# Speculative cascades: faster cascades via speculative decoding


Hari Narasimhan


Wittawat Jitkrittum


Ankit Singh Rawat


Seungyeon Kim[†]


Neha Gupta[‡]


Aditya Menon

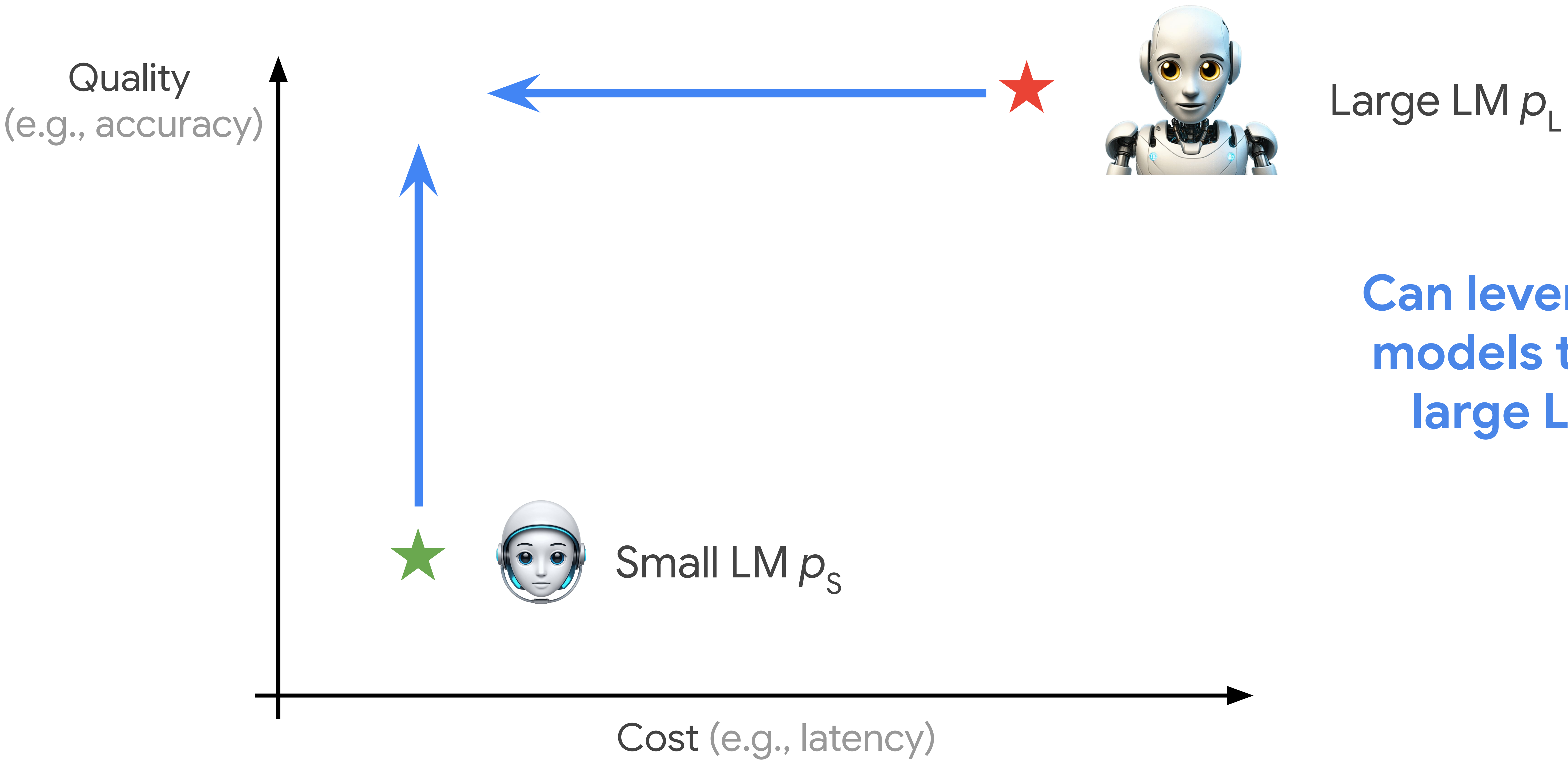

Sanjiv Kumar

Google Research

† Now at Meta ‡ Now at Mistral

# Tradeoffs in LM scaling

- Scaling LMs generally imposes **inference cost versus quality** tradeoffs

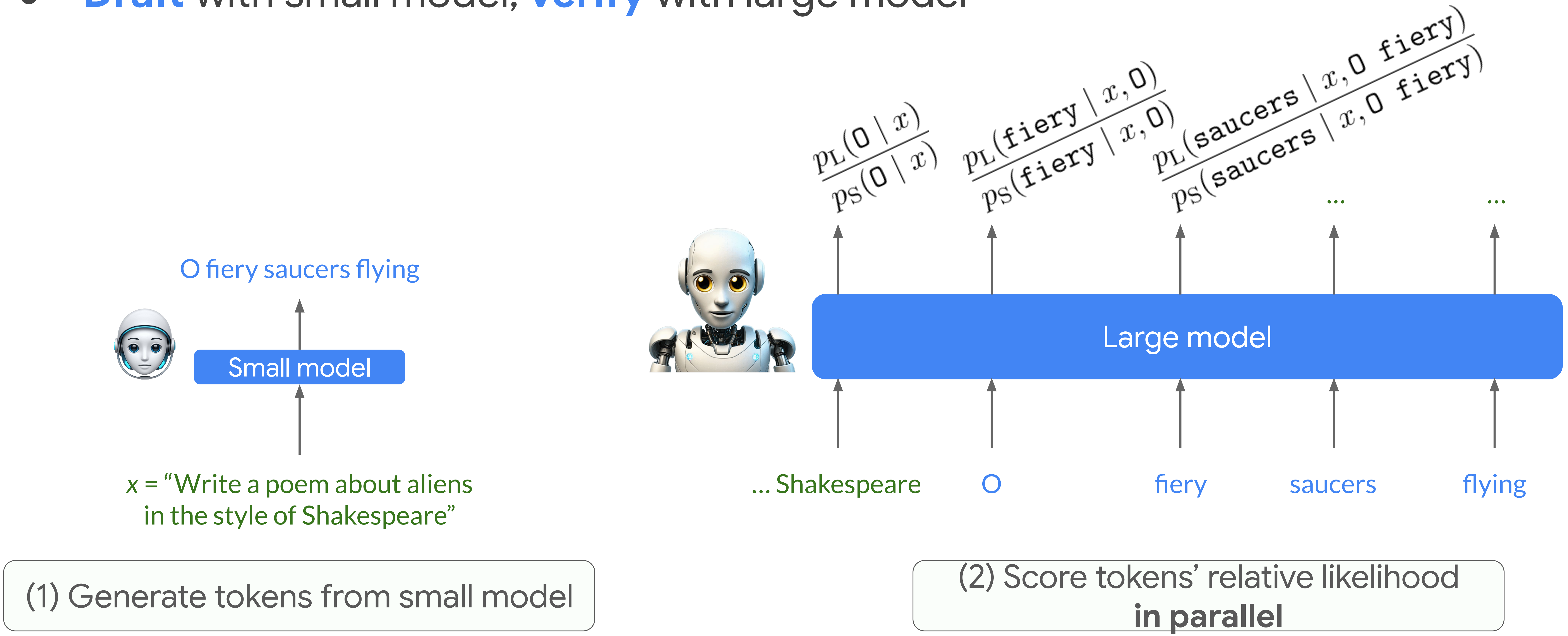



**Can leverage both models to reduce large LM cost!**

Icons from emojis.com

# Speculative decoding

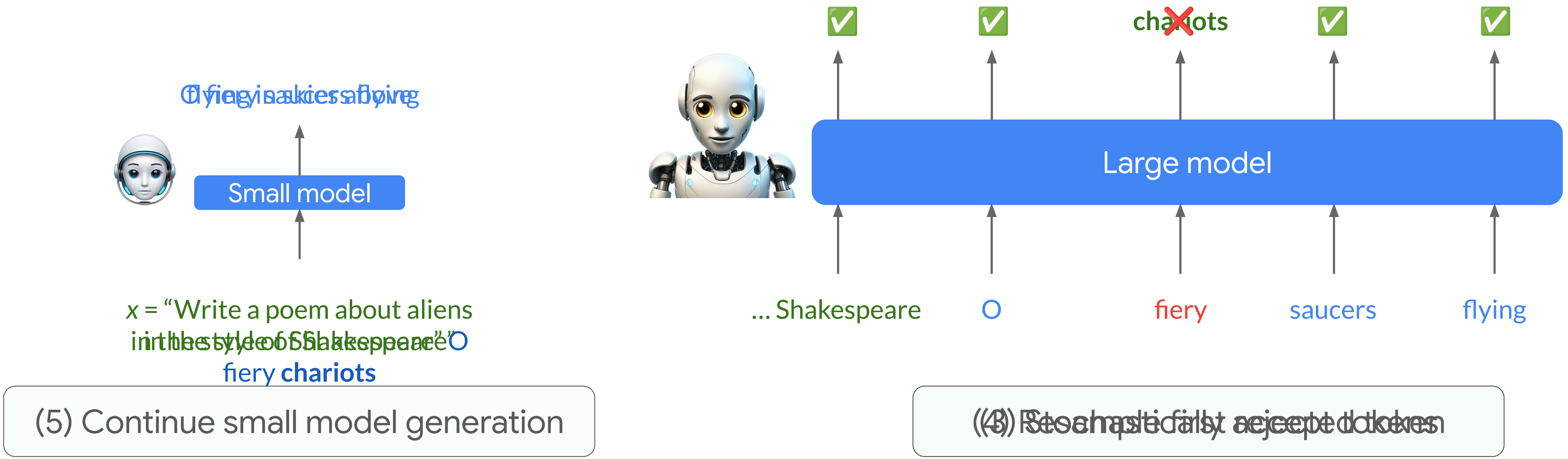
- **Draft** with small model; **verify** with large model




Leviathan et al. Fast Inference from Transformers via Speculative Decoding. ICML 2023.
Chen et al. Accelerating large language model decoding with speculative sampling. arXiV 2023.

