ANLP PROJECT

Team -28 Strawberrries

FAST INFERENCE FROM TRANSFORMERS VIA SPECULATIVE DECODING

NEED FOR FAST INFERENCE

- **Scalability**: Accelerating inference enables efficient handling of multiple concurrent requests, ensuring consistent performance under high-demand scenarios like e-commerce and customer service.
- **User Engagement**: Fast inference is crucial for chatbots and virtual assistants, minimizing delays to enhance user satisfaction and retention.
- **Business Impact**: Optimized inference supports seamless operations in real-time applications, fostering better customer experiences and preventing service abandonment.

PROJECT OVERVIEW

- Autoregressive sequence-to-sequence models face a critical challenge: slow inference due to **sequential decoding**, where generating K tokens requires K serial runs. This limits their efficiency in real-time applications.
- To address this, the project introduces **speculative decoding**, an algorithm that accelerates inference by computing multiple tokens in parallel while maintaining output quality. Unlike traditional methods, this approach requires no retraining or architectural changes.
- Empirical experiments demonstrate **1.5X-3X speedups** using speculative decoding on T5-Large, providing identical outputs while significantly enhancing the suitability of transformer models for time-sensitive tasks.

INTUITION FOR ALGORITHM

- Hard language-modeling tasks often include easier subtasks that can be approximated well by more efficient models.
- How to choose smaller models:
  - The ratio of inference time of smaller model to larger model should be very small.
  - The task should be approximated well by small model.
  - Smaller model should have same vocabulary as the larger model.

SPECULATIVE DECODING

Let **Mp** be the target model **Mq** be a more efficient approximation model for the same task. The proposed speculative decoding method operates by:

- Utilizing a smaller approximation model to generate **γ** speculative token completions.
- Mp is then run **γ + 1** times in parallel to evaluate these guesses and their respective probabilities from Mq in parallel, accepting all those that can lead to an identical distribution.
- Sampling an additional token from an adjusted distribution to fix the first one that was rejected, or to add an additional one if they are all accepted

Algorithm 1 Speculative Decoding Step

Inputs: M_p, M_q, *prefix*
Sample γ guesses $x_1, \ldots, x_\gamma$ from M_q autoregressively.
for $i = 1$ to γ **do**
  $q_i(x) \leftarrow M_q(\text{prefix} + [x_1, \ldots, x_{i-1}])$
  $x_i \sim q_i(x)$
end for
Run M_p in parallel.
$p(x_1), \ldots, p(x_{\gamma+1}) \leftarrow M_p(\text{prefix}), \ldots, M_p(\text{prefix} + [x_1, \ldots, x_\gamma])$
Determine the number of accepted guesses n.
$r_1 \sim U(0,1), \ldots, r_\gamma \sim U(0,1)$
$n \leftarrow \min(\{i - 1 \mid 1 \leq i \leq \gamma, r_i > \frac{p_i(x)}{q_i(x)}\} \cup \{\gamma\})$
Adjust the distribution from M_p **if needed**.
$p'(x) \leftarrow p_{n+1}(x)$
if $n < \gamma$ **then**
  $q'(x) \leftarrow \text{norm}(\max(0, p_{n+1}(x) - q_{n+1}(x)))$
end if
Return one token from M_p, **and** n **tokens from** M_q.
$t \sim p'(x)$
return $\text{prefix} + [x_1, \ldots, x_n, t]$

EXPERIMENTS

1. We test a standard encoder-decoder T5 model on two tasks:
   - English to German translation (EnDe)
   - Text summarization (CNNDM)
2. For both the tasks we use **T5-Large** for Mp. For the approximation model Mq we use **T5-small** as the cost factor, **c** is around 0.11
3. The base model performance is as follows:

Table 1. Empirical results for base models

Task	Model	Ag. Inference Time	Avg. Bleu Score
EnDe	T5-Large	6.39 sec	0.717
Ende	T5-Small	0.75 sec	0.554
CNNDM	T5-Large	216.62 sec	0.38
CNNDM	T5-Small	12.02 sec	0.35

RESULTS AND ANALYSIS

- The acceptance rate $\beta_{x<t}$,given a prefix x<t, is the probability of accepting $x_t \sim q(x_t|x < t)$
- We denote **α = E(β)** where E(β) is a natural measure of how well Mq approximates Mp.
- We calculate α by calculating the running average of the acceptance rate **β.**
- The cost factor **c** is defined as the inference time (IT) ratio: $c = \frac{\mathrm{IT}(M_q)}{\mathrm{IT}(M_p)}$
- The expected walltime improvement factor is given by:

$$\frac{1-\alpha^{\gamma+1}}{(1-\alpha)*(\gamma c+1)}$$

RESULTS AND ANALYSIS

- Results for running Speculative Decoding for different tasks and combination of :

