Multi-source Transfer Learning for Signal Detection over a Fading Channel with Co-channel Interference

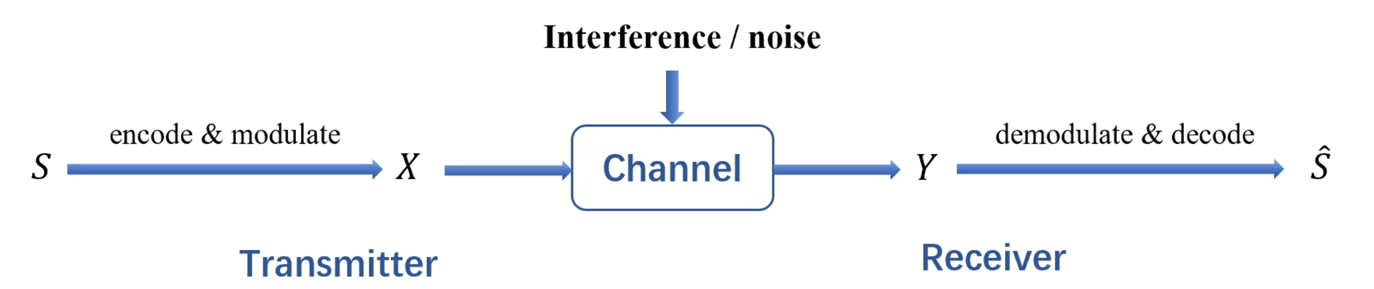
Ziyan Zheng, Xinyi Tong, Xinchun Yu, Xiangxiang Xu, Shao-Lun Huang

清华-伯克利深圳学院 TBSI Tsinghua-Berkeley Shenzhen Institute

Abstract

For signal detection tasks in wireless communications, most of the existing algorithms either ignore the co-channel interference or treat it as Gaussian noise, which may result in unsatisfactory performance if the interference is complex-distributed. When neural networks are motivated, difficulty arises due to the lack of accessible pilots in each packet for training. In this paper, based on multi-source transfer learning, we propose a data-driven detector for signal detection in a time-varying fading channel with interference, which is able to transfer channel knowledge of previous packets into the latest detection. Finally, numerical simulations validate the advantages of our algorithms.

Background

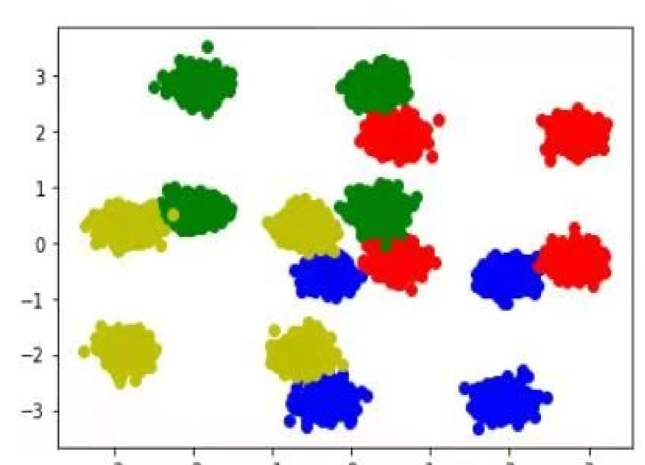


Signal detection task -- detect $\boldsymbol{X}$ from $\boldsymbol{Y}$

Standard model-based approaches may be inapplicable if:

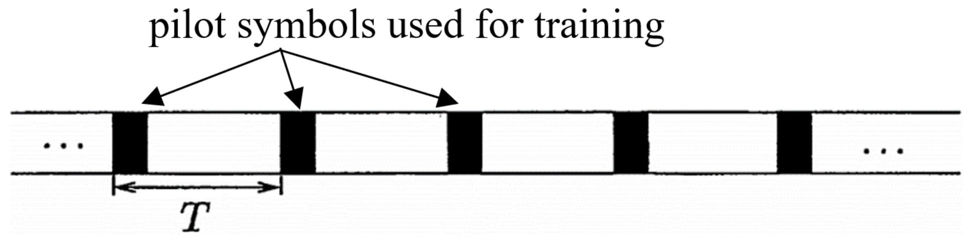
- The precise channel model is unavailable
- The optimal receiver is complicated or unknown

Co-channel noise, mainly caused by multiple radios transmitting on the same frequency at the same time, is inevitable in many wireless communication systems. The figure on the right is an example of a constellation of transmitted QPSK-modulated signals with strong QPSK-modulated interference.



