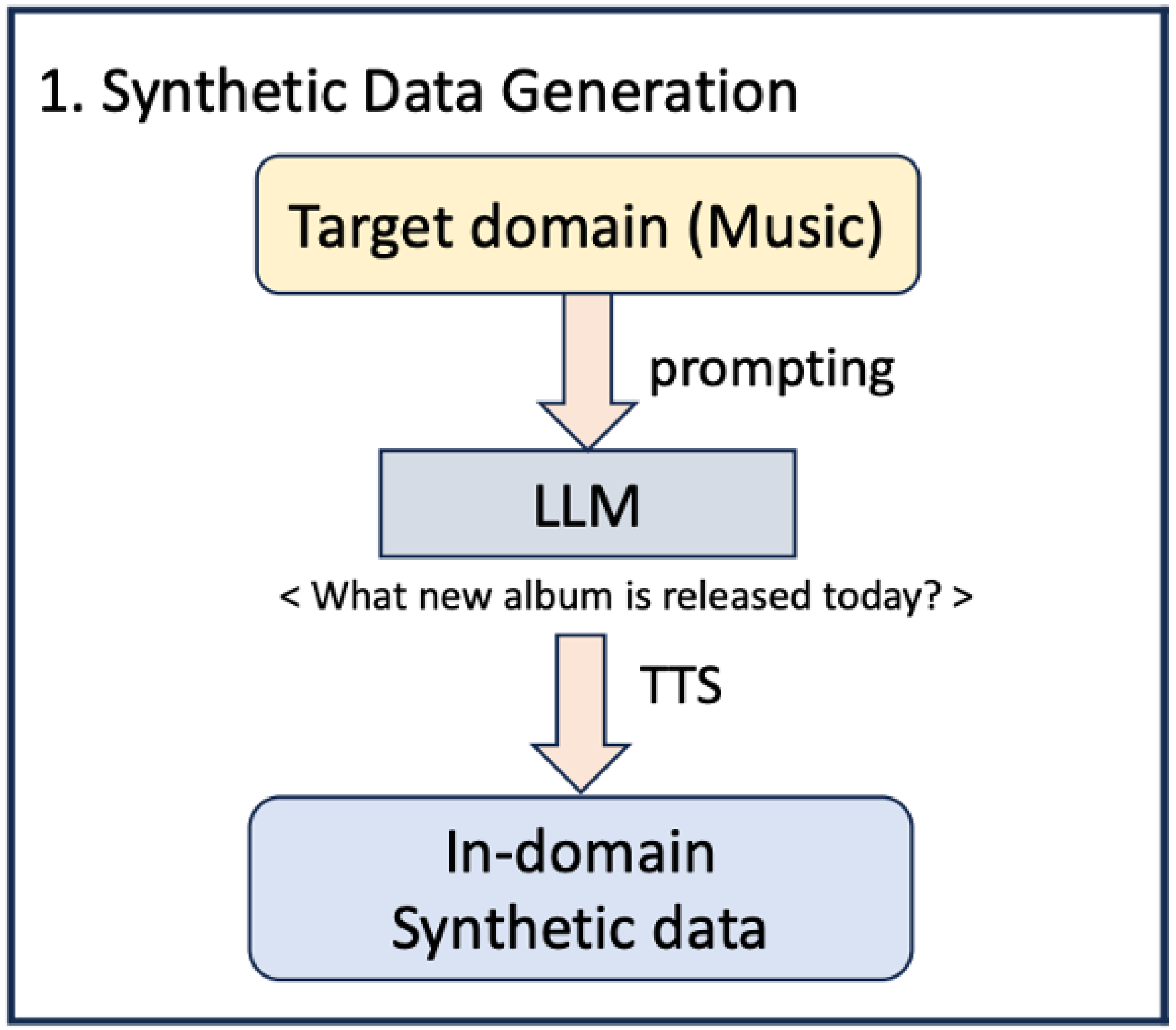
A Domain Adaptation Framework for Speech Recognition Systems with Only Synthetic data

Minh Tran, **Debjyoti Paul**, Yutong Pang, Laxmi Pandey, Kevin Jiang, Jinxi Guo, Ke Li, Shun Zhang, Xuedong Zhang, Xin Lei

