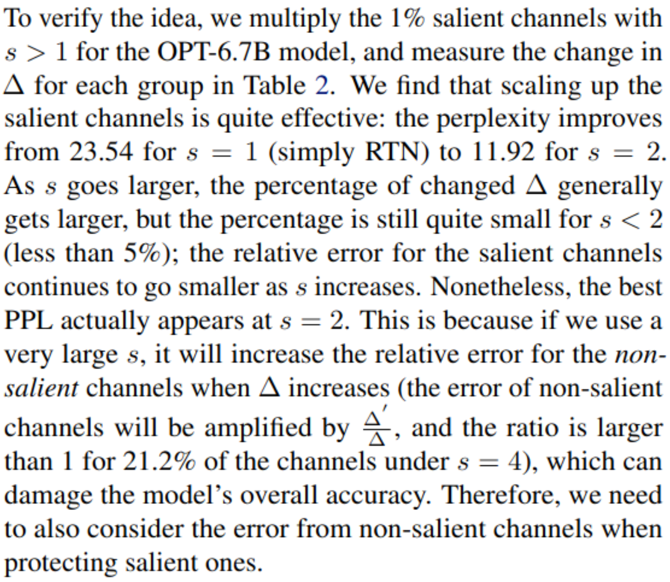
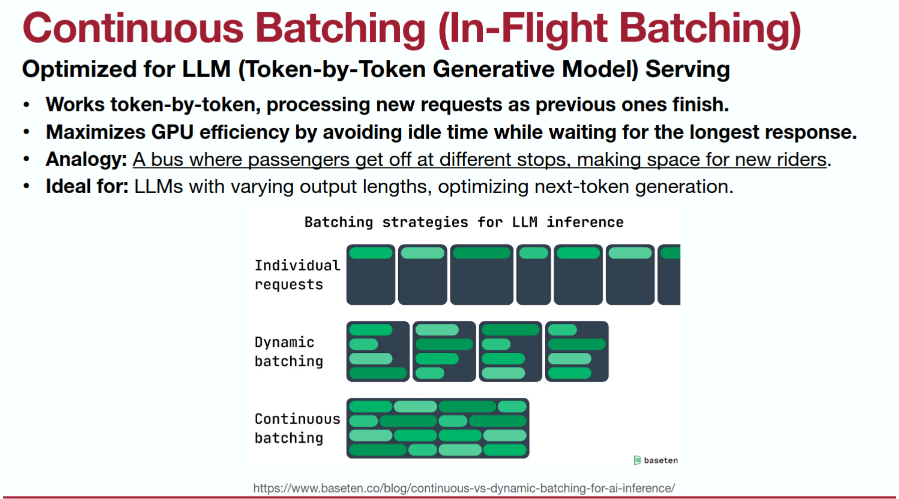
## Why does perplexity go down and then increase?



To verify the idea, we multiply the 1% salient channels with $s > 1$ for the OPT-6.7B model, and measure the change in $\Delta$ for each group in Table 2. We find that scaling up the salient channels is quite effective: the perplexity improves from 23.54 for $s = 1$ (simply RTN) to 11.92 for $s = 2$. As $s$ goes larger, the percentage of changed $\Delta$ generally gets larger, but the percentage is still quite small for $s < 2$ (less than 5%); the relative error for the salient channels continues to go smaller as $s$ increases. Nonetheless, the best PPL actually appears at $s = 2$. This is because if we use a very large $s$, it will increase the relative error for the *non-salient* channels when $\Delta$ increases (the error of non-salient channels will be amplified by $\frac{\Delta'}{\Delta}$, and the ratio is larger than 1 for 21.2% of the channels under $s = 4$), which can damage the model's overall accuracy. Therefore, we need to also consider the error from non-salient channels when protecting salient ones.