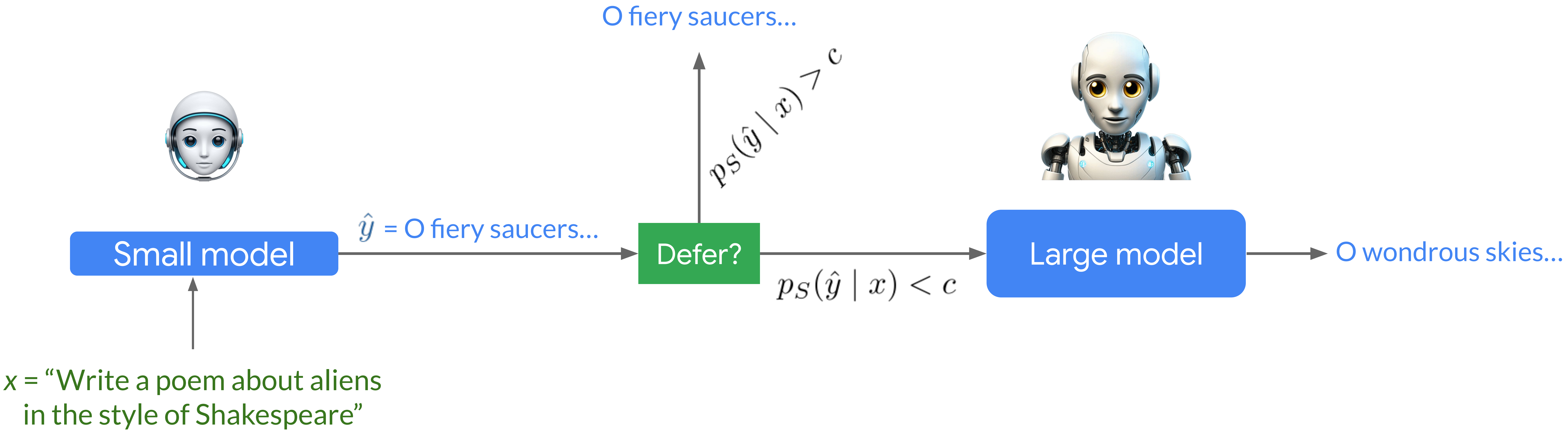
# Speculative decoding

- **Draft** with small model; **verify** with large model



# Cascades

- Try to use small model; if **uncertain**, defer to large model



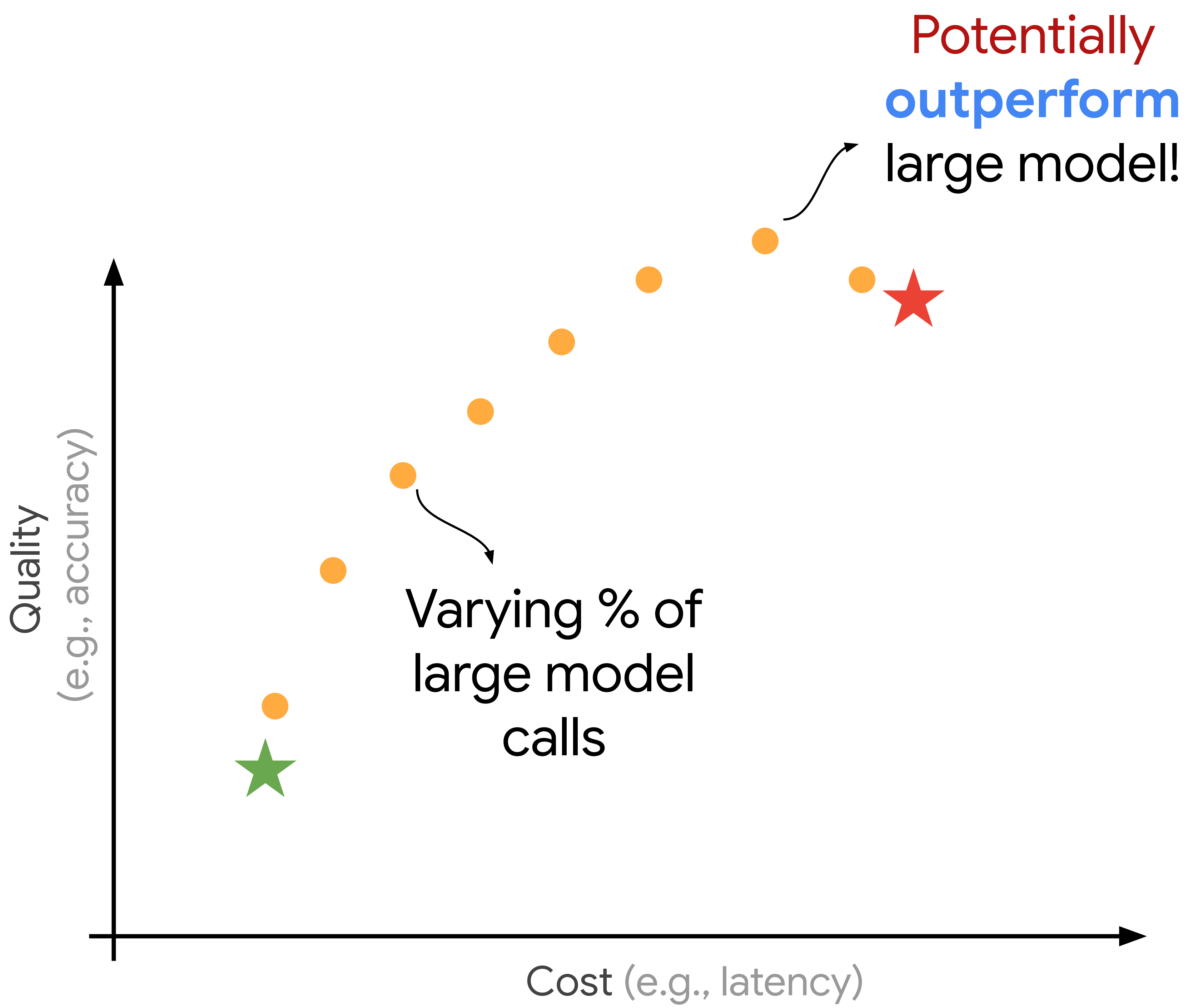


Chen et al. FrugalGPT: How to use large language models while reducing cost and improving performance. TMLR 2024.
Gupta et al. Language model cascades: token-level uncertainty and beyond. ICLR 2024.

# A tale of two inference strategies

## Speculative Decoding


## Cascades





Kim et al. Speculative Decoding with Big Little Decoder. NeurIPS 2023.
Jitkrittum et al. When does confidence based cascade deferral suffice? NeurIPS 2023.


Google

# A tale of two inference strategies

**Speculative Decoding**

**Draft** with small model,
**verify** with large model

Quality-**preserving** speedup

Mimic
**large model distribution**

**Cascades**

Try to use small model;
if **uncertain**, use large model

Quality-**enhancing** speedup

(potentially )

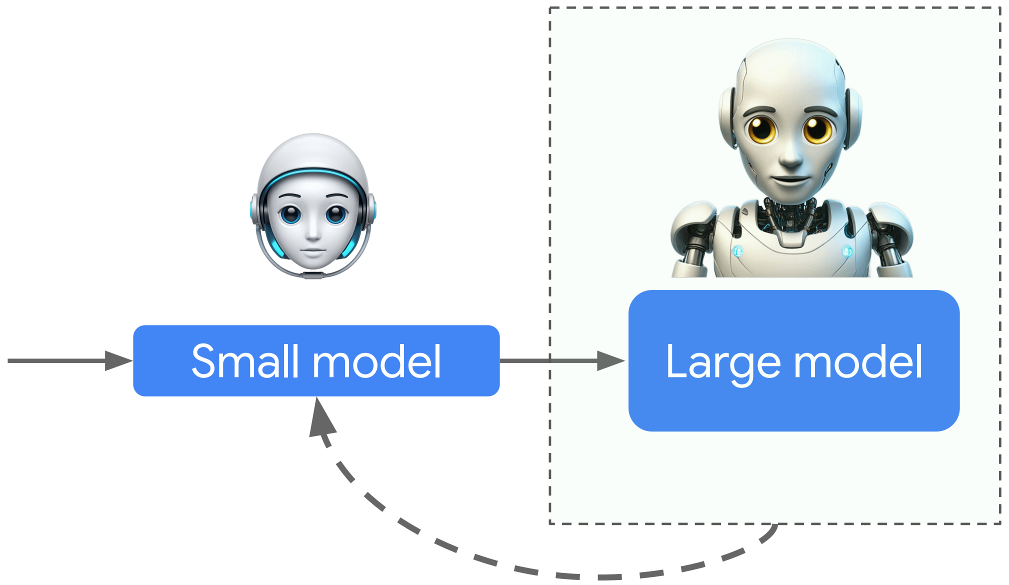
Mimic
**data-generating distribution**

**Can we leverage the best of both approaches?**

Google

# Speculative cascades: summary

**Target = $p_L$**



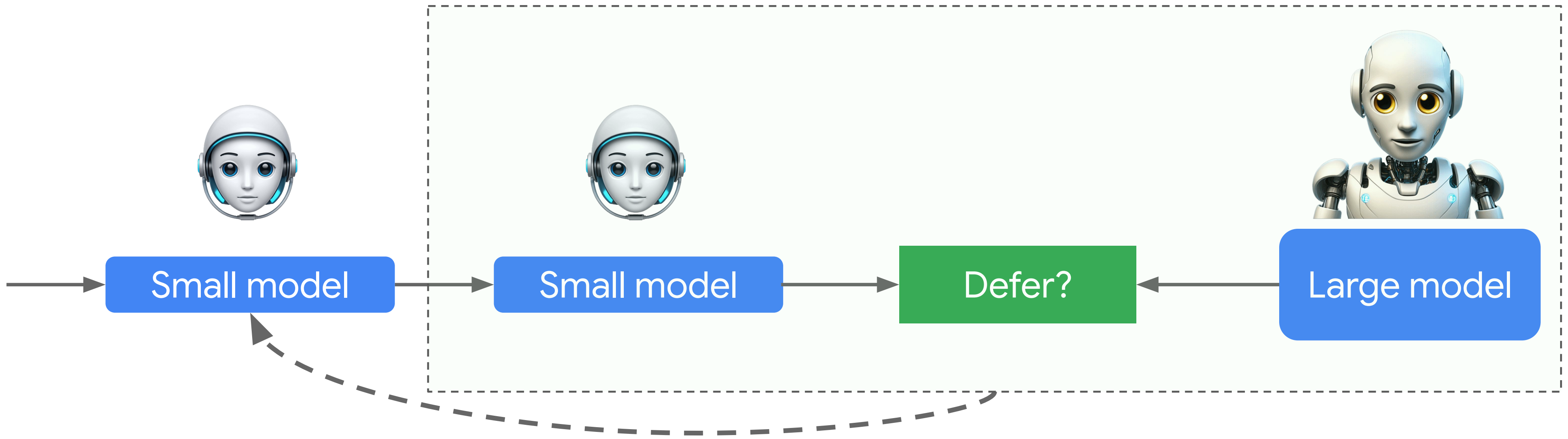

Drafter
(sequential)

Verifier
(parallel)

**Guarantee**
Probability of sampling $v$ is $p_L(v)$

**Speculative Decoding**

**Target = $\pi$**




Drafter
(sequential)

Verifier
(parallel)

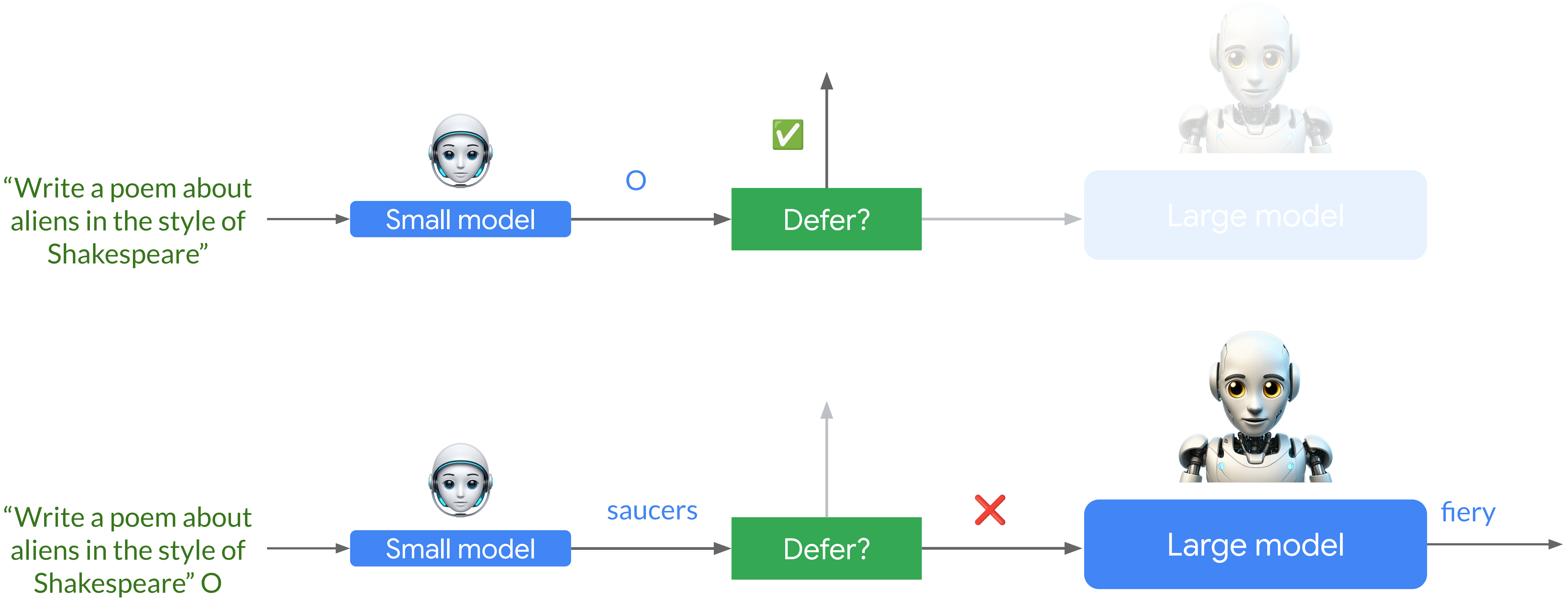
**Guarantee**
Probability of sampling $v$ is $\pi(v)$

**Speculative Cascading**
**(our proposal)**

Google

# Token-level cascades

- Consider a **token-level** cascade




**How to design the deferral rule?**

# (Bayes-)Optimal cascade deferral

- Given any sequence $x_{<t} = x_1, \ldots, x_{t-1}$, we want a **deferral rule** $r(x_{<t}) \in \{0, 1\}$
    - $r(x_{<t}) = 1 \Leftrightarrow$ invoke large model

- What does the ideal rule look like?

