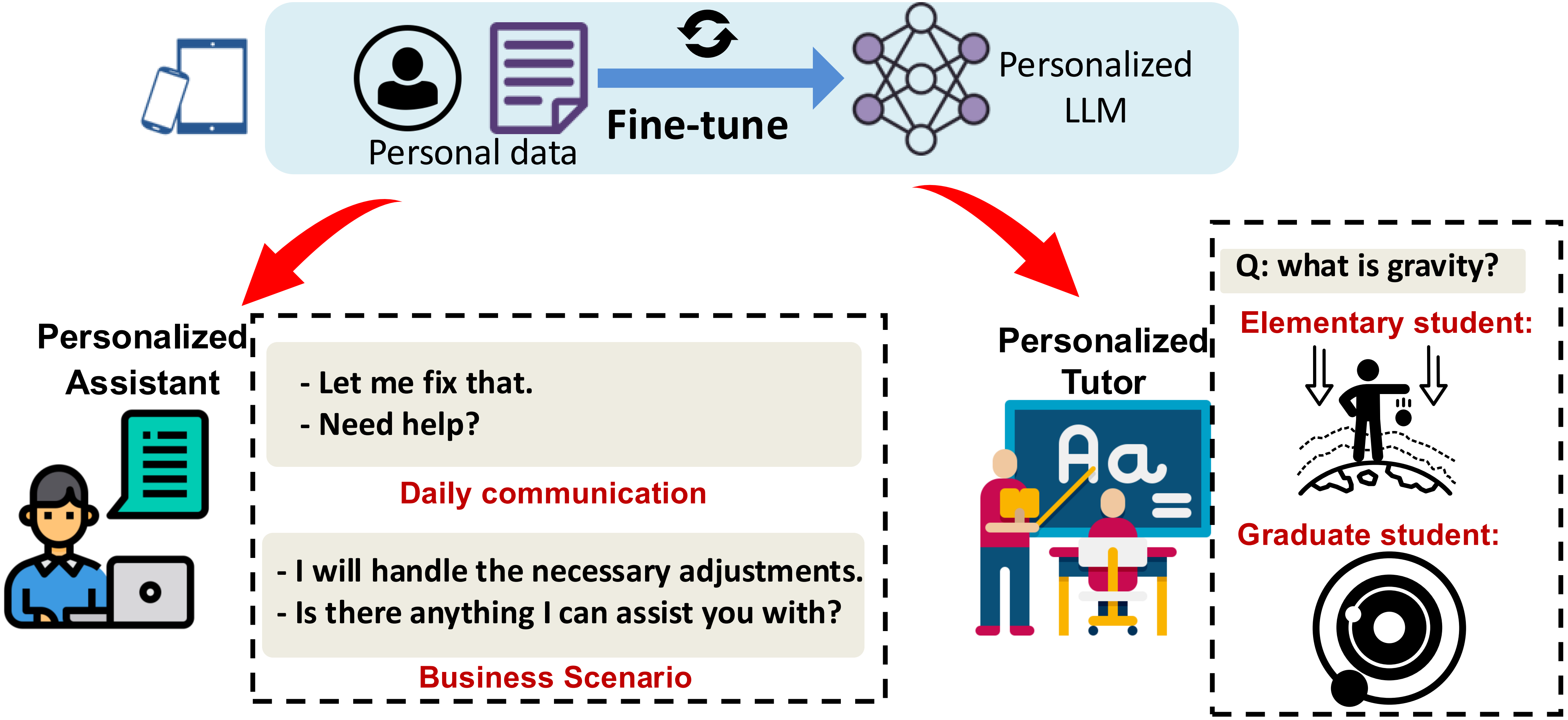
Never Start from Scratch: Expediting On-Device LLM Personalization via Explainable Model Selection

Haoming Wang, Boyuan Yang, Xiangyu Yin, and Wei Gao
University of Pittsburgh

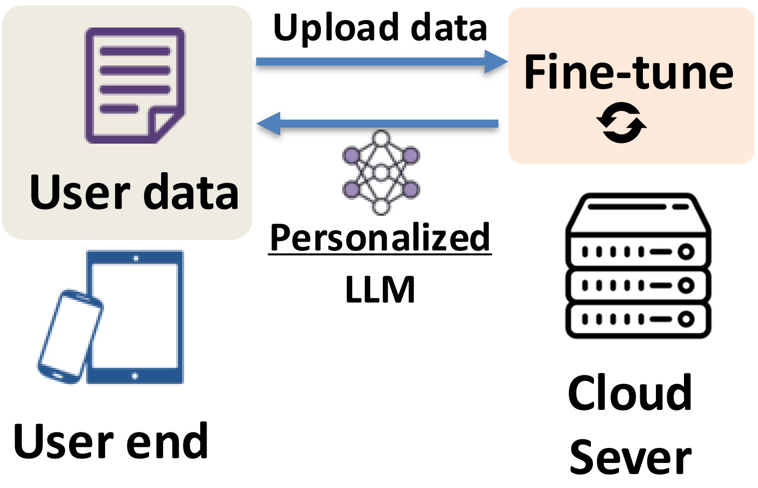
ACM MobiSys 2025

LLM personalization on Mobile Devices



Existing Solutions

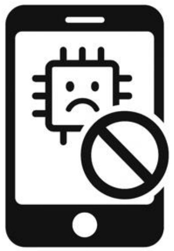
Upload user data to the cloud



Impairing user's data privacy

Fine-tune LLM at the local device

How to address such on-device challenges?



Limited compute power

Insufficient personal data

On-device personalization challenges

Limited compute power

- ❖ **Efficient fine-tuning method**
  - LoRA [1] (Low-Rank Adaptation)
  - Prompt tuning [2]
  - …

- ❖ **Not efficient enough**
  - ~1 second per training steps on a flagship smartphone (Qwen2-0.5B on Google Pixel 9 Pro)

Insufficient personal data

Accumulating enough data → **Take very long time**

Continual learning [3] → **Too expensive for mobile devices**

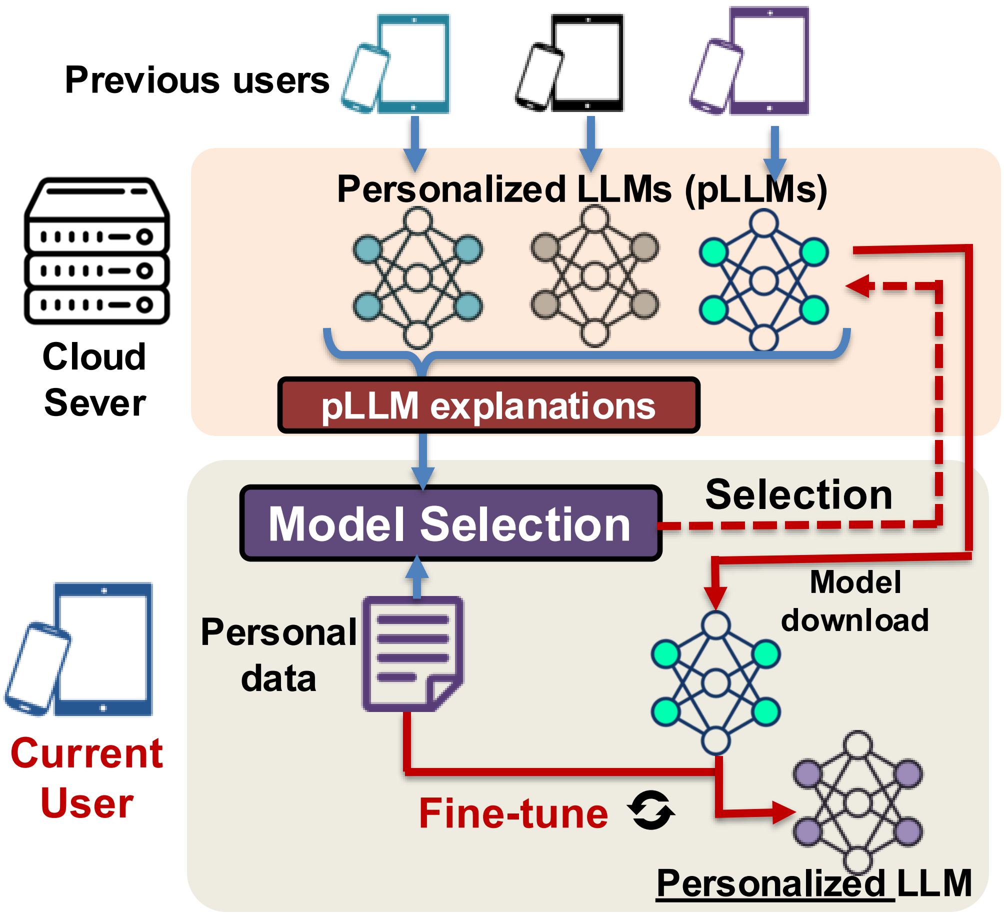
[1] J Lin, et al. Lora: Low-rank adaptation of large language models. ICLR 2022

[2] B Lester, et al. The Power of Scale for Parameter-Efficient Prompt Tuning. Arxiv 2021

[3] A Razdaibiedina, et al. Progressive prompts: Continual learning for language models. ICLR 2023

Our Solution: Never Start from Scratch!

