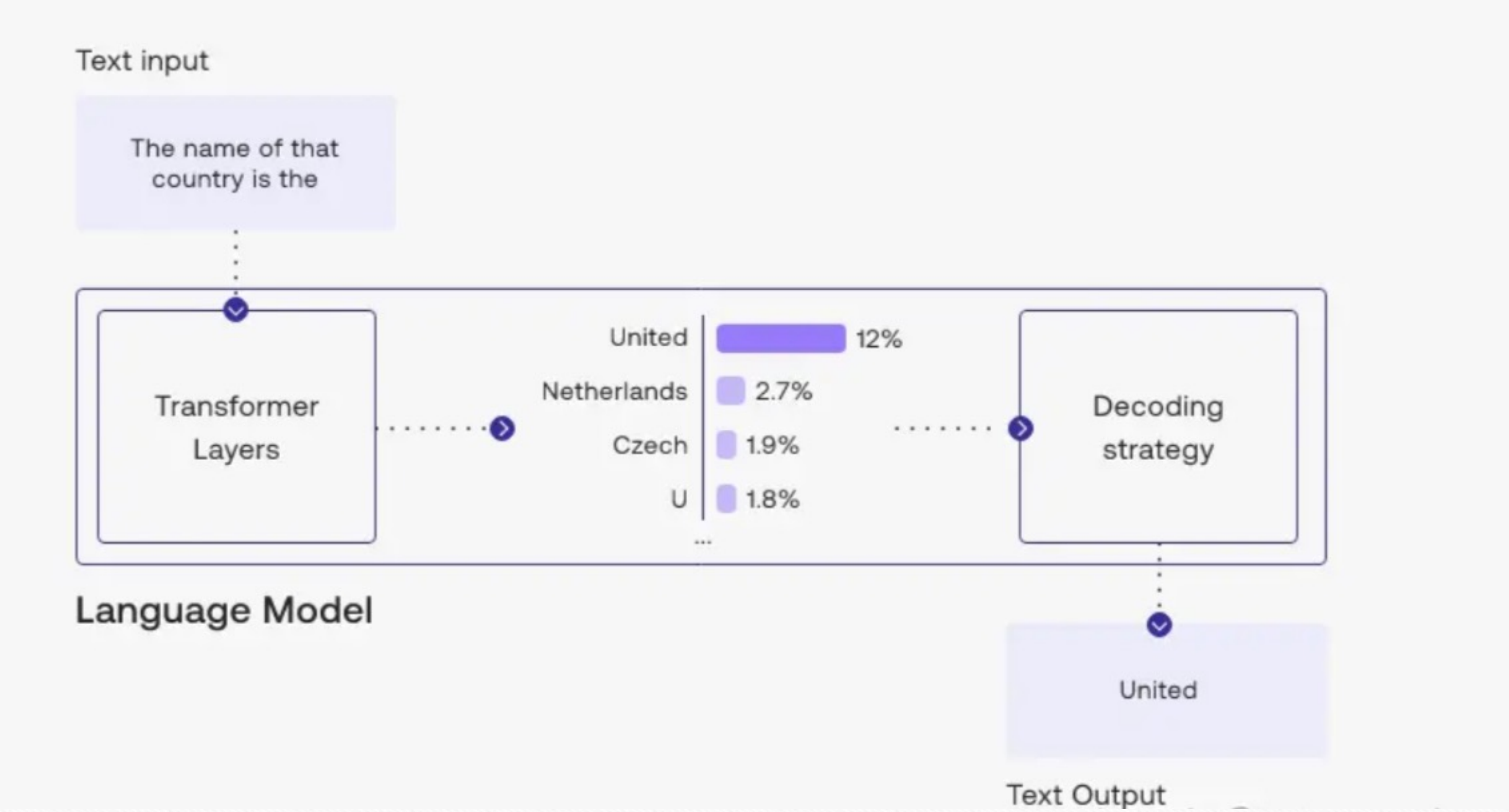
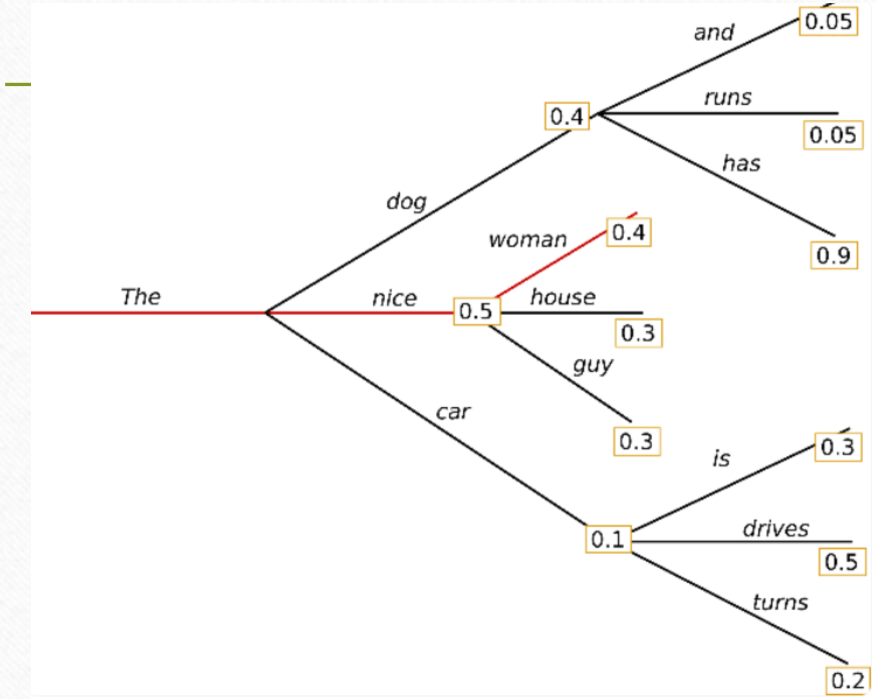
Decoding Strategy in LLM

