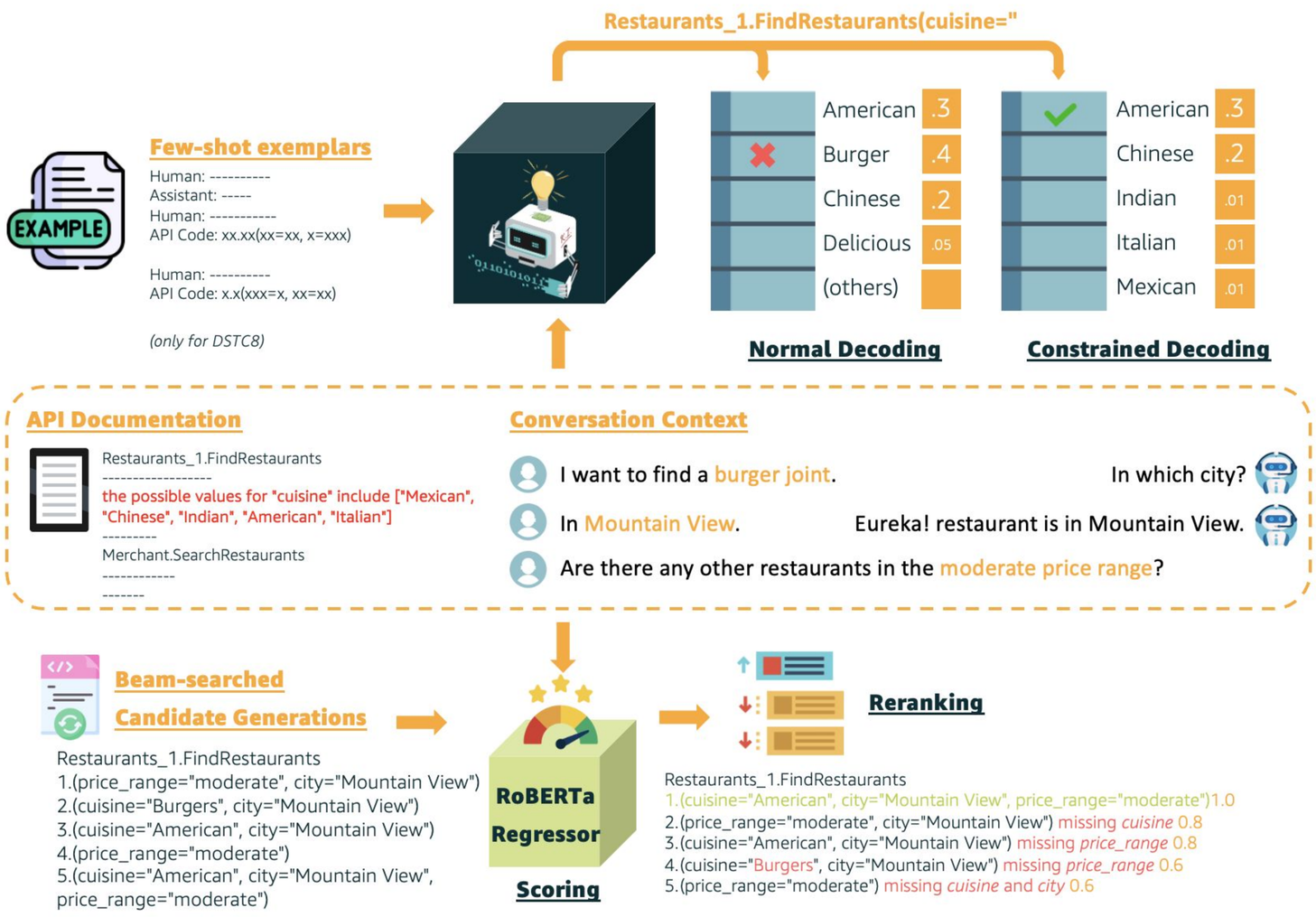
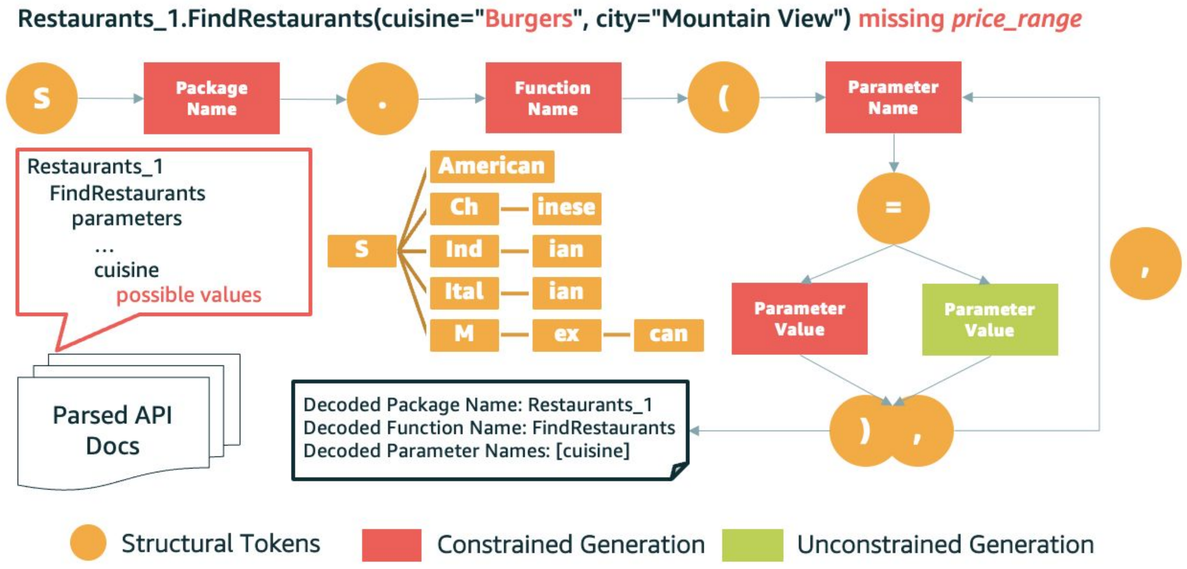
FANTAstic SEquences and Where to Find Them: Faithful and Efficient API Call Generation through State-tracked Constrained Decoding and Reranking

