"Alexa, can you *forget* me?"
Machine Unlearning Benchmark in Spoken Language Understanding

Alkis Koudounas[1], Claudio Savelli*[1], Flavio Giobergia[1], Elena Baralis[1]*

[1]Politecnico di Torino, Italy

name.surname@polito.it

Abstract

Machine unlearning, the process of efficiently removing specific information from machine learning models, is a growing area of interest for responsible AI. However, few studies have explored the effectiveness of unlearning methods on complex tasks, particularly speech-related ones. This paper introduces UnSLU-BENCH, the first benchmark for machine unlearning in spoken language understanding (SLU), focusing on four datasets spanning four languages. We address the unlearning of data from specific speakers as a way to evaluate the quality of potential *"right to be forgotten"* requests. We assess eight unlearning techniques and propose a novel metric to simultaneously better capture their efficacy, utility, and efficiency. UnSLU-BENCH sets a foundation for unlearning in SLU and reveals significant differences in the effectiveness and computational feasibility of various techniques.

Index Terms: machine unlearning, spoken language understanding, speech recognition, transformers

1. Introduction

Machine unlearning (MU) refers to the process of efficiently removing specific data points from a trained machine learning model without the need for a complete retraining from scratch [1]. This capability is crucial for complying with data privacy regulations, such as the European Union's General Data Protection Regulation (GDPR) [2] and the California Consumer Privacy Act (CCPA) [3], which promote the *"right to be forgotten"*. By removing the influence of specific data points on machine learning models, MU helps maintain compliance with legal standards and protects user privacy [4].

In the context of speech, MU plays an even more important role. Speech data often contains personally identifiable information, making it particularly sensitive [5–7]. The ability to unlearn specific data ensures that individuals can exercise control over their personal information, thus increasing trust in AI systems. In addition, unlearning mechanisms can help reduce the influence of unreliable data and mitigate biases, contributing to the development of more fair speech recognition models [8–10].

One important example is the interaction with vocal assistants. These models process large amounts of user speech data to perform tasks such as intent classification [11]. Ensuring that these systems can unlearn data from individual users upon request is essential to maintain user autonomy and privacy [12, 13]. Despite the critical nature of this capability, there is a non-negligible gap in existing research on MU tailored to speech tasks. While MU has been explored in other domains, including text [14, 15] and image [16, 17] processing, its application to speech tasks remains under-developed.

The authors of [18] first explore the application of MU techniques for audio and speech processing. However, their study is limited to audio classification tasks and uses only a single speech dataset focused on keyword spotting, a task semantically much less complex than the intent classification challenges faced in Spoken Language Understanding (SLU). This emphasizes the need for MU techniques specifically designed to handle the complexities of SLU tasks.

To fill this gap, we introduce UnSLU-BENCH, the first comprehensive benchmark for machine unlearning in SLU. It includes four intent classification datasets in four different languages: Fluent Speech Commands (FSC) [19] and SLURP [20] in English, ITALIC [21] in Italian, and SpeechMASSIVE [22] in both German and French. For each dataset, we evaluate two transformer models, wav2vec 2.0 [23] and HuBERT [24] for English datasets, and XLS-R-128 [25] and XLS-R-53 [26] for the other languages. The latter model has been fine-tuned on Automatic Speech Recognition (ASR) for each target language.

UnSLU-BENCH offers a complete analysis of the effectiveness of MU techniques across different model architectures and dataset complexities. We evaluate eight distinct unlearning methods, examining both their effectiveness and computational efficiency in removing specific speakers' data from the models.

Our contributions can be summarized in four points: (1) we introduce the first benchmark for machine unlearning in SLU, with four datasets in four languages and two models per dataset; (2) we evaluate eight unlearning techniques, measuring their impact on data removal and model performance; (3) we propose GUM, a novel MU metric considering efficacy, efficiency, and utility of unlearning methods simultaneously; and (4) we provide an in-depth analysis of unlearning performance across datasets, languages, model sizes and architectures.

This benchmark[1] aims to advance the development of privacy-preserving techniques in speech tasks, facilitating future research on more trustworthy voice assistant systems.

2. Machine Unlearning

2.1. Problem definition

We assume a given model θ that has been trained on a SLU dataset $\mathcal{D}$. Each data point is represented as a triplet $(x, y, s) \in \mathcal{D}$, where x denotes the utterance, y indicates the target intent, and s is the speaker's identity. We refer to the set of all speakers in the training set as $\mathcal{S}$. We now assume that a subset of speakers $\mathcal{S}_f \subset \mathcal{S}$ asks for their data to be deleted. From a data perspective, this simply implies deleting from the database all

* Both authors contributed equally to this work.

