LLM의 만진법 수치 처리 향상을 위한 전처리 방법론

김혜련, 목경서, 이호진, 이래형*

{helena1129, mok0726, leefine16, *rh}@kodata.co.kr

Preprocessing Methodology for Enhancing LLM Numeric Processing in Ten-Thousand System

Kim Hyeryeon, Mok Kyeongseo, Lee Hojin, Lee Raehyeong

요 약

대형 언어 모델(Large Language Model, LLM)은 자연어 이해에 뛰어난 성능을 보이는 반면, 재무 데이터의 수치 처리에서는 지속적인 한계가 관찰되고 있다. 본 연구는 만진법(萬進法) 체계가 LLM의 수치 처리에 미치는 부정적 영향을 완화하기 위해, 수치를 4자리씩 나누어 만(萬), 억(億) 등의 단위를 명시하는 전처리 방법론 Chunk4를 제안한다. 재무 데이터 기반 두 가지 태스크—수치 읽기(Spell-out)와 수치 변환(Conversion)에서 GPT-4o, Gemini2.5 Flash, Qwen2.5-72B를 대상으로 Raw, Comma, Chunk4 전처리 조건을 비교한 결과, Chunk4가 9–13자리의 '큰 수' 구간에서 정확도를 유의하게 개선하였다. 이는 만진법 체계와 LLM 간의 구조적 불일치로 인한 수치 처리 한계를 완화하고, 재무 데이터 분석의 신뢰성을 높이는 데 기여한다.

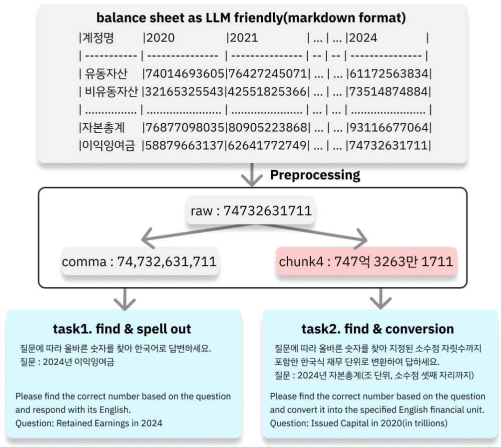
Ⅰ. 서 론

대형 언어 모델(Large Language Model, LLM)은 자연어 이해 및 생성 작업에서 뛰어난 성능을 보이지만, 수치 처리 영역에서는 지속적인 한계가 관찰되고 있다. 여기서 수치 처리란 숫자의 검색, 읽기(Spell-out), 변환 등의 작업을 포함한다[1]. 예를 들어 '123456'을 '십이만 삼천사백오십육'으로 읽거나, '856그램'을 '856000밀리그램'으로 변환하는 등의 작업을 지칭한다. 특히 재무 분야에서는 수치 처리 오류가 본래의 의미를 왜곡할 수 있어 정확성이 매우 중요하다.

LLM의 수치 처리 한계는 숫자의 크기가 커질수록 숫자 간 구분 능력이 떨어지는 경향성과 숫자의 토큰화 방향성에 기인한다. 첫째, LLM은 작은 숫자는 정밀하게 구분하지만, 큰 숫자로 갈수록 정밀한 구분 능력이 떨어진다[2]. 특히, 10^8-10^9구간에 정밀도가 떨어지는 현상이 관찰되었는데, 이는 재무 문서에서 자주 다루는 억(10^8)에서 조(10^{12}) 단위의 수치에서 오류가 발생할 가능성이 있음을 시사한다.

둘째, 숫자 토큰화의 방향이 수치 처리 성능에 큰 영향을 준다. GPT 계열의 기본 토크나이저는 긴 숫자를 왼쪽에서 오른쪽으로 세 자리씩 끊는 경향이 있어 자릿수에 대한 맥락을 잃어버릴 수 있다. 선행 연구[3]는 입력을 1,234,567처럼 표기해 오른쪽 기준으로 분절을 유도하면 숫자의 자릿수가 올바르게 정렬된 상태로 유지되어 산술 정확도가 크게 개선된다고 보고했다. 이는 모델을 재학습하지 않고도 입력 표현만 바꿔 성능을 높일 수 있음을 보여준다.

한편, 한자 문화권에서는 추가적인 도전 과제가 존재한다. 한자 문화권의 만진법(萬進法) 체계는 영어권의 천진법(千進法) 체계와 근본적으로 다른 명수법(命數法) 체계를 사용한다. 만진법은 수를 10,000 단위(조·억·만 등)로 구분하는 반면, 천진법은 1,000단위(billion · million · thousand 등)로 구분한다. 문제는 LLM이 주로 영어 기반 코퍼스로 학습되어, 만진법 체계에 대한 노출 빈도가 상대적으로 낮다는 점이다. 이에 따라 해당 체계와 관련된 숫자 표현이 충분히 학습되지 못해, 수치 해석에서 오류가 발생할 가능성이 높다. 실제로 중국어 연구에서는 이러한 성능 저하가 확인되었으나[1], 한국어 맥락에서 만진법 기반 수치 체계가 LLM에 미치는 영향을 실증적으로 검증한 사례는 여전히 부족하다.



