Zero-to-Hero: Knowledge-guided Unsupervised Active Learning for EEG-based Detection of Interictal Epileptiform Discharges

Shen Liang[1], Cihang Yu[2], Qitong Wang✉[3], Delphine Roussel[4], Stephen Whitmarsh[5], Katia Lehongre[5], Alexandra Levchenko[6], Vincent Navarro[5], and Themis Palpanas[1]

[1] LIPADE, Université Paris Cité, Paris, France
shen.liang@u-pariscite.fr, themis@mi.parisdescartes.fr
[2] ENSEA, Cergy, France
cihang.yu@ensea.fr
[3] School of Engineering and Applied Sciences, Harvard University, Cambridge, United States
qitong@seas.harvard.edu
[4] Centre Institut de l'Audition, Institut Pasteur, Paris, France
delphine.roussel@pasteur.fr
[5] Sorbonne Université, Institut du Cerveau (ICM), Inserm, CNRS, APHP, Pitié-Salpêtrière Hospital, Paris, France
{stephen.whitmarsh, katia.lehongre, vincent.navarro}@icm-institute.org
[6] Isep, LISITE, Issy-les-Moulineux, France
alexandra.levchenko@isep.fr

Abstract. In electroencephalography (EEG) data, fast, high-amplitude transients called Interictal Epileptiform Discharges (IEDs) are highly valuable for epilepsy diagnosis and clinical research. Despite advances in deep learning-based IED detection, studies on reducing manual labeling of training data remain limited, particularly in selecting the most informative examples to label, a challenge largely due to data imbalance and heterogeneity. In this work, we propose Zero-to-Hero (Z2H), an Unsupervised Active Learning (UAL) pipeline that starts from a *completely unlabeled* EEG dataset to select high-quality labeled data with a limited manual labeling budget. With a knowledge-guided initialization module, an actively enhanced positive-unlabeled transduction module, and a knowledge-guided data filtering module, Z2H can achieve an optimal trade-off between data heterogeneity, balance, and label correctness. The data selected by Z2H can be used to both train a small IED detection model from scratch and fine-tune a pre-trained EEG Foundation Model (FM). Z2H can outperform multiple baselines and enable real-time user interaction. When paired with an FM, Z2H can reduce manual labeling by 95% while yielding similar or better IED detection accuracy.

Keywords: Interictal Epileptiform Discharge (IED) · electroencephalography (EEG) · Unsupervised Active Learning (UAL) · domain knowledge

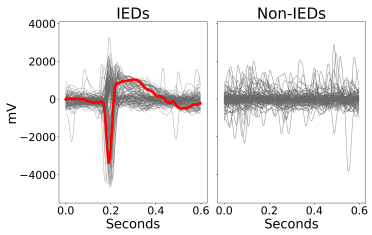

Fig. 1: Butterfly plots of 100 IEDs (with one shown in red) and 100 non-IEDs. Here we only plotted a single channel, while our method is adapted to *multi-channel* EEG.

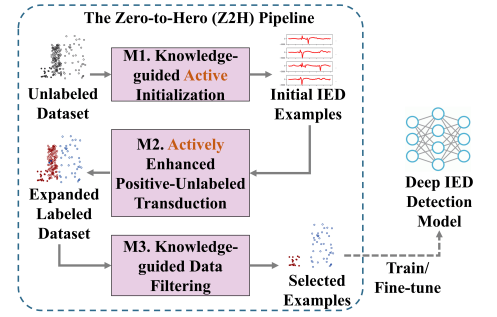

Fig. 2: A summary of the Z2H pipeline, with three modules M1-M3.

1 Introduction

Epilepsy, the second most burdensome neurological disorder, is characterized by recurrent seizures. For some drug-resistant patients with focal epilepsy, surgery targeting the epileptogenic zone might be an option. As part of the clinical evaluation, recordings from implanted electroencephalography (EEG) electrodes are used to identify the seizure onset zone [12]. Such recordings might include Interictal Epileptiform Discharges (IEDs) (Fig. 1a), which are fast, high-amplitude EEG transients, typically followed by slow waves [5, 31]. IEDs are valuable for epilepsy diagnosis and clinical research [26]. However, manual detection of IEDs within EEG recordings lasting days to weeks remains time-intensive.

To date, numerous deep learning models have been proposed for IED detection, mostly formulating IED detection as two-class (IED/non-IED) classification [7, 8, 21, 24, 29]. Most of these models are fully supervised [21, 24, 29], trained on fixed training sets that can be highly costly to manually label. Some reduce human effort by semi-supervision [7, 24] or data augmentation [7, 8], yet an important question remains unanswered: *Given an unlabeled EEG training set, which examples to select for manual labeling to minimize human effort while retaining high IED detection accuracy?*

This question translates to **Unsupervised Active Learning (UAL)**, a type of Active Learning (AL) that aims to select a limited number (known as the AL *budget*) of candidate examples for manual labeling, starting from an *unlabeled* dataset. Existing UAL methods [10, 14–17, 20, 27] typically favor examples that best reconstruct all or part of the dataset in a latent space. These methods have limited capacity to address two challenges within IED detection. First, IEDs are rare events, leading to significant **data imbalance** [29]. Second, both IEDs and non-IEDs exhibit **data heterogeneity**: On the one hand, despite their relatively well-defined morphologies [5], as shown in Fig. 1 *left)*, IEDs can entail a degree of diversity even for a single subject. On the other hand, as shown in Fig. 1 *right)*, the non-IED class is not well-defined and has a variance far greater than that of the IEDs. Notably, while some existing UAL methods [15,27]

address data heterogeneity and imbalance with K-means in the latent space to preserve local clusters, as shown in Fig. 1 *right)*, the heterogeneity of the non-IEDs is characterized by highly random patterns, not by defined clusters. Thus, K-means (and other clustering methods) often have limited effectiveness, and we need to *increase the selected data size* to fully represent the non-IED space. However, existing UAL methods only produce training sets that are purely manually labeled and disregard any data that could be labeled automatically via semi-supervision. Thus, their selected data size is limited to the AL budget, which is often small. While these methods can be fused with a separate semi-supervised learner to expand the data size in a posterior manner, the lack of *built-in* semi-supervision can still limit their effectiveness.

In this work, we propose **Zero-to-Hero** (**Z2H**, Fig. 2), a UAL pipeline that starts from an EEG dataset with *zero* labeled examples and selects a set of labeled data to train/fine-tune an accurate IED detection model (hence *Zero-to-Hero*). Z2H includes three modules: 1) A knowledge-guided active initialization module (M1) that is *intentionally* biased to the IED class to address data imbalance; 2) A heterogeneity-aware active Positive-Unlabeled (PU) [18,30] transduction module (M2) that labels more data both *manually* and *automatically*, thereby expanding the labeled data size beyond the AL budget; 3) A knowledge-guided data filtering module (M3) that removes likely mislabeled data while enhancing data balance, yielding a high-quality set of labeled examples.

In summary, our contributions in this work are as follows:

• We propose Z2H, which is, to the best of our knowledge, the first UAL method tailored for IED detection, adopting a knowledge-guided, active-PU approach to achieve an optimal trade-off between data balance, heterogeneity, and label correctness. Z2H is *model-agnostic*, and the selected data can be used to build any *user-defined* IED detection model, including both training a small model from scratch and fine-tuning a pre-trained EEG Foundation Model (FM).

