Tabi: An Efficient Multi-Level Inference System for Large Language Models

Yiding Wang, Kai Chen (HKUST), Haisheng Tan (USTC), Kun Guo

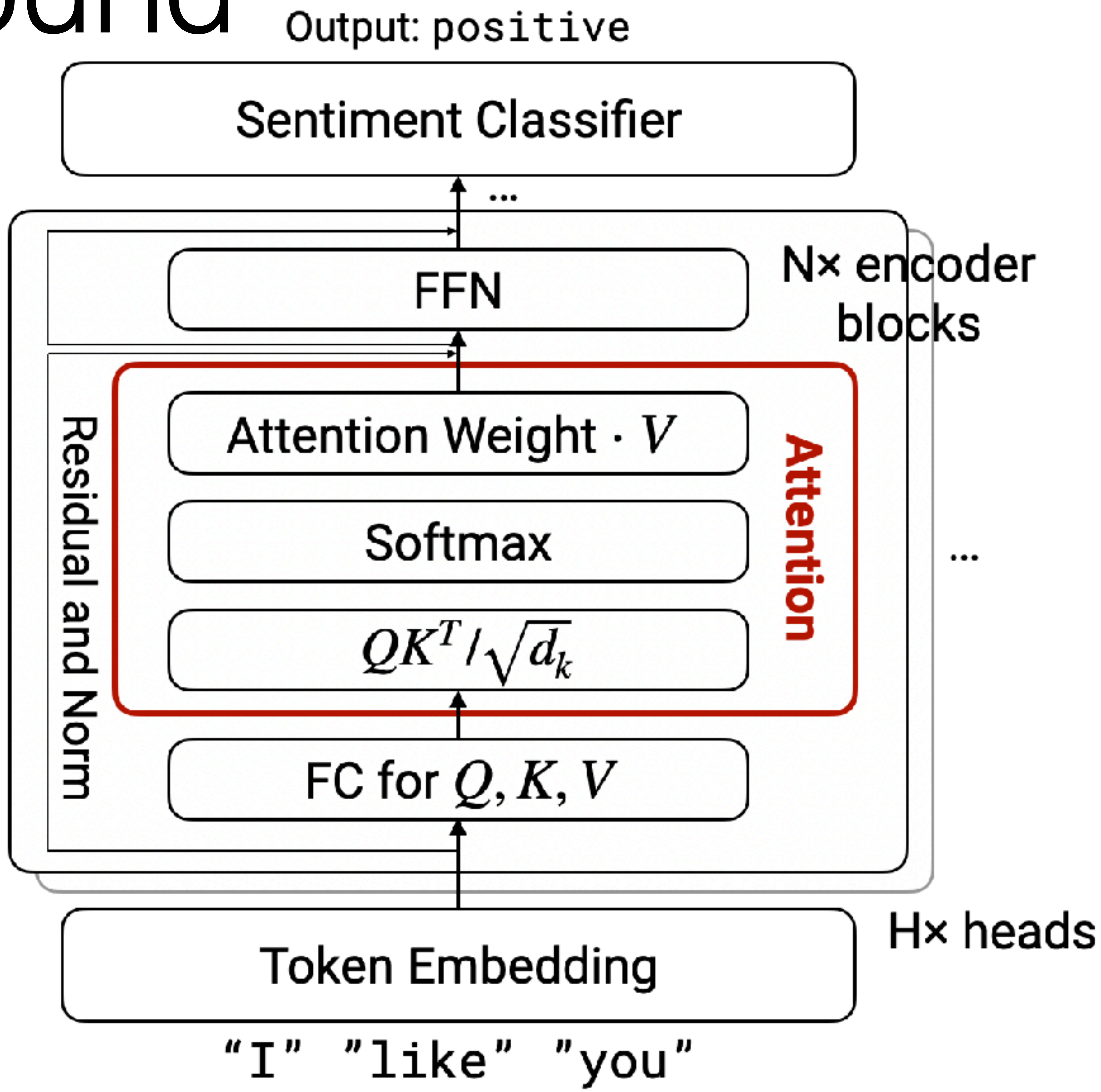
EuroSys 2023

Background

- Scope: Transformer-based **discriminative** models, rather than **generative** models.

- Classification and regression tasks, e.g., sentiment analysis.

- **Popular**: 25 out of the 30 most downloaded models on Huggingface are BERT-like encoder-only models.



THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

BERT-like and GPT-like Models

- BERT-like models consist of Transformer encoders.
- Input: text → encoding representation → predictions
- Work similar to traditional DNNs like CNN for image classification.

	BERT-like	**GPT-like**
Structure	Encoder-only	Decoder-only
Task	Prediction	Generation
Output	All at once	Token by token
#Params	300 million - 1 billion	1.5 - 175 billions

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Language Models Scale-Up Fast

- For a few % of SOTA accuracy, they are adding a lot of parameters and latency.
- Example: from DistilBERT to RoBERTa-Large, **7% acc.**✅, **4× latency**❌, **5× #params**❌.
- The accuracy return of adding parameters is diminishing.