Chengsong Huang

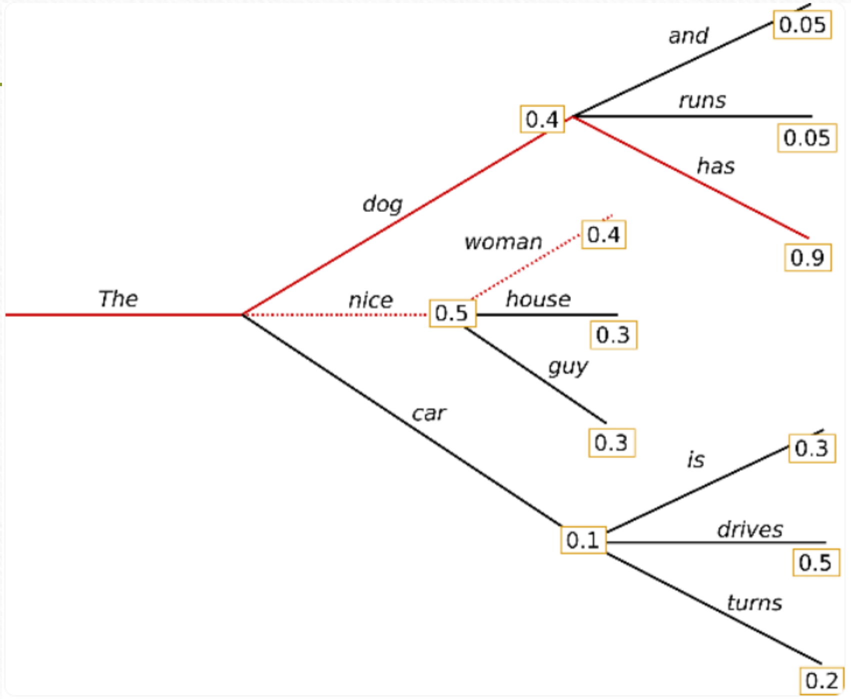
What is decoding



Search methods



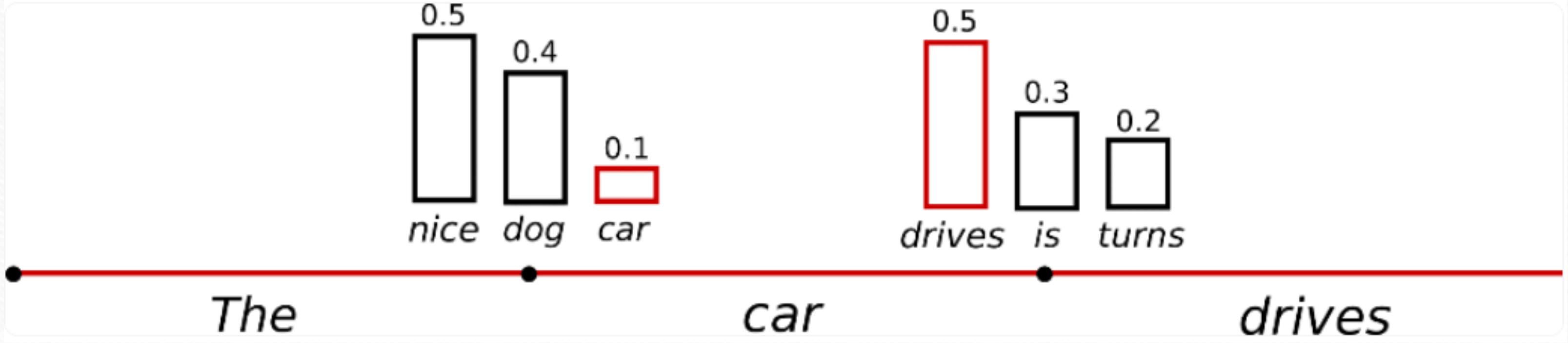
Greedy Search



Beam Search (beam=2)

Sampling Methods

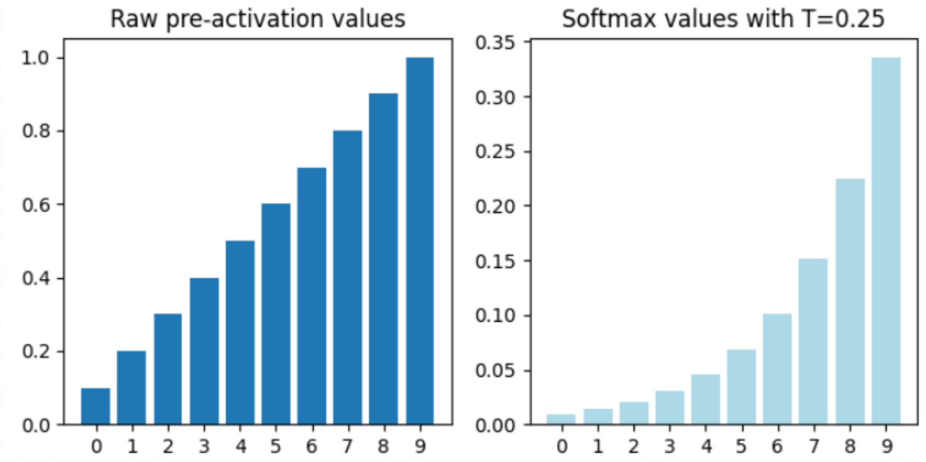
The basic method: sampling words from $w_t \sim P(w|w_{1:t-1})$

