Mitigating Proxy-to-Wild Domain Gap in Deepfake Speech

Xuanjun Chen, Yun-Shing Wu, Wei-Chung Lu, Claire Lin, Haibin Wu,
Hung-yi Lee, Jyh-Shing Roger Jang

National Taiwan University · NTU AI-CoRE

ISCSLP 2026 · Selected Outstanding Paper

arxiv.org/abs/2606.07494

From conventional deepfakes to CodecFake

BEFORE

Conventional deepfakes

- Text-to-speech (TTS) and voice conversion (VC)
- Waveforms synthesized by vocoders
- Benchmarks: ASVspoof

NOW

Codec-based speech generation (CoSG)

- More and more TTS backbones build on neural audio codecs
- Speech decoded from discrete codec tokens
- Zero-shot voice cloning from a few seconds

CodecFake detection = detecting fake speech produced by CoSG systems

Two challenges for CodecFake detection

1 Poor generalization

- Countermeasure (CM) models trained on standard datasets (e.g., ASVspoof) fail on unseen CoSG data

0.12% EER → **18.92 / 32.06% EER**

ASVspoof19 (in-domain) — CoSG Eval / CoSG ExtEval

Same ASVspoof19-trained Wav2Vec2-AASIST detector

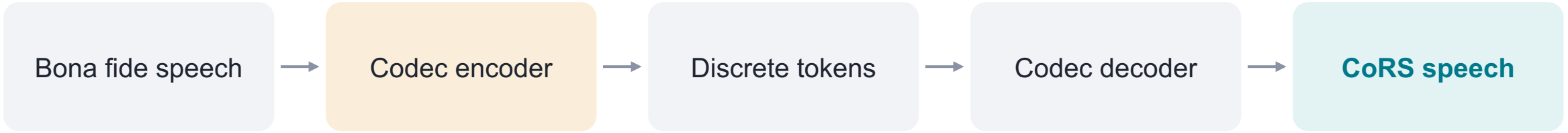
2 High data collection costs

- Every emerging CoSG system would need its own training data
- Collecting it for each one is expensive and impractical

So we need: training data that is cheap to get, yet still generalizes to new CoSG systems

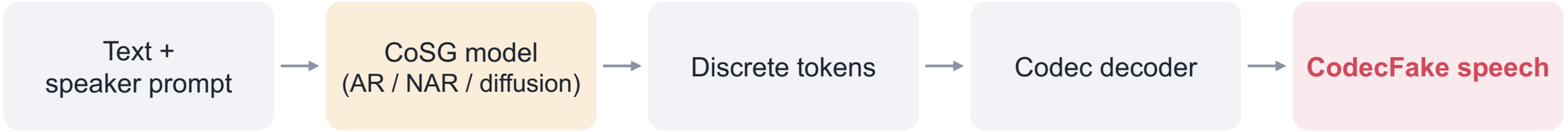
Our previous work: CoRS as proxy training data

Training (proxy): CoRS · CodecFake (INTERSPEECH 2024), CodecFake+ (IEEE TASLP 2026)



Both decode discrete codec tokens → shared reconstruction artifacts

Testing (in the wild)



Why CoRS helps

- Shares codec reconstruction artifacts with real CodecFake speech
- Cheap and scalable: any real speech through any neural codec (CodecFake+: 31 codecs on VCTK)

But detectors easily overfit to CoRS

SETUP

- Same Wav2Vec2-AASIST detector, trained on ASVspoof 2019 or on CoRS from the top-3 codecs
- Tested on CoSG Eval (17 unseen models) and CoSG ExtEval (40 unseen models)

CoSG Eval · EER

18.92 → 14.09% **(better)**

Trained on ASVspoof → CoRS

CoRS helps on the familiar test set…

CoSG ExtEval · EER

32.06 → 38.93% **(worse)**

Trained on ASVspoof → CoRS

…but hurts on unseen CoSG models

Why

- CMs latch onto codec- and dataset-specific artifacts of the proxy
- Real CoSG fakes come from full generation models (AR / NAR / diffusion) with unseen prompts and content

Proxy-to-wild domain gap = mismatch between proxy training data and in-the-wild CodecFake speech

Motivation: The Proxy-to-Wild Domain Gap

Why these three: many prior studies show these three factors can hurt the generalization of CMs

1 Artifact mismatch

- New codecs and generators leave new signatures
- Never seen in the proxy training data

Studied in: Chen et al., INTERSPEECH 2025

2 Silence mismatch

- Pauses at clip boundaries differ across datasets
- Models overfit to noise in silent parts

Studied in: Zhang et al., TASLP 2023 · Chen et al., IEEE ASRU 2025

3 Content & speaker mismatch

- Unseen phonetic and prosodic patterns
- Unseen speakers
- Fall outside the learned distribution

Studied in: Chen et al., IEEE ASRU 2025

Problem: we can't collect proxy data for every unseen codec, pause pattern, or speaker

Idea: the gap shows up as shifts in feature statistics → simulate such shifts while training the CM

Our contributions

1 Domain-Shift Feature Augmentation (DSFA) targets the gap

- Turns fixed feature statistics (μ, σ) into distributions and samples shifted “styles”
- Training only, on a post-trained SSL backbone; no extra parameters

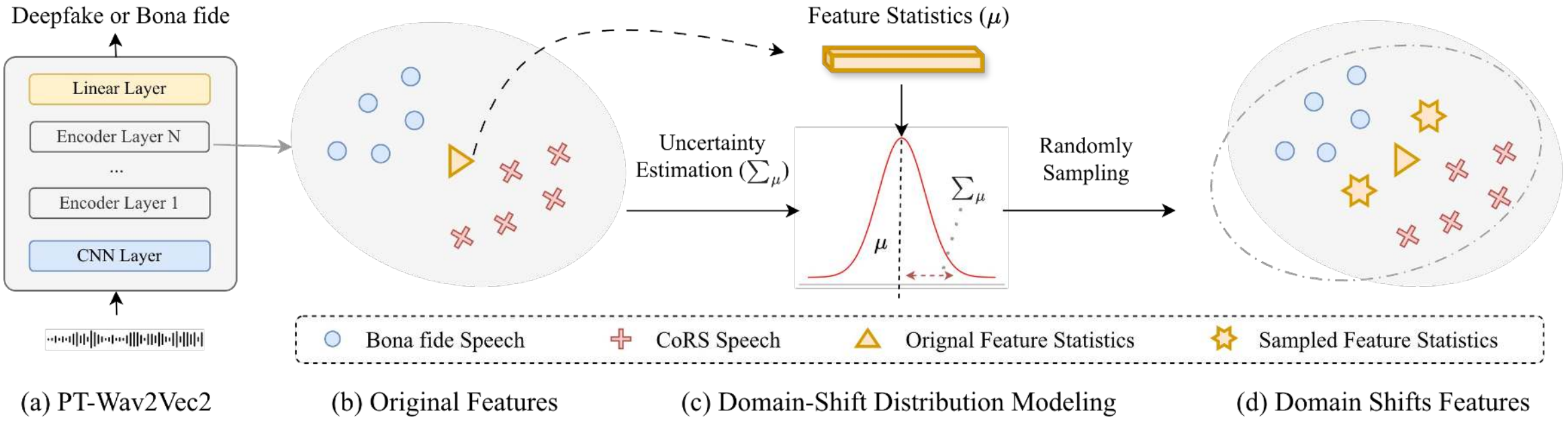
2 CoSG ExtEval measures the gap

- 40 unseen CoSG models, four generation paradigms
- Long-form audio: clips up to 149 s

