Parameter-efficient methods for LLMs

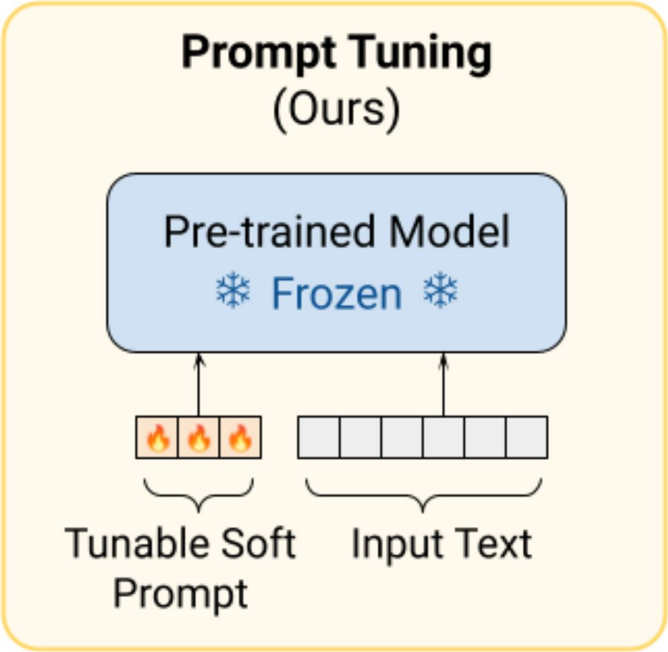
Jessica Hoffmann,
Google Deepmind

Agenda

1. **Agile classifiers:** safety text classifiers for all
2. **PE-RL:** from classifiers to reward models
3. **Application:** Hallucination detection and mitigation in Retrieval Augmented Generation

Agile classifiers

PEFT: prompt-tuning



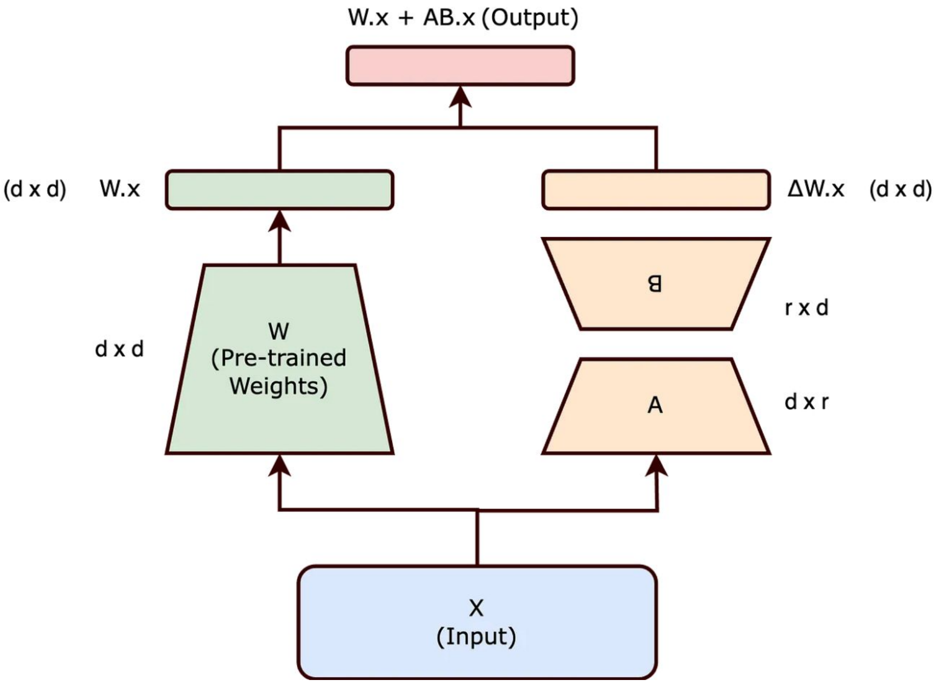
Model learns soft prompts

Attention to the soft prompt maps input to output

Soft prompt:
5 tokens/embeddings enough

[1] The Power of Scale for Parameter-Efficient Prompt Tuning. *Brian Lester, Rami Al-Rfou, Noah Constant.*

PEFT: LoRA



Model learns a **low-rank approximation of ΔW** = AB**.**

ΔW: d x d
A: d x r
B: r x d

r ∈ {1,16} in practice. **r = 4** very common.