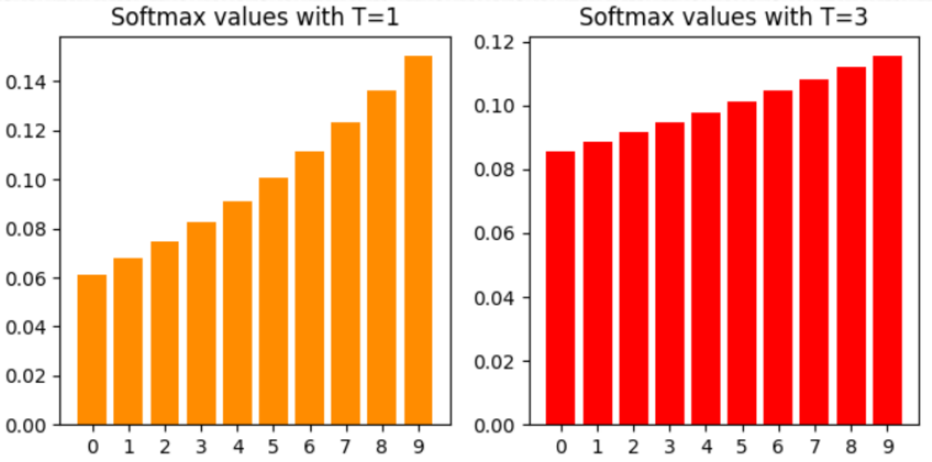


Sampling Methods

- Add Temperature in logits

$$p'(y_t|y_{<t},x) = \frac{\exp(\frac{(u(y_t|y_{<t},x))}{T})}{\sum_{j=1}^{n}\exp(\frac{(u(y_j|y_{<t},x))}{T})}$$





Top-k/Top-p

Top-K sampling works like this:

