

REST: RETRIEVAL-BASED SPECULATIVE DECODING

Đỗ Minh Khôi - 240101051

Tóm tắt

- Lớp: CS2205.FEB2025
- Link Github của nhóm:
<https://github.com/Jun0se7en/CS2205.FEB2025>
- Link YouTube video: <https://youtu.be/-DtLfs5Ypi8>



Đỗ Minh Khôi - 240101051

- Tổng số slides không vượt quá 10

Giới thiệu

- Các LLMs (GPT, Llama, Vicuna, ...) có khả năng suy luận rất mạnh, nhưng để chạy lại rất tốn tài nguyên.
- Phương pháp thông thường khi chạy LLMs:

Sinh từng token -> Push forward LLMs -> Chọn lọc token dựa trên xác suất -> Loop

- Phương pháp Speculative Decoding:

Dùng các LLMs nhỏ hơn nhiều lần dự đoán trước k token -> Verify sử dụng LLMs giống phương pháp trước -> Loop

Giới thiệu

- Phương pháp Speculative Decoding tuy đã tăng tốc được cho LLMs nhưng vẫn cần phải train lại các LLMs nhỏ và kiểm định chất lượng tạo sinh của các LLMs này
- REST đã ra đời phương pháp mới để tăng tốc LLMs vẫn sử dụng cơ chế Speculative Decoding nhưng không sử dụng các LLMs nhỏ làm draft model thay vào đó sẽ xây dựng một cơ sở dữ liệu (datastore) để xây dựng các draft token và verify lại các token này sử dụng LLMs

Mục tiêu

- Tăng tốc các LLMs (GPT, Llama, Vicuna, ...) giảm thiểu số lần push forward qua các LLMs này khi sinh văn bản
- Xây dựng được một datastore đủ lớn, đủ độ phủ để tăng độ chính xác cũng như tốc độ khi sinh văn bản của các LLMs
- Giữ nguyên chất lượng đầu ra - mọi token vẫn được LLM gốc xác nhận, không đánh đổi độ chính xác hay mạch lạc

Nội dung và Phương pháp

- REST gồm 3 bước chính, được thực hiện tuần tự tại mỗi bước sinh token:
 - Truy xuất (Retrieval)
 - Xây dựng cây Trie (Draft Construction)
 - Xác minh bằng Tree Attention (Verification)

Nội dung và Phương pháp

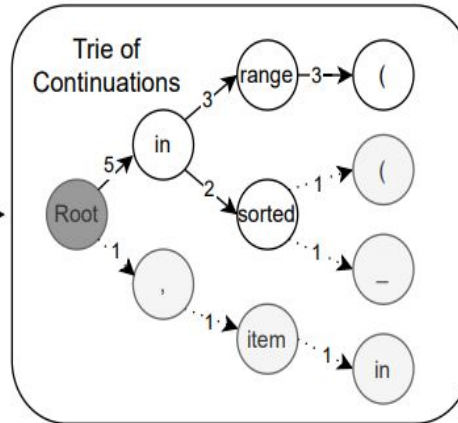
Step 1: Retrieve docs

| Retrieved Context | Continuations |
|----------------------------|---------------|
| numbers = [...]\n for i | in range(|
| dictionary = {...}\n for i | , item in |
| import math\n for i | in range(|
| numbers = [...]\n for i | in sorted(|
| file = open(...)\n for i | in range(|
| def sorted_c(...)\n for i | in sorted |



f = lambda num: [i for i

Step 2: Construct Trie



Step 3: Verify candidates

Tree Attention

| | | | | |
|--------|---|---|---|---|
| in | ✓ | | | |
| range | ✓ | ✓ | | |
| sorted | ✓ | | ✓ | |
| (| ✓ | ✓ | | ✓ |

Candidates

| | | | |
|----------|---|-----------|---|
| in | ✓ | in range(| ✓ |
| in range | ✓ | in sorted | ✗ |

