Check for updates

Journal of KIIT. Vol. 24, No. 10, pp. 25-32, Oct. 31, 2026. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2026.24.10.25

보존 신호 기반 콘텐츠 인지 적응형 다층 토큰 압축 게이트웨이

김 경 민*

A Preservation-Signal-Guided, Content-Aware Adaptive Multi-Layer Token Compression Gateway for Large Language Models

Gyeongmin Kim*

요 약

대형언어모델(LLM)의 운용 비용과 지연시간은 처리 토큰 수에 비례하지만, 기존 압축 기법은 입력 특성과 압축 강도에 따라 과제 성능을 크게 저하시킬 수 있다. 본 논문은 26편의 1차 문헌을 검토하고, 입력 압축·컨텍스트 선별·캐시 재사용·출력 라우팅을 기능적으로 구분한 4층 게이트웨이와 보존 신호 기반 적응 가드를 제안한다. 결정론적 재현 실험에서 도구 출력과 RAG 컨텍스트의 토큰을 각각 51.35%와 42.12% 절감하였다. 로컬 모델 탐색 실험에서 GSM8K 정답률은 기준선 0.500, 단순 압축 0.167, 가드 압축 0.208이었고, MBPP Pass@1은 각각 0.700, 0.550, 0.650, HotpotQA 정보이득 선별은 컨텍스트를 60.02% 줄이며 F1을 0.595에서 0.511로 바꾸었다. 보존 신호는 압축 위험을 줄이는 제어 변수이지만 과제 성능을 무조건 보장하지는 않으며, 고밀도 추론 입력에는 압축 후퇴가 필요하다.

Abstract

The operating cost and latency of Large Language Models (LLMs) scale with processed tokens, but compression can substantially degrade task performance depending on input type and compression strength. We review 26 primary studies and propose a four-layer gateway that functionally separates input compression, context selection, cache reuse, and output/routing control, coupled with an adaptive guard driven by preservation signals. Deterministic reproduction experiments reduced tokens by 51.35% for tool output and 42.12% for RAG context. In exploratory local-model experiments, GSM8K accuracy was 0.500 (baseline), 0.167 (naive), and 0.208 (guarded); MBPP Pass@1 was 0.700, 0.550, and 0.650; information-gain selection reduced HotpotQA context by 60.02% while changing F1 from 0.595 to 0.511. Preservation signals are useful control variables for reducing compression risk but do not guarantee task performance; reasoning-dense inputs may require complete backoff.

Keywords

large language model, token efficiency, prompt compression, retrieval-augmented generation, adaptive routing

* 삼정데이타서비스(주) 차장
- ORCID: https://orcid.org/0009-0008-7075-1455

· Received: Jul. 02, 2026, Revised: Jul. 27, 2026, Accepted: Jul. 30, 2026

· Corresponding Author: Gyeongmin Kim
SamJung Data Service Co., Ltd., 27, Digital-ro 34-gil, Guro-gu, Seoul, Korea
Tel.: +82-2-2029-0156, Email: formin97@gmail.com

Ⅰ. 서 론

대형언어모델(LLM, Large Language Model) 기반 서비스가 확산되면서 API 과금과 응답 지연이 운영의 핵심 제약이 되었다. 두 비용 모두 처리 토큰 수에 비례하므로, "동일한 작업을 더 적은 토큰으로" 수행하는 토큰 효율화가 직접적인 비용·속도 개선 수단이 된다. 실무에서는 RTK[1]와 같이 셸/도구 출력을 압축하는 프록시, Headroom[2]과 같이 콘텐츠 유형별로 컨텍스트를 압축하는 시스템이 등장하여 60~95%의 토큰 절감을 보고한다.

