IV-06

Disentangling Sources of Variability in Sensor-Channel Importance Rankings for Human Activity Recognition

Maksym LAPIN[1], Dalius NAVAKAUSKAS[1]

E-mail: maksym.lapin@vilniustech.lt, [1]Department of Electronic Systems, Vilnius Gediminas Technical University

Introduction

A study may name the most important **wearable sensor channel** for a model, for example, the gyroscope y [4]. However, the channel ranking changes with the random **seed** [5] and with the way a channel is modified to probe the model – the **operator** [1–3]. In the following we measure how much the operator matters, in units of the change caused by a new seed.

The Aim – to separate

what a channel ranking tells us about **the data**
from
what it tells us about **the setup** behind it

Same Models, Four Diverse Rankings



UCI-HAR, mean importance over all 24 840 models.

Experimental Setup

DATA public datasets

2	30 | 8	9	6	128
datasets UCI-HAR, PAMAP2	participants wearing sensors	channels sensor signals per window	activities to classify	samples per window, 2.56 s at 50 Hz

TRAINING per dataset

12	×	69	×	30	=	24 840
splits participants in train|val|test: 16|5|9, 5|1|2		architectures 1–4 GRU layers, 4–64 neurons each		seeds random starts of the training		models per dataset

Shapes	Number tested	Depth frequency
equal in all layers	22	1 layer: 6
narrowing	17	2 layers: 18
widening	18	3 layers: 23
widest in the middle	12	4 layers: 22

ANALYSIS 300 test windows per split

Operators

Zero	Average	Random window	Window average
the channel is set to zero	the training mean of the channel	the channel of a random training window	the channel mean over the window

Metrics

Ranking
$\pi^r = (c_1, \dots, c_9)$ with $I^r_{c_1} \geq \dots \geq I^r_{c_9}$
$I^r_c = \frac{1}{6}\sum_{k=1}^{6} \mathrm{mean}_{x \in k} \, |e^r_c(x)|, \qquad e^r_c(x) = z_{\hat{y}}(x) - z_{\hat{y}}(x^{c \to r})$
π^r: channels c sorted by importance I^r_c under operator r; $e^r_c(x)$: effect in window x; $z_{\hat{y}}$: predicted-class logit; $x^{c \to r}$: channel c of x modified by r; k: true activity.

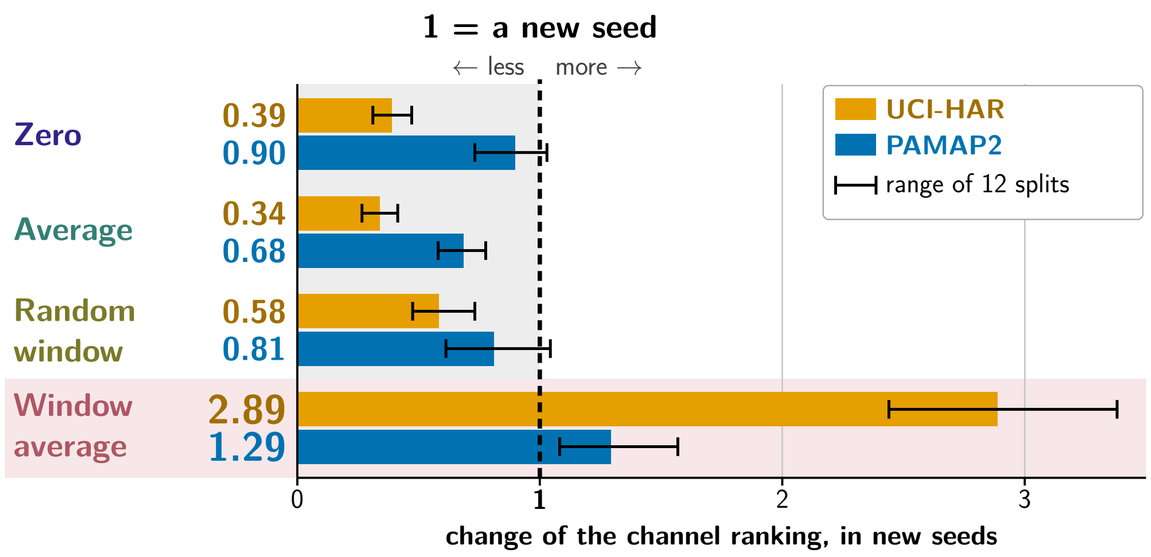
Ranking difference
$d(\pi, \sigma) = 1 - \tau_b(\pi, \sigma)$, reported as $d(\pi^r, \pi^{r'}) \,/\, d_{\text{seed}}$
τ_b: Kendall correlation of rankings π, σ; d_{seed}: mean d between two seeds of one architecture (larger of r, r'); $1 =$ a new seed.

Deletion gain
$G(\pi; r) = \mathrm{AUC}_{\text{rev}} - \mathrm{AUC}_{\text{fwd}}$
AUC: area under the predicted-class probability $p_{\hat{y}}$ as r modifies channels in the order of π, most important first (fwd) or last (rev). Highest G wins the test.

Hypothesis	Key result UCI-HAR | PAMAP2	Verdict
1. The operator changes the channel ranking no more than a new seed does. $d(\pi^r, \pi^{r'}) \,/\, d_{\text{seed}} \leq 1$ for all $r \neq r'$	**2.89 | 1.29** Window average, as a ratio to the change from a new seed. Other three operators: mean below 1 on both datasets	**Rejected** Window average moves the ranking more than a new seed does in every split.
2. In a deletion test, one ranking wins whichever operator is used. $\exists\, r^*\colon\ G(\pi^{r^*}; r) \geq G(\pi^{r'}; r)$ for all r, r'	**44 | 39** of 48 tests are won by the ranking built with the test operator	**Rejected** In no split does one ranking win all four tests. A deletion test alone cannot choose the operator.

Conclusion – Report the operator with every channel ranking.

H1 Evidence – Window Average Exceeds a New Seed



H2 Evidence – No Ranking Wins Every Test

UCI-HAR

Ranking built with \ Deletion test run with	Zero	Average	Random window	Window average
Zero	**12**	**1**	0	0
Average	0	**8**	0	0
Random window	0	**3**	**12**	0
Window average	0	0	0	**12**

framed cells together: **44 of 48**

PAMAP2

Ranking built with \ Deletion test run with	Zero	Average	Random window	Window average
Zero	**12**	**1**	0	0
Average	0	**3**	0	0
Random window	0	**8**	**12**	0
Window average	0	0	0	**12**

framed cells together: **39 of 48**

Each cell: number of the 12 splits in which the row ranking wins the **deletion test**. On held-out windows the framed cells give **39** and **37** of 48.

References: [1] jmlr.org/papers/v22/20-1316 (no DOI) [2] doi:10.48550/arXiv.2401.06654 [3] doi:10.1007/978-3-031-44070-0_9 [4] doi:10.3390/s25144420 [5] doi:10.1109/WACV51458.2022.00159

VILNIUS TECH
Department of Electronic Systems