• Experimentally, Z2H can outperform multiple existing UAL methods while enabling real-time user interaction. When paired with an FM, it can reduce human labeling by 95% with little to no decay in IED detection accuracy.

2 The Proposed Method

2.1 M1: Knowledge-guided Active Initialization

In Module M1 of Z2H, starting with an unlabeled dataset, we use AL to query the user for the labels of some initial seed examples. To handle data imbalance, we aim to *query examples likely to be IEDs first* to create an *intentional* bias to the IED class, using two pieces of domain knowledge:

1) IEDs are usually of high amplitude [5] compared to background activity, and likely have higher **Peak-to-Peak (P2P) amplitudes** (the difference between the highest and lowest points), as shown in Fig. 3a.

2) Despite a degree of heterogeneity, IED morphology is generally well-defined by a sharp spike, followed by a slow wave [5]. This suggests that the IED space

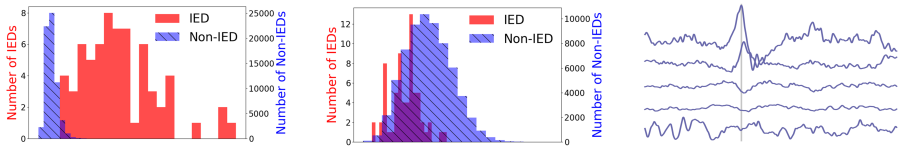

Fig. 3: Rationale behind the Amp_{Dist} sampling strategy. (a)(b) Histograms of P2P amplitudes and NN distances of single-channel EEG signals from a mouse. IEDs tend to have higher amplitudes and lower NN distances. (c) An IED in a multi-channel EEG snippet. Its spike is present in only part of all channels.

is likely more compact than the non-IED space which entails more randomness. Consequently, the **Nearest Neighbor (NN) distance** of an IED, i.e., its distance (here we use z-normalized Euclidean Distance) to its nearest neighbor, is not likely to be large, as shown in Fig. 3b.

With this knowledge, we propose a *single-channel* $\boldsymbol{Amp_{Dist}}$ **sampling strategy**: *For a single-channel EEG dataset, query the examples in P2P amplitude descending order, but skip those with NN distances larger than some threshold* δ.

Here, we use P2P amplitudes to decide the primary query order, relegating NN distances to data filtering, as P2P amplitudes are more discriminative than NN distances. As shown in Fig. 3a and b, the histograms of P2P amplitudes are more separated between IEDs and non-IEDs than NN distances, as IEDs vary more in morphology than in amplitude.

Next, we extend Amp_{Dist} to *multi-channel* EEG as follows.

Step 1: *For a multi-channel EEG dataset with* n *examples, denoting the P2P amplitude of the* j*-th channel of the* i*-th example as* $a_{i,j}$*, let the mean-normalized P2P amplitude of the* j*-th channel of the* i*-th example be* $\tilde{a}_{i,j} = na_{i,j} / \sum_{i'=0}^{n-1} a_{i',j}$*. We define the normalized P2P amplitude of the* j*-th channel as* $\max_{i'} \tilde{a}_{i',j}$*. Rank all channels in normalized P2P amplitude-descending order.*

Step 2: *In the aforementioned order, go through each channel in a round-robin manner; for each channel, query one example using the single-channel* Amp_{Dist} *before moving on to the next channel, until an IED is queried in a certain "target channel". Afterward, disregard all other channels and continue applying the single-channel* Amp_{Dist} *on the target channel.*

Here, we opt for a single target channel rather than all channels, as IED spikes can often be present in only certain channels (Fig. 3c). In Step 1, we mean-normalize the P2P amplitudes to account for differing baseline amplitudes across channels. Then, for each channel, we use the amplitude of the example with the highest amplitude in that channel to represent the channel's amplitude, as such an example is most likely to be an IED. However, the channel ranking obtained this way can be interfered with by high-amplitude noise. Thus, we use the round-robin approach in Step 2 to robustify the choice of the target channel.

Efficiency-wise, in interactive AL, we mainly focus on *user interaction response time* [19], i.e., the delay between two consecutive user queries. For Amp_{Dist}, by pre-computing and pre-sorting the amplitudes and distances, we can minimize response time by using simple lookups to find the next example to query.

2.2 M2: Actively Enhanced Positive-Unlabeled (PU) Transduction

From M1, we could get a set PL of manually labeled positive (i.e., IED) examples and a set NL of negative (i.e., non-IED) ones. We denote the set of all remaining unlabeled examples as U. In M2, we label (part of) U examples with both *manual labeling* and *automatic inference* to expand the labeled data size beyond the AL budget, thus handling data heterogeneity, especially the highly random non-IEDs. To do so, we draw on *Positive Unlabeled (PU)* [18] transduction, a type of one-class semi-supervision that jointly uses PL and U to label U. We favor this one-class approach over using both PL and NL, for M1 created an intentional bias to the IEDs. Thus, NL cannot accurately represent the negative space.

Concretely, we use the one-class **Self-Training 1-Nearest-Neighbor (ST-1NN)** [30] algorithm (Fig. 4) for PU transduction, with two phases:

ST-1NN Phase 1: *Initialize an ordered set R with PL examples; iteratively find the U example with the smallest NN distance to R (measured by z-normalized Euclidean distance over all channels in our case) and move it from U to R. Examples moved into R earlier are considered more likely to be positive.*

ST-1NN Phase 2: *Decide the decision boundary with a* ***stopping criterion*** *[9, 30] splitting R into two: the first positive and the second negative.*

While simple, ST-1NN is generally suitable for use cases such as IED detection, where the positive class is relatively compact, while the negative class is highly diverse and without well-defined clusters. However, ST-1NN alone often yields sub-optimal results. In particular, besides a diverse negative class, IED detection also involves a heterogeneous positive class. Despite relatively well-defined morphologies, IEDs can exhibit a degree of diversity even within a single patient. ST-1NN largely disregards this diversity, as in Phase 2, it draws on a **Positive/Negative Conglomeration (PNC)** assumption: *in the ordered set R, the positive and negative examples conglomerate at the beginning and the end, with a single boundary between the two defined by the stopping criterion.* This often fails to hold. For instance, in Fig. 4b, with heterogeneous initial PL examples 0, 1, 2, the neighborhoods around positive examples 6 and 7 are different, with 7 closer to negative examples 8 and 9 than 6 is to positive example 11. As a result, 11 (whose NN in U was 6 when 11 was moved from U to R) and 12 were moved to R after negative examples 8 (whose NN was 7), 9 and 10.

