UnSLU-BENCH+: Extended Machine Unlearning Benchmark for Spoken Language Understanding

UnSLU-BENCH+: Extended Machine Unlearning Benchmark for Spoken Language Understanding

Claudio Savelli, Alkis Koudounas, *Member, IEEE*, Flavio Giobergia, and Elena Baralis, *Member, IEEE*

***Abstract*—With the growing adoption of voice assistants and Spoken Language Understanding (SLU) systems, the need for *machine unlearning*, i.e., the ability to remove the influence of specific training data without full retraining, has become increasingly critical to ensure user privacy and regulatory compliance. In this work, we present UnSLU-BENCH+, an extended benchmark for systematically evaluating unlearning methods in SLU. Our benchmark spans six datasets in four languages, each paired with three self-supervised models, and covers eleven state-of-the-art unlearning techniques. These methods are evaluated based on three foundational principles of unlearning: *utility*, i.e., the performance of the unlearned model, *efficacy*, i.e., the actual removal of the information to be forgotten, and *efficiency*, i.e., the computational cost.To support this evaluation, we introduce CGUM, a classification-aware extension of the Global Unlearning Metric (GUM), which provides a unified, interpretable score that jointly captures these three dimensions. Our extensive experiments demonstrate that unlearning performance varies significantly across models and datasets, with no single method universally dominating. Our results establish a solid foundation for evaluating and developing future machine unlearning techniques in speech applications while promoting CGUM as a robust evaluation metric for every classification-based unlearning scenario.**

***Index Terms*—Machine unlearning, spoken language understanding, intent classification, transformers.**

I. Introduction

THE increasing adoption of voice assistants and spoken language understanding (SLU) systems has intensified concerns regarding user privacy, data retention, and compliance with legal frameworks such as the General Data Protection Regulation in Europe (GDPR) [1], the Personal Information Protection Law of the People's Republic of China [2], or the Bill C-27 in Canada. These regulations establish the "right to be forgotten," requiring machine learning (ML) models to remove the influence of specific user data upon request [3]. To meet this demand, *machine unlearning* (MU) has emerged as a critical research direction, aiming to remove the effect of selected training samples without full model retraining.

MU has obtained a growing interest in various fields. In vision, several benchmarks have been proposed to evaluate unlearning methods on tasks that use facial identities [4], [5]. In the LLM domain, recent efforts have focused on unlearning specific facts or user-related data [6], [7]. While these benchmarks have helped standardize MU evaluation in their respective areas, the speech domain, despite its strong privacy implications, still lacks comparable benchmarks, particularly for complex SLU tasks. In fact, MU is particularly relevant in this task as spoken signals often encode both linguistic information and paralinguistic traits such as identity, emotion, or mental state. While initial explorations have addressed low-complexity tasks like keyword spotting [8], more recent studies have begun to explore unlearning in paralinguistic applications. For example, Phukan et al. [9] proposed MU strategies for emotion recognition and depression detection, relying on structured retraining approaches based on exact unlearning methods. However, these methods require rigid assumptions about data access and model architectures, making them less applicable to large-scale SLU models already deployed. Despite these advances, the challenges of unlearning in realistic and complex SLU scenarios remain largely unexplored.

To address these limitations, we present an extended version of the UnSLU-BENCH [10], designed to evaluate MU techniques in SLU systematically. UnSLU-BENCH+ is a large-scale, multilingual benchmark spanning six intent classification datasets covering four languages, each paired with three transformer-based models. It supports the evaluation of eleven state-of-the-art MU strategies applied under realistic unlearning scenarios, i.e., speaker forgetting. In this context, speaker forgetting refers to the removal of a specific user's data – including all utterances associated with a given speaker identity – from a trained SLU model, thereby erasing their influence without retraining the model from scratch. This form of unlearning aligns closely with the already cited "right to be forgotten" requests, where users seek to have their voice data removed from ML models. The evaluation of the benchmark is based on the three foundational principles of MU: *utility*, which measures how well the model retains its performance; *efficacy*, which quantifies how effectively the influence of the forgotten data has been removed; and *efficiency*, which captures the computational cost of the unlearning process, reflecting its practicality for real-world deployment. By jointly considering these aspects, our benchmark facilitates a comprehensive and balanced assessment of the trade-offs associated with each method. This extension builds on top of the original benchmark by introducing:

- *Additional Dataset:* We include the publicly available *Timers&Such* dataset [11], a corpus of spoken English commands for common voice control use cases involving numbers, with annotated speaker identities.

Received 29 September 2025; revised 6 February 2026; accepted 12 March 2026. Date of publication 19 March 2026; date of current version 6 April 2026. The associate editor coordinating the review of this article and approving it for publication was Dr. Yue Zhang. *(Corresponding author: Claudio Savelli.)*

The authors are with Politecnico di Torino, 10129 Turin, Italy (e-mail: claudio.savelli@polito.it).

Digital Object Identifier 10.1109/TASLPRO.2026.3675768

- *Additional Models:* We expand upon previous work employing HuBERT [12], wav2vec 2.0 [13], XLS-R 53 [14], and XLS-R 128 [15] by incorporating *WavLM Base+* [16] for English SLU datasets (FSC [17], SLURP [18], and Timers&Such [11]). We further introduce *mHuBERT-147* [19], a multilingual model pre-trained on 147 languages, to support MU evaluation on non-English datasets, specifically Italian (ITALIC [20]), German, and French (SpeechMASSIVE [21])
- *Additional Unlearning Techniques:* We add to the benchmark three unlearning approaches, reaching a total of 11 MU methods. *Selective Random Labelling (SRL)* [22] injects noise by randomly relabelling the forget set. *Exact Unlearning of Last-k Layers (EU-k)* [23] re-trains the k final layers of the model from scratch on the retain set while keeping earlier layers frozen. *Saliency-based Unlearning (SalUn)* [24] applies a saliency analysis to identify which weights are most influenced by the forget set and modifies only those, enabling targeted parameter updates.
- *Refined Evaluation Metric:* We propose the Classification Global Unlearning Metric (CGUM), an improved version of GUM [10], explicitly adapted for classification tasks. Unlike the original formulation, this version introduces a lower bound for the utility of a model: if the unlearned model achieves task performance comparable to that of a naive classifier, CGUM drops to 0.

The paper is organized as follows. Section II reviews prior work on machine unlearning in speech and highlights the limitations of existing benchmarks. Section III describes the set of unlearning methods and evaluation metrics adopted in our study, introducing the Classification-GUM metric and formalizing its theoretical foundations. Section IV details the experimental setup, covering datasets, models, and hyperparameter choices, while Section IV presents the empirical findings and analyzes the trade-offs among methods. Finally, Section V summarizes conclusions and outlines future research directions.

II. Related Works

A. Machine Unlearning and Evaluation Dimensions

Machine unlearning refers to the task of removing the influence of specific training samples from a learned model without requiring full retraining from scratch. Given a model θ trained on a dataset $\mathcal{D}$, each data point is represented as a triplet (x, y, s), where x denotes the input (e.g., an utterance in SLU tasks), y is the corresponding label (e.g., the intent associated with the utterance), and s is the identity of the data source (e.g., a speaker). Let $\mathcal{S}$ be the set of all identities in $\mathcal{D}$, and let $\mathcal{S}_f \subset \mathcal{S}$ denote the subset of identities that request data removal. This results in a forget set $\mathcal{D}_f = \{(x, y, s) \in \mathcal{D} \mid s \in \mathcal{S}_f\}$ and a retain set $\mathcal{D}_r = \mathcal{D} \setminus \mathcal{D}_f$. The goal of a MU process $\phi(\cdot)$ is to produce a new model $\hat{\theta} = \phi(\theta, \mathcal{D}_r, \mathcal{D}_f)$ in which the influence of $\mathcal{D}_f$ has been erased. Ideally, $\hat{\theta}$ should be close to the Gold model θ', which is trained from scratch using only $\mathcal{D}_r$, i.e., $\theta' = \mathcal{A}(\mathcal{D}_r)$ where $\mathcal{A}$ denotes the original training procedure. However, retraining θ' for every unlearning request is computationally expensive and often impractical, motivating the need for approximate unlearning procedures [22].

To assess the performance of MU methods, the literature emphasizes the need to evaluate three key dimensions: utility, efficacy, and efficiency [25]. **Utility** captures how well the model retains its ability to perform the original task on test data. For classification problems, it is commonly quantified using accuracy or macro F1 score on a test set: high utility ensures that unlearning does not cause catastrophic forgetting [26]. **Efficacy** refers to the degree to which the unlearned model $\hat{\theta}$ forgets the influence of $\mathcal{D}_f$, typically measured through Membership Inference Attacks (MIA) [25]. Lower MIA scores indicate that the unlearned model has reduced its memorization of the forget set and better approximates the behavior of the Gold model in terms of privacy leakage. MIA-based evaluation measures how distinguishable the forgotten data is from unseen test samples. A perfectly unlearned model should produce similar MIA distributions for the forgotten and unseen sets, reflecting the successful removal of memorized information. Deviations from this ideal, either by under-unlearning (where forget samples were not fully erased) or over-unlearning (where forget samples diverge even from the unseen data), lead to increased privacy leakage [27]. Therefore, the distance between the unlearned and Gold MIA scores is a critical indicator of the quality of unlearning. **Efficiency** measures the computational cost of the unlearning procedure. Since retraining remains the fallback option, MU methods are only practical if they achieve significant gains in wall-clock time, the number of updates, or energy consumption [28].