Image credit: https://towardsdatascience.com/understanding-lora-low-rank-adaptation-for-finetuning-large-models-936bce1a07c6

[2] LoRA: Low-Rank Adaptation of Large Language Models. *Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen*

Agile classifiers

Model	Dialogue Safety					Neutral Responses		
	ParlAI Single Standard	ParlAI Single Adversarial	ParlAI Multi	BAD-2	BAD-4	Multiple Perspectives	Neutral	Well-Explained
PaLM 62B best few-shot	0.89	0.67	0.56	0.54	0.54	0.84	0.87	0.87
T5 XXL - 80	0.18	0.18	0.19	0.29	0.48	0.94	0.96	0.76
T5 XXL - 2,000	0.90	0.91	0.48	0.20	0.44	—	—	—
Human Agreement	—	—	—	—	—	**0.94**	0.95	**0.90**
Previous SOTA	0.88	0.67	0.66	—	—	—	—	—
PaLM 62B - 80	0.87	0.77	0.71	0.60	0.65	0.94	**0.96**	0.88
PaLM 62B - 2,000	**0.95**	**0.91**	**0.81**	**0.68**	**0.70**	—	—	—

Model	Unhealthy Comment Corpus						
	Antagonistic	Condescending	Dismissive	Generalization	Hostile	Sarcastic	Unhealthy
PaLM 62B best few-shot	0.79	0.78	0.81	0.76	0.79	0.76	0.70
T5 XXL - 80	0.50	0.55	0.56	0.49	0.57	0.54	0.51
T5 XXL - 2,000	0.74	0.74	0.75	0.80	0.80	0.74	0.66
Human Agreement	0.71	0.72	0.68	0.73	0.76	0.72	0.62
Previous SOTA	0.82	0.78	0.82	0.74	0.84	0.64	0.69
PaLM 62B - 80	0.80	0.80	0.74	0.81	0.84	0.81	0.63
PaLM 62B - 2,000	**0.86**	**0.84**	**0.87**	**0.90**	**0.89**	**0.85**	**0.77**

[3] Towards Agile Text Classifiers for Everyone. *Maximilian Mozes*, Jessica Hoffmann*, Katrin Tomanek, Muhamed Kouate, Nithum Thain, Ann Yuan, Tolga Bolukbasi, Lucas Dixon.*

Agile classifiers

Model	Dialogue Safety					Neutral Responses		
	PARLAI SINGLE STANDARD	PARLAI SINGLE ADVERSARIAL	PARLAI MULTI	BAD-2	BAD-4	Multiple Perspectives	Neutral	Well-Explained
PaLM 62B best few-shot	0.89	0.67	0.56	0.54	0.54	0.84	0.87	0.87
T5 XXL - 80	0.18	0.18	0.19	0.29	0.48	0.94	0.96	0.76
T5 XXL - 2,000	0.90	0.91	0.48	0.20	0.44	—	—	—
Human Agreement	—	—	—	—	—	**0.94**	0.95	**0.90**
Previous SOTA	0.88	0.67	0.66	—	—	—	—	—
PaLM 62B - 80	0.87	0.77	0.71	0.60	0.65	0.94	**0.96**	0.88
PaLM 62B - 2,000	**0.95**	**0.91**	**0.81**	**0.68**	**0.70**	—	—	—

Model	Unhealthy Comment Corpus						
	Antagonistic	Condescending	Dismissive	Generalization	Hostile	Sarcastic	Unhealthy
PaLM 62B best few-shot	0.79	0.78	0.81	0.76	0.79	0.76	0.70
T5 XXL - 80	0.50	0.55	0.56	0.49	0.57	0.54	0.51
T5 XXL - 2,000	0.74	0.74	0.75	0.80	0.80	0.74	0.66
Human Agreement	0.71	0.72	0.68	0.73	0.76	0.72	0.62
Previous SOTA	0.82	0.78	0.82	0.74	0.84	0.64	0.69
PaLM 62B - 80	0.80	0.80	0.74	0.81	0.84	0.81	0.63
PaLM 62B - 2,000	**0.86**	**0.84**	**0.87**	**0.90**	**0.89**	**0.85**	**0.77**