In response, we draw on the *Self-Training Tree (ST-Tree)* [19] (Fig. 4c), where each example is a node with an edge between it and its NN in R when it was moved from U to R. ST-Tree naturally encodes data heterogeneity, as examples in different *branches* (paths from an initial PL example to a leaf node) tend to diverge in morphology. Thus, while the PNC assumption is too rigid, the following **Branch-wise PNC (BPNC)** assumption is far more realistic: *each*

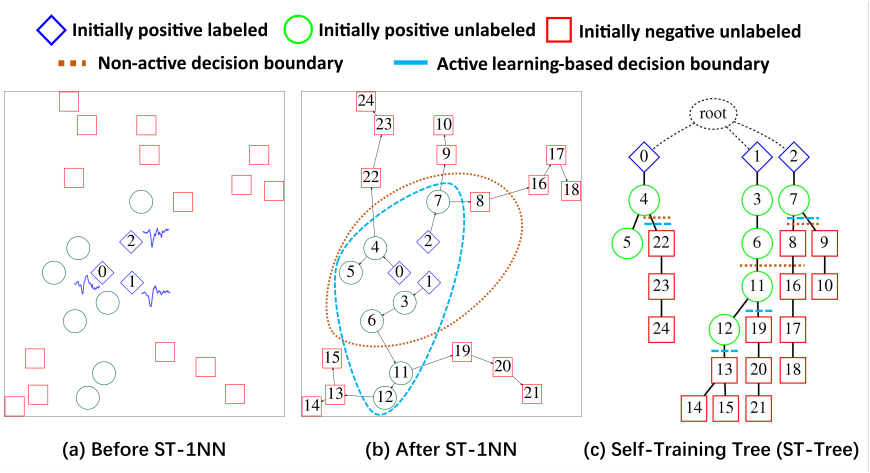

Fig. 4: An illustration of ST-1NN. (a)(b) are the states before and after ST-1NN. Numbers in the nodes are in the order in which they were moved to the ordered set R. The edges link them to their NNs in R when they were moved. Drawing on the ST-Tree in (c), our AL-based approach can yield a better decision boundary.

branch has 1) all positive nodes, or 2) positive and negative nodes conglomerated at the beginning and the end, separated by a branch-wise decision boundary. For instance, in Fig. 4c, the branch 0 — 4 — 5 is all positive, while 0 — 4 — 22 — 23 — 24 has a clear branch-wise boundary between 4 and 22. Note that a branch cannot be all negative, as it must start with a PL example.

Based on BPNC, we propose a novel AL-based branch-wise inference method to replace the non-AL-based stopping criterion in ST-1NN Phase 2. As shown in Alg. 1, we actively infer the labels of U examples in as many branches as possible (we will discuss the ordering of the branches later), finding an individualized decision boundary for each branch. For a given branch, we first query from the user the ground truth label of its leaf node, which can be one of 2 cases.

Case 1: The leaf is positive (line 4 of Alg. 1). By BPNC, this branch is most likely all positive.

E.g.: For the branch 0 — 4 — 5 in Fig. 4c, with 5 actively queried as positive, 4 is automatically inferred as positive.

Case 2: The leaf is negative (line 5-line 7). By BPNC, we find the branch-wise boundary via an *active binary search*. Concretely, for each node checked, we query the ground truth labels of it and its parent: if both are positive, then we are in the positive space and need to check the examples after this node; if both are negative, we check those before it. The search ends when we have found a negative node $firstN$ whose parent $lastP$ is positive, with the branch-wise boundary in between.

E.g., in the branch 1 — 3 — 6 — 11 — 12 — 13 — 14, we observe that the leaf node 14 is negative. Following binary search, we query 11 and its parent

Algorithm 1: Active Branch-wise Inference

Input : initial unlabeled examples U, an ST-Tree *tree* from ST-1NN Phase 1
Output: inferred positive and negative examples $infP$ and $infN$

```
1 infP, infN = [], [];
2 while AL budget remains do
3     branch = nextBranch(tree); label = activeQuery(branch.leaf);
4     if isPositive(label) then infP.add(branch);
5     else
6         firstN, lastP = activeBinarySearchBranchwiseBoundary(branch);
7         infP.add(branch.first : lastP); infN.add(tree.subTree(firstN));
8     tree.trim(infP, infN);
```

6 (which are at the midpoint of the branch) and find both to be positive. This indicates that the decision boundary is after 11. Again, following binary search, we then query 13 and its parent 12, and find that 13 is negative and 12 is positive. Thus, we determine 13 as $firstN$ and 12 as $lastP$, with the branch-wise boundary between them. All unlabeled examples before 13 are inferred as positive (i.e., 3). Also, by BPNC, all unlabeled examples in the sub-tree rooted at $firstN$ (i.e., 15) are inferred as negative.

Having checked the current branch, in line 8, we trim part of the sub-tree rooted at its initial PL example, removing all nodes in it with both themselves and all their descendants already labeled (queried or inferred). For instance, having labeled 1 — 3 — 6 — 11 — 12 — 13 — 14 (and 15), we can remove 12, 13, 14, 15, but keep 1, 3, 6, 11 as they have unlabeled descendants 19, 20, 21.

As for the order in which the branches are queried, we aim to maximize the number of examples inferred per query. Concretely, by finding $firstN$ in one branch $\mathcal{B}$, we can potentially trim the entire sub-tree $\mathcal{T}$ rooted at its initial PL example. We denote the number of U nodes in $\mathcal{T}$ as u. Also, we need $O(lg(l))$ queries to find $firstN$, l being the number of examples yet to be labeled in $\mathcal{B}$. We score $\mathcal{B}$ by its estimated number of examples inferred per query, i.e. $u/lg(l)$. The branch with the highest score is queried next.

Note that in [19], we proposed a similar AL-based post-processing method for ST-1NN. However, that method can only select one or several non-AL-based ST-1NN configurations from a set of candidates, whereas in M2 of Z2H, we can surpass the non-AL-based alternatives. Critically, [19] relies on the ensemble of non-AL-based candidates to determine the decision boundary, while in M2, the branch-wise boundaries are independent of any non-AL-based stopping criteria. As we will show in Section 3.2, compared to these non-AL-based stopping criteria, our AL-based branch-wise method can significantly enhance the automatic inference accuracy of ST-1NN. Efficiency-wise, our method has a worst-case user interaction response time of $O(|U|)$, where $|U|$ is the number of U examples before ST-1NN. See [1] for a proof of this complexity. As will be shown in Section 3.2, the delays are usually negligible.

2.3 M3: Knowledge-guided Data Filtering

M2 expanded the labeled data beyond the AL budget, and our AL-based decision boundary greatly enhanced inference quality. Still, some misinferred data may remain. To filter them, in M3, we again draw on the domain knowledge that IEDs tend to have high amplitudes, and *filter all examples inferred as IEDs/non-IEDs but with P2P amplitudes (in the target channel found in M1) below/above some threshold* γ. The remaining labeled data is used to train/fine-tune a user-defined deep IED detection model (called the **core model**).

While simple, M3 is effective for three reasons: 1) Rooted in domain knowledge, it can further enhance **label correctness**. 2) It can still preserve a large number of automatically inferred examples in M2, retaining the ability to break the **data size** limit imposed by the AL budget. 3) Besides duly filtering misinferred examples, M3 can filter some correctly inferred examples, which may seem undesirable, but can actually improve data quality. This is because M3 typically filters a higher proportion of (correctly inferred) non-IEDs than IEDs, enhancing the **data balance** of the resulting dataset. Specifically, we have the following theorem:

Theorem 1. *Suppose that IEDs account for* $x\%$ *of all examples in the dataset, where* $x < 50$*. Assume that the distributions of correctly inferred IEDs and non-IEDs are the same as those of all IEDs and non-IEDs. Let* A_P *and* A_N *denote the P2P amplitudes of a randomly chosen IED and a randomly chosen non-IED. Define* $F_P(a) = P(A_P < a)$ *and* $F_N(a) = P(A_N < a)$ *as the proportions of IEDs and non-IEDs whose P2P amplitudes are lower than some threshold* a*. Assume that* A_P *first-order stochastically dominates* A_N*, namely,* $\forall a, F_P(a) \leq F_N(a)$*. Assume further that the relevant distributions are continuous and that the mixture distribution of all P2P amplitudes is strictly increasing in the relevant range. Let* a_t *be a threshold satisfying* $F_P(a_t) + F_N(a_t) = 1$*. Suppose that the M3 filtering threshold* γ *is set to the* y*-th percentile of the P2P amplitudes of all examples. Let* r_P *and* r_N *be the proportions of correctly inferred IEDs and non-IEDs that are filtered by M3 among all correctly inferred IEDs and non-IEDs, respectively. Then,*

$$F_P(a_t) \leq 0.5 \tag{1}$$

and

$$r_N > r_P \iff y < (100 - x) - (100 - 2x)F_P(a_t). \tag{2}$$

From Theorem 1, we have the following lemma.