3 Best EER on both test sets

- 2.78% EER on CoSG Eval, 21.80% on CoSG ExtEval
- Best CodecFake+ benchmark: 11.91% / 27.07%

Framework overview



Backbone

Wav2Vec2 post-trained on deepfake data, fine-tuned on CoRS

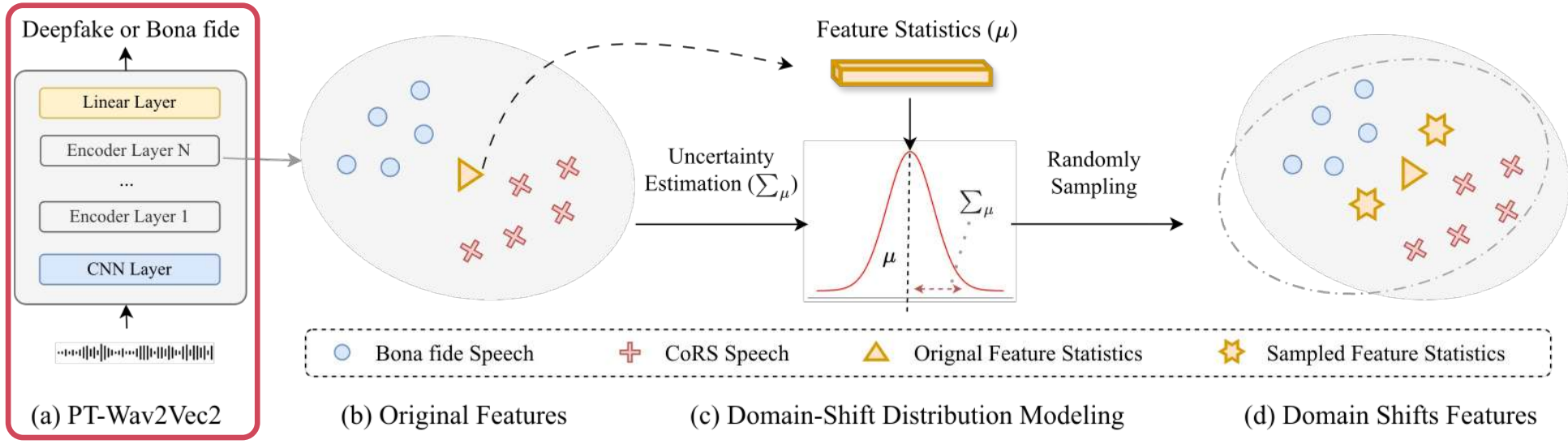
Domain-Shift Feature Augmentation (DSFA, ours)

Spirit
Simulate unseen domains by shifting feature statistics

Input
SSL features of a CoRS training batch

Output
Re-styled features: same clip and label, shifted domain

① Post-trained SSL backbone



Start from a deepfake-aware backbone

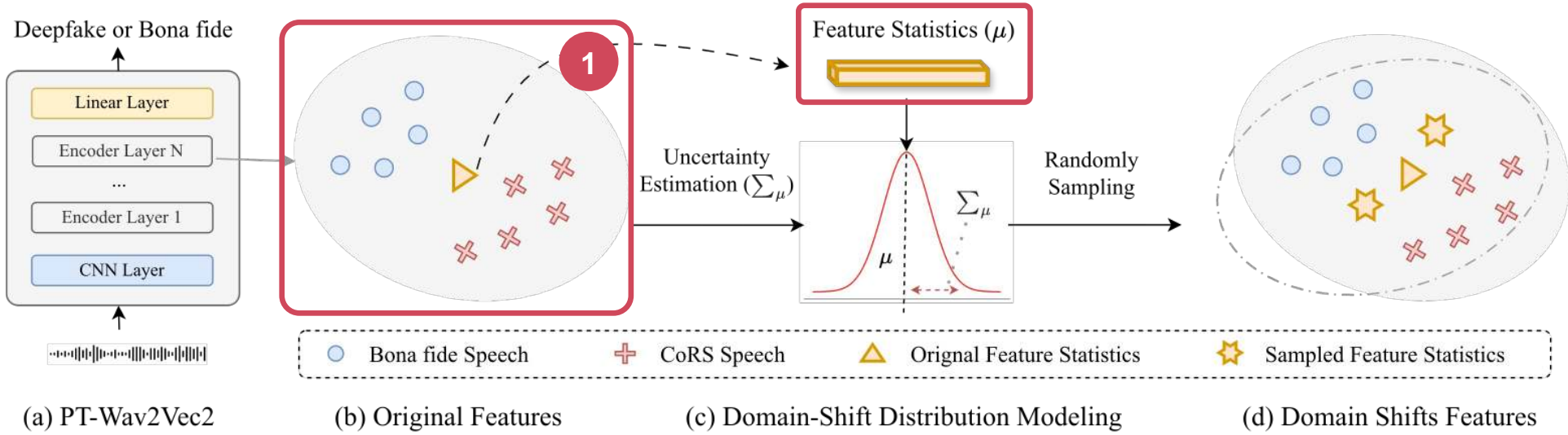
- Wav2Vec2 AntiDeepfake (NII): SSL model post-trained for deepfake detection on a large, mixed corpus
- Already sensitive to deepfake artifacts and to diverse speakers and codecs
- We fine-tune it on CoRS with a linear head

What the backbone brings

- Sensitive to deepfake artifacts
- Robust across speakers and codec signatures
- A stable starting point for fine-tuning with DSFA

Results: see Experiments

② Domain-Shift Feature Augmentation (DSFA)



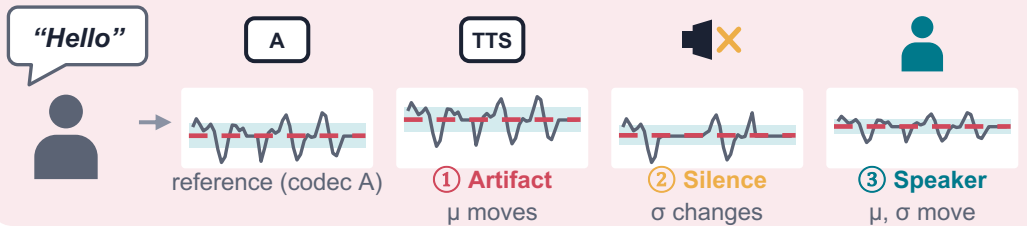
1 Feature statistics describe the domain

