Speculative Decoding

Fast Inference from Transformers via Speculative Decoding
Leviathan et al. 2023

summarized by Michael Scherbela

Deep Learning Seminar
September 13, 2023

universität
wien

Highest voted question on Stackoverflow of all time

Why is processing a sorted array faster than processing an unsorted array?

```
// Generate data
const unsigned arraySize = 32768;
int data[arraySize];

for (unsigned c = 0; c < arraySize; ++c)
    data[c] = std::rand() % 256;

// !!! With this, the next loop runs faster.
std::sort(data, data + arraySize);

// Test
clock_t start = clock();
long long sum = 0;
for (unsigned i = 0; i < 100000; ++i)
{
    for (unsigned c = 0; c < arraySize; ++c)
    {   // Primary loop.
        if (data[c] >= 128)
            sum += data[c];
    }
}
```

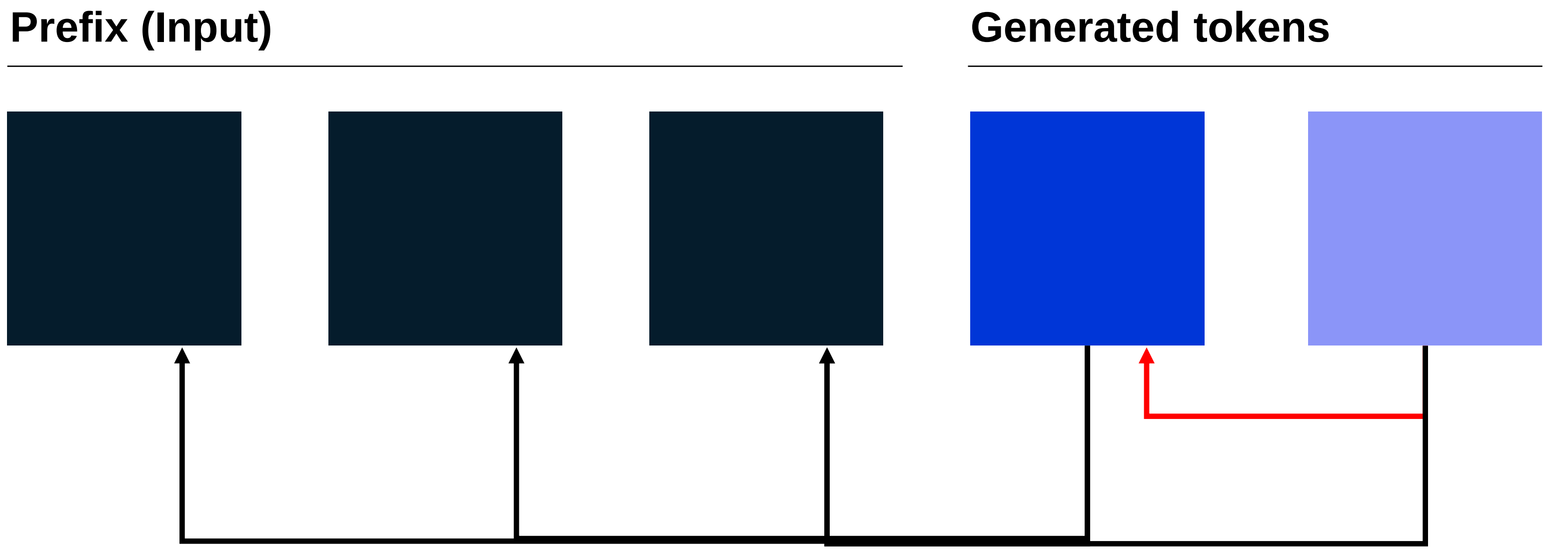
https://stackoverflow.com/questions/11227809/why-is-processing-a-sorted-array-faster-than-processing-an-unsorted-array