Model	#Parameters (million)	Latency (ms)	Accuracy (%)	*Pf*
BERT-small	28	6	72.1	6
DistilBERT* [59]	66	7	83.2	5
ALBERT* [43]	11	14	84.5	5
PruneBERT* [60]	110	15	81.2	4
DeBERTa-small	142	12	86.9	4
BERT-base [18]	110	17	84.1	4
RoBERTa-base [47]	124	19	86.3	4
DeBERTa-base [35]	184	20	88.8	3
BERT-large	340	24	86.7	2
RoBERTa-large	355	26	90.6	2
DeBERTa-large	406	29	91.3	2
DeBERTa-xlarge	886	38	91.7	1

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

How Current Inference Systems Work

- Model-less: The system selects the model to serve a task (rather than by hand).
- Key module: **Model selection**. Because the real cost is running the selected model.
- Idea: **One best config for all queries of a task workload**.
- Cocktail (NSDI '22) works similarly: ensemble vs. single.

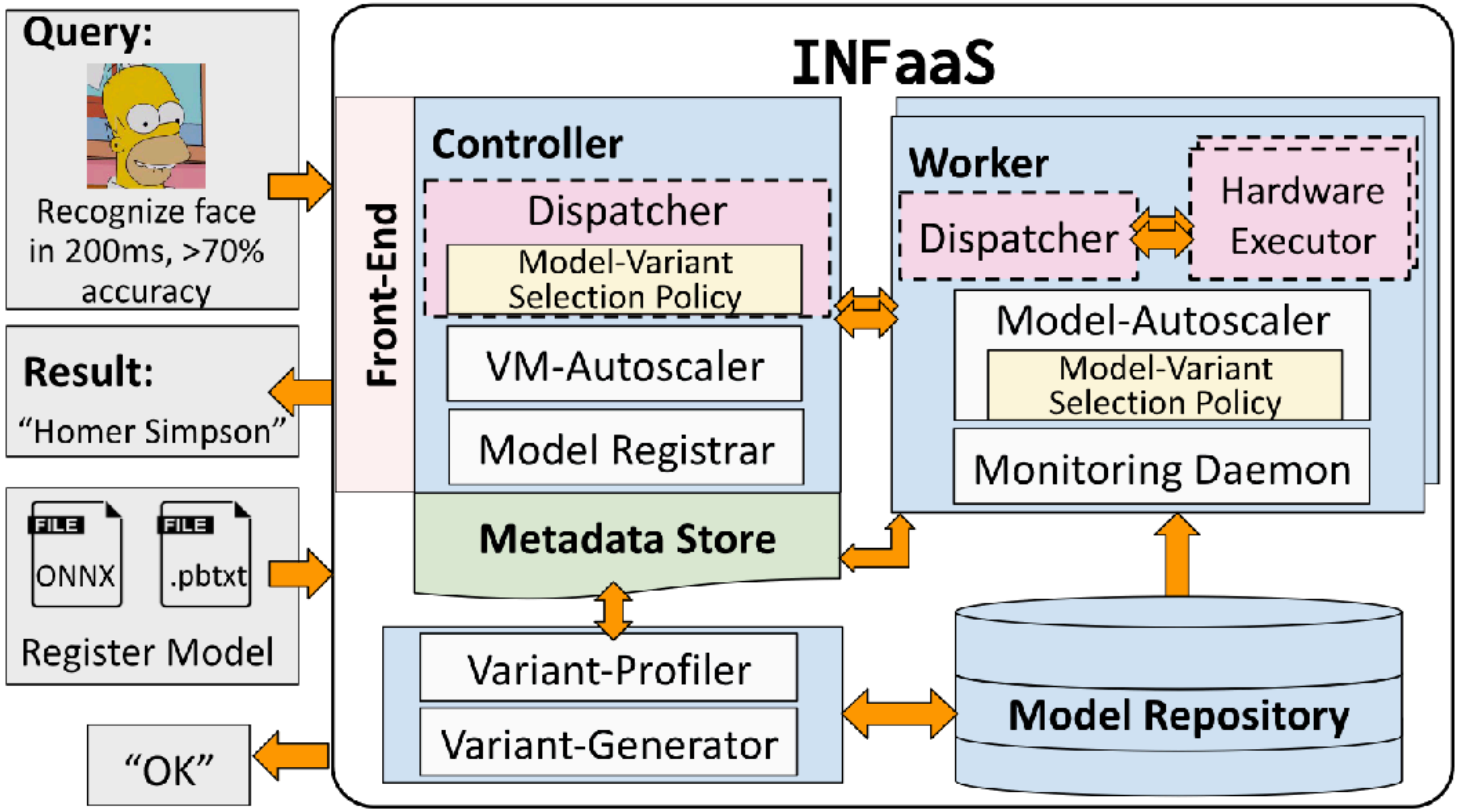
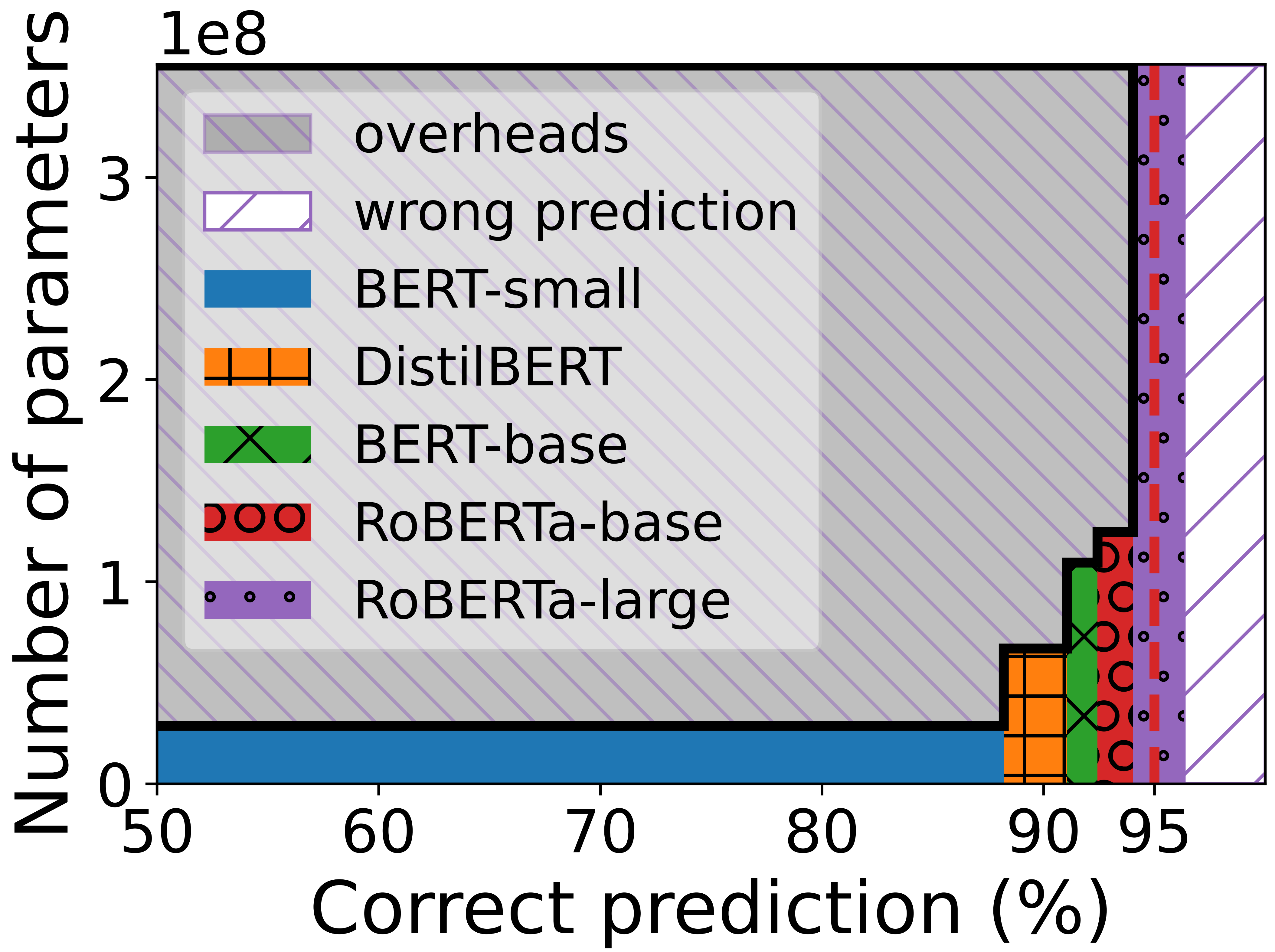


