Self-RAG: AI That Knows When to Double Check

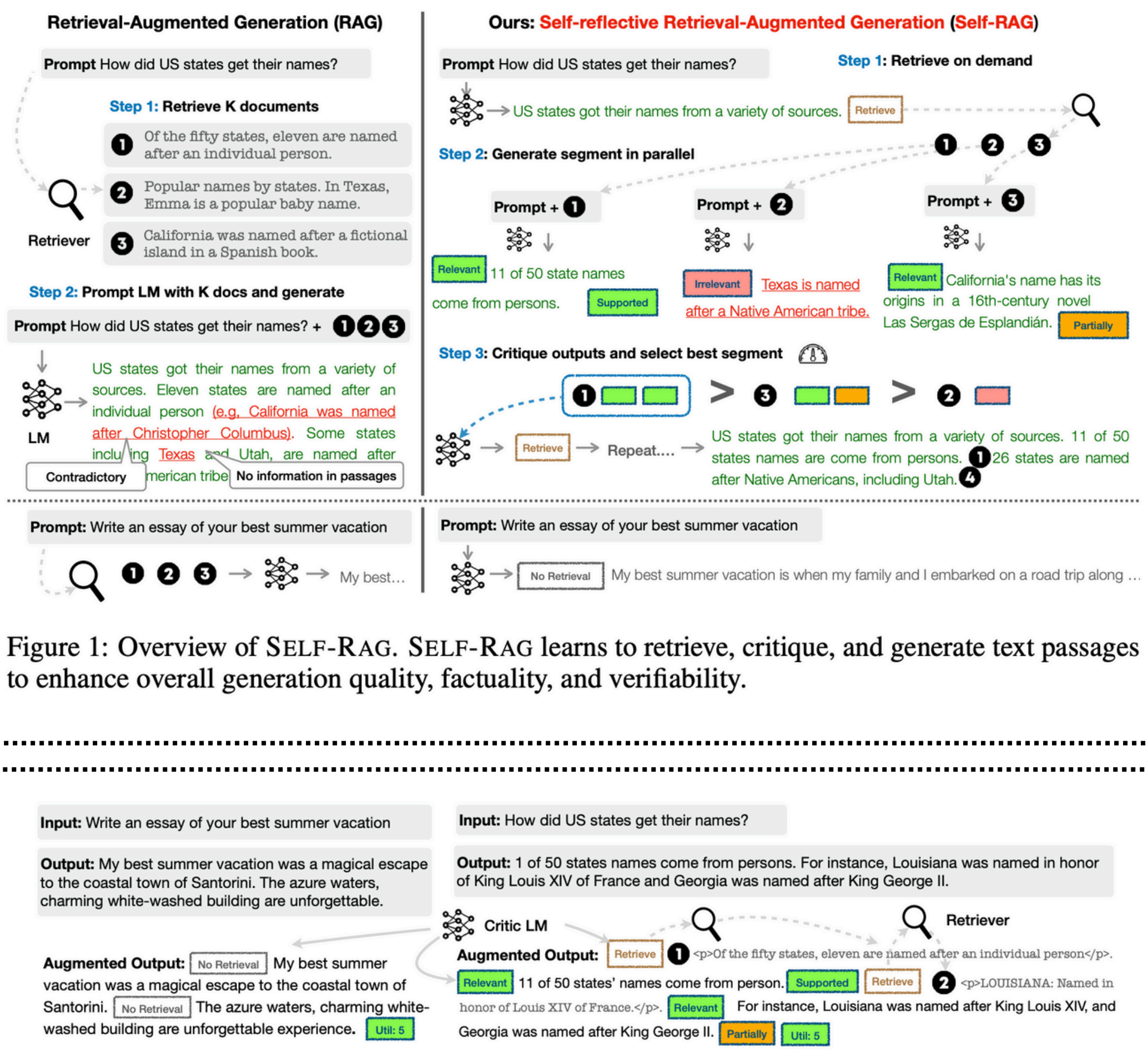

Figure 1: Overview of SELF-RAG. SELF-RAG learns to retrieve, critique, and generate text passages to enhance overall generation quality, factuality, and verifiability.

Figure 2: SELF-RAG training examples. The left example does not require retrieval while the right one requires retrieval; thus, passages are inserted.

Problem with Standard RAG

While RAG mitigates factual inaccuracies in LLMs using external knowledge, but has limitations. Standard RAG approaches suffer from several key problems:

- **Indiscriminate Retrieval**: RAG retrieves a fixed number of documents, regardless of relevance or need. This wastes resources and can introduce irrelevant information which causes lower-quality outputs.
- **Lack of Adaptability**: Standard RAG methods don't adjust to different task requirements. They lack the control to determine when and how much to retrieve, unlike Self-RAG which can adapt retrieval frequency.
- **Inconsistency with Retrieved Passages**: The generated output often fails to align with the retrieved information because the models lack explicit training to use it.
- **No Self-Evaluation or Critique**: RAG doesn't evaluate the quality or relevance of retrieved passages, nor does it critique its output. It blindly incorporates passages, unlike Self-RAG which does a self-assessment.
- **Limited Attribution**: Standard RAG doesn't offer detailed citations or indicate if the generated text is supported by the sources. Self-RAG, in contrast, provides detailed citations and assessments.

Self-RAG

Self-reflective retrieval-augmented Generation (Self-RAG) improves the quality and factuality of LLMs by incorporating retrieval and self-reflection mechanisms. Unlike traditional RAG methods, Self-RAG trains an arbitrary LM to adaptively retrieve passages on demand. It generates text informed by these passages and critiques its output using special reflection tokens.

Here are the key components and characteristics of Self-RAG:

- **On-Demand Retrieval**: It retrieves passages on-demand using a “retrieve token,” only when needed, which makes it more efficient than standard RAG.

- **Use Reflection Tokens**: It uses special reflection tokens (both retrieval and critique tokens) to assess its generation process. Retrieval tokens signal the need for retrieval. Critique tokens evaluate the relevance of retrieved passages (ISREL), the support provided by passages to the output (ISSUP), and the overall utility of the response (ISUSE).

- **Self-Critique and Evaluation**: Self-RAG critiques its own output, assessing the relevance and support of retrieved passages, and the overall quality of the generated response.

- **Train End-to-End**: The model generates both the output and reflection tokens by using a critic model offline to create reflection tokens, which it then incorporates into the training data. This eliminates the need for a critic during inference.

- **Enable Customizable Decoding**: Self-RAG allows for flexible adjustment of retrieval frequency and adaptation to different tasks, enabling hard or soft constraints via reflection tokens. This allows for test-time customizations (e.g. balancing citation precision and completeness) without retraining.

Advantages

There are several key advantages of Self-RAG, including:

- On-demand retrieval reduces factual errors by retrieving external knowledge only when needed.

- By evaluating its own output and selecting the best segment, it achieves higher factual accuracy compared to standard LLMs and RAG models.

- Self-RAG maintains the versatility of LMs by not always relying on retrieved information.

- Adaptive retrieval with a threshold allows the model to dynamically adjust retrieval frequency for different applications.

- Self-RAG cites each segment and assesses whether the output is supported by the passage, making fact verification easier.

- Training with a critic model offline eliminates the need for a critic model during inference, reducing overhead.

- The use of reflection tokens enables controllable generation during inference, allowing the model to adapt its behavior.

- The model's use of a segment-level beam search allows for the selection of the best output at each step, combining generation with self-evaluation.

To read more, kindly visit this article