Table 2. Empirical results for speeding up inference from a T5-Large model

Task	M_q	Avg. Bleu Score	γ	α	Speed
EnDe	T5-Small	0.73	3	0.79	2.68
EnDe	T5-Small	0.68	7	0.55	2.56
CNNDM	T5-Small	0.37	3	0.48	2.11
CNNDM	T5-Small	0.36	7	0.33	2.17

- **Performance Boost**: Speculative decoding achieves up to 2.68× faster inference, maintaining comparable BLEU scores (e.g., 0.73 for EnDe and 0.37 for CNNDM) by leveraging smaller approximation models like Mq

NOVELTY

We have implemented following three future improvements for the proposed algorithm given in paper:

1. Speculative Decoding on beam search
2. Speculative Decoding with adaptive gamma
3. Hierarchical Speculative Decoding

SPECULATIVE DECODING ON BEAM SEARCH

1. To incorporate beam search, we use the beam width k and max length as parameters.
2. At each generation step, we maintain the top k beams, scoring them based on their running log probabilities.
3. From all candidates generated at a step, the top k beams are selected. The process stops when an EOS token is generated or the max length is reached.

RESULTS AND ANALYSIS

- Results for running Speculative Decoding for different tasks and combination of :

Task	M_q	Bleu Score	No. of beams	α	Speed
EnDe	T5-Small	0.73	3	0.79	0.79
EnDe	T5-Small	0.72	5	0.75	0.45

- In greedy decoding with k beams, inference time increases by about k folds since each beam is processed sequentially. In contrast, speculative decoding minimizes this increase by leveraging parallelism and using a smaller approximation model Mq.
- This approach reduces inference time while improving output quality. Empirical results show minimal speed reduction, with inference times of 0.79× for beam width 3 and 0.45× for beam width 5, demonstrating significant efficiency gains.

SPECULATIVE DECODING WITH ADAPTIVE GAMMA

- Traditionally, γ is fixed, but the proposed method predicts β (the expected acceptance rate) using a running average of the acceptance rate during execution.
- This dynamic prediction allows for the adjustment of γ throughout the algorithm's run.
- By applying gradient ascent to the wall-time improvement equation, the algorithm dynamically maximises the efficiency of inference, ensuring that the optimal γ is selected in real-time, leading to improved overall performance and faster inference.

RESULTS AND ANALYSIS

- Results for running Speculative Decoding using dynamic gamma are as follows:

Task	M_q	Avg. Bleu Score	α	Speed
EnDe	T5-Small	0.71	0.68	2.61
CNNDM	T5-Small	0.35	0.47	2.33

- Dynamically adjusting γ in speculative decoding optimises the balance between speed and output quality by adapting to real-time model performance.
- Using a running average to predict the acceptance rate β and applying gradient ascent on the wall-time improvement equation (given before) , the algorithm achieves higher speedups compared to static γ approaches.
- This dynamic adjustment ensures both improved efficiency and quality retention without manual tuning, making it more effective for diverse tasks and applications.

HIERARCHICAL SPECULATIVE DECODING

- We tried to explore a hierarchical version of the algorithm, where the approximation model is itself accelerated by an even faster model, which could allow for more capable approximation models.
- This approach employs speculative decoding across three models. First, γ tokens are generated sequentially using the small model. The medium model then processes the resulting γ + 1 prefixes in parallel, applying an acceptance-rejection mechanism as given in paper.
- If n tokens are accepted, the medium model sequentially generates the remaining γ − n tokens.
- These prefixes are then evaluated in parallel by the large model and the tokens are accepted according to acceptance-rejection mechanism as given in paper

RESULTS AND ANALYSIS

- Results for running hierarchal speculative decoding are as follows:

Task	Combination	Avg. Bleu Score	Speed
EnDe	T5-Large + T5-Small	0.723	2.68
EnDe	T5-Large + T5-Base	0.730	1.13
EnDe	Hierarchy	0.728	1.46

- The hierarchical speculative decoding approach balances quality and speed by using three models. It achieves a BLEU score of 0.728, slightly lower than the large-base pair (0.730), but higher than the large-small pair (0.723).
- The model offers a 1.46× speedup, outperforming the large-base pair (1.13×). This trade-off results from the intermediate T5-Base model, which enables more informed approximations, balancing high-quality outputs and moderate speed improvements for efficient inference.

ADVANTAGES OF THIS ALGORITHM

- Through this algorithm we have achieved a low latency while maintaining identical outputs.
- We don’t require to
  - change the model architecture
  - change the training procedure
  - retrain the models.

DEMO

TEAM MEMBERS

NIPUN TULSIAN (2021101055)
VYOM GOYAL (2021101099)
RHYTHM AGGARWAL (2021101081)

PROF. MANISH SHRIVASTAVA

Thank You!