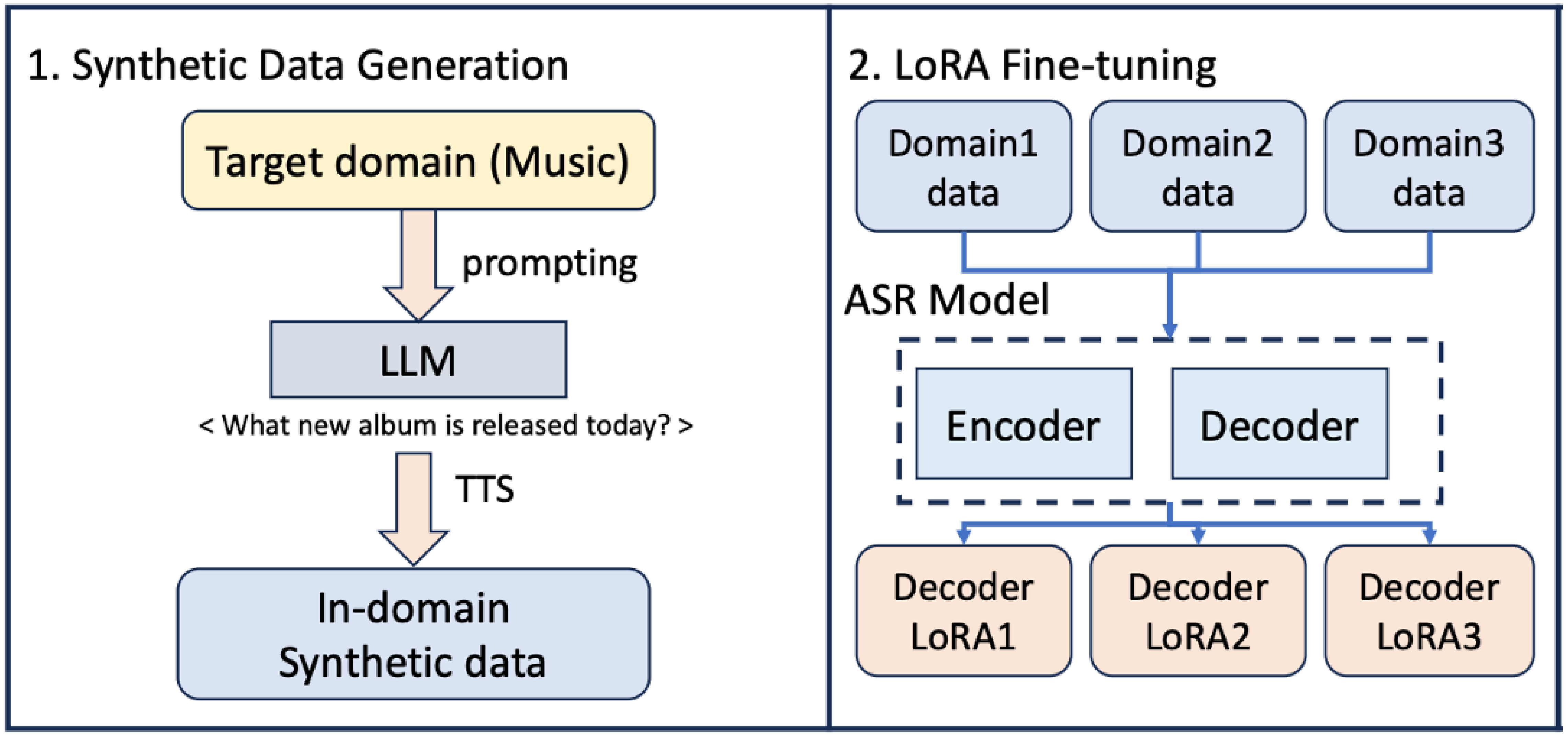
Problem Statement

- **Given a pretrained ASR model (Whisper in this work), how can we adapt the model weights to:**
  - Perform better (lower WER) on some language-defined domains.
  - No access to real speech training data (except for the test sets used for evaluation)
  - No performance regression on out-of-domain data
- **Language-defined domain: Speech utterances with content relating to a domain. For example:**
  - Sports: Where was the world cup help in 2016?
  - Weather: How is the weather in Seattle today?