Applications of deep neural networks (DNN)

- Strength -- Feature extraction from complex-distributed data
- Difficulty -- Lack of pilots for training in time-varying fading channels



In a detector for time-varying channels, can historical reserved data be used in training? -- Transfer learning says 'Yes'.

Model

received signal — discrete transmitted symbol

$$Y_t = H_t X_t + I_t + n_t$$

time-varying channel state matrix — co-channel interference — AWGN

◆ **Notations and Assumptions**

- $t \in \mathbb{Z}$: time period / packet index
- Each period t contains many symbols to transmit, the first several ones are pilots
- $H_t,\ X_t,\ I_t,\ n_t$ are pairwise independent

◆ **Goal**

- Detect X_T from Y_T at the latest period $T \in \mathbb{Z}$

◆ **Available training samples for the detector**

- Source samples: $\left\{\left(x_t^{(i)}, y_t^{(i)}\right)\right\}_{i=1}^{N_t}$ for $t = 1,\ 2,\ \ldots,\ T-1$
- Target samples: $\left\{\left(x_T^{(i)}, y_T^{(i)}\right)\right\}_{i=1}^{N_T}$, note that $N_T \ll N_t\ (\forall t < T)$

Theoretical Framework

Use the linear combination of sources and target learned distribution for each period t

$$Q_{X_T Y_T}^{(\boldsymbol{w})} \triangleq \sum_{t=1}^{T} w_t \hat{P}_{X_t Y_t}$$

to fit the real underlying target distribution $P_{X_T Y_T}$ (by minimizing the χ^2-distance), with combining weights

$$\boldsymbol{w} \in \{(w_1, w_2, \ldots, w_T) : \sum_{t=1}^{T} w_t = 1, w_t \geq 0\}$$

The key factors to determine usefulness of historical data are

- Channel similarity: $\chi^2\left(P_{X_T Y_T}, \sum_{t=1}^{T} w_t P_{X_t Y_t}\right)$
- Number of pilots and previous signals for training: N_t for $t = 1, 2, \ldots, T$
- Domain needed to capture / Model complexity

Adopt the discriminative model

$$\tilde{P}_{X_T|Y_T}^{(\boldsymbol{f},\boldsymbol{g})}(x|y) \triangleq P_{X_T}(x)\left(1 + \boldsymbol{f}^T(x)\,\boldsymbol{g}(y)\right)$$

with weights $\hat{\boldsymbol{f}}_t$ defined as

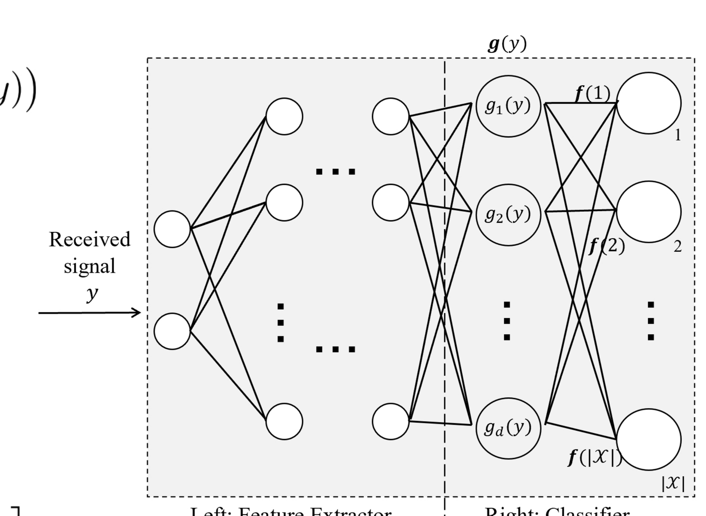
$$\hat{\boldsymbol{f}}_t \triangleq \arg\min_{\boldsymbol{f}}\ \chi^2_{R_{XY}}\left(\hat{P}_{X_t Y_t}, P_{Y_T}\tilde{P}_{X_T|Y_T}^{(\boldsymbol{f},\boldsymbol{g})}\right)$$