Lemma 1. ***Lemma 2.*** *Under the same assumptions as in Theorem 1, if* $x \ll 50$ *and* $F_P(a_t) \ll 0.5$*, then* $r_N > r_P$ *holds for most choices of* y *below* $100 - x$*, except when* y *is very close to* $100 - x$*.*

See [1] for the proofs of Theorem 1 and Lemma 1, which indicate that if the IEDs generally have greater P2P amplitudes than non-IEDs, with good separation between the distributions of the two, and that the proportion of IEDs in the dataset is very small, then γ can usually be set to a very wide range of values to

Table 1: Data statistics. $|P|$ and $|N|$ are the numbers of IEDs and non-IEDs. Unless an *inter*-subject setting is explicitly stated, all experiments were done *independently* for each subject, without referencing data from other subjects.

Dataset	Sampling Rate (Hz)	Length (seconds)	Number of Channels	Train (initially unlabeled)			Test		
				|P|	|N|	|P|/|N|	|P|	|N|	|P|/|N|
H1	512	1.5	5	1782	93788	0.019	357	18758	0.019
H2				2769	95981	0.029	554	19197	0.029
H3				10178	84477	0.120	2036	16896	0.121
H4				3043	93137	0.033	609	18628	0.033
H5				5526	90297	0.061	1105	18060	0.061
H6				2561	87229	0.029	512	17446	0.029
H7				8089	72915	0.111	1618	14583	0.111
H8				2208	90339	0.024	442	18068	0.024
MO1		0.09		2639	56048	0.047	528	11210	0.047
MO2				394	56666	0.007	79	11334	0.007
MO3			4	66	66110	0.001	14	13222	0.001

improve data balance among the correctly inferred examples. As we will show in Section 3.1, it is an optimal trade-off between label correctness, data size and data balance that enables Z2H to yield high-quality labeled data.

3 Experimental Evaluation

We evaluate Z2H using data from two groups of real-world datasets: 1) **H1-H8:** 24-hour EEG data from 8 human patients. All intracranial EEG procedures for the requisition of these data were performed according to clinical practice and the epilepsy features of the patients [13]. Patients gave written informed consent (Project C11-16 conducted by INSERM and approved by the local ethics committee, CPP Paris VI). 2) **MO1-MO3:** 1-hour EEG data from 3 mice, for which experimental procedures were conducted in compliance with the European Committee Council Directive (2010/63/EU) and received approval from the French Ministry for Research and the local Ethical Committee. Following the practice of [29], we bandpass-filtered (1-50 Hz) and downsampled (from 4,096 to 512 Hz) the data before using sliding windows (1.5s for humans, 0.09s for mice) to segment the continuous recordings into individual examples, which are split into training and test sets at $5:1$. Table 1 summarizes the data statistics. See [1] for detailed data acquisition and preprocessing protocols.

Z2H can be fused with any user-defined core model for IED detection. Here we consider the following two groups of core models: 1) *AiED [21], iEDeal [29]*, which are small models trained from scratch with data selected by Z2H. 2) *EEGPT [28], Neuro-GPT [4]*, which are pre-trained EEG Foundation Models (FMs) using Z2H for fine-tuning data selection. See [1] for their detailed hyperparameter settings.

We ran our experiments with an Intel Xeon Platinum 8468 CPU, an NVIDIA H100 GPU and 512 GB RAM on the Jean Zay supercomputer. All code is Python. Next, we will first compare Z2H with rival UAL methods, and its ablations, in both *intra-* and *inter-subject* settings. We then explore the effects of parameter settings and design choices in each module, as well as user-interaction efficiency. All our source code and the mouse data MO1-MO3 are available at [1].

3.1 Comparison with Baselines and Ablation

We compare Z2H with the following baseline and ablation methods.

- **Fully-Supervised (FS):** manually label *all* available data. Intuitively, this is an *upper bound* of any UAL method, but it can be outperformed sometimes.
- **Random:** a random sampling strategy, averaged over 10 seeds.
- **ALSL-U [16], ALLG [20], ALMS [14], DUAL-S3P [15]:** UAL methods.
- **Z2H-Only-M1, Z2H-M1-M2:** Z2H ablations. Z2H-Only-M1 only selects examples manually labeled in M1; Z2H-M1-M2 omits data filtering in M3.

We set the total AL budget to $5\%, 10\%, \ldots, 30\%$ of all training examples. Z2H has 3 parameters: 1) allocation of AL budget between M1 and M2 (M3 does not have an AL component), for which we use a $7:3$ ratio; 2) the NN distance threshold δ in M1, which we set to the 50-percentile (median) of all NN distances; 3) the amplitude threshold γ in M3, which we also set to the 50-percentile. These parameters are shared by Z2H-Only-M1 and Z2H-M1-M2, except that for Z2H-Only-M1, we reallocate the AL budget intended for M2 to M1 for a fair comparison. See [1] for the parameter settings of the other methods.

We compare the methods above by pairing them with the core deep models mentioned earlier and evaluate them by their online IED detection F1-scores on test sets. Due to page limits, we cannot present all raw results here. Instead, we draw on the Critical Difference (CD) diagram [6] to compare their statistical differences (Fig. 5). Specifically, we compare the methods across all combinations of core models, human/mouse subjects, and total AL budget, using the Friedman test and the Wilcoxon signed-rank test [3, 19]. In each CD diagram, methods to the right rank higher on average. Should several methods show no statistically significant difference, they would be grouped into the same clique (not present in our results). Besides the CD diagrams, we focus on the case with the smallest total AL budget, just 5% of all examples, as this reduces human labor the most (by 95%). The raw F1-scores in this case are shown in Table 2.

As shown in Fig. 5, except Z2H and its ablations, the best rival UAL method is DUAL-S3P, likely because it explicitly addresses data heterogeneity and imbalance with K-Means. However, it is still limited in its capacity to handle the high level of randomness in the non-IED space. By contrast, Z2H significantly outperforms all other non-FS rivals, as is corroborated by the raw results in Table 2. Specifically, Z2H generally performs better when paired with pre-trained FMs (especially EEGPT) than with models trained from scratch, highlighting the superiority of FMs. Also, the FMs with Z2H often have results very close to

Table 2: F1-scores of core models on the test sets with a total AL budget of 5% of all data, using Z2H, its ablations, and rival methods for training/fine-tuning data selection. Best results except for *Fully Supervised (FS)* are shown in **bold**; second bests are underlined. "-" indicates failed training or invalid fields.