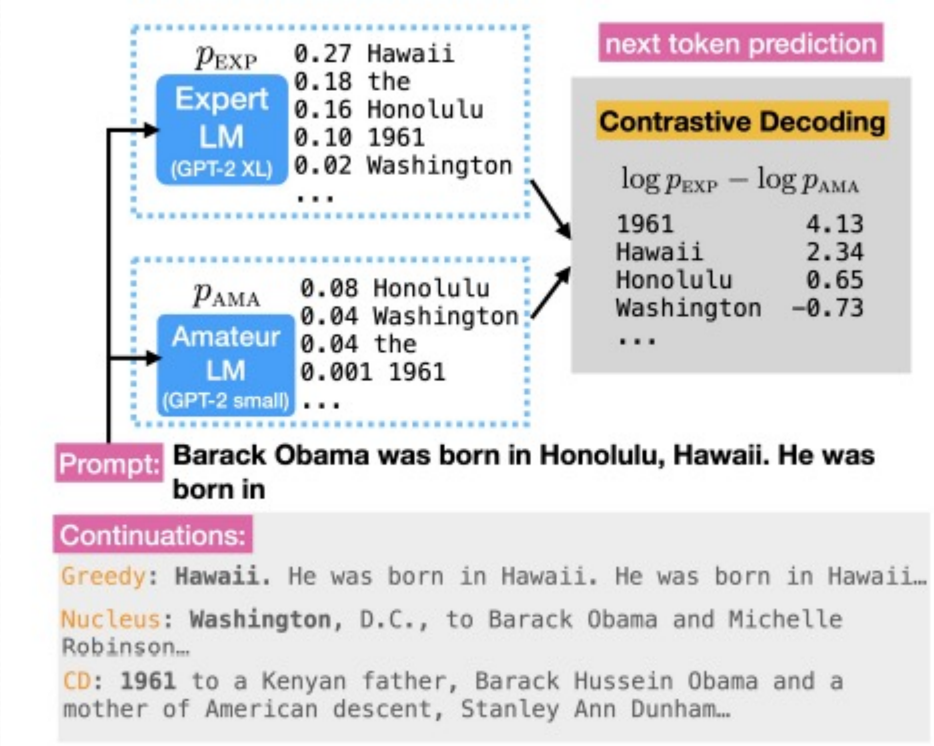
1. Order the tokens in descending order of probability.
2. Select the first K tokens to create a new distribution.
3. Sample from those tokens.

Top-p sampling works like this

1. Order the tokens in descending order of probability.
2. Select the smallest number of top tokens such that their cumulative probability is at least *p*.
3. Sample from those tokens.

Contrastive Decoding: Open-ended Text Generation as Optimization

- Contrastive：using negative sample to better learn



Li, Xiang Lisa et al. "Contrastive Decoding: Open-ended Text Generation as Optimization." *ACL*(2022).

Two potential problems

- False positives : Some tokens have both small probabilities in Expert model and weak model, but the probability in weak model is very very very small to make $\log p_{\text{EXP}} - \log p_{\text{AMA}}$ large.

- False negatives: Weak model are also very confident in some easy predictions, making $\log p_{\text{EXP}} - \log p_{\text{AMA}}$ small.

Solution to these problems

- Adaptive plausibility constraint
- Similar to top-p sampling

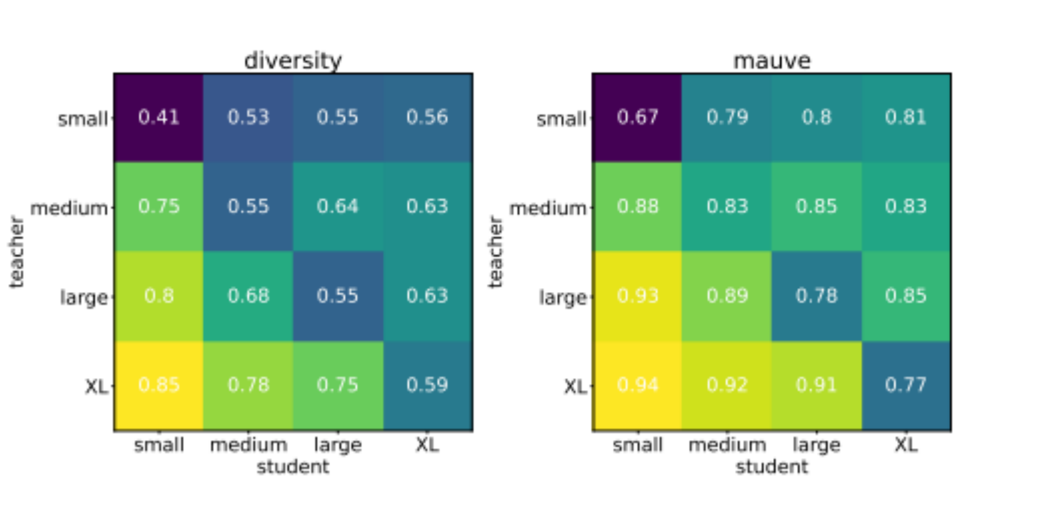
$$\mathcal{V}_{\text{head}}(x_{<i}) = \tag{1}$$
$$\{x_i \in \mathcal{V} : p_{\text{EXP}}(x_i \mid x_{<i}) \geq \alpha \max_w p_{\text{EXP}}(w|x_{<i})\}$$