Image by the courtesy of INFaaS (Romero etc., ATC '21).

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

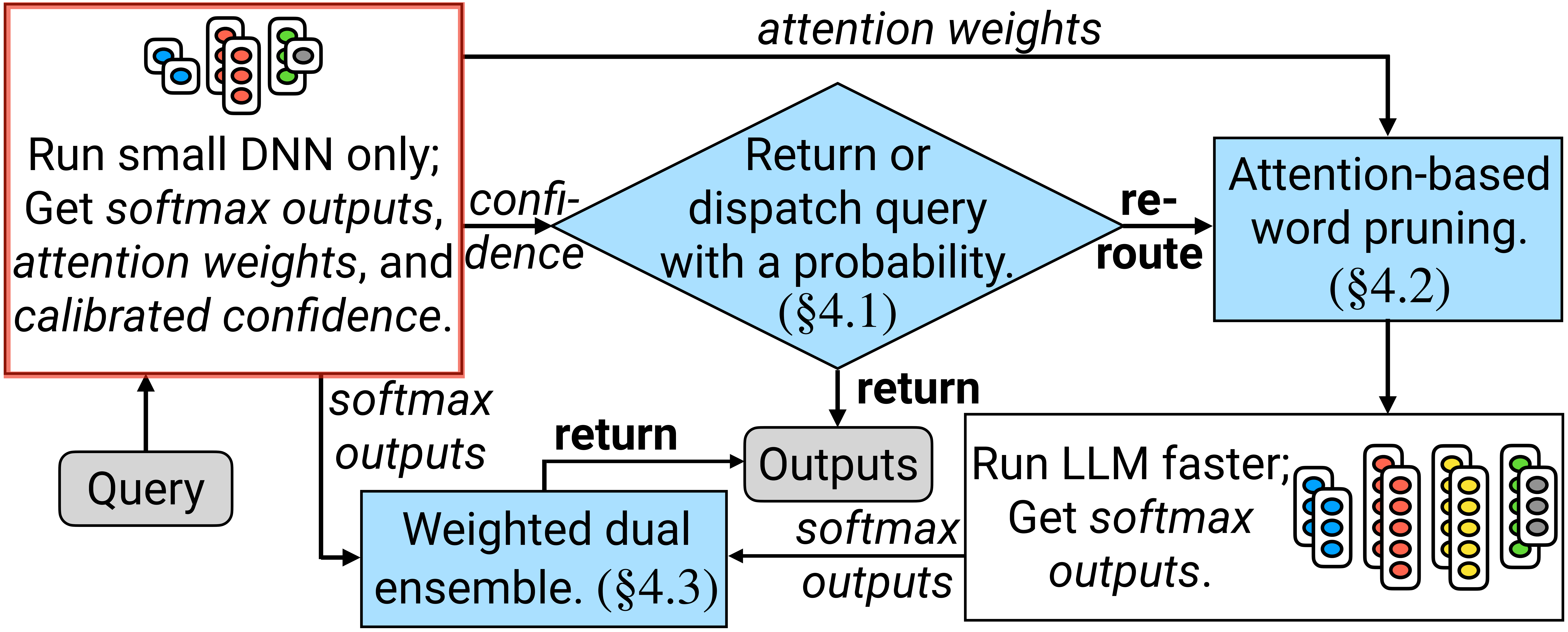
Overheads of Inference Systems

- Model-less inference systems select models at the application level: **One model for all.**
- Observation: A natural dataset is a mixture of simple and difficult queries.
- **Resource overheads for LMs:** A much smaller model with slightly lower accuracy won't get selected.
- Only select LLMs for demanding tasks.



THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

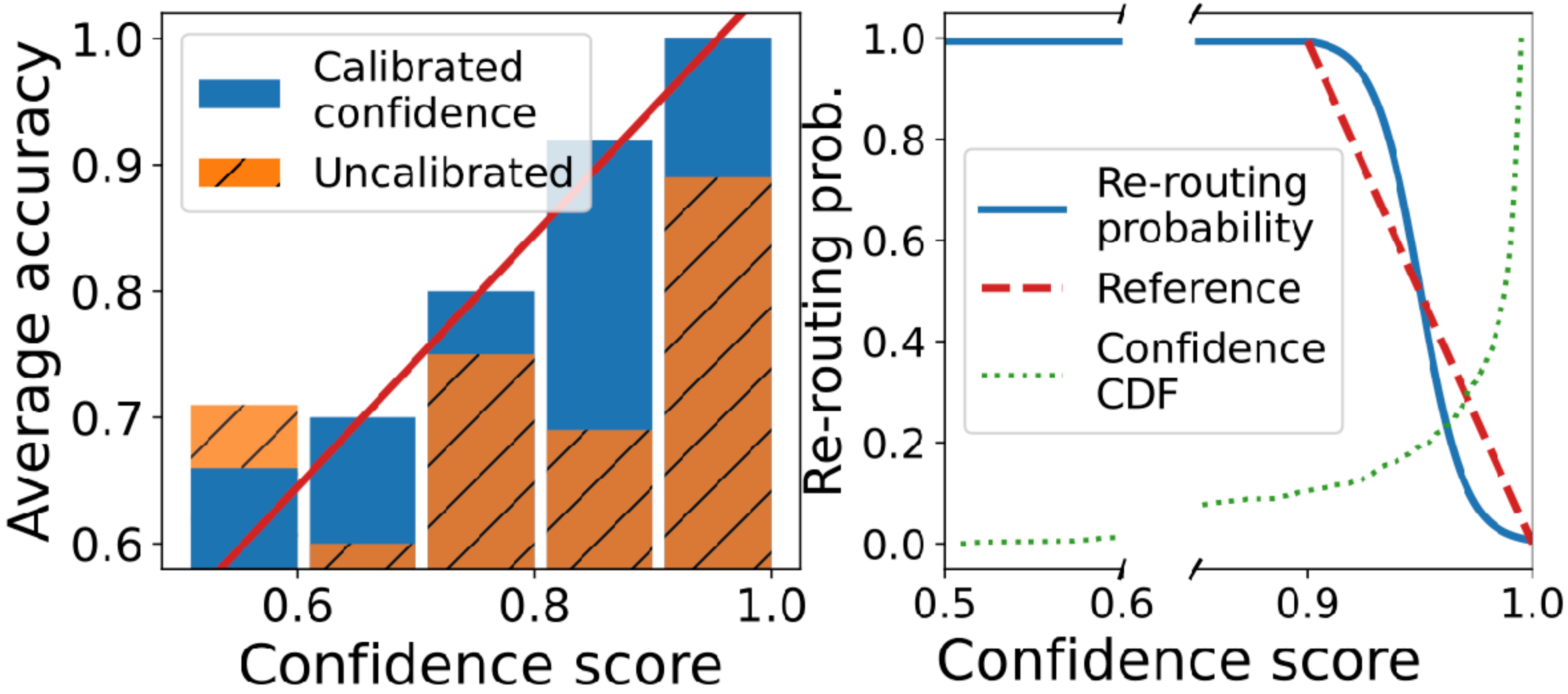
Design of Tabi



THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Confidence-Based Early Return