그러나 이러한 절감이 품질을 보존하는가는 별개의 문제다. 공격적 압축이 명령어를 파괴하면 코드 생성 정답률이 0.56에서 0.02로 붕괴하고, 단문 수리 추론에서는 절반 수준의 압축만으로 정확도가 3분의 1가량 하락한다는 보고가 있다(2.6절). 이러한 급락은 기존 기법의 동작 방식에서 비롯된다. 기존 입력 압축은 대리 모델의 자기정보량이나 학습된 토큰 분류 점수로 단위별 중요도를 매긴 뒤, 사전에 정한 목표 압축률만큼 하위 단위를 일괄 삭제하는 개방루프 구조여서, 삭제 결과가 과제 요건을 여전히 충족하는지 확인하는 단계가 없다[3]-[5]. 이때 중요도 점수는 말뭉치 통계 기준이므로, 출현 확률은 낮지 않지만 과제 수행에 필수적인 피연산자·부정어·식별자·형식 지시문이 삭제 대상에 포함될 수 있다. 다단계 추론과 코드 생성은 이러한 소수 토큰에 기능적으로 의존하므로, 피연산자 하나만 빠져도 산술 사슬 전체가 끊어지듯 성능은 압축률에 비례해 완만히 감소하지 않고 임계점을 넘는 순간 급락한다(4.5절 실측에서 고정 비율 압축은 GSM8K 계산에 필요한 숫자의 28%를 파괴하였다). 삭제된 정보는 생성 단계에서 복구할 수 없으므로, 압축을 적용하기 전에 필수 토큰의 생존을 신호로 측정하고 임계 위반 시 압축을 후퇴시키는 폐루프 가드가 필요하다는 것이 본 논문의 문제 제기다. 본 논문은 다음 두 질문을 다룬다. 첫째, 어떤 조건에서 토큰 절감이 과제 성능을 근사 보존하는가. 둘째, 보존 신호를 이용하여 압축 위험을 시스템 수준에서 제어하는 아키텍처는 무엇인가.

본 논문의 기여는 다음과 같다. 1) 6개 각도·26편 1차 문헌(참고문헌 [3]-[28]; 도구 저장소 [1][2] 제외)의 정량 근거를 적대적 검증으로 정리하고, "무손실 압축"류 과장 주장을 반증한다(Ⅱ장). 2) 성능 보존이 신호의 추정 정확도에 조건부로 의존한다는 설계 가설을 제시한다(3.1절). 3) 이 가설을 구현하는 콘텐츠 인지 적응형 다층 압축 게이트웨이와 보존 제약 기반 적응 가드를 설계한다(3.2~3.4절). 4) 기능적으로 구분한 층의 조합과 가드의 압축 후퇴 동작을 재현 실험과 실제 모델 정확도 측정으로 탐색 평가한다(Ⅳ장). 질문 1은 Ⅱ장의 문헌 검증과 4.5~4.6절의 실측으로, 질문 2는 Ⅲ장의 설계와 4.3~4.6절의 단계별 평가로 각각 다룬다. 기존 연구가 개별 기법의 압축 성능에 집중한 것과 달리, 본 논문은 이들을 보존 신호의 측정이라는 단일 관점으로 통합하고 신호 제약 아래 압축 강도를 자동 조정하는 게이트웨이 계층을 제안한다는 점에서 차별화된다.

Ⅱ. 관련 연구

2.1 입력측 프롬프트 압축

LLMLingua[3]는 예산 컨트롤러, 토큰 단위 반복 압축, 분포 정렬의 coarse-to-fine 구조로 최대 20배 압축에서 성능 손실을 최소화한다. LLMLingua-2[4]는 압축을 토큰 분류로 정식화하여 2~5배 압축과 1.6~2.9배 지연 단축을 달성한다. Selective Context[5]는 자기정보가 낮은 단위를 제거한다. 소프트 프롬프트 계열인 ICAE[6]는 4배 컨텍스트 압축을 소수 메모리 슬롯으로 달성하며, 500xCompressor[7]는 고압축에서 능력의 62~72% 보존에 그친다. 율-왜곡 관점의 분석[8]은 현행 기법과 최적 전략 사이의 큰 격차와 질의 인지 압축의 우월성을 보인다.

2.2 검색 컨텍스트 압축

RECOMP[9]은 검색 문서를 6% 압축률까지 요약하며 손실을 최소화한다. LongLLMLingua[10]는 질의 인지 압축으로 NaturalQuestions 정확도를 21.4% 향상시키며 약 4배 토큰을 절감한다. Provence[11]는 가지치기를 시퀀스 라벨링으로 정식화하여 reranking과 통합하고, 정보이득 가지치기(IGP)[12]는 관련도

대신 생성기 정보이득으로 문단을 선별하여 최종 입력 토큰을 76~79% 줄이면서 QA 품질을 향상시킨다. 압축 기법 특성화 연구[13]는 추출형 압축이 최대 10배 압축에서 손실을 최소화함을 보인다.

2.3 캐시 재사용

프롬프트 캐싱[14][15]은 반복 프리픽스의 캐시 읽기 비용을 기본 입력 단가의 0.1배로 낮추어 비용 최대 90%, 지연 최대 85%를 절감하며, 출력 생성에 영향을 주지 않는다(출력 중립).