Method	H1	H2	H3	H4	H5	H6	H7	H8	MO1	MO2	MO3	Avg. F1	Avg. Rank
Core Model: AiED													
FS	0.66	0.59	0.83	0.70	0.87	0.81	0.90	0.75	0.92	0.96	-	0.80	-
Random	0.51	-	0.72	0.45	0.79	0.55	0.82	0.60	0.86	-	-	0.66	5.62
ALSL-U	0.55	-	0.73	-	0.75	0.56	0.47	-	0.84	-	-	0.65	6.00
ALLG	0.54	-	**0.74**	0.49	0.78	0.58	0.83	0.62	0.86	0.87	-	0.70	4.11
ALMS	0.56	-	0.72	0.49	0.80	0.51	0.82	0.61	**0.87**	-	-	0.67	4.25
DUAL-S3P	0.57	0.16	0.67	0.49	0.80	0.48	0.79	0.67	0.86	0.89	-	0.64	4.50
Z2H-Only-M1	0.60	0.25	**0.74**	0.53	0.80	0.64	0.83	0.70	0.81	0.91	-	0.68	2.70
Z2H-M1-M2	**0.63**	0.39	0.63	0.58	0.68	0.65	0.72	0.70	0.72	0.89	-	0.66	4.30
Z2H (ours)	0.58	**0.43**	**0.74**	**0.66**	**0.82**	**0.77**	**0.86**	**0.73**	**0.87**	**0.94**	**0.78**	**0.74**	**1.18**
Core Model: iEDeal													
FS	0.84	0.75	0.93	0.87	0.94	0.93	0.96	0.88	0.91	0.79	0.60	0.85	-
Random	0.56	0.39	0.75	0.58	0.78	0.61	0.84	0.71	0.87	0.59	0.47	0.65	4.64
ALSL-U	0.55	0.22	0.75	0.46	**0.80**	0.63	0.84	0.71	0.86	0.76	-	0.66	4.70
ALLG	0.55	0.39	0.74	0.53	**0.80**	0.59	0.84	0.69	0.86	0.47	-	0.65	5.20
ALMS	0.54	0.39	0.75	0.58	0.79	0.59	0.85	0.71	0.80	0.43	0.58	0.64	5.00
DUAL-S3P	0.58	0.40	0.71	0.53	0.79	0.57	0.82	0.73	**0.88**	0.76	0.65	0.67	4.55
Z2H-Only-M1	0.63	0.43	0.77	0.62	**0.80**	0.65	0.84	0.75	0.87	0.81	**0.67**	0.71	**2.27**
Z2H-M1-M2	**0.71**	**0.58**	0.68	**0.73**	0.74	**0.82**	0.78	0.79	0.71	0.82	0.50	0.71	4.09
Z2H (ours)	0.61	0.37	**0.86**	0.60	0.79	0.70	**0.90**	**0.84**	**0.88**	**0.83**	0.61	**0.73**	2.45
Core Model: EEGPT (Foundation Model)													
FS	0.84	0.60	0.90	0.84	0.93	0.86	0.95	0.89	0.91	0.86	0.73	0.85	-
Random	0.77	0.59	0.87	0.82	0.90	0.76	0.93	0.87	0.89	0.82	0.13	0.76	4.55
ALSL-U	0.79	0.60	**0.88**	0.78	**0.91**	0.76	**0.94**	**0.88**	0.89	0.80	0.35	0.78	3.82
ALLG	0.77	0.58	0.87	0.80	0.90	0.79	0.93	0.86	**0.90**	0.82	-	0.82	4.40
ALMS	0.81	0.58	0.87	0.82	**0.91**	0.77	0.93	0.86	0.89	0.85	0.64	0.81	3.27
DUAL-S3P	0.79	0.62	0.87	0.81	0.90	0.77	0.93	0.85	0.89	0.83	0.80	0.82	3.82
Z2H-Only-M1	0.36	0.66	**0.88**	**0.84**	**0.91**	0.81	0.93	0.87	**0.90**	0.83	0.78	0.80	2.55
Z2H-M1-M2	0.81	0.43	0.68	0.76	0.77	0.70	0.78	0.79	0.62	0.83	0.60	0.71	6.73
Z2H (ours)	**0.82**	**0.70**	0.87	**0.84**	**0.91**	**0.83**	**0.94**	**0.88**	0.88	**0.88**	**0.83**	**0.85**	**1.73**
Core Model: Neuro-GPT (Foundation Model)													
FS	0.87	0.71	0.92	0.87	0.94	0.93	0.95	0.89	0.86	0.94	0.25	0.83	-
Random	0.82	0.65	**0.90**	0.82	0.91	0.89	0.93	0.84	0.76	0.57	0.02	0.74	4.00
ALSL-U	0.83	0.65	0.89	0.79	0.90	0.89	0.92	0.83	0.76	0.37	0.08	0.72	5.09
ALLG	0.82	0.66	**0.90**	0.81	0.91	0.88	0.92	0.83	0.76	0.52	-	0.80	4.40
ALMS	0.84	0.65	**0.90**	0.84	0.91	0.88	0.93	0.87	0.59	0.34	0.35	0.74	3.82
DUAL-S3P	0.81	0.69	0.89	0.83	**0.92**	0.88	0.93	0.87	0.69	0.67	0.26	0.77	3.73
Z2H-Only-M1	0.80	0.64	0.89	**0.85**	0.90	**0.93**	**0.94**	0.86	0.58	0.76	0.36	0.77	4.09
Z2H-M1-M2	**0.86**	0.60	0.70	0.75	0.67	0.82	0.79	0.86	0.64	**0.91**	0.13	0.70	5.91
Z2H (ours)	0.85	**0.71**	0.89	**0.85**	**0.92**	**0.93**	**0.94**	**0.88**	**0.81**	0.87	**0.55**	**0.84**	**1.45**

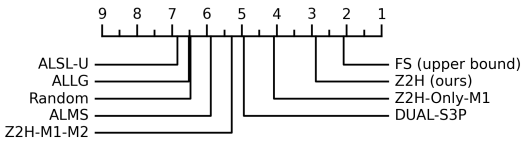

Fig. 5: CD diagram of Z2H, its ablations and rival UAL methods. Methods to the right are better. *FS* denotes fully supervised training with all available data manually labeled and is an intuitive *upper bound* for UAL methods.

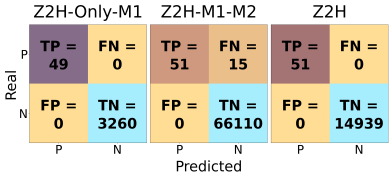

Fig. 6: Confusion matrices of the selected training data on MO3 (not to be confused with the online IED detection results on the test set in Fig. 5 and Table 2).

FS with just 5% of manual labeling, and in certain cases (e.g., MO3), it greatly outperforms FS, thanks to the power of Z2H to select the most informative examples. Among the Z2H ablations, Z2H-Only-M1 generally outperforms Z2H-M1-M2, which seems to suggest that M2 has a negative effect. However, we now explore *why* Z2H performs well and how M2 has a positive impact on it.