[3] Towards Agile Text Classifiers for Everyone. *Maximilian Mozes*, Jessica Hoffmann*, Katrin Tomanek, Muhamed Kouate, Nithum Thain, Ann Yuan, Tolga Bolukbasi, Lucas Dixon*

Agile classifiers: main result

SOTA classifiers w/ ~80 training examples

[3] Towards Agile Text Classifiers for Everyone. *Maximilian Mozes*, Jessica Hoffmann*, Katrin Tomanek, Muhamed Kouate, Nithum Thain, Ann Yuan, Tolga Bolukbasi, Lucas Dixon*

Agile classifiers: main result

SOTA classifiers w/ ~80 training examples

→ Anyone can make a safety classifier adapted to their need in a few hours

[3] Towards Agile Text Classifiers for Everyone. *Maximilian Mozes*, Jessica Hoffmann*, Katrin Tomanek, Muhamed Kouate, Nithum Thain, Ann Yuan, Tolga Bolukbasi, Lucas Dixon*

Agile classifiers: main result

SOTA classifiers w/ ~80 training examples

→ Anyone can make a safety classifier adapted to their need in a few hours
→ See the RAI toolkit of Gemma release (open source)

[3] Towards Agile Text Classifiers for Everyone. *Maximilian Mozes*, Jessica Hoffmann*, Katrin Tomanek, Muhamed Kouate, Nithum Thain, Ann Yuan, Tolga Bolukbasi, Lucas Dixon*

PE-RL: Parameter-Efficient RLHF

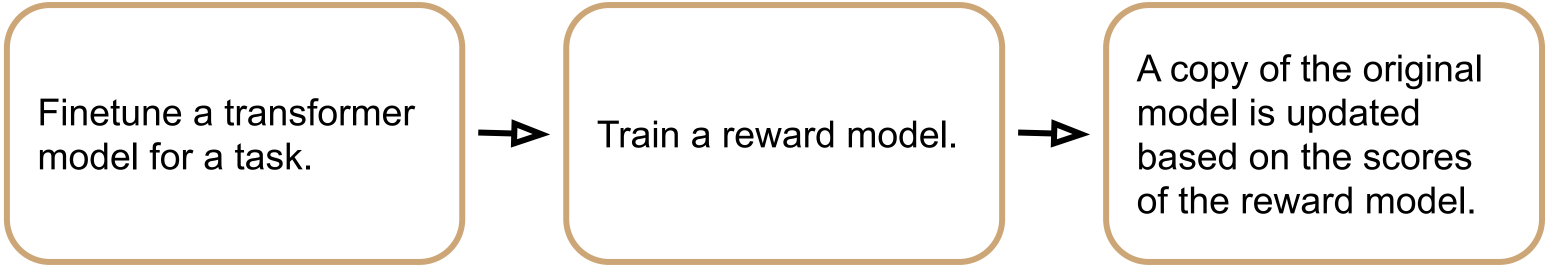
RLHF: Reinforcement Learning from Human Feedback

- ChatGPT uses it

- For those familiar w/ Reinforcement Learning:
  - Transformers predict answers word by word, which is equivalent to moving from state to state
  - Full answer are trajectories
  - Quality of answer is reward
  - Best way to answer a query is best policy

- Gather ~10k pairwise comparisons between answers, use it to train a reward model, which then yield the best policy for answering.

RLHF


Finetune a transformer model for a task.
Train a reward model.
A copy of the original model is updated based on the scores of the reward model.

~~RLHF~~ PE-RL (Parameter-Efficient RLHF)

LoRA-tune
~~Finetune~~ a transformer model for a task.

→

LoRA
Train a reward model.

→

The few parameters of LoRA
~~A copy of the original model is~~ updated based on the scores of the reward model.