The convex combination becomes

$$Q_{X_T|Y_T}^{(\boldsymbol{w})} \triangleq \sum_{t=0}^{T} w_t \tilde{P}_{X_T|Y_T}^{(\hat{\boldsymbol{f}}_t,\boldsymbol{g})} = \tilde{P}_{X_T|Y_T}^{(\hat{\boldsymbol{f}},\boldsymbol{g})}$$

and the testing loss is

$$L_{\text{test}}^{(\boldsymbol{w})} \triangleq \mathbb{E}\left[\chi^2_{R_{XY}}\left(P_{X_T Y_T}, P_{Y_T} Q_{X_T|Y_T}^{(\boldsymbol{w})}\right)\right]$$



Algorithm

Algorithm 1 Algorithm for training the MSTL detector

Input:

Historical samples $\left\{\left(x_t^{(i)} y_t^{(i)}\right)\right\}_{i=1}^{N_t}$ $(t = 1, 2, \ldots, T-1)$

Pilots $\left\{\left(x_T^{(i)} y_T^{(i)}\right)\right\}_{i=1}^{N_T}$

Randomly initialize $\boldsymbol{w}^*$ and DNN parameters

Output:

1: **repeat**
2: $(\boldsymbol{f}^*, \boldsymbol{g}^*) \leftarrow \arg\min_{\boldsymbol{f},\boldsymbol{g}} L^{(\boldsymbol{w}^*, \boldsymbol{f}, \boldsymbol{g})}$ — Train neural networks
3: $\boldsymbol{w}^* \leftarrow \arg\min_{\boldsymbol{w}} L_{\text{test}}^{(\boldsymbol{w})}$ — Solve a non-negative quadratic programming problem
4: **until** $\boldsymbol{w}^*$ converges
5: $(\boldsymbol{f}^*, \boldsymbol{g}^*) \leftarrow \arg\min_{\boldsymbol{f},\boldsymbol{g}} L^{(\boldsymbol{w}^*, \boldsymbol{f}, \boldsymbol{g})}$
6: **return** $\boldsymbol{f}^*, \boldsymbol{g}^*$;

- Loss function optimizing $(\boldsymbol{f}, \boldsymbol{g})$

$$L^{(\boldsymbol{w},\boldsymbol{f},\boldsymbol{g})} \triangleq \sum_{t=1}^{T} w_t \chi^2_{R_{XY}}\left(\hat{P}_{X_t Y_t}, P_{Y_T}\tilde{P}_{X_T|Y_T}^{(\boldsymbol{f},\boldsymbol{g})}\right)$$

- Loss function optimizing $\boldsymbol{w}$

$$L_{test}^{(\boldsymbol{w})} = \chi^2_{R_{XY}}\left(P_{Y_T}\tilde{P}_{X_T|Y_T}^{(\boldsymbol{f}_T,\boldsymbol{g})}, \sum_{t=1}^{T} w_t \tilde{P}_{X_T|Y_T}^{(\boldsymbol{f}_t,\boldsymbol{g})}\right) + \sum_{t=1}^{T} \frac{w_t^2}{N_t}\tilde{V}_t + \chi^2_{R_{XY}}\left(P_{X_T Y_T}, P_{Y_T}\tilde{P}_{X_T|Y_T}^{(\boldsymbol{f}_T,\boldsymbol{g})}\right)$$

Determine $\hat{x}$ by the MAP decision rule

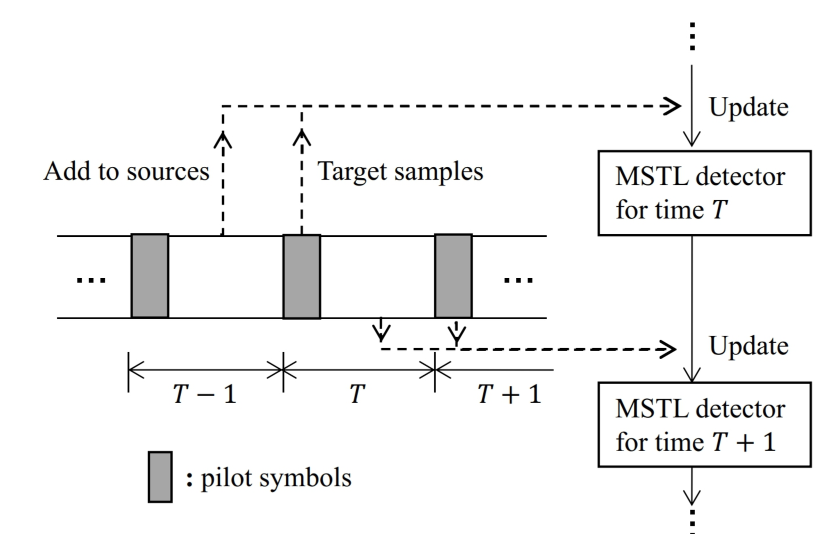
$$\hat{x}(y) = \arg\max_{x\in\mathcal{X}}\ \tilde{P}_{X_T|Y_T}^{(\boldsymbol{f}^*,\boldsymbol{g}^*)}(x|y) = \arg\max_{x\in\mathcal{X}}\ P_{Y_T}(y)\left(1 + \boldsymbol{f}^{*T}(x)\,\boldsymbol{g}^*(y)\right)$$