Initialize personalization from the existing personalized LLMs



Server end:

(1) Personalized LLM pre-cached on the cloud server

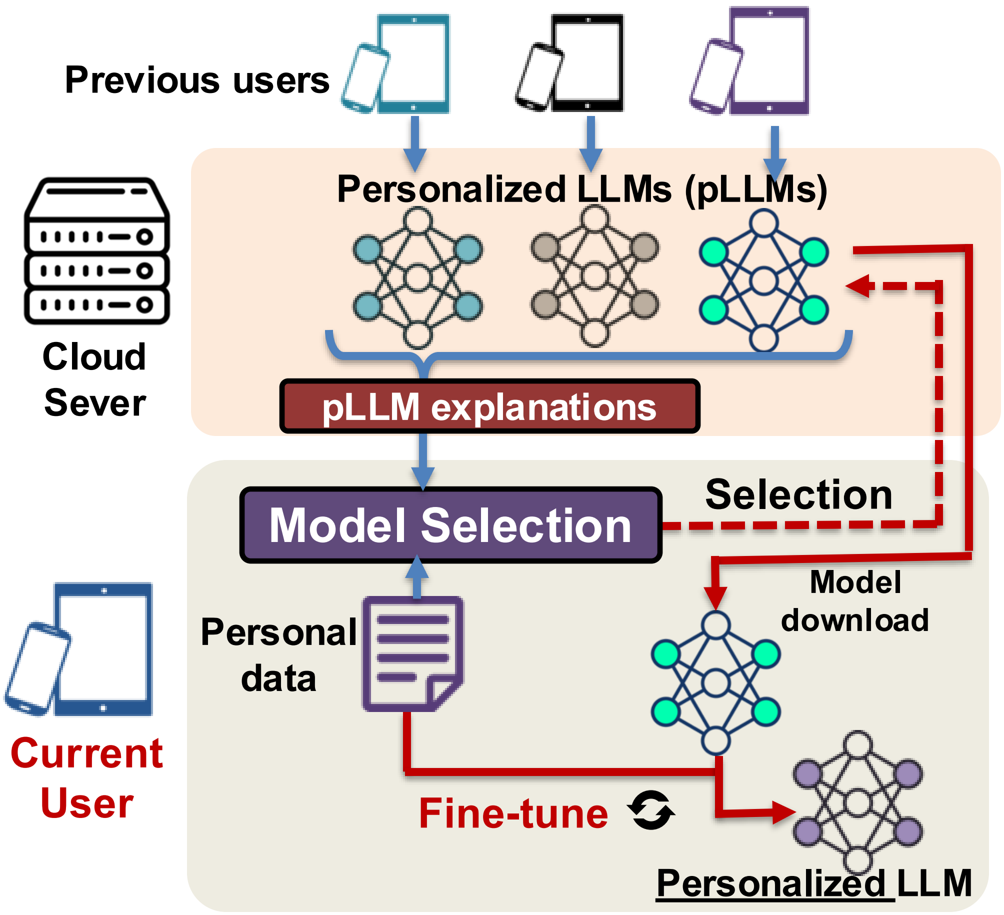
(2) Pre-compute the explanations for pLLMs

On device :

(3) Select the pLLM that best resembles the personal data based on the explanations of pLLMs

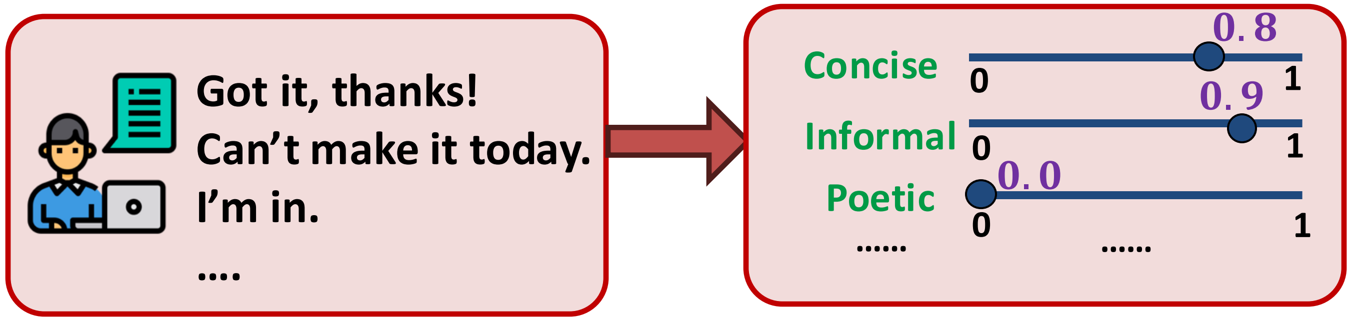
(4) Locally fine-tune the selected pLLM with personal data

Our Solution: Never Start from Scratch!



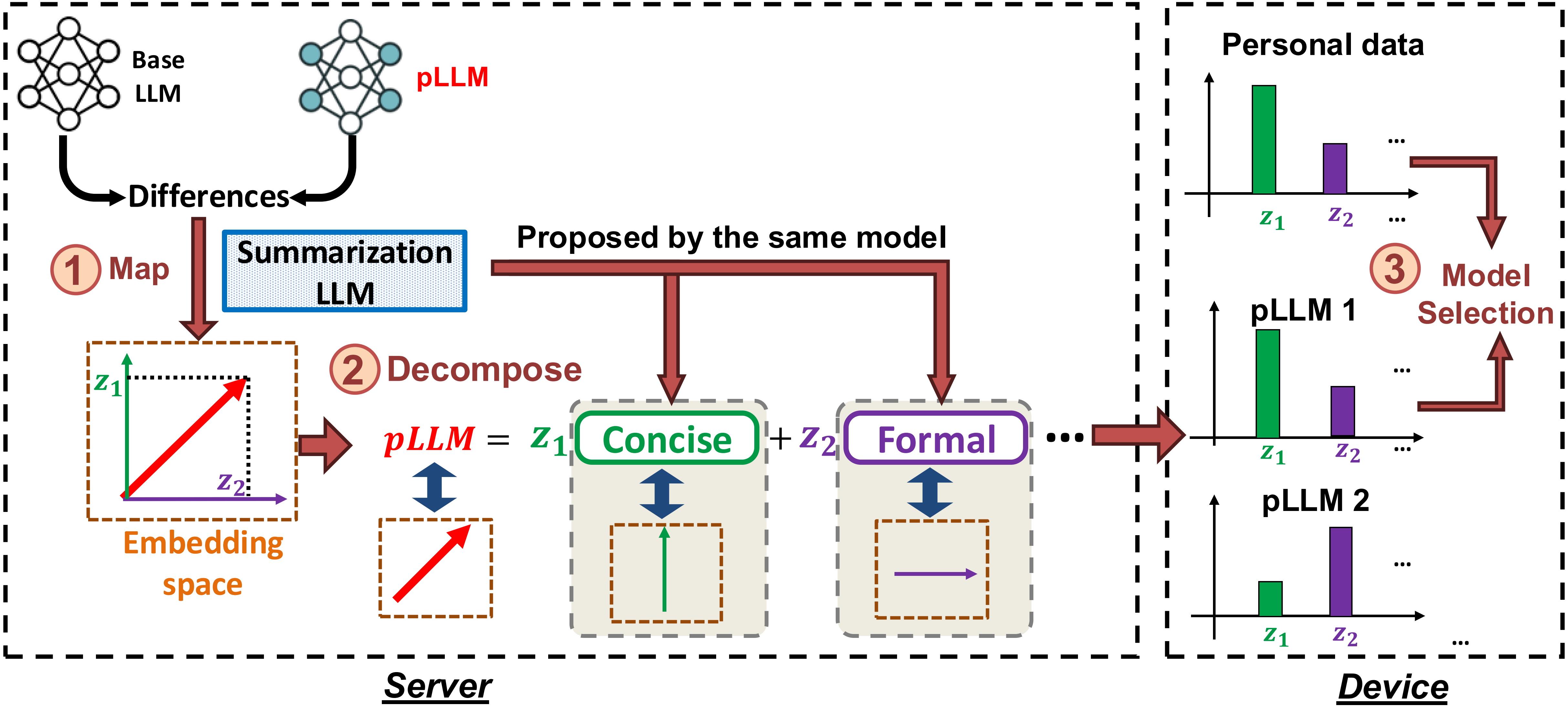
❖ **Requirements for pLLM explanations:**
- **Explainable**: in natural language to ensure users' trust
- **Quantitative:** facilitate model selection