2.4 출력측 추론 길이 제어

Chain of Draft[16]는 단계별 최소 초안으로 토큰을 평균 80% 절감하면서 정확도를 유지한다. TALE[17]은 토큰 예산을 프롬프트에 명시하고, 저엔트로피 단계 가지치기[18]는 CoT(Chain-of-Thought) 중간 단계의 80%를 제거한다. 정보이론 기반 적응 정지[19]는 토큰 50.8% 절감과 정확도 1.1%p 향상을 동시에 달성하며, CAR[20]은 신뢰도가 낮을 때만 추론을 트리거한다. 한편 CRANE[21]은 엄격한 제약 디코딩이 추론을 저하시킴을 보여, 구조화 출력이 항상 이득이 아님을 시사한다.

2.5 시스템측 라우팅과 토크나이저

RouteLLM[22][23]은 질의별 강/약 모델 라우팅으로 GPT-4 성능의 95%를 26% 라우팅과 85% 이상의 비용 절감으로 달성하고, FrugalGPT[24]는 캐스케이드로 최대 98% 비용 절감을 보고한다. 토크나이저 fertility 분석[25]은 단어당 토큰 수가 정확도 분산의 20~50%를 설명함을 보인다.

2.6 한계와 반례

정보 보존 분석[26]은 압축이 장문 세부 정보를 잃어 hallucination을 유발하고, 입도를 높이면 EM(Exact Match)/F1이 15~20점 회복(압축비 하락)됨을 보인다. 단문 추론[27]에서는 0.5 압축비가 GSM8K 정확도를 0.337에서 0.22 수준으로 떨어뜨리고, 명령어 파괴 시[28] MBPP Pass@1이 0.56에서 0.02로 붕괴한다. 이 반례들은 모두 무엇을 버려도 되는지를 측정하지 못한 데서 비롯된다. 또한 본 검증에서 압축 충실도의 "보장"[4], 원문 정보의 "포괄적 보존"[6], 캐시 응답의 "비트 단위 동일"[14] 등 3건의 무손실류 주장은 원문 근거가 부족한 과장으로 반증되었다.

2.6절의 반례와 2.1~2.5절의 성공 사례를 대조하면, 실패는 무엇을 버려도 되는지를 추정하는 신호의 부재에서 비롯되고, 성공 기법은 모두 보존 측정 신호(정보이득, 시퀀스 라벨, 신뢰도)를 내장한다는 공통점이 드러난다. 본 논문은 이 신호들을 개별 기법 내부가 아니라 시스템 수준의 공통 제어로 일반화한다. 각 절의 기법군은 제안 게이트웨이의 계층과 대응된다. 2.1절은 L1 입력 압축, 2.2절은 L2 컨텍스트 선별, 2.3절은 L3 캐시 재사용, 2.4절과 2.5절은 L4 출력·라우팅의 기반 기법이다. 이는 Ⅲ장의 설계로 이어진다.

Ⅲ. 제안 방법

3.1 설계 가설: 보존 신호에 따른 조건부 제어

설계 원리 1(가정 기반). 보존 신호 s가 제거 정보의 작업 기여도 손실을 유효하게 상계하고 압축 연산 C가 임계 위반 후보를 배제할 때, 위험은 신호의 보정 오차 범위에서 제한된다. 본 프로토타입의 s는 휴리스틱 프록시이므로 이를 자동 보장하지 않으며, 4.6절처럼 정확도 손실이 남을 수 있다.

보존 신호의 예와 산정 방식은 다음과 같다. (i) 명령어 생존율 Ψ[28]: 압축 전 세그먼트에서 추출한 필수 토큰 집합 R(내용어, 숫자, 식별자)에 대해 압축 후 잔존 토큰과의 비율 $\Psi=|R\cap R'|/|R|$로 계산한다. (ii) 생성기 정보이득 IG[12]: 문단 추가에 따른 생성기 출력 불확실성의 감소량으로, 로짓 접근이 불가한 블랙박스 환경에서는 질의 용어의 신규 커버리지와 위험 가중치로 근사한다. (iii) 응답 신뢰도[20]: 생성 로그확률 기반 perplexity의 역수로 산정한다. (iv) 압축 입도 g[26]. 이하에서 "보존"은 무압축 기준선과의 동등 수준, "근사 보존"은 절감 대비 소폭 하락, "붕괴"는 상대 50% 이상 하락을 가리키는 것

으로 용어를 통일한다.

3.2 전체 구조: 기능 분리형 4층 게이트웨이

입력 요청은 순차 4층으로 처리하며 각 층은 개별 설정할 수 있다. 기능은 구분되지만 동일 요청에서 상호작용하므로 통계적 독립성은 가정하지 않는다(그림 1).