**Retrieval-Based
Speculative Decoding (REST)**

Kết quả dự kiến

- Bộ dữ liệu đánh giá:
 - HumanEval (sinh mã Python)
 - MT-Bench (hội thoại nhiều lượt)
- Mô hình thử nghiệm:
 - CodeLlama 7B/13B
 - Vicuna 7B/13B
- Tăng tốc đạt được:
 - REST đạt tốc độ nhanh hơn so với phương pháp speculative decoding truyền thống
 - Datastore càng lớn chất lượng truy vấn và tốc độ càng cao

Kết quả dự kiến

| Benchmark | Model | Method | Mean Token Time(↓) | Speedup(↑) |
|---------------------|---------------|--------------------------|--------------------|------------|
| HumanEval (1 shot) | CodeLlama 7B | Autoregressive (Greedy) | 27.89 ms/token | 1× |
| | CodeLlama 7B | Speculative (Greedy) | 15.90 ms/token | 1.75× |
| | CodeLlama 7B | REST (Greedy) | 11.82 ms/token | 2.36× |
| | CodeLlama 13B | Autoregressive (Greedy) | 44.32 ms/token | 1× |
| | CodeLlama 13B | Speculative (Greedy) | 19.39 ms/token | 2.29× |
| | CodeLlama 13B | REST (Greedy) | 19.53 ms/token | 2.27× |
| HumanEval (10 shot) | CodeLlama 7B | Autoregressive (Nucleus) | 27.99 ms/token | 1× |
| | CodeLlama 7B | Speculative (Nucleus) | 18.83 ms/token | 1.49× |
| | CodeLlama 7B | REST (Nucleus) | 13.18 ms/token | 2.12× |
| | CodeLlama 13B | Autoregressive (Nucleus) | 44.46 ms/token | 1× |
| | CodeLlama 13B | Speculative (Nucleus) | 22.68 ms/token | 1.96× |
| | CodeLlama 13B | REST (Nucleus) | 20.47 ms/token | 2.17× |
| MT-Bench | Vicuna 7B | Autoregressive (Greedy) | 25.48 ms/token | 1× |
| | Vicuna 7B | Speculative (Greedy) | 19.44 ms/token | 1.31× |
| | Vicuna 7B | REST (Greedy) | 15.12 ms/token | 1.69× |
| | Vicuna 13B | Autoregressive (Greedy) | 44.30 ms/token | 1× |
| | Vicuna 13B | Speculative (Greedy) | 29.80 ms/token | 1.49× |
| | Vicuna 13B | REST (Greedy) | 25.08 ms/token | 1.77× |
| MT-Bench | Vicuna 7B | Autoregressive (Nucleus) | 25.93 ms/token | 1× |
| | Vicuna 7B | Speculative(Nucleus) | 20.65 ms/token | 1.26× |
| | Vicuna 7B | REST(Nucleus) | 16.02 ms/token | 1.62× |
| | Vicuna 13B | Autoregressive (Nucleus) | 44.32 ms/token | 1× |
| | Vicuna 13B | Speculative (Nucleus) | 31.78 ms/token | 1.39× |
| | Vicuna 13B | REST (Nucleus) | 25.92 ms/token | 1.71× |

Tài liệu tham khảo

- [1]. Zhenyu He, Zexuan Zhong, Tianle Cai, Jason D. Lee, Di He: REST: Retrieval-Based Speculative Decoding. NAACL 2024: 1582–1595.
- [2]. Chiang, Wei-Lin et al. (Mar. 2023). Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.
- [3]. Kocetkov, Denis et al. (2022). The Stack: 3 TB of permissively licensed source code. arXiv: 2211.15533 [cs.CL].
- [4]. Touvron, Hugo et al. (2023). Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv: 2307.09288 [cs.CL].
- [5]. Wang, Hanrui, Zhekai Zhang, and Song Han: “SpAtten: Efficient Sparse Attention Architecture with Cascade Token and Head Pruning”. 2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA) 2021:
- [6]. Manber, Udi and Gene Myers (1993). “Suffix Arrays: A New Method for On-Line String Searches”. In: SIAM Journal on Computing 22.5, pp. 935–948. eprint: <https://doi.org/10.1137/0222058>.
- [7]. Spector, Benjamin and Chris Re (2023). Accelerating LLM Inference with Staged Speculative Decoding. arXiv: 2308.04623 [cs.AI].