$$\text{CD-score}(x_i; x_{<i}) \tag{3}$$
$$= \begin{cases} \log \frac{p_{\text{EXP}}(x_i|x_{<i})}{p_{\text{AMA}}(x_i|x_{<i})}, & \text{if } x_i \in \mathcal{V}_{\text{head}}(x_{<i}), \\ -\inf, & \text{otherwise.} \end{cases}$$

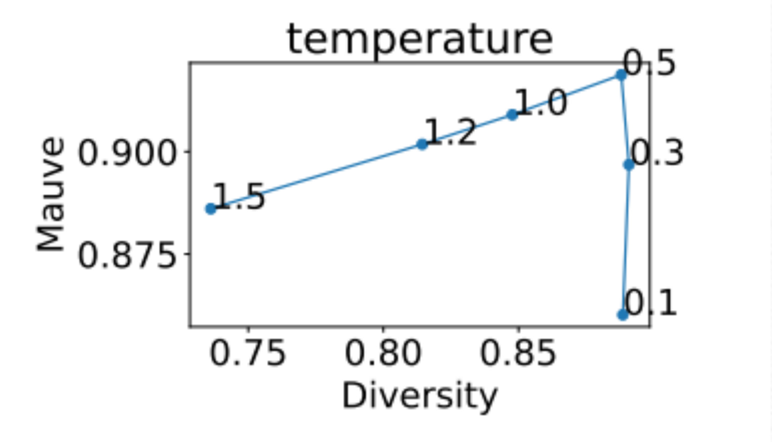
Evaluation

	name	wikinews DIV	wikinews MAUVE	wikinews COH	wikitext DIV	wikitext MAUVE	wikitext COH	story DIV	story MAUVE	story COH
OPT-13B	max prob	0.08	0.3	0.65	0.03	0.08	0.63	0.02	0.05	0.51
	k=50	0.91	0.92	0.64	0.72	0.77	0.64	0.91	0.9	0.51
	p=0.95	0.92	0.92	0.62	**0.92**	0.89	0.55	0.93	0.91	0.48
	typical=0.95	**0.94**	0.9	0.59	0.89	0.86	0.58	**0.95**	0.91	0.46
	CS(Su et al., 2022)	0.92	0.87	0.59	0.87	0.77	0.52	0.81	0.78	0.47
	CD	**0.94**	**0.94**	**0.69**	0.91	**0.91**	**0.69**	0.89	**0.94**	**0.62**
GPT2-XL	max prob	0.04	0.14	0.65	0.02	0.05	0.62	0.01	0.03	0.49
	k=50	0.92	0.88	0.64	0.87	0.79	0.61	0.91	0.87	0.51
	p=0.95	0.94	0.9	0.6	0.92	0.87	0.57	0.94	0.91	0.46
	typical=0.95	**0.95**	0.91	0.56	**0.95**	0.84	0.53	**0.96**	0.88	0.43
	CS(Su et al., 2022)	0.93	0.82	0.62	0.86	0.75	0.59	0.88	0.78	0.48
	CD	0.92	**0.94**	**0.69**	0.89	**0.92**	**0.69**	0.83	**0.94**	**0.64**