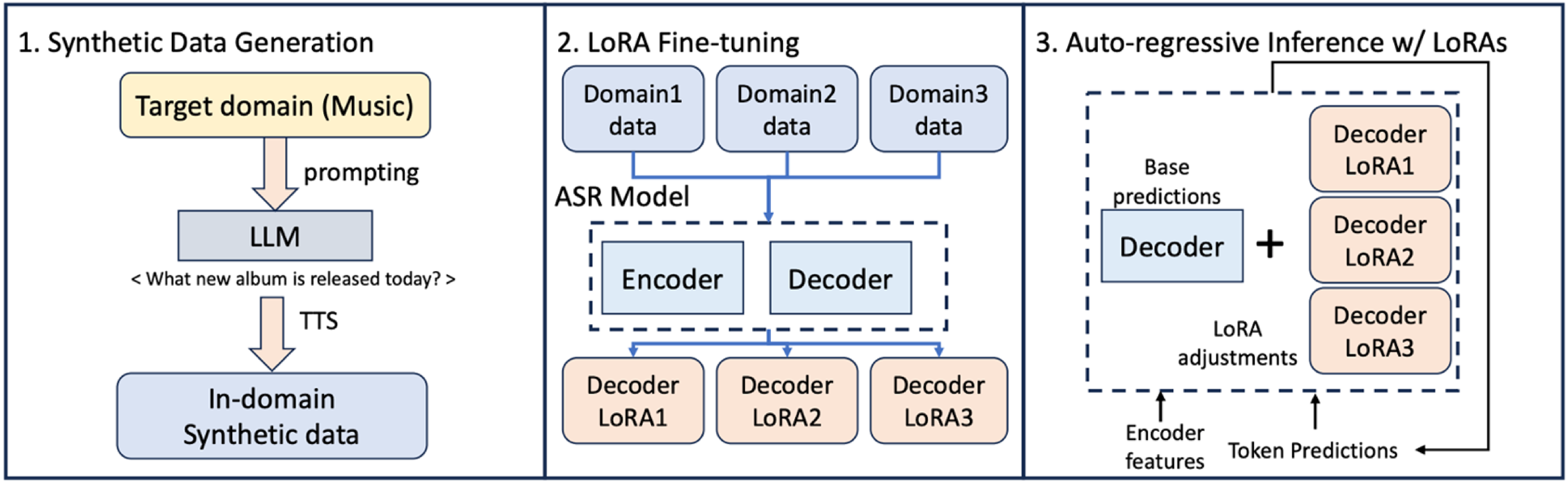
Proposed Method Overview



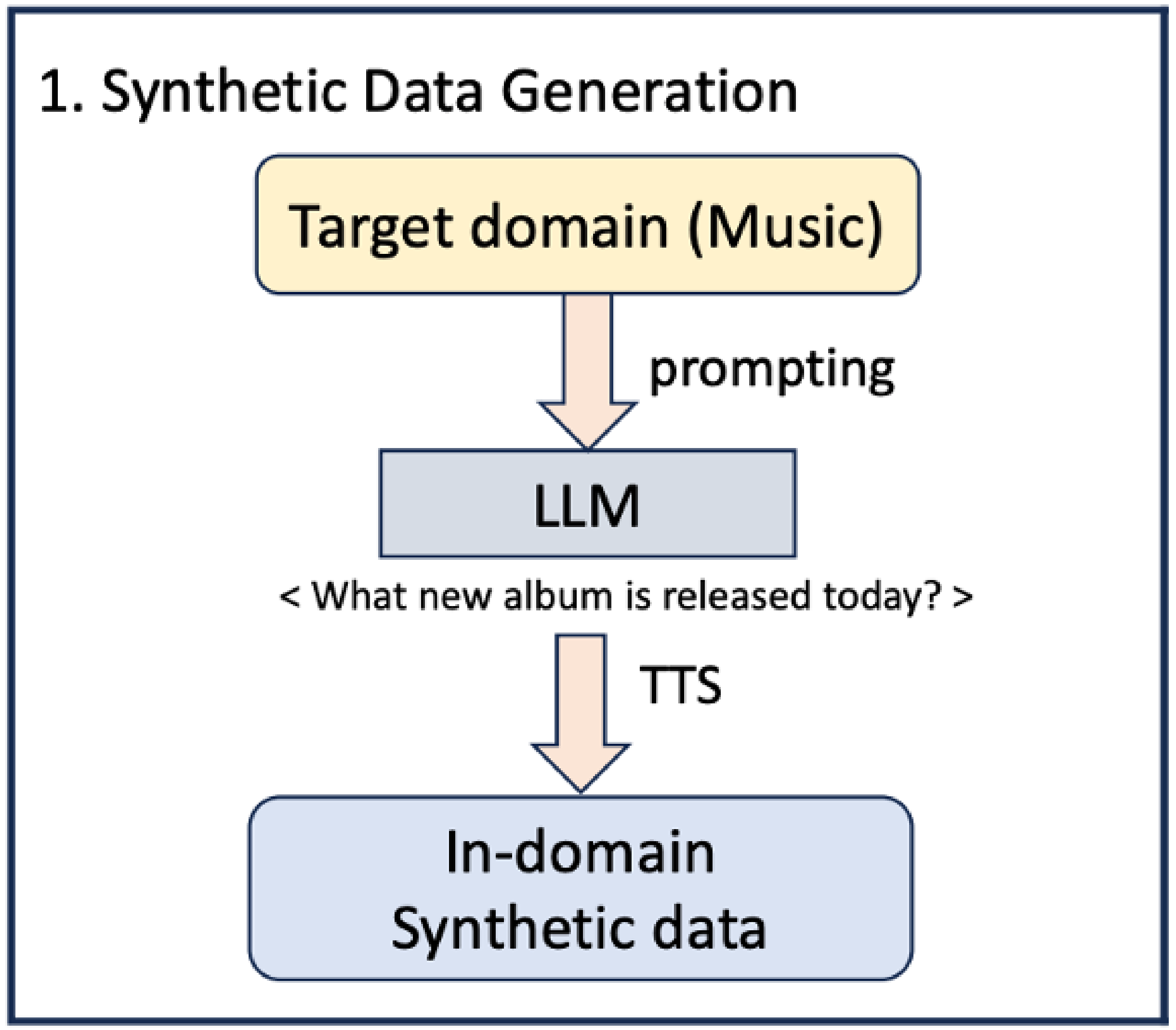
Proposed Method Overview



Proposed Method Overview



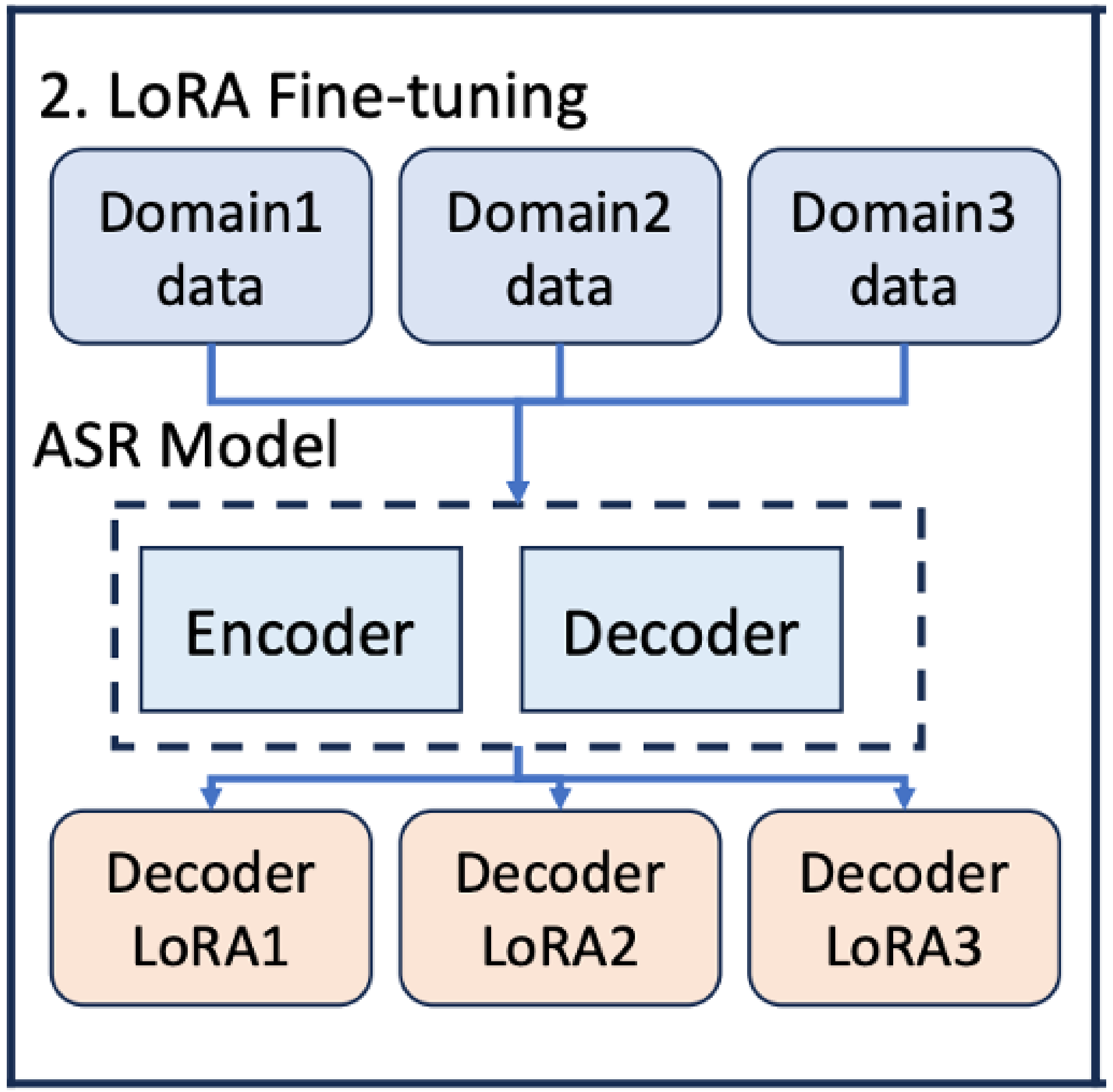
Stage 1: Synthetic text generation



- We prompt LLM (Llama3-70B) to generate large amount of synthetic text for a specific domain
- We use Codec [1] text generation pipeline
- Advantage: No seed text data needed (as opposed to [2])
- We feed our generated text data into a TTS system [3] to create paired text-audio data for ASR

[1] Zheng et al. CodecLM: Aligning Language Models with Tailored Synthetic Data. Findings of ACL 2024.
[2] Huang et al. Text Generation with Speech Synthesis for ASR Data Augmentation. Arxiv 2023.
[3] Wu et al. Transformer-based acoustic modeling for streaming speech synthesis. INTERSPEECH 2021.