~~RLHF~~ PE-RL (Parameter-Efficient RLHF)

LoRA-tune
~~Finetune~~ a transformer model for a task.

→

LoRA
Train a reward model.

→

The few parameters of LoRA
~~A copy of the original model is~~ updated based on the scores of the reward model.

→ Results comparable to RLHF despite 1000x reduction in parameters

→ Conjecture: more robust to parameters picking, more sample-efficient

[4] Parameter Efficient Reinforcement Learning from Human Feedback. *Hakim Sidahmed, Samrat Phatale, Alex Hutcheson, Zhuonan Lin, Zhang Chen, Zac Yu, Jarvis Jin, Simral Chaudhary, Roman Komarytsia, Christiane Ahlheim, Yonghao Zhu, Bowen Li, Saravanan Ganesh, Bill Byrne, Jessica Hoffmann, Hassan Mansoor, Wei Li, Abhinav Rastogi, Lucas Dixon*

Application: hallucinations mitigation

Application: hallucination reduction

The New York Times

Here's What Happens When Your Lawyer Uses ChatGPT

A lawyer representing a man who sued an airline relied on artificial intelligence to help prepare a court filing. It did not go well.

[...]

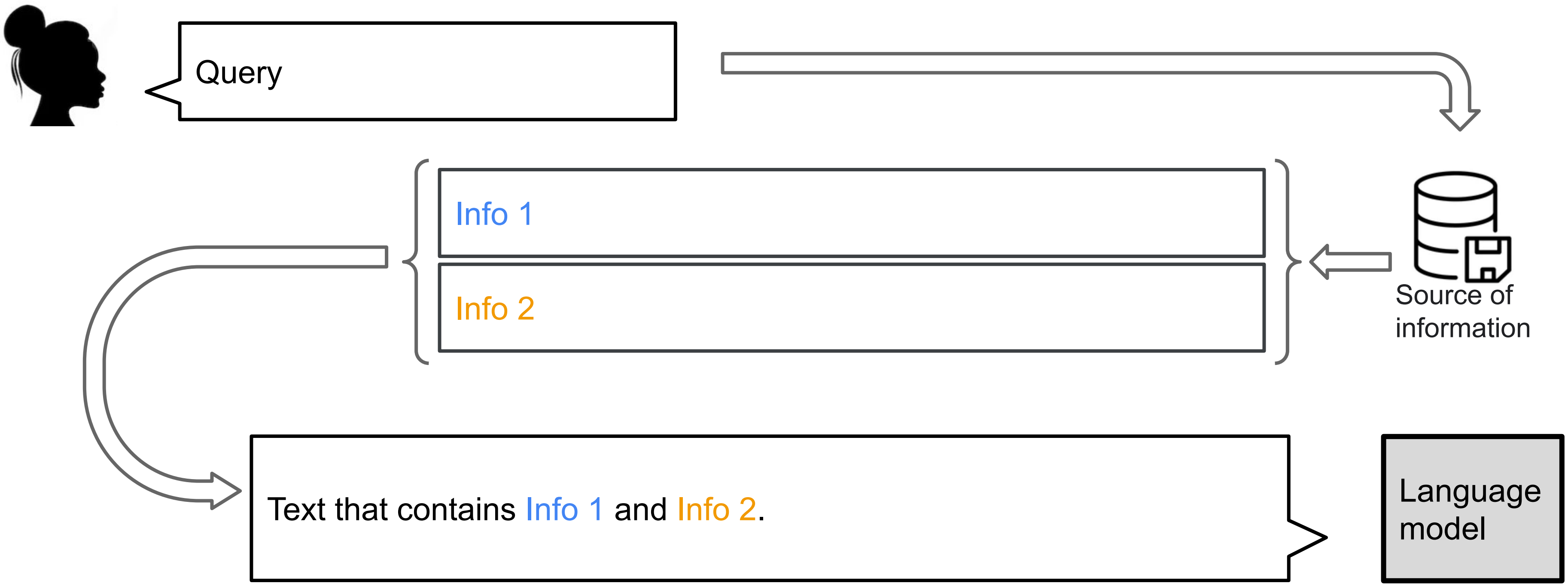
There was just one hitch: No one — not the airline's lawyers, not even the judge himself — could find the decisions or the quotations cited and summarized in the brief.

