ESIT2026-D2I competition

Charting the Path to Bifröst

Channel Charting, Robust Learning, and Unlearning Noisy Contributions

Organized by

IEEE Information Theory Society — Student & Outreach Subcommittee

In conjunction with the 2026 European School of Information Theory (ESIT 2026)

ellis | UNIT MADRID

IEEE Information Theory Society

Nordfjordeid, Norway · June 1-5, 2026

What we'll walk through

01

The big picture

What the competition is and why it exists

02

The two tasks

Learning under noise · Machine unlearning

03

How scoring works

Kaggle leaderboards, the overall ranking, tie-breaks

04

How to take part

Teams, timeline, prizes, rules, and what we need from you

05

Little Demo

Task 1 and 2 Kaggle hands-on demo

The team behind the competition

ORGANIZING COMMITTEE

Alejandro Lancho

Universidad Carlos III de Madrid, Spain

Khac-Hoang Ngo

Linköping University, Sweden

Stefano Rini

National Yang Ming Tung University, Taiwan

TECHNICAL SUPPORT

Rodrigo Oliver Coimbra

Universidad Carlos III de Madrid, Spain

Ricardo Vazquez Alvarez

Universidad Carlos III de Madrid, Spain

uc3m | Universidad Carlos III de Madrid

LINKÖPING UNIVERSITY

國立陽明交通大學

NATIONAL YANG MING CHIAO TUNG UNIVERSITY

SUPPORTED BY

ellis | UNIT MADRID

IEEE Information Theory Society

Data-to-Information Competitions

The Data-to-Information (D2I) initiative

is a series of student competitions at the intersection of **data science, machine learning, and information theory**.

This is the first edition running alongside the ESIT.

We ask one concrete question:

Given noisy, partial, imperfect wireless-channel measurements, can you perform robust localization?

CSI-based localization

2 scored tasks

One overall ranking, one prize pool

Real-world dataset

[DICHASUS datasets]

DICHASUS — real CSI from an industrial testbed

What is CSI?

Channel State Information describes how a radio signal propagates between transmit and receive antennas.

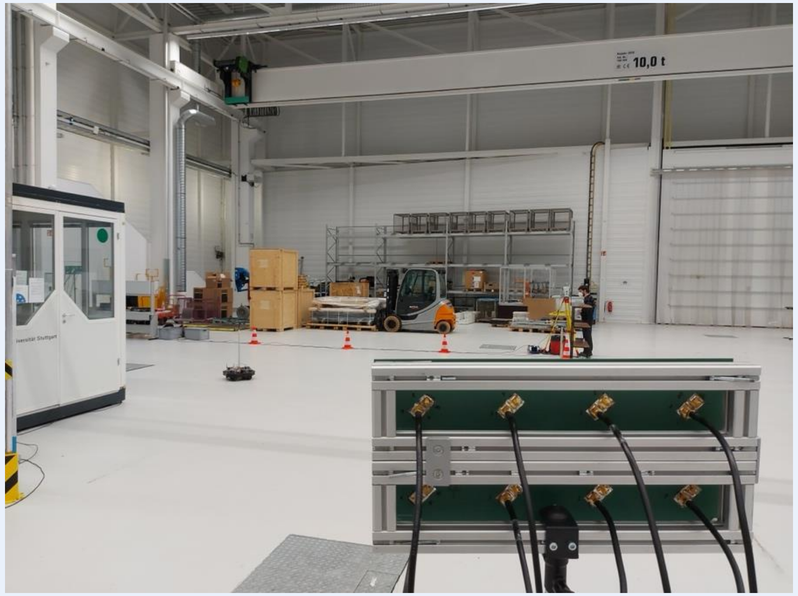
It encodes geometry, multipath, scattering, attenuation, motion, i.e., everything you'd need to recover position.

What is DICHASUS?

An open CSI dataset collected at the **Arena2036 research campus** (University of Stuttgart) → a multi-antenna indoor testbed with calibrated ground-truth positions.

Arena2036 testbed schematic

Four distributed receive arrays in an L-shape;
a mobile transmitter following a known trajectory;
every CSI snapshot tagged with its true (x, y, z).



What you'll actually do

TASK 1

Learning under noisy conditions

Train a localization model from training data whose CSI *and* position labels have been deliberately corrupted with realistic artifacts:

E.g., label drift, phase offsets, gain perturbations, antenna dropout.

Predict `(x, y)` positions for a held-out test set. Lower error wins.

TASK 2

Machine unlearning

Start from a pretrained model that has memorized a known subset of **corrupted users** — and remove their influence *without retraining from scratch.*

You're given a tight fine-tuning budget. Show that the unlearned model behaves like one that never saw the bad data.

Both tasks share the same dataset family (DICHASUS), but use different recordings, different splits, and different metrics.

Learning under noisy conditions

WHAT GETS CORRUPTED

Position labels

Observed positions drift over time

CSI measurements

phase offsets, gain perturbations, measurement noise, frequency-bin dropout, etc.

SCORING

Combined localization loss

½ · MEDE + ½ · R90

`MEDE` · mean 2D Euclidean error.