❖ **Format of explanations:**

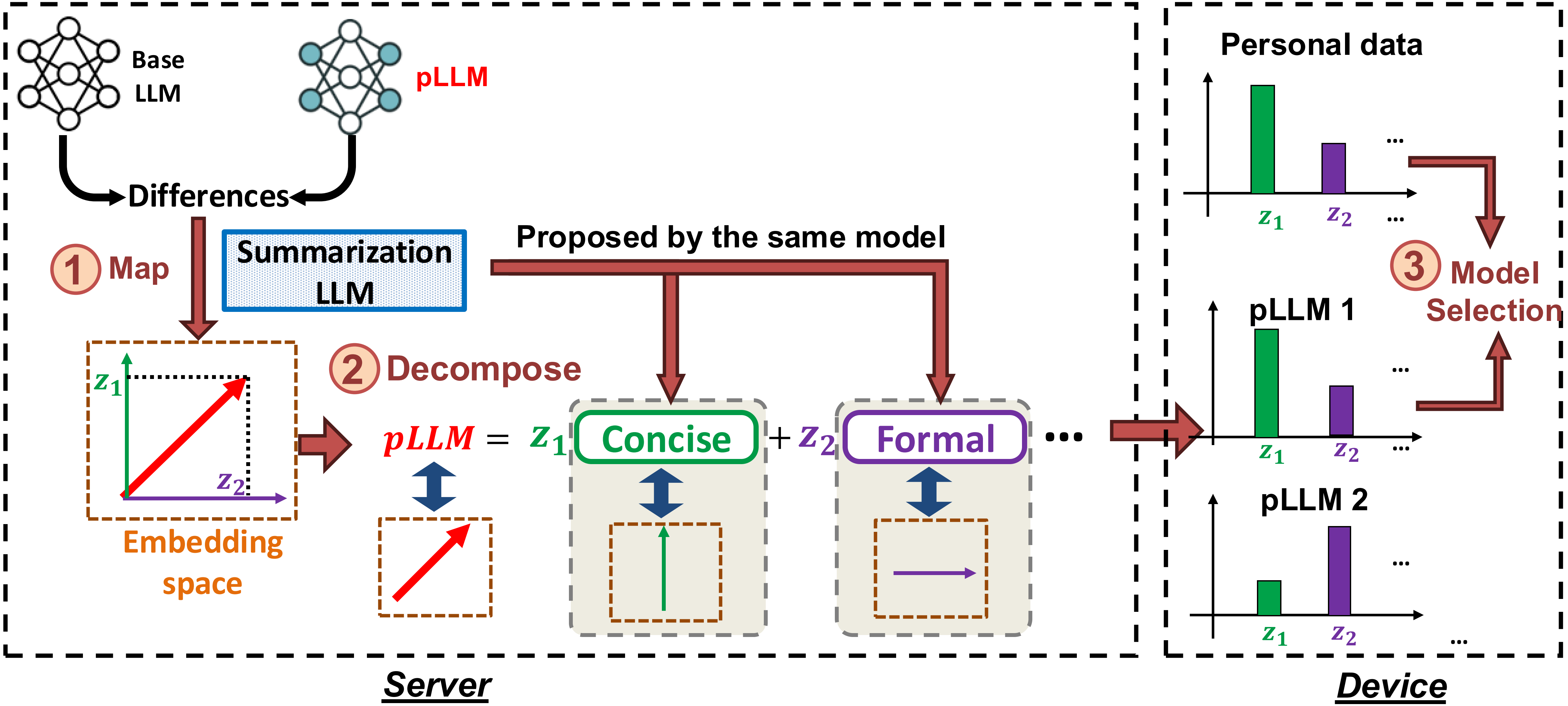


Personalized messages **Quantitative explanation**

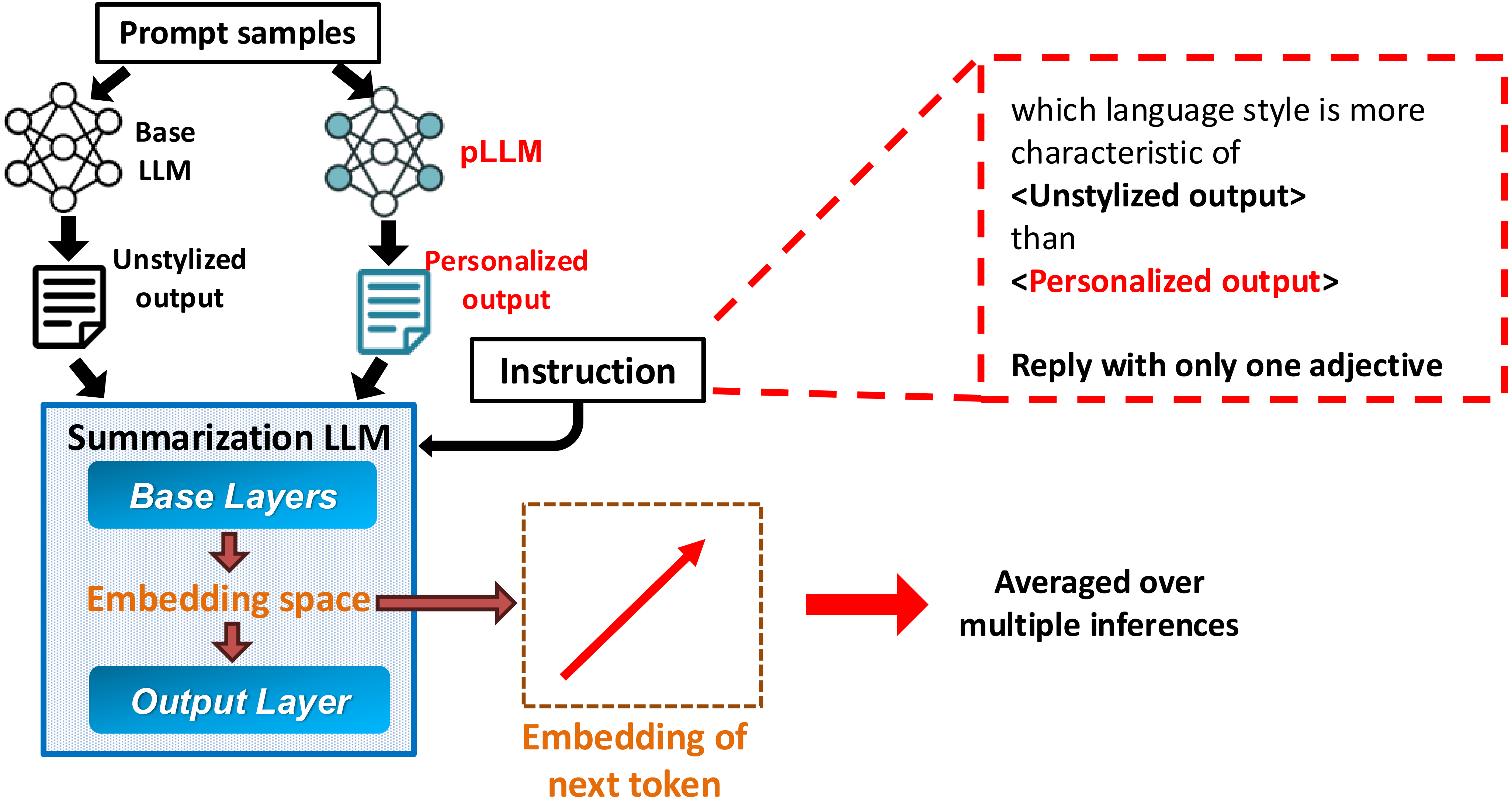
eXplainable Personalized Tuning (XPerT)