- **Inspired by:** AdaIN uses μ, σ as a feature map's "style"
  - *Huang & Belongie, ICCV 2017*
- **Why μ, σ:** time-averaging removes content, keeps style
  - **μ:** constant coloration (generator, speaker, noise floor)
  - **σ:** variation over time (silence, loudness, prosody)
- **So: shift μ, σ → same content & label, new domain**

$$\mu_{b,c} = \mathbb{E}_t[x_{b,c,t}], \quad \sigma^2_{b,c} = \mathbb{E}_t\left[(x_{b,c,t} - \mu_{b,c})^2\right]$$

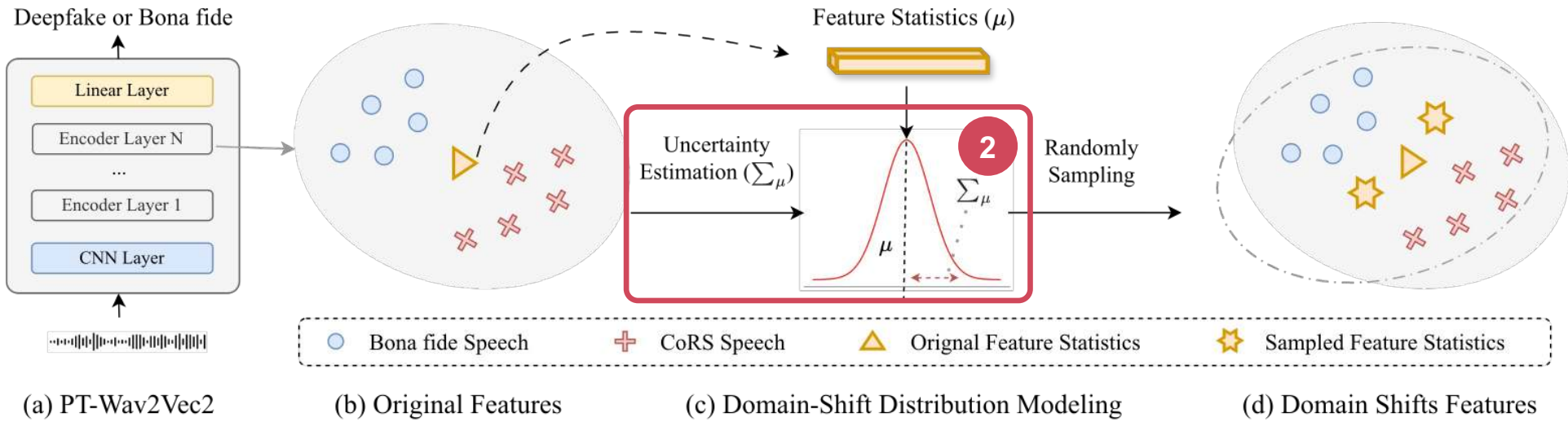
In words: *μ = the clip's average level, σ = how much it varies over time*

e.g. same words, different domain factors



Evidence · Fig. 2 CoRS vs. CoSG overlap: μ 42.91% · σ 65.01%

② Domain-Shift Feature Augmentation (DSFA)



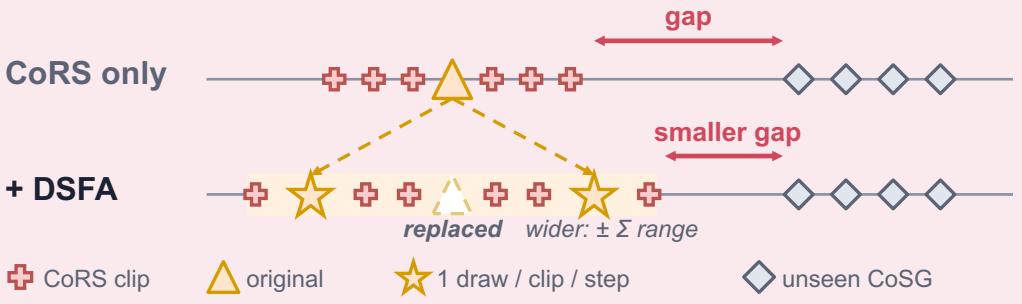
2 The batch sets how far to shift

- **Goal:** **extend CoRS styles beyond the proxy, toward CoSG**
- **Inspired by:** DSU treats feature statistics as uncertain
  - *Li et al., ICLR 2022*
- **Idea:** μ, σ can move; how far = their batch spread Σ
  - **large Σ:** channel varies across clips → shift it more
  - **small Σ:** stable channel → shift it less
- **So:** **shift size comes from data, not a hand-set noise level**

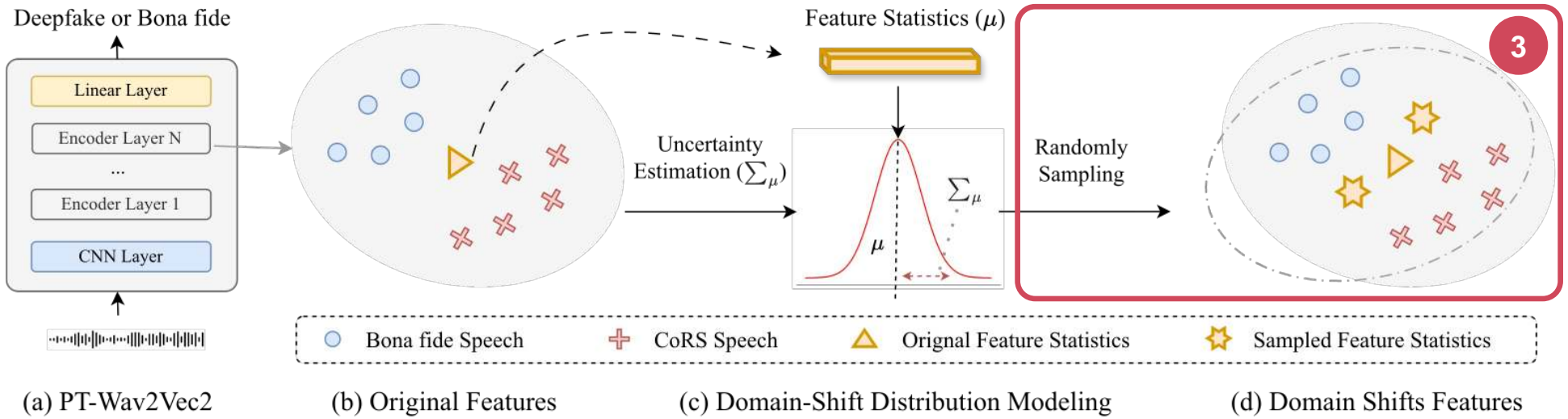
$$\Sigma_\mu^2 = \frac{1}{B}\sum_{i=1}^{B}\left(\mu(x_i) - \mathbb{E}_b[\mu]\right)^2, \quad \Sigma_\sigma^2 = \frac{1}{B}\sum_{i=1}^{B}\left(\sigma(x_i) - \mathbb{E}_b[\sigma]\right)^2$$

In words: *Σ = how much the clips in this batch differ from each other*

e.g. μ of one channel: CoRS vs. unseen CoSG



② Domain-Shift Feature Augmentation (DSFA)