Evaluating utility, efficacy, and efficiency in isolation often fails to capture the actual trade-offs involved in unlearning. For this reason, composite metrics have been proposed to assess these dimensions jointly. One example is NoMUS [4], which combines efficacy and utility into a single score. However, NoMUS does not account for computational cost, making it insufficient to assess the practical utility of unlearning methods. To address this limitation, GUM [10] was introduced as a three-factor metric that simultaneously incorporates utility (e.g., F1 or accuracy), efficacy (e.g., via MIA), and efficiency (e.g., runtime), offering a more comprehensive evaluation of unlearning quality in real-world settings. In this work, we further extend this metric and propose CGUM, a refined, fully bounded variant designed explicitly for classification tasks. CGUM preserves GUM's core integration of utility, efficacy, and efficiency, while addressing class imbalance in multiclass settings. Its task-independent formulation allows for it to be applied directly to any classification scenario and domain.

B. Machine Unlearning in Speech

While MU has been widely adopted in computer vision and natural language processing [6], [22], [29], its application in the speech domain remains limited in scope and generalizability. Recent studies have explored the feasibility of MU in speech tasks; however, they often suffer from limitations in methodology, scalability, or domain relevance.

Phukan et al. [9] investigated unlearning in paralinguistic speech processing, such as emotion recognition and depression

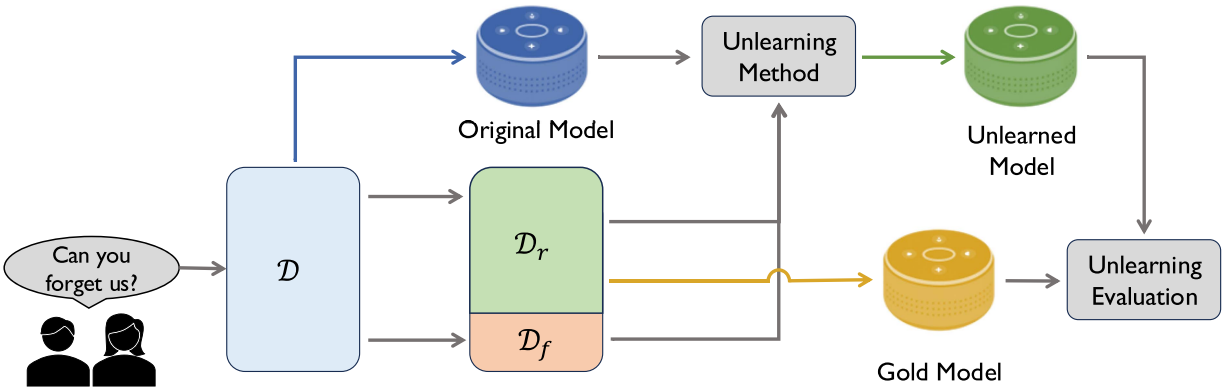

Fig. 1. Machine Unlearning Pipeline in SLU setting. Given a dataset $\mathcal{D}$ composed of utterances labeled with intent and speaker identity, a subset of users requests data deletion. This results in a split into the retain set $\mathcal{D}_r$ and the forget set $\mathcal{D}_f$. The *Original* model is initially trained on the full dataset. An unlearning method then modifies this model to produce an *Unlearned* model, which ideally removes the influence of $\mathcal{D}_f$. A separate `Gold` model is trained from scratch only on $\mathcal{D}_r$ to serve as an ideal reference. The unlearned model is then evaluated against the `Gold` model to assess the quality of unlearning.

detection. Their work focuses exclusively on *exact unlearning* methods, specifically variants of the SISA framework [30], where retraining is decomposed into multiple data shards to support efficient removal of individual points. While promising in terms of performance retention and memory efficiency, these techniques are fundamentally constrained by their architectural assumptions: they require initial control over the data pipeline, typically inapplicable to pre-trained or deployed models, and cannot generalize well to large-scale or dynamic environments. Choi et al. [31] study MU in the context of generative speech and music models. Their proposed framework aims to remove the influence of specific training songs in diffusion-based generators. While this is an essential development in safeguarding copyright, the problem-setting and evaluation objectives differ significantly from classification-based MU in speech understanding systems.

More recently, the work of Cheng et al. [8] evaluates unlearning on keyword spotting and speaker identification tasks using models such as wav2vec 2.0 and datasets like Speech Commands [32] and VoxCeleb1 [33]. However, the benchmark is limited in both linguistic and architectural diversity. It focuses exclusively on English and does not explore SLU complexity. Additionally, the study compares only five unlearning techniques, thus limiting the benchmark's coverage of existing methods. To address these gaps, the UnSLU-BENCH benchmark [10] was introduced as the first systematic evaluation framework for MU in SLU. It targets speaker-level unlearning across five SLU datasets in four languages, using two transformer models per dataset and a core set of eight MU techniques, and introduces the GUM metric. Building on this foundation, `UnSLU-BENCH+` expands both the coverage (with new datasets, models, and MU strategies) and the evaluation protocol, with the introduction of `CGUM`.

III. Benchmark Definition

`UnSLU-BENCH+` is a comprehensive benchmark designed to evaluate MU techniques in SLU. It extends the original UnSLU-BENCH [10] by incorporating a wider range of datasets, models, and unlearning methods, as well as a refined evaluation protocol. Specifically, the benchmark covers 6 datasets[1] in 4 languages, 3 transformer-based models per dataset, and 11 state-of-the-art unlearning techniques. Evaluation is grounded in the three core dimensions of MU (utility, efficacy, and efficiency) and is summarized using the proposed `CGUM` metric.

A. Unlearning Methods

`UnSLU-BENCH+` includes 11 MU techniques as follows.

Fine Tuning (`FT`) continues training the model on the retain set $\mathcal{D}_r$ for one epoch. This baseline assumes that the absence of the forget set $\mathcal{D}_f$ alone will reduce its influence without explicit forgetting operations.

Successive Random Labels (`SRL`) [22] The model is fine-tuned with both $\mathcal{D}_r$ and $\mathcal{D}_f$. The correct label is used as supervision for each sample from $\mathcal{D}_r$. However, for any sample from $\mathcal{D}_f$, the true label is replaced by a randomly selected label different from the correct one.

Negative Gradients (`NG`) [22] applies gradient ascent, i.e., gradient updates reversed in sign, only on the forget set $\mathcal{D}_f$, actively pushing the model away from memorizing the removed examples.

Advanced Neggrad (`NG+`) [4], [24] combines `NG` on $\mathcal{D}_f$ with `FT` on $\mathcal{D}_r$, aiming to balance explicit forgetting with retention of overall performance.

Catastrophically Forgetting Last k Layers (`CF` k) [23] restricts fine-tuning to the final k layers, reducing computational cost and targeting representations most closely associated with memorized information.

Exact Unlearning Last k Layers (`EU` k) [23]: This method re-initializes the final k layers (instead of fine-tuning), then retrains them on $\mathcal{D}_r$, effectively erasing potentially memorized weights more aggressively than `CF` k.

Saliency-based Unlearning (`SalUn`) [24] targets only a small, high-impact subset of model parameters identified via saliency analysis. The key idea is to focus gradient updates

[1]We utilize the FSC, SLURP, Timers&Such, ITALIC, and SpeechMassive datasets, covering both the French (fr-FR) and German (de-DE) linguistic versions of the latter. For our experiments, each language variant is treated as a separate and distinct dataset.

on those parameters that contribute most significantly to the model's behavior on $\mathcal{D}_f$. First, it performs a forward and backward pass on $\mathcal{D}_f$ to compute the saliency score for each model parameter. Parameters are then ranked by saliency, and a fixed fraction of the most influential ones are selected, and SRL is applied only to these weights.

Unlearning by Selective Impair and Repair (UNSIR) [28] is a two-stage unlearning method. The first phase, called *impair*, involves computing error-maximizing adversarial noise for each sample in $\mathcal{D}_f$. This noise is added to the input, and the model is fine-tuned on these perturbed samples to reduce its confidence in forgotten data explicitly. In the second phase, called *repair*, the model is fine-tuned on the retain set $\mathcal{D}_r$ for one additional epoch to recover any loss in task utility. The method is adapted following the sample-level variant introduced by [4], as the original method targeted full-class forgetting.