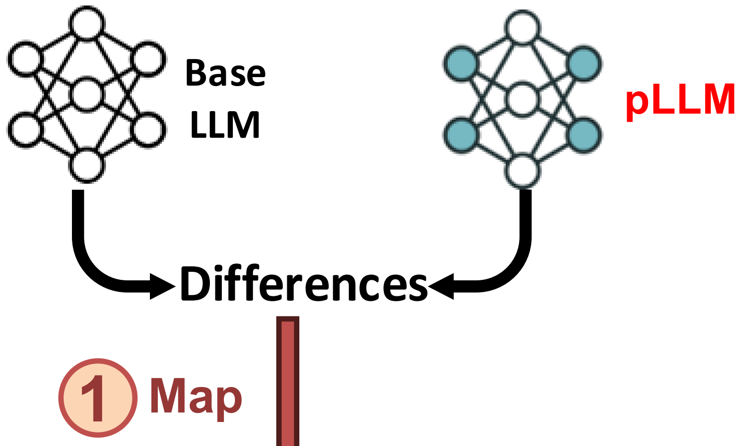
XPerT: 1 Mapping differences to Embedding space



XPerT: ① Mapping Differences to Embedding Space



XPerT: 2 Decomposing the Embedding

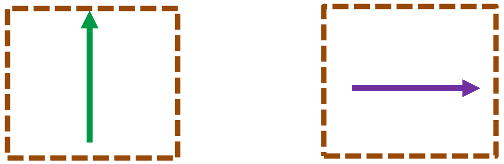


Steps of decomposition:

1 Propose **candidate natural language explanations**:

Concise Formal •••

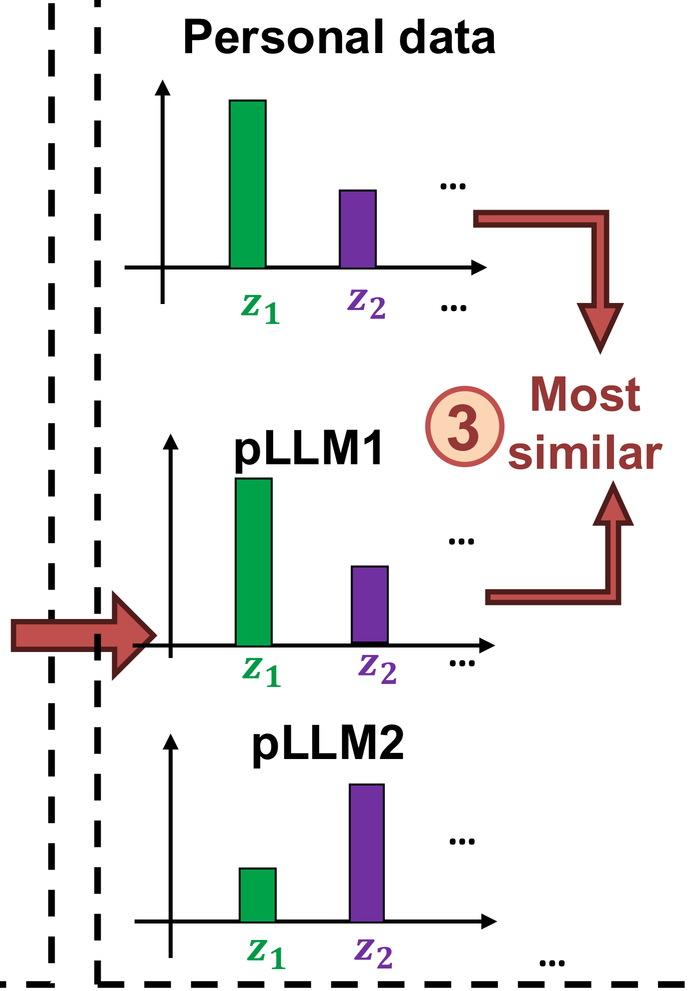
2 Compute **embeddings** for each candidate:



•••

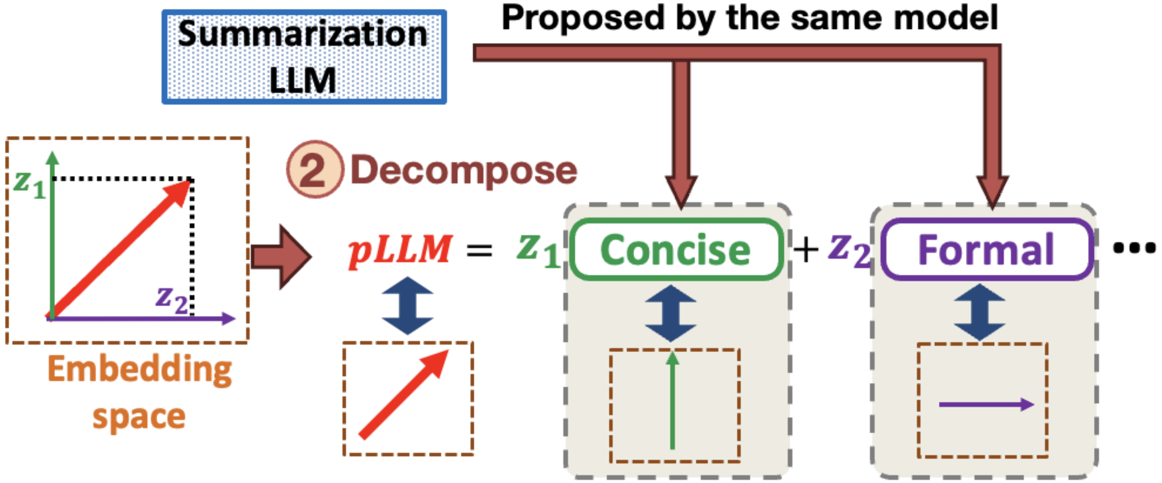
3 Repeat until decomposition error is small enough

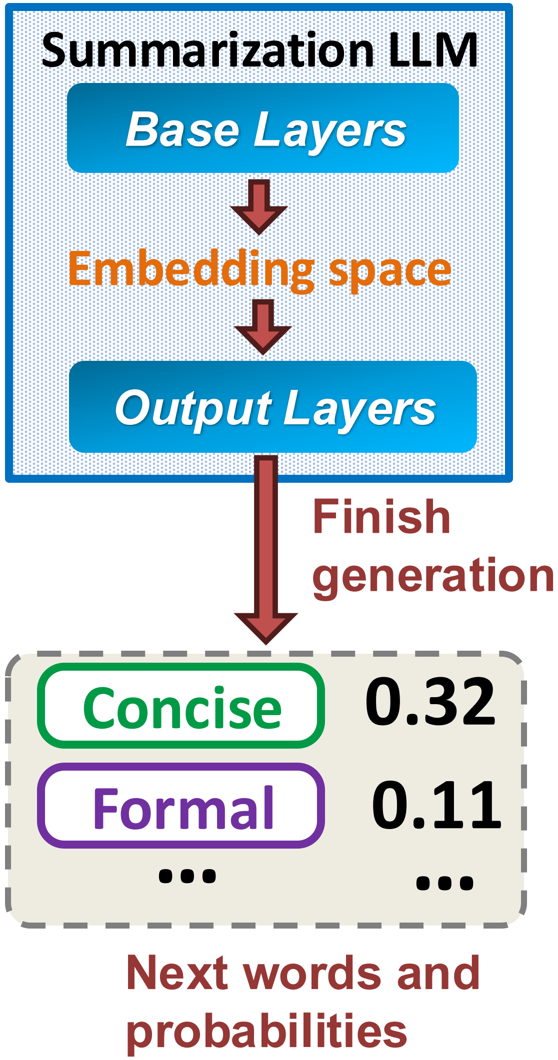
Server



Device

XPerT: 2 Decomposing the Embedding



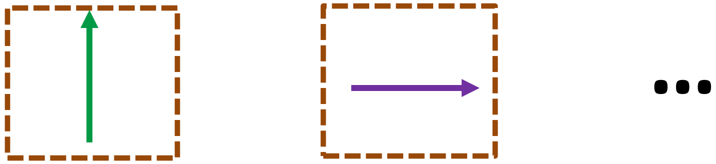


Steps of decomposition:

1 Propose **candidate natural language explanations**:

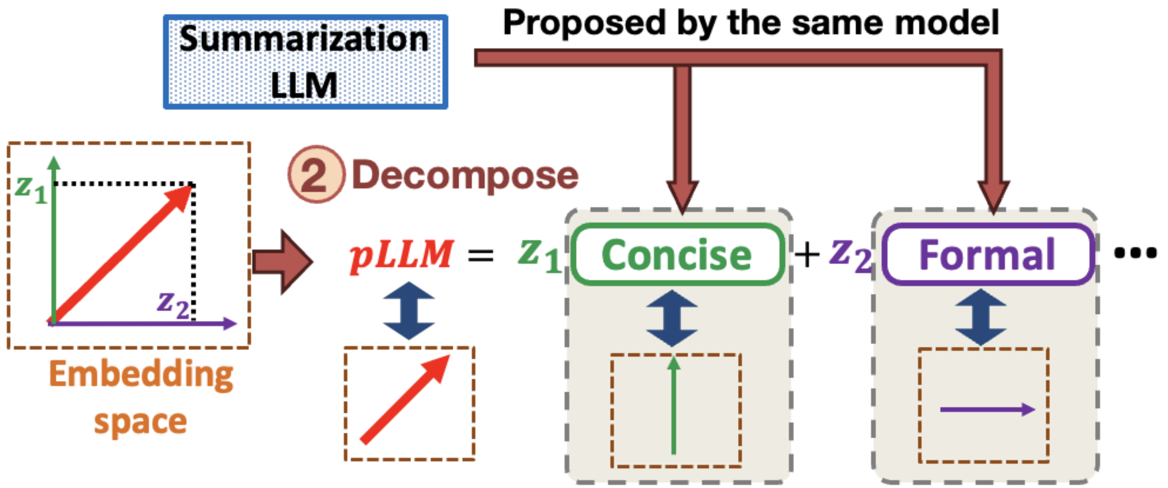
Concise Formal ...