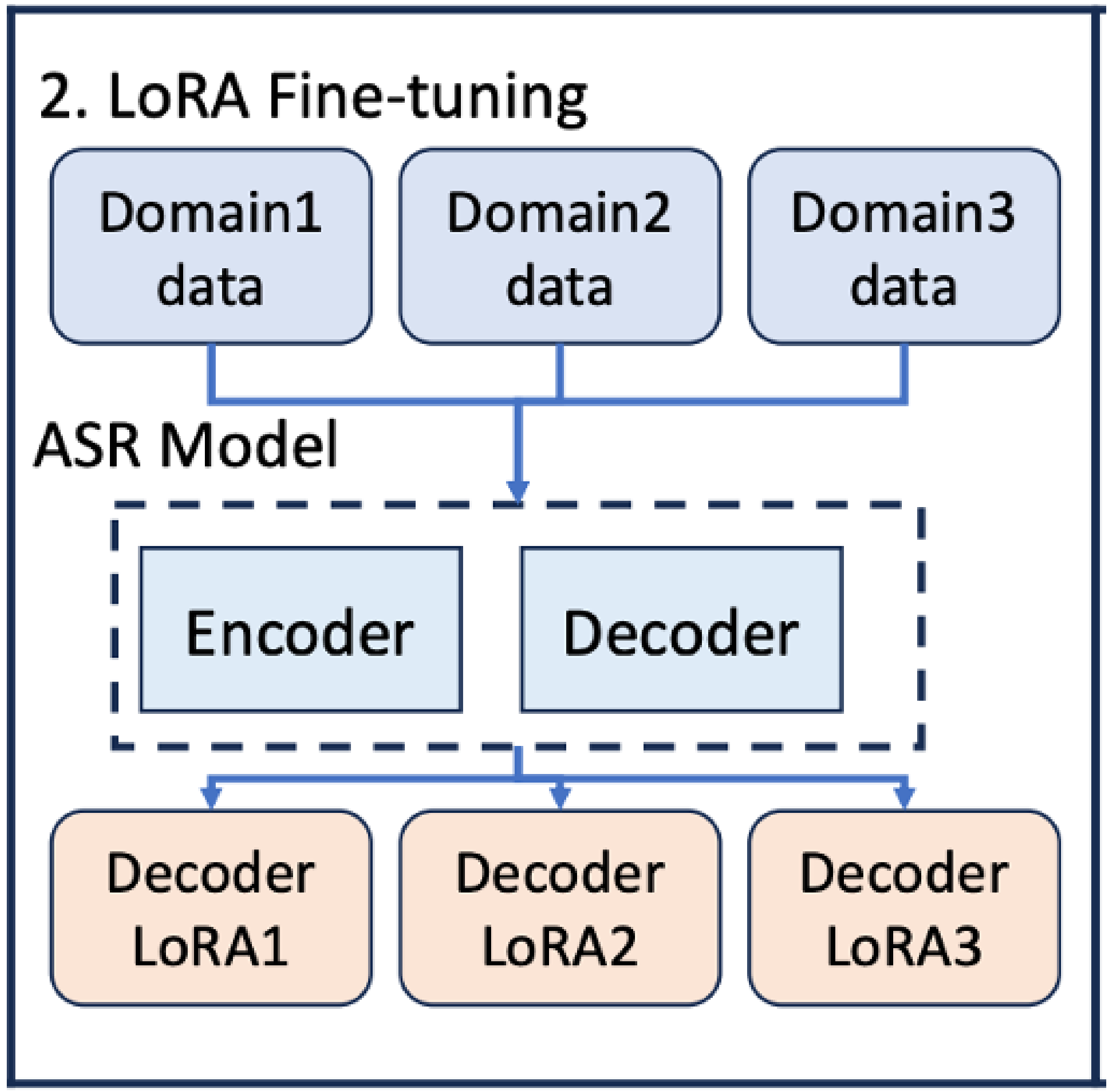
Stage 2: Model tuning on synthetic data



- We use the synthetic data in stage 1 to finetune Whisper [4]. Some experimentally verified observations (refer paper):
  - It is better to tune only the decoder instead of the whole model (encoder + decoder)
  - It is better to tune with LoRA [5] adapters instead of full fine-tuning
  - Advantage: LoRA adapters are efficient in both runtime and memory usage
- We train one LoRA adapter for each domain – using corresponding synthetic data generated in Stage 1

[4] Radford et al. Robust speech recognition via large-scale weak supervision. ICML 2023.
[5] Hu et al. LoRA: Low-Rank Adaptation of Large Language Models. ICLR 2022.