L1 입력 압축 층은 콘텐츠 라우터가 세그먼트를 코드(AST, Abstract Syntax Tree), JSON, 자연어, 명령어로 분류하고, 자연어에는 LLMLingua류, 구조화 데이터에는 구조 보존 압축을 적용하며, 명령어 세그먼트는 Ψ 가드로 보호한다(압축 금지 또는 최소화). L2 컨텍스트 선별 층은 RAG(Retrieval-Augmented Generation) 컨텍스트를 정보이득/시퀀스 라벨로 선별하여, 관련도가 아닌 생성기 효용으로 가지치기한다. L3 캐시 재사용 층은 반복 프리픽스를 정렬하여 캐시 적중률을 높이며, 출력 중립이므로 가장 안전한 무손실 절감이다. L4 출력·라우팅 층은 신뢰도 기반 적응형 추론 길이 제어와 질의 난이도 기반 모델 라우팅을 결합한다. 그림 1의 모듈 번호는 본문과 다음과 같이 대응된다. 콘텐츠 라우터(110)는 3.4절의 분류를, 입력 압축(120), 컨텍스트 선별(130), 캐시 재사용(140), 출력 라우팅(150)은 각각 L1~L4를, 보존 가드(160)와 가역 저장부(170)는 3.3절의 적응 가드와 CCR을 구현한다.

3.3 보존 제약 기반 적응 가드

각 층 i는 보존 신호 s_i, 임계값 τ_i, 위험 예산 ε로 압축 강도를 자동 결정한다(적응 동작). 세그먼트 x에 대해 토큰 유지율 $r \in \{0.4, 0.55, 0.7, 0.85, 1.0\}$의 후보를 강한 압축부터 순차 평가하여, 압축 후보의 보존 신호가 $s_i(x, r) \geq \tau_i$이고 예측 품질 저하 $1 - s_i(x, r) \leq \varepsilon$를 만족하는 가장 낮은 r를 채택한다. 어떤 후보도 조건을 만족하지 못하면 r=1(원문)로 후퇴한다. 참조 구현의 기본값은 τ_i=0.8, ε=0.2이며, 두 값은 검증 집합의 절감률-보존율 곡선에서 재조정할 수 있다. CCR(Compress-Cache-Retrieve)은 가역 압축 저장소로, 제거되는 원문을 고유 식별자와 함께 로컬에 보관하고 압축 텍스트에는 식별자만 남기며, 모델이나 후속 단계가 식별자로 요청하면 원문을 반환하는 손실 복구 경로다. 이는 생성 후 과제 점수 피드백이 아니라 생성 전 신호 기반 제어다.

가드는 무손실을 보장하지 않고 임계 위반 시 압축을 줄인다. 신호가 잘못 보정되면 부분 압축 입력에서 과제 손실이 발생할 수 있다.

3.4 콘텐츠 인지 라우팅

세그먼트 유형별 최적 기법 매핑은 다음과 같다. 코드는 AST 구조 압축, JSON은 스키마 기반 축약, 자연어는 토큰 중요도 기반 압축, 명령어와 시스템 프롬프트는 압축 금지와 캐싱, 검색 문서는 정보이득 선별을 적용한다[2].

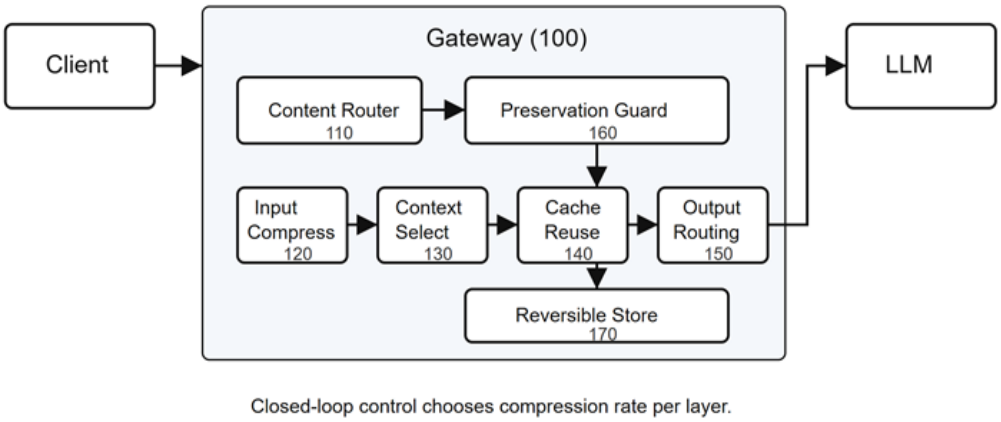


그림 1. 제안 게이트웨이의 기능 분리형 4층 구조
Fig. 1. Four functionally distinct layers of the proposed gateway

Ⅳ. 평 가

본 장의 실험은 재현 가능하도록 구성하였다. 4.3~4.5절은 외부 LLM API 없이 결정론적으로 실행되며, 4.6절은 로컬 LLM 서버(OpenAI 호환)를 사용한다. 고정 입력과 전체 실측 결과 데이터는 GitHub 저장소(https://github.com/formin/token-compression-gateway)로 제공한다.

