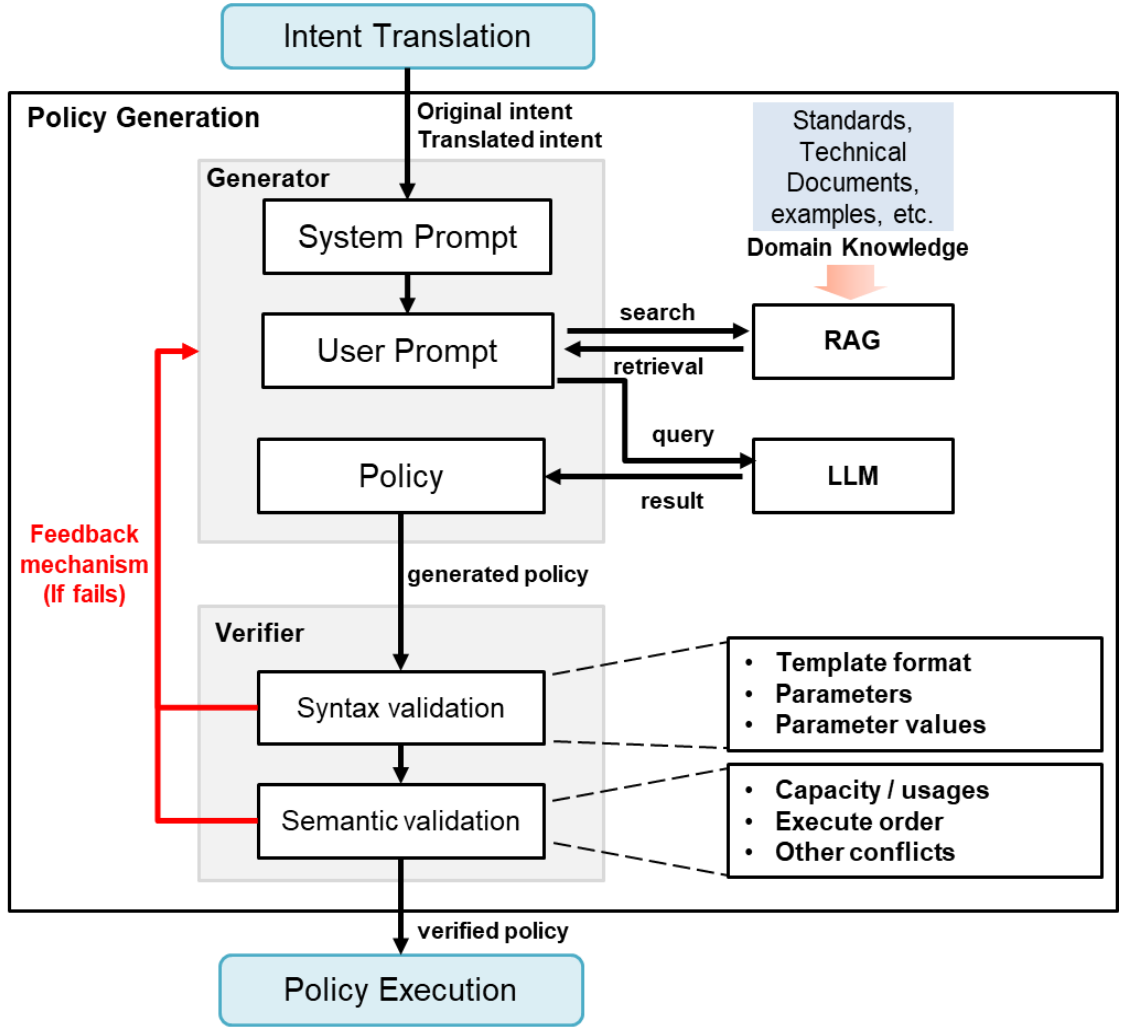
- Intent Translation
  - Understand and translate input using LLM and Identify “what user want” and keywords
  - Refer IETF RFC documents
    - RFC 9315 - IBN: Concepts and Definitions
    - RFC 9316 - Intent Classification
  - Define target management tasks, action, requirements, constraints
    - Tasks: deployment, SFC, auto-scaling, etc.
    - Actions: create, update, delete
    - Targets: instance/service information
    - Requirements
    - Constraints



Overall Design of Intent Translation

Policy Generation (1/3)

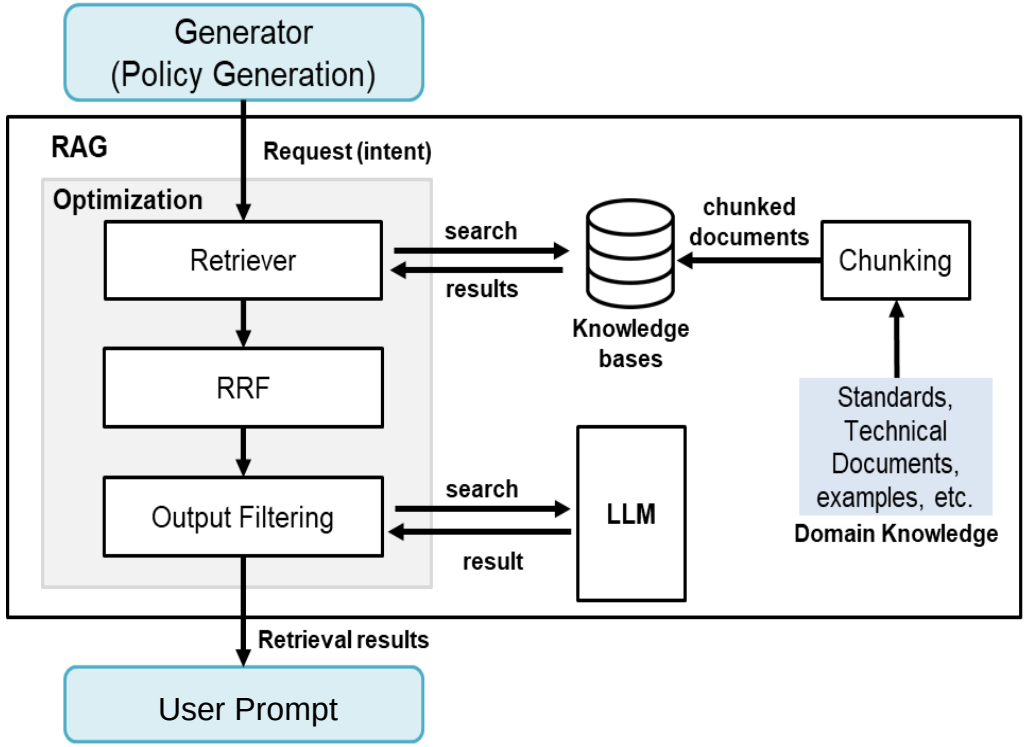
- Policy Generation
  - Management policy generation
    - Target environment (nodes, services, instances, etc.)
    - Management tasks
      - Deployment
      - SFC
      - Auto-scaling
      - Security configuration
      - Network configuration
    - Output
      - YAML manifest template
  - Verify the generate policy
    - Syntax validation
    - Semantic validation