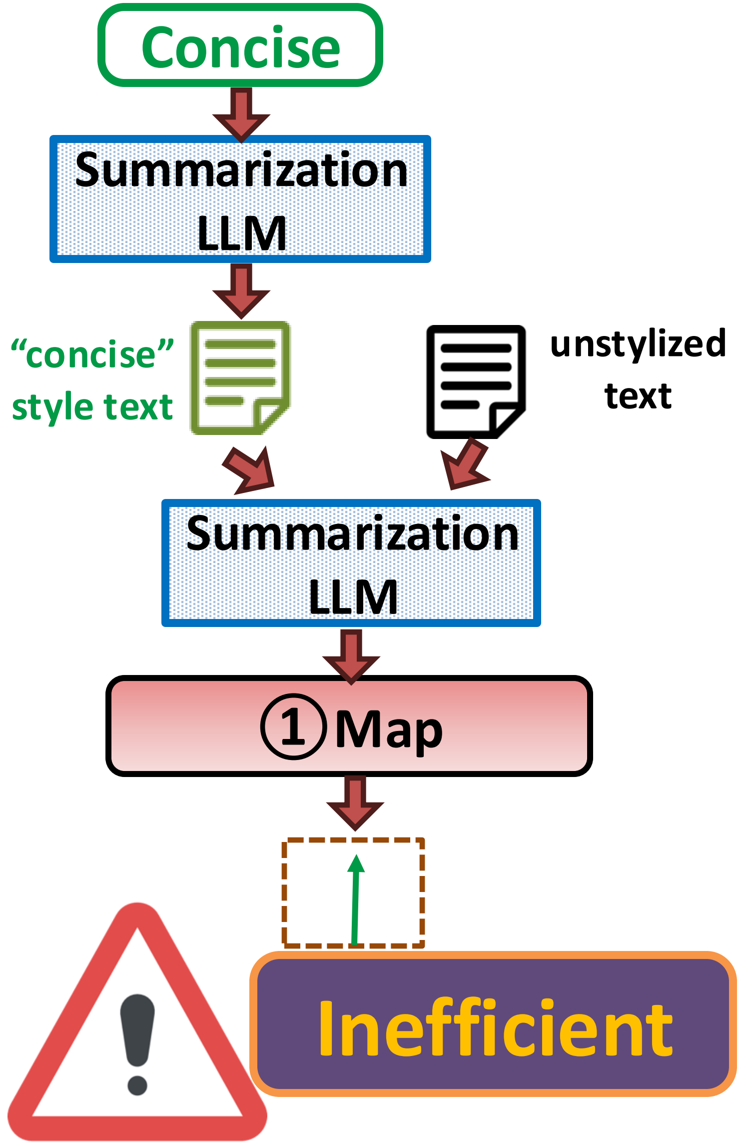
2 Compute **embeddings** for each candidate:



3 Repeat until decomposition error is small enough

XPerT: ② Decomposing the Embedding





Steps of decomposition:

1 Propose **candidate natural language explanations**:

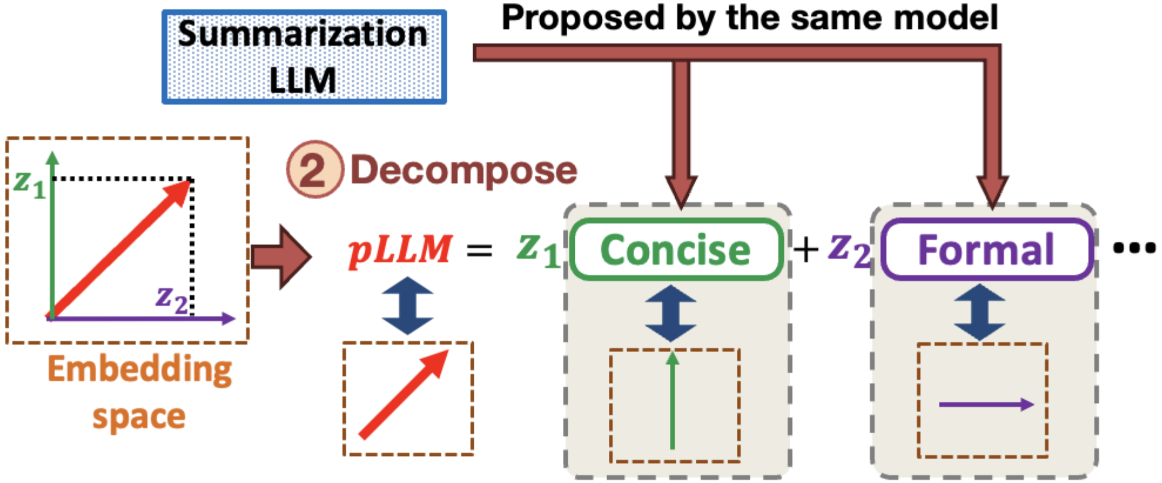
Concise Formal ...

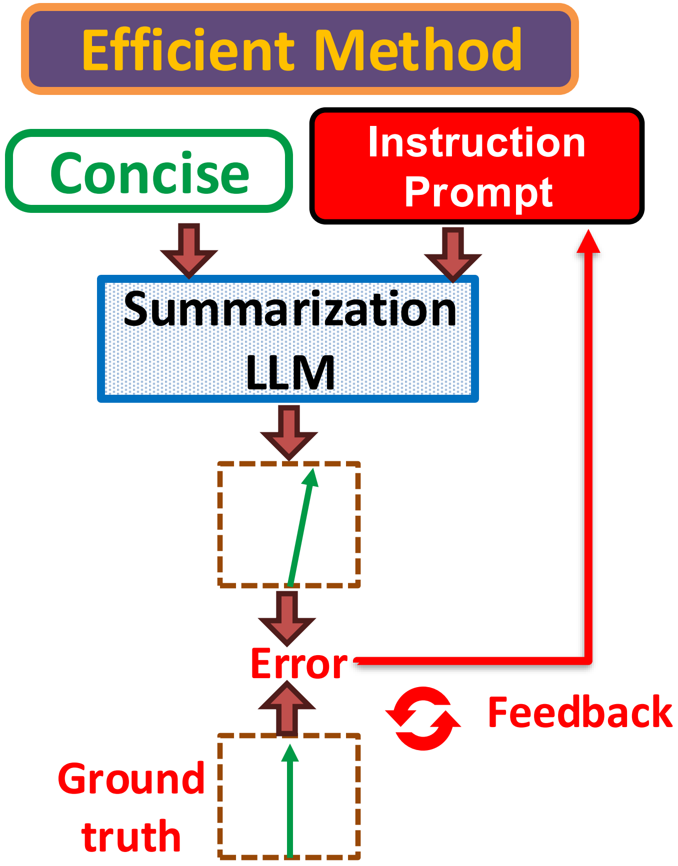
2 Compute **embeddings** for each candidate:

...

3 Repeat until decomposition error is small enough

XPerT: 2 Decomposing the Embedding





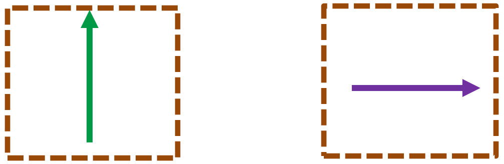
Embedding generated by original inefficient method

Steps of decomposition:

1 Propose **candidate natural language explanations**:

Concise Formal ...