`R90` · 90th-percentile of those errors.

Lower is better. *Models that are accurate on average but occasionally catastrophic get penalized by R90.*

$$e^{(n)} = \| \hat{\mathbf{y}}^{(n)} - \mathbf{y}^{(n)} \|_2$$

$$\mathrm{MEDE} = \frac{1}{N_{\text{test}}} \sum_{n=1}^{N_{\text{test}}} e^{(n)}, \qquad R90 = \mathrm{quantile}_{0.9}\{e^{(n)}\}$$

Machine unlearning

THE SETUP

You receive a pretrained localization model. It was trained on:

- a Retain set (clean / mildly noisy), and
- a Forget set (heavily corrupted) the model has effectively memorized.

Your job: apply an unlearning procedure

The model should behave on Forget samples as if it had never seen them.

→ A final logistic regressor must be able to distinguish Retain vs Forget samples

SCORING

Membership-inference accuracy

A fixed logistic regression is trained on per-sample errors to predict `is_forget`.

If unlearning worked, the classifier can retain and forget samples.

Higher accuracy = better unlearning.

Baseline (naive fine-tune): ~0.605.
From-scratch reference: ~0.843.

Kaggle Task 1 Kaggle Task 2

If you've never used Kaggle before

01

Create a Kaggle account

One account per person. Kaggle's rules forbid multiple accounts — and we enforce that.

02

Join each competition

Two Kaggle pages: one for Task 1, one for Task 2. Accept the rules on both.

03

Form your team in Kaggle

Send / accept team invites on the competition page. Team identity must match across Task 1 and Task 2.

04

Submit a CSV with your predictions

Up to 5 submissions per day. Pick up 1 as your final entries before the deadline.

Kaggle: Public vs. private leaderboard — why it matters?

*The **test set** is one piece of data. Kaggle splits it behind the scenes between the public leaderboard and private leaderboard:*

Public · 30%	Private · 70%
visible during the competition	*revealed at the end — determines final ranking*

You see the public score

every time you submit, treat it as a noisy signal, not the truth.

The private score decides

rankings collapse and reshuffle when the private board reveals.

Don't overfit the leaderboard

tuning hyperparameters against the 30% slice is a known trap.

Combining the two tasks into one ranking

STEP 1 · COMBINED RANK

Take your private-leaderboard rank on Task 1 and on Task 2. **Sum them.** *Lowest sum wins.*

STEP 2 · TIE-BREAK

If two teams tie on summed ranks, we compute, for each task, the **normalized relative difference** between the team's score and the winner's score, then take the average across the two tasks (see Kaggle rules).

Smaller average distance to the top wins.

To enter the overall ranking, your team must (1) submit valid entries to both Task 1 and Task 2, and (2) keep the same team composition across both.

Teams and registration

TEAM SIZE & ELIGIBILITY

Up to 5 people per team. Recommended minimum is 2.

But solo entries are allowed.

At least one team member must be a registered ESIT 2026 and/or belong to a European institution for the team to be prize-eligible.

Your team must stay consistent across Task 1 and Task 2 → different team compositions = no overall ranking.

HOW TO REGISTER

1. Fill in the registration form by **June 3, 2026**.
2. Team formation finalized in person at ESIT, **June 4**.
3. Register your team on Kaggle (both task pages).

After June 3, new teams can still form if they reach the 3-person minimum, or new arrivals can join in solo mode.

The calendar at a glance

June 3
Registration deadline
individual or team

June 4
Team formation

June 4-25
Build & submit on Kaggle
open submissions

June 26
Finalists announced
top teams notified

July 7
Live presentations
10 min per team

July 8
Winners announced

All deadlines are end-of-day Norway time.

Prize pool

Best Team	Second Place	Third Place	Junior Prize
500 €	250 €	150 €	100 €

Youngest team that impresses the judges

Prizes are awarded based on the combined ranking. To receive a prize, at least one team member must have attended ESIT 2026. Final amounts and any additional non-monetary prizes will be confirmed on the official competition website: https://data-to-information.github.io/esit2026/

Two things every winner must do

01

Present at the live event

Winning teams have to give a **10-minute presentation** of their solution at the official results event on July 7.

At least one team member must present.
No presentation, no prize.

02

Share reproducible code

Winners hand over the code that produced their final submissions, plus enough documentation to rerun it.

We re-run it. If results aren't reproducible, or if the rules were broken (multi-account, prohibited data, leakage), the team is removed from the leaderboard and the next eligible team is promoted (and goes through the same check).

Four places to bookmark right now

The website

`data-to-information.github.io/esit2026`

Single source of truth for rules, dates, dataset, and updates.

The Slack workspace

`Linked from the website`

*Where the action happens (announcements, Q&A, team-formation). **Join today!***

The Kaggle pages

`kaggle.com → Task 1 & 2 →`

*Where you submit. URLs go live on June 4th → **watch Slack!***

The registration form

`Linked from the website`

Closes June 3. Don't put this off.

✦ *Join Slack today, even before you're registered* ✦

Questions?

Then come build something we don't expect.

Join Slack

Register by June 3

See you on the leaderboard