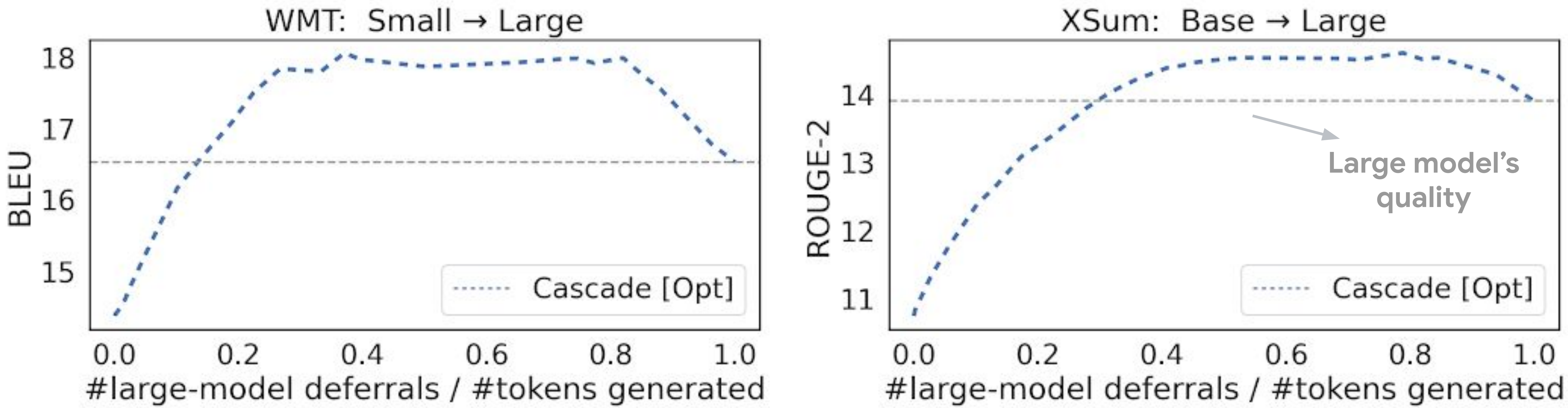
# (Bayes-)Optimal cascade deferral

- Given any context $x_{<t} = x_1, \ldots, x_{t-1}$, we want a **deferral rule** $r(x_{<t}) \in \{0, 1\}$
    - $r(x_{<t}) = 1 \Leftrightarrow$ invoke large model

**Requires calling large model!**

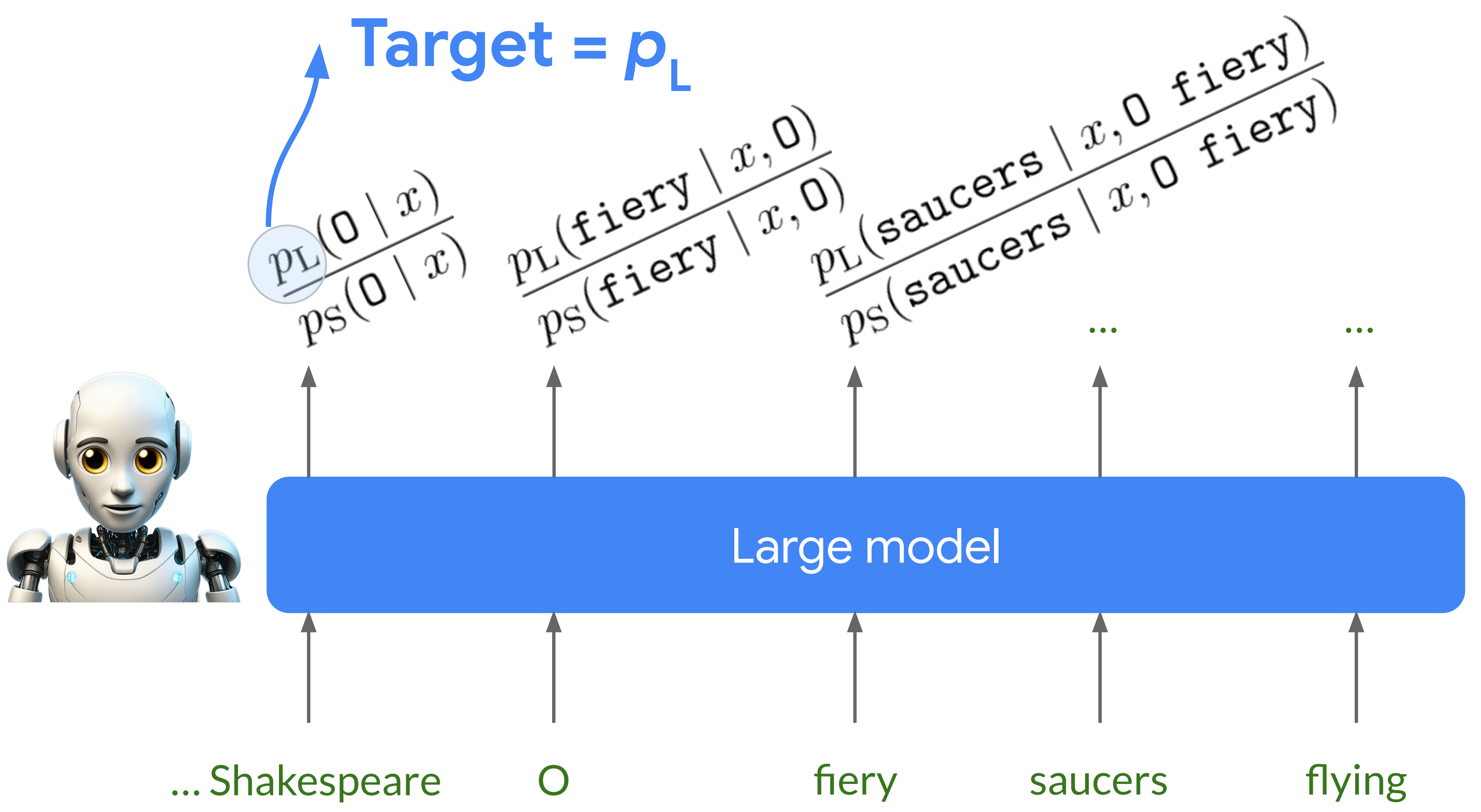
- How does the ideal rule perform?

$$r^*(x_{<t}) = 1 \iff \mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\ell(v, p_{\text{Small}})] - \mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\ell(v, p_{\text{Large}})] > c$$

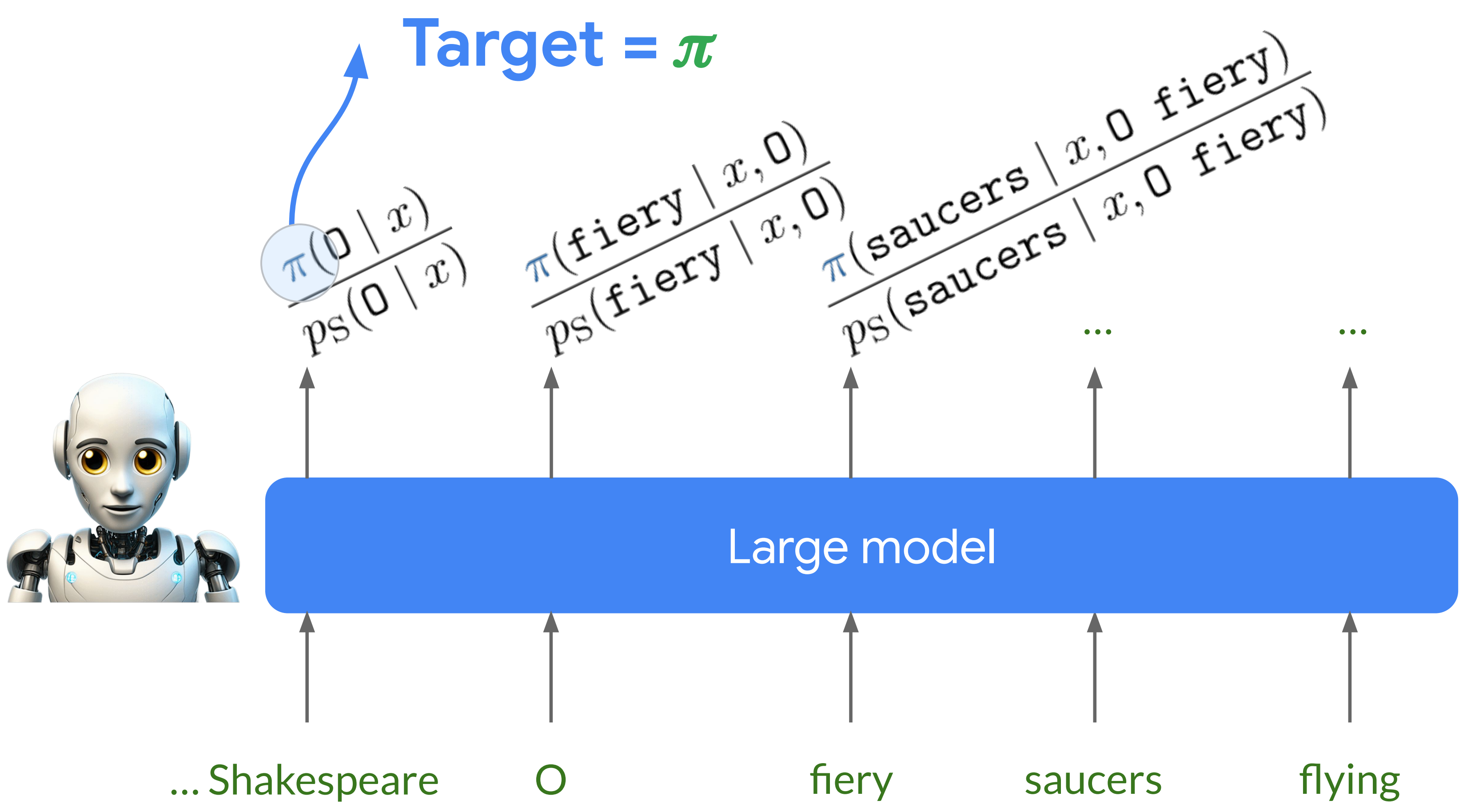


# Speculative cascades

- Speculative execution using **alternate target distribution** for verification!