Overall Design of Policy Generation

Policy Generation (2/3)

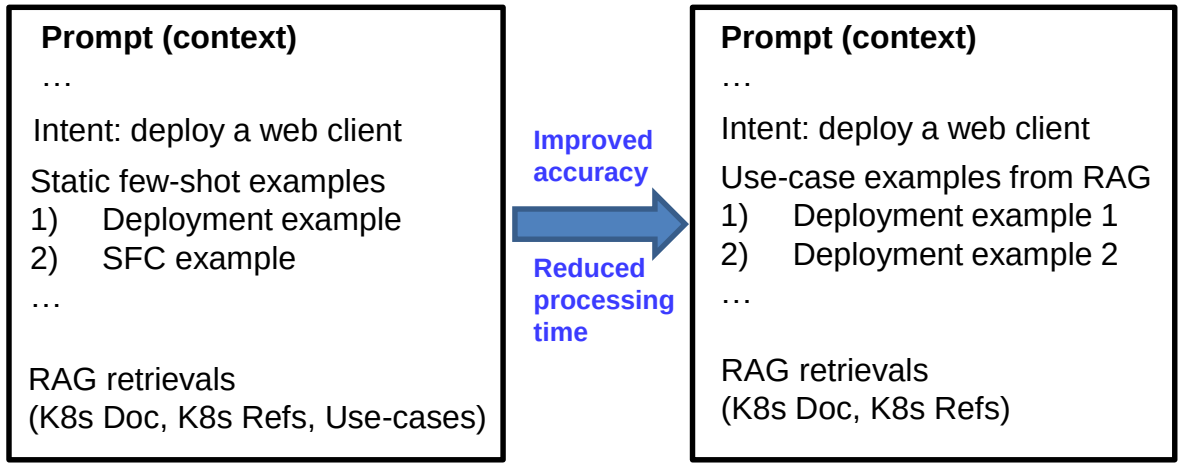
- RAG (Retrieval-Augmented Generation) for Policy Generation
  - Use additional domain knowledges for NFV Management
    - Tasks/Definitions: Kubernetes technical documents (K8s Docs)
    - Attributes: Reference API documents (K8s Refs)
    - Examples: Executable use-case examples (Use-cases)
  - RAG Optimization techniques
    - Chunking
    - Dynamic few-shot examples
    - RRF (Reciprocal Rank Fusion)
    - Output filtering



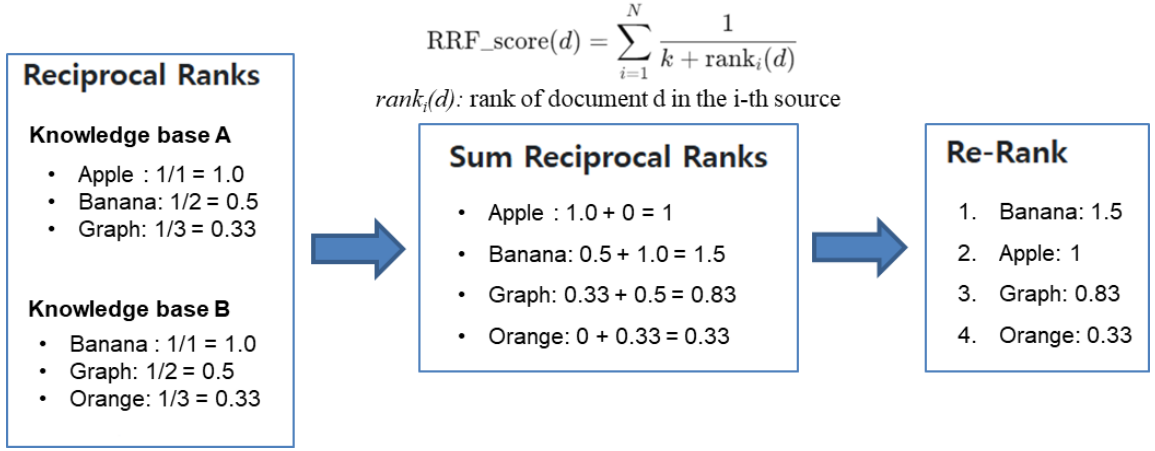
Overall Design of Policy Generation

Policy Generation (3/3)

- Dynamic Few-shot Examples
  - Provide few-shot examples by given intent dynamically
    - Retrieve 3 few-shot examples with RAG
    - Uses “Use-cases” as a knowledge base
- Reciprocal Rank Fusion (RRF)
  - Create integrated ranks by weighted rank from multiple RAG retrievers
  - Optimize RAG retrievals, reduce the context length of prompt



Static few-shot examples approach vs Dynamic few-shot examples approach



Basic concept of RRF algorithm

Implementation

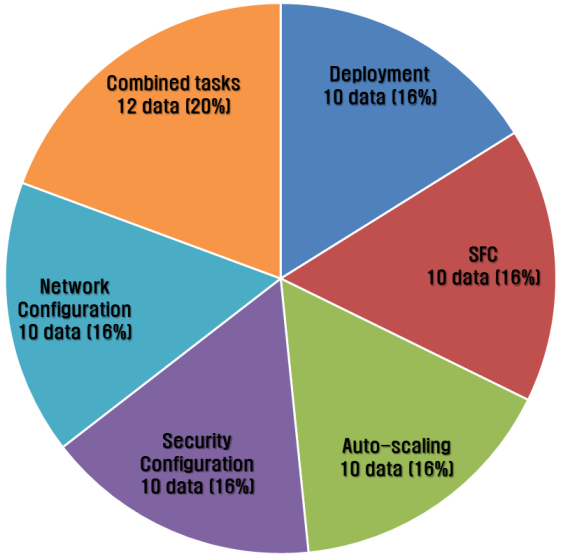
- Data Generation
- Intent Translation
- Policy Generation

Data Generation

- Intent Dataset
  - Refer the intent data from documents by ETSI, Kubernetes, and related studies
  - User commands and instructions expressed in natural language
    - Include Target task, action, target resource/network, requirements, constraints
  - Include 5 management tasks (10 data each) + combined tasks (12 data)
    - Design the data from low complexity intent to more complex intent

Target Tasks	Intent Examples
Deployment	Create an Intrusion Detection System (IDS) instance `ids1` and Deep Packet Inspection (DPI) instance `dpi2` with 1 replica on `k8s-worker02` node. Resource requests: 500m CPU and 512Mi memory; resource limits: 1000m CPU and 1024Mi each.
SFC	Configure a Service Function Chain (SFC) with Firewall–IDS–DPI VNF instances in order. The names of instances are Firewall=`fw`, IDS=`ids`, and DPI=`dpi` in the `sfc` namespace.
Auto-scaling	Update a web server named `web-scale` in the `scaling` namespace with 2 replicas, providing an auto-scaling function when CPU utilization exceeds 60%; maximum replica = 5, minimum replica = 1.
Security Configuration	Block IP `10.10.10.16` and port 80 for VNF instances with selector `vnf-sec-block` in the `security` namespace.
Network Configuration	Limit the traffic shaping for Load Balancer (LB) instance named `lb1` to 10 Mbps bandwidth for ingress/egress traffic.