More Important Analysis



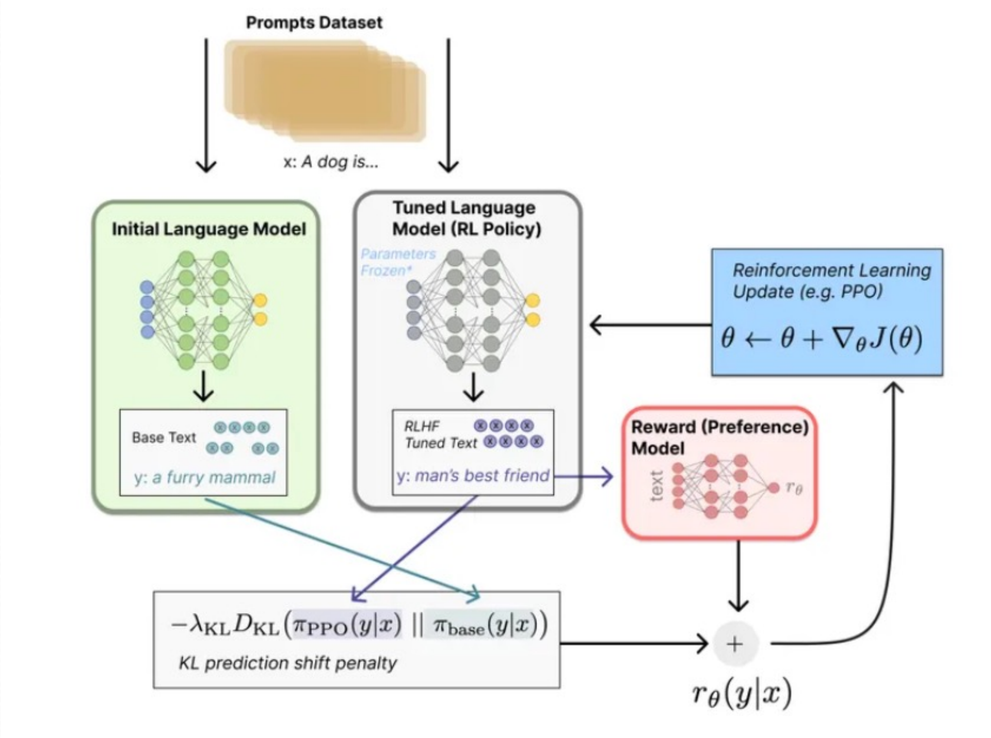
The larger gap between two models, the better performance improvement



The temperature will influence the diversity and generation quality

Using Reward model in Decoding

- LLM as policy model
- When decoding, we use LLM alone



Liu, Jiacheng et al. “Don't throw away your value model! Making PPO even better via Value-Guided Monte-Carlo Tree Search decoding.” (2023).

Proximal Policy Optimization

- To train a policy network(LLM)
- Reward model is for the whole sentences. Policy loss is for the next words.

Policy Objective Function

$$L^{PG}(\theta) = E_t[\underbrace{\log \pi_\theta(a_t|s_t)}_{\text{log probability of taking that action at that state}} * \underbrace{A_t}_{\text{Advantage}}]$$

log probability of taking that action at that state

Advantage if A>0, this action is better than the other action possible at that state