Specifically, we examine MO3, the most imbalanced subject with an extreme positive-to-negative ratio of 0.001. We show the confusion matrices of the selected training data by all variants of Z2H in Fig. 6. Here, Z2H-Only-M1 has the best data balance, yet its performance is limited by the labeled data size, which is restricted to the AL budget. This prevents Z2H-Only-M1 from adequately capturing the heterogeneity of the negative space. On the other hand, Z2H-M1-M2 greatly expands the data size, but is affected by misinferred data and poor data balance. By contrast, Z2H, as we claimed in Section 2.3, has the best trade-off between data size, data balance, and label correctness. This highlights the contribution of M2, which enables Z2H to exceed the AL budget limit and achieve far greater data size, thus better handling data heterogeneity.

The reader may have noticed that with iEDeal as the core model, Z2H has weaker results than when paired with the other core models, while Z2H-M1-M2 is stronger than usual, outperforming Z2H in multiple cases. This is likely because iEDeal is tailored to imbalanced data, with a loss function that simulates the F-score as well as a negative sampling strategy. Thus, the gains of Z2H over Z2H-M1-M2 in terms of data balance are not as pronounced for iEDeal as for other core models.

[*Inter*-subject results] So far, we have focused on an *intra*-subject setting where the training and testing were done on the same subject. We now examine an *inter*-subject setting where the training and testing are on different subjects, offering greater real-world utility. Here, we apply a leave-one-out strategy for the human data, training the core model on all humans except one and using the latter for testing. Due to limited GPU hours, we only consider the total AL budget setting of 5% of all data, and use EEGPT (which generally performed best in the *intra*-subject setting in Table 2) as the core model. As shown in Table 3, where Hi' means the subject Hi is used for testing, Z2H again has

Table 3: F1-scores of EEGPT in an *inter*-subject setting, with a total AL budget of 5% of all data, using Z2H, its ablations and rivals for fine-tuning data selection.

Method	H1′	H2′	H3′	H4′	H5′	H6′	H7′	H8′	Avg. F1	Avg. Rank
FS	0.59	0.30	0.85	0.82	0.85	0.57	0.74	0.84	0.70	-
Random	0.64	0.30	0.85	0.80	0.80	0.49	0.70	0.83	0.68	3.38
ALSL-U	0.67	0.28	0.84	**0.81**	0.75	0.35	0.68	0.84	0.65	4.50
ALLG	0.66	0.24	0.85	0.80	0.70	**0.52**	0.69	0.83	0.66	3.88
ALMS	0.54	0.32	**0.86**	0.80	0.80	0.47	0.66	0.82	0.66	4.00
DUAL-S3P	0.59	0.34	**0.86**	0.80	0.74	0.50	0.74	0.82	0.67	3.50
Z2H-Only-M1	0.66	0.37	0.85	0.78	0.80	0.41	**0.88**	0.78	0.69	3.88
Z2H-M1-M2	0.52	0.11	0.82	0.58	0.65	0.12	0.21	0.85	0.48	7.25
Z2H (ours)	**0.72**	**0.41**	0.85	0.78	**0.82**	0.46	0.66	**0.86**	**0.70**	**3.00**

the best overall results. However, its advantage has been diminished. It is likely that we need to incorporate Z2H with some domain adaptation/generalization technique to fully unleash its potential in the inter-subject setting where the data distributions of the training and testing data can be drastically different, which we will address in our future work.

3.2 Parameter Analysis and Further Experiments

We now examine the effects of parameter settings and the design of each module, before looking into the efficiency of user interaction.

[Module M1] We examine the effect of the NN distance threshold δ in our $\boldsymbol{Amp_{Dist}}$ strategy in M1, as well as some rival methods. For δ, we consider the $10, 20, \ldots, 90$-percentiles on all NN distances. For the rivals of Amp_{Dist}, besides the UAL methods used in Section 3.1, we consider 3 knowledge-guided variants of Amp_{Dist}: ***Dist*** which follows an NN distance ascending order with no consideration of amplitude, ***Amp*** which follows a P2P distance descending order with no NN distance-based skipping, and $\boldsymbol{Dist_{Amp}}$ which follows an NN distance ascending order while skipping examples with P2P amplitudes *below* some threshold, for which we also choose from $10, 20, \ldots, 90$-percentiles and only report the best result among them for fairness to this rival method.

Recall that in M1, we aim to query likely IED examples first. Thus, for the evaluation metric, we use the *precision at k (p@k)*, i.e. the proportion of IEDs among the first k examples queried. The results are shown in Fig. 7 where we focus on $k = 100$. As shown in Fig. 7 *left)*, when δ is between 40 and 60-percentiles, the p@100 value is generally high for most datasets, thus justifying our choice of the 50-percentile in Section 3.1. Note that the trend on H2 differs from the other subjects, where the 50-percentile yields a relatively low p@100, likely due to unusually high noise on H2. In fact, the scores on H2 are generally lower than those on the others.

In Fig. 7 *right)*, among the four knowledge-guide methods, $Dist$ and $Dist_{Amp}$ are generally outperformed by Amp and Amp_{Dist}, justifying our choice to use

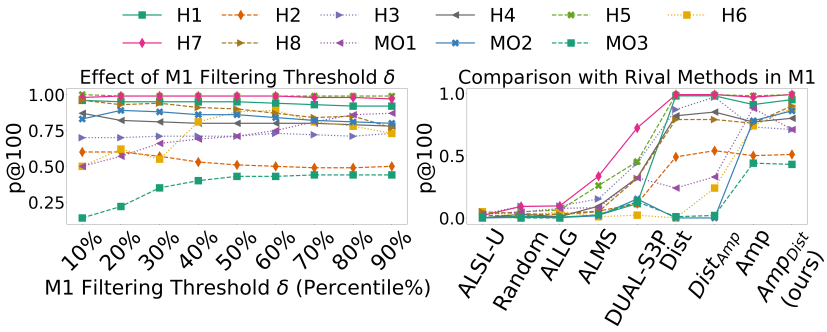

Fig. 7: Precision at 100 (p@100) in Module M1. *left)* The effect of the NN distance threshold δ on our Amp_{Dist} sampling strategy. *right)* Comparison between Amp_{Dist} (with δ being 50-percentile) and its rivals. The methods on the x-axis are placed from left to right in ascending order of their average p@100.

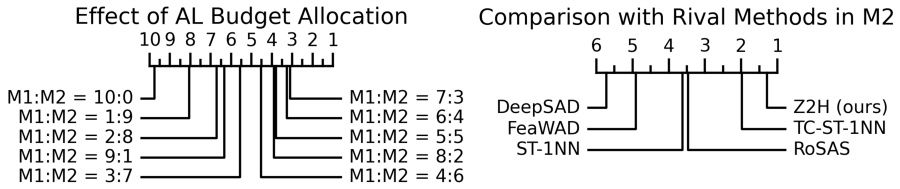

Fig. 8: CD diagrams of transduction F1-scores in M2. Methods to the right are better. *left)* Effect of AL budget allocation between M1 and M2. *right)* Comparison between Z2H ($M1 : M2 = 7 : 3$) with rival methods.

the P2P amplitudes (rather than the NN distances) to decide the primary query order. *Amp* and Amp_{Dist} also outperform the non-knowledge-guided methods, with Amp_{Dist} edging out *Amp* thanks to the incorporation of NN distances.