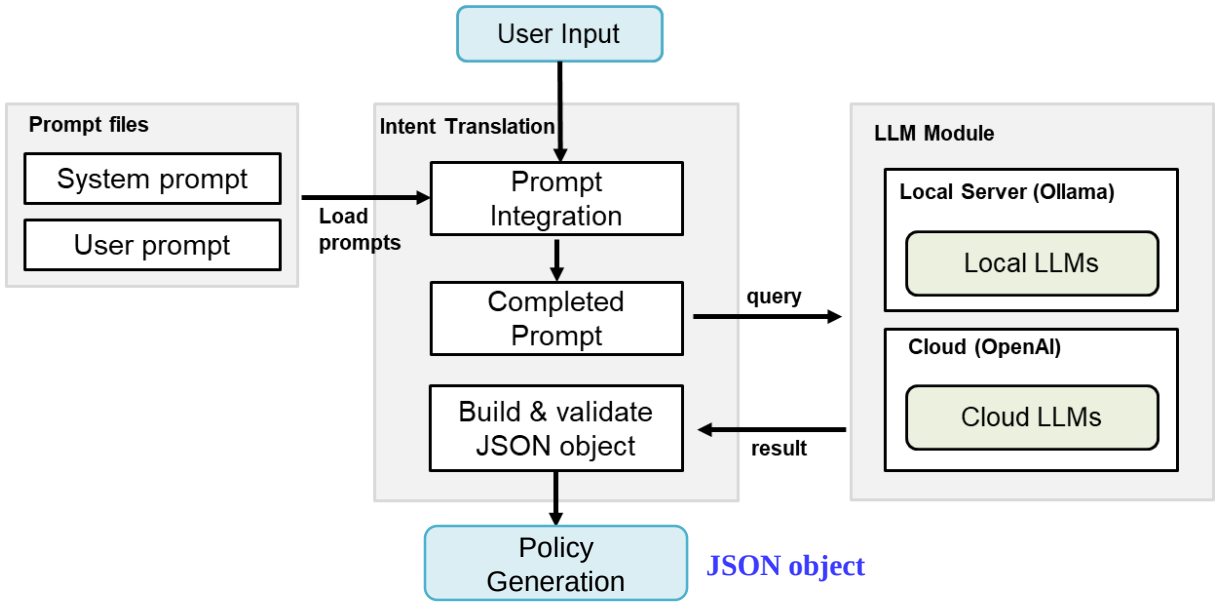
Examples of NFV management intents



The proportion of intent dataset

Intent Translation

- Implementation of Intent Translation
  - Prompt design with CoT and few-shot learning
    - System Prompt (CoT): step-by-step reasoning, define the rules and requirements
    - User Prompt (Few-shot learning): 5 few-shot examples
  - Build and validate the JSON object template based on the output of LLM module



Implementation of Intent Translation

Example Intent: Create a http web server named 'web-svc' with 2 replicas in 'web-ns' namespace.

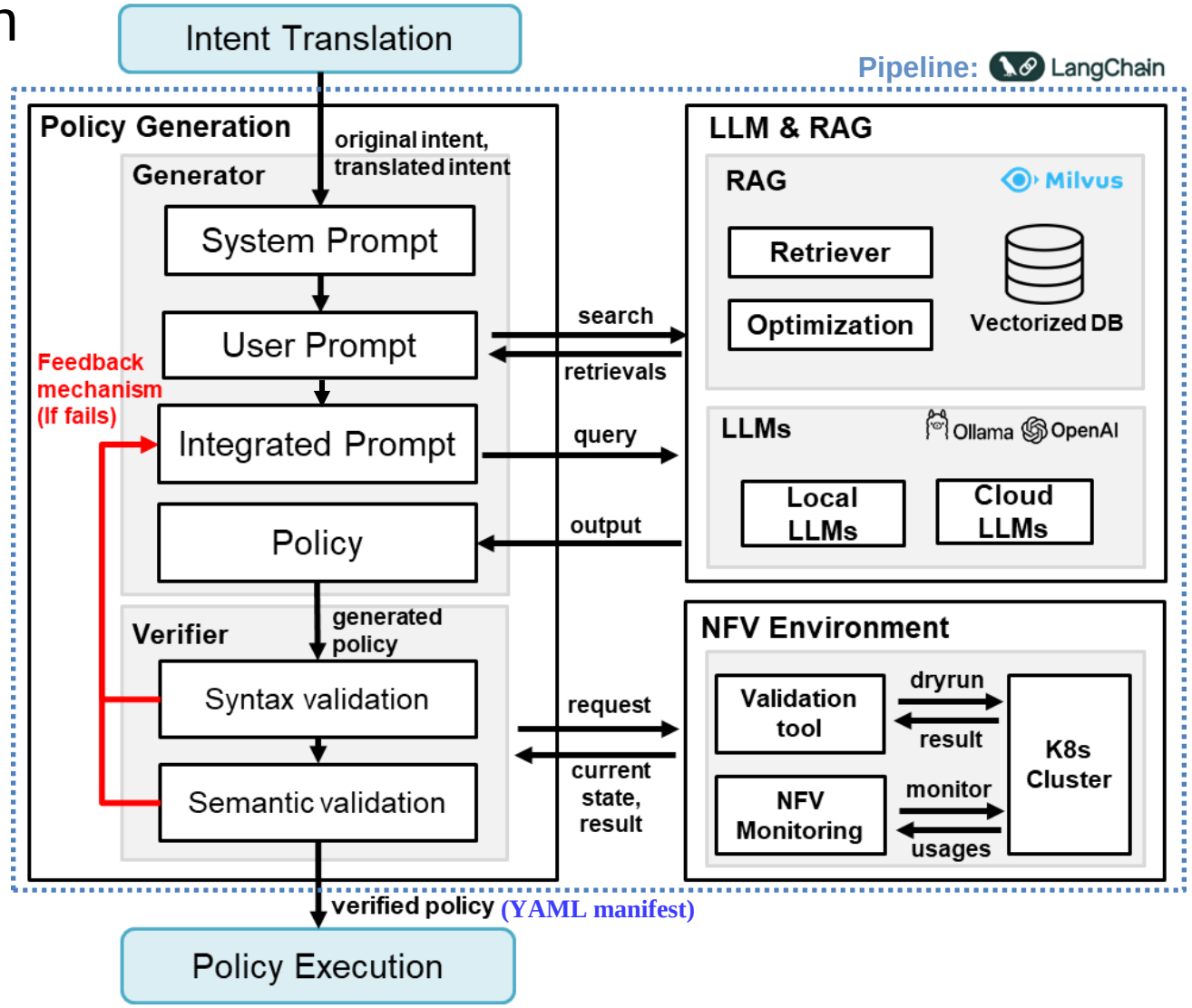
Expected Output:

```
{
  "task": "deployment",
  "action": "create",
  "target_instances": [
  {
    "type": "web",
    "name": "web-svc",
    "replicas": 2,
    "networking": {
      "namespace": "web-ns",
      "ports": [
      {
        "port": 80,
        "protocol": "TCP"
      }
      ]
    }
  }
  ],
  "requirements": {
    "sfc": {
      "enabled": false
    },
    "autoscaling": {
      "enabled": false
    },
    "slo": {
      "enabled": false
    }
  },
  "constraints": {}
}
```