Bad Teaching (BT) [34] adopts a teacher-student framework involving two reference models: a competent teacher, which is a copy of the original fully trained model, and an incompetent teacher, which is usually a randomly initialized architecture. The student model is initialized from the competent teacher and fine-tuned using a distillation objective that combines guidance from both teachers. During training, the student is encouraged to match the competent teacher's logits on $\mathcal{D}_r$, and the incompetent teacher's on $\mathcal{D}_f$. Following [10], we also propose a **lightweight variant (BT-L)** that replaces the incompetent teacher with a random predictor generator to reduce computational overhead.

SCalable Remembering and Unlearning unBound (SCRUB) [24] is another teacher-student unlearning approach. It operates using a single teacher, that is, the original trained model, and optimizes the student model via a combined loss that integrates three distinct objectives. First, the student is encouraged to align with the teacher on $\mathcal{D}_r$ and to diverge from the teacher's behavior on $\mathcal{D}_f$. Then, a supervised task loss is applied to $\mathcal{D}_r$ to reinforce end-task performance and stabilize training.

B. Metrics

We adopt one representative metric per dimension to evaluate the three core aspects of machine unlearning. **Utility** is measured as the model's ability to retain task performance after unlearning. In classification tasks, where class imbalance is common, we report the *macro* F1 score on the test and forget sets. **Efficacy** is quantified using Membership Inference Attack (MIA) scores, which estimate the model's ability to distinguish between samples in the forget set and unseen test data. An effective unlearning procedure should reduce this distinguishability, driving the MIA score of the unlearned model toward that of the retrained Gold model. **Efficiency** is evaluated in terms of computational cost, reported as the *Speedup* with respect to the Gold model, defined as the ratio between the retraining time and the time required to apply unlearning.

In addition to reporting these metrics independently, we introduce the **Classification Global Unlearning Metric** (CGUM), a classification-aware variant of GUM [10], which redefines each component to ensure boundedness, robustness to class imbalance, and comparability across datasets. A detailed description of CGUM and its theoretical properties is presented in Section III-C and in the Appendix A.

C. Classification GUM

To evaluate unlearning performance in classification tasks, we adopt a refined version of the GUM metric introduced in [10], which we term CGUM (Classification GUM). The original GUM metric is designed to jointly assess three critical aspects of unlearning: utility (U), measured as the similarity in performance between the unlearned model and the Gold model on the test set; efficacy (E), estimated via a Membership Inference Attack (MIA) score that captures how well the forget set has been removed; and efficiency (T), quantified as the relative computational cost of unlearning compared to full retraining.

In the original formulation of GUM, the metric reaches 0 when the unlearning process is either not effective (i.e., the unlearned model "remembers" data points in the same way as the original model), or it is not efficient (i.e., the unlearning process takes as much time as the retraining from scratch). However, there is no guarantee in terms of model utility: an unlearning process that entirely breaks the model (i.e., the unlearned model has performance comparable to a naive classifier) may have GUM $\neq 0$.

To address this asymmetry, CGUM introduces a bounded version of U. This version maintains the original GUM formulation as a weighted harmonic mean over these three dimensions. However, we redefine each component to ensure strict boundedness and compatibility across classification settings. The CGUM score is defined as:

$$\text{CGUM} = \frac{(1+\alpha+\beta)\,U\,E\,T}{\alpha\,E\,T + \beta\,U\,T + U\,E + \varepsilon} \tag{1}$$

where α and β control the relative importance of the three terms and ε is a small constant for numerical stability.

The choice of the weighting coefficients α and β in CGUM is inherently task- and objective-dependent, and should reflect the primary motivation behind the unlearning process. Different application scenarios impose different priorities across the three axes of utility, efficacy, and efficiency. For instance, in privacy-preserving or GDPR-driven settings, efficacy and efficiency are typically prioritized to ensure that sensitive information is removed promptly and reliably, even at the cost of some utility degradation. Conversely, in model maintenance or bias-correction scenarios, preserving downstream performance becomes critical, and utility should be given greater weight. In this work, we select $\alpha = \beta = 1$ to emphasize balanced unlearning, where none of the three dimensions dominates the final score, yielding an equal-weight formulation that penalizes failures in any single dimension.

Utility (U): In CGUM, utility measures how well the unlearned model performs on the original task, normalized relative to a principled lower bound. In this work, we adopt the macro F1 score as the performance metric of interest. Let $F1^{(u)}$ and $F1^{(g)}$ denote the macro F1 score of the unlearned and Gold models, respectively. We define:

$$U = \min\left(\max\left(\frac{F1^{(u)} - F1^{(r)}}{F1^{(g)} - F1^{(r)} + \varepsilon},\ 0\right),\ 1\right) \tag{2}$$

where $F1^{(r)} = 1/C$ denotes the expected macro F1 score of any random classifier in a C-class setting. As we demonstrate in Appendix A, this value constitutes a theoretical upper bound for the macro F1 score achievable by any naive predictor –

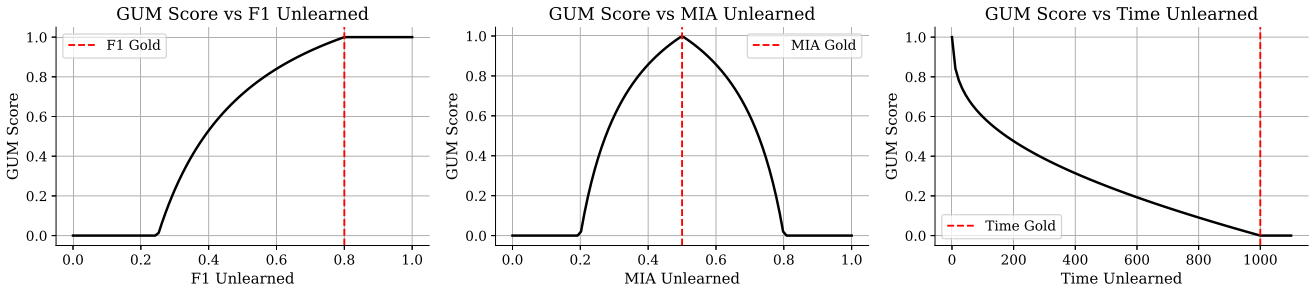

Fig. 2. Behavior of GUM to F1 (left), MIA (center), and unlearning time (right) on a sample dataset with $C = 4$ classes. Each red dashed vertical line indicates the performance of the Gold model. $\text{MIA}^{(o)}$ is set to be 0.8.

either uniform, prior-based, or majority-class. The utility score U is explicitly clipped to the interval $[0, 1]$: a value of $U = 0$ indicates that the unlearned model performs no better than a naive classifier, whereas $U = 1$ corresponds to performance equal to or exceeding that of the Gold model.

Example: In Fig. 2 (left), we assume that the Gold model obtains a macro F1 score of 0.8. The utility score U will peak at 1 if the unlearned model reaches a macro F1 score of 0.8.

Efficacy (E): Efficacy measures how effectively the unlearned model has reduced the influence of the forget set. As often done in literature, we rely on the MIA score, and define:

$$E = 1 - \frac{|\text{MIA}^{(u)} - \text{MIA}^{(g)}|}{|\text{MIA}^{(o)} - \text{MIA}^{(g)} + \varepsilon|} \tag{3}$$

where $\text{MIA}^{(u)}$, $\text{MIA}^{(g)}$, and $\text{MIA}^{(o)}$ are the MIA scores of the unlearned, Gold, and original models, respectively. To prevent instabilities in edge cases, we apply bounded saturation, as done in [10]:

$$\text{MIA}^{(u)} = \begin{cases} \min\left(\text{MIA}^{(u)}, \text{MIA}^{(o)}\right), & \text{if } \text{MIA}^{(o)} \geq \text{MIA}^{(g)} \\ \max\left(\text{MIA}^{(u)}, \text{MIA}^{(o)}\right), & \text{otherwise} \end{cases} \tag{4}$$

Based on this definition of E, it follows that the efficacy score depends on the distance between the unlearned model's MIA and that of the Gold model.

Example: In the case illustrated in Fig. 2 (center), the MIA of the original model is $MIA^{(o)} = 0.8$ and the MIA of the Gold model is $MIA^{(g)} = 0.5$. This bounds the meaningful deviation range to ± 0.3. A score of $E = 1$ is achieved when the unlearned model matches the Gold model in MIA. In contrast, E reaches 0 when the unlearned model has a MIA comparable to the original model, or worse (i.e., the model has not forgotten anything).

Efficiency (T): Efficiency captures the computational benefit of unlearning over full retraining. As in the original GUM [10], we define:

$$T = \min\left(\max\left(1 - \frac{\log(T^{(u)} + 1)}{\log(T^{(g)} + 1)}, 0\right), 1\right) \tag{5}$$

where $T^{(u)}$ and $T^{(g)}$ are the execution times (in seconds) for unlearning and Gold retraining, respectively.