[Module M2] We now examine the AL budget allocation between M1 and M2, and compare our method with rival methods in terms of translation F1-score, i.e., the F1-score of the automatic inference of U examples in M2. For budget allocation, we consider $M1 : M2 = 1 : 9, 2 : 8, \ldots, 10 : 0$ (note that $0 : 10$ is not feasible). When the budget is not enough to find the decision boundaries for all ST-Tree branches, we use the non-AL-based stopping criterion GBTRM [9] to decide the labels of U examples on those branches, as GBTRM tends to outperform other stopping criteria empirically. For rival methods, we consider **ST-1NN**, **TC-ST-1NN**, **RoSAS** [32], **FeaWAD** [34] and **DeepSAD** [23], which are semi-supervised learners. ST-1NN equates M2 with no AL-based enhancement, using GBTRM as the stopping criterion; TC-ST-1NN is the two-class variant of ST-1NN that uses both the PL and NL sets obtained in M1; the others are deep learners. Since these methods cannot benefit from AL in M2, we set $M1 : M2 = 10 : 0$ for them to fully exploit the total AL budget and ensure fairness. See [1] for detailed settings of rival methods.

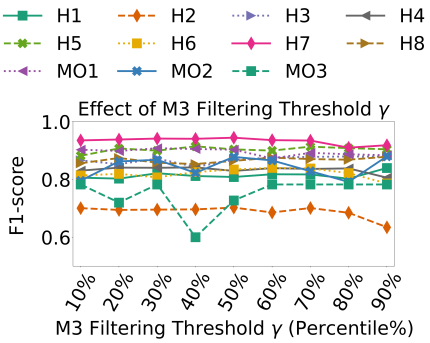

Fig. 9: F1-scores of EEGPT with Z2H under varying settings of the M3 filtering threshold γ, with a total AL budget of 5% of all data.

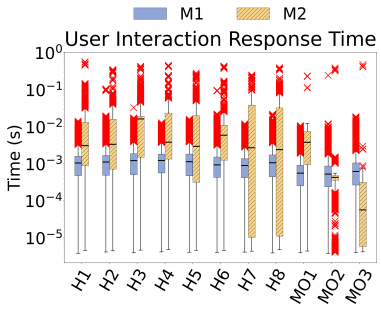

Fig. 10: User interaction response times in M1 and M2. Time axis on a log scale. "×" indicates outliers. These response times are mostly negligible in real-time interactions.

The results are shown in Fig. 8, which are CD diagrams for all combinations of the total AL budget $(5\%, 10\%, \ldots, 30\%)$ and the subject. Fig. 8 *left)* shows $M1 : M2 = 7 : 3$ as the best budget allocation, justifying our use of it in Section 3.1. Fig. 8 *right)* shows that ST-1NN is beaten by TC-ST-1NN, which seems to contradict our choice of this one-class solution over the two-class one. However, this is mainly due to the limited power of ST-1NN's non-AL-based stopping criterion to handle data heterogeneity. With our AL-based decision boundary, Z2H significantly outperforms all rival methods, including TC-ST-1NN.

[Module M3] We now examine the P2P amplitude threshold γ in M3, for which we consider the $10, 20, \ldots, 90$-percentiles. Again limited by GPU hours, we only consider the F1-score of EEGPT with a total AL budget of 5% of all data. As shown in Fig. 9, Z2H is generally insensitive to γ, with the 50-percentile generally yielding good results, which justifies our choice of it in Section 3.1.

[Efficiency] We examine the user interaction response time in M1 and M2 (M3 does not entail user interaction), i.e., the delays between user queries, averaged over 50 runs. As shown in Fig. 10, on average, M1 has smaller delays than M2, as the former requires only simple lookups (which empirically take constant time in most cases) while the latter takes $O(|U|)$ time. The delays in M2 exhibit large variance because the actions to take between two queries vary. In general, the delay will be longer if the current query is the last query made on an ST-Tree branch, as we need to find the next branch to query. In most cases, the delays in both M1 and M2 are negligible, enabling real-time interaction.

4 Related Work

[IED Detection] To date, various machine learning methods have been proposed for IED detection. While early works drew on shallow learners with pre-

defined features [2, 25], recent ones mainly adopt deep models such as CNN [7, 21, 24, 29], LSTM [8], and Transformer [22]. Some works have explored reducing manual labeling via semi-supervision [7] and data augmentation [7, 8], yet they can still require large numbers of pre-existing manual labels, entailing no AL elements. Regarding data imbalance, in [29], we proposed a novel loss function to optimize the F-score, coupled with a negative sampling strategy. However, that method relied on a fully labeled training set, without looking into training data selection. In the broader EEG community, several Transformer-based foundation models [4, 11, 28, 33] have been proposed, which can also be fine-tuned for IED detection. Our Z2H can drastically reduce (by 95% in Table 2) the amount of manual labeling required to train or fine-tune the aforementioned models.

[UAL] With no prior labeled data, existing UAL methods rank the candidate examples by their power to reconstruct the data [10, 14–17, 20, 27] or reduce uncertainty [27] in a latent space. To address data heterogeneity, some methods [10, 17] account for data locality by reconstructing examples using those close to them as much as possible. ALLG [20] conducts UAL on learned graphs derived from nearest neighbor information, which helps better capture local characteristics. ALSL-U [16] transforms the data into a low-rank representation to better preserve the underlying subspaces. Some recent works [15, 27] more explicitly handle data imbalance and heterogeneity using K-means in the latent space to preserve local clusters. However, such methods can be limited in their ability to handle highly random patterns that do not form well-defined clusters (such as non-IEDs), and are constrained by the AL budget regarding the selected data size, while our Z2H can effectively address these issues.

5 Conclusions

In this work, we proposed Z2H, a model-agnostic, knowledge-guided UAL pipeline for training/fine-tuning data selection for a user-defined deep IED detection model. By effectively handling data imbalance and heterogeneity, Z2H can reduce manual labeling by 95% while yielding similar or better results when paired with a pre-trained EEG foundation model [4, 28].

Z2H may offer a promising avenue for improving the neurological examination of EEG data by drastically reducing the time and effort required to annotate the data visually, particularly for long, continuous recordings. Moreover, Z2H has shown promising results on both human and mouse subjects that have varying data acquisition parameters and electrode configurations. This can facilitate scientific inferences from diverse datasets, potentially significantly accelerating clinical decision-making and research. For future work, we plan to further develop Z2H for more diverse cohorts, recording setups, and clinical contexts. We will also further improve the robustness of Z2H against high-amplitude noise and artifacts, as well as its performance in *inter*-subject settings, in conjunction with domain adaptation/generalization techniques.

Acknowledgments. This work is supported by EU Horizon projects AI4Europe (101070000), TwinODIS (101160009), ARMADA (101168951), DataGEMS (101188416)

and RECITALS (101168490), and by $Y\Pi AI\Theta A$ & NextGenerationEU project HARSH ($Y\Pi 3TA-0560901$) that is carried out within the framework of the National Recovery and Resilience Plan "Greece 2.0" with funding from the European Union – NextGenerationEU, and was provided with HPC and storage resources by GENCI at IDRIS thanks to the grant 2025-A0191012641 on the supercomputer Jean Zay's H100 partition. Part of this work was performed at the Paris Brain Institute Electrophysiology Core Facility (RRID:SCR_026412), the Data Analysis Core Facility (RRID:SCR_026138), and the CENIR neuroimaging Core Facility. This work was done while Cihang Yu was a master's intern at LIPADE and while Qitong Wang was with LIPADE.