Example of intent translation result

Policy Generation

- Implementation of Policy Generation
  - Build pipelines using open-source tools and frameworks
    - Overall pipeline: LangChain
    - LLM: Ollama, OpenAI
    - RAG: Milvus
  - Build and validate the K8s YAML manifest template based on the output of LLM module
  - Interacts with NFV environment



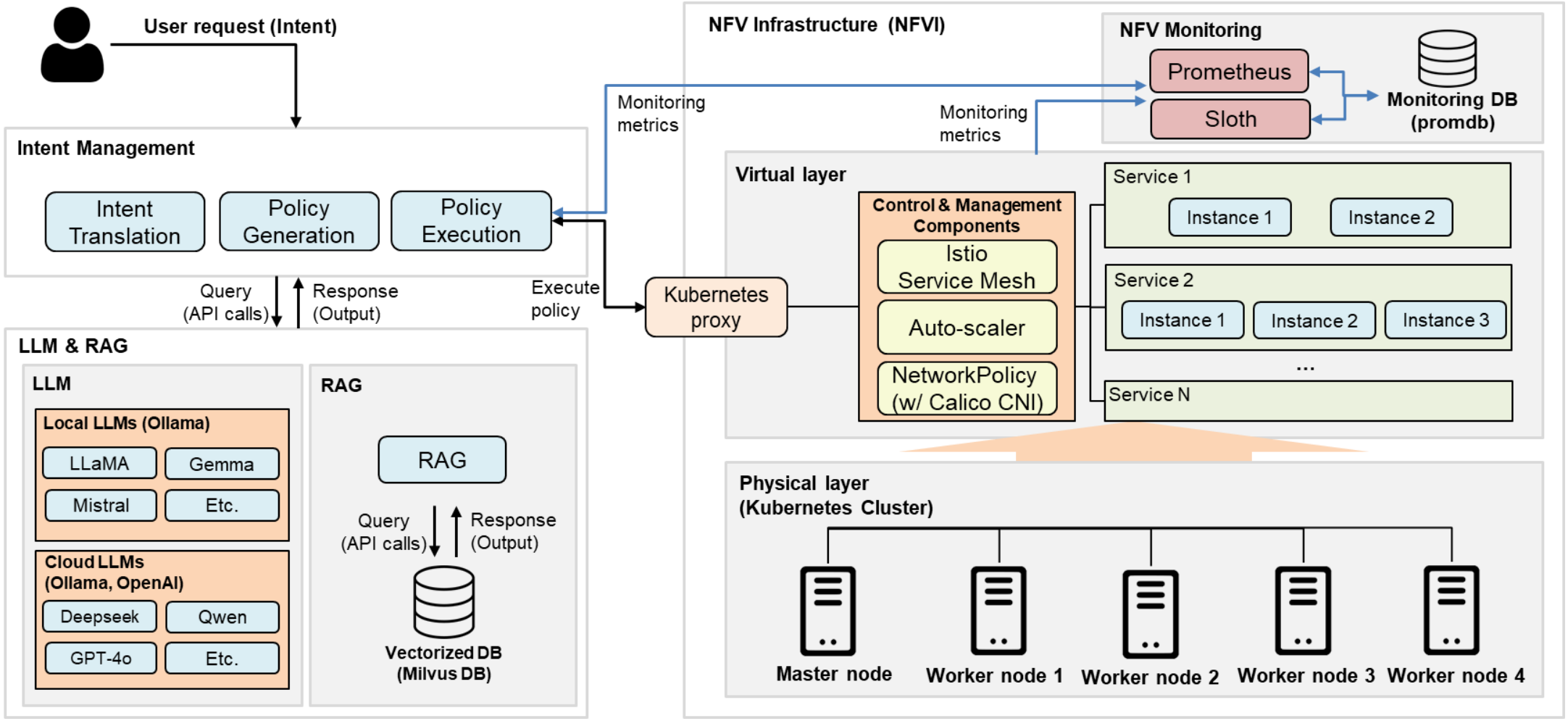
Implementation of Policy Generation

Evaluation

- Experimental Setup
- Intent Translation
- Policy Generation
- Entire Framework Analysis

Experimental Setup (1/2)

- End-to-End (E2E) Testbed Environment



E2E testbed for evaluation based on open-source framework and tools

Experimental Setup (2/2)

- Used LLM models for evaluation

LLMs	Model Names (Parameters)	Company (Release date)	Descriptions
Local models (GPU: RTX A6000 VRAM 48GB)			
Mistral	mistral:7b (7b)	Mistral AI (2023.09)	Light-weight model
Phi4	Phi4-reasoning:plus (14b)	Microsoft (2025.04)	Light-weight, reasoning model
GPT-OSS	gpt-oss:20b (20b)	OpenAI (2025.08)	Open-weight model of GPT for local deployment, support reasoning
Gemma3	gemma3:27b (27b)	Google (2025.03)	Multi-modal/lingual model with long-context support
Qwen3	qwen3-coder:30b (30b)	Alibaba (2025.04)	Optimized model for coding/agentic tasks, support reasoning
DeepSeek	deepseek-r1:70b (70b)	DeepSeek (2025.01)	High-capacity open model, support reasoning
LLaMA3.3	llama3.3:70b (70b)	Meta (2024.12)	High-capacity open model, support multi-modal/lingual reasoning
Cloud models			
GPT-OSS	gpt-oss:120b (120b)	OpenAI (2025.08)	Open-weight model of GPT for local deployment, support reasoning
MiniMax-M2	minimax-m2:cloud (230b)	MiniMax AI (2025.10)	High-parameter MoE model optimized for coding, agentic workflows
GLM-4.6	glm-4.6:cloud (357b)	Z.ai (2025.09)	Large MoE model with advanced reasoning
Qwen3	qwen3-coder:480b-cloud (480b)	Alibaba (2025.07)	Agentic MoE model, multi-modal and long-context support
DeepSeek v3.1	deepseek-v3.1:671b-cloud (671b)	DeepSeek (2025.08)	Hybrid reasoning model
Kimi-K2	kimi-k2:1t-cloud (1t)	Moonshot AI (2025.07)	Trillion-parameter class agentic model, built for largescale reasoning
GPT-4o	gpt-4o (200b, estimated)	OpenAI (2024.05)	Closed-weight model, CoT and multi-modal reasoning
GPT-5.1	gpt-5.1 (1.3~1.8t, estimated)	OpenAI (2025.10)	Frontier Closed-weight model of GPT