Example: If the training of the Gold model requires 1000 seconds, the evolution of T as a function of the unlearning time is shown in Fig. 2 (right), T is also constrained to the interval $[0, 1]$, where higher values indicate greater efficiency relative to retraining. When the time required to apply the unlearning method approaches zero, the efficiency score tends toward 1, indicating maximum gain over retraining. In contrast, as the time required by the unlearning procedure approaches that of full retraining, T approaches 0. When an unlearning method becomes nearly as expensive as retraining from scratch, the rationale for using a dedicated unlearning method weakens, rendering the unlearning approach useless.

TABLE I
NUMBER OF SPEAKERS AND SAMPLES PER SPLIT FOR EACH DATASET

Dataset	Retain		Forget		Val		Test	
	Spk	Samp	Spk	Samp	Spk	Samp	Spk	Samp
FSC	75	22304	2	828	10	3118	10	3793
TAS	56	31991	34	809	11	271	10	240
SLURP*	76	55464	3	2590	20	8014	75	6330
ITALIC	55	12526	1	597	7	1957	7	1441
SM-DE	116	11174	1	340	68	2033	82	2974
SM-FR	102	11188	1	326	55	2033	75	2974

D. Datasets

The work includes six publicly available SLU datasets. Timers & Such [11], FSC [17], and SLURP [18] cover English, while ITALIC [20] extends to Italian, and SpeechMASSIVE [21] de-DE and fr-FR to German and French, respectively. For FSC, SLURP, ITALIC, and SpeechMASSIVE (German and French), we employ the same splits as those used in [10]. SLURP does not originally provide speaker-independent splits, which are critical for machine unlearning to ensure that identities in the forget, retain, and test sets do not overlap. To address this, we used the identity-disjoint splits for SLURP (SLURP*) as introduced in [10].

Additionally, we introduce in this work Timers&Such (TAS), an English corpus containing commands for voice control use cases involving numbers (e.g., for setting the timer or the alarm). The original training set of TAS (*'train-real'*) includes only 1,640 samples, which is usually insufficient for fine-tuning large speech models. Therefore, we augmented the dataset by adding 7,790 synthetic samples per intent (31,160 in total), generated and released by the original authors [11]. To simulate deletion requests, we keep the original train/test splits and construct the forget set by selecting individuals with at least 20 samples from each dataset, ensuring sufficient representation during training. Approximately 4% of the total training data were selected to construct the forget set. The statistics of the resulting splits are summarized in Table I. The considered datasets exhibit substantial variation in task complexity, with the number of target

Fig. 3. Heatmap summary of CGUM across datasets, models, and unlearning methods. Each cell reports the best CGUM score obtained by a given unlearning method for a specific model-dataset pair. Darker colors indicate higher overall unlearning quality.

TABLE II
UNLEARNING ON FSC. $F1_T$ DENOTES MACRO F1 ON TEST SET, WHILE $F1_F$ ON FORGET SET. BEST RESULTS (I.E., CLOSEST TO GOLD MODEL FOR F1 AND MIA, HIGHEST FOR OTHERS) ARE IN **BOLD**, SECOND-BEST UNDERLINED. ORIGINAL AND GOLD MODELS ARE HIGHLIGHTED.

Method	HuBERT					wav2vec 2.0					WavLM base plus				
	$F1_T$	$F1_F$	MIA	CGUM	Speedup	$F1_T$	$F1_F$	MIA	CGUM	Speedup	$F1_T$	$F1_F$	MIA	CGUM	Speedup
Orig.	.993	1.000	.511	.000	1.000×	.994	1.000	.508	.000	1.000×	.996	1.000	.495	.000	1.000×
Gold	.991	.996	.507	.000	1.000×	.993	.997	.503	.000	1.000×	**.995**	**1.000**	**.500**	.000	1.000×
FT	.979	.993	**.508**	.492	7.690×	**.993**	**.999**	**.504**	.499	7.960×	.955	.954	.499	.491	6.480×
SRL	**.992**	1.000	.505	.439	7.380×	.994	.960	.665	.001	7.580×	.997	**1.000**	**.500**	.487	6.220×
NG	**.992**	**.996**	.514	.001	**201.1**×	.987	.976	.501	**.722**	**206.9**×	.996	**1.000**	.510	**.856**	**169.1**×
NG+	.979	.929	.510	.282	3.900×	.933	.558	.734	.001	4.030×	.996	**1.000**	.495	.358	3.300×
CF-k	.993	1.000	.505	**.562**	26.70×	.994	1.000	.501	.553	16.97×	.996	**1.000**	**.500**	.610	13.62×
EU-k	.993	1.000	.489	.000	26.22×	.995	**.999**	.513	.001	21.33×	.996	**1.000**	.507	.612	13.74×
SalUn	**.992**	.999	.505	.427	6.709×	.994	.982	.551	.001	6.890×	.997	**1.000**	**.500**	.473	5.761×
UNSIR	.993	.998	**.508**	.464	6.380×	.991	1.000	.506	.392	6.550×	.996	**1.000**	.496	.458	5.310×
BT	.993	.999	.504	.301	4.650×	.994	1.000	.508	.001	4.780×	**.995**	**1.000**	.498	.397	3.930×
BT-L	.993	.997	**.506**	.445	5.690×	**.993**	.992	.513	.001	5.870×	.996	**1.000**	.537	.439	4.820×
SCRUB	.993	.998	**.508**	.460	6.220×	.994	1.000	.506	.386	6.210×	.996	**1.000**	.495	.449	5.070×

intent classes ranging from $C = 4$ in TAS to $C = 88$ in SLURP*, while FSC comprises 31 classes and ITALIC, together with SpeechMASSIVE de-De and fr-FR), includes 60 classes.

E. Models

We fine-tune three transformer-based speech encoders per dataset. For English datasets (FSC, TAS, SLURP*), we employ wav2vec 2.0 [13], HuBERT [12], and WavLM Base+ [16], which provide superior pretraining through masked prediction and audio augmentation. For multilingual datasets (ITALIC, SpeechMASSIVE de-De and fr-FR), we use XLS-R 128 [15], XLS-R 53 [14] fine-tuned for the specific language, and mHuBERT-147 [19].

IV. EXPERIMENTAL RESULTS

In this section, we present the benchmark results, along with additional implementation details necessary for reproducibility.

Implementation Details: We evaluate 11 unlearning methods, including: FT, SRL, NG, NG+, CF-k, EU-k, SalUn, UNSIR, BT, BT-L, and SCRUB, as described in Section III. For SalUn, we compute saliency scores over all model parameters and apply a binary mask to update only the top 50% most salient weights.

For each method, we evaluated different learning rate ranges to account for destructive tendencies. Specifically, we use learning rates {5e-07, 1e-06, 5e-06} for SRL, NG, NG+, SalUn, BT, BT-L, SCRUB, and {1e-05, 5e-05, 1e-04} for FT, CF-k, EU-k, UNSIR. The best-performing configuration is selected using the CGUM metric, maximizing the trade-off between utility, efficacy, and efficiency. We empirically observe that these learning rate ranges provide a good balance between efficacy and utility, while maintaining high efficiency. This configuration strategy has also been adopted in prior work [10]. For all the experiments, models are fine-tuned using the original training protocol for each dataset and then post-processed using the corresponding unlearning procedure.

Fine-tuning procedures and hyperparameters can be found in the project repository.[2] All experiments are conducted on a single NVIDIA A6000 GPU.

Results: The analysis of Tables II–VII reveals consistent trends in MU performance across datasets, with notable variations depending on model complexity and language. Results are interpreted relative to the Gold model for F1 and MIA, while speedup quantifies the computational advantage. To facilitate a global comparison across datasets, models, and unlearning strategies, Fig. 3 summarizes the CGUM scores providing a compact view of relative robustness and stability between datasets, unlearning methods, and models that complements the detailed numerical results reported.

FT remains a robust baseline, particularly for medium-sized models like wav2vec 2.0, HuBERT, and XLS-R 128, where it delivers stable utility with modest MIA values. However, its effectiveness in reducing privacy leakage is limited in specific

[2]The complete code repository will be made publicly available upon acceptance.

TABLE III
UNLEARNING ON TAS. $F1_T$ DENOTES MACRO F1 ON TEST SET, WHILE $F1_F$ ON FORGET SET. BEST RESULTS (I.E., CLOSEST TO GOLD MODEL FOR F1 AND MIA, HIGHEST FOR OTHERS) ARE IN **BOLD**, SECOND-BEST UNDERLINED. ORIGINAL AND GOLD MODELS ARE HIGHLIGHTED.