[그림 1] 재무상태표를 Raw, Comma, Chunk4 형식으로 변환한 뒤, 수치 읽기와 수치 변환 태스크에 적용하는 과정

Ⅱ. 본론

Ⅱ.1. 제안 방법

본 연구에서 제안하는 Chunk4 방법론은 한국어의 만진법 수치 체계에 맞춰 설계된 전처리 방법론이다. 기존 연구[3]에서 효과가 입증된 3자리 콤마 구분법(예: '12,345,678')이 영어권의 천진법 체계를 반영한다면, Chunk4는 만진법의 4자리 단위 구분 방식에 최적화한 시도라는 점에서 의의가 있다. 구체적으로, 큰 숫자를 조, 억, 만 등의 단위로 분할하고 각 블록에 해당 단위명을 명시하는 방식이다. 예를 들어, '123456789'는 '1억2345만6789'로 변환된다.

Ⅱ.2. 실험 설계

본 연구는 Chunk4 방법론의 효과를 검증하기 위한 실험을 설계하였다. 연구에서 비교 대상으로 설정한 전처리 방법론은 (1) Raw: 원시 숫자 그대로 제시(예: 74732631711), (2) Comma: 선행 연구[3]에서 제시된 천 단위 쉼표 표기법(예: 74,732,631,711), (3) Chunk4: 숫자를 네 자리씩 분할하고 각 묶음에 해당하는 단위 명시(예: 747억3263만1711)의 세 가지이다.

전처리 방법론 간 차이를 검증하기 위해 [그림 1]과 같이 두 가지 태스크를 설계하여 실험을 진행하였다.

첫 번째, 수치 읽기(Spell-out) 태스크는 LLM이 재무상태표의 수치를 "칠백사십칠억 삼천이백육십삼만 천칠백십일"과 같이 자연어로만 이루어진 형태로 정확히 변환하는 능력을 평가한다.

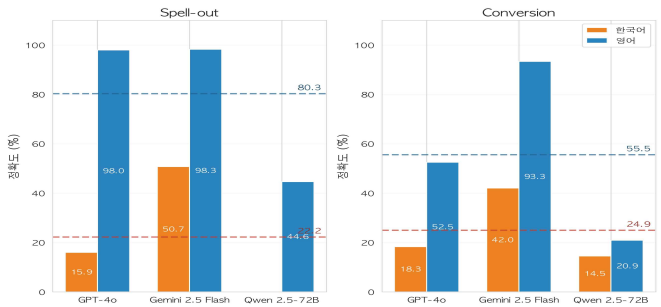
두 번째, 수치 변환(Conversion) 태스크는 LLM이 지정된 계정 항목의 수치를 정확히 식별하여 추출한 뒤, 주어진 수치 단위와 소수점 자릿수 규칙에 맞게 변환하는 능력을 평가한다. 예를 들어, "2024년 자본금을 억 단위로 소수점 둘째 자리까지 표시하시오."와 같은 요구에 LLM이 "931.16억"으로 정확히 추출하여 응답하는지를 평가한다. 이는 LLM이 만진법 체계를 이해하고, 금융권에서 통용되는 표현 방식 내에서 적절한 응답을 생성할 수 있는지를 종합적으로 평가한다.

데이터셋의 경우, 국내 상장사 115곳의 실제 재무상태표에서 추출한 숫자 데이터를 기반으로 한다. 이를 마크다운 형식으로 변환한 뒤, 프롬프트에 포함시켜 모델에 제공하였다. 수집된 재무 수치 데이터는 9-13자리 규모로, 10^8(억) 단위에서 10^{12}(조) 단위에 해당하는 범위에 분포하였다. 이 구간은 LLM이 수치를 정밀하게 구분하지 못하는 영역으로, 같은 간격 차이라도 작은 수치 구간 대비 구분 능력이 현저히 떨어지는 특성을 보인다.

모델의 경우, 상용 모델(GPT-4o, Gemini2.5 Flash)과 오픈소스 모델(Qwen2.5-72B)을 선정하였으며, 평가 기준은 금융권에서 통용되는 다양한 표현 방식을 모두 정답으로 인정하는 Pass Rate를 적용하여 실무 적용성을 제고하였다. 이를 통해 금융 문서 처리 과정에서 LLM이 직면하는 수치 인식의 한계를 실험적으로 검증할 수 있도록 구성하였다.

Ⅱ.3. 실험 결과 및 분석

먼저 한국어 환경과 영어 환경에서의 LLM의 수치 처리 성능 차이를 검증하기 위해 원시 숫자(Raw)에 대한 정확도를 분석하였다. 분석 결과, [그림 2]와 같이 성능에 유의한 차이가 있음을 확인하였다.