4.1 절감의 곱셈성

기능적으로 구분한 층의 이상적 절감 상한은 층별 잔존 비율의 곱으로 계산할 수 있다. 다만 L2 컨텍스트 4배 절감[10], L4 출력 5배 절감[16], L3 캐시 입력 비용 0.1배[14]를 조합한 값은 워크로드 수준의 예시적 상한이며 실측 통합 배수가 아니다. 실제 절감은 입력 대 출력 비율과 층 간 상호작용에 의존하므로 4.3~4.5절에서는 층별 결과를 분리하여 보고한다.

4.2 임계 위반 시 압축 후퇴

명령어 파괴 시 MBPP Pass@1은 0.56에서 0.02로 하락한다[28]. 가드는 Ψ가 임계 이하인 세그먼트를 원문으로 후퇴시킨다. 압축률 0은 입력 동일성만 보장하며 부분 압축의 과제 성능은 보장하지 않는다. GSM8K 결과가 이 한계를 보인다.

4.3 소규모 재현 실험

외부 LLM API를 사용하지 않는 고정 입력 기반 재현 실험을 구성하였다. 입력은 (1) 테스트 로그와 검색 결과가 섞인 도구 출력, (2) 오류·보안·일반 설명 문단이 섞인 RAG JSON, (3) 명령어 보존 여부를 확인하는 짧은 프롬프트로 구성하였다. 비교군은 무압축 기준선, 도구 출력 요약(L1), CCR 기반 컨텍스트 선별(L2), 보존 가드 적용/미적용이다. 표 1에서 Savings는 토큰 절감률, Quality는 보존 판정, Survival은 명령어 생존율 Ψ, naive는 가드 미적용, guard는 가드 적용을 뜻한다. 표 1은 결과를 보인다.

표 1. 소규모 재현 실험 결과
Table 1. Results of the small-scale reproduction

Scenario	Method	Savings	Quality	Survival
tool-output	summary (L1)	51.35%	preserved	-
rag-context	CCR (L2)	42.12%	preserved	-
safe-log	naive	24.14%	risk	0.67
safe-log	guard	0.00%	preserved	-
risky-code	naive	20.00%	risk	0.67
risky-code	guard	0.00%	preserved	-

도구 출력과 RAG 입력은 42~51% 절감되었다. 단순 압축은 20~24% 절감했지만 명령어 생존율 0.67로 위험 판정되었고, 가드는 원문으로 후퇴했다. 이는 후퇴 동작을 확인한 것이며 과제 정확도는 4.6절에서 평가한다.

4.4 통합 게이트웨이 프로토타입

3.2절의 기능 분리형 4층과 3.3절의 보존 가드, 3.4절의 콘텐츠 라우팅을 하나의 파이프라인으로 결합한 참조 구현을 작성하고(외부 LLM API 미사용, 결정론적), 반복 시스템 프롬프트, 저난이도 질의, 고난이도 추론 질의, RAG 컨텍스트로 구성된 세션에 적용하였다. 표 2에서 Routing은 선택된 모델 경로, Backoff는 가드 후퇴 횟수다. 표 2는 결과를 보인다.

표 2. 통합 게이트웨이 프로토타입 결과
Table 2. Results of the integrated gateway prototype

Request	Routing	Savings	Backoff
easy summary	small	75.40%	0
hard reasoning	large	22.15%	0
easy listing	small	67.69%	1
L3 cache reuse	-	55.03%	-

통합 결과는 Ⅲ장의 제어 흐름을 결정론적으로 확인한다. 휴리스틱 라우터는 저난이도 질의에 소형 모델과 40토큰 예산을, 고난이도 질의에 대형 모델과 200토큰 예산을 지정하였다. 이 수치는 실제 생성 정확도가 아니라 라우팅 및 예산 선택의 출력이다. 또한 필수어 보존, 조건 미충족 세그먼트의 원문 후퇴, 저위험 문단의 식별자 대체와 복원, 반복 프리픽스의 캐시 재사용 동작을 확인하였다.