**Speculative Decoding**




**Speculative Cascading (our proposal)**

# Speculative cascades: deferral rules

- Speculative execution using **alternate target distribution** for verification!
- Target distribution $\pi$ is defined by a **deferral rule**:

$$\pi(\cdot) = (1 - r(x_{<t})) \cdot p_{\text{Small}}(\cdot) + r(x_{<t}) \cdot p_{\text{Large}}(\cdot)$$

**Fact**: The **Bayes-optimal** token deferral rule $r^*$ is

$$r^*(x_{<t}) = 1 \iff \underbrace{\mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\ell(v, p_{\text{Small}})] - \mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\ell(v, p_{\text{Large}})]}_{\text{Expected loss gap}} > c \cdot \underbrace{D_{\text{TV}}(p_{\text{Large}}, p_{\text{Small}})}_{\text{Total probability gap}}$$

* The rule is Bayes-optimal for minimising the expected loss against the ground-truth token, subject to a bound on the rejection rate.

# Speculative cascades: deferral rules

- Speculative execution using **alternate target distribution** for verification!
- Target distribution $\pi$ is defined by a **deferral rule**:

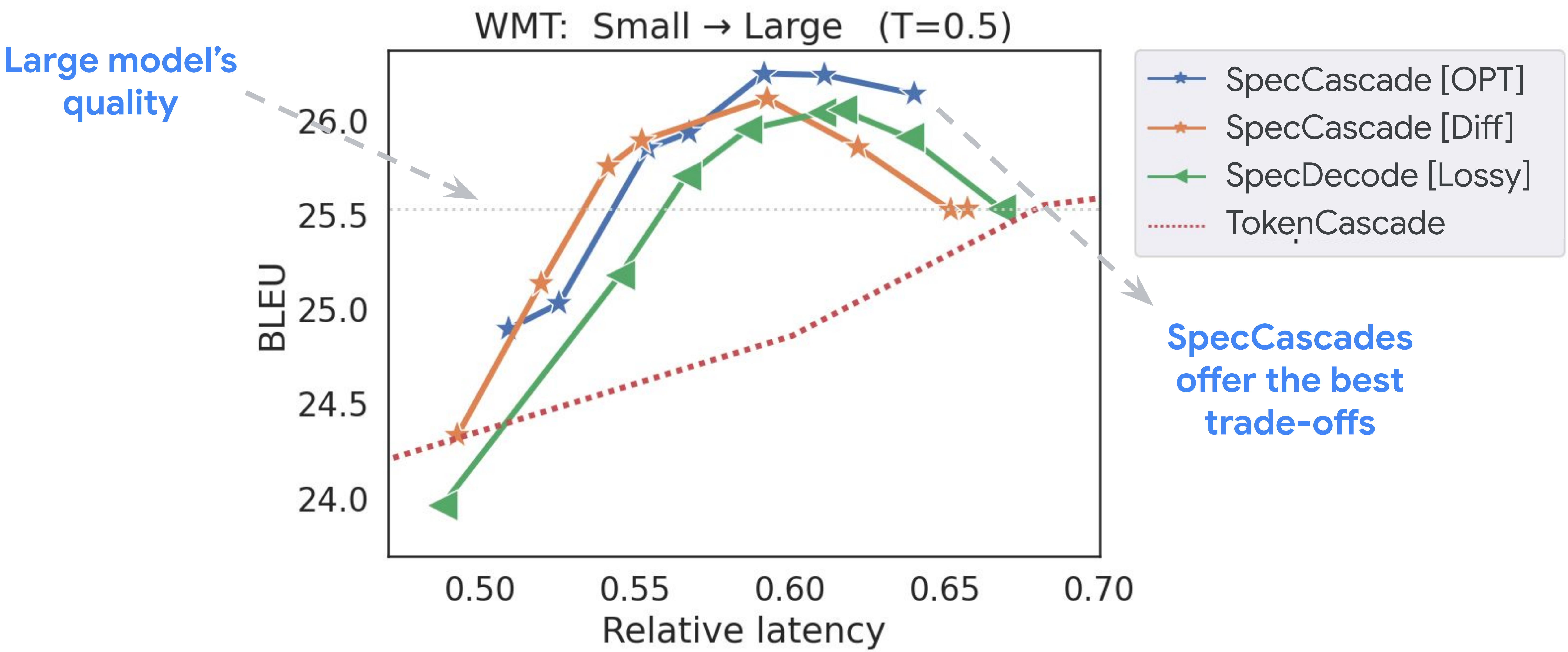
$$\pi(\cdot) = (1 - r(x_{<t})) \cdot p_{\text{Small}}(\cdot) + r(x_{<t}) \cdot p_{\text{Large}}(\cdot)$$

An approximation to the Bayes-optimal rule *r* is

$$\hat{r}(x_{<t}) = 1 \iff \underbrace{\max_{v \in \mathcal{V}} p_{\text{Large}}(v \mid x_{<t}) - \max_{v \in \mathcal{V}} p_{\text{Small}}(v \mid x_{<t})}_{\text{Confidence gap}} > c \cdot \underbrace{D_{\text{TV}}(p_{\text{Large}}, p_{\text{Small}})}_{\text{Total probability gap}}$$

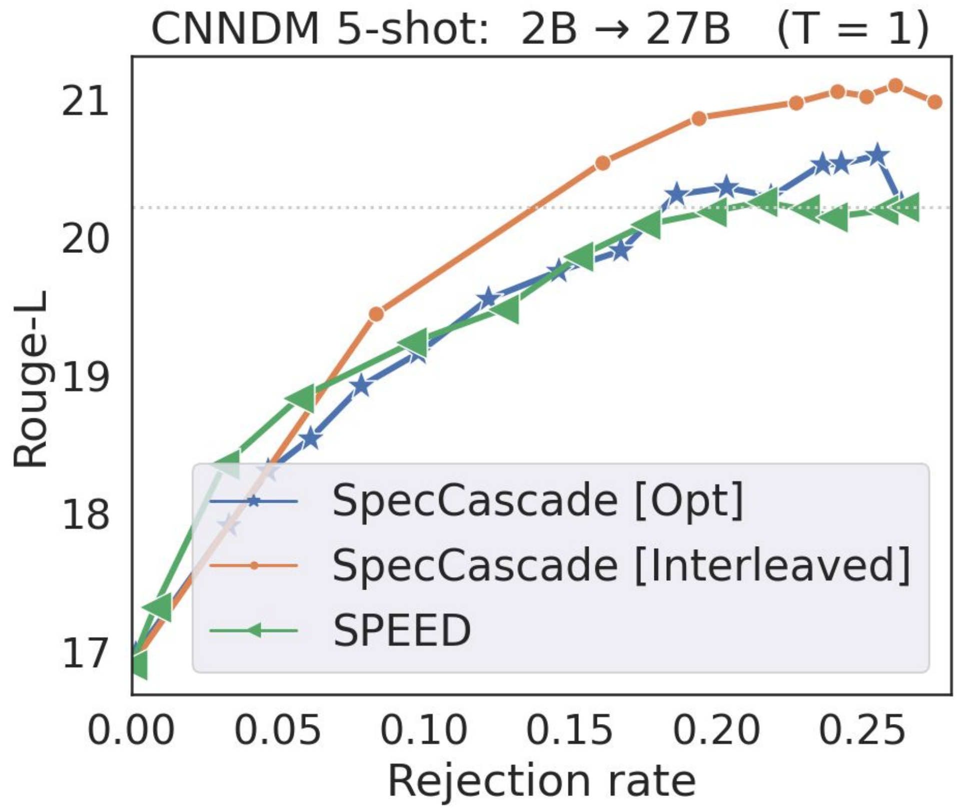
* The rule is Bayes-optimal for minimising the expected loss against the ground-truth token, subject to a bound on the rejection rate.

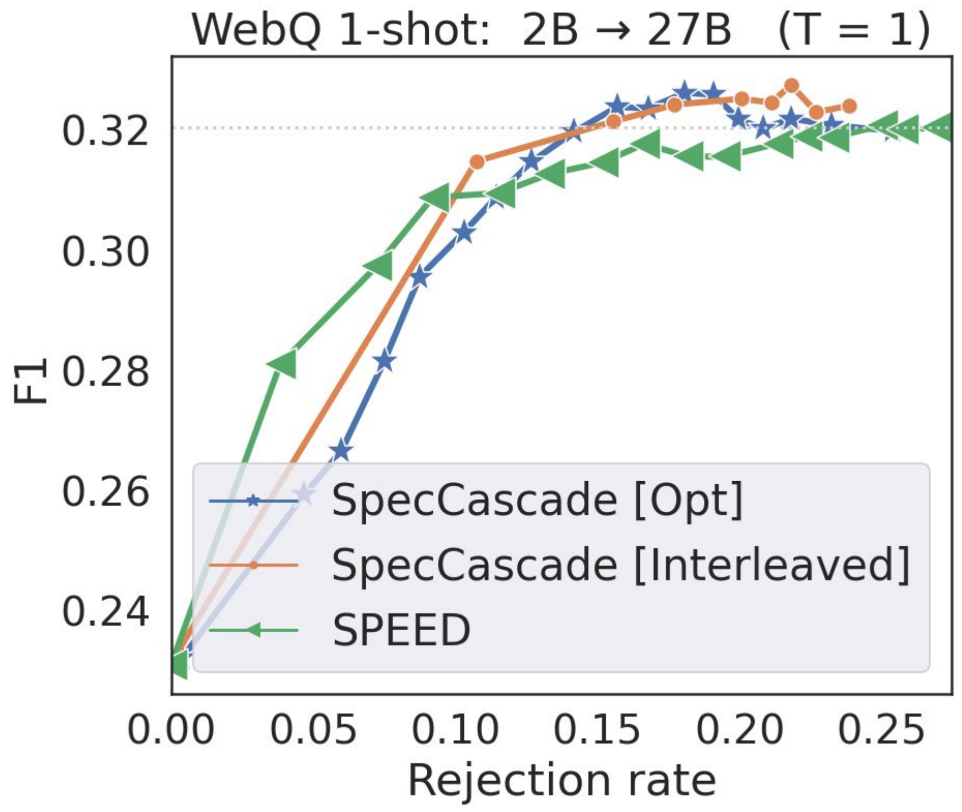
# Empirical results: fine-tuned T5 models