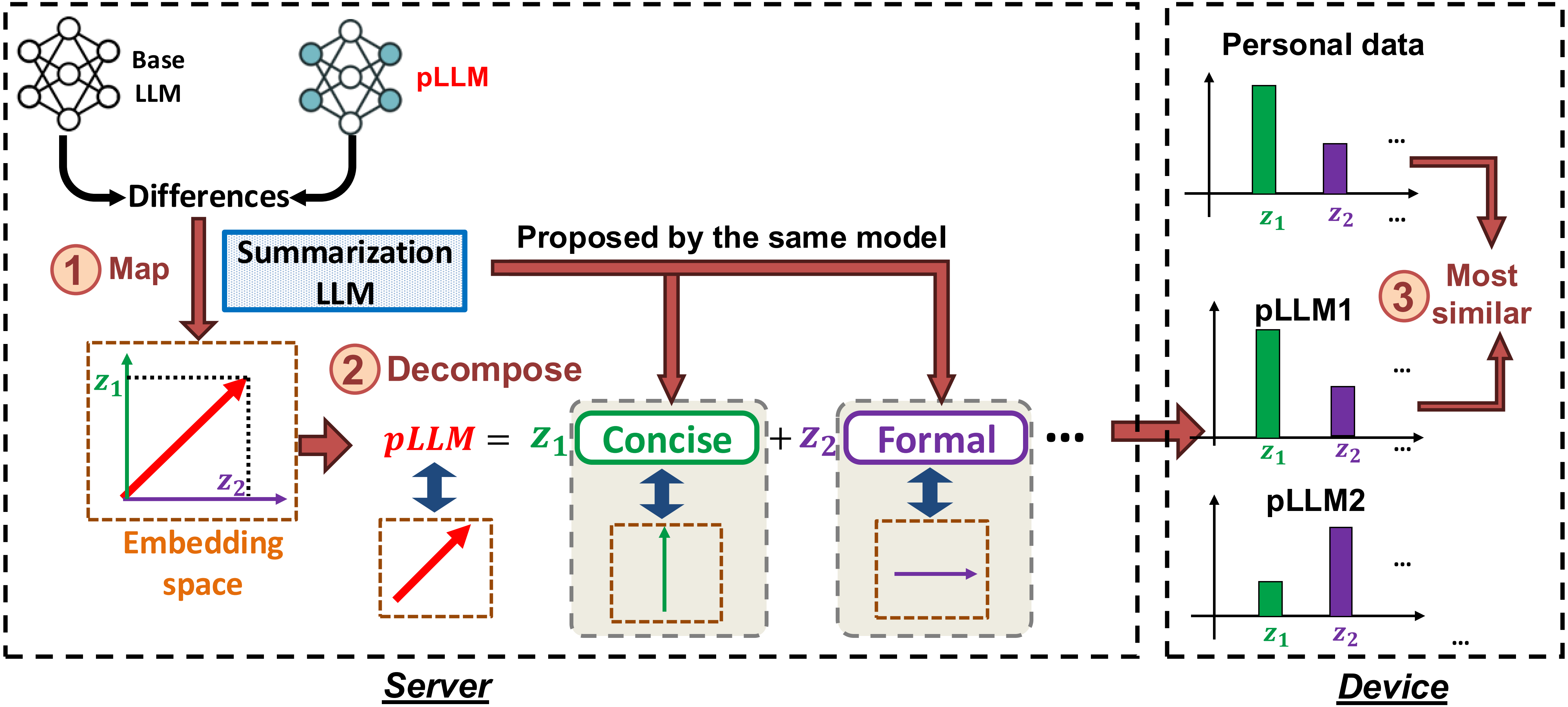
2 Compute **embeddings** for each candidate:



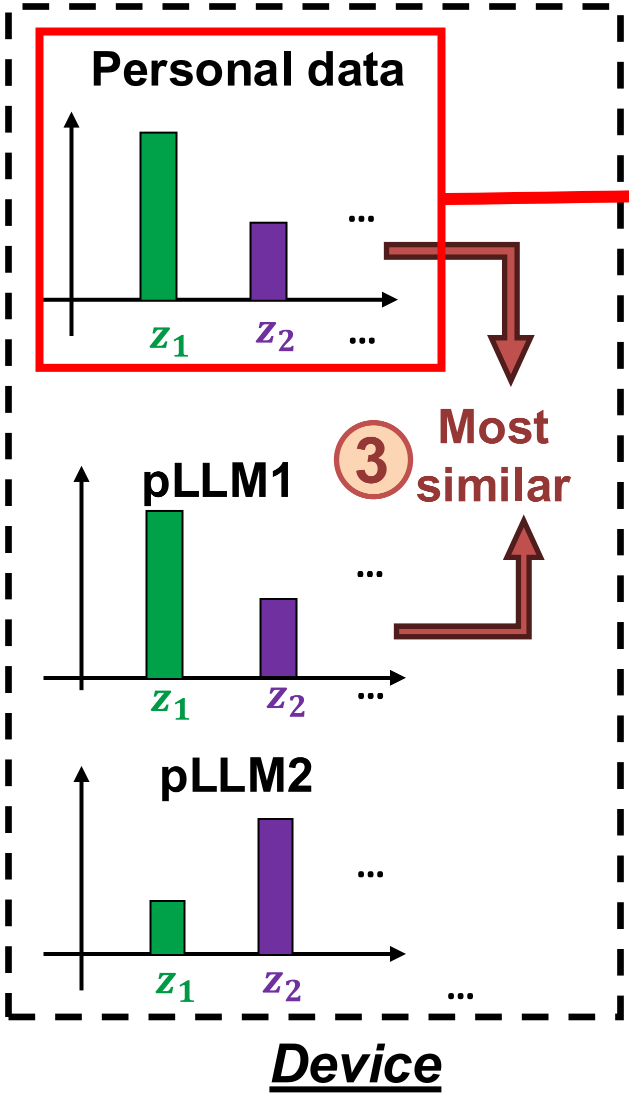
...

3 Repeat until decomposition error is small enough

XPerT: ③ On-device Model Selection



XPerT: 3 On-device Model Selection

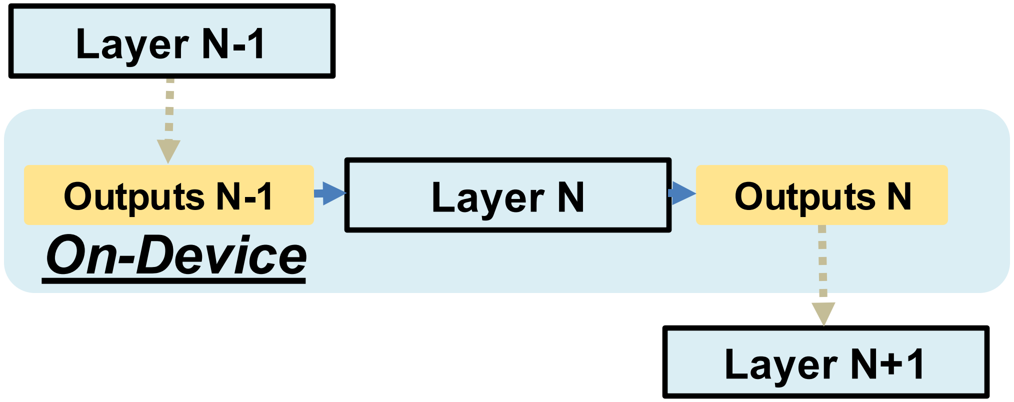


Step 1: Map

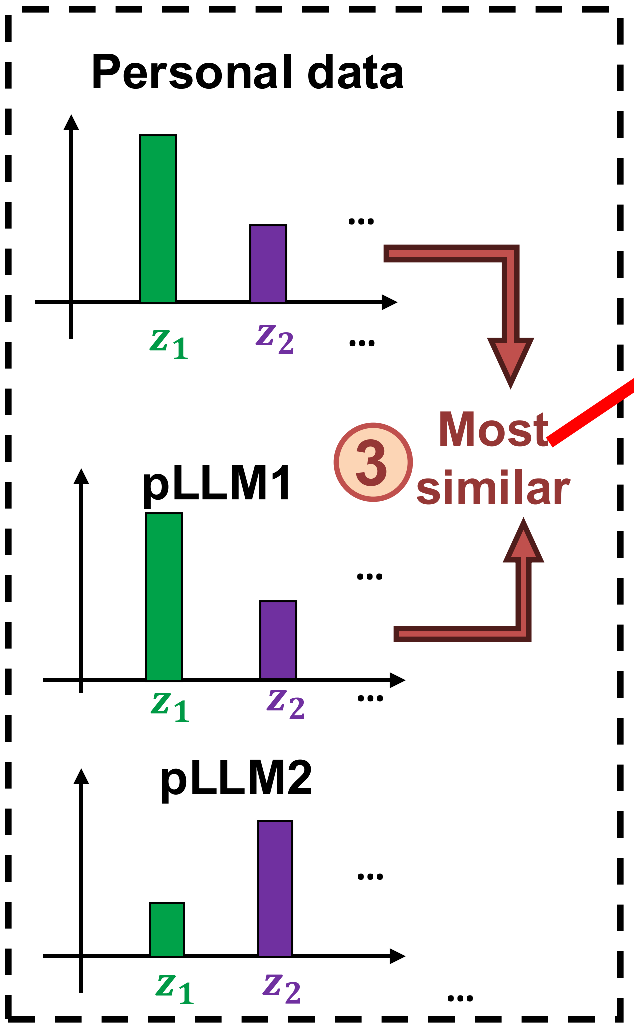
Run Step 1 on device

Model partition to fit summarization LLM into mobile device's memory

...