Branch prediction allows parallelization of (potentially) serial tasks

Time ↓	Conventional				Pipelined				Pipelined with branch prediction			
	Fetch	Decode	Process	#	Fetch	Decode	Process	#	Fetch	Decode	Process	#
1	A			0	A	*	*	0	A	*	*	0
2		A		0	B	A	*	0	B	A	*	0
3			A	1	C	B	A	1	C	B	A	1
4	B			1		C	B	2	E	C	B	2
5		B		1			C	3	F	E	C	3
6			B	2	E	GAP		3	G	F	E	4
7	C			2	F	E		3	H	G	F	5
8		C		2	G	F	E	4	I	H	G	6
9			C	3	H	G	F	5	J	I	H	7
10	E			3	I	H	G	6	K	J	I	8
11		E		3	J	I	H	7	L	K	J	9
12			E	4	K	J	I	8	M	L	K	10

http://www.edboas.com/science/branch_prediction/index.html

Same problem with LLMs:
Each token depends on all previous tokens



Proposed algorithm

Algorithm

1. Serially generate γ tokens using cheap model Q and keep probabilities $q(x_i)$
2. In parallel compute γ probabilities $p(x_i)$ using tokens from Q as prefix
3. For each generated token
   1. Keep it with probability $\frac{p}{q}$
   2. If rejected, draw a new token and throw away all remaining completions

Example

[START] japan ' s benchmark ~~bond~~ n

[START] japan ' s benchmark nikkei 22 ,5

[START] japan ' s benchmark nikkei 225 index rose 22 ,6

[START] japan ' s benchmark nikkei 225 index rose 226 . 69 , points

[START] japan ' s benchmark nikkei 225 index rose 226 . 69 points , or 0 1

[START] japan ' s benchmark nikkei 225 index rose 226 . 69 points , or 1 . 5 percent , to 10 , 9859

Key metrics of model Q

Accuracy of Q:

α… mean acceptance probability

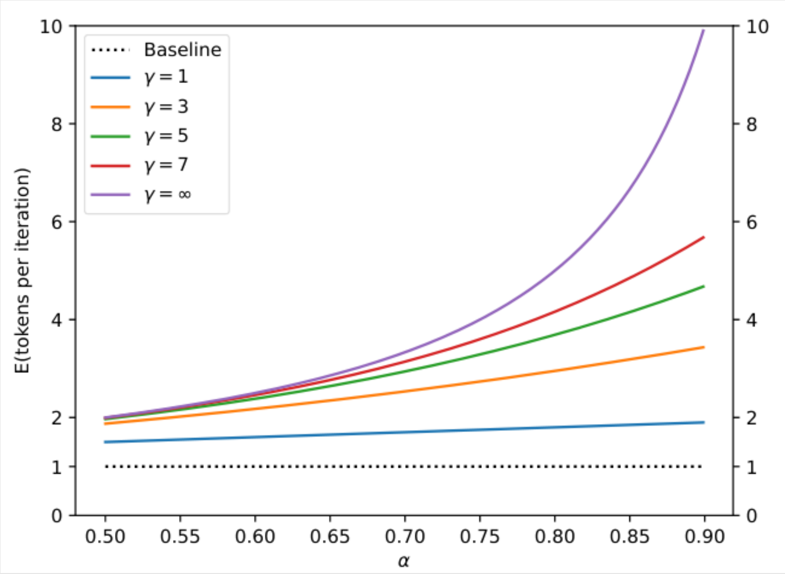
Cost of Q:

c … cost ratio of model Q vs. model P

Expected number of generated tokens

$$E[n] = 1 + \alpha + \alpha^2 + \ldots + \alpha^\gamma$$

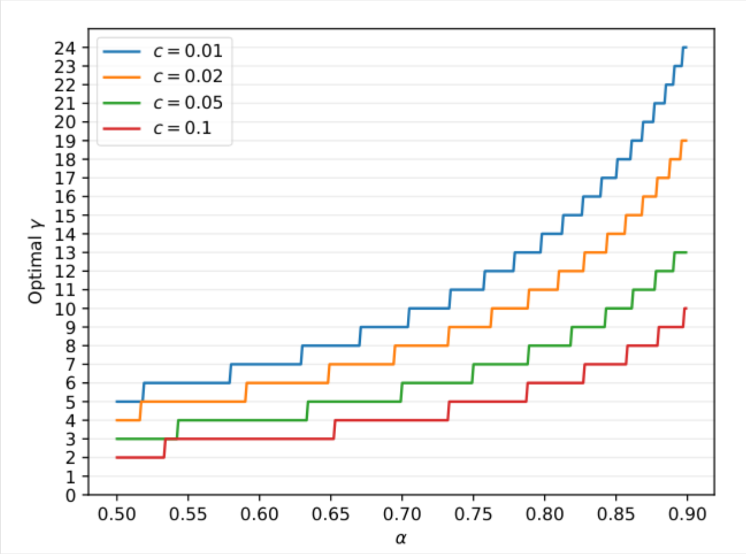
$$= \sum_{k=0}^{\gamma} \alpha^k = \frac{1 + \alpha^{\gamma+1}}{1 + \alpha}$$



Run-time per step

$$T = T_P(1 + c\gamma)$$

Optimal choice of γ



Typical acceptance rates are 50-80%

Table 3. Empirical α values for various target models M_p, approximation models M_q, and sampling settings. T=0 and T=1 denote argmax and standard sampling respectively[6].