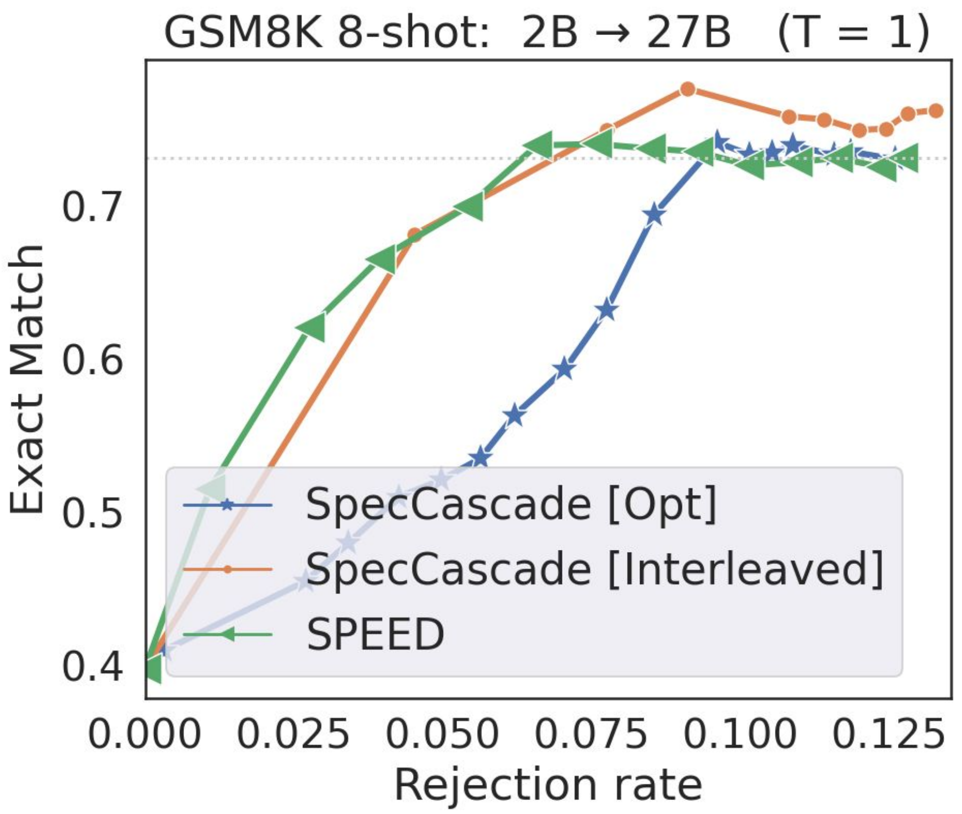


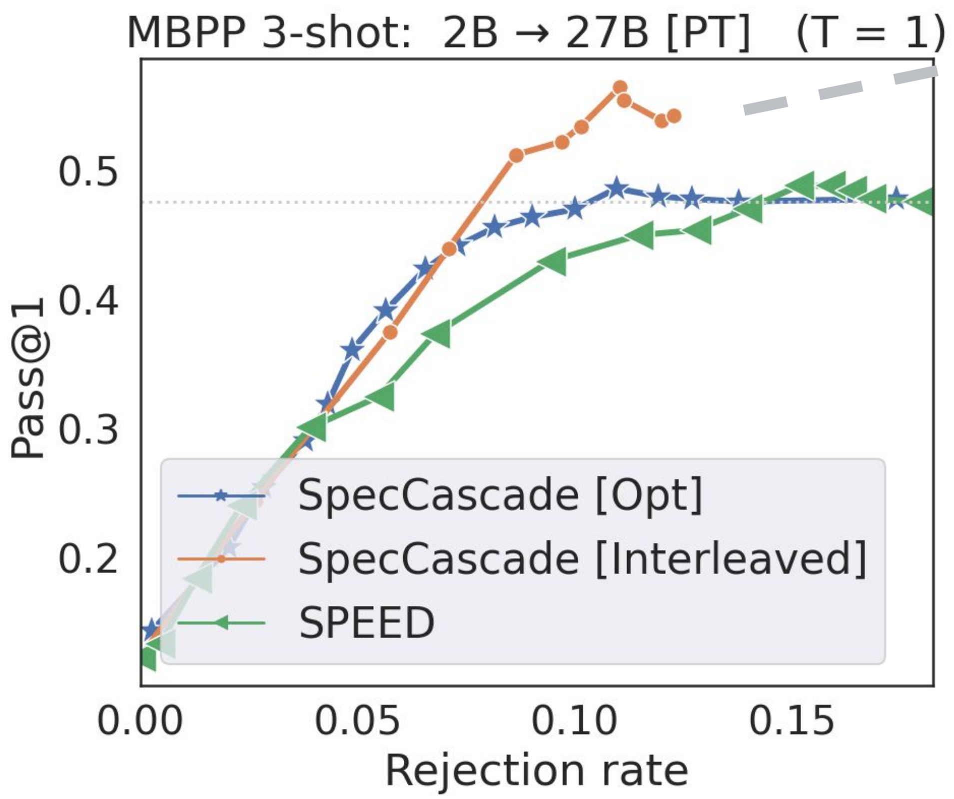

Compared to speculative decoding, **1.61x ⟶ 1.95x** speed-up

# Empirical results: few-shot Gemma models



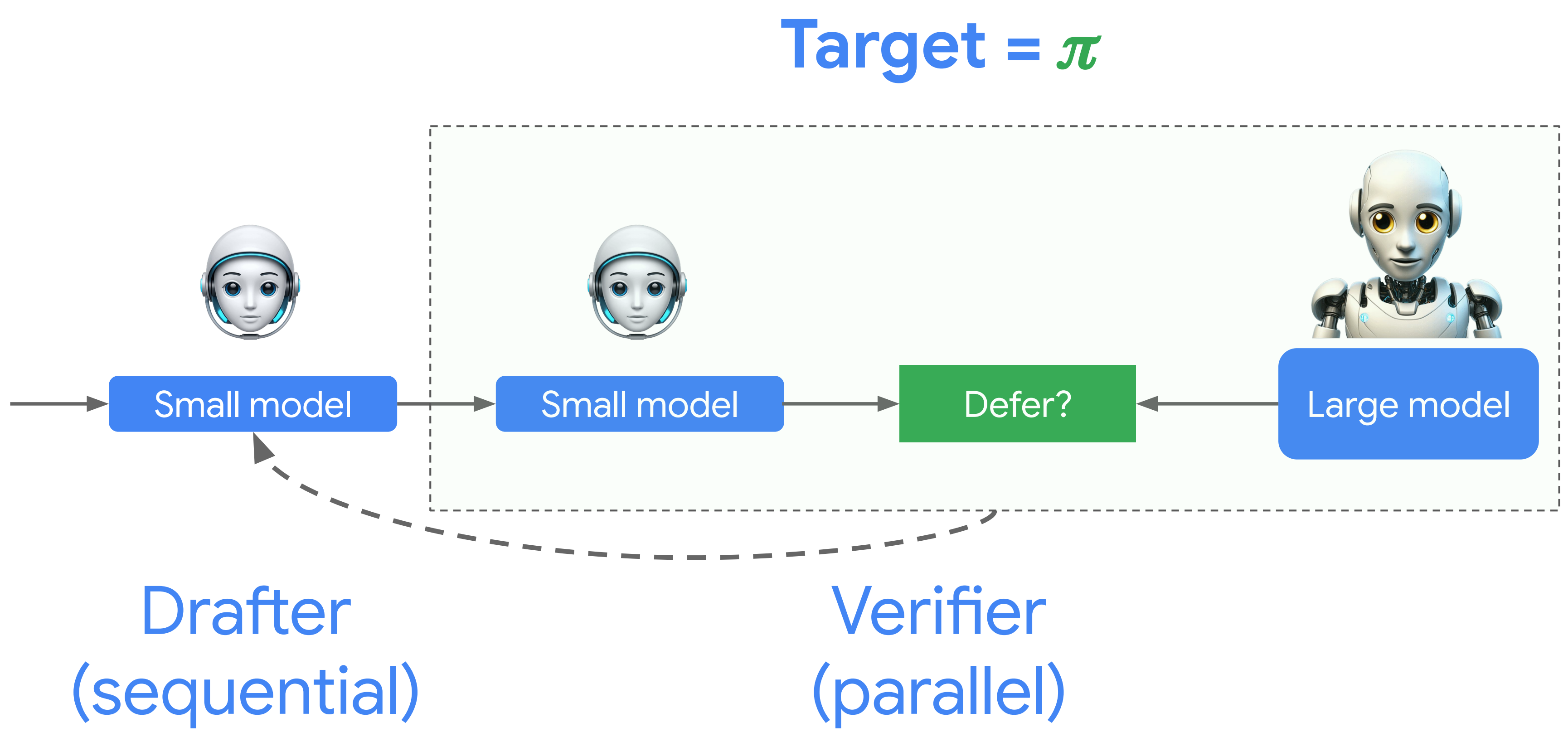




Can exceed large model's quality!

# Summary

- Speculative execution with a **modified target distribution** $\pi$
  - Mimic **data-generation** versus **large model** distribution
- Define $\pi$ via a **deferral rule**
  - Potentially exceed large model performance!
- More details in the paper!
  - Relation to approximate verification techniques
  - Token-specific deferral




**Guarantee**
Probability of sampling $v$ is $\pi(v)$

# Acknowledgements

Emojis from emojis.com

- AI Large
- AI Small

# Appendix

Google Research

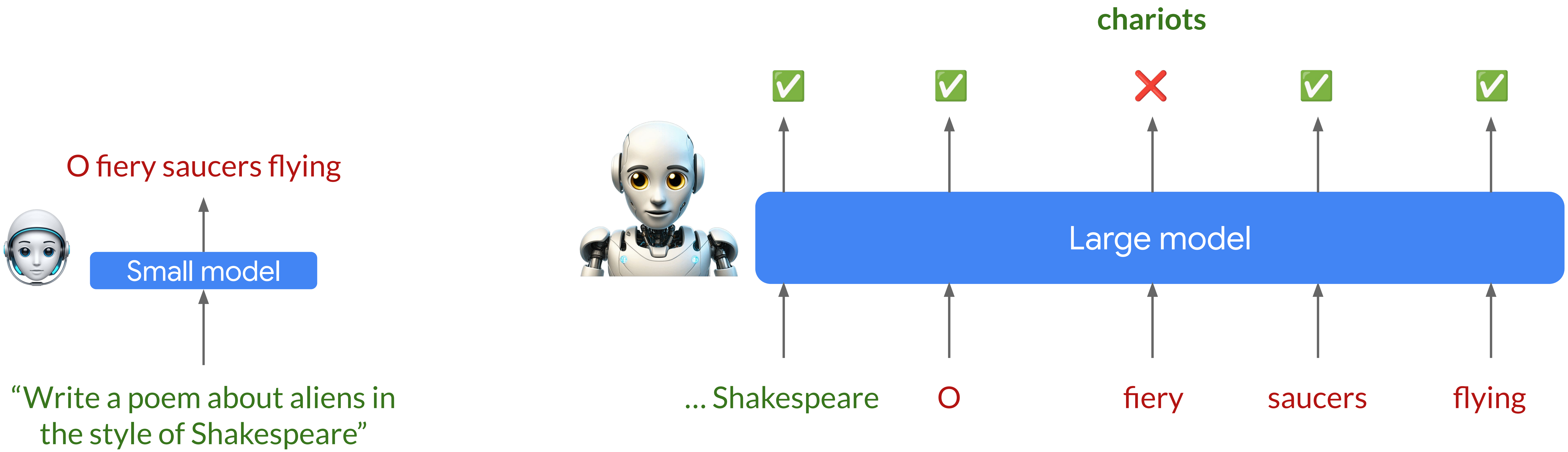
# Empirical results: fine-tuned T5 models

| | Latency↓ when matching large model's quality | | | | | | Best quality *without* exceeding large model's latency | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Small → Large | | | Small → XL | | | Small → Large | | | Small → XL | | |
| Method | WMT | XSum | CNNDM | WMT | XSum | CNNDM | WMT | XSum | CNNDM | WMT | XSum | CNNDM |
| SeqCascade [Chow] | 1.55× | 0.84× | 0.98× | 2.46× | 0.93× | 0.94× | 16.56 | 12.97 | 9.91 | 16.29 | 16.40 | 11.18 |
| TokenCascade [Chow] | 1.03× | 0.93× | 1.40× | 1.46× | 0.82× | 1.51× | 16.52 | 13.30 | 10.36 | 16.65 | 17.09 | 11.44 |
| SpecDecode [Lossy] | 1.61× | 1.10× | 1.57× | 2.17× | 1.28× | 2.07× | 17.26 | 13.90 | 10.43 | 16.94 | 17.36 | 11.53 |
| BiLD* | 1.34× | 1.04× | 1.38× | 1.85× | 1.28× | 1.84× | 16.49 | 13.81 | 10.14 | 15.90 | 17.35 | 11.35 |
| SpecCascade [Chow] | 1.43× | 1.04× | 1.41× | 2.01× | 1.28× | 1.97× | 17.76 | 13.82 | 10.28 | 16.35 | 17.36 | 11.39 |
| SpecCascade [Diff] | 1.79× | 1.17× | 1.75× | 2.44× | 1.30× | 2.15× | 18.04 | 14.00 | 10.64 | 18.07 | 17.37 | 11.67 |
| SpecCascade [OPT] | **1.95**× | 1.17× | 1.80× | **2.61**× | 1.34× | **2.21**× | 18.33 | 14.10 | 10.86 | 18.09 | 17.48 | 11.85 |
| SpecCascade [Token] | 1.85× | **1.18**× | **1.89**× | 2.50× | **1.40**× | 1.89× | **22.50** | **15.85** | **12.63** | **22.70** | **18.79** | **12.63** |