Method	HuBERT					wav2vec 2.0					WavLM base plus				
	$\mathbf{F1_T}$	$\mathbf{F1_F}$	MIA	CGUM	Speedup	$\mathbf{F1_T}$	$\mathbf{F1_F}$	MIA	CGUM	Speedup	$\mathbf{F1_T}$	$\mathbf{F1_F}$	MIA	CGUM	Speedup
Orig.	.981	.985	.485	.000	1.000×	.972	.985	.481	.000	1.000×	.976	.984	.479	.000	1.000×
Gold	.970	.979	.508	.000	1.000×	.980	.979	.500	.000	1.000×	.938	.969	.496	.000	1.000×
FT	.986	.983	.500	.409	3.980×	.927	.971	.502	.423	4.430×	.923	.950	.504	.380	3.480×
SRL	**.968**	.949	**.508**	.403	3.870×	.950	.956	.488	.420	4.330×	.968	.977	.485	.374	3.370×
NG	.981	.975	.506	**.859**	**151.4**×	.972	**.977**	.490	**.859**	**168.7**×	.976	.980	.477	.003	**131.5**×
NG+	.250	.125	.504	.000	2.000×	.633	.557	.548	.244	2.220×	.679	.677	.500	.188	1.730×
CF-k	.981	.985	.483	.003	13.64×	**.981**	.985	.485	.632	15.20×	.976	.984	.479	**.528**	7.330×
EU-k	.981	.985	.483	.003	13.73×	.977	.985	.494	.631	15.23×	.976	.984	.479	**.528**	7.320×
SalUn	.986	.981	.485	.386	3.576×	.971	.976	.506	.406	4.005×	.954	.983	.488	.356	3.121×
UNSIR	.976	**.979**	.492	.384	3.550×	.969	.975	.490	.403	3.950×	.981	.989	**.498**	.352	3.070×
BT	.981	.984	.492	.291	2.420×	.977	.986	**.500**	.315	2.700×	.968	.983	.481	.255	2.120×
BT-L	.981	.976	.583	.343	2.980×	.919	.938	.481	.361	3.330×	.987	.977	.669	.311	2.600×
SCRUB	.981	.984	.492	.366	3.280×	.971	**.977**	.490	.373	3.450×	**.948**	**.971**	.483	.324	2.750×

TABLE IV
UNLEARNING ON SLURP∗. $F1_T$ DENOTES MACRO F1 ON TEST SET, WHILE $F1_F$ ON FORGET SET. BEST RESULTS (I.E., CLOSEST TO GOLD MODEL FOR F1 AND MIA, HIGHEST FOR OTHERS) ARE IN **BOLD**, SECOND-BEST UNDERLINED. ORIGINAL AND GOLD MODELS ARE HIGHLIGHTED.

Method	HuBERT					wav2vec 2.0					WavLM base plus				
	$\mathbf{F1_T}$	$\mathbf{F1_F}$	MIA	CGUM	Speedup	$\mathbf{F1_T}$	$\mathbf{F1_F}$	MIA	CGUM	Speedup	$\mathbf{F1_T}$	$\mathbf{F1_F}$	MIA	CGUM	Speedup
Orig.	.712	1.000	.613	.000	1.000×	.689	1.000	.628	.000	1.000×	.758	.998	.585	.000	1.000×
Gold	.704	.715	.492	.000	1.000×	.707	.711	.506	.000	1.000×	.734	.790	.514	.000	1.000×
FT	.734	1.000	.611	.047	78.99×	.685	.984	.629	.000	83.76×	.764	1.000	.598	.000	69.02×
SRL	.713	.999	.638	.000	75.52×	.658	.945	**.595**	**.410**	79.87×	.760	.999	.606	.000	65.95×
NG	.718	.959	.587	.419	**1654**×	.695	.986	.604	.395	**1749**×	.727	.964	.581	.148	**1446**×
NG+	.630	**.852**	**.453**	**.534**	39.30×	.701	.995	.603	.338	41.63×	.756	**.936**	**.559**	**.438**	34.74×
CF-k	.715	1.000	.608	.110	274.2×	**.709**	1.000	.626	.047	292.0×	.755	.999	.588	.000	147.4×
EU-k	.703	.978	.633	.000	275.2×	.693	.980	.641	.000	291.6×	.741	.983	.611	.000	147.2×
SalUn	.708	.999	.646	.000	7.282×	.657	.944	.599	.381	74.125×	.763	1.000	.614	.000	62.53×
UNSIR	.722	1.000	.613	.000	6.440×	.693	1.000	.629	.000	64.08×	**.739**	.999	.613	.000	52.54×
BT	.711	1.000	.613	.000	47.42×	.710	.999	.619	.172	5.340×	.758	.997	.583	.076	4.730×
BT-L	.685	.907	.558	.504	58.11×	.691	**.631**	.719	.000	61.74×	.753	.997	.610	.000	51.03×
SCRUB	**.704**	1.000	.600	.231	65.40×	.697	.999	.608	.307	64.81×	.766	.996	.579	.192	55.18×

TABLE V
UNLEARNING ON ITALIC. $F1_T$ DENOTES MACRO F1 ON TEST SET, WHILE $F1_F$ ON FORGET SET. BEST RESULTS (I.E., CLOSEST TO GOLD MODEL FOR F1 AND MIA, HIGHEST FOR OTHERS) ARE IN **BOLD**, SECOND-BEST UNDERLINED. ORIGINAL AND GOLD MODELS ARE HIGHLIGHTED.

Method	XLS-R 53-IT					XLS-R 128					mHuBERT-147				
	$\mathbf{F1_F}$	$\mathbf{F1_F}$	MIA	CGUM	Speedup	$\mathbf{F1_T}$	$\mathbf{F1_F}$	MIA	CGUM	Speedup	$\mathbf{F1_T}$	$\mathbf{F1_F}$	MIA	CGUM	Speedup
Orig.	.778	1.000	.615	.000	1.000×	.688	.894	.632	.000	1.000×	.766	.892	.587	.000	1.000×
Gold	.784	.736	.478	.000	1.000×	.643	.568	.532	.000	1.000×	.751	.656	.547	.000	1.000×
FT	.711	.850	.550	.482	31.10×	**.638**	.671	.555	**.564**	3.800×	.763	.857	.552	.745	226.3×
SRL	.733	.954	.591	.305	29.08×	.691	.882	.626	.145	28.64×	.778	.894	.604	.000	212.8×
NG	.590	.621	**.525**	**.671**	**623.0**×	.679	.868	.603	.495	**613.4**×	**.761**	.858	.565	.741	**4740**×
NG+	.743	.936	.582	.334	15.37×	.658	.001	.932	.000	15.14×	.765	**.793**	.554	.684	11.47×
CF-k	.781	1.000	.609	.115	98.99×	.677	.871	.626	.151	98.59×	.777	.915	.592	.000	663.9×
EU-k	.751	.910	.582	.403	99.76×	.618	.823	.590	.529	98.39×	.677	.821	.539	**.762**	664.7×
SalUn	.740	.988	.601	.215	27.03×	.696	.888	.629	.080	26.70×	.776	.895	.602	.000	147.5×
UNSIR	**.782**	1.000	.612	.060	22.26×	.636	.830	.621	.223	22.01×	.772	.858	.569	.568	143.5×
BT	.731	**.848**	.557	.428	17.94×	.683	**.639**	.481	.453	17.90×	.768	.848	.550	.716	126.7×
BT-L	.729	.876	.564	.423	22.21×	.686	.651	.518	.546	22.02×	.768	.861	**.547**	.747	159.0×
SCRUB	.770	.990	.610	.095	22.66×	.442	.357	**.533**	.521	23.25×	.772	.891	.585	.131	167.8×

settings. Notably, when applied to SLURP∗, the MIA scores obtained by FT and other unlearning methods are often higher than those of the original model, suggesting a tendency toward over-forgetting or instability. This pattern is consistent across models, suggesting that SLURP∗ may pose a more challenging unlearning scenario. This is likely rooted in the dataset's high linguistic and acoustic complexity. While FSC contains only 16 unique entities, SLURP∗ includes more than 5,000. Additionally, the audio recordings were obtained in diverse environments, including both near- and distant-microphone conditions. We hypothesize that these factors likely cause the model to develop highly entangled latent representations: speaker-specific features and semantic intents are closely interleaved in the model weights. As a result, attempting to remove specific training examples via gradient-based unlearning inadvertently damages other capabilities since erasing targeted data points

TABLE VI
UNLEARNING ON SPEECHMASSIVE DE. $F1_T$ DENOTES MACRO F1 ON TEST SET, WHILE $F1_F$ ON FORGET SET. BEST RESULTS (I.E., CLOSEST TO GOLD MODEL FOR F1 AND MIA, HIGHEST FOR OTHERS) ARE IN **BOLD**, SECOND-BEST UNDERLINED. ORIGINAL AND GOLD MODELS ARE HIGHLIGHTED.