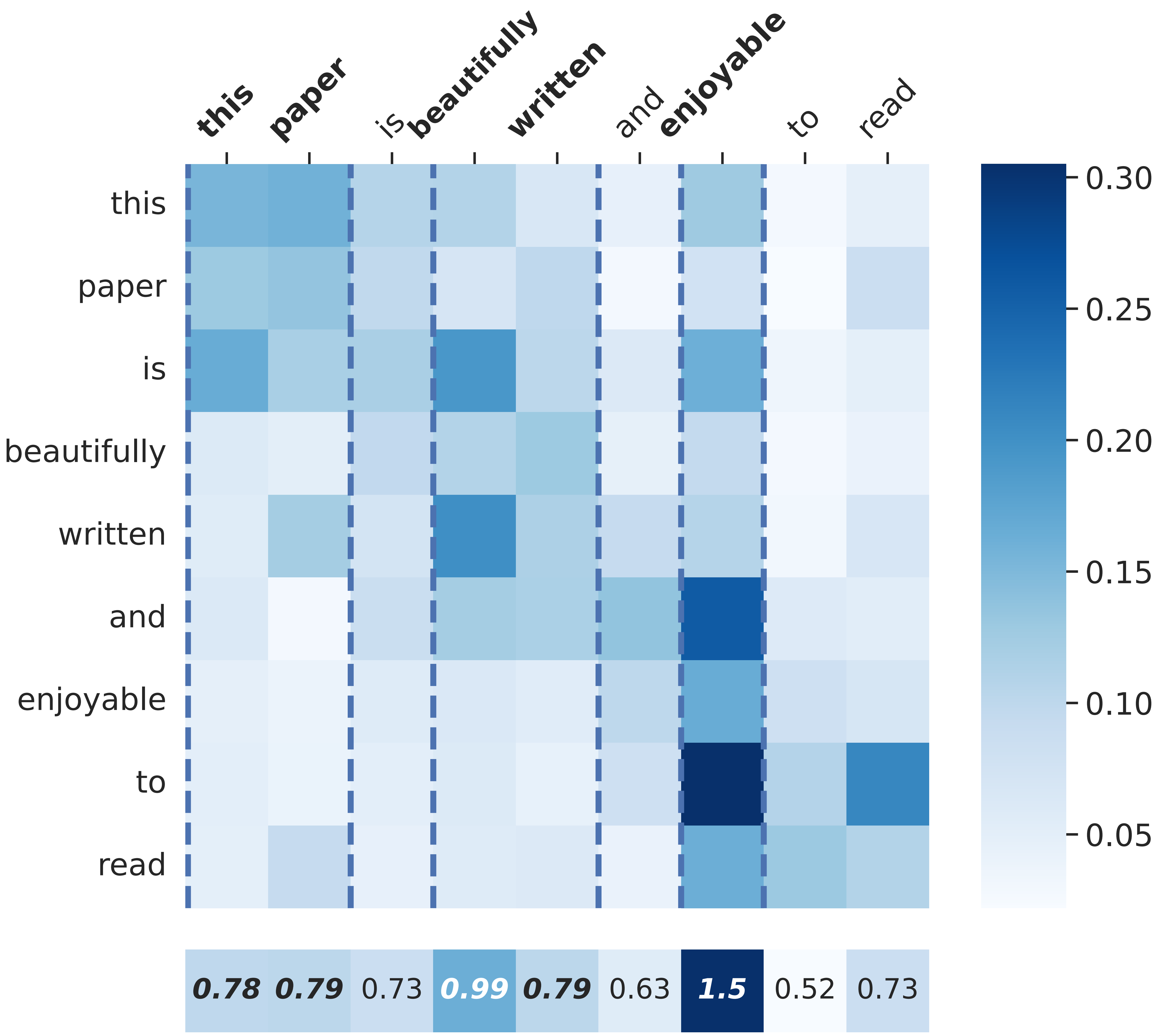
- **Calibrated confidence** (Temperature scaling)
- 50%-70% queries do not even invoke LLM.
- Same overall accuracy.
- Reduce the average latency by up to 40%.
- Tail latency?



THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Attention-Based Word Pruning

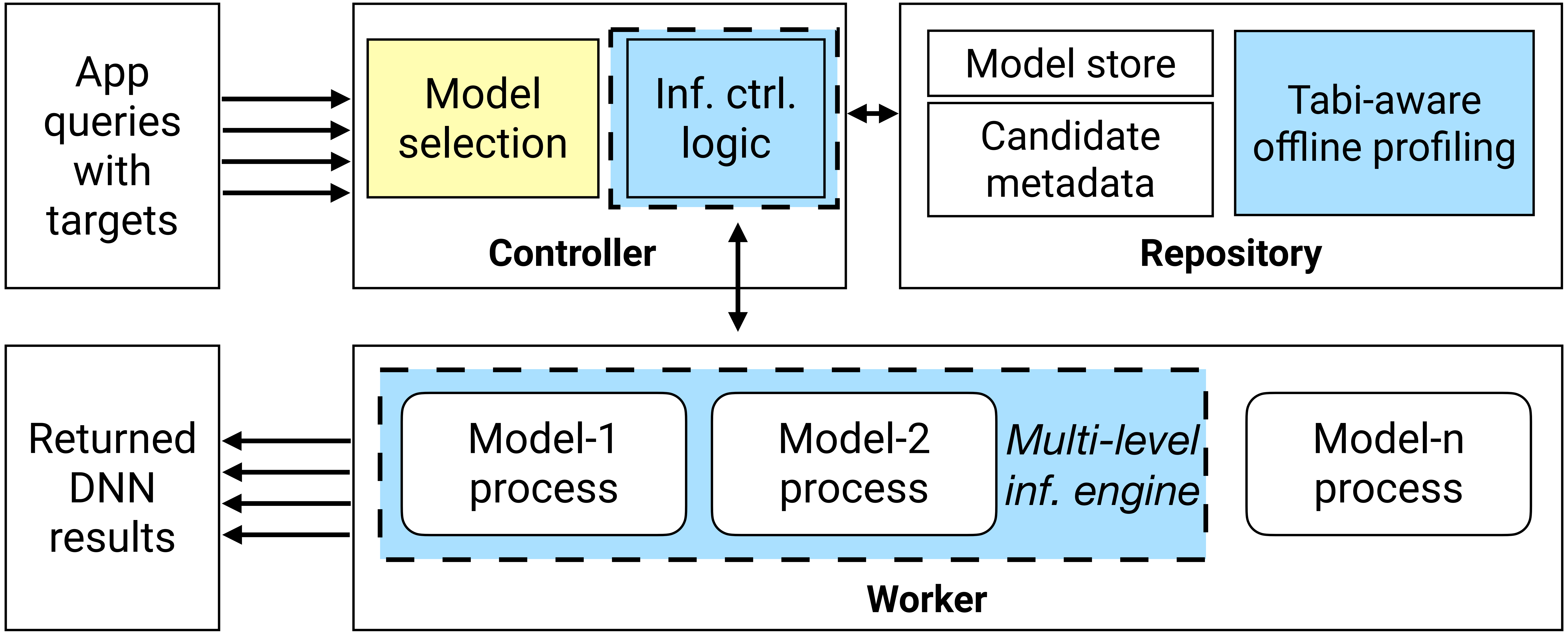
- Transformer-based language models build on the **attention mechanism**.
- **Some tokens are more important.**
- Longer sentences take more time. Time complexity: **O(n²)**.
- We **prune re-routed query texts** to accelerate LLM inference by ~15%.



THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Setting System Hyperparameters

- **Tabi-aware offline profiling**: In addition to the accuracy and latency of available models, we also **profile various hyperparameters and model pairings**. We use early-stop techniques to limited the overheads.



THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Evaluation: Average Latency

- GLUE benchmark and similar classification tasks. Single GPU.
- **Over 20%** of average **latency reduction** compared to INFaaS (ATC '21).

Same accuracy

	Method	SST-2	MNLI (-mm)	RTE	QQP	MRPC	CoLA	QNLI	STS-B	MASSIVE	CLINC
Tgt. acc. (%)	-	95	90	85	92	90	65	94	92	92	97
Accuracy (%)	INFaaS	96.1	91.2	86.6	92.3	90.9	67.6	94.7	92.4	92.5	97.3
	Cocktail	95.4	90.4	85.2	92.1	90.0	65.1	94.3	92.1	92.1	97.0
	Tabi	95.6	90.4	86.0	92.1	90.1	65.2	94.6	92.0	92.1	97.0
Latency (ms)	INFaaS	22.0	25.8	38.1	25.4	24.9	22.5	26.0	21.2	20.9	21.3
	Cocktail	17.8	22.9	34.7	20.8	20.2	18.2	22.3	17.5	15.4	17.2
	Tabi	13.2	20.3	30.0	16.0	16.5	15.7	18.7	13.4	13.5	14.8
Estimated cost & tput	INFaaS	11.6/42.7	13.3/36.1	19.5/24.6	13.2/36.3	13.2/36.4	11.8/40.7	13.5/35.5	11.4/41.9	11.2/42.4	11.5/41.8
	Cocktail	9.4/53.2	15.3/40.0	20.3/26.8	11.7/43.7	12.1/44.0	10.7/50.1	12.8/40.6	9.6/53.0	9.5/55.9	10.5/52.9
	Tabi	5.8/63.8	9.3/42.1	14.2/29.2	7.2/53.5	7.9/52.2	6.5/53.8	8.8/46.5	6.0/62.2	5.9/61.9	6.3/59.3
Latency reduction (%)	-	40/26	22/11	21/12	37/23	34/18	30/14	28/16	37/23	35/12	30/14

10%+ improvement compared to a recent baseline Cocktail (NSDI '22)

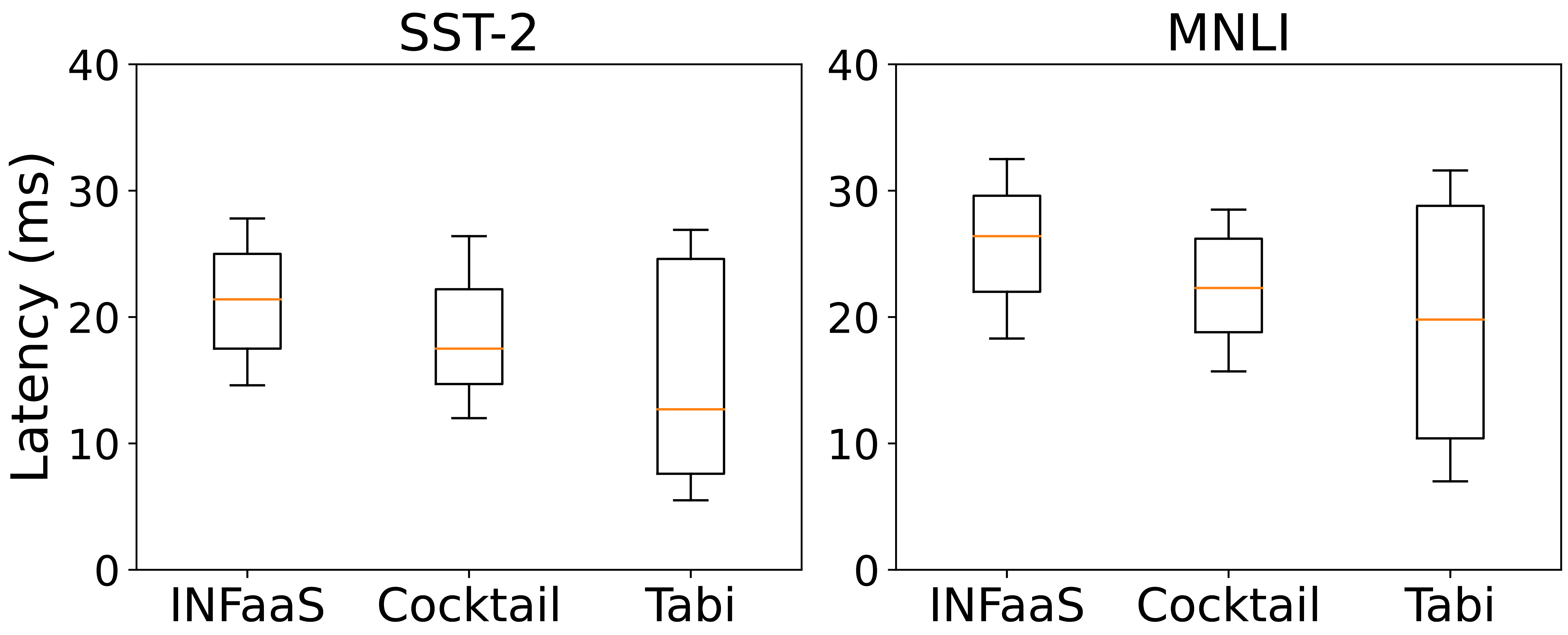
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