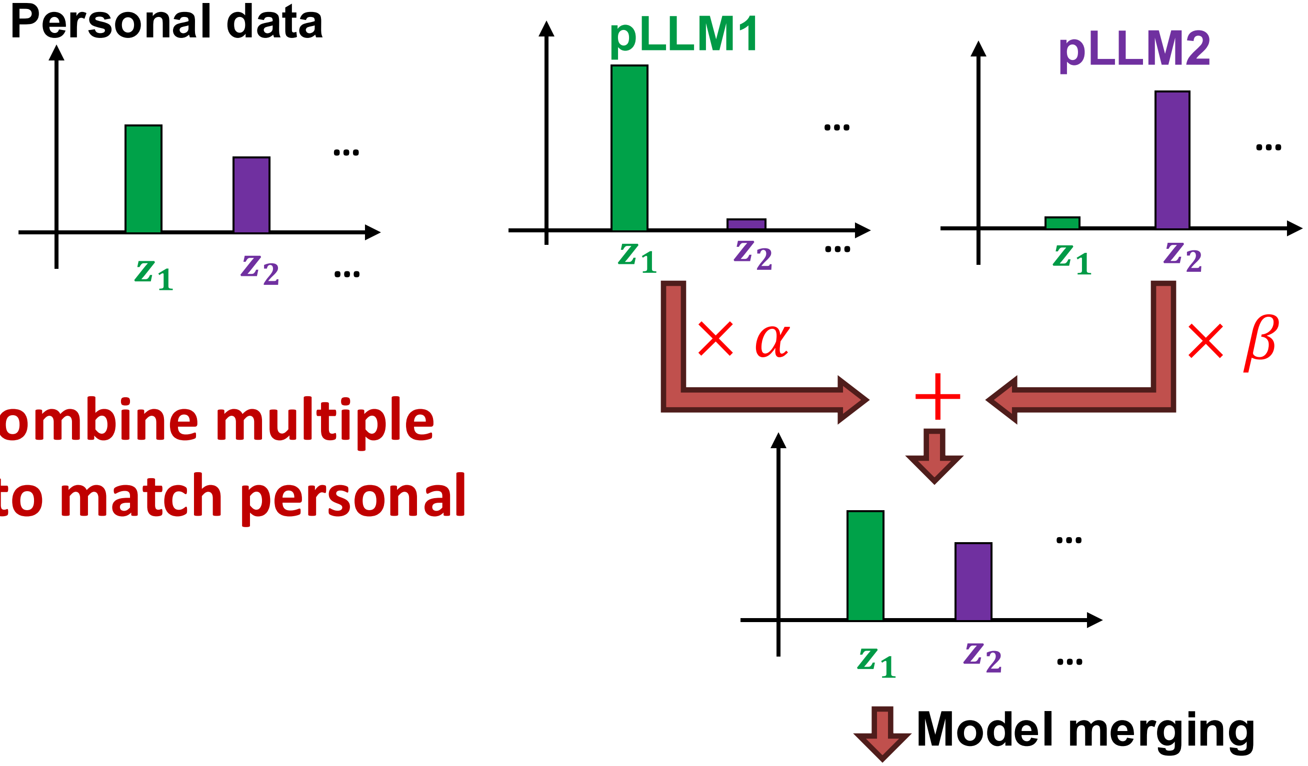
XPerT: 3 On-device selection



Device

If such a pLLM doesn't exist:

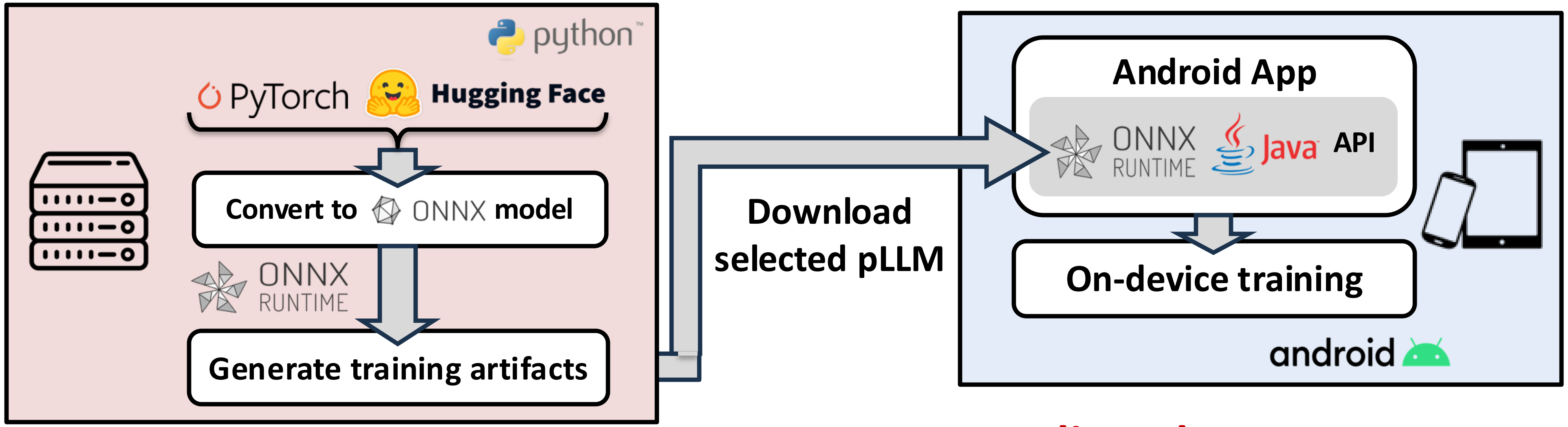
Try to combine multiple pLLMs to match personal data:



Merged pLLM = $\boldsymbol{\theta}_{base} + \boldsymbol{\alpha}\ (\boldsymbol{\theta}_{pLLM1} - \boldsymbol{\theta}_{base}) + \boldsymbol{\beta}\ (\boldsymbol{\theta}_{pLLM2} - \boldsymbol{\theta}_{base})$

Implementation

Implement LLM Fine-tuning on smartphones:



Offline Phase:
Convert Model and Data format

Online Phase:
Model training as background Android service

Experiment Settings

- **Datasets**
  - **Synthetic**: QA data with diverse language styles generated by ChatGPT

Expertise	elementary / expert
Informativeness	concise / informative
Style	friendly/ unfriendly/ sassy/ sarcastic / persuasive / neutral / poetic

  - **Real-world**: Combination of 3 text datasets with multiple language styles

CDS[1]	poetry, lyrics, tweets, Shakespeare
Gutenberg3[2]	fantasy, romance, and sci-fi
ScientificPapers[3]	academic

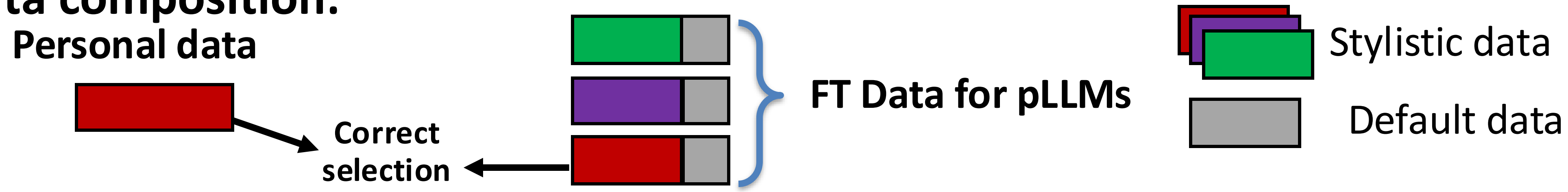
- **pLLMs and smartphone models**
  - Llama-3.2-1B on One Plus 12R
  - Qwen2-0.5B on Pixel 9 Pro
  - SmolLM-360M on Pixel 7
- **Baseline Selection Method**
  - **Exhaustive Search**: evaluates each pLLM's output with the personal data and selects the best one.
  - **Bayesian Optimization**: Frames pLLM selection as a hyperparameter optimized via Bayesian optimization
  - **HyperBand**: Leverages the bandit principle to find optimal hyperparameters