Intent Translation (1/3)

- Evaluation for Intent Translation
  - Experiment 1 – Performance comparison with existing approaches
    - Accuracy and processing time analysis
    - Existing studies: Rule-based heuristic approaches, baseline LLM, previous studies
  - Experiment 2 – Performance comparison with various LLM models
    - Accuracy and processing time analysis
    - LLM models
      - 7 local models (from lightweight models to heavy capacity models) in Ollama platform
      - 2 cloud models (GPT-4o and GPT-5.1) in OpenAI platform

Intent Translation (2/3)

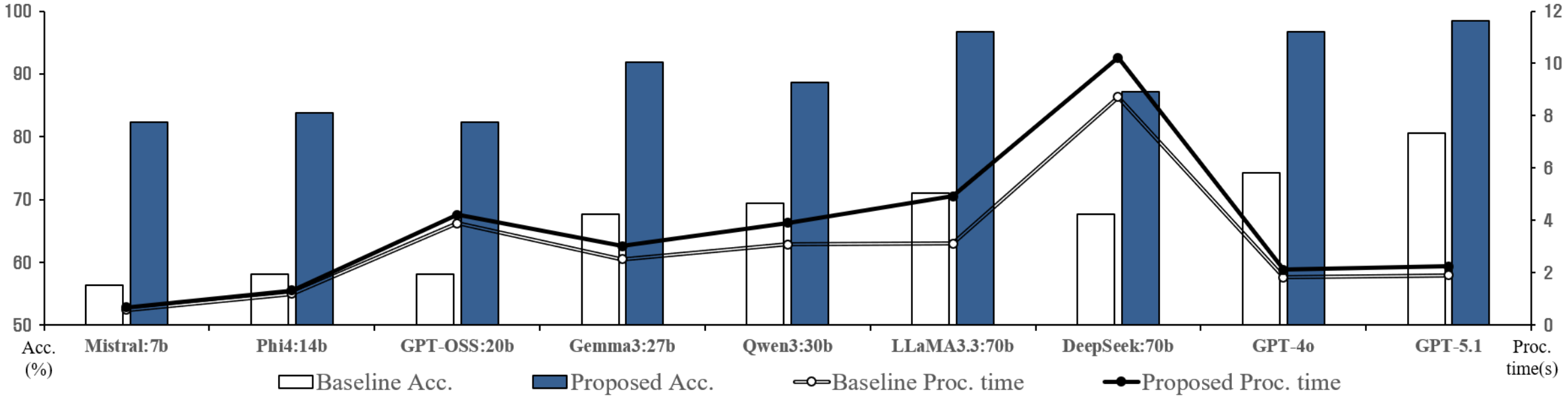
- Experiment 1: Comparison with Existing Approaches
  - Datasets
    - Previous work dataset in Tu et al. [10]
      - Deployment, SFC tasks with 40 intent data
    - Our work dataset
      - Deployment, SFC, Auto-scaling, Security configuration, Network configuration, combined tasks with 62 intent data
  - Methods
    - Rule-based(heuristic) approach vs LLM-based approach
    - Baseline LLM vs Previous studies vs Proposed

Methods		Rule-based (Heuristic)	LLM-based				
			Baseline LLM (LLaMA 3.3:70b)	Tu et al. [10] (few-shot, 9 examples)		Proposed (CoT + Few-shot, 5 examples)	
				LLaMA3.1:70b	LLaMA3.3:70b	LLaMA3.1:70b	LLaMA 3.3:70b
Acc.	Tu et al. [10] dataset (2 tasks, 40 data)	64.0	75.5	87.5	97.5	90	97.5
	Our dataset (5 tasks, 62 data)	59.7	72.6	-	-	90.3	96.7
Processing Time		0.41	1.83	6.91	7.20	4.57	4.93

Experiment 1: Performance comparison with existing studies

Intent Translation (3/3)

- Experiment 2: Comparison with various LLMs



Models		Mistral:7b	Phi4:14b	GPT-OSS:20b	Gemma3:27b	Qwen3:30b	LLaMA3.3:70b	DeepSeek:70b	GPT-4o	GPT-5.1
Baseline	Acc.	56.4	58.1	58.1	67.7	69.4	71	67.7	74.2	80.6
	Proc. time	0.58	1.19	3.89	2.52	3.09	3.12	8.71	1.83	1.91
Proposed	Acc.	82.3	83.8	82.3	91.9	88.7	96.7	87.1	96.7	98.4
	Proc. time	0.68	1.32	4.21	3.01	3.91	4.93	10.21	2.12	2.24

Experiment 2: Baseline LLMs vs Proposed methods (Accuracy, Processing time)

Policy Generation (1/4)