Method	XLS-R 53-DE					XLS-R 128					mHuBERT-147				
	$F1_T$	$F1_F$	MIA	CGUM	Speedup	$F1_T$	$F1_F$	MIA	CGUM	Speedup	$F1_T$	$F1_F$	MIA	CGUM	Speedup
Orig.	.778	1.000	.622	.000	1.000×	.584	.841	.621	.000	1.000×	.697	.916	.610	.000	1.000×
Gold	.745	.706	.493	.000	1.000×	.566	.529	.513	.000	1.000×	.694	.684	.497	.000	1.000×
FT	.661	.905	.585	.376	17.79×	.498	.548	.543	.551	16.85×	.712	.970	.622	.000	116.4×
SRL	.732	**.827**	.537	**.505**	17.02×	.570	.467	.566	.507	16.01×	.710	.862	.584	.408	113.1×
NG	.764	.957	.587	.482	**558.9**×	.550	.726	.562	**.691**	**527.8**×	**.697**	.910	.613	.000	**3800**×
NG+	.759	.878	.568	.383	8.780×	.540	.567	.487	.499	8.280×	.689	**.702**	**.565**	**.512**	57.28×
CF-k	.777	1.000	.616	.121	56.93×	.587	.865	.622	.000	53.85×	.702	.925	.621	.000	365.9×
EU-k	.216	.263	**.521**	.410	56.69×	.504	.717	.584	.480	109.1×	.183	.258	.593	.243	361.1×
SalUn	**.737**	.879	.547	.481	15.65×	.573	**.516**	**.525**	.582	3.078×	.712	.880	.585	.396	99.76×
UNSIR	.785	1.000	.619	.063	14.23×	**.565**	.788	.616	.117	13.46×	.706	.919	.609	.026	92.32×
BT	.726	.945	.585	.347	1.430×	.585	.838	.615	.134	9.840×	.709	.823	.597	.251	68.68×
BT-L	.729	.948	.587	.354	12.94×	.584	.786	.576	.454	12.19×	.708	.810	.594	.294	85.45×
SCRUB	.781	1.000	.615	.130	13.41×	.584	.780	.600	.323	13.17×	.709	.915	.618	.000	90.25×

TABLE VII
UNLEARNING ON SPEECHMASSIVE FR. $F1_T$ DENOTES MACRO F1 ON TEST SET, WHILE $F1_F$ ON FORGET SET. BEST RESULTS (I.E., CLOSEST TO GOLD MODEL FOR F1 AND MIA, HIGHEST FOR OTHERS) ARE IN **BOLD**, SECOND-BEST UNDERLINED. ORIGINAL AND GOLD MODELS ARE HIGHLIGHTED.

Method	XLS-R 53-FR					XLS-R 128					mHuBERT-147				
	$F1_T$	$F1_F$	MIA	CGUM	Speedup	$F1_T$	$F1_F$	MIA	CGUM	Speedup	$F1_T$	$F1_F$	MIA	CGUM	Speedup
Orig.	.756	1.000	.635	.000	1.000×	.410	.572	.629	.000	1.000×	.710	.945	.618	.000	1.000×
Gold	.772	.800	.520	.000	1.000×	.469	.460	.509	.000	1.000×	.654	.620	.511	.000	1.000×
FT	.759	.974	.627	.160	18.44×	.400	**.465**	**.539**	.512	18.12×	.677	.908	.602	.310	129.6×
SRL	.767	1.000	.635	.000	17.65×	.431	.334	.558	.486	17.42×	.714	.947	.653	.000	125.8×
NG	.768	**.935**	.617	.337	**61.15**×	.317	.349	.564	**.615**	**597.4**×	.709	.918	.608	.233	**4413**×
NG+	.759	**.943**	.620	.230	9.230×	.394	.000	.988	.000	8.910×	**.661**	**.607**	**.508**	**.692**	64.11×
CF-k	.770	1.000	.624	.217	58.91×	**.436**	.594	.612	.285	58.28×	.715	.946	.611	.168	405.4×
EU-k	.171	.243	**.508**	**.358**	59.26×	.371	.497	.597	.405	58.23×	.170	.229	.570	.375	404.7×
SalUn	.771	1.000	.632	.070	16.24×	.431	.552	.568	.458	16.01×	.714	.947	.653	.000	111.4×
UNSIR	.768	1.000	.633	.048	14.93×	.420	.591	.620	.166	14.68×	.686	.935	.642	.000	102.5×
BT	**.772**	.981	.621	.226	1.830×	.411	.583	.597	.333	1.610×	.704	.897	.580	.498	76.61×
BT-L	.727	.981	.623	.209	13.43×	.412	.574	.591	.372	13.18×	.702	.903	.570	.562	95.19×
SCRUB	.769	1.000	.633	.048	13.92×	.409	.532	.611	.260	13.68×	.714	.950	.608	.218	101.4×

may also corrupt shared features that the model needs to perform well on the data it should retain [35].

Among all methods, NG demonstrates the most consistent performance across datasets and architectures, often ranking first in terms of CGUM. Its efficiency is a standout feature, with speedups frequently exceeding 1000×, and its efficacy is generally high – MIA scores often approach those of the Gold model. This makes NG particularly appealing for scalable MU pipelines. However, despite its strong average performance, NG exhibits critical failure cases in some settings. In particular, we observe that in FSC and TAS, NG occasionally produces MIA scores that exceed those of the original model. This suggests that the model becomes overconfident in misclassifying or distorting the forget set, resulting in *over-forgetting* [27]. In such cases, the MIA metric indicates higher distinguishability of forgotten examples, a counterproductive effect that undermines privacy guarantees. These results highlight that while NG is generally effective, it can induce excessive perturbations when applied without sufficient regularization or early stopping. NG+ improves $F1_T$ and $F1_F$ over NG in several settings, as expected, but does so at a cost. Its efficiency drops significantly (speedups reduced by 50×–100×); however, in some cases, it manages to reach an MIA score closer to the Gold model.

CF-k and EU-k represent efficient unlearning strategies that restrict updates to a targeted subset of model parameters, specifically the final k layers or high-sensitivity blocks, respectively. These methods consistently rank second in efficiency after NG, offering substantial speedups while preserving a degree of model stability. Their selective fine-tuning approach reduces computational overhead without requiring full-network updates. Notably, on simpler datasets such as FSC and TAS, both CF-k and EU-k consistently achieve among the top CGUM scores, indicating successful unlearning with strong utility retention in a fraction of the time required by heavier methods. These results suggest that when the unlearning task is less complex (e.g., limited class diversity, shorter utterances, or well-separated speaker distributions), lightweight, localized updates can be sufficient to remove sensitive data while maintaining overall performance effectively. A clearer distinction between CF-k and EU-k emerges when considering how the final classification layers are modified. In EU-k, the final classification head is completely removed and retrained from random initialization, which disrupts previously learned task-specific structure and, in most cases, leads to a degradation in performance on retained data with respect to CF-k, which continues to fine-tune the already trained classification head, preserving the learned decision boundary and achieving higher utility across most settings.

SRL emerges as a strong and informative baseline, exhibiting a performance profile largely comparable to FT across most experimental settings. It provides a favorable balance between

utility preservation and forgetting efficacy, while maintaining low implementation complexity. SalUn, which constrains gradient updates to the most salient 50% of parameters as identified from the forget set, achieves results that are generally close to those of SRL, reflecting their shared reliance on random-label supervision. However, while SalUn has been shown to be highly effective in other learning paradigms and application domains [36], its advantages do not consistently translate to the speech and SLU setting considered in this work. In particular, its performance exhibits higher variability across datasets and model architectures and does not reliably surpass SRL. This empirical evidence suggests that, in SLU tasks, saliency-aware parameter masking alone is insufficient to yield systematic improvements in unlearning efficacy. We argue that this constitutes an important domain-specific insight: unlike other modalities, speech representations appear to rely on more distributed and entangled parameter interactions, limiting the effectiveness of sparsely targeted updates. Nevertheless, SalUn remains an appealing building block due to its parameter efficiency and compatibility with pre-trained models. Future work may explore its integration with more aggressive unlearning mechanisms tailored to speech, such as NG, which is particularly effective at disrupting localized representations associated with the forget set. Combining SalUn's selective focus on salient parameters with NG's capacity to erase memorized patterns could enable targeted gradient ascent on the most influential weights, while mitigating unnecessary degradation of model performance.

BT and BT-L adopt a distillation-based approach, utilizing both competent and incompetent teacher models to guide the student during unlearning. While BT employs a fully trained model as the bad teacher, BT-L replaces it with a fixed random predictor, significantly reducing memory and computational requirements by requiring only two models to be loaded instead of three. Notably, this simplification does not appear to compromise the efficacy or utility of the resulting model. On the contrary, BT-L often achieves comparable or even superior results to BT in terms of efficacy and utility, with a critical gain in efficiency. This gain translates into strong CGUM scores, with BT-L ranking as the second-best method in SLURP$*$, ITALIC, and SpeechMASSIVE fr-FR. These findings suggest that using a random teacher can be a practical and lightweight alternative for unlearning in speech tasks, especially when resource constraints are a consideration.