## Batching:

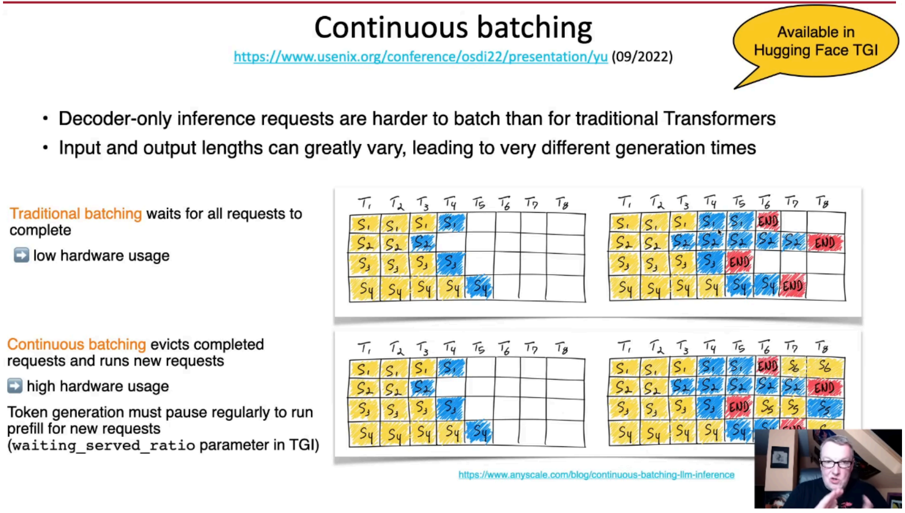


# Continuous Batching (In-Flight Batching)

**Optimized for LLM (Token-by-Token Generative Model) Serving**

- **Works token-by-token, processing new requests as previous ones finish.**
- **Maximizes GPU efficiency by avoiding idle time while waiting for the longest response.**
- **Analogy:** A bus where passengers get off at different stops, making space for new riders.
- **Ideal for:** LLMs with varying output lengths, optimizing next-token generation.

https://www.baseten.co/blog/continuous-vs-dynamic-batching-for-ai-inference/



# Continuous batching

https://www.usenix.org/conference/osdi22/presentation/yu (09/2022)

- Decoder-only inference requests are harder to batch than for traditional Transformers
- Input and output lengths can greatly vary, leading to very different generation times

Traditional batching waits for all requests to complete

➡️ low hardware usage

| $T_1$ | $T_2$ | $T_3$ | $T_4$ | $T_5$ | $T_6$ | $T_7$ | $T_8$ |
|---|---|---|---|---|---|---|---|
| $S_1$ | $S_1$ | $S_1$ | $S_1$ |  |  |  |  |
| $S_2$ | $S_2$ | $S_2$ |  |  |  |  |  |
| $S_3$ | $S_3$ | $S_3$ | $S_3$ |  |  |  |  |
| $S_4$ | $S_4$ | $S_4$ | $S_4$ | $S_4$ |  |  |  |

| $T_1$ | $T_2$ | $T_3$ | $T_4$ | $T_5$ | $T_6$ | $T_7$ | $T_8$ |
|---|---|---|---|---|---|---|---|
| $S_1$ | $S_1$ | $S_1$ | $S_1$ | $S_1$ | END |  |  |
| $S_2$ | $S_2$ | $S_2$ | $S_2$ | $S_2$ | $S_2$ | $S_2$ | END |
| $S_3$ | $S_3$ | $S_3$ | $S_3$ | END |  |  |  |
| $S_4$ | $S_4$ | $S_4$ | $S_4$ | $S_4$ | $S_4$ | END |  |

Continuous batching evicts completed requests and runs new requests

➡️ high hardware usage

Token generation must pause regularly to run prefill for new requests (`waiting_served_ratio` parameter in TGI)

| $T_1$ | $T_2$ | $T_3$ | $T_4$ | $T_5$ | $T_6$ | $T_7$ | $T_8$ |
|---|---|---|---|---|---|---|---|
| $S_1$ | $S_1$ | $S_1$ | $S_1$ |  |  |  |  |
| $S_2$ | $S_2$ | $S_2$ |  |  |  |  |  |
| $S_3$ | $S_3$ | $S_3$ | $S_3$ |  |  |  |  |
| $S_4$ | $S_4$ | $S_4$ | $S_4$ | $S_4$ |  |  |  |

| $T_1$ | $T_2$ | $T_3$ | $T_4$ | $T_5$ | $T_6$ | $T_7$ | $T_8$ |
|---|---|---|---|---|---|---|---|
| $S_1$ | $S_1$ | $S_1$ | $S_1$ | $S_1$ | END | $S_6$ | $S_6$ |
| $S_2$ | $S_2$ | $S_2$ | $S_2$ | $S_2$ | $S_2$ | $S_2$ | END |
| $S_3$ | $S_3$ | $S_3$ | $S_3$ | END | $S_5$ | $S_5$ | $S_5$ |
| $S_4$ | $S_4$ | $S_4$ | $S_4$ | $S_4$ | [illegible] | [illegible] | [illegible] |

https://www.anyscale.com/blog/continuous-batching-llm-inference

## Why perplexity and not accuracy?

The choice of these particular zero-shot tasks was mainly motivated by previous work (Dettmers et al., 2022a; Yao et al., 2022; Xiao et al., 2022). However, in our evaluation, we find that perplexity is a superior metric since its continuous value per sample leads to less noisy evaluations. This has also been noted by Frantar et al. (2022). For example, when using perplexity to evaluate data types, quantile quantization is the best data type. Still, when we use zero-shot accuracy as an evaluation metric, the float data type is sometimes better because zero-shot accuracy is noisier. Furthermore, we find that across more than 35,000 zero-shot experiments, the **Pearson correlation coefficient between The Pile Common Crawl perplexity and zero-shot performance is -0.94**.