3 Sample a new style, keep the content

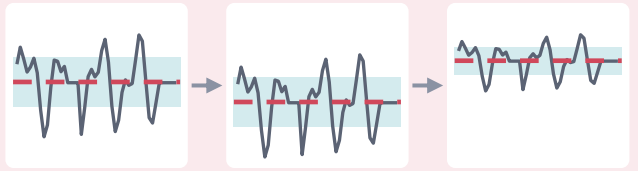
- **Goal: re-styled CoRS clips reach toward CoSG**
- **Sample β:** β = μ + ε × Σ (ε uniform or Gaussian)
- **Sample γ:** γ = σ + ε × Σ (ε uniform or Gaussian)
- **Re-style (AdaIN):** remove own style (μ, σ), apply new one (γ, β)
  - **content:** the normalized pattern, unchanged
  - **label:** unchanged → CM must use cues that survive the shift
- **Mix: apply with probability p, so clean features are still seen**

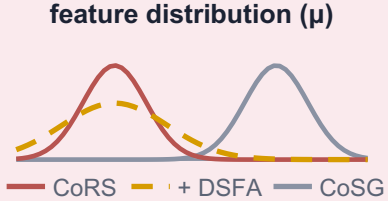
$$\beta = \mu + \epsilon_\mu \Sigma_\mu, \quad \gamma = \sigma + \epsilon_\sigma \Sigma_\sigma, \quad \epsilon \sim \mathcal{U}(-1,1) \text{ or } \mathcal{N}(0,1)$$

$$\mathrm{DSFA}(x) = \gamma \frac{x-\mu}{\sigma} + \beta, \qquad \hat{x} = \begin{cases} \mathrm{DSFA}(x) & \text{w.p. } p \\ x & \text{otherwise} \end{cases}$$

In words: *new style = own style + noise × batch spread, then swap it in*

e.g. one CoRS clip, re-styled



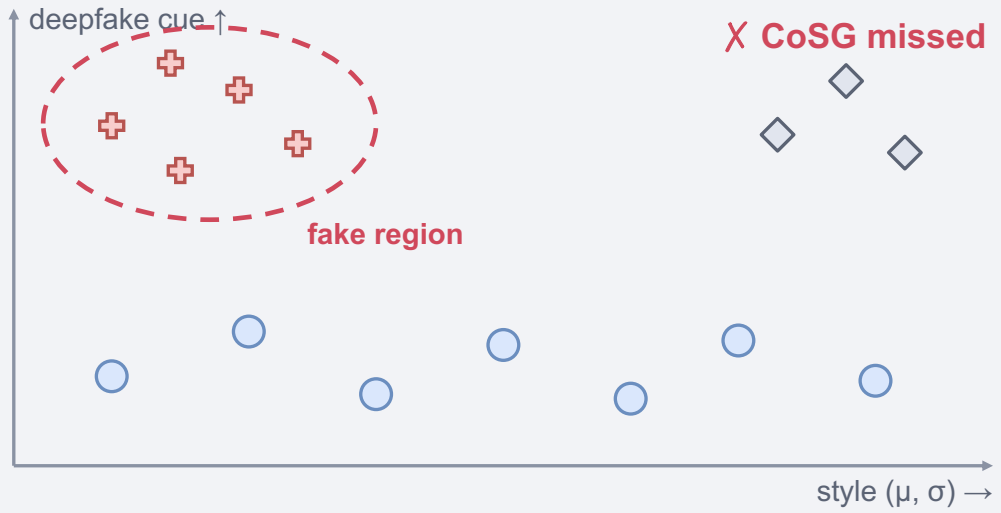


→ wider, tail nears CoSG

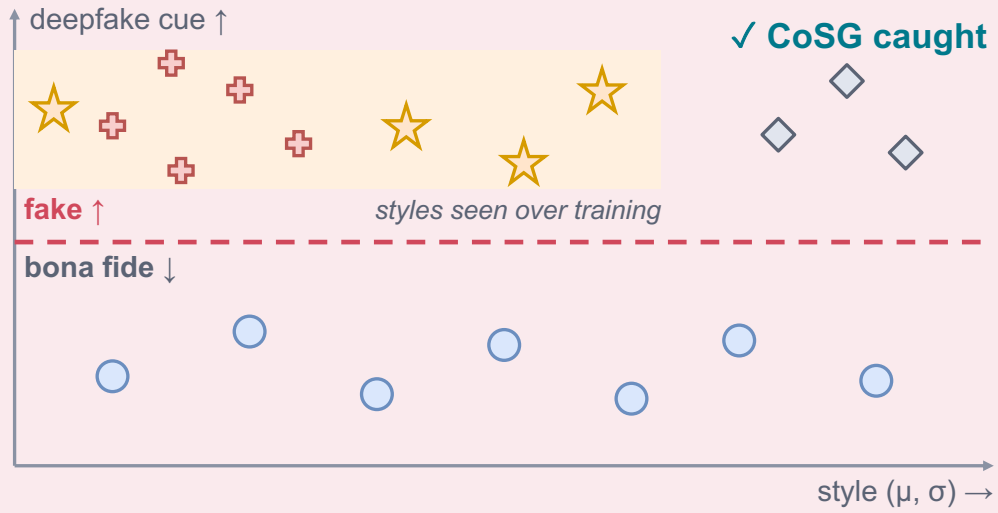
② Why DSFA helps CM improve its generalizability

Many styles, same label → the CM can't use proxy style as a shortcut; it must learn the deepfake cue

Baseline: the boundary follows CoRS style



+ DSFA: the boundary follows the deepfake cue



bona fide · CoRS (fake) · sampled stats (over steps) · unseen CoSG (fake)

Steps: ① μ, σ = domain → ② Σ sets the range → ③ re-style, keep the label

Training only · no extra parameters · inference unchanged → **tested on CoSG ExtEval and in feature space (Fig. 2)**

③ Training objective

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda \mathcal{L}_{SupCon}$$

L_{CE}

Cross-entropy loss

- Bona fide vs. spoofed
- Class weights 0.1 / 0.9 to emphasize bona fide samples
- Used in every model

$\lambda \cdot L_{SupCon}$

Supervised contrastive loss

- Pulls same-class embeddings together, pushes the other class apart
- Optional, weighted by λ
- Helps CoSG Eval, hurts ExtEval (22.19 → 24.08)

DSFA

Training-only augmentation

- Active only in fine-tuning, with probability p
- No extra parameters
- Inference unchanged: backbone + linear head

Fine-tuned end-to-end · Adam, lr 10^{-6}, weight decay 10^{-4} · batch 14 · 4 s crops + RawBoost · one RTX 3090

④ CoSG ExtEval: a harder in-the-wild test set

Eval set	Type	#Utt.	Models	Hours	Min / Mean / Max (s)
CoSG Eval	Bona fide	850	—	1.69	1.27 / 6.55 / 28.58
	Spoofed	931	17	1.32	0.85 / 5.51 / 16.63
CoSG ExtEval	Bona fide	1,222	—	1.96	0.81 / 5.78 / 139.67
	Spoofed	1,366	**40**	3.08	**0.80 / 8.13 / 149.38**