[그림 2] 언어별 정확도 비교

[그림 2]는 LLM의 수치 처리 정확도를 언어별로 비교한 결과를 보여준다. 수치 읽기 태스크에서는 영어 환경에서 평균 80.3%의 정확도를 기록한 반면, 한국어 환경에서는 22.2%에 그쳐 약 58%p의 성능 차이가 나타났다. 수치 변환 태스크에서도 언어 간 평균 약 30%p의 정확도 차이가 확인되었다. 이러한 경향은 모든 모델에서 일관되게 관찰되었으며, 이는 한국어 환경에서의 LLM의 수치 처리 성능이 상대적으로 낮아 한국어의 구조적 특성을 반영한 별도의 데이터 전처리 전략이 요구됨을 시사한다.

한국어 환경에서의 전처리 방법론 비교 실험에서는, 두 가지 태스크 모두에서 Chunk4 전처리 방법론을 적용했을 때의 평균 정확도가 높았다. [표 1]에서 보듯이, 수치 읽기 태스크에서는 전체 모델 평균 정확도가 Raw에서 22.22%, Comma 방법론에서 23.28%, Chunk4 방법론에서 45.70%로 나타나 Chunk4 방법론이 Raw 대비 23.48%p, Comma보다 22.42%p의 정확도 향상을 달성하였다. 수치 변환 태스크에서도 Chunk4 방법론이 평균 43.67%로 Raw 대비 18.74%p, Comma보다 13.82%p의 정확도 향상을 보여, 두 태스크 모두에서 일관된 성능 개선 효과를 확인하였다.

	Spell-out			Conversion		
	Raw	Comma	Chunk4	Raw	Comma	Chunk4
GPT-4o	15.94	12.75	50.15	18.26	30.43	57.68
Gemini2.5 Flash	50.72	57.10	77.68	42.03	48.41	45.51
Qwen2.5-72B	0.00	0.00	9.28	14.49	10.72	27.83
Average	**22.22**	**23.28**	**45.70**	**24.93**	**29.85**	**43.67**

(단위 : %)

[표 1] 한국어 환경에서의 전처리 방법론에 따른 태스크별 정확도 비교

모델별 세부 분석에서는 GPT-4o, Qwen2.5-72B 모델의 경우 두 태스크 모두에서 Chunk4 전처리 방법론이 가장 큰 개선 효과를 보였다. Comma 방법론은 Gemini2.5 Flash 수치 변환 태스크에는 가장 높은 개선 효과를 보였지만 Qwen2.5-72B에서 수치 읽기 태스크에서는 개선이 없었고, 수치 변환 태스크에서는 오히려 성능 하락(- 3.77%p)을 보였는데 이는 천진법 기반의 구분이 한국어 수 체계 인식에 혼동을 유발할 수 있음을 시사한다.

Ⅲ. 결론

본 연구는 Chunk4 전처리 방법론을 적용한 입력 데이터 전처리 방식이 한국어 환경에서 효과적임을 입증하였다. 재무 데이터 기반 두 가지 태스크를 통해 Chunk4 전처리의 효용성을 검증했으며, 모든 구간에서 기존 Raw·Comma 방식 대비 일관된 성능 향상을 보였다. 이는 본 방법론이 한국어 수치 처리 문제 해결에 효과적임을 의미한다. 본 연구 결과는 한국어뿐만 아니라 중국어, 일본어 등 만진법 문화권 전반에 적용 가능할 것으로 기대된다. 특히 재무 분야에서 LLM 적용의 신뢰성과 실무적 안정성을 높이는 데 기여할 수 있다.

Ⅳ. 연구의 한계점 및 향후 연구 방향

본 연구는 LLM이 만진법 체계에서 겪는 수치 처리 오류를 완화하고, 안정성을 향상하기 위해 Chunk4 전처리 방법론을 제안하였으나, 적용 LLM 모델과 데이터 도메인이 제한적이라는 한계가 있다. 또한 실제로 금융 현장에 적용함에 있어 운영 적용 가능성이나 비용 - 효익 분석은 검토되지 못하였다. 향후 연구에서는 다양한 LLM 모델을 대상으로 한 검증, 과학·공공분야 등 타 도메인 확장, 실무적 효용성 평가를 통해 방법론의 일반화 가능성과 산업적 활용성을 강화할 필요가 있다.

ACKNOWLEDGMENT

본 연구는 한국평가데이터 주식회사의 지원을 받아 수행되었습니다.

참 고 문 헌

[1] A. Xu, M. Tan, L. Wang, M. Yang, and R. Xu, "NUMCoT: Numerals and Units of Measurement in Chain-of-Thought Reasoning using Large Language Models," 2024.

[2] H. V. AlquBoj, H. AlQuabeh, V. Bojkovic, T. Hiraoka, A. O. El-Shangiti, M. Nwadike, and K. Inui, "Number Representations in LLMs: A Computational Parallel to Human Perception," 2025.

[3] A. K. Singh and D. J. Strouse, "Tokenization counts: the impact of tokenization on arithmetic in frontier LLMs," 2024.