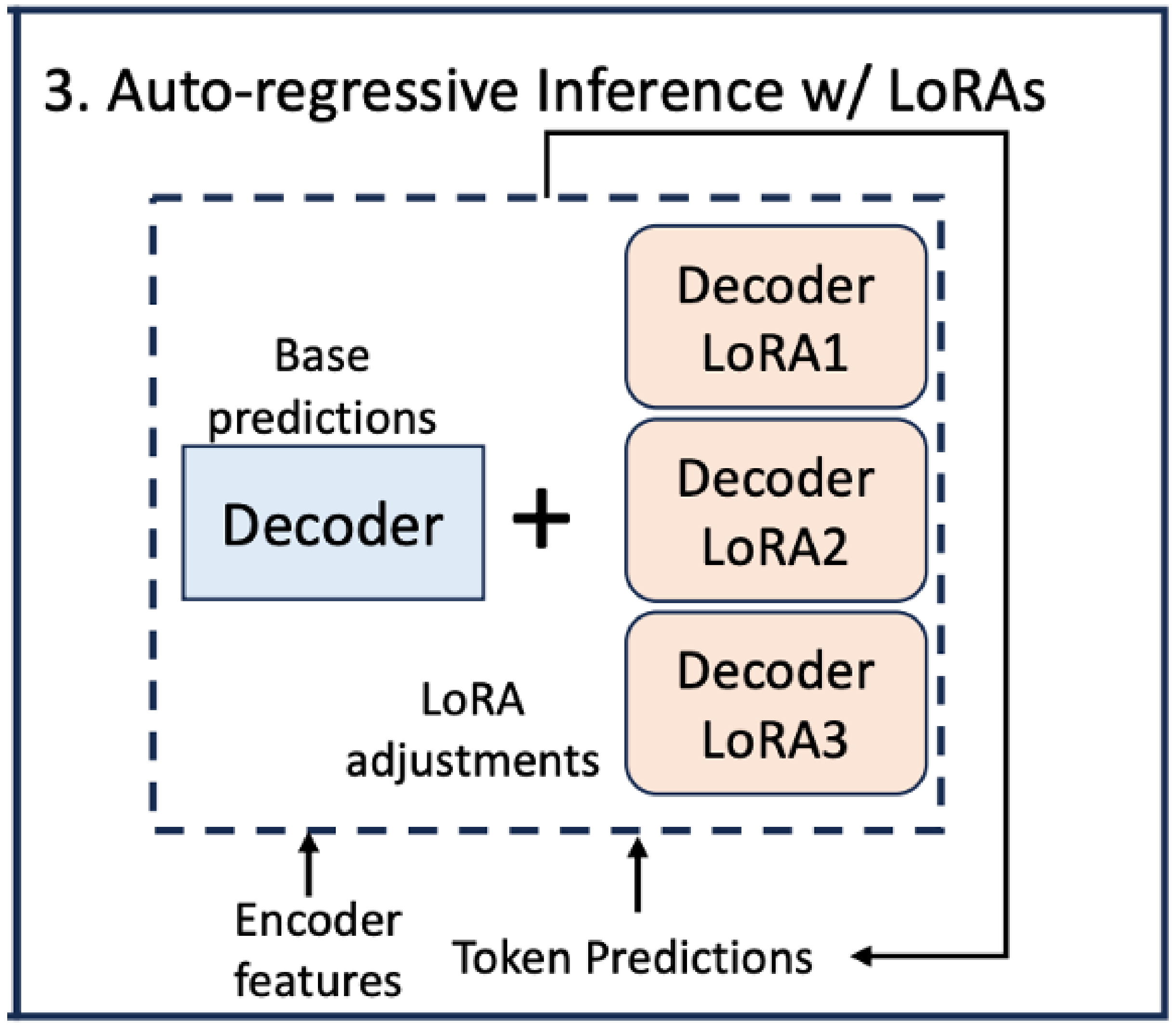
Stage 2: Model tuning on synthetic data



- We use the synthetic data in stage 1 to finetune Whisper [4]. Some experimentally verified observations (refer paper):
  - It is better to tune only the decoder instead of the whole model (encoder + decoder)
  - It is better to tune with LoRA [5] adapters instead of full fine-tuning
  - Advantage: LoRA adapters are efficient in both runtime and memory usage
- We train one LoRA adapter for each domain – using corresponding synthetic data generated in Stage 1