That was because ChatGPT had invented everything.

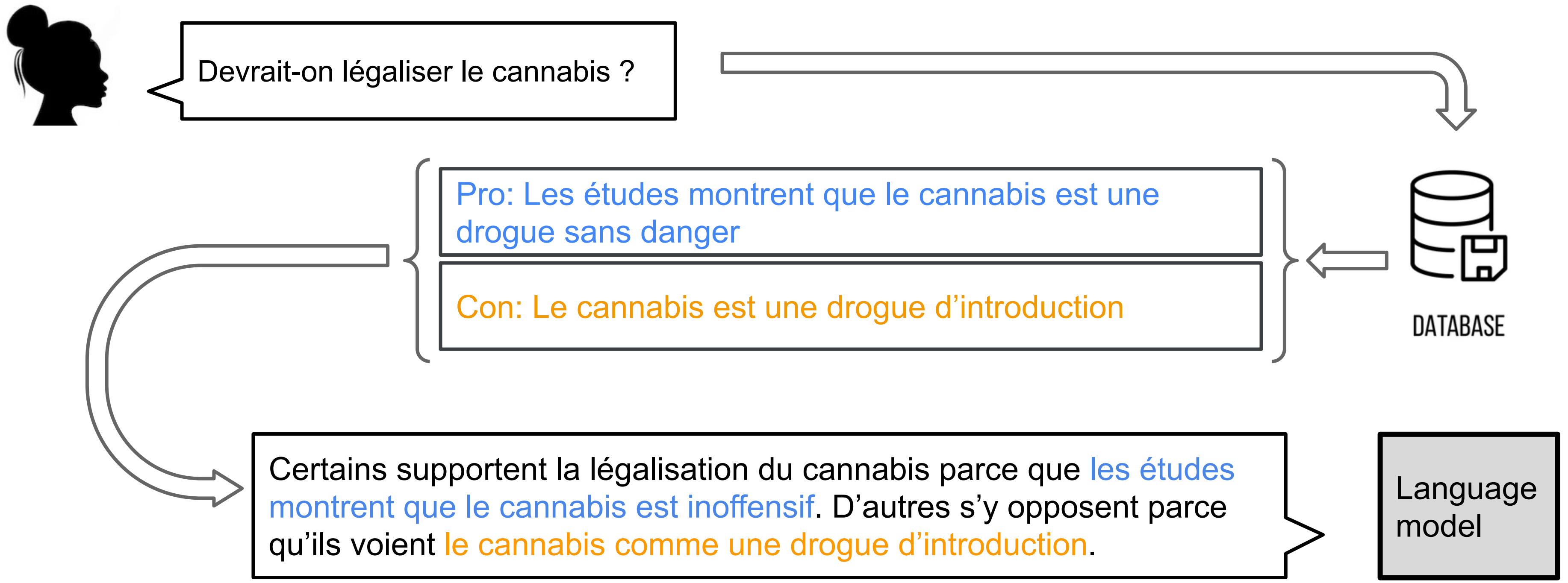
Application: hallucination reduction

- Hallucination: when LLMs don’t behave like we expected: false information, not on topic, rambling, toxic…
- In general, hard to define
- In Retrieval Augmented Generation (RAG), well-defined
- This section focuses on hallucinations in RAG