Evaluation for Policy Generation

Experiment 3 – Performance Comparison with various LLM models

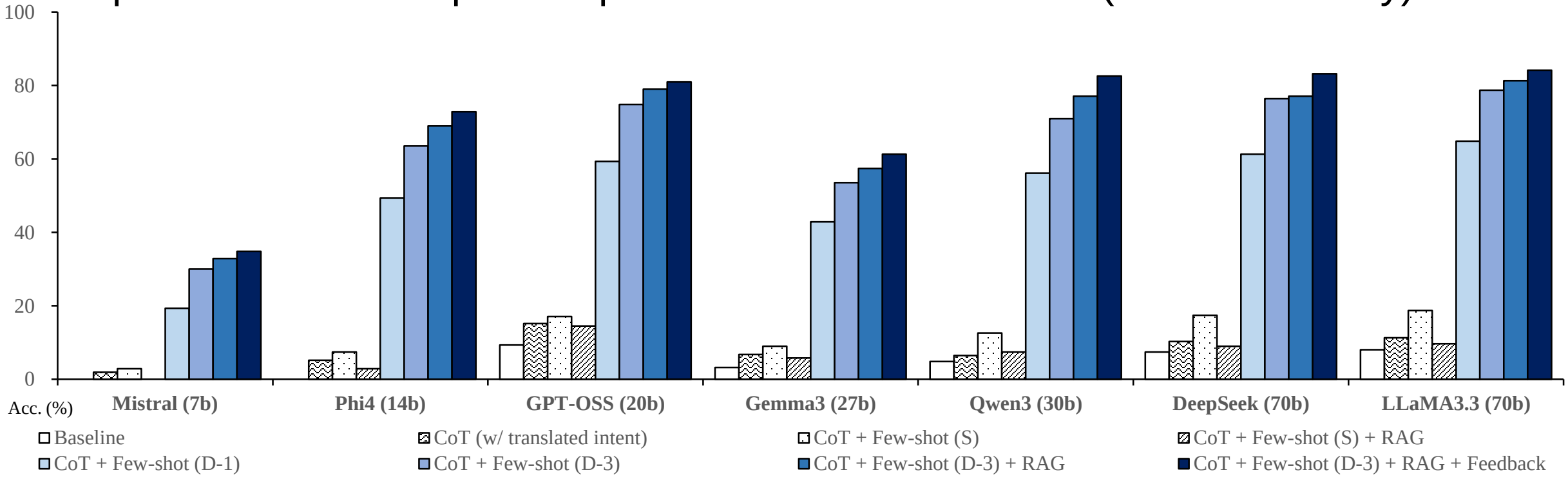
- Metrics
  - Validation accuracy: syntax and semantic validation accuracy by the verifier (provided in Appendix)
  - **Actual accuracy**: accuracy that policy is deployed and conduct the task aligning with the given intent
  - **Processing time**: time for LLM generation from intent input to YAML manifest output
- LLM models
  - 7 local models (from lightweight models to heavy capacity models) in Ollama platform
  - 8 cloud models in OpenAI and Ollama platforms (provided in Appendix)

Experiment 4 – RAG Performance Analysis

- Performance analysis of RAG module (Provided in Appendix)
- Metrics:
  - Embedding phase: embedding time, size of constructed DB
  - Retrieval phase: Cosine Similarity (CS), Precision, retrieval time

Policy Generation (2/4)

Experiment 3.1: Comparison per Local LLM and Method (actual accuracy)

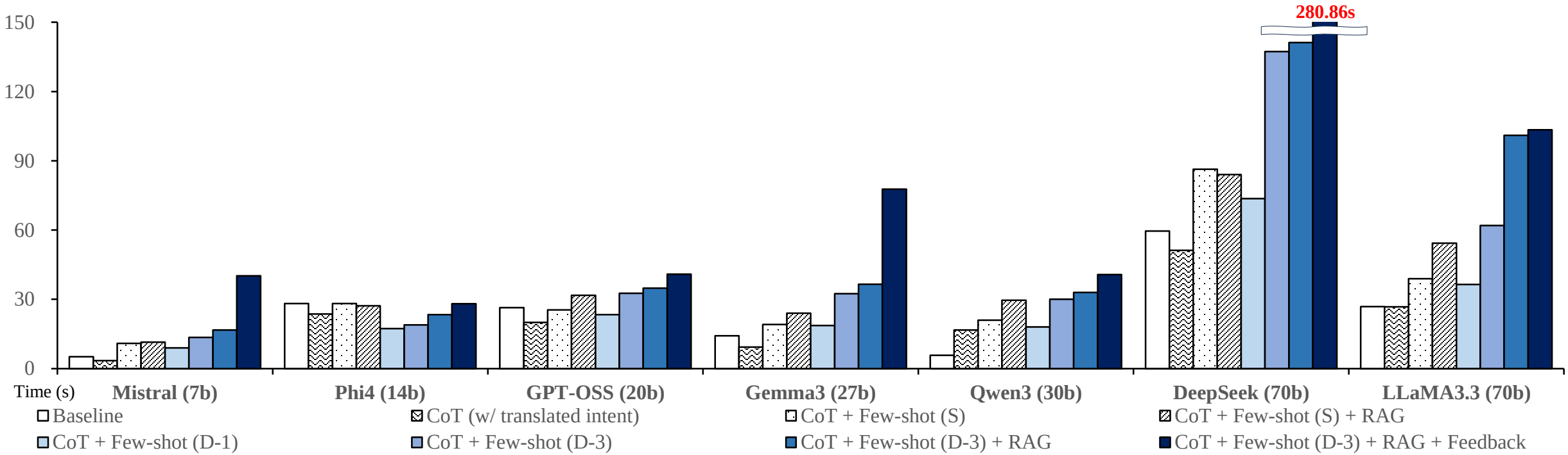


Models	Mistral:7b	Phi4:14b	GPT-OSS:20b	Gemma3:27b	Qwen3:30b	DeepSeek:70b	LLaMA3.3:70b
Baseline	**0.00**	**0.00**	**9.35**	**3.23**	**4.84**	**7.42**	**8.06**
CoT (w/ translated intent)	1.94	5.16	15.16	6.77	6.45	10.32	11.29
CoT + Few-shot (S)	2.90	7.42	17.10	9.03	12.58	17.42	18.71
CoT + Few-shot (S) + RAG	0.00	2.90	14.52	5.81	7.42	9.03	9.68
CoT + Few-shot (D-1)	**19.35**	**49.35**	**59.35**	**42.90**	**56.13**	**61.29**	**64.84**
CoT + Few-shot (D-3)	30.00	63.55	74.84	53.55	70.97	76.45	78.71
CoT + Few-shot (D-3) + RAG	32.90	69.03	79.03	57.42	77.10	77.10	81.29
CoT + Few-shot (D-3) + RAG + Feedback	**34.84**	**72.90**	**80.97**	**61.29**	**82.58**	**83.23**	**84.19**

Experiment 3.1: Actual accuracy comparisons for each method (local LLMs)

Policy Generation (3/4)

- Experiment 3.2: Comparison per Local LLM and method (processing time)



Models	Mistral:7b	Phi4:14b	GPT-OSS:20b	Gemma3:27b	Qwen3:30b	DeepSeek:70b	LLaMA3.3:70b
Baseline	5.11	28.16	26.42	14.17	5.73	59.58	26.80
CoT (w/ translated intent)	3.42	23.62	19.95	9.28	16.67	51.2	26.76
CoT + Few-shot (S)	**10.94**	**28.19**	**25.38**	**19.06**	**20.96**	**86.31**	**38.93**
CoT + Few-shot (S) + RAG	11.43	27.15	31.72	23.97	29.56	84.00	54.29
CoT + Few-shot (D-1)	**8.94**	**17.33**	**23.38**	**18.65**	**18.05**	**73.66**	**36.41**
CoT + Few-shot (D-3)	13.48	18.91	32.64	32.41	30.02	137.32	61.99
CoT + Few-shot (D-3) + RAG	16.72	23.38	34.85	36.53	32.98	141.24	101.02
CoT + Few-shot (D-3) + RAG + Feedback	**40.20**	**28.09**	**40.91**	**77.72**	**40.71**	**280.86**	**103.41**

Experiment 3.2: Processing time comparisons for each method (local LLMs)

Policy Generation (4/4)

Summary of Experiment 3 with Cloud LLMs

Methods	Actual Accuracy (%)	Processing time (s)
Baseline	**12.26**	**4.05**
CoT (w/ translated intent)	14.84	2.42
CoT + Few-shot (S)	**35.16**	**4.32**
CoT + Few-shot (S) + RAG	30.32	8.32
CoT + Few-shot (D-1)	**78.06**	**6.67**
CoT + Few-shot (D-3)	88.71	7.10
CoT + Few-shot (D-3) + RAG	90.97	9.38
CoT + Few-shot (D-3) + RAG + Feedback	**93.23**	**15.69**

Experimental results summary with Cloud LLM (GPT-5.1)

Summary of Experiment 4 (RAG Performance)

Embedding Model		K8s Docs		K8s Refs		Use-cases
		Document	Chunking	Document	Chunking	
Qwen3-embedding (8b)	**Cosine Similarity**	0.963	**0.966**	0.952	**0.965**	**0.968**
	Processing time(s)	0.275	0.200	0.243	0.193	0.188
	Precision	0.997	**0.929**	1.0	**0.910**	**0.997**

Experimental results summary in Experiment 4 (Qwen3-embedding)

Entire Framework Analysis

- Experiment 5: Entire Processing Time Analysis per Management Task
  - Analyze the processing time for each process by proposed methods and management tasks
  - Use best performing models in local and cloud LLMs
    - Local model: LLaMA3.3
    - Cloud model: GPT-5.1

Management Tasks		Deployment		SFC		Auto-scaling		Security Config.		Network Config.		Combined Tasks	
		Local	Cloud	Local	Cloud	Local	Cloud	Local	Cloud	Local	Cloud	Local	Cloud
Intent Translation		3.15	2.01	6.86	3.46	6.19	3.13	3.08	1.78	4.34	1.73	5.68	2.37
Policy Generation	RAG Retrievals	0.87	0.82	0.82	0.84	0.84	0.83	0.87	0.82	0.83	0.82	0.85	0.84
	Prompt Completion	1.09	1.56	1.1	1.40	1.1	1.50	1.09	1.49	1.08	1.35	1.09	1.49
	LLM Generation	34.75	4.91	303.81	19.41	87.66	4.82	39.58	1.99	29.31	2.38	81.4	6.18
	Validation & Retry	1.44	1.51	3.1	2.87	1.56	1.48	1.48	1.29	1.49	1.45	3.02	1.64
Policy Execution (deploy policy)		64.39	65.67	68.2	67.13	66.57	66.76	0.61	0.62	0.65	0.64	70.23	65.36
Total Processing Time		105.69	76.48	383.89	95.11	163.92	78.52	46.71	7.99	37.7	8.37	162.27	77.88

Experiment 5: Entire processing time analysis by tasks and processes with local and cloud LLMs

Result Analysis

Result Analysis

Summary

- Proposed methods **achieve higher accuracy** than baseline methods
- **LLM optimization techniques more affect the performances** than the sizes of the models
  - CoT, few-shot learning, RAG, feedback mechanism
- Overall **processing time varies by task complexity**, especially for deployment and SFC tasks

Limitations

- Larger dataset than related studies, still limited number of intent data
- Evaluation mainly focus on Kubernetes NFVI, need to extend other environments
- Optimization to reduce processing time for local LLMs

Conclusion

- Summary
- Future Work

Summary

- Intent-based NFV Management Methods using LLMs
  - The limitations of existing IBN approaches
    - Mainly focus on only intent translation or instance deployment
    - Provide still low accuracy for policy generation
  - Proposed Intent Translation and Policy Generation methods
    - Use LLM and domain adaptation techniques for network management domain
  - Evaluation
    - Analyze accuracy and processing time, compare with baselines
    - The proposed methods showed significant accuracy improvement and gradual increase of processing time

Future Work

- LLM and RAG Optimization
  - Improve the robustness for diverse management tasks and constraints
    - Extend the dataset to more diverse tasks and natural language expressions
  - Achieve the cloud-level accuracy and processing time with local LLM models
    - Apply fine-tuning, distillation, and prompt optimization for local LLM models
    - Optimize the lightweight LLM models to migrate the resource-constraint, edge environment
- Extend the proposed methods to other environments and tasks
  - Evaluate the proposed methods with multi-cloud and multi-domain environment
    - e.g., OpenStack-based NFVI, open-source platforms such as Open5GS, Open Air Interface (OAI), etc.
  - Define an intent abstraction layer across the heterogeneous infrastructure

Q & A

Publications (1/2)

International Journal Papers (3 papers)

1. **Jibum Hong**, Nguyen Van Tu, James Won-Ki Hong, A Comprehensive Survey on LLM-Based Network Management and Operations; International Journal of Network Management (IJNM, SCIE), vol. 35, issue 6, e70029, Nov. 2025.
2. Khizar Abbas, **Jibum Hong**, Nguyen Van Tu, Jae-Hyoung Yoo, James Won-Ki Hong; Autonomous DRL-based energy efficient VM consolidation for cloud data centers; Physical Communication, vol. 55, 101925, Dec. 2022.
3. Stanislav Lange, Nguyen Van Tu, Seyeon Jeong, Do-Young Lee, **Jibum Hong**, Hee-Gon Kim, Jae-Hyoung Yoo, James Won-Ki Hong, A network intelligence architecture for efficient VNF lifecycle management; IEEE Transactions on Network and Service Management (TNSM), vol. 18, pp. 1476–1490, Jun 2021.

International Conference Papers (9 papers)

1. **Jibum Hong**, Chungjun Lee, Dongnyeong Heo, Heeyoul Choi, Jae-Hyoung Yoo, James Won-Ki Hong, "Sequential Deep Learning Architectures for Anomaly Detection in Virtual Network Function Chains", The 12th International Conference on ICT Convergence (ICTC 2021), Jeju, Oct. 2021.
2. **Jibum Hong**, Suhyun Park, Jae-Hyoung Yoo, James Won-Ki Hong, "Machine Learning based SLA-Aware VNF Anomaly Detection for Virtual Network Management", 16th International Conference on Network and Service Management (CNSM 2020), Nov. 2020.
3. **Jibum Hong**, Seyeon Jeong, Jae-Hyoung Yoo, James Won-Ki Hong, "Design and Implementation of eBPF-based Virtual TAP for Inter-VM Traffic Monitoring", 14th International Conference on Network and Service Management (CNSM 2018), Rome, Italy, pp. 402-407, Nov. 2018.