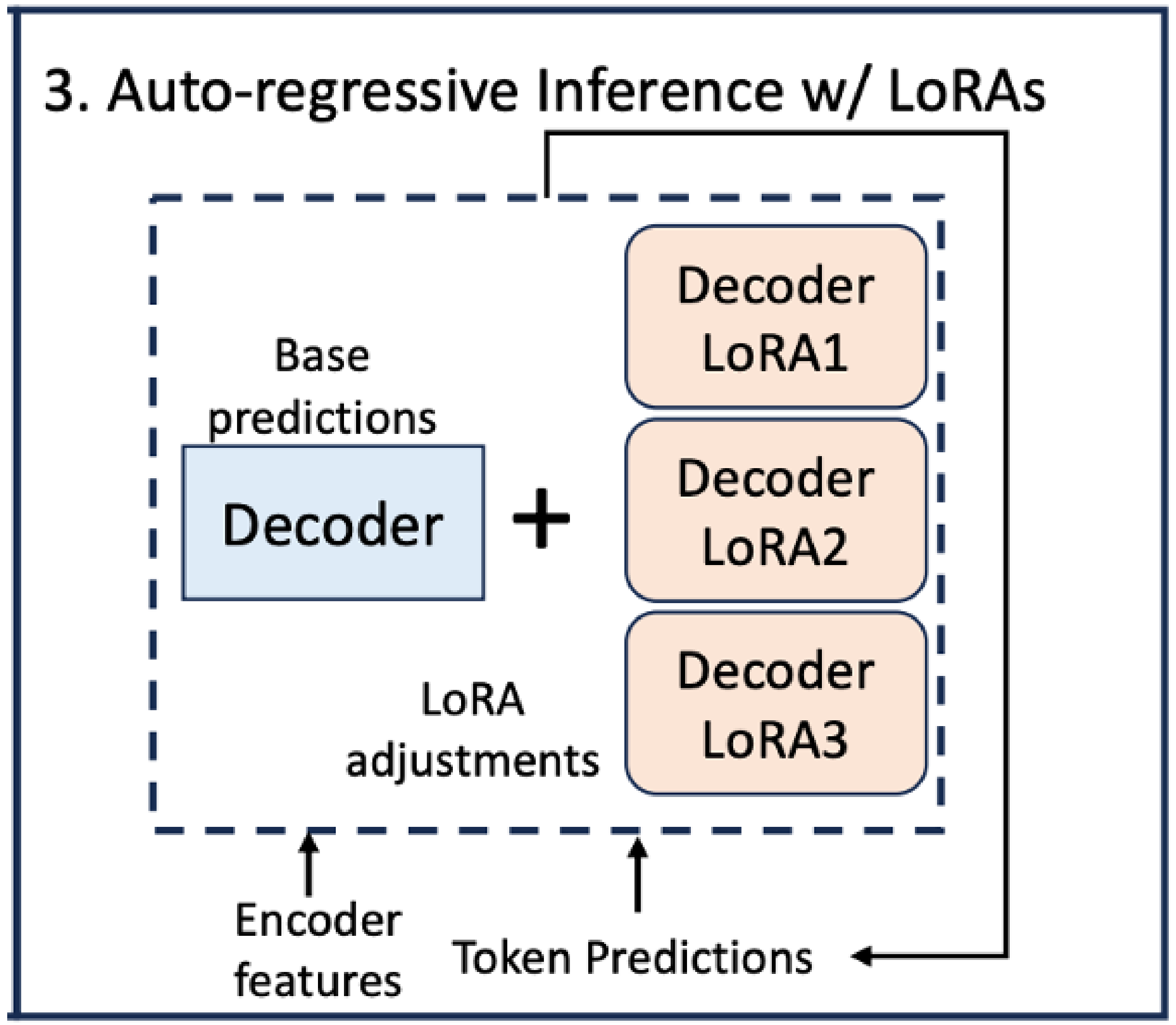
[4] Radford et al. Robust speech recognition via large-scale weak supervision. ICML 2023.
[5] Hu et al. LoRA: Low-Rank Adaptation of Large Language Models. ICLR 2022.

Stage 3: Inference with multiple adapters



- From stage 2, we have multiple LoRA adapters; one adapter per domain.
- During inference, how can we efficiently process an utterance without prior domain knowledge?
  - **Solution 1:** Original model → text (domain) classifier → select corresponding LoRA adapter
    - 2 passes
    - Cannot extend to new domain (need to re-train the text classifier)
  - **Solution 2:** Generate speech transcription with each LoRA adapter, then select the transcription with highest confidence (avg. predicted token probabilities)
    - Slow: k adapters -> k passes
    - Can extend to new domain

Stage 3: Inference with multiple adapters



- From stage 2, we have multiple LoRA adapters; one adapter per domain.
- During inference, how can we <u>efficiently</u> process an utterance <u>without prior domain knowledge</u>?
  - **Solution 1:** Original model → text (domain) classifier → select corresponding LoRA adapter
    - 2 passes
    - Cannot extend to new domain (need to re-train the text classifier)
  - **Solution 2:** Generate speech transcription with each LoRA adapter, then select the transcription with highest confidence (avg. predicted token probabilities)
    - Slow: k adapters -> k passes
    - Can extend to new domain