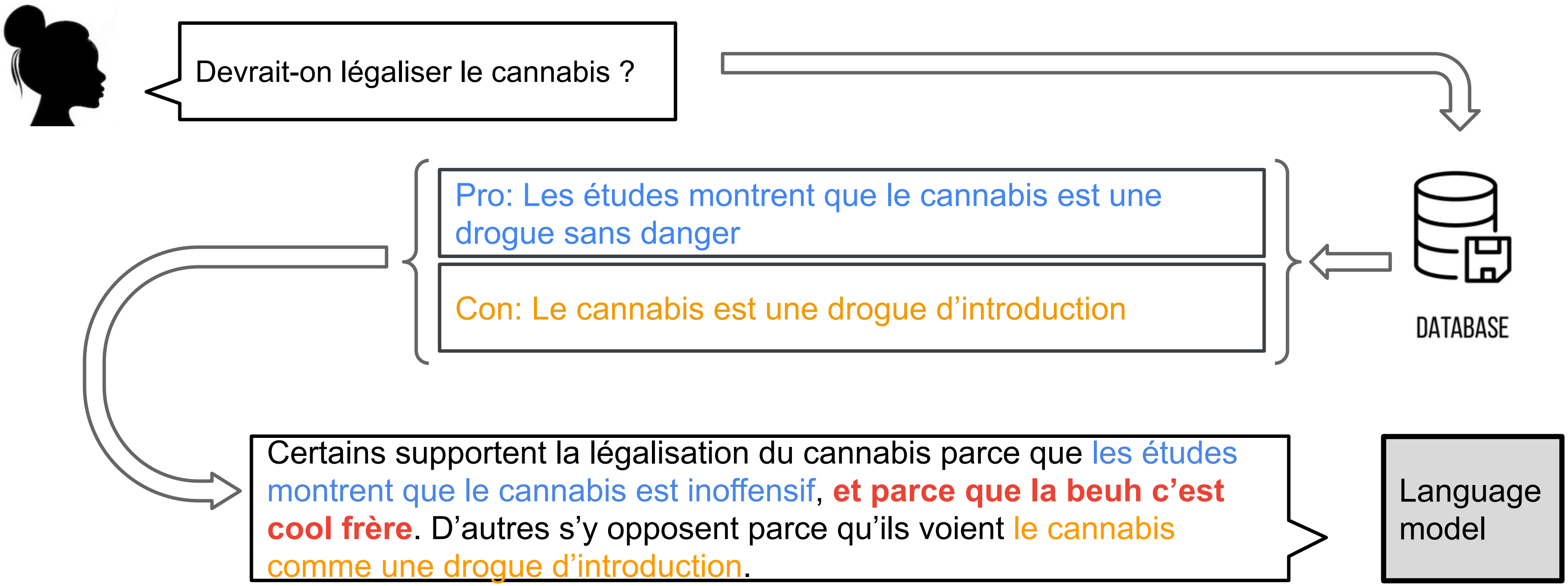
RAG: Retrieval Augmented Generation



RAG: Neutral Point of View generation



RAG: Neutral Point of View generation avec hallucination



Synthetic data generation

- No data + generating text until hallucination takes very long (~1 out 7 generations has hallucination)

- Synthetic data!
  - Make a few verified pairs (list of arguments, generation)
  - Remove 1 or 2 arguments, keep the generation
  - Synthetic hallucination!

- ~700 examples. Not enough for finetuning. Does not scale.

Synthetic data generation

- No data + generating text until hallucination takes very long (~1 out 7 generations has hallucination)

- Synthetic data!
  - Make a few verified pairs (list of arguments, generation)
  - Remove 1 or 2 arguments, keep the generation
  - Synthetic hallucination!

- ~700 examples. Not enough for finetuning. Does not scale.

→ Agile classifier

Agile classifier

Detecting hallucinations:

- Trained on **synthetic** hallucinations, evaluated on “**organic**” hallucinations
- **0.95 AUC**!
- (as baseline, 90% annotator agreement)

[5] Detecting Hallucination and Coverage Errors in Retrieval Augmented Generation for Controversial Topics. *Tyler A. Chang, Katrin Tomanek, Jessica Hoffmann, Nithum Thain, Erin van Liemt, Kathleen Meier-Hellstern, Lucas Dixon*

PE-RL

- Adapt classifier to reward model through PE-RL
- **3x** reduction in hallucinations, **15% → 5%**!

[6] Decoding-time Realignment of Language Models. *Tianlin Liu, Shangmin Guo, Leonardo Bianco, Daniele Calandriello, Quentin Berthet, Felipe Llinares, Jessica Hoffmann, Lucas Dixon, Michal Valko, Mathieu Blondel*

Thank you!