Better tradeoffs ⟶
gains in latency
*or* quality

# Speculative decoding

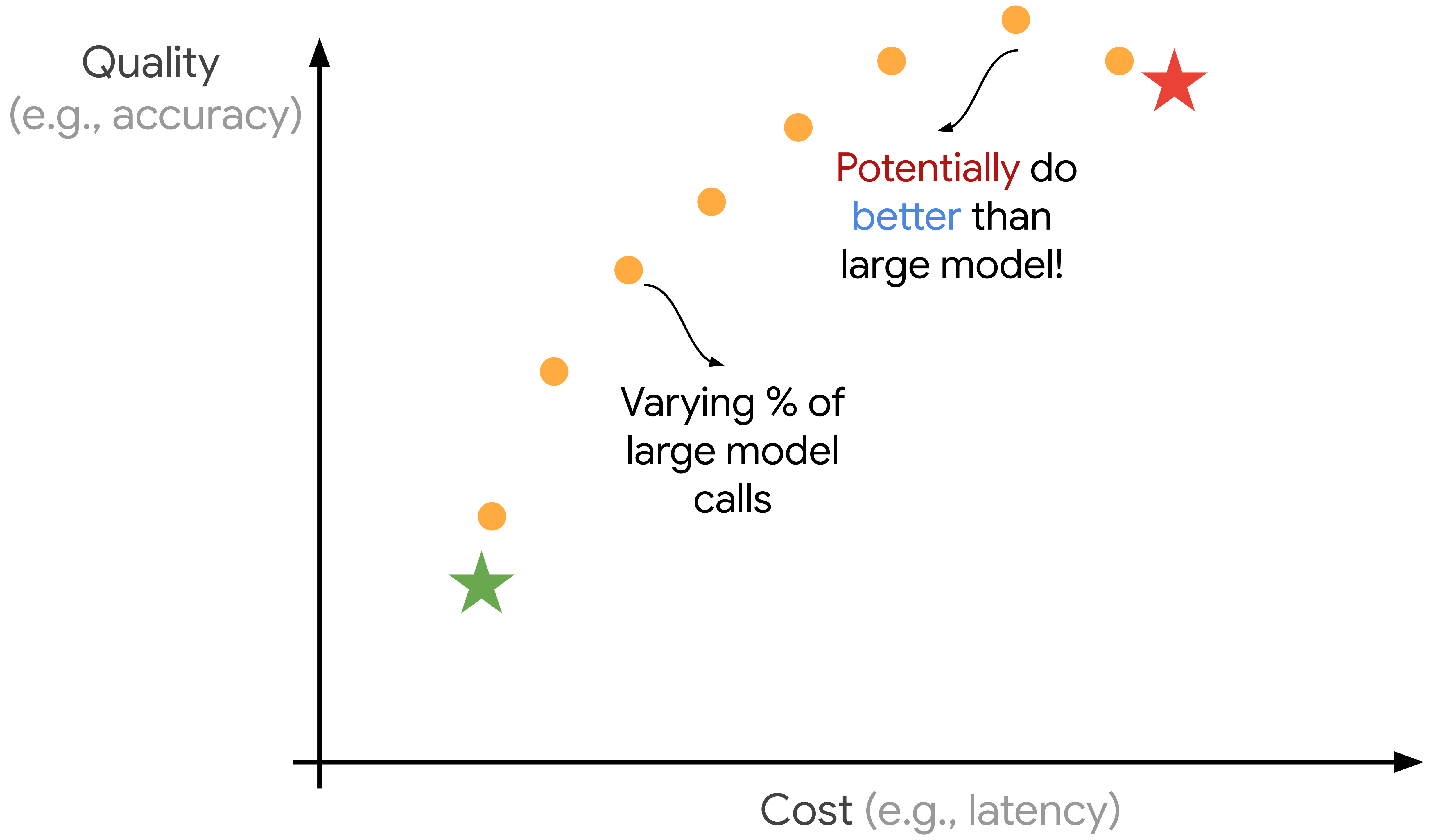
- **Draft** with small model; **verify** with large model




Leviathan et al. Fast Inference from Transformers via Speculative Decoding. ICML 2023.
Chen et al. Accelerating large language model decoding with speculative sampling. arXiV 2023.

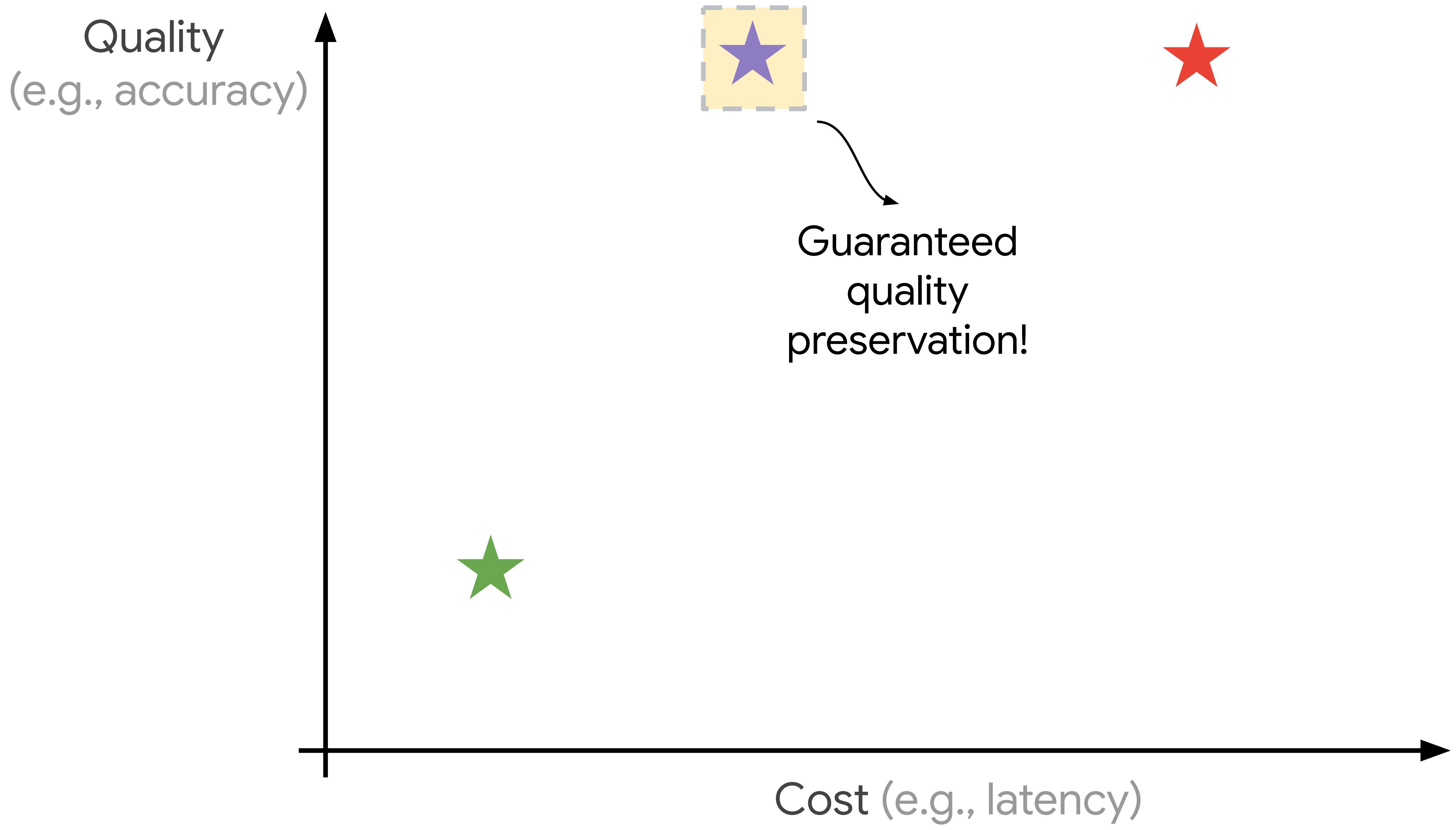
# Cascades: cost-quality tradeoff

- Try to use small model; if uncertain, use large model



# Speculative decoding: cost-quality tradeoff

- Draft with small model; (stochastically) verify with large model




Google

# Cascades versus speculative decoding

## Cascades

😎 Quality-enhancing speedup

(sometimes 🤞 )

💡 Mimic data-generating distribution


## Speculative Decoding

😎 Quality-preserving speedup

💡 Mimic verifier distribution


**Can we combine both approaches?**

Google

# Post-hoc deferral

- Let $h^{(1)}$, $h^{(2)}$ denote the small & large model
- For $x \in X$, let $r(x) \in \{0, 1\}$ denote whether or not to invoke $h^{(2)}$
- **Goal**: learn $r(x)$ achieving
  - minimal average loss $\ell$ of chosen model

$$\min_{r\colon \mathcal{X}\to\{0,1\}} \mathbb{E}_{(x,y)}\left[\overbrace{\mathbb{1}(r(x)=0)}^{r \text{ decides to call } h^{(1)}} \cdot \underbrace{\ell(y, h^{(1)}(x))}_{\text{Loss of small model}} + \overbrace{\mathbb{1}(r(x)=1)}^{r \text{ decides to call } h^{(2)}} \cdot \underbrace{\ell(y, h^{(2)}(x))}_{\text{Loss of large model}}\right] + c \cdot \overbrace{\mathbb{P}_x(r(x)=1)}^{\text{rate of calling } h^{(2)}}$$

Google

# (Bayes-)Optimal token deferral

- Suppose we have a context $x_{<t} = x_1, \ldots, x_{t-1}$
- The **Bayes-optimal** token deferral takes the form:

**Fact:** The Bayes-optimal token deferral rule $r^*$ is

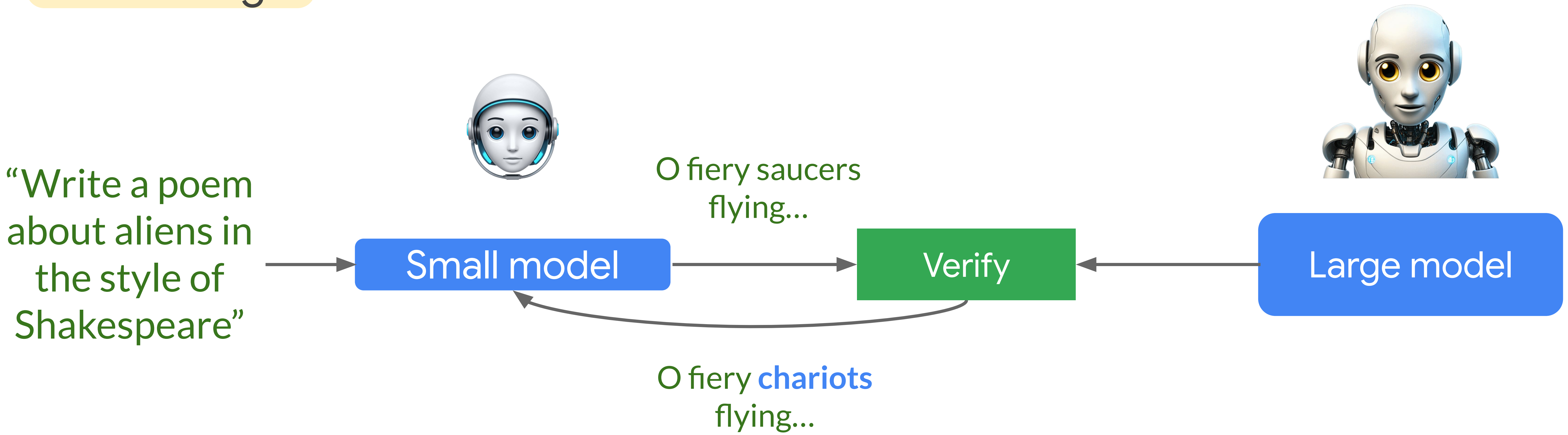
$$r^*(x_{<t}) = 1 \iff \underbrace{\mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\ell(v, p_{\text{Small}}(\cdot \mid x_{<t}))] - \mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\ell(v, p_{\text{Large}}(\cdot \mid x_{<t}))]}_{\text{Expected loss gap}} > c$$

Cost of invoking $p_{\text{Large}}$

# How to allow adaptive compute paths?

**Key idea:** use larger models sparingly, only for few (“hard”) cases

Speculative decoding




Leviathan et al. Fast inference from Transformers via speculative decoding. ICML 2023.