- Collected from official demo pages and public repositories of recent CoSG systems
- No overlap with training data
- Release planned on GitHub

40 unseen CoSG models (vs. 17 in CoSG Eval)

4 paradigms: AR · NAR · diffusion · multi-stage

149 s longest clip (vs. 29 s) — long-form audio

Experimental setup

1 Training data

- CodecFake+ CoRS: 31 neural codecs on VCTK
- DEC-balanced subset (time / frequency decoders)
- 42,965 bona fide + 42,965 spoofed

2 Backbone

- Post-trained Wav2Vec2 AntiDeepfake (NII Yamagishi Lab)
- Linear classification head

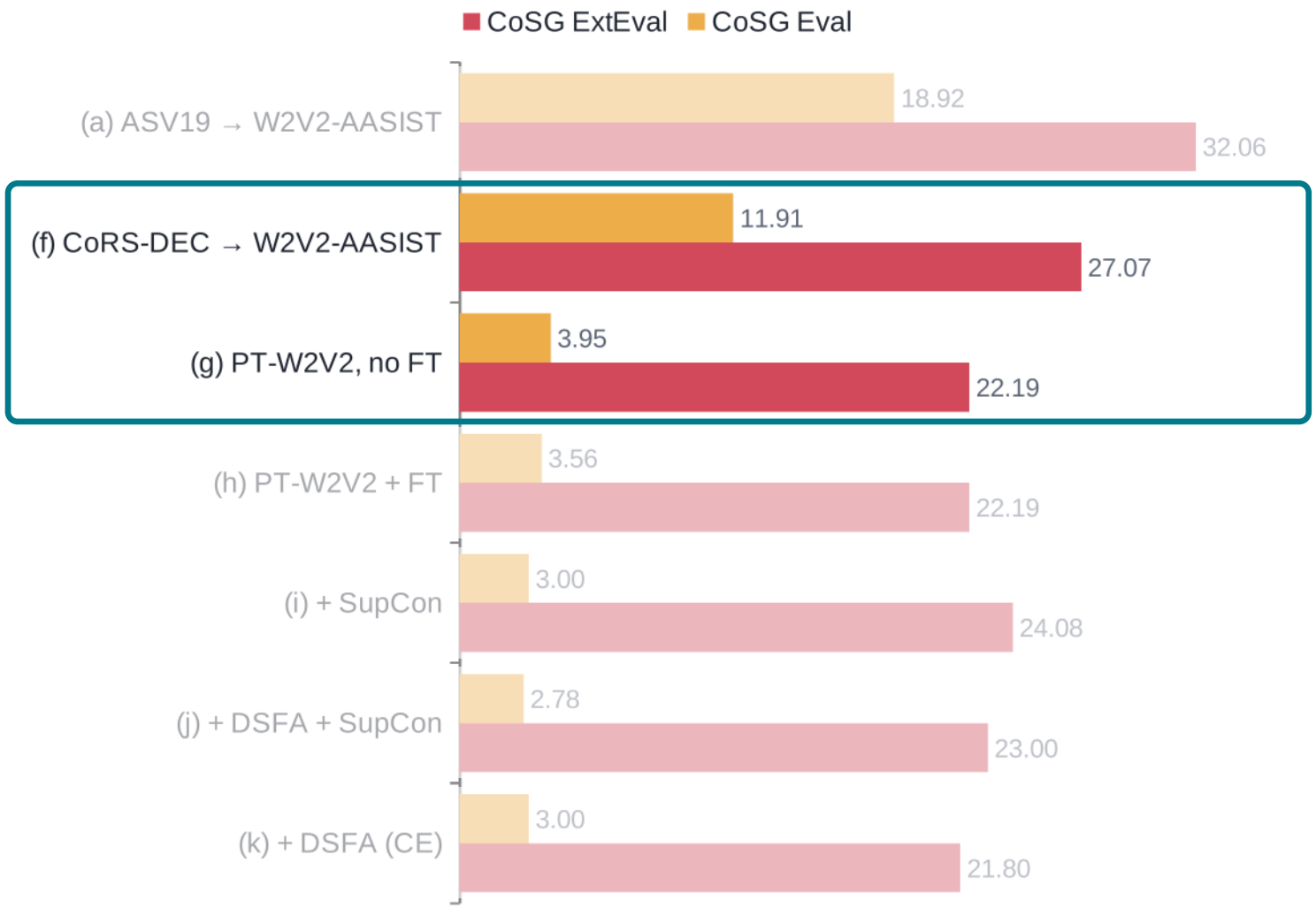
3 Training

- 16 kHz raw waveform, 4 s crops, RawBoost
- Adam, lr 10^{-6}, batch 14
- Weighted CE (0.1 / 0.9) · one RTX 3090

4 Evaluation

- Metric: EER (%)
- ASVspoof 2019 LA eval
- CoSG Eval (17 models) · CoSG ExtEval (40 models)

Main results: EER (%) on CodecFake attacks



SETUP

- (f) W2V2-AASIST, trained on CoRS (DEC)
- (g) post-trained W2V2, no fine-tuning
- Same test sets: CoSG Eval / ExtEval

1 Backbone: the biggest jump

CoSG Eval

11.91 → 3.95% (better)

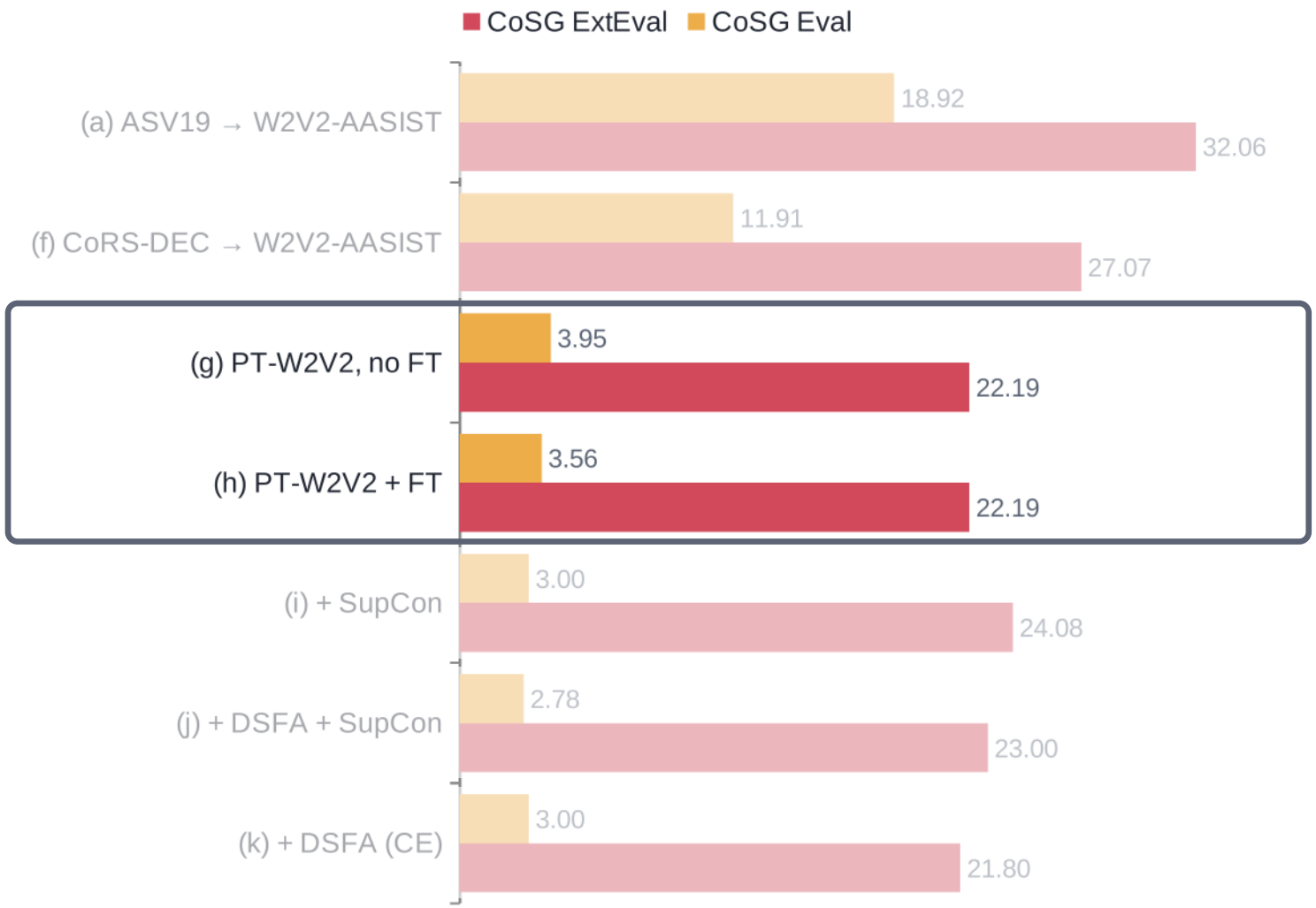
CoSG ExtEval

27.07 → 22.19% (better)

A deepfake-aware backbone matters most

EER: lower is better

Main results: EER (%) on CodecFake attacks



SETUP

- (g) post-trained W2V2, no fine-tuning
- (h) same backbone, fine-tuned on CoRS (CE)
- Same test sets: CoSG Eval / ExtEval

2 **CoRS fine-tuning: Eval only**

CoSG Eval

3.95 → 3.56% **(better)**

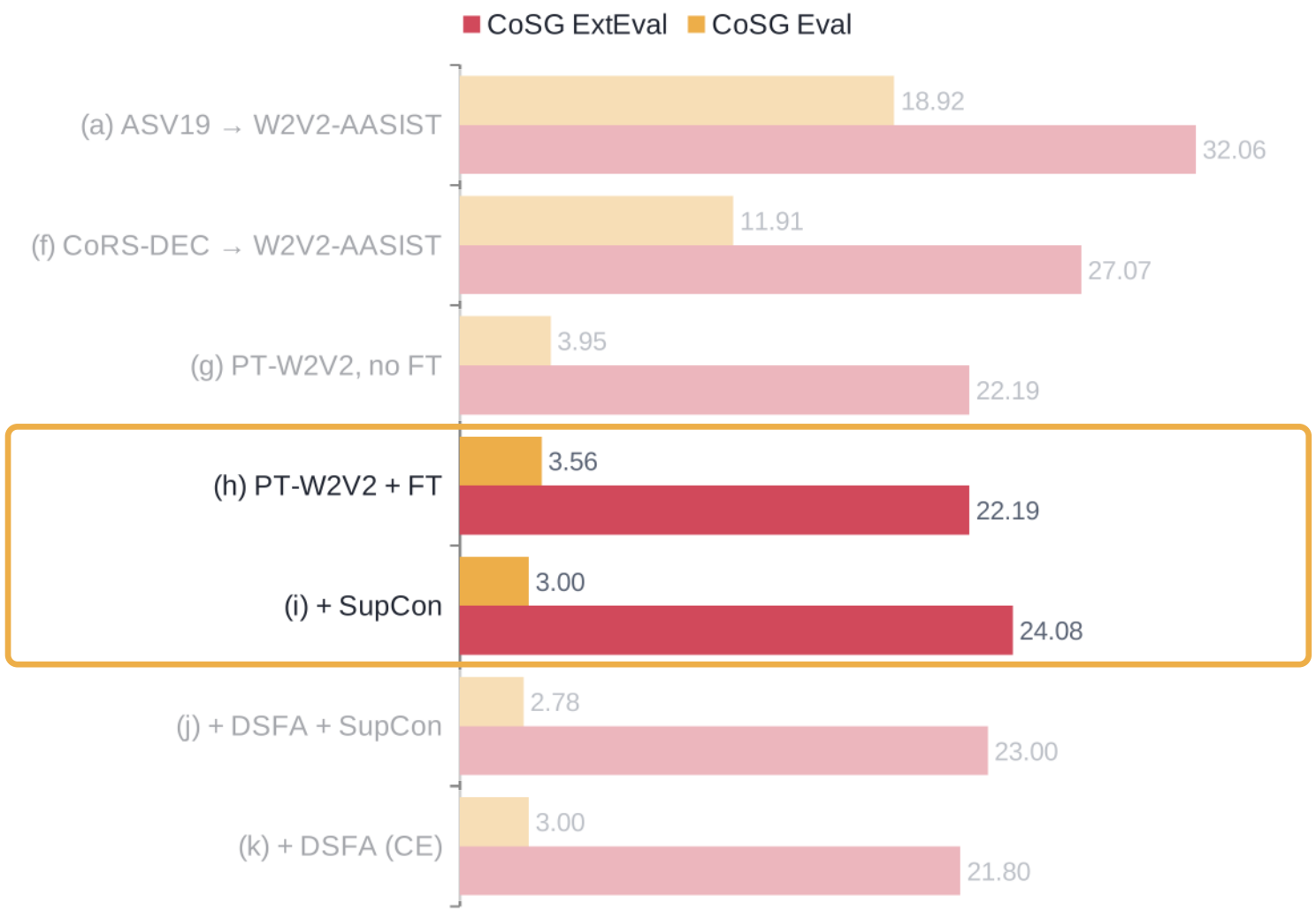
CoSG ExtEval

22.19 → 22.19% **(same)**

Proxy alone can't close the gap → DSFA

EER: lower is better

Main results: EER (%) on CodecFake attacks



SETUP

- Both: post-trained W2V2 fine-tuned on CoRS
- (h) CE loss
- (i) CE + SupCon

3

SupCon helps Eval, hurts ExtEval

CoSG Eval

3.56 → 3.00% (better)

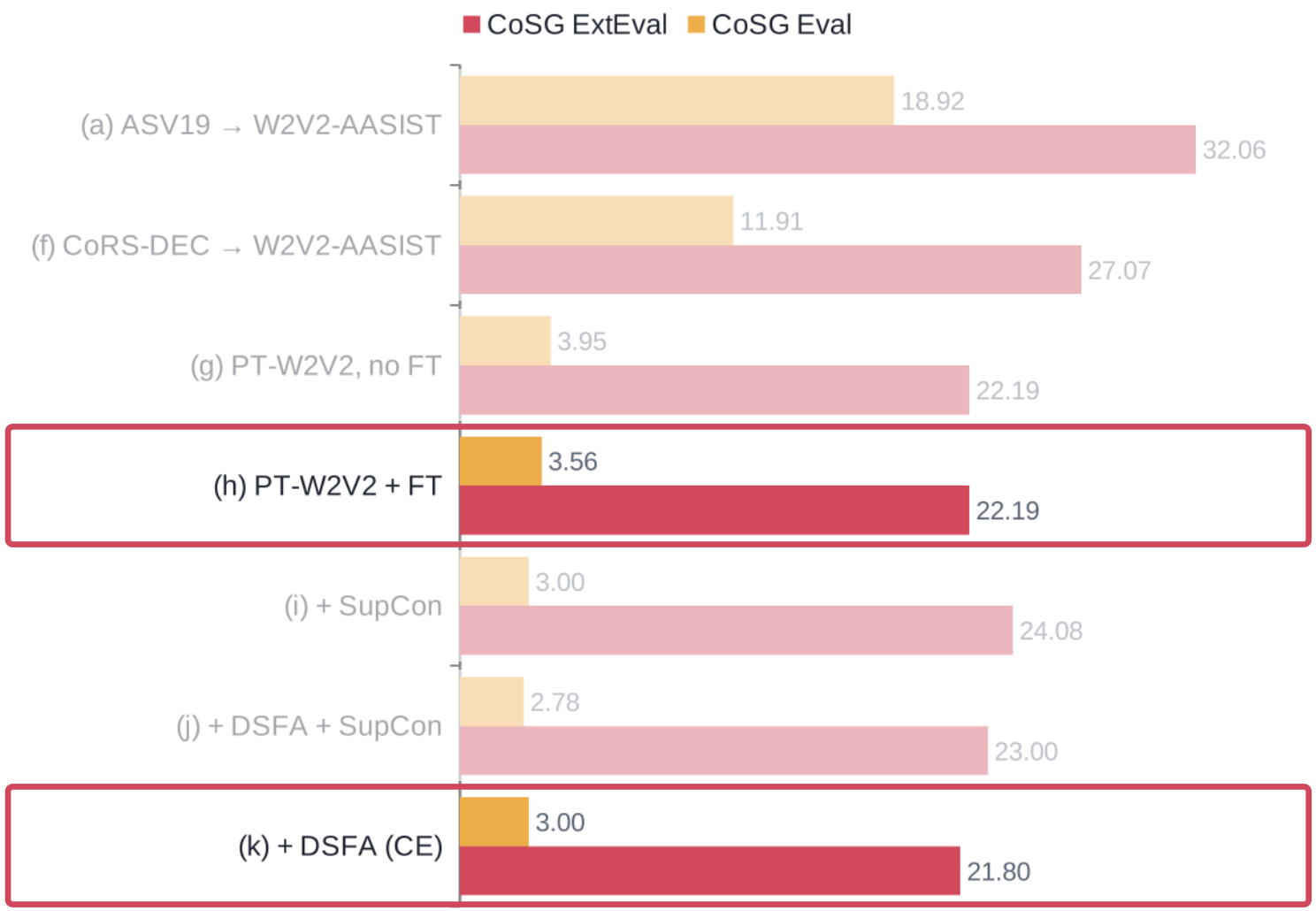
CoSG ExtEval

22.19 → 24.08% (worse)

Better on familiar data ≠ better generalization

EER: lower is better

Main results: EER (%) on CodecFake attacks



SETUP

- Both: post-trained W2V2 fine-tuned on CoRS, CE
- (h) RawBoost only
- (k) RawBoost + DSFA

4 **DSFA + CE improves both**

CoSG Eval

3.56 → 3.00% **(better)**

CoSG ExtEval

22.19 → 21.80% **(best)**

Only DSFA improves both test sets

EER: lower is better · (j) DSFA + SupCon: best Eval 2.78%, ExtEval 23.00%

Ablation: where to apply DSFA

Layer and noise distribution (EER %)

Layer	Uniform		Gaussian	
	Eval	ExtEval	Eval	ExtEval
Baseline (i)	3.00	24.08	3.00	24.08
1	3.12	23.11	**2.78**	**22.61**
6	3.12	23.00	3.00	23.58
12	3.00	23.50	3.00	24.00
18	3.00	23.93	3.45	24.58
24	**2.78**	**22.85**	**2.78**	23.00

SETUP

- Model (i): post-trained W2V2, CE + SupCon, fine-tuned on CoRS
- Insert DSFA after SSL layer 1, 6, 12, 18 or 24
- Noise ε: Uniform U(−1, 1) or Gaussian N(0, 1)
- Grey row: same model without DSFA (3.00 / 24.08%)

1 Best layer depends on the noise type

Gaussian @ layer 1 · Eval / ExtEval

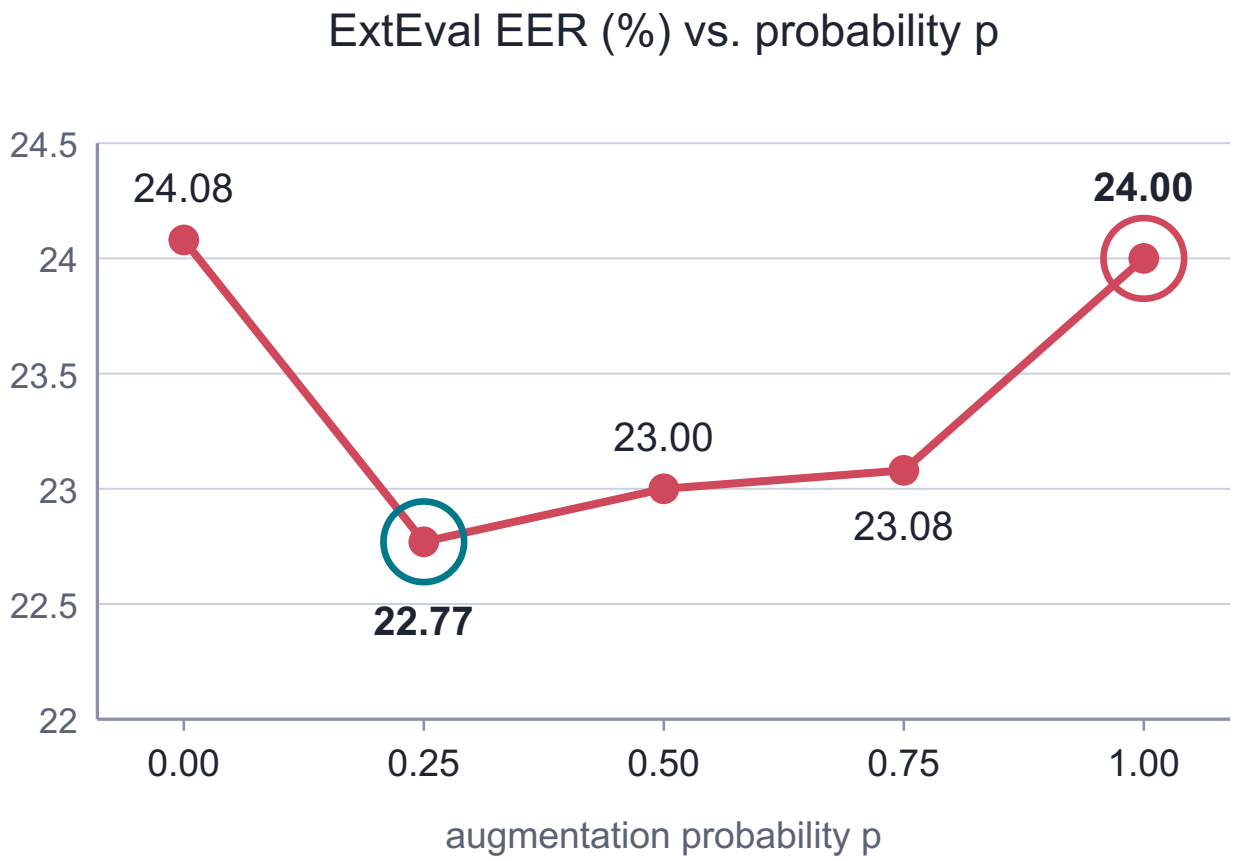
2.78 / 22.61% (better)