M_p	M_q	SMPL	α
GPT-LIKE (97M)	UNIGRAM	T=0	0.03
GPT-LIKE (97M)	BIGRAM	T=0	0.05
GPT-LIKE (97M)	GPT-LIKE (6M)	T=0	0.88
GPT-LIKE (97M)	UNIGRAM	T=1	0.03
GPT-LIKE (97M)	BIGRAM	T=1	0.05
GPT-LIKE (97M)	GPT-LIKE (6M)	T=1	0.89
T5-XXL (ENDE)	UNIGRAM	T=0	0.08
T5-XXL (ENDE)	BIGRAM	T=0	0.20
T5-XXL (ENDE)	T5-SMALL	T=0	0.75
T5-XXL (ENDE)	T5-BASE	T=0	0.80
T5-XXL (ENDE)	T5-LARGE	T=0	0.82
T5-XXL (ENDE)	UNIGRAM	T=1	0.07
T5-XXL (ENDE)	BIGRAM	T=1	0.19
T5-XXL (ENDE)	T5-SMALL	T=1	0.62
T5-XXL (ENDE)	T5-BASE	T=1	0.68
T5-XXL (ENDE)	T5-LARGE	T=1	0.71

M_p	M_q	SMPL	α
T5-XXL (CNNDM)	UNIGRAM	T=0	0.13
T5-XXL (CNNDM)	BIGRAM	T=0	0.23
T5-XXL (CNNDM)	T5-SMALL	T=0	0.65
T5-XXL (CNNDM)	T5-BASE	T=0	0.73
T5-XXL (CNNDM)	T5-LARGE	T=0	0.74
T5-XXL (CNNDM)	UNIGRAM	T=1	0.08
T5-XXL (CNNDM)	BIGRAM	T=1	0.16
T5-XXL (CNNDM)	T5-SMALL	T=1	0.53
T5-XXL (CNNDM)	T5-BASE	T=1	0.55
T5-XXL (CNNDM)	T5-LARGE	T=1	0.56
LAMDA (137B)	LAMDA (100M)	T=0	0.61
LAMDA (137B)	LAMDA (2B)	T=0	0.71
LAMDA (137B)	LAMDA (8B)	T=0	0.75
LAMDA (137B)	LAMDA (100M)	T=1	0.57
LAMDA (137B)	LAMDA (2B)	T=1	0.71
LAMDA (137B)	LAMDA (8B)	T=1	0.74

Inference from a 11B model can be sped up 2-3x

Table 2. Empirical results for speeding up inference from a T5-XXL 11B model.

TASK	M_q	TEMP	γ	α	SPEED
ENDE	T5-SMALL ★	0	7	0.75	**3.4X**
ENDE	T5-BASE	0	7	0.8	2.8X
ENDE	T5-LARGE	0	7	0.82	1.7X
ENDE	T5-SMALL ★	1	7	0.62	**2.6X**
ENDE	T5-BASE	1	5	0.68	2.4X
ENDE	T5-LARGE	1	3	0.71	1.4X
CNNDM	T5-SMALL ★	0	5	0.65	**3.1X**
CNNDM	T5-BASE	0	5	0.73	3.0X
CNNDM	T5-LARGE	0	3	0.74	2.2X
CNNDM	T5-SMALL ★	1	5	0.53	**2.3X**
CNNDM	T5-BASE	1	3	0.55	2.2X
CNNDM	T5-LARGE	1	3	0.56	1.7X

Model sizes Q:

- T5-small: 77 mio
- T5-base: 250 mio
- T5-large: 800 mio

But what about computational cost for the parallel M_p evaluations?

1. You might be willing to **trade off** total cost vs. latency
2. Inference is typically **memory bound** => Batch-size 1 and batch-size γ have almost identical cost

References

- **Fast Inference from Transformers via Speculative Decoding**
  Leviathan et al., 2023
  http://arxiv.org/abs/2211.17192