# Bayes-optimal deferral: approximation

- In practice, we cannot compute expectations under $\mathbb{P}$

- Depending on the choice of loss $\ell$, we can construct plug-in estimators:

$$\hat{r}(x_{<t}) = 1 \iff \max_{v \in \mathcal{V}} p_{\text{Large}}(v \mid x_{<t}) - \max_{v \in \mathcal{V}} p_{\text{Small}}(v \mid x_{<t}) > c \cdot D_{\text{TV}}(p_{\text{Large}}, p_{\text{Small}})$$

- Under temperature sampling, we may further condition on drafted tokens; e.g.,

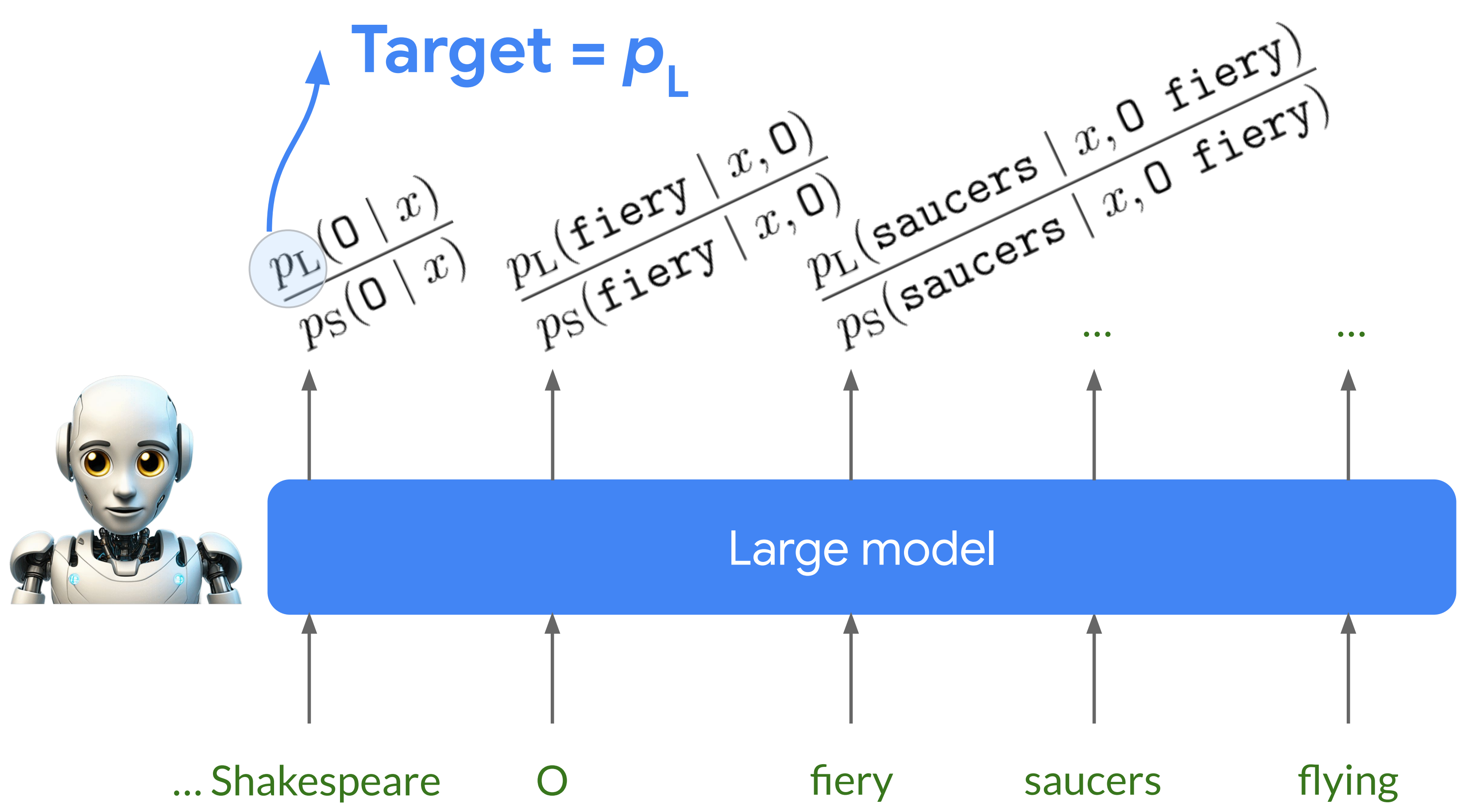
$$r^*(x_{<t}, v_{\text{Samp}}) = 1 \iff \max_{v \in \mathcal{V}} p_{\text{Large}}(v \mid x_{<t}) - p_{\text{Small}}(v_{\text{Samp}} \mid x_{<t}) > \alpha$$

# (Bayes-)Optimal cascade deferral

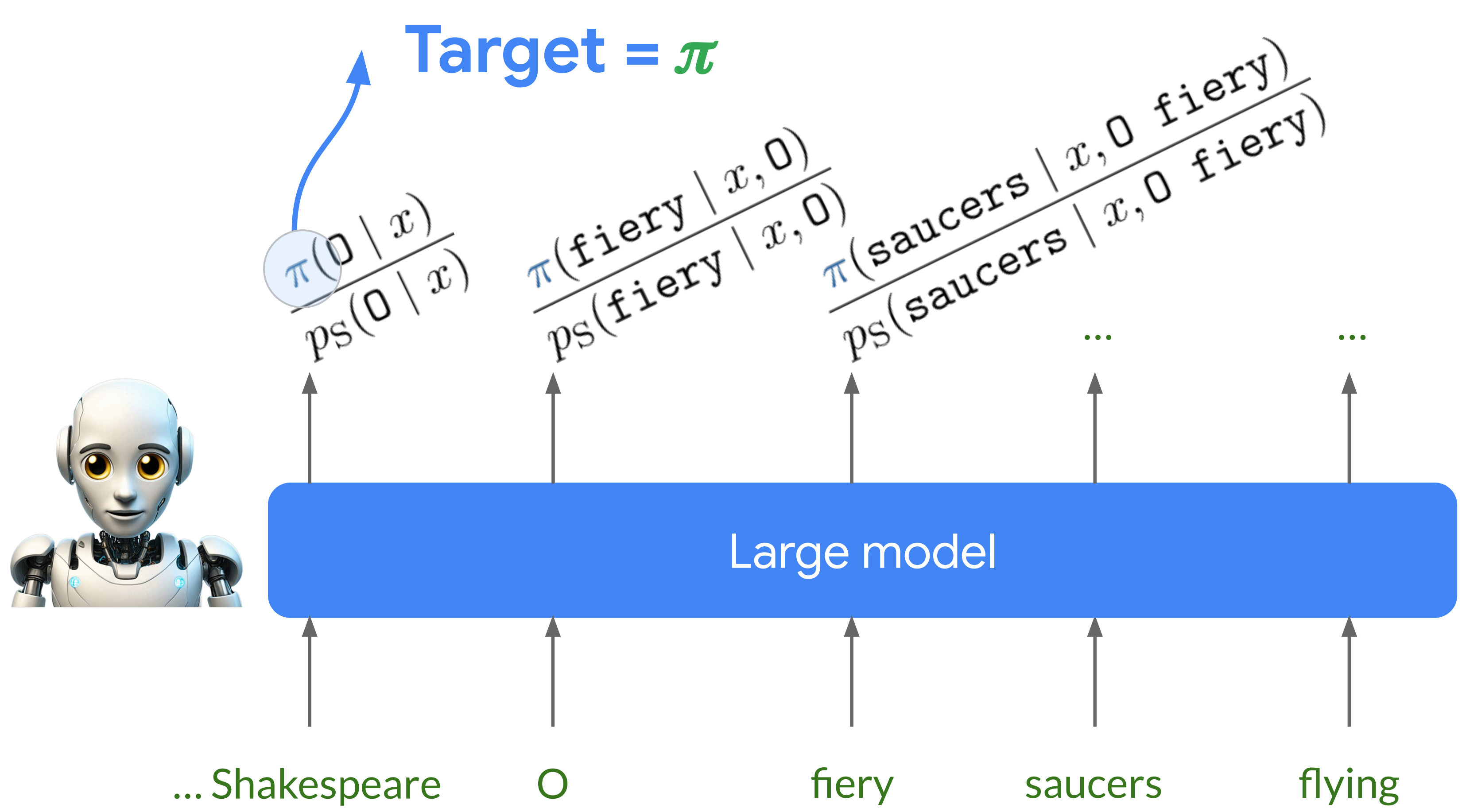
- Given any context $x_{<t} = x_1, \ldots, x_{t-1}$, we want a **deferral rule** $r(x_{<t}) \in \{0, 1\}$
  - $r(x_{<t}) = 1 \Leftrightarrow$ invoke large model

# From speculative decoding to speculative cascades

- Speculative execution using **alternate target distribution** for verification!