[1] K Krishna, et al, Reformulating Unsupervised Style Transfer as Paraphrase Generation. EMNLP2020
[2] R Csaky, et al. The Gutenberg dialogue dataset. Arxiv 2020
[3] A Cohan, et al. A Discourse-Aware Attention Model for Abstractive Summarization of Long Documents. Arxiv 2018

Experiment Results

- **Comparing with fine-tuning from scratch**

Data composition:



	Llama-3.2-1B on One Plus 12R			
Synthetic	Acc	FT-time	Energy	Data
From scratch	-	97.8min	15.7kJ	0%
30% similarity	25.0%	92.4min	14.9kJ	4.6%
50% similarity	53.6%	81.8min	13.3kJ	16.7%
70% similarity	85.7%	56.7min	9.0kJ	17.1%
80% similarity	96.4%	32.9min	5.3kJ	24.7%
90% similarity	96.4%	17.9min	2.8kJ	35.7%

Cost of model fine-tuning

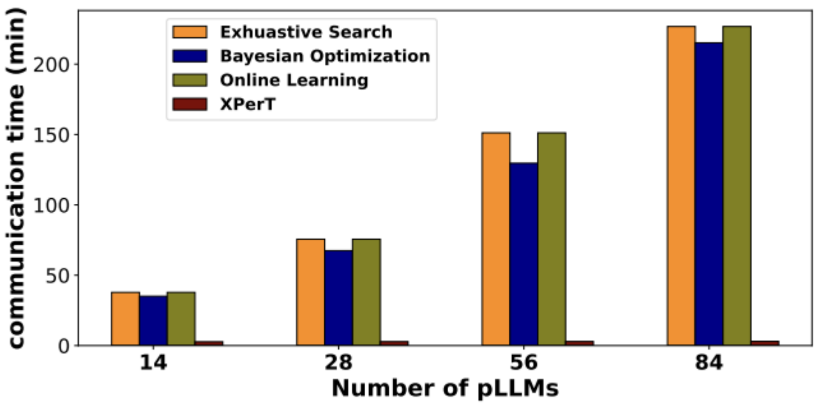
	Llama-3.2 1B on One Plus 12R		
Synthetic	BLEU	ROUGE-1	ROUGE-L
From scratch	0.13	0.32	0.23
30% similarity	0.13	0.33	0.21
70% similarity	0.12	0.33	0.21
90% similarity	0.15	0.33	0.22

Performance of fine-tuned model

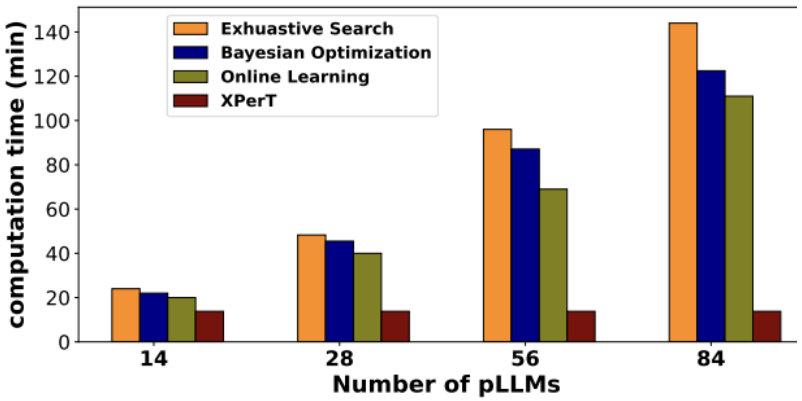
- reduce **computation cost** (up to 83%) and improve **data efficiency** (up to 51%)
- without decreasing model performance

Experiment Results

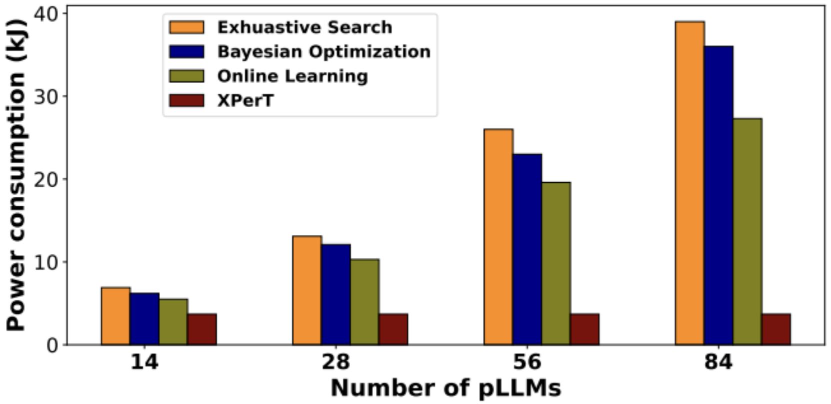
- **Comparing with baseline selection methods:**



Communication cost



Computation cost



Time consumption

- **The selection cost of**
  - **Baselines**: linearly increase with the number of pLLLMs
  - **XPerT**: retain a constantly low level

Experiment Results

- **Validating the Explainable Latent Space**

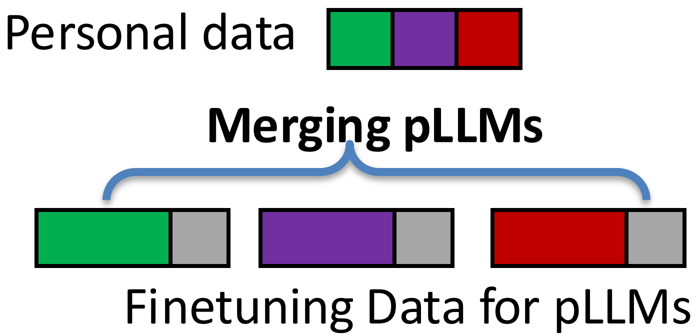
Style	Level 1	Level 2	Level 3	Level 4
Elementary	Elementary school students	Middle school students	Undergraduates	PhD students in the field
Formality	Slang, casual expressions	Everyday language, for friendly chat	Professional but with a more conversational tone	Professional language, used in corporate settings

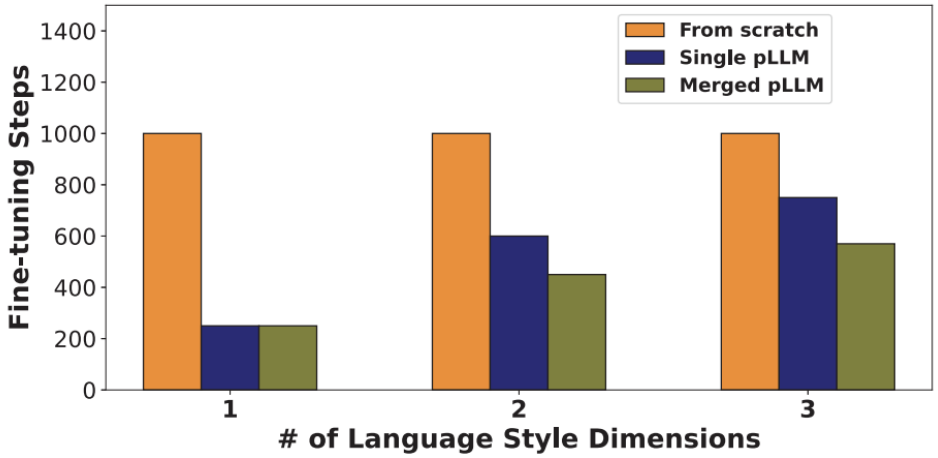
Synthesize language style with different levels

Level	2	3	4
1	0.34	0.76	1
2		0.47	0.72
3			0.28

Measure the distance of coefficeints by L1 norm

- **On-Device Model Merging**





personal data as combinations of language styles

Summary

❖ Efficient on-device LLM personalization

- **XPerT**: fine-tune the proper pLLM cached at the cloud server with on-device personal data
- **Explainability** for trustworthy model selection
- reduce **computation cost** (up to 83%) and improve **data efficiency** (up to 51%)

❖ QR code for more information

Lab Website

https://pittisl.github.io/

(presentation slides included)

Github repo

https://github.com/pittisl/ExplainablePersonalization

Thank you!