Node Efficiency Index (NEI): A Cost–Quality–Latency Trade-off Metric for Agentic LLM Systems

Author: Kaan Yedikardaşlar
Affiliation: Department of Software Engineering, Istinye University
Date: December 2025

Abstract

LLM-based workflows constructed with frameworks such as **LangChain** and **LangGraph** are typically composed of multiple modular nodes, including planners, retrievers, reasoning components, and tool-execution units. While these frameworks provide powerful abstractions for building agentic systems, their evaluation mechanisms largely rely on aggregate metrics such as total token usage or end-to-end latency. Such coarse-grained measurements fail to capture the marginal efficiency contribution of individual nodes and provide limited guidance when extending or refactoring workflows.

In this work, we propose a unified efficiency formulation that jointly considers execution time, token consumption, and output quality. Based on this formulation, we introduce the **Node Efficiency Index (NEI)**, a node-level metric that quantifies the intrinsic cost–benefit efficiency of a workflow component. **While efficiency measures workflow-level impact, NEI isolates the marginal return of a single node.**
NEI is framework-agnostic and can be applied to LangChain, LangGraph, or any modular LLM orchestration pipeline.

The proposed approach enables fine-grained analysis of node impact in agentic systems and supports cost-aware architectural decision-making.

Keywords: LLM Evaluation, LangChain, Workflow Optimization, Cost–Benefit Analysis, Agentic Systems.

1. Introduction

The rapid adoption of Large Language Models (LLMs) has shifted system design from single-prompt interactions to complex, multi-step workflows. Frameworks such as LangChain and LangGraph enable developers to construct *agentic systems*, where distinct nodes perform specialized roles including query rewriting, document retrieval, reasoning, and response synthesis.

As workflows evolve, developers frequently introduce additional nodes to improve reasoning depth or output quality. However, each added node inherently increases **latency** (ΔT) and **token cost** (ΔN). This raises a fundamental question:

> *Does the marginal improvement in output quality (ΔQ) justify the additional resource consumption?*

Current evaluation practices exhibit two key limitations:

1. **Aggregate Metrics:** Total tokens or total execution time obscure individual bottlenecks.
2. **Quality-only Metrics:** Tools such as **RAGAS**, **DeepEval**, or **Promptfoo** focus on correctness or faithfulness while ignoring the operational cost of intelligence.

To address this gap, we propose the **Node Efficiency Index (NEI)**, a unified scalar metric that quantifies the trade-off between quality, latency, and cost at the node level.

2. Theoretical Background

Evaluating heterogeneous systems requires comparing fundamentally different units: **seconds** (time), **tokens or dollars** (cost), and **scalar scores** (quality). Direct comparison of such quantities is not meaningful. NEI resolves this issue using **dimensionless normalization**.

Rather than relying on absolute differences, we compute relative changes:

$$\Delta x = \frac{x_{new} - x_{old}}{|x_{old}|}$$

This formulation converts all metrics into unitless ratios, enabling linear combinations across dimensions. As a result, a 10% increase in latency becomes mathematically comparable to a 10% increase in quality, subject to user-defined sensitivity weights.

3. The Efficiency Model

3.1 Fundamental Variables

Let:

- T: Total execution time (latency, in seconds)
- N: Token usage or monetary cost
- Q: Output quality score, normalized to $[0, 1]$

Quality scores may be derived from automated evaluators (e.g., RAGAS, DeepEval), human-annotated preference judgments, or any consistent scoring mechanism normalized to the $[0, 1]$ range.

3.2 General Efficiency Equation

The global efficiency score is defined as:

$$Efficiency = (\gamma \cdot \Delta Q - \alpha \cdot \Delta T - \beta \cdot \Delta N) \times 100$$

where $\alpha, \beta, \gamma \geq 0$ are user-defined sensitivity weights reflecting deployment priorities (e.g., latency-critical vs. quality-critical systems). The weights may be normalized but are not required to sum to one.

3.3 The Node Efficiency Index (NEI)

To evaluate the return-on-investment (ROI) of a specific node, we define the **Node Efficiency Index (NEI)**. Two scenarios are considered.

Case A: Expansion (Adding Nodes)

When adding a node (e.g., a re-ranking or verification module), both latency and cost are expected to increase:

$$NEI_{expansion} = \frac{\Delta Q}{\alpha \cdot \Delta T + \beta \cdot \Delta N + \epsilon}$$

where ϵ is a small constant to avoid division by zero.

- **NEI > 1.0**: Efficient (quality gain outweighs cost)

- **NEI < 1.0**: Inefficient (diminishing returns)

Case B: Optimization (Reducing Cost)

When optimizing a workflow (e.g., caching, quantization, or pruning), ΔT or ΔN may be negative. In this case, we consider the net benefit:

$$Score_{opt} = \Delta Q - (\alpha \cdot \Delta T + \beta \cdot \Delta N)$$

A positive score indicates a successful optimization, either through quality improvement or cost reduction without significant quality loss.

4. Experimental Evaluation (Case Study)

Scenario: Baseline RAG pipeline vs. Candidate RAG pipeline with an additional re-ranking node.

Metric	Baseline	Candidate	Normalized Delta (Δ)
Latency (T)	1.20 s	1.80 s	+0.50 (+50%)
Tokens (N)	800	850	+0.0625 (+6.25%)
Quality (Q)	0.75	0.88	+0.173 (+17.3%)

Configuration: $\alpha = 0.33$, $\beta = 0.33$, $\gamma = 0.34$

$$Efficiency = (0.34 \cdot 0.173 - 0.33 \cdot 0.50 - 0.33 \cdot 0.0625) \times 100 \approx -12.68$$

Conclusion: In a latency-sensitive configuration, the 50% increase in execution time outweighs the observed quality improvement, resulting in a negative efficiency score.

5. Limitations

NEI assumes stable and consistent quality scoring functions and homogeneous execution environments. Variance in API latency, stochastic model outputs, dynamic pricing models, and inter-

node dependencies are not explicitly modeled. Future work may incorporate uncertainty-aware or probabilistic extensions of NEI.

6. Conclusion

The Node Efficiency Index (NEI) provides a principled method for evaluating architectural decisions in agentic LLM systems. By unifying quality, latency, and cost into a single dimensionless metric, NEI enables developers and researchers to make informed, cost-aware decisions when extending or optimizing workflows.

References

1. LangChain AI. *LangChain Documentation.* 2024.
2. Es, S., et al. *RAGAS: Automated Evaluation of Retrieval Augmented Generation.* 2023.
3. DeepEval. *Open-Source LLM Evaluation Framework.* 2024.
4. Promptfoo. *CLI for Testing LLM Prompts.* 2024.