How to Used the Reward Model

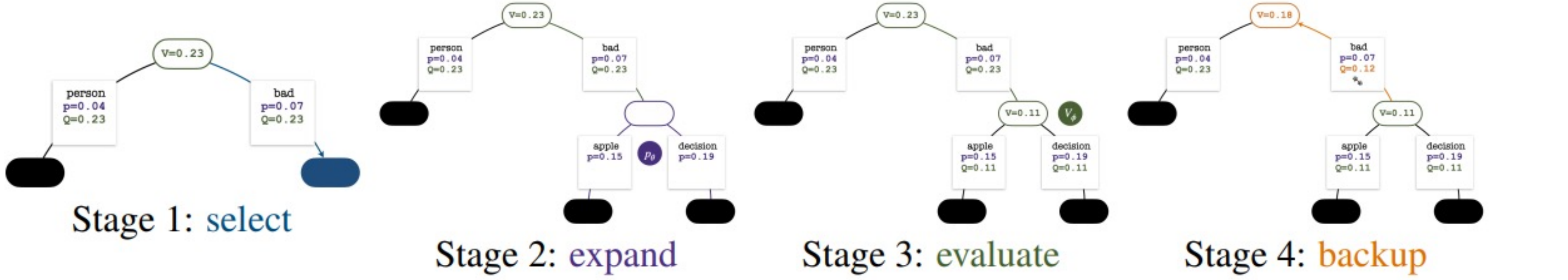
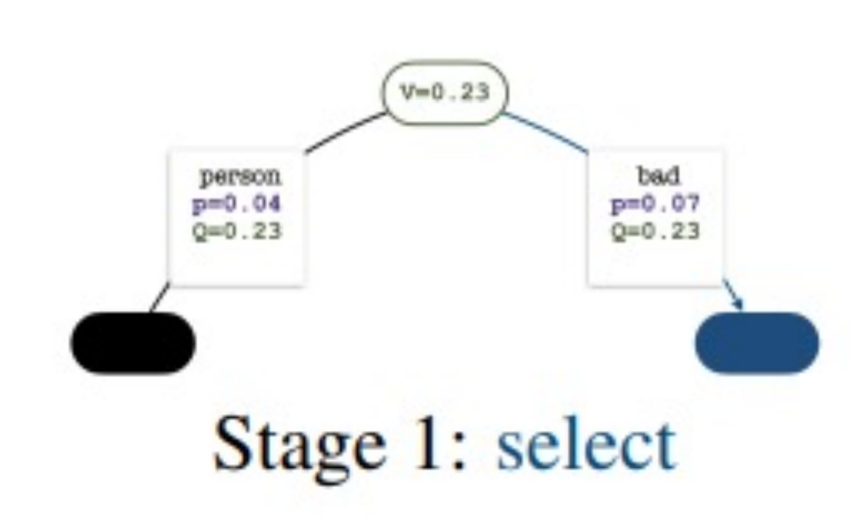


Figure 2: The four stages of one simulation in MCTS. Note: we displayed the node visit count $N(s)$ on its parenting edge as the number of "paws" (e.g., in the `bad` token in the backup stage).

Select Stage

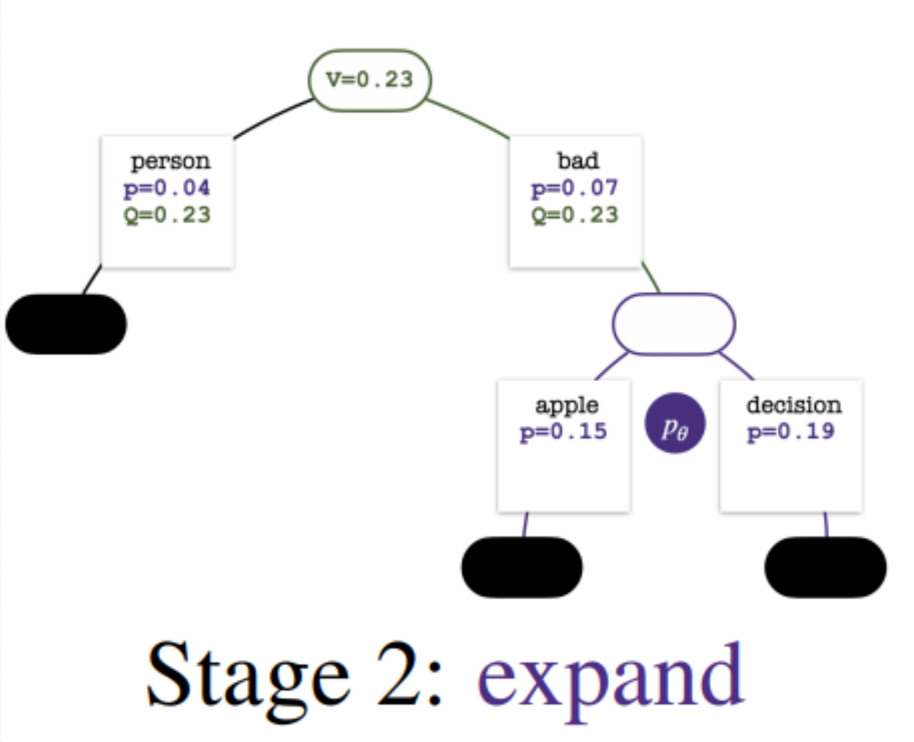
- Polynomial Upper Confidence Trees:

$$a^* = \arg\max_a \left[Q(s,a) + c_{\text{puct}} \cdot p_\theta(a|s) \frac{\sqrt{N(s)}}{1+N(s')} \right]$$