Stage 3: Auto-regressive decoding with LoRAs

Algorithm 1 Auto-regressive decoding with multiple LoRAs

Require: W, $\{(A_i, B_i)\}$ for $i \in [k]$, x: encoder features

$tokens \leftarrow []$
while $[eos] \notin tokens$ **do**
    h = $Softmax(W(x, tokens))$
    $(\text{next}_0, \text{c}_0) = \text{Argmax}(h), \text{Max}(h)$ ▷ c denotes the confidence
    $h_i = \text{Softmax}((W + B_i A_i)(x, tokens))$ for $i \in [k]$
    $(\text{next}_i, \text{c}_i) = \text{Argmax}(h_i), \text{Max}(h_i)$ for $i \in [k]$
    SELECT next from $\{\text{next}_0, \text{next}_1, \ldots, \text{next}_k\}$
    INSERT next to $tokens$
end while
return $tokens$

- Gist:
    - Generate one token at a time (in an autoregressive manner)
    - For each token, generate all tokens predicted by each LoRA adapter
    - We use the confidence level of each token to select the best one

Experiment settings

- **Dataset**
  - We evaluate on three domains: music, weather, sports
- **Validation data**
  - Real speech samples collected via Meta RayBan glasses
  - Manually categorized into each of the three domain

	music	weather	sports
Synthetic dataset	44K	31K	46K
Evaluation dataset	2.1K	2.8K	5.1K

Number of samples for each domain on train/test sets

Experiment settings

- **Evaluation metric:**
  - Word Error Rate without wake words (e.g., Hey Meta)
- **Baselines**
  - FT: full fine-tuning (decoder) (for each domain)
  - LoRA-ft: fine-tuning (decoder) with LoRA (for each domain)
  - FT-Multi: full fine-tuning (decoder) on 3 domain synthetic data combined
  - LoRA-ft-Multi: fine-tuning (decoder) with LoRA on 3 domain synthetic data combined

Results

	Train set	music	weather	sports
Original	-	27.94	14.97	15.59
FT	TTS-Music	**23.20** (↑ 17.0%)	14.45 (↑ 3.5%)	20.1 (↓ 28.8%)
FT	TTS-Weather	33.05 (↓ 18.3%)	**12.10** (↑ 19.2%)	17.7 (↓ 13.5%)
FT	TTS-Sports	25.05 (↓ 10.3%)	15.96 (↓ 6.6%)	**15.3** (↑ 1.9%)
LoRA-ft	TTS-Music	**23.23** (↑ 16.8%)	13.27 (↑ 11.3%)	16.51 (↓ 5.9%)
LoRA-ft	TTS-Weather	26.65 (↑ 4.6%)	**11.70** (↑ 21.8%)	15.08 (↑ 3.3%)
LoRA-ft	TTS-Sports	27.14 (↑ 2.9%)	14.05 (↑ 6.1%)	**13.37** (↑ 14.2%)
FT-Multi	TTS(M+W+S)	**24.71** (↑ 11.6%)	24.53 (↓ 64.0%)	15.84 (↓ 1.6%)
LoRA-ft-Multi	TTS(M+W+S)	25.09 (↑ 10.2%)	13.70 (↑ 8.4%)	14.61 (↑ 6.3%)
DAS	TTS(M/W/S)	24.87 (↑ 11.0%)	**12.39** (↑ 17.2%)	**13.98** (↑ 10.3%)

- FT and LoRA-ft: one model for each domain
- FT-Multi/LoRA-ft-Multi/DAS (ours): a single model for all domains
- DAS is the only method that can extend to new domains (without retraining)
  - only need to train new LoRA adapter and attach to the model

Our method can maintain (most) improvements across all three domains

Out-of-domain regression experiment

	OOD_1	OOD_2	OOD_3	OOD_4
Original	12.02	5.04	10.87	10.36
LoRA-ft Multi	13.79	5.78	11.48	10.82
DAS	12.25	5.1	11.06	10.29
% change	-1.02	-1.01	-1.02	+0.99

TABLE V

ASR PERFORMANCE COMPARISON BETWEEN DAS AND ORIGINAL (UNADAPTED) MODEL ACROSS FOUR OUT-OF-DOMAIN TEST SETS. OOD_1: LIBRISPEECH TEST-OTHER, OOD_2: LIBRISPEECH TEST-CLEAN, OOD_3: FLEURS-EN, OOD_4: VOXPOPULI-EN.

Our method shows minimal out-of-domain performance regression.

Conclusion

- We propose a novel framework for ASR systems that can
  - Improve WER for a set of target language-defined domains.
  - Minimal generalizability loss.
  - No real data needed.

Feel free to refer to our paper to more details.

Look forward to related paper on representation learning (encoder) with synthetic TTS data at Interspeech 2025.

2025 ICASSP

HYDERABAD, INDIA

Celebrating Signal Processing

GOLDEN JUBILEE 50

Meta