Simulations

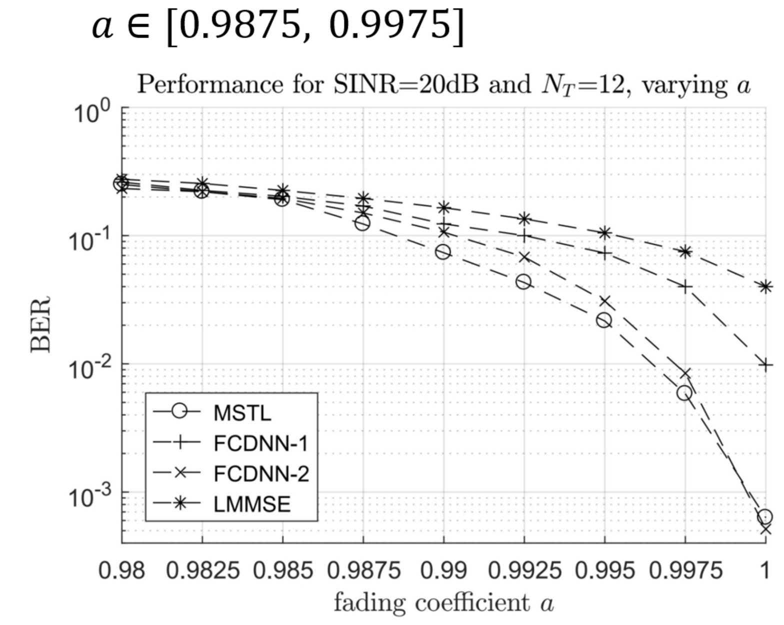
The update process of the MSTL detector is shown in the figure on the right.

Comparisions in simulations:

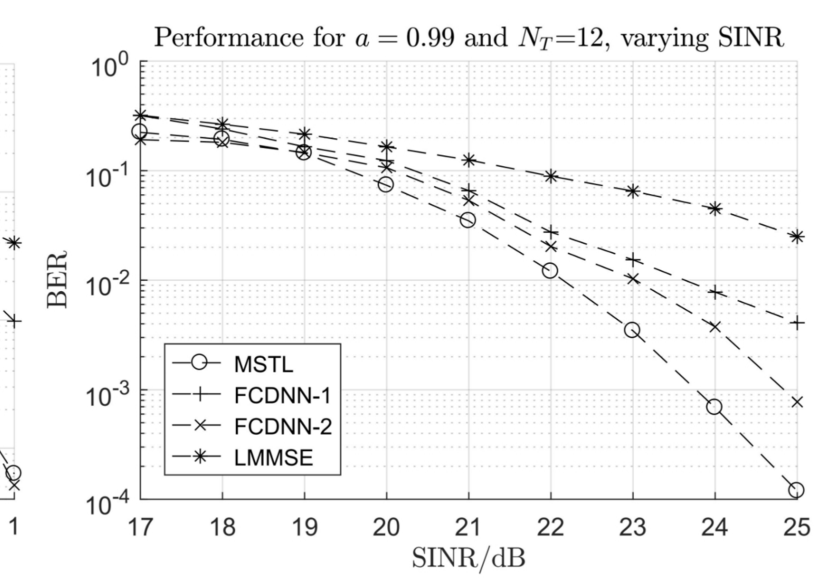
- FCDNN-1: Conventional fully-connected DNN based on L_2 loss, using N_T pilots for training
- FCDNN-2: Using N_T pilots and $\sum_{t=T-9}^{T-1} N_t$ reserved samples for training
- LMMSE: Estimate CSI by N_T pilots