4.5 실제 벤치마크 확장

합성 입력을 넘어, 게이트웨이의 실제 신호·가드·선별 함수를 공개 벤치마크 데이터(GSM8K 100문항, MBPP 100문항, HotpotQA distractor 50문항)에 적용하였다. 외부 LLM API 없이 결정론적으로 재현되며, 특히 HotpotQA의 정답 근거(Supporting facts) 라벨을 이용하여 모델 채점 없이 정답 근거 보존율을 직접 측정하였다.

HotpotQA에서 10개 문단 중 4개를 선별할 때, 제안하는 생성기 정보이득(IG) 선별의 정답 근거 문단 보존율은 0.730으로 어휘 유사도 방식(0.520)과 위치 기반 방식(0.370)을 크게 상회하였고, 컨텍스트 토큰은 문항당 평균 58.85% 절감되었다. GSM8K에서 가드 없는 단순 압축은 계산에 필요한 숫자의 28%를 파괴(숫자 보존율 0.719)한 반면, 숫자를 필수 토큰으로 보존하는 가드 압축은 보존율 1.000을 유지하면서 오히려 더 큰 절감(49.07% 대 32.79%)을 달성하였다. 이는 버리면 안 되는 토큰을 먼저 고정하면 잔여 불용어를 더 공격적으로 제거해도 신호 조건이 유지되기 때문이며, 4.6절의 실제 모델 평가에서 가드의 절감률이 단순 압축보다 높은 것도 같은 원리다. MBPP에서 가드 없는 압축은 명령어 생존율을 0.650으로 떨어뜨린 반면 가드 압축은 0.993을 유지하였다. 표 3은 결과를 요약한다.

표 3. 실제 벤치마크 보존 신호 측정 결과
Table 3. Preservation-signal results on real benchmarks

Benchmark	Metric	Naive	Guarded
HotpotQA	evidence retention	0.520	0.730
GSM8K	digit survival	0.719	1.000
MBPP	instruction survival	0.650	0.993

표 4. 실제 모델(Gemma 3 4B) GSM8K 정답률
Table 4. GSM8K accuracy and wilson 95% CI (Gemma 3 4B)

Condition	Accuracy	Savings
baseline	0.500 (0.364 – 0.636)	0.00%
naive	0.167 (0.087 – 0.296)	32.92%
guarded	0.208 (0.117 – 0.343)	44.35%

4.6 실제 모델 기반 과제 정확도

Gemma 3 4B(temperature=0)로 GSM8K(n=48), HotpotQA(n=24), MBPP(n=20)를 탐색 평가했다. 소표본 집계 비교이므로 인과 효과를 주장하지 않으며 비율에는 Wilson 95% 신뢰구간을 제시한다. 표 4는 GSM8K 정답률과 토큰 절감률을 보인다.

MBPP Pass@1은 기준선 0.700(95% CI 0.481 – 0.855), 단순 0.550(0.342 – 0.742), 가드 0.650(0.433 – 0.819)이었다(가드 절감 23.70%). 구간이 겹쳐 회복은 예비 관측이다. HotpotQA는 컨텍스트 60.02% 절감, EM 0.500(0.314 – 0.686)→0.458(0.279 – 0.649), F1 0.595→0.511이었다. GSM8K는 기준선 0.500(0.364 – 0.636), 단순 0.167(0.087 – 0.296), 가드 0.208(0.117 – 0.343)이었다. 26B(n=12)도 같은 방향이었으나 일반화하지 않는다.

과제별 관측은 압축 후퇴 규칙의 필요성을 지지하지만 성능 보존을 확증하지 않는다. 표본이 작고 항목별 대응 결과가 없어 paired 검정을 하지 못했다. 향후 사전등록 대표본·장문 평가와 신호 보정이 필요하다.

V. 결론 및 향후 과제

토큰 절감의 성능 보존은 입력 유형과 보존 신호의 보정 정확도에 의존한다. 본 논문은 이를 다루는 기능 분리형 4층 게이트웨이와 보존 제약 기반 적응 가드를 제시했다. 탐색 평가에서 가드는 임계 위반 시 후퇴했지만 성능을 항상 보존하지는 않았다. 본 연구의 학술적 파급효과는 두 가지다. 첫째, 흩어져 있던 이질적 토큰 효율화 기법들을 보존 신호 측정이라는 단일 제어 관점으로 통합하여, 후속 연구를 시스템 수준에서 비교·조합할 수 있는 공통 좌표를 제공한다. 둘째, 어떤 신호가 임계를 벗어날 때 어떤 과제가 붕괴하는지 압축의 실패 조건을 실측으로 정량화하여, “언제 압축하지 말아야 하는가”를 압축 연구의 일급 설계 질문으로 제기하였다. 실용적 파급효과로, 제안 게이트웨이는 기존 기법과 모델을 대체하지 않고 감싸는 계층이므로 운영 중인 LLM 파이프라인에 재학습 없이 점진 도입할 수