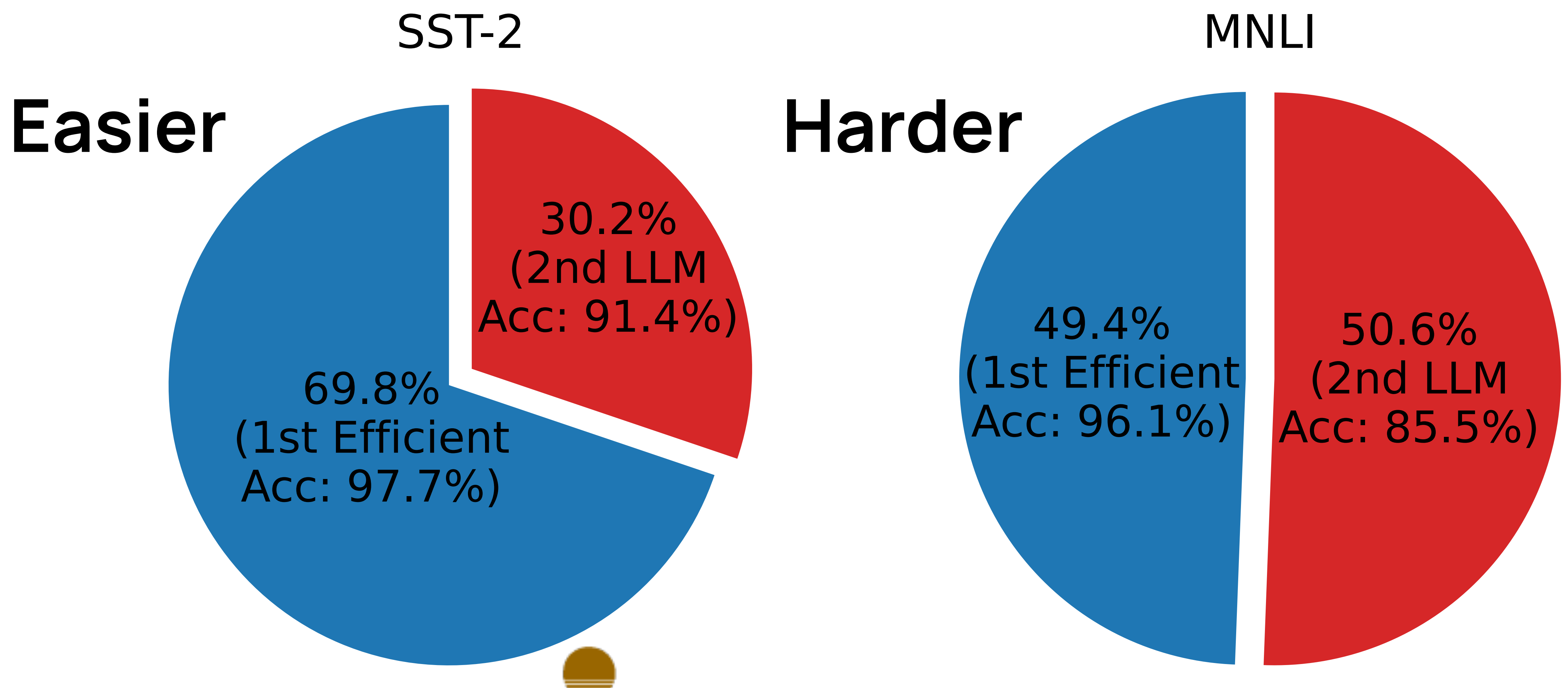
Evaluation: Tail Latency

- **Similar tail accuracy**: Attention-based word pruning **offsets the overhead** of the extra small-model level.

- For different tasks, the early-return rate is different. Higher speed-up for simpler tasks.

- A task is easy means the accuracy gap between a large and a small LM is smaller.

