KCTS: Knowledge-Constrained Tree Search Decoding with Token-Level Hallucination Detection

Sehyun Choi, Tianqing Fang, Zhaowei Wang, and Yangqiu Song
Department of CSE, Hong Kong University of Science and Technology, HK
{schoiaj, tfangaa, zwanggy, yqsong}@cse.ust.hk

Motivation

- Goal: Perform **knowledge-grounded generation** to solve the LM hallucination problem without incurring expensive training of the LM.
- Objective: Use **guided decoding** approach to control the output of LM at inference-time without tuning the LM weights.

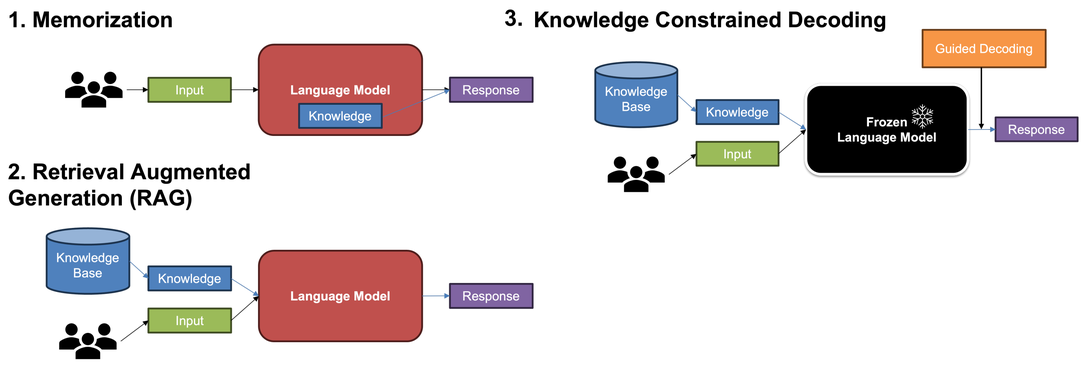


Figure 1: A schematic diagram showing the difference between memorizing knowledge in model parameters, augmenting model input with knowledge and training the LM (RAG), and our proposed knowledge constrained decoding, which does not incur LM weight update.

Guided Decoding for Knowledge-Grounded Generation

➢Knowledge-Groundedness has distinctive features:
❑Defined with a reference knowledge
❑Defined on the fully generated sequence.
➢Proposal:
❑ Define $\boldsymbol{f}(\boldsymbol{\alpha}, \boldsymbol{y}, \boldsymbol{k})$, which denotes the groundedness ($\boldsymbol{\alpha}$) of the generated sequence $\boldsymbol{y}$ with respect to the reference $\boldsymbol{k}$.
❑ Approximate sequence-level groundedness to token-level. (RIPA)
❑ Use Monte-Carlo Tree Search decoding to better estimate the future impact of current token selection. (MCTS Decoding)

Reward Inflection-Point Approximation (RIPA)

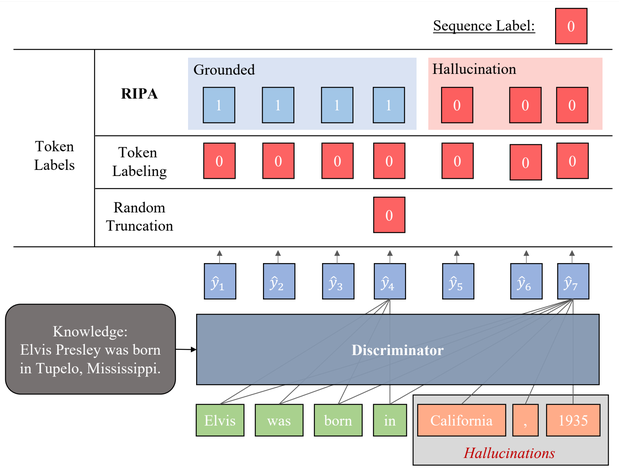
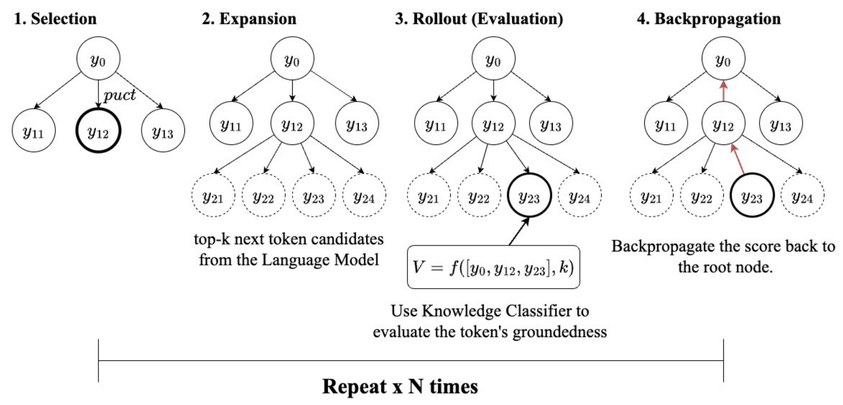


Figure 2: Comparison of RIPA against previous token-level approximations from guided decoding literature.

➢When approximating sequence-level groundedness to token-level, focus on the first position of hallucination.
❑Train on all subsequence of the input → sample efficient (vs. Random truncation)
❑Does not associate benign tokens before hallucination with the hallucination label. (vs. Token Labeling)

Monte Carlo Tree Search Decoding (MCTS)



➢Each node (step 3) is evaluated directly (no rollout) by RIPA → higher efficiency
➢MCTS selects the token that maximizes future score of f (groundedness) based on simulations

Pseudo-Negative Data Generation

➢Most knowledge-grounded generation benchmarks **do not** include negative data. To train the discriminator, we employed 2 approaches to **pseudo-negative data generation**:
❑ Knowledge Shuffle: Given a positive example (input, knowledge, response) from the dataset, swap the ground truth knowledge with another one randomly sampled from the dataset.
❑ Partial Hallucination: Given a positive example (input, knowledge, response), Perform the knowledge shuffle first, then randomly truncate the response and let the LM complete the response with high temperature sampling.

The KCTS Method

➢Knowledge-Constrained Tree Search Decoding (KCTS)
❑ KCTS = RIPA + MCTS
❑ KWD = RIPA + Weighted Decoding
➢Overview:
1. We train RIPA with lightweight adapters from LoRA on top of the the base LM (Flan-T5-XL in our experiments).
2. We decode each token using MCTS with fixed budget (50 simulations), using the groundedness score from the classifier (step 1) to evaluate partial sequences.

➢*Hypothesis: RIPA + MCTS (KCTS) together better estimates full-sequence groundedness, leading to more faithful sequences being generated.*

Experiment

Type	Model	K-Overlap KF1	K-Overlap K-Copy	Token Overlap F1	BLEU	RougeL	ChrF	METEOR	UniEval N	C	G	f
LLM	ChatGPT	49.41	39.71	30.32	6.91	26.24	34.95	31.67	57.62	96.41	96.15	95.82
	GPT-3.5	25.91	28.22	22.32	3.01	18.70	27.86	23.06	42.77	98.07	92.42	92.63
SFT	FT5-XL	39.85	37.79	28.08	9.41	25.11	31.17	25.40	76.44	92.36	95.16	97.90
Zero-Shot	FT5-XL	34.50	37.07	21.18	6.81	19.64	24.88	18.53	71.69	82.21	75.70	88.75
	FT5-XXL	28.20	32.33	19.11	5.53	17.55	24.15	17.16	72.37	84.24	75.51	85.89
	T0++	26.94	28.80	17.57	4.13	16.14	19.84	13.37	52.79	85.26	70.14	88.61
Decoding Baselines	FUDGE	55.30	54.04	29.43	11.72	27.35	31.50	26.00	73.68	88.20	83.53	94.54
	NADO	50.20	50.10	27.86	10.57	26.01	29.84	24.51	74.14	88.35	81.10	92.76
	MCTS	55.54	54.21	29.56	11.69	27.48	31.60	26.08	**74.54**	88.16	83.90	95.07
Ours	KWD	**58.19**	56.58	**30.71**	**12.74**	**28.27**	33.40	28.10	70.27	90.51	87.86	97.54
	KCTS	56.06	51.90	30.54	11.42	27.43	**35.22**	**28.92**	62.32	**92.78**	**91.78**	**98.30**

Table 1: Results on WoW Test set (unseen topics). SFT stands for supervised fine-tuning, and FT5 is shorthand for Flan-T5. Under the UniEval metrics, each letter stands for the following: **N** - Naturalness, **C** - Coherence, **G** - Groundedness. **Boldface** denotes the best performance, and underline denotes the second best. LLM performance is for reference and not for direct comparison.

Type	Model	K-Overlap KF1	K-Overlap K-Copy	Token Overlap F1	BLEU	RougeL	ChrF	METEOR	UniEval Coh.	Cons.	fluency	Relv.	MFMA score
LLM	ChatGPT	29.43	17.92	40.45	11.75	27.85	42.96	37.66	93.85	91.67	87.15	87.11	80.62
	GPT-3.5	27.54	16.94	38.96	10.78	26.63	41.17	35.38	92.56	90.33	85.73	85.78	78.74
SFT	FT5-XL	17.04	10.18	32.21	8.74	24.02	30.27	24.47	84.82	86.02	89.90	81.28	64.55
	FT5-XXL	17.45	10.42	31.55	8.43	23.38	29.95	23.91	87.17	88.58	90.00	82.28	68.37
	T0++	22.79	13.65	38.82	13.64	28.06	38.53	33.68	86.57	87.47	89.03	81.09	69.38
Decoding Baselines	FUDGE	18.68	10.70	33.51	9.32	24.83	31.06	24.93	90.52	90.61	83.37	82.00	71.35
	NADO	20.35	11.72	35.10	10.93	26.22	33.50	27.34	92.26	93.72	88.41	84.49	72.01
	MCTS	17.86	10.04	34.59	9.00	25.85	30.90	25.12	94.30	94.28	86.51	85.90	71.28
Ours	KWD	20.39	11.63	36.24	12.30	27.20	34.25	28.46	**96.24**	**96.64**	**91.60**	**88.48**	85.11
	KCTS	**22.97**	13.29	**38.27**	**14.21**	**28.10**	**37.18**	**31.37**	95.85	96.03	90.24	87.16	**85.36**

Table 2: Results on CNN/DM Test set. The guided decoding was conducted with FT5-XL model as the base model. **Coh.**, **Cons.**, and **Relv.** stand for coherence, consistency, and relevance, respectively.

➢In both Dialogue (WoW) and Summarization (CNN/DM) tasks, Knowledge Constrained Decoding (KCTS and KWD) outperforms previous decoding baselines, and even LLMs in some cases, in the knowledge-groundedness metrics. (KF1, Groundedness in WoW, and Consistency in CNN/DM)

Ablation Study

➢*Does KCTS really estimate future groundedness?*

T	K-Overlap KF1	K-Overlap K-Copy	Token Overlap BLEU	Token Overlap RougeL	UniEval C	UniEval G	f
5	48.78	48.22	10.17	25.39	90.58	85.87	90.58
10	48.24	48.05	9.98	25.87	90.22	86.41	85.43
16	51.49	48.67	11.07	26.44	92.83	89.99	92.76
32	56.06	51.90	11.42	27.43	92.78	91.78	98.30

Table 3: Number of initial tokens to be constrained to the knowledge with KCTS.

➢As more tokens are constrained using KCTS at the beginning, it provides the LM better starting point → the completion generated by LM is also more grounded.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rock-täschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In Advances in Neural Information Processing Systems, volume 33, pages 9459–9474. Curran Associates, Inc.
Rémi Coulom. 2007. Efficient selectivity and backup operators in monte-carlo tree search. In Computers and Games, pages 72–83, Berlin, Heidelberg. Springer Berlin Heidelberg.