References

1. Source code, mouse datasets and additional experimental results. `https://github.com/sliang11/Z2H`
2. Abdi-Sargezeh, B., Valentin, A., Alarcon, G., Sanei, S.: Incorporating uncertainty in data labeling into automatic detection of interictal epileptiform discharges from concurrent scalp-EEG via multi-way analysis. Int. J. Neural Syst **31**(08) (2021)
3. Bagnall, A., Lines, J., Bostrom, A., Large, J., Keogh, E.: The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data Min. Knowl. Discov. **31**(3), 606–660 (2017)
4. Cui, W., Jeong, W., Th"olke, P., Medani, T., Jerbi, K., Joshi, A.A., Leahy, R.M.: Neuro-GPT: Towards a foundation model for EEG. In: ISBI (2024)
5. de Curtis, M., Avanzini, G.: Interictal spikes in focal epileptogenesis. Prog. Neurobiol. **63**(5) (2001)
6. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine learning research **7**(Jan), 1–30 (2006)
7. Fürbass, F., Kural, M.A., Gritsch, G., Hartmann, M., Kluge, T., Beniczky, S.: An artificial intelligence-based EEG algorithm for detection of epileptiform EEG discharges: Validation against the diagnostic gold standard. Clin. Neurophysiol. **131**(6) (2020)
8. Geng, D., Alkhachroum, A., Bicchi, M.A.M., Jagid, J.R., Cajigas, I., Chen, Z.S.: Deep learning for robust detection of interictal epileptiform discharges. J. Neural Eng. **18**(5) (2021)
9. González, M., Bergmeir, C., Triguero, I., Rodríguez, Y., Benítez, J.: On the stopping criteria for k-nearest neighbor in positive unlabeled time series classification problems. Inf. Sci. **328** (2016)
10. Hu, Y., Zhang, D., Jin, Z., Cai, D., He, X.: Active learning via neighborhood reconstruction. In: IJCAI (2013)
11. Jiang, W., Wang, Y., Lu, B.l., Li, D.: NeuroLM: A universal multi-task foundation model for bridging the gap between language and EEG signals. In: ICLR 2025
12. Jobst, B.C., Bartolomei, F., Diehl, B., Frauscher, B., Kahane, P., Minotti, L., Sharan, A., Tardy, N., Worrell, G., Gotman, J.: Intracranial EEG in the 21st century. Epilepsy Curr. **20**(4) (2020)
13. Lehongre, K., Lambrecq, V., Whitmarsh, S., Frazzini, V., Cousyn, L., Soleil, D., Fernandez-Vidal, S., Mathon, B., Houot, M., Lemarechal, J.D., Clemenceau, S., Hasboun, D., Adam, C., Navarro, V.: Long-term deep intracerebral microelectrode recordings in patients with drug-resistant epilepsy: Proposed guidelines based on 10-year experience. NeuroImage p. 119116 (2022)
14. Li, C., Li, R., Yuan, Y., Wang, G., Xu, D.: Deep unsupervised active learning via matrix sketching. TIP **30**, 9280–9293 (2021)

15. Li, C., Ma, H., Yuan, Y., Wang, G., Xu, D.: Structure guided deep neural network for unsupervised active learning. TIP **31**, 2767–2781 (2022)
16. Li, C., Mao, K., Liang, L., Ren, D., Zhang, W., Yuan, Y., Wang, G.: Unsupervised active learning via subspace learning. In: AAAI 2021
17. Li, C., Wang, X., Dong, W., Yan, J., Liu, Q., Zha, H.: Joint active learning with feature selection via cur matrix decomposition. TPAMI **41**(6), 1382–1396 (2018)
18. Li, X.L., Liu, B.: Learning from positive and unlabeled examples with different data distributions. In: ECML (2005)
19. Liang, S., Zhang, Y., Ma, J.: Active model selection for positive unlabeled time series classification. In: ICDE (2020)
20. Ma, H., Li, C., Shi, X., Yuan, Y., Wang, G.: Deep unsupervised active learning on graphs. TNNLS **35**(2), 2894–2900 (2022)
21. Quon, R.J., Meisenhelter, S., Camp, E.J., Testorf, M.E., Song, Y., Song, Q., Culler, G.W., Moein, P., Jobst, B.C.: AiED: Artificial intelligence for the detection of intracranial interictal epileptiform discharges. Clin. Neurophysiol. (2022)
22. Rao, W., Wang, H., Zhuang, K., Guo, J., Gu, P., Zhang, L., Wang, X., Jiang, J., Chen, D.: Automatic multi-label classification of interictal epileptiform discharges (ied) detection based on scalp EEG and transformer. In: ICIC 2024
23. Ruff, L., Vandermeulen, R.A., Görnitz, N., Binder, A., Müller, E., Müller, K.R., Kloft, M.: Deep semi-supervised anomaly detection. In: ICLR 2019
24. de Sousa, A.M.A., van Putten, M.J., van den Berg, S., Haeri, M.A.: Detection of interictal epileptiform discharges with semi-supervised deep learning. Biomedical Signal Processing and Control **88**, 105610 (2024)
25. Spyrou, L., Martín-Lopez, D., Valentín, A., Alarcón, G., Sanei, S.: Detection of intracranial signatures of interictal epileptiform discharges from concurrent scalp EEG. Int. J. Neural Syst. **26**(04) (2016)
26. Staley, K.J., Dudek, F.E.: Interictal Spikes and Epileptogenesis. Epilepsy Curr. **6**(6) (2006)
27. Wan, C., Jin, F., Qiao, Z., Zhang, W., Yuan, Y.: Unsupervised active learning with loss prediction. Neural Computing and Applications **35**(5), 3587–3595 (2023)
28. Wang, G., Liu, W., He, Y., Xu, C., Ma, L., Li, H.: EEGPT: Pretrained transformer for universal and reliable representation of EEG signals. NeurIPS **37** (2024)
29. Wang, Q., Whitmarsh, S., Navarro, V., Palpanas, T.: iedeal: A deep learning framework for detecting highly imbalanced interictal epileptiform discharges. VLDB **16**(3) (2022)
30. Wei, L., Keogh, E.: Semi-supervised time series classification. In: KDD (2006)
31. Whitmarsh, S., Nguyen-Michel, V.H., Lehongre, K., Mathon, B., Adam, C., Lambrecq, V., Frazzini, V., Navarro, V.: Sleep increases firing rate modulation during interictal epileptiform discharges in mesial temporal structures. Brain Communications **8**(3) (2026)
32. Xu, H., Wang, Y., Pang, G., Jian, S., Liu, N., Wang, Y.: RoSAS: Deep semi-supervised anomaly detection with contamination-resilient continuous supervision. Inf. Process. Manag. **60**(5) (2023)
33. Zhou, X., Liu, C., Chen, Z., Wang, K., Ding, Y., Jia, Z., Wen, Q.: Brain foundation models: A survey on advancements in neural signal processing and brain discovery. IEEE Signal Processing Magazine **42**(5), 22–35 (2025)
34. Zhou, Y., Song, X., Zhang, Y., Liu, F., Zhu, C., Liu, L.: Feature encoding with autoencoders for weakly supervised anomaly detection. TNNLS **33**(6) (2021)