Zhuoer Wang, Leonardo F. R. Ribeiro, Alexandros Papangelis,
Rohan Mukherjee, Tzu-Yen Wang, Xinyan Zhao, Arijit Biswas, James Caverlee, and Angeliki Metallinou

API call generation is the cornerstone of large language models' tool-using ability that provides access to the larger world. However, **existing supervised and in-context learning approaches suffer from high training costs, poor data efficiency, and generated API calls that can be unfaithful to the API documentation and the user's request**. To address these limitations, we propose **an output-side optimization approach called FANTASE**. Two of the unique contributions of FANTASE are its State-Tracked Constrained Decoding (SCD) and Reranking components. **The SCD component dynamically incorporates appropriate API constraints in the form of Token Search Trie for efficient and guaranteed generation faithfulness with respect to the API documentation**. The **Reranking component efficiently brings in the supervised signal by leveraging a lightweight model as the discriminator to rerank the beam-searched candidate generations of the large language model**. We demonstrate the superior performance of FANTASE in API call generation accuracy, inference efficiency, and context efficiency with DSTC8 and API Bank datasets.



State-Tracked Constrained Decoding (SCD)



- FANTASE tracks the state of the generation by monitoring model-generated structural tokens that indicate the start/end of different API call units.
- Based on the state of the generation, FANTASE dynamically retrieves appropriate API constraints in the form of Token Search Trie from the parsed API document.
- FANTASE enforces constrained decoding at appropriate inference steps for efficient and guaranteed generation faithfulness with respect to the API documentation.

Reranking

Instead of returning the beam-searched candidate generation that has the highest sequence probability as the final generated API call, FANTASE employs a scorer to discriminate each of the candidate generations and rerank them accordingly. To train the scorer, we generate data as follows: we prompt the Alpaca-13B model with training set samples and obtain associated beam-searched candidate generations for each sample. For each candidate generation, the matching score with respect to the ground truth is calculated as the target of the scorer. Such data is used for the tuning of a RoBERTa-base model that has 125M parameters to predict the matching score.

Experiment Results

Datasets	DSTC8	API Bank
Basedline Methods		
GPT4	51.33	63.66
GPT3.5-turbo	49.28	59.40
Alpaca-13B	40.49	24.31
AlpDSTC/Lynx-7B	47.44	50.53
FANTASE		
Alpaca-13B + Reranking	46.42	33.33
Alpaca-13B + SCD	44.17	62.66
Alpaca-13B + SCD + Reranking	48.88	64.41
AlpDSTC/Lynx-7B + SCD	62.78	67.17

Generation Accuracy

- The SCD and Reranking components are effective and complementary.
- FANTASE can make the accuracy of small LLMs comparable to much larger models like GPT 3.5/4 with labeled data

Inference Efficiency

FANTASE improves the inference speed by 1.5~2.4 times with the novel proposal of decoding with constrained token search trie

Decoding Strategy	DSTC8 Inference Speed (sec/sample)	DSTC8 Speed Up	API Bank Inference Speed (sec/sample)	API Bank Speed Up
Greedy Search	5.32	-	5.85	-
SCD Greedy Search	3.42	x1.56	3.33	x1.76
Beam Search	15.12	-	23.15	-
SCD Beam Search	6.33	x2.39	10.25	x2.25

Setting	w. API Doc	w.o. API Doc	Δ
DSTC8 Unconstrained	37.63	33.74	-3.89
DSTC8 SCD	42.33	40.70	-1.63
API Bank Unconstrained	24.06	4.76	-19.3
API Bank SCD	56.64	22.81	-33.83

Context Efficiency

FANTASE is better at maintaining the performance when the API doc is absent from the input