**Speculative Decoding**

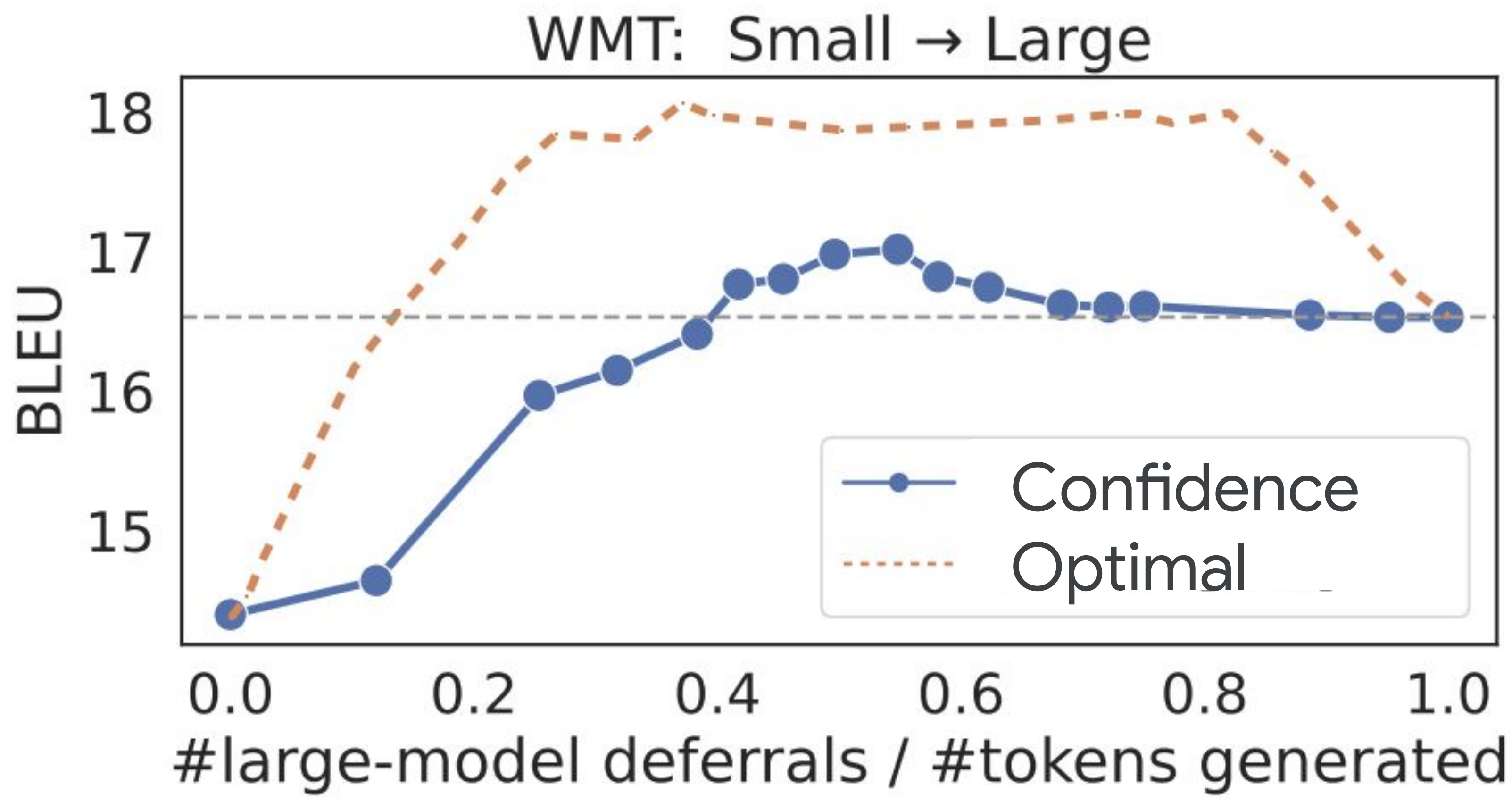


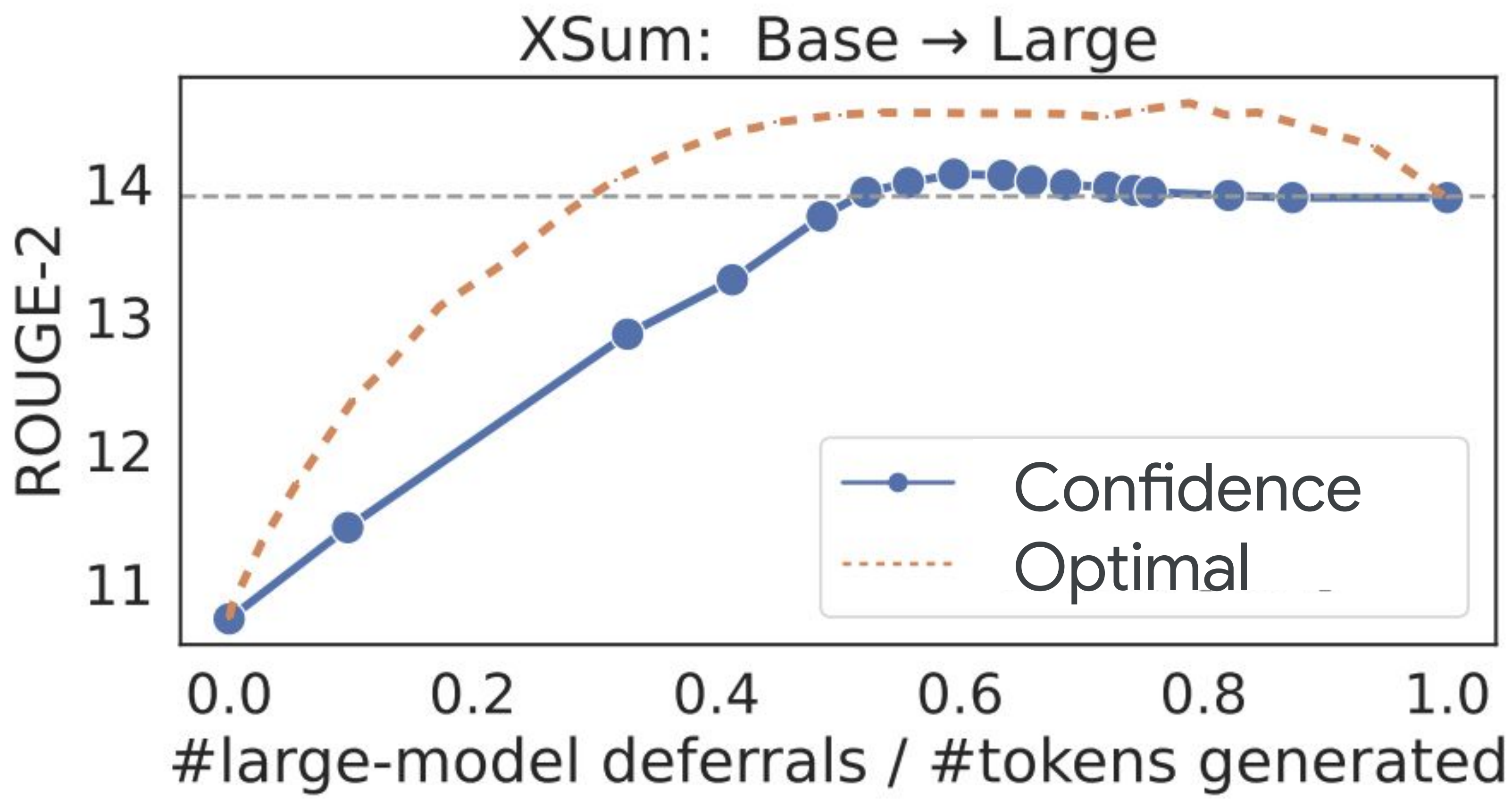

**Speculative Cascading (our proposal)**

Google

# (Bayes-)Optimal cascade deferral

- Optimal cascade deferral can outperform large model!



# Verification via deferral rules

- We choose the target distribution $\pi$ based on a **deferral rule** $r$:

$$\pi(\cdot) = (1 - r(x_{<t})) \cdot p_{\text{Small}}(\cdot) + r(x_{<t}) \cdot p_{\text{Large}}(\cdot)$$

**Fact:** The Bayes-optimal token deferral rule $r^*$ is

$$r^*(x_{<t}) = 1 \iff \mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\underbrace{\ell(v, p_{\text{Small}})] - \mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\ell(v, p_{\text{Large}}}_{\text{Expected loss gap}})] > c \cdot D_{\text{TV}}(p_{\text{Large}}, p_{\text{Small}})$$

Total probability gap

* The rule is Bayes-optimal for minimising the expected loss against the ground-truth token, subject to a bound on the rejection rate.

# Approximating optimal deferral

**Fact:** The Bayes-optimal token deferral rule $r^*$ is

$$r^*(x_{<t}) = 1 \iff \mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\ell(v, p_{\text{Small}})] - \mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\ell(v, p_{\text{Large}})] > c \cdot D_{\text{TV}}(p_{\text{Large}}, p_{\text{Small}})$$

- In practice, we cannot compute expectations under $\mathbb{P}$!
- Depending on the choice of loss $\ell$, we can construct plug-in estimators:

$$\hat{r}(x_{<t}) = 1 \iff \underbrace{\max_{v \in \mathcal{V}} p_{\text{Large}}(v \mid x_{<t}) - \max_{v \in \mathcal{V}} p_{\text{Small}}(v \mid x_{<t})}_{\text{Confidence gap}} > c \cdot \underbrace{D_{\text{TV}}(p_{\text{Large}}, p_{\text{Small}})}_{\text{Total probability gap}}$$

# Verification via deferral rules

- Speculative execution using **alternate target distribution** for verification!
- Target distribution $\pi$ is defined by a **deferral rule**:

$$\pi(\cdot) = (1 - r(x_{<t})) \cdot p_{\text{Small}}(\cdot) + r(x_{<t}) \cdot p_{\text{Large}}(\cdot)$$

**Fact:** The Bayes-optimal token deferral rule $r^*$ is

$$r^*(x_{<t}) = 1 \iff \mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\underbrace{\ell(v, p_{\text{Small}})] - \mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\ell(v, p_{\text{Large}})}_{\text{Expected loss gap}}] > c \cdot D_{\text{TV}}(p_{\text{Large}}, p_{\text{Small}})$$

Total probability gap

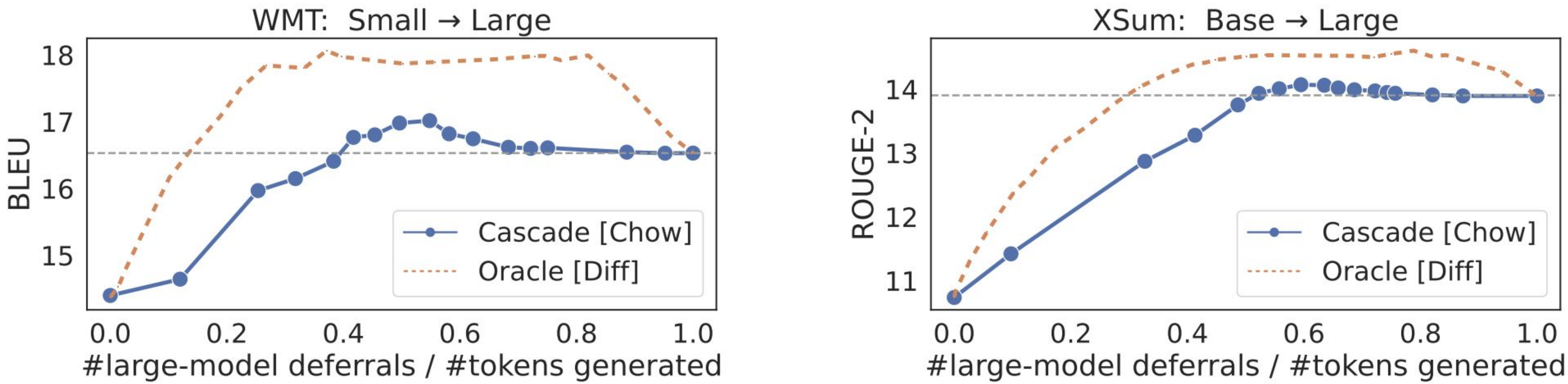
* The rule is Bayes-optimal for minimising the expected loss against the ground-truth token, subject to a bound on the rejection rate.

# (Bayes-)Optimal cascade deferral

- Given any context $x_{<t} = x_1, \ldots, x_{t-1}$, we want a **deferral rule** $r(x_{<t}) \in \{0, 1\}$
  - $r(x_{<t}) = 1 \Leftrightarrow$ invoke large model

Requires calling large model!

- How does the optimal rule perform? $r^*(x_{<t}) = 1 \iff \mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\ell(v, p_{\mathrm{Small}})] - \mathbb{E}_{v \sim \mathbb{P}(\cdot|x_{<t})}[\ell(v, p_{\mathrm{Large}})] > c$



# Speculative cascades: summary

**Speculative Decoding**

- Sample from small model:

$$v \sim p_{\text{small}}(\cdot)$$

- Should we accept the sampled token?

$$a \sim \min\left\{1,\ \frac{p_{\text{large}}(v)}{p_{\text{small}}(v)}\right\}$$

- If $a$ = 1, return $v$. Else, re-sample from:

$$v \sim \max\{0,\ p_{\text{large}}(\cdot) - p_{\text{small}}(\cdot)\}$$

**Guarantee**

Probability of sampling $v$ is $p_{\text{large}}(v)$

There are lossy variants that allow deviations from $p_{\text{large}}$ (Leviathan et al. '23, Tran-Thien '23)

**Speculative Cascading**

- Sample from small model:

$$v \sim p_{\text{small}}(\cdot)$$

- Should we accept the sampled token?

$$a \sim \min\left\{1,\ \frac{\boxed{\pi(v)}}{p_{\text{small}}(v)}\right\}$$

- If $a$ = 1, return $v$. Else, re-sample from:

$$v \sim \max\{0,\ \boxed{\pi(\cdot)} - p_{\text{small}}(\cdot)\}$$

**Guarantee**

Probability of sampling $v$ is $\boxed{\pi(v)}$

We allow $\pi$ to depend on $p_{\text{Large}}$, $p_{\text{Small}}$