SCRUB and UNSIR achieve moderate results. While they offer clear theoretical motivation, their performance in practice is inconsistent across CGUM components. UNSIR often ranks high in utility but lacks efficacy, and SCRUB achieves only modest gains across all three metrics.

V. Conclusion

UnSLU-BENCH+ extends the UnSLU-BENCH benchmark to provide a comprehensive evaluation of machine unlearning methods in spoken language understanding (SLU). Our benchmark incorporates new datasets, including the introduction of the *Timers & Such* dataset; additional pretrained models such as WavLM Base+ and mHuBERT-147; and three new unlearning strategies: Selective Random Labelling (SRL), Exact Unlearning Last k Layers (EU-k), and Saliency-based Unlearning (SalUn). To evaluate these methods, we adopt macro F1, MIA, and runtime speedup as representative metrics for utility, efficacy, and efficiency, respectively. We also introduce CGUM – a classification-aware extension of the GUM score – which offers a bounded and interpretable aggregate measure of unlearning quality. In addition, we demonstrated that the expected upper bound of macro F1 for any random classifier (majority, uniform, or prior-based) is $1/C$, providing a grounded foundation for the normalization used in CGUM.

Our empirical findings show that NG remains a robust method overall, consistently achieving high CGUM scores across languages and architectures. However, our benchmark highlights that different datasets pose varying levels of unlearning difficulty, with SLURP$*$ emerging as particularly challenging due to its complexity and the required identity-disjoint splits. Notably, the results also reveal that no single method dominates across all datasets and model configurations; each approach exhibits trade-offs between utility, efficacy, and efficiency depending on the context. This highlights the importance of context-aware method selection and further underscores the need for composite metrics, such as CGUM, to capture these nuanced differences.

We hope that this benchmark will foster future research into efficient, effective, and deployable unlearning methods for speech applications. Beyond establishing a standard evaluation framework for SLU, our work also introduces CGUM as a principled and general-purpose metric for assessing trade-offs between utility, efficacy, and efficiency in any classification-based unlearning scenario.

Appendix A
Boundary on the Utility of Naïve Classifiers

This section provides a theoretical justification for the choice of $1/C$ as the expected upper bound for random classifiers in multi-class settings. In particular, we demonstrate that three baselines cannot exceed an expected macro-F1 score of $1/C$, where C is the number of target classes. These include: (1) the *majority-class classifier*, which always predicts the most frequent class; (2) the *uniform random classifier*, which samples labels uniformly from the set of possible classes; and (3) the *prior-based random classifier*, which samples labels proportionally to the empirical class distribution.

Majority Class: The *majority-class classifier* always predicts the most frequent class in the dataset, denoted by k^*, where

$$k^* = \arg\max_{k \in \{1,\ldots,C\}} p_k.$$

This strategy is deterministic and, despite yielding a high overall accuracy in imbalanced settings, it still performs poorly under macro-F1 because it entirely overlooks minority classes. Let $y \in \{1, \ldots, C\}$ be the true label, and $\hat{y} = k^*$ the fixed prediction. Then:

$$\mathbb{E}[\text{TP}_{k^*}] = \mathbb{P}(y = k^* \wedge \hat{y} = k^*) = p_{k^*}$$
$$\mathbb{E}[\text{FP}_{k^*}] = \mathbb{P}(y \neq k^* \wedge \hat{y} = k^*) = 1 - p_{k^*}$$
$$\mathbb{E}[\text{FN}_{k^*}] = \mathbb{P}(y = k \wedge \hat{y} \neq k) = 0$$

From these, the precision, recall, and F1-score for the majority class k^* are:

$$\Pr_{k^*} = \frac{\mathbb{E}[\text{TP}_k]}{\mathbb{E}[\text{TP}_k] + \mathbb{E}[\text{FP}_k]} = \frac{p_{k^*}}{p_{k^*} + (1 - p_{k^*})} = p_{k^*}$$

$$\text{Re}_{k^*} = \frac{\mathbb{E}[\text{TP}_k]}{\mathbb{E}[\text{TP}_k] + \mathbb{E}[\text{FN}_k]} = \frac{p_{k^*}}{p_{k^*}} = 1$$

$$\text{F1}_{k^*} = \frac{2 \cdot \Pr_k \cdot \text{Re}_k}{\Pr_k + \text{Re}_k} = \frac{2 \cdot p_{k^*} \cdot 1}{p_{k^*} + 1} = \frac{2p_{k^*}}{p_{k^*} + 1}$$

In the following, we define precision, recall, and F1 by applying their standard formulas to the analytically derived expected counts of TP, FP, and FN. While this does not correspond to the formal expectation of the F1 random variable, it provides a valid approximation for analyzing the behavior of deterministic or probabilistic classifiers under known label distributions. For every other class $k \neq k^*$, the classifier never predicts class k, so:

$$\mathbb{E}[\text{TP}_k] = \Pr_k = \text{Re}_k = \text{F1}_k = 0$$

The expected macro-F1 becomes:

$$\text{F1}_{\text{macro}} = \frac{1}{C}\left(\frac{2p_{k^*}}{p_{k^*} + 1} + \sum_{k \neq k^*} 0\right) = \frac{1}{C} \cdot \frac{2p_{k^*}}{p_{k^*} + 1}$$

This value is upper bounded by $\frac{1}{C}$, since the function $\frac{2p}{p+1} \leq 1$ for all $p \in [0, 1]$. Specifically:

$$\text{F1}_{\text{macro}} \leq \frac{1}{C}$$

with equality if and only if $p_{k^*} = 1$, i.e., the classification task is degenerate and only one class is present.

Uniform Random Classifier: The *uniform random classifier* assigns each class label independently and uniformly at random, regardless of the input or the true distribution of labels. In a classification problem with C classes and class priors $p_k = \mathbb{P}(y = k)$, we compute the expected macro-F1 achieved by such a strategy.

Let $y \in \{1, \ldots, C\}$ denote the true label and $\hat{y}$ the predicted label. Under uniform random prediction, the probability of predicting any specific class is defined according to:

$$\mathbb{P}(\hat{y} = k) = \frac{1}{C} \quad \text{for all } k.$$

We now compute the expected number of true positives (TP), false positives (FP), and false negatives (FN). For each class $k \in \{1, \ldots, C\}$:

$$\mathbb{E}[\text{TP}_k] = p_k \cdot \frac{1}{C} \qquad \mathbb{E}[\text{FP}_k] = (1 - p_k) \cdot \frac{1}{C}$$

$$\mathbb{E}[\text{FN}_k] = p_k \cdot \left(1 - \frac{1}{C}\right)$$

From the above, the precision, recall, and F1 for the class k are:

$$\Pr_k = \frac{p_k \cdot \frac{1}{C}}{p_k \cdot \frac{1}{C} + (1 - p_k) \cdot \frac{1}{C}} = p_k$$

$$\text{Re}_k = \frac{p_k \cdot \frac{1}{C}}{p_k} = \frac{1}{C}$$

$$\text{F1}_k = \frac{2 \cdot p_k \cdot \frac{1}{C}}{p_k + \frac{1}{C}} = \frac{2p_k}{C \cdot p_k + 1}$$

And the expected macro-F1, obtained by averaging over all classes:

$$\text{F1}_{\text{macro}} = \frac{1}{C} \sum_{k=1}^{C} \frac{2p_k}{C \cdot p_k + 1}$$

To prove that the expected value of F1_{macro} is upper bounded by $\frac{1}{C}$, it is sufficient to show that the value of the summation is ≤ 1. To demonstrate this result, we make use of *Jensen's Inequality* for concave functions.

Jensen's Inequality: Let $f : \mathbb{R} \to \mathbb{R}$ be a concave function and let $\{x_1, \ldots, x_n\}$ be real numbers. If $\{w_1, \ldots, w_n\}$ is a set of non-negative weights such that $\sum_{i=1}^{n} w_i = 1$, then:

$$f\left(\sum_{i=1}^{n} w_i x_i\right) \geq \sum_{i=1}^{n} w_i f(x_i).$$

Equality holds if and only if all x_i are equal, or f is affine on the convex hull of $\{x_i\}$. We define the function:

$$f(p) = \frac{2p}{C \cdot p + 1}$$

and rewrite the macro-F1 as:

$$\text{F1}_{\text{macro}} = \frac{1}{C} \sum_{k=1}^{C} f(p_k)$$

Step 1. Proof that $f(p)$ is concave: We compute the first and second derivatives of $f(p)$:

$$f'(p) = \frac{2(C \cdot p + 1) - 2C \cdot p}{(C \cdot p + 1)^2} = \frac{2}{(C \cdot p + 1)^2}$$

$$f''(p) = \frac{\partial}{\partial p}\left(\frac{2}{(C \cdot p + 1)^2}\right) = -\frac{4C}{(C \cdot p + 1)^3} < 0$$

for all $p > 0$. Hence, f is strictly concave on $(0, 1]$.