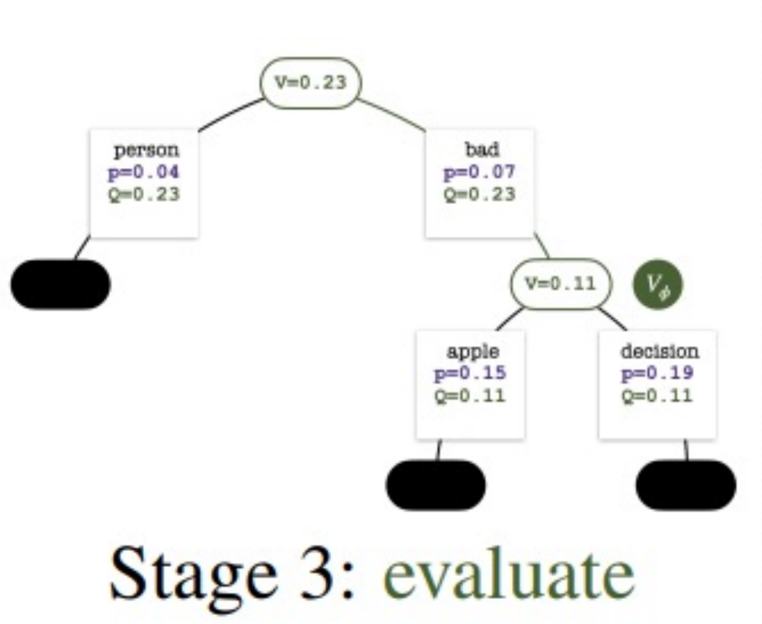
Expand Stage

- Using top-k to find new nodes



Evaluate Stage

- Using reward model to get the value of the modes, then using the average value as the value of the father nodes.



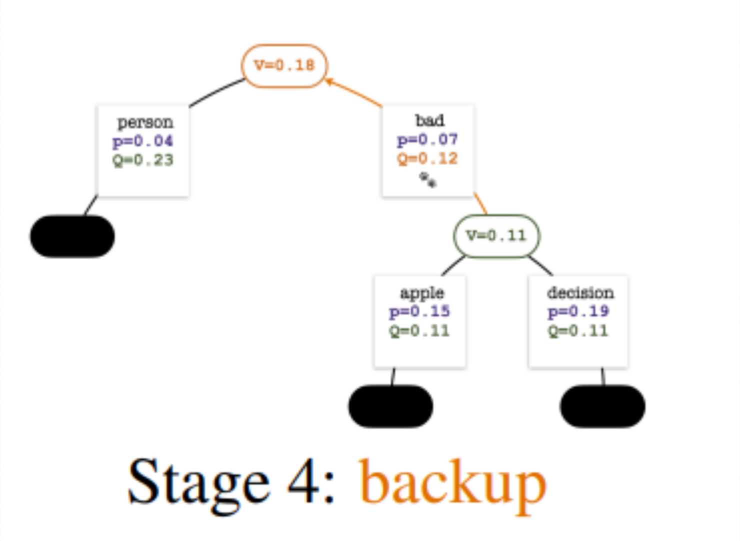
Backup Stage

- Update the visit counts and the value in the line to this nodes.

$$Q(s,a) \leftarrow r + \gamma \bar{V}(s'),$$

$$\bar{V}(s) \leftarrow \sum_a N(s')Q(s,a) / \sum_a N(s'),$$

$$N(s) \leftarrow N(s) + 1.$$



Main Results

Table 1: Results on *sentiment steering*. **Upper:** automatic evaluation (the middle lines are ablation results discussed later in §5.1). **Lower:** human evaluation.

	Desired sentiment: POSITIVE				Desired sentiment: NEGATIVE			
	% Desired	**Fluency**	**Diversity**		**% Desired**	**Fluency**	**Diversity**	
	(↑)	output ppl (↓)	dist-2 (↑)	dist-3 (↑)	(↑)	output ppl (↓)	dist-2 (↑)	dist-3 (↑)
PPO (Lu et al., 2022)	52.44	3.57	0.82	0.81	65.28	3.57	0.83	0.83
PPO + best-of-n	51.47	3.56	0.83	0.82	65.62	3.57	0.83	0.83
PPO-MCTS[R]	81.00	3.80	0.85	0.84	–	–	–	–
PPO + stepwise-value	62.47	4.94	0.89	0.87	–	–	–	–
PPO (4x more steps)	75.50	3.87	0.83	0.82	83.63	3.37	0.82	0.83
PPO-MCTS (ours)	**86.72**	3.42	0.79	0.81	**91.09**	3.44	0.80	0.82

Questions?