International Patents (2 registered, 1 pending)

1. James Won-Ki Hong, Jae-Hyoung Yoo, **Jibum Hong**, “Traffic Categorization Method and Device”, US Patent No.: 17/768837, 2022.04.13 (Applicant: POSTECH)
2. James Won-Ki Hong, Jae-Hyoung Yoo, **Jibum Hong**, Suhyun Park, “Machine Learning-Based VNF Anomaly Detection System and Method for Virtual Network Management”, US Patent No.: 17/480070, 2021.09.20 (Applicant: POSTECH)
3. James Won-Ki Hong, Jae-Hyoung Yoo, **Jibum Hong**, “Method for Classifying Traffic and Apparatus Thereof”, Patent No.: PCT/KR2020/004905, 2020.04.10 (Applicant: POSTECH) (Pending)

Publications (2/2)

Domestic Journal Papers (6 papers)

1. **홍지범**, 홍원기, "NFV 관리를 위한 LLM 기반 Intent Translation 방법 연구", KNOM Review, Vol.28, No. 2, pp.1-12, Dec. 2025.
2. **홍지범,** 유재형, 홍원기, "딥러닝 기반 네트워크 공격 및 침입 탐지 방법 연구", KNOM Review, Vol.25, No. 2, pp.10-21, Dec. 2022.
3. **홍지범**, 정세연, 유재형, 홍원기, "가상 네트워크 트래픽 모니터링을 위한 eBPF 기반 Virtual TAP 설계 및 구현", KNOM Review, Vol. 21, No. 2, pp.1-9, Dec. 2018.

Domestic Conference Papers (12 papers)

1. **홍지범**, 홍원기, "네트워크 관리 자동화를 위한 LLM 기반 정책 및 네트워크 서비스 디스크립터 생성 연구", KNOM Conference 2025, Daejeon, Korea, April 24-25, 2025.
2. **홍지범**, 허동녕, 남석현, 유재형, 홍원기, "가상 네트워크 관리를 위한 기계학습 기반 네트워크 공격 및 침입 탐지 시스템 설계", KNOM Conference 2022, Chuncheon, Korea, May 12-13, 2022.
3. **홍지범,** 정세연, 남석현, 유재형, 홍원기, "머신러닝을 이용한 VNF 이상 탐지 및 고장 예측 기반 NFV 관리 시스템 설계", 2022 한국통신학회 동계종합학술발표회, Pyeongchang, Korea, Feb. 9-11, 2022.

Domestic Patents (3 registered, 5 pending)

1. 최건, **홍지범**, 조광래, “멀티모달 인공지능 기반의 척추질환 진단 정보 제공 방법 및 장치”, 10-2025-0141507, 2025.09.29 (출원인: 주식회사 올댓스파인)
2. 홍원기, 유재형, **홍지범**, Khizar Abbas, 정의동, "데이터 센터를 위한 심층강화학습 기반의 가상 머신 통합 관리 방법 및 제어장치", 10-2023-0082510, 2023.06.27 (출원인: 포항공과대학교 산학협력단).
3. 홍원기, 유재형, **홍지범**, 박수현, "가상 네트워크 관리를 위한 머신 러닝 기반 VNF 이상 탐지 시스템 및 방법", 제 10-2522005호 (10-2021-0018674), 2023.04.11 (2021.02.09) (출원인: 포항공과대학교 산학협력단).

Other Publications (9 domestic standards, 6 domestic software)

1. 홍원기, 유재형, **홍지범**, “제로터치 네트워크 및 서비스 관리(ZSM): 참조 아키텍처”, TTA, 표준번호: TTAE.ET-GS ZSM 002, 2023.12.06
2. 홍원기, 유재형, **홍지범**, “NFV 환경의 네트워크 공격 및 침입 트래픽 탐지 모델 및 성능평가 도구”, 한국저작권위원회, 등록번호: C-2023-016742, 2023.03.17
3. 홍원기, 유재형, **홍지범**, “오픈스택 기반 SLA 위반 탐지 모듈”, 한국저작권위원회, 등록번호: C-2021-028810, 2021.07.19

References

1. ETSI. Etsi gs nfv 006, network functions virtualisation (nfv) release 4; management and orchestration;architectural framework specification. https://docbox.etsi.org/ISG/NFV/Open/Publications_pdf/Specs-Reports/NFV%20006v4.5.1%20-%20GS%20-%20MANO%20Arch%20Fwk.pdf. Accessed on 2025-12-02.
2. Gartner. Hype cycle for enterprise networking, 2024. https://www.gartner.com/en/documents/5500595. Accessed on 2025-12-02.
3. Market.us. Global ai in networks market size, share and demand analysis report – industry segment outlook, market assessment, competition scenario, trends and forecast 2024-2033. https://market.us/report/ai-in-networks-market/. Accessed on 2025-12-02.
4. Kubernetes, "Kubernetes Technical Documents - Overview", https://kubernetes.io/docs/concepts/overview/. Accessed on 2025-12-02.
5. Kubernetes, "Kubernetes Technical Documents - Kubernetes Components", https://kubernetes.io/docs/concepts/overview/components/. Accessed on 2025-12-02.
6. K. Dzeparoska, J. Lin, A. Tizghadam and A. Leon-Garcia, "LLM-Based Policy Generation for Intent-Based Management of Applications," 2023 19th International Conference on Network and Service Management (CNSM), pp. 1-7. Oct, 2023.
7. Mekrache, Abdelkader, and Adlen Ksentini. "LLM-enabled intent-driven service configuration for next generation networks." 2024 IEEE 10th International Conference on Network Softwarization (NetSoft). IEEE, 2024.
8. J. Wang et al., "Network Meets ChatGPT: Intent Autonomous Management, Control and Operation," in Journal of Communications and Information Networks, vol. 8, no. 3, pp. 239-255, Sep. 2023.
9. S. Kou, C. Yang and M. Gurusamy, "GIA: LLM-Enabled Generative Intent Abstraction to Enhance Adaptability for Intent-Driven Networks," in IEEE Transactions on Cognitive Communications and Networking, vol. 11, no. 2, pp. 999-1012, April 2025.
10. Nguyen Van Tu, Sukhyun Nam, James Won-Ki Hong, "Intent-Based Network Configuration Using Large Language Models", International Journal of Network Management (IJNM), Nov. 2024.
11. Sukhyun Nam, Nguyen Van Tu, James Won-Ki Hong, "LLM-based VNF Deployment Automation in OpenStack Environment", 2025 IEEE/IFIP Network Operations and Management Symposium (NOMS 2025), May 2025.
12. IETF, "Intent classification", IETF RFC 9316, October 2022.