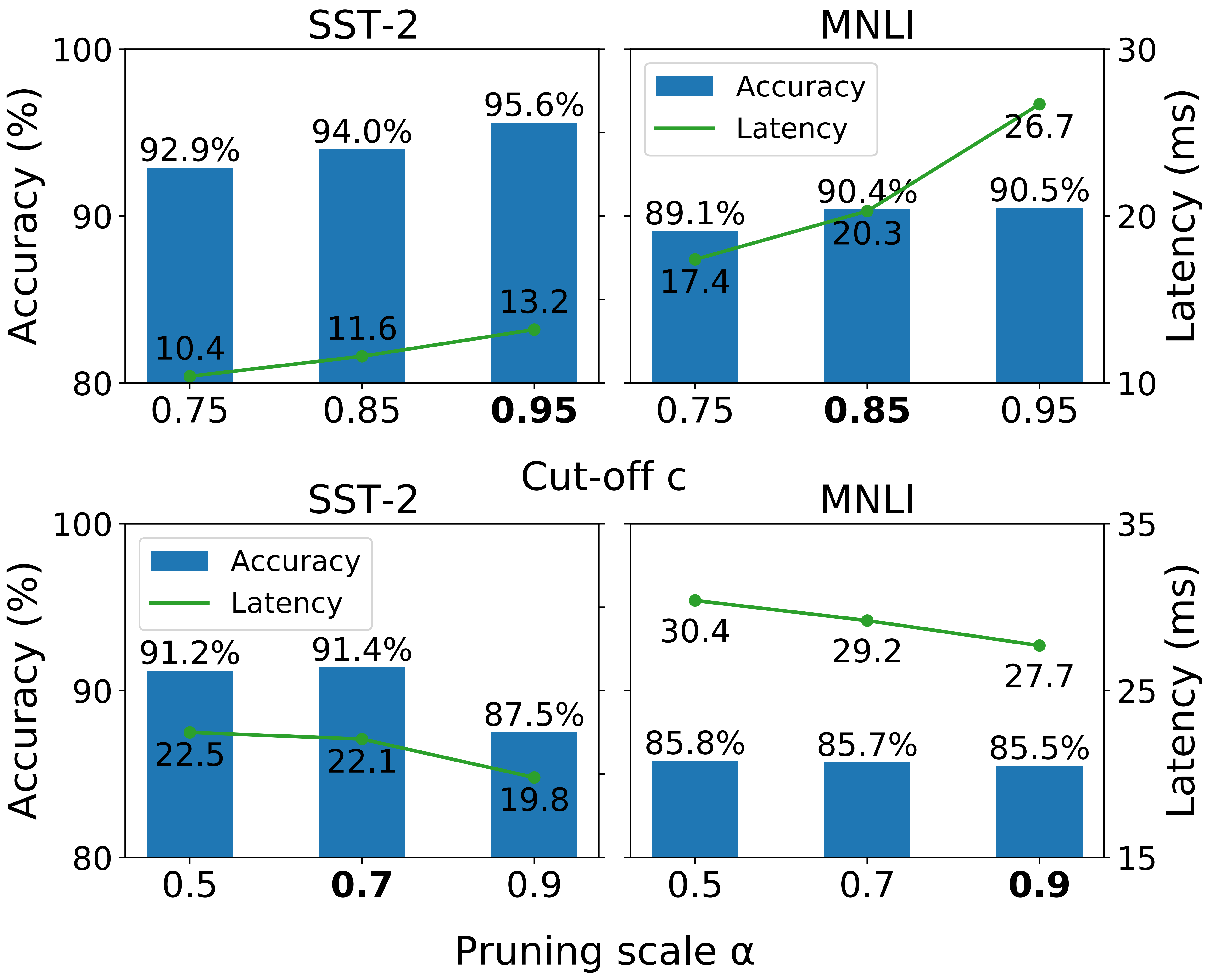




THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Evaluation: System Hyperparameters

- We set online hyperparameters through **offline profiling**.
- Dispatcher's cut-off threshold (top fig.): A higher value → re-routing more queries to the large model.
- Attention pruning scale: A higher value → more words are pruned, e.g., 4%, 14%, 63% in SST-2.
- Meet acc. target & reduce latency.



THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Evaluation: Other ML Optimizations

- Tabi (w/ vanilla models) performs similarly to early-exit DNN, in spite of the system overhead.
- Because we have multiple aspects of optimization (e.g., pruning).
- **Tabi is for high accuracy targets. Break-even points.**
- What about using more models rather than 2? Tail latency will degrade a lot.

Task	Tabi	DeeBERT-BERT-base	DeeBERT-RoBERTa-base	DeeBERT*-RoBERTa-L
SST-2	95.6/**40%**	93/**40%**	94.4/26%	**95.9**/38%
MNLI	**90.4**/22%	83.9/14%	87/19%	**90.4/24%**

Table 4. Accuracy (%) and latency reduction of Tabi and DeeBERT [80]. Tabi has similar performance even compared to a customized LLM. * denotes requiring ML expertise.

Accuracy (%)	Mean latency	Median latency	99% latency	Level return distribution
90.2 (-0.2%)	22.0 (+8.4%)	12.7 (-2.3%)	49.4 (+70.3%)	45.6%/36.8% /17.6%

Table 5. Compared to Tabi's two-level decision, using three models invokes the LLM less but prohibitively increases the tail latency by 70.3%, and so does the mean.

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Thank you!

- Tabi is a model-less inference system optimizing for **discriminative BERT-like models** with fast parameter scaling.
- Tabi uses a **multi-level** structure with small and large models to reduce latency by **invoking LLMs less frequently** and **on optimized data**.
- Tabi in essence is a **system implementation of ML techniques** like early-exit and attention-based token pruning but **with vanilla models**.
- Tabi optimizes the **inference pipeline** and targets accuracy-demanding applications.

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY