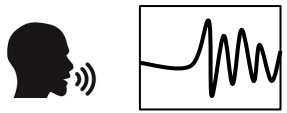
From Adversarial Reprogramming to *Voice2Series*: Learning for Low-Resource Time Series Classification

C.-H. Huck Yang
Georgia Institute of Technology
Sep. 18th

huckiyang@gatech.edu

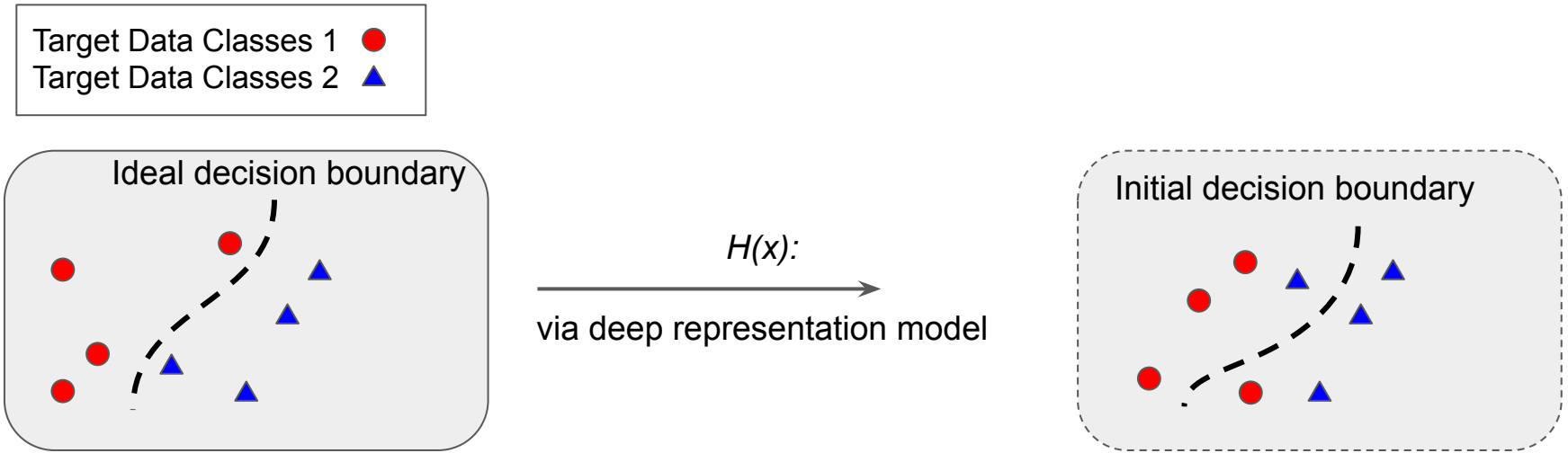


Outline

- Challenges in Low-Resource Data Classification
- Adversarial Reprogramming (AR)
- New Theoretical Justification of AR
- Voice2Series
- Results and Conclusion

Background: Training Processes on ML Models

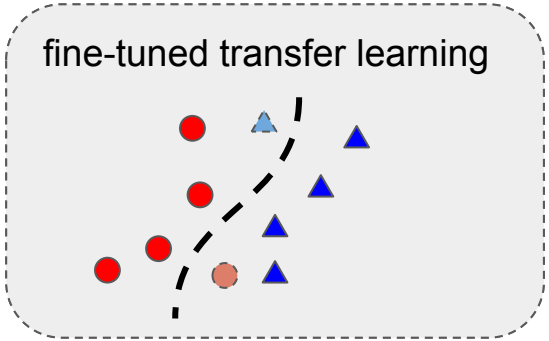
Say we have two prediction classes and few training samples...



Images drafted by c.-h. huck yang 2021

From Pre-Training to Adversarial Reprogramming (1)

Say we have a “pre-training model” and few training samples...



Quick Takeaways

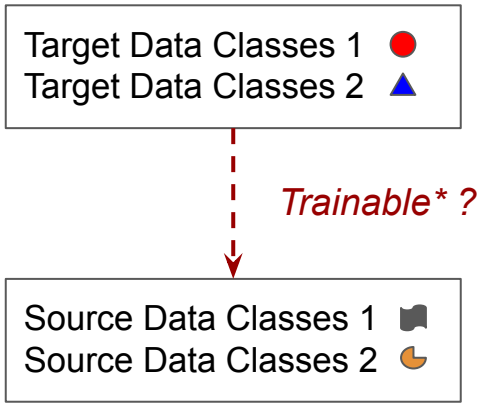
- This is hard to fine-tune a large pretrained model (source domain) with few target domain samples. (1) Small training data, (2) Domain mismatching, (3) Smoothness
- New test data are often outliers sampling from a (low-resource) target domain.

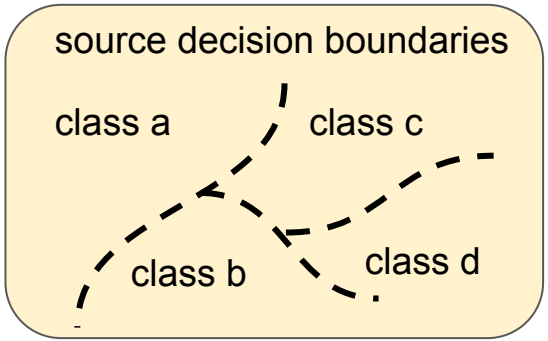
Images drafted by c.-h. huck yang 2021

From Pre-Training to Adversarial Reprogramming (2)

The representation power of the pre-training model is good but …

“Could we also use the **established decision boundaries** from pre-trains?”





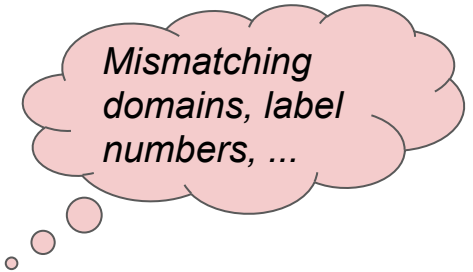
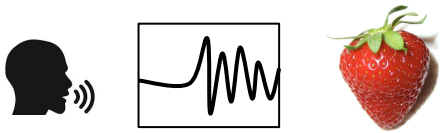


Illustration drafted by c.-h. huck yang 2021

Overview of Voice2Series *[Yang et al. ICML 21]*

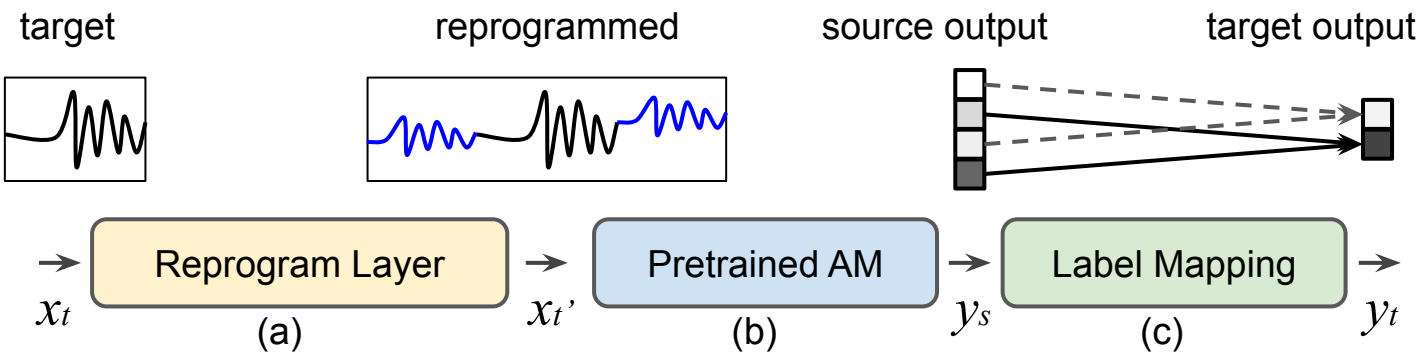
Voice2Series: Reprogramming Acoustic Models for Time Series Classification

- Develop a very first theoretical justification to explain the effects of adversarial reprogramming.
- Use mismatch domain data for adaptation from ***human speech*** to ***time series data*** (e.g., ECG, Earthquake, ...)
- Attain competitive performance and new benchmarks for sequence modeling



I. Introduce Voice-to-Series (V2S)

- Schematic illustration of the proposed Voice-to-Series



Our Contributions in this Work

1. We propose **Voice-to-Series (*V2S*)**. To the best of our knowledge, V2S services as the **first** method that enables reprogramming for time series tasks.

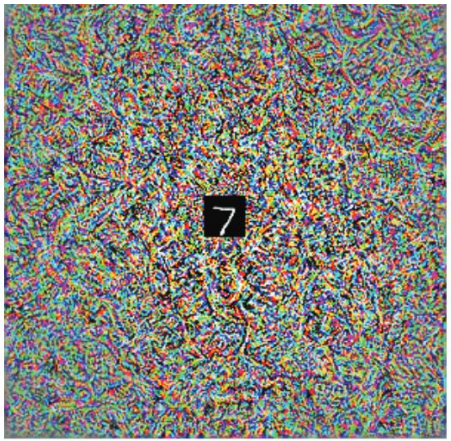
2. Tested on a standard UCR time series classification benchmark with 30 different univariate tasks, ***V2S*** outperforms or is tied with the best reported results on 20 datasets and improves their average accuracy by **1.84%.**

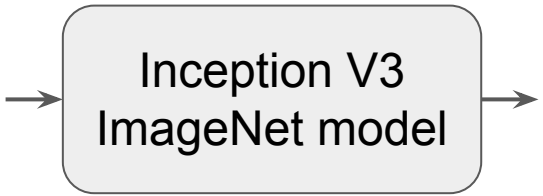
3. We develop a **theoretical risk analysis,** which can be used to assess the performance of reprogramming.

What is Model (Adversarial) Reprogramming?

Reprogramming works for Image to Image Classification (*Elsayed et al. 2018*)

- Training Weights (Perturbation) and Freeze a Pretrained Model





“Seven”

Reprogramming for a MNIST Classifier

II. Proposed Theoretical Analysis for Reprogramming (1)

Table 1. Mathematical notation for reprogramming

Symbol	Meaning
$\mathcal{S}$ / $\mathcal{T}$	source/target domain
$\mathcal{X}_\mathcal{S}$ / $\mathcal{X}_\mathcal{T}$	the space of source/target data samples
$\mathcal{Y}_\mathcal{S}$ / $\mathcal{Y}_\mathcal{T}$	the space of source/target data labels
$\mathcal{D}_\mathcal{S} \subseteq \mathcal{X}_\mathcal{S} \times \mathcal{Y}_\mathcal{S}$ / $\mathcal{D}_\mathcal{T} \subseteq \mathcal{X}_\mathcal{T} \times \mathcal{Y}_\mathcal{T}$	source/target data distribution
$(x, y) \sim \mathcal{D}$	data sample x and one-hot coded label y drawn from $\mathcal{D}$
K	number of source labels
$f_\mathcal{S} : \mathbb{R}^d \mapsto [0,1]^K$	pre-trained K-way source classification model
$\eta : \mathbb{R}^K \mapsto [0,1]^K$	softmax function in neural network, and $\sum_{k=1}^{K} [\eta(\cdot)]_k = 1$
$z(\cdot) \in \mathbb{R}^K$	logit (pre-softmax) representation, and $f(x) = \eta(z(x))$
$\ell(x, y) \triangleq \|f(x) - y\|_2$	risk function of (x, y) based on classifier f
$\mathbb{E}_\mathcal{D}[\ell(x, y)] \triangleq \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(x, y)] = \mathbb{E}_\mathcal{D}\|f(x) - y\|_2$	population risk based on classifier f
δ, θ	additive input transformation on target data, parameterized by θ

II. Proposed Theoretical Analysis for Reprogramming (2)

1. The source risk is $\epsilon_{\mathcal{S}}$, that is, $\mathbb{E}_{\mathcal{D}_{\mathcal{S}}}[\ell(x_s, y_s)] = \epsilon_{\mathcal{S}}$.
2. The source-target label space has a specified surjective one-to-one label mapping function h_t for every target label t, such that $\forall y_t \in \mathcal{Y}_{\mathcal{T}}$, $y_t = h_t(\mathcal{Y}_{\mathcal{S}}) \triangleq y_s \in \mathcal{Y}_{\mathcal{S}}$, and $h_t \neq h_{t'}$ if $t \neq t'$.
3. Based on reprogramming, the target loss function $\ell_{\mathcal{T}}$ with an additive input transformation function δ can be represented as $\ell_{\mathcal{T}}(x_t + \delta, y_t) \overset{(a)}{=} \ell_{\mathcal{T}}(x_t + \delta, y_s) \overset{(b)}{=} \ell_{\mathcal{S}}(x_t + \delta, y_s)$, where (a) is induced by label mapping (Assumption 2) and (b) is induced by reprogramming the source loss with target data.
4. The learned input transformation function for reprogramming is denoted by $\delta^* \triangleq \arg\min_{\delta} \mathbb{E}_{\mathcal{D}_{\mathcal{T}}}[\ell_{\mathcal{S}}(x_t + \delta, y_s)]$, which is the minimizer of the target population risk with the reprogramming loss objective.
5. Domain-independent drawing of source and target data: Let $\Phi_{\mathcal{S}}(\cdot)$ and $\Phi_{\mathcal{T}}(\cdot)$ denote the probability density function of source data and target data distributions over $\mathcal{X}_{\mathcal{S}}$ and $\mathcal{X}_{\mathcal{T}}$, respectively. The joint probability density function is the product of their marginals, i.e., $\Phi_{\mathcal{S},\mathcal{T}}(x_s, x_t) = \Phi_{\mathcal{S}}(x_s) \cdot \Phi_{\mathcal{T}}(x_t)$.

Presented by c.-h. huck yang 2021

II. Proposed Theoretical Analysis for Reprogramming (3)

Lemma 1: Given a K-way neural network classifier $f(\cdot) = \eta(z(\cdot))$. Let μ_z and μ'_z be the probability measures of the logit representations $\{z(x)\}$ and $\{z(x')\}$ from two data domains $\mathcal{D}$ and $\mathcal{D}'$, where $x \sim \mathcal{D}$ and $x' \sim \mathcal{D}'$. Assume independent draws for x and x', i.e., $\Phi_{\mathcal{D},\mathcal{D}'}(x, x') = \Phi_{\mathcal{D}}(x) \cdot \Phi_{\mathcal{D}'}(x')$. Then

$$\mathbb{E}_{x\sim\mathcal{D},\, x'\sim\mathcal{D}'} \|f(x) - f(x')\|_2 \leq 2\sqrt{K} \cdot \mathcal{W}_1(\mu_z, \mu'_z),$$

where $\mathcal{W}_1(\mu_z, \mu'_z)$ is the Wasserstein-1 distance between μ_z and μ'_z.

II. Proposed Theoretical Analysis for Reprogramming (4)

- Population Risk via Reprogramming (Optimal Transport)

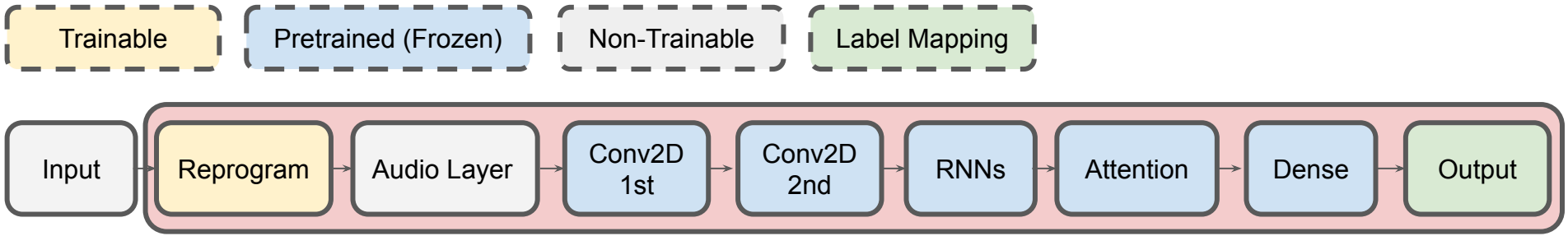
Theorem 1: Let δ^* denote the learned additive input transformation for reprogramming. The population risk for the target task via reprogramming a K-way source neural network classifier $f_{\mathcal{S}}(\cdot) = \eta(z_{\mathcal{S}}(\cdot))$, denoted by $\mathbb{E}_{\mathcal{D}_{\mathcal{T}}}[\ell_{\mathcal{T}}(x_t + \delta^*, y_t)]$, is upper bounded by:

$$\mathbb{E}_{\mathcal{D}_{\mathcal{T}}}[\ell_{\mathcal{T}}(x_t + \delta^*, y_t)] \leq \underbrace{\epsilon_{\mathcal{S}}}_{\text{source risk}} + 2\sqrt{K} \cdot \underbrace{\mathcal{W}_1(\mu(z_{\mathcal{S}}(x_t + \delta^*)), \mu(z_{\mathcal{S}}(x_s)))_{x_t \sim \mathcal{D}_{\mathcal{T}},\, x_s \sim \mathcal{D}_{\mathcal{S}}}}_{\text{representation alignment loss via reprogramming}}$$

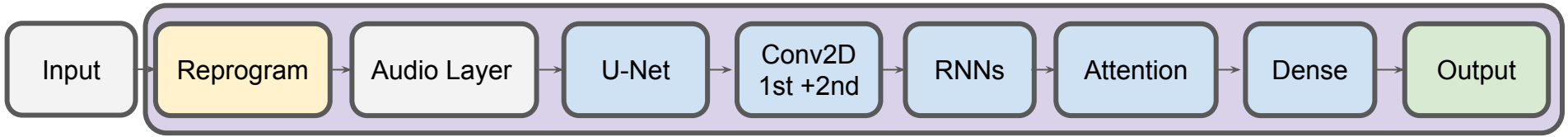
This results suggest that reprogramming can perform **better** (lower risk) when the source model has a lower source loss and smaller representation loss.

I. Voice-to-Series (V2S) Design

- Schematic illustration of the proposed Voice-to-Series



(a) Voice-to-Series with Transformer-based Attention (V2S$_a$)



(b) Voice-to-Series with U-Net Transformer-based Attention (V2S$_u$)

Open Source Implemented Layer and Code: https://github.com/huckiyang/Voice2Series-Reprogramming

I. Voice-to-Series (V2S) Performance on UCR Archive

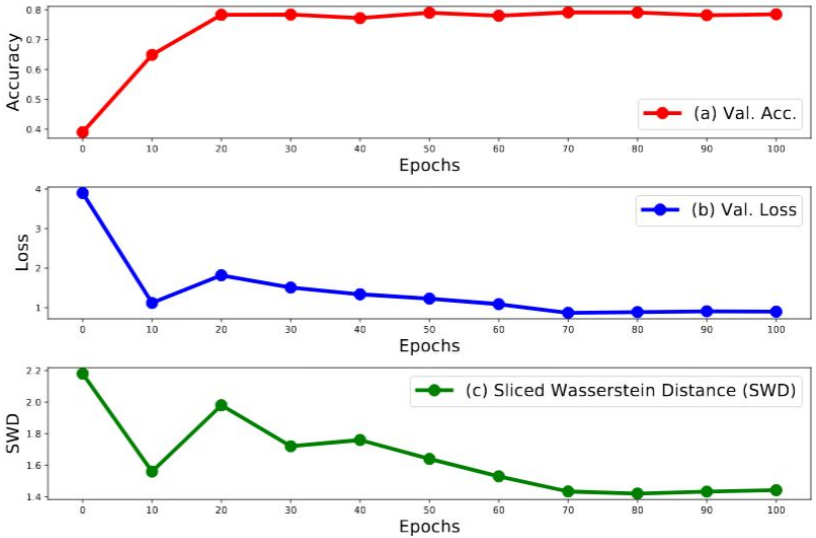
Table 2. Performance comparison of test accuracy (%) on 30 UCR time series classification datasets (Dau et al., 2019). Our proposed $V2S_a$ outperforms or ties with the current SOTA results (discussed in Section 5.3) on 20 out of 30 datasets.

Dataset	Type	Input size	Train. Data	Class	SOTA	$V2S_a$	$V2S_u$	TF_a
Coffee	SPECTRO	286	28	2	**100**	**100**	**100**	53.57
DistalPhalanxTW	IMAGE	80	400	6	**79.28**	79.14	75.34	70.21
ECG 200	ECG	96	100	2	90.9	**100**	**100**	**100**
ECG 5000	ECG	140	500	5	**94.62**	93.96	93.11	58.37
Earthquakes	SENSOR	512	322	2	76.91	**78.42**	76.45	74.82
FordA	SENSOR	500	2500	2	96.44	**100**	**100**	**100**
FordB	SENSOR	500	3636	2	92.86	**100**	**100**	**100**
GunPoint	MOTION	150	50	2	**100**	96.67	93.33	49.33
HAM	SPECTROM	431	109	2	**83.6**	78.1	71.43	51.42
HandOutlines	IMAGE	2709	1000	2	**93.24**	**93.24**	91.08	64.05
Haptics	MOTION	1092	155	5	51.95	**52.27**	50.32	21.75
Herring	IMAGE	512	64	2	**68.75**	**68.75**	64.06	59.37
ItalyPowerDemand	SENSOR	24	67	2	97.06	**97.08**	96.31	97
Lightning2	SENSOR	637	60	2	86.89	**100**	**100**	**100**
MiddlePhalanxOutlineCorrect	IMAGE	80	600	2	72.23	**83.51**	81.79	57.04
MiddlePhalanxTW	IMAGE	80	399	6	58.69	**65.58**	63.64	27.27
Plane	SENSOR	144	105	7	**100**	**100**	**100**	9.52
ProximalPhalanxOutlineAgeGroup	IMAGE	80	400	3	88.09	**88.78**	87.8	48.78
ProximalPhalanxOutlineCorrect	IMAGE	80	600	2	**92.1**	91.07	90.03	68.38
ProximalPhalanxTW	IMAGE	80	400	6	81.86	**84.88**	83.41	35.12
SmallKitchenAppliances	DEVICE	720	375	3	**85.33**	83.47	74.93	33.33
SonyAIBORobotSurface	SENSOR	70	20	2	**96.02**	**96.02**	91.71	34.23
Strawberry	SPECTRO	235	613	2	**98.1**	97.57	91.89	64.32
SyntheticControl	SIMULATED	60	300	6	**100**	98	99	49.33
Trace	SENSOR	271	100	4	**100**	**100**	**100**	18.99
TwoLeadECG	ECG	82	23	2	**100**	96.66	97.81	49.95
Wafer	SENSOR	152	1000	2	99.98	**100**	**100**	100
WormsTwoClass	MOTION	900	181	2	83.12	**98.7**	90.91	57.14
Worms	MOTION	900	181	5	80.17	**83.12**	80.34	42.85
Wine	SPECTRO	234	57	2	**92.61**	90.74	90.74	50
Mean accuracy (↑)	-	-	-	-	88.02	**89.86**	87.92	56.97
Median accuracy (↑)	-	-	-	-	92.36	**94.99**	91.40	53.57
MPCE (mean per class error) (↓)	-	-	-	-	2.09	**2.01**	2.10	48.34

Achieve or outperform SOTA in 20 out of 30 datasets

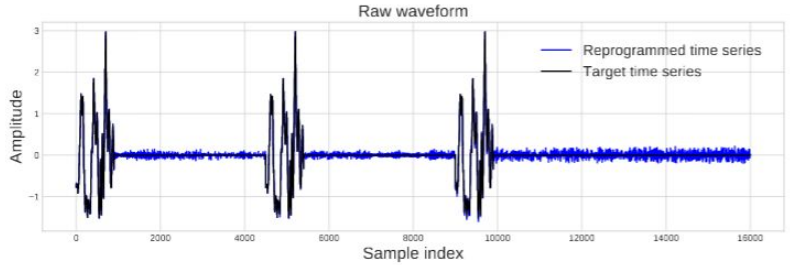
II. Proposed Theoretical Analysis for Reprogramming

- Training-time reprogramming analysis using V2S and DistalPhalanxTW dataset (Davis, 2013)

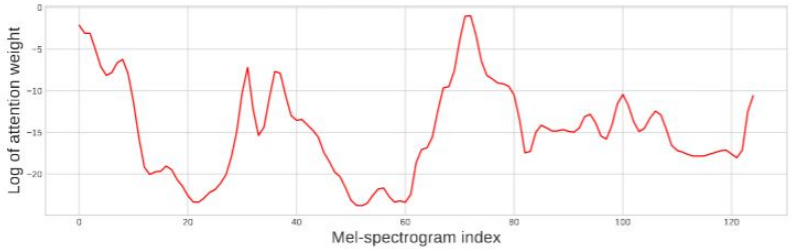


III. Voice-to-Series (V2S) Visualization - (1)

- Proposed Voice-to-Series on the Worms dataset (Bagnall et al., 2015)



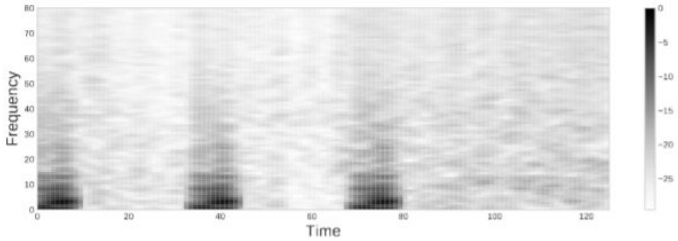
(a) Targeted (blue) and reprogrammed (black) time series



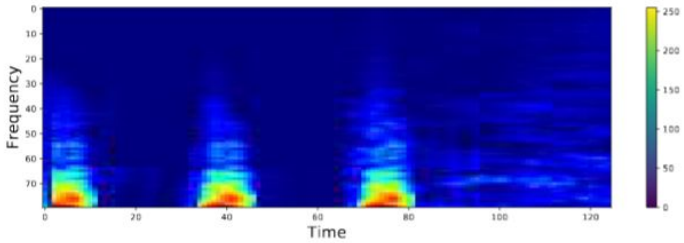
(b) Attention weight of reprogrammed input

III. Voice-to-Series (V2S) Visualization - (2)

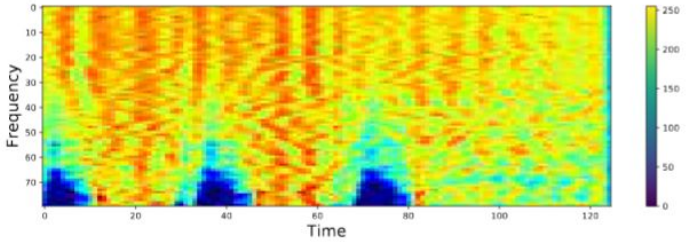
- Proposed Voice-to-Series on the Worms dataset (Bagnall et al., 2015)



(c) Mel-spectrogram of reprogrammed input



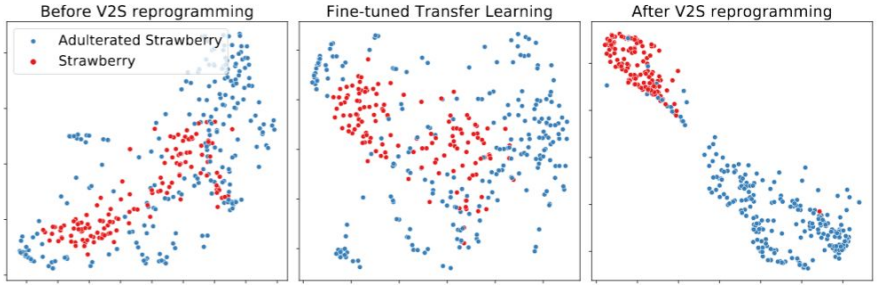
(d) Class activation mapping of (c) from 1_{st} conv-layer



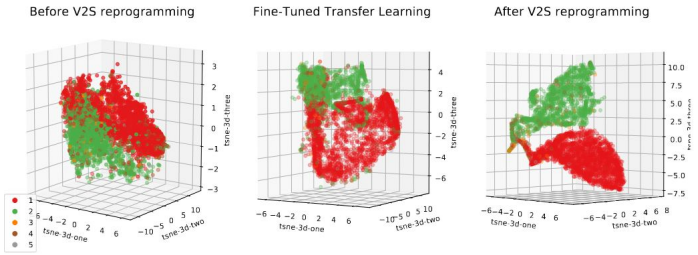
(e) Class activation mapping of (c) from 2_{nd} conv-layer

III. Voice-to-Series (V2S) Visualization - (3)

- tSNE plots of the logit representations using the Strawberry (Holland et al., 1998)



(e) Task: Strawberry 2D tSNE.



(b) Task: ECG 5000 with 3D tSNE

Ongoing and Future Works

- Reprogramming looks also working well for Speech to Speech Processing, Sequence Modeling and Language Modeling.

- If you are interested to collaborate, please feel free to email Huck and Pin-Yu.

Huck Yang
Georgia Tech

Yun-Yun Tsai
Columbia

Pin-Yu Chen
IBM Research

Acknowledgement

A. Large-Scale Pretrained Speech and Acoustic Models

1. *Choi et al.* “Kapre: On-GPU Audio Preprocessing Layers for a Quick Implementation of Deep Neural Network Models,” ***ICML Workshop* 2017**
2. Yang *et al.* “Decentralizing feature extraction with quantum convolutional neural network for automatic speech recognition,” ***ICASSP 2021***, Code
3. *Hu et al.* “A Two-Stage Approach to Device-Robust Acoustic Scene Classification,” ***ICASSP 2021, DCASE 20 Task-1 Best System,*** Code

B. Time Series Classification

1. *Wang et al.* “Time Series Classification from Scratch with Deep Neural Networks: A Strong Baseline,” **IJCNN 2019**
2. *Dau et al.* “The UCR Time Series Archive,” ***IEEE/CAA Journal of Automatica Sinica***

References

C. Adversarial Reprogramming

1. *Elsayed et al.* “Adversarial reprogramming of neural networks,” ***ICLR 2018***
2. *Tsai et al.* “Transfer learning without knowing: Reprogramming black-box machine learning models with scarce data and limited resources,” ***ICML 2020***
3. *Neekhara et al.* “Adversarial Reprogramming of Text Classification Neural Networks,” ***EMNLP 2019***

D. Transfer Learning in Time Series Classification

1. *Fawaz et al.* “Transfer learning for time series classification” ***Big Data 2018***
2. *Kashiparekh et al.* “ConvTimeNet: A pre-trained deep convolutional neural network for time series classification. ***IJCNN 2019***