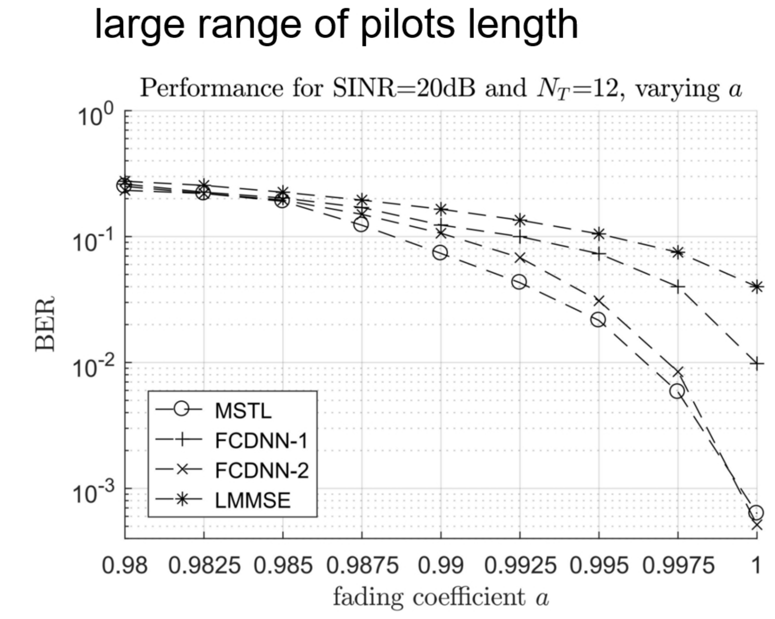
- Largest gain occurs in the range $a \in [0.9875,\ 0.9975]$



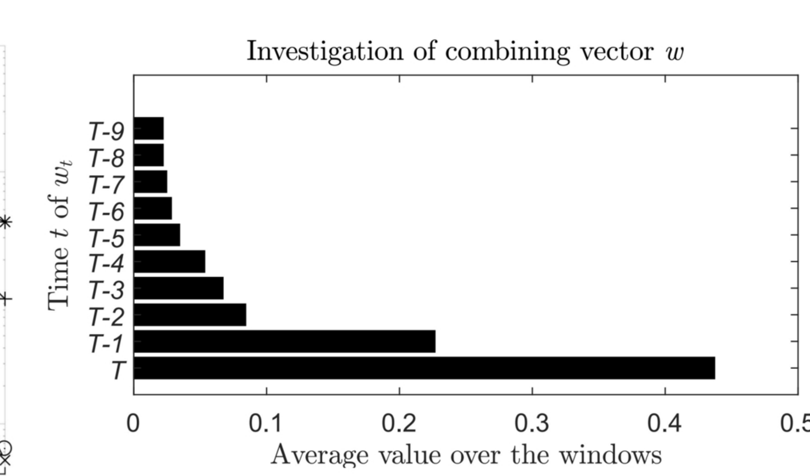
- Low BER for high SINR



- Outperforms other algorithms in a large range of pilots length



- Monotonous similarity of the sources



Setups for simulations

H_t: Gauss-Markov process $H_t = aH_{t-1} + u_t,\quad u_t \sim \mathcal{CN}\left(0, \left(1-a^2\right)\sigma_H^2\right)$; X_t: QPSK-modulated signals $X_t = K_t + jQ_t$, $(K_t, Q_t) \in \{(1,1), (-1,1), (-1,-1), (1,-1)\}$; I_t: a QPSK-modulated interference with stronger power; n_t: AWGN with variance σ_n^2 .

References

S. Park, H. Jang, O. Simeone, and J. Kang, "Learning to demodulate from few pilots via offline and online meta-learning," IEEE Transactions on Signal Processing, vol. 69, pp. 226–239, 2020.
C. Liu, Y. Chen, and S.-H. Yang, "Signal detection with co-channel interference using deep learning," Physical Communication, vol. 47, p. 101343, 2021.
X. Tong, X. Xu, S.-L. Huang, and L. Zheng, "A mathematical framework for quantifying transferability in multi-source transfer learning," Advances in Neural Information Processing Systems, vol. 34, 2021.