Uniform @ layer 24 · Eval / ExtEval

2.78 / 22.85% (better)

9 of 10 settings improve ExtEval

Ablation: how often to apply DSFA



SETUP

- Same model (i) with DSFA
- Each batch is augmented with probability p
- p = 0: no DSFA · p = 1: always on

2 **p = 0.25 is best; always-on hurts**

p = 0 (no DSFA) · CoSG ExtEval

24.08%

p = 0.25 · CoSG ExtEval

22.77% **(best)**

p = 1.0 · CoSG ExtEval

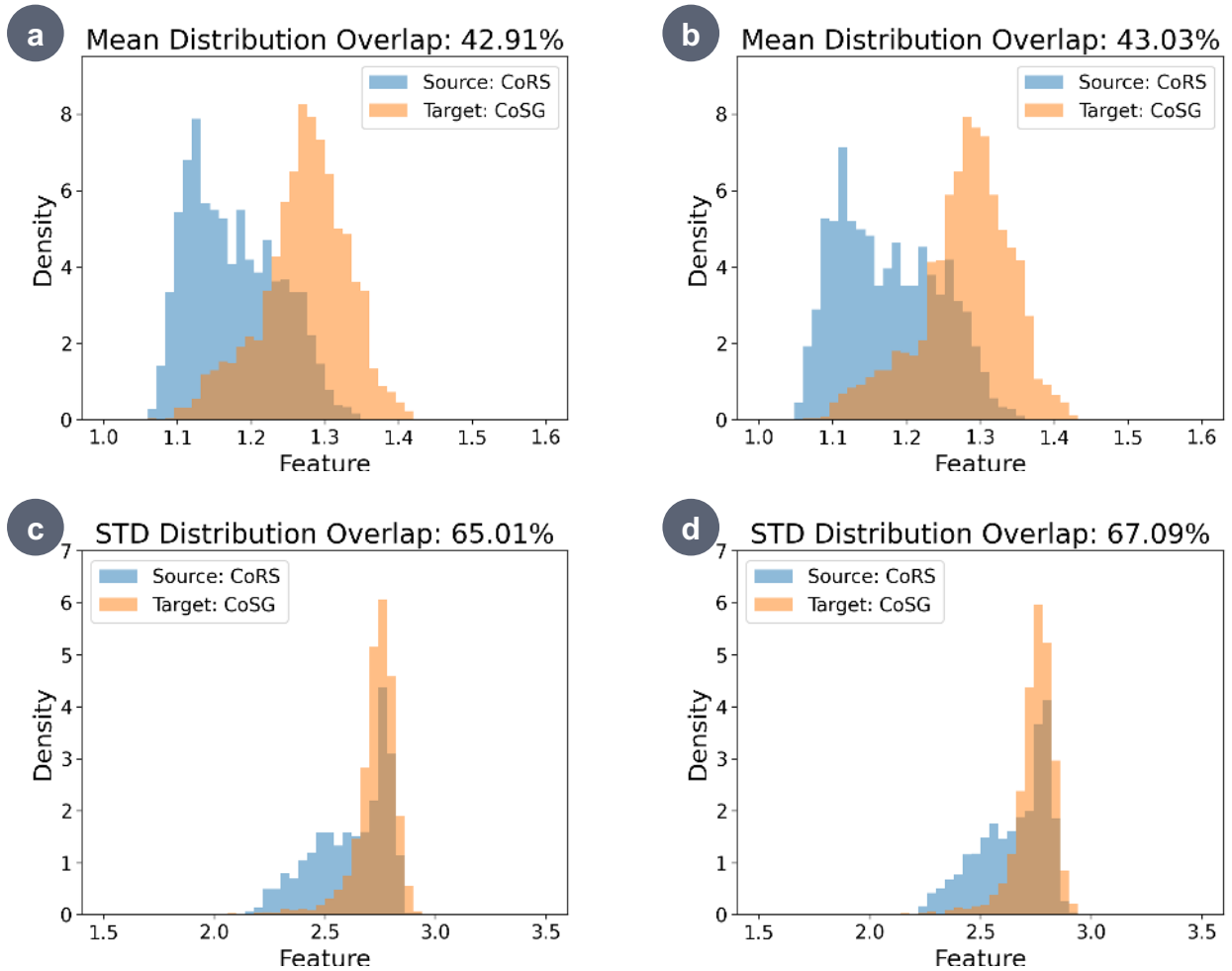
24.00% **(gain gone)**

Mix clean and shifted features

CoSG Eval stays 2.78% for every p > 0

Does DSFA narrow the gap in feature space?

Baseline | **+ DSFA**



SETUP

- Channel mean / std after the 1st Transformer layer
- Blue: CoRS (train) · orange: CoSG Eval (test)
- Metric: overlap of the two histograms

Mean overlap · Fig. 2 (a) → (b)

42.91% → **43.03%**

Std overlap · Fig. 2 (c) → (d)

65.01% → **67.09%**

DSFA pulls train and test statistics closer

Takeaways

1

Proxy data has a ceiling

- CoRS helps CodecFake detection
- But CMs overfit to proxy artifacts: the proxy-to-wild gap

2

DSFA: cheap, train-only, no new parameters

- Sampling shifted feature statistics gives the most consistent gains
- Best EER: 2.78% on CoSG Eval, 21.80% on CoSG ExtEval

3

CoSG ExtEval exposes what's left

- 40 unseen models + long-form audio: EER still > 20%
- Open problems: long audio, acoustic / linguistic mismatch

Thank you — questions?

Paper: arxiv.org/abs/2606.07494

Homepage: www.xjchen.tech · Email: d12942018@ntu.edu.tw
CoSG ExtEval will be released on GitHub.

Paper

Homepage

Xuanjun Chen · National Taiwan University