있고, 가역 저장부(CCR)가 삭제 원문의 복구 경로를 보존하므로 처리 이력의 감사가 필요한 운영 환경에도 적용할 수 있다. 이 성과는 소표본·단일 소형 모델·휴리스틱 신호라는 검증 범위 안에서 확인된 것이므로, 대표본 대응 검정과 효과크기 보고, 장문 벤치마크 확장, 신호의 학습 기반 보정으로 검증 범위를 넓히는 것이 향후 과제다.

References

[1] RTK: Rust Token Killer, https://github.com/rtk-ai/rtk. [accessed: Jul. 01, 2026]

[2] Headroom, AI Context Compression, https://github.com/headroomlabs-ai/headroom. [accessed: Jul. 01, 2026]

[3] H. Jiang, Q. Wu, C. Y. Lin, Y. Yang, and L. Qiu, "LLMLingua: Compressing Prompts for Accelerated Inference of Large Language Models", Proc. EMNLP 2023, Singapore, pp. 13358-13376, Dec. 2023. https://doi.org/10.18653/v1/2023.emnlp-main.825.

[4] Z. Pan, Q. Wu, H. Jiang, M. Xia, X. Luo, J. Zhang, Q. Lin, V. Ruhle, Y. Yang, C. Y. Lin, H. V. Zhao, L. Qiu, and D. Zhang, "LLMLingua-2: Data Distillation for Efficient and Faithful Task-Agnostic Prompt Compression", Findings of ACL 2024, Bangkok, Thailand, pp. 963-981, Aug. 2024. https://doi.org/10.18653/v1/2024.findings-acl.57.

[5] Y. Li, B. Dong, F. Guerin, and C. Lin, "Compressing Context to Enhance Inference Efficiency of Large Language Models", Proc. EMNLP 2023, Singapore, pp. 6342-6353, Dec. 2023. https://doi.org/10.18653/v1/2023.emnlp-main.391.

[6] T. Ge, J. Hu, L. Wang, X. Wang, S. Q. Chen, and F. Wei, "In-context Autoencoder for Context Compression in a Large Language Model", Proc. ICLR 2024, Vienna, Austria, May 2024. https://doi.org/10.48550/arXiv.2307.06945.

[7] Z. Li, Y. Su, and N. Collier, "500xCompressor: Generalized Prompt Compression for Large Language Models", arXiv:2408.03094, Aug. 2024. https://doi.org/10.48550/arXiv.2408.03094.

[8] A. Nagle, A. Girish, M. Bondaschi, M. Gastpar, A. V. Makkuva, and H. Kim, "Fundamental Limits of Prompt Compression: A Rate-Distortion Framework for Black-Box Language Models", Proc. NeurIPS 2024, Vancouver, Canada, Dec. 2024. https://doi.org/10.48550/arXiv.2407.15504.

[9] F. Xu, W. Shi, and E. Choi, "RECOMP: Improving Retrieval-Augmented LMs with Compression and Selective Augmentation", Proc. ICLR 2024, Vienna, Austria, May 2024. https://doi.org/10.48550/arXiv.2310.04408.

[10] H. Jiang, Q. Wu, X. Luo, D. Li, C. Y. Lin, Y. Yang, and L. Qiu, "LongLLMLingua: Accelerating and Enhancing LLMs in Long Context Scenarios via Prompt Compression", Proc. ACL 2024, Bangkok, Thailand, pp. 1658-1677, Aug. 2024. https://doi.org/10.18653/v1/2024.acl-long.91.

[11] N. Chirkova, T. Formal, V. Nikoulina, and S. Clinchant, "Provence: Efficient and Robust Context Pruning for Retrieval-Augmented Generation", Proc. ICLR 2025, Singapore, Apr. 2025. https://doi.org/10.48550/arXiv.2501.16214.

[12] Z. Song, Y. Zhou, X. Kong, J. Jiao, X. Bao, X. You, X. Shi, Y. Zhou, and H. Qi, "Less is More for RAG: Information Gain Pruning for Generator-Aligned Reranking and Evidence Selection", arXiv:2601.17532, Jan. 2026. https://doi.org/10.48550/arXiv.2601.17532.

[13] S. Jha, L. E. Erdogan, S. Kim, K. Keutzer, and A. Gholami, "Characterizing Prompt Compression Methods for Long Context Inference", arXiv:2407.08892, Jul. 2024. https://doi.org/10.48550/arXiv.2407.08892.