Step 2. Application of Jensen's Inequality: Given that f is concave and $\sum_{k=1}^{C} p_k = 1$, Jensen's inequality yields:

$$\frac{1}{C} \sum_{k=1}^{C} f(p_k) \leq f\left(\frac{1}{C} \sum_{k=1}^{C} p_k\right) = f\left(\frac{1}{C}\right)$$

We compute:

$$f\left(\frac{1}{C}\right) = \frac{2 \cdot \frac{1}{C}}{C \cdot \frac{1}{C} + 1} = \frac{2/C}{1 + 1} = \frac{1}{C}$$

Hence we conclude that:

$$\text{F1}_{\text{macro}} = \frac{1}{C} \sum_{k=1}^{C} f(p_k) \leq \frac{1}{C}$$

Prior-based Sampling: The *prior-based random classifier* assigns class labels independently according to the observed class distribution: each label is sampled with probability $p_k = \mathbb{P}(y = k)$. In the following, we compute the expected macro-F1 score achieved by this classifier. Let $y \in \{1, \ldots, C\}$ be the true label and $\hat{y}$ the predicted label. Since $\hat{y} \sim \text{Categorical}(p_1, \ldots, p_C)$, the probability of correctly predicting class k is p_k^2. We now compute the expected values for true positives (TP), false positives (FP), and false negatives (FN):

$$\mathbb{E}[\text{TP}_k] = p_k^2 \qquad \mathbb{E}[\text{FP}_k] = p_k(1 - p_k)$$

$$\mathbb{E}[\text{FN}_k] = p_k(1 - p_k)$$

It follows that the precision, recall, and F_1 for class k are:

$$\underset{k}{\mathrm{Pr}} = \frac{p_k^2}{p_k^2 + p_k(1-p_k)} = p_k$$

$$\mathrm{Re}_k = \frac{p_k^2}{p_k^2 + p_k(1-p_k)} = p_k$$

$$\mathrm{F1}_k = \frac{2 \cdot p_k \cdot p_k}{p_k + p_k} = p_k$$

Finally, the expected macro-F1 is given by:

$$\mathrm{F1}_{\mathrm{macro}} = \frac{1}{C}\sum_{k=1}^{C} \mathrm{F1}_k = \frac{1}{C}\sum_{k=1}^{C} p_k = \frac{1}{C}$$

This result holds for any class distribution, and shows that even a classifier perfectly aligned with the class priors achieves at most $1/C$ in macro-F1.

REFERENCES

[1] P. Voigt and A. Von dem Bussche, "The EU general data protection regulation (GDPR)," *A Practical Guide*, vol. 10, 1st ed. Cham, Switzerland: Springer International Publishing, 2017, pp. 10–5555.

[2] "Personal information protection law of the people's Republic of China," Nov. 2021. [Online]. Available: https://personalinformationprotectionlaw.com/

[3] A. Mantelero, "The EU proposal for a general data protection regulation and the roots of the right to be forgotten," *Comput. Law Secur. Rev.*, vol. 29, no. 3, pp. 229–235, 2013.

[4] D. Choi and D. Na, "Towards machine unlearning benchmarks: Forgetting the personal identities in facial recognition systems," 2023, *arXiv:2311.02240*.

[5] K. Grimes, C. Abidi, C. Frank, and S. Gallagher, "Gone but not forgotten: Improved benchmarks for machine unlearning," 2024, *arXiv:2405.19211*.

[6] R. Eldan and M. Russinovich, "Who's Harry Potter? Approximate unlearning in LLMs," 2023, *arXiv:2310.02238*.

[7] Z. Jin, et al., "RWKU: Benchmarking real-world knowledge unlearning for large language models," *Adv. Neural Inf. Process. Syst.*, vol. 37, 2024,

[8] J. Cheng and H. Amiri, "Speech unlearning," in *Proc. Conf. Interspeech*, 2025, pp. 3209–3213.

[9] O. C. Phukan et al., "Towards machine unlearning for paralinguistic speech processing," in *Proc. Conf. Interspeech*, 2025, pp. 4473–4477.

[10] A. Koudounas, C. Savelli, F. Giobergia, and E. Baralis, ""Alexa, can you forget me?" Machine unlearning benchmark in spoken language understanding," in *Proc. Conf. Interspeech*, 2025, pp. 1768–1772.

[11] L. Lugosch, P. Papreja, M. Ravanelli, A. Heba, and T. Parcollet, "Timers and such: A practical benchmark for spoken language understanding with numbers," in *Proc. 35th Conf. Neural Inf. Process. Syst.*, 2021, pp. 1–11.

[12] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 3451–3460, 2021.

[13] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 12449–12460.

[14] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli, "Unsupervised cross-lingual representation learning for speech recognition," in *Proc. INTERSPEECH*, 2021, pp. 2426–2430.

[15] A. Babu et al., "XLS-R: Self-supervised cross-lingual speech representation learning at scale," in *Proc. INTERSPEECH*, 2022, pp. 2278–2282.

[16] S. Chen et al., "WavLM: Large-scale self-supervised pre-training for full stack speech processing," *IEEE J. Sel. Topics Signal Process.*, vol. 16, no. 6, pp. 1505–1518, Oct. 2022.

[17] L. Lugosch, M. Ravanelli, P. Ignoto, V. S. Tomar, and Y. Bengio, "Speech model pre-training for end-to-end spoken language understanding," in *Proc. INTERSPEECH* 2019, pp. 814–818.

[18] E. Bastianelli, A. Vanzo, P. Swietojanski, and V. Rieser, "SLURP: A spoken language understanding resource package," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, 2020, pp. 7252–7262.

[19] M. Zanon Boito, V. Iyer, N. Lagos, L. Besacier, and I. Calapodescu, "mHuBERT-147: A compact multilingual huBERT model," in *Proc. Interspeech*, 2024, pp. 3939–3943.

[20] A. Koudounas et al., "ITALIC: An Italian intent classification dataset," in *Proc. INTERSPEECH*, 2023, pp. 2153–2157.

[21] B. Lee, I. Calapodescu, M. Gaido, M. Negri, and L. Besacier, "Speechmassive: A multilingual speech dataset for SLU and beyond," in *Proc. INTERSPEECH*, 2024, pp. 817–821.

[22] A. Golatkar, A. Achille, and S. Soatto, "Eternal sunshine of the spotless net: Selective forgetting in deep networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9304–9312.

[23] S. Goel, A. Prabhu, A. Sanyal, S.-N. Lim, P. Torr, and P. Kumaraguru, "Towards adversarial evaluations for inexact machine unlearning," 2022, *arXiv:2201.06640*.

[24] M. Kurmanji, P. Triantafillou, J. Hayes, and E. Triantafillou, "Towards unbounded machine unlearning," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2024, pp. 1957–1987.

[25] J. Hayes, I. Shumailov, E. Triantafillou, A. Khalifa, and N. Papernot, "Inexact unlearning needs more careful evaluations to avoid a false sense of privacy," 2024, *arXiv:2403.01218*.

[26] M. Jagielski et al., "Measuring forgetting of memorized training examples," 2022, *arXiv:2207.00099*.

[27] W. Shi et al., "MUSE: Machine unlearning six-way evaluation for language models," 2024, *arXiv:2407.06460*.

[28] A. K. Tarun, V. S. Chundawat, M. Mandal, and M. Kankanhalli, "Fast yet effective machine unlearning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 9, pp. 13046–13055, Sep. 2024.

[29] P. Maini, Z. Feng, A. Schwarzschild, Z. C. Lipton, and J. Z. Kolter, "TOFU: A task of fictitious unlearning for LLMs," 2024, *arXiv:2401.06121*.

[30] L. Bourtoule et al., "Machine unlearning," in *Proc. IEEE Symp. Secur. Privacy*, 2021, pp. 141–159.

[31] W. Choi et al., "Large-scale training data attribution for music generative models via unlearning," 2025, *arXiv:2506.18312*.

[32] P. Warden, "Speech Commands: A public dataset for single-word speech recognition," 2017. [Online]. Available: http://download.tensorflow.org/data/speech_commands_v0

[33] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Comput. Speech Lang.*, vol. 60, 2020, Art. no. 101027.

[34] V. S. Chundawat, A. K. Tarun, M. Mandal, and M. Kankanhalli, "Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher," in *Proc. AAAI Conf. Artif. Intell.*, 2023, pp. 7210–7217.

[35] K. Zhao, M. Kurmanji, G.-O. Bărbulescu, E. Triantafillou, and P. Triantafillou, "What makes unlearning hard and what to do about it," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, pp. 12293–12333.

[36] Y. Zhang et al., "Unlearncanvas: A stylized image dataset for enhanced machine unlearning evaluation in diffusion models," in *Proc. 38th Int. Conf. Neural Inf. Process. Syst.*, Red Hook, NY, USA: Curran Associates Inc., 2024, pp. 96387–96423.

Open Access funding provided by 'Politecnico di Torino' within the CRUI CARE Agreement