[14] Anthropic, "Prompt caching", Claude Docs, https://docs.anthropic.com/en/docs/build-with-claude/prompt-caching. [accessed: Jun. 30, 2026]

[15] Anthropic, "Prompt caching with Claude", https://www.anthropic.com/news/prompt-caching. [accessed: Jun. 30, 2026]

[16] S. Xu, W. Xie, L. Zhao, and P. He, "Chain of Draft: Thinking Faster by Writing Less", arXiv:2502.18600, Feb. 2025. https://doi.org/10.48550/arXiv.2502.18600.

[17] T. Han, Z. Wang, C. Fang, S. Zhao, S. Ma, and Z. Chen, "Token-Budget-Aware LLM Reasoning", arXiv:2412.18547, Dec. 2024. https://doi.org/10.48550/arXiv.2412.18547.

[18] Z. Li, J. Zhong, Z. Zheng, X. Wen, Z. Xu, Y. Cheng, F. Zhang, and Q. Xu, "Making Slow Thinking Faster: Compressing LLM Chain-of-Thought via Step Entropy", arXiv:2508.03346, Aug. 2025. https://doi.org/10.48550/arXiv.2508.03346.

[19] X. Yong, X. Zhou, Y. Zhang, J. Li, Y. Zheng, and X. Wu, "Think or Not? Exploring Thinking Efficiency in Large Reasoning Models via an Information-Theoretic Lens", arXiv:2505.18237, May 2025. https://doi.org/10.48550/arXiv.2505.18237.

[20] J. Lu, H. Yu, S. Xu, S. Ran, G. Tang, S. Wang, B. Shan, T. Fu, H. Feng, J. Tang, H. Wang, and C. Huang, "Prolonged Reasoning Is Not All You Need: Certainty-Based Adaptive Routing for Efficient LLM/MLLM Reasoning", arXiv:2505.15154, May 2025. https://doi.org/10.48550/arXiv.2505.15154.

[21] D. Banerjee, T. Suresh, S. Ugare, S. Misailovic, and G. Singh, "CRANE: Reasoning with Constrained LLM Generation", arXiv:2502.09061, Feb. 2025. https://doi.org/10.48550/arXiv.2502.09061.

[22] I. Ong, A. Almahairi, V. Wu, W. L. Chiang, T. Wu, J. E. Gonzalez, M. W. Kadous, and I. Stoica, "RouteLLM: Learning to Route LLMs with Preference Data", Proc. ICLR 2025, Singapore, Apr. 2025. https://doi.org/10.48550/arXiv.2406.18665.

[23] LMSYS Org, "RouteLLM: An Open-Source Framework for Cost-Effective LLM Routing", https://lmsys.org/blog/2024-07-01-routellm/. [accessed: Jun. 30, 2026]

[24] L. Chen, M. Zaharia, and J. Zou, "FrugalGPT: How to Use Large Language Models While Reducing Cost and Improving Performance", arXiv:2305.05176, May 2023. https://doi.org/10.48550/arXiv.2305.05176.

[25] J. M. Lundin, A. Zhang, N. Karim, H. Louzan, V. Wei, D. Adelani, and C. Carroll, "The Token Tax: Systematic Bias in Multilingual Tokenization", arXiv:2509.05486, Sep. 2025. https://doi.org/10.48550/arXiv.2509.05486.

[26] W. Lajewska, M. Hardalov, L. Aina, N. A. John, H. Su, and L. Marquez, "Understanding and Improving Information Preservation in Prompt Compression for LLMs", arXiv:2503.19114, Mar. 2025. https://doi.org/10.48550/arXiv.2503.19114.

[27] Z. Zhang, J. Li, Y. Lan, X. Wang, and H. Wang, "An Empirical Study on Prompt Compression for Large Language Models", arXiv:2505.00019, May 2025. https://doi.org/10.48550/arXiv.2505.00019.

[28] W. Johnson, "Compression Method Matters: Benchmark-Dependent Output Dynamics in LLM Prompt Compression", arXiv:2603.23527, Mar. 2026. https://doi.org/10.48550/arXiv.2603.23527.

저자소개

김 경 민 (Gyeongmin Kim)



2017년 9월 ~ 2019년 8월 :
한국공학대학교
소프트웨어융합공학과(공학석사)
2025년 1월 대한민국산업현장교수
2007년 9월 ~ 현재 :
삼정데이타서비스(주) 차장
관심분야 : 대형언어모델 응용 시스템, 토큰 효율화, 프롬프트 압축, 소프트웨어 아키텍처