[1] github.com/koudounasalkis/UnSLU-BENCH

samples $\mathcal{D}_f = \{(x, y, s) | s \in \mathcal{S}_f)\}$, referred to as the *forget set*. However, those samples have affected the learning process of θ. We refer to the remaining samples, i.e., $\mathcal{D}_r = \mathcal{D} \setminus \mathcal{D}_f$ as the *retain set*. MU is tasked to remove the influence of points in $\mathcal{D}_f$ from θ. In other words, MU algorithms produce a new model $\hat{\theta} = \phi(\theta, \mathcal{D}_r, \mathcal{D}_f)$. As introduced in [27], we adopt the idea of a *gold model*, i.e., the model θ' that has been trained using only $\mathcal{D}_r$. The gold model represents MU's ideal target (i.e., we want $\hat{\theta} \approx \theta'$). However, retraining the model from scratch for every *forget* request is generally unfeasible, especially for larger models – hence the need for unlearning methods.

2.2. Unlearning methods

`UnSLU-BENCH` includes eight MU techniques as follows.

Fine-Tuning (`FT`) continues to train the model using all $\mathcal{D}_r$ for one epoch. Thus, being $\mathcal{D}_f$ unseen for one additional epoch, it should be less influential than $\mathcal{D}_r$. This method is commonly used as a baseline in the unlearning framework.

Negative Gradients (`NG`) [27] finetunes the model using all $\mathcal{D}_f$ only. Instead of a normal `FT`, the gradient direction is reversed during the backpropagation to make the model forget $\mathcal{D}_f$.

NegGrad+ (`NG+`) [28,29] was proposed as an extension to Negative Gradients to avoid the so-called *"catastrophic forgetting"*, i.e., the destruction of the model's utility. To do this, in addition of `NG` on $\mathcal{D}_f$, `FT` is done on the whole $\mathcal{D}_r$.

Catastrophically forgetting the last k layers (`CF-`k) [30] applies `FT` only to the final k layers of the model. In this way, the unlearning model is faster, as it applies backpropagation on those layers with the most relevant representations and keeps the rest of the network untouched.

UNSIR (`UNSIR`) [31] includes two phases, first it destroys the model ("*impair*"), and then it rebuilds its utility ("*repair*"). In the first, an error-maximizing noise is created for each element of $\mathcal{D}_f$, which is then used to train the model in combination with `FT`. The second phase consists of another epoch of `FT` only.

Bad Teaching (`BT`) [32] uses a competent teacher, i.e., a copy of the original model, and an incompetent teacher, i.e., the same model not fine-tuned on the task, in a distillation setup to train a student to behave like the first on $\mathcal{D}_r$ and like the second on $\mathcal{D}_f$. We also evaluate a **light variant (`BT-L`)** of the method with a random prediction generator as the incompetent teacher.

SCRUB (`SCRUB`) [29] uses a teacher-student setup with a single teacher, i.e., a copy of the original model. This method combines three different losses: a first loss maximizes student similarity with the teacher on $\mathcal{D}_r$, a second loss minimizes it on $\mathcal{D}_f$, while a third task loss improves the final model utility.

2.3. Unlearning metrics

The evaluation of unlearning algorithms is not trivial. In literature [33], the three main aspects of interest are *efficacy* (whether the unlearning process effectively erased the required information), *efficiency* (how costly the unlearning process is) and *utility* (whether the unlearned model still successfully addresses the original task). We argue that all three aspects should be considered at the same time. Ignoring any one of them can lead to trivial solutions. If we ignore *efficacy*, the best "unlearned" model is simply the original model. Since we are not checking whether the model has actually forgotten anything, this maximizes efficiency (no computational cost) and utility (performance remains the same). If we ignore *efficiency*, the best solution is to retrain the model from scratch. As we do not consider the cost of retraining, this maximizes efficacy (the model has never seen the forget set) and utility (the model performs as well as possible). If we ignore *utility*, the best unlearning method is a model that predicts random values. Since we do not care about the quality of the results, this maximizes efficacy (the model does not retain any knowledge of the forget set) and efficiency (no additional computation is needed).

Table 1: ***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	**FSC**									
	wav2vec 2.0					**HuBERT**				
	$\mathbf{F1_T}$	$\mathbf{F1_F}$	**MIA**	**GUM**	**Speedup**	$\mathbf{F1_T}$	$\mathbf{F1_F}$	**MIA**	**GUM**	**Speedup**
`Orig.`	.994	1.00	.508	.000	1.00×	.993	1.00	.511	.000	1.00×
`Gold`	.993	.997	.503	.000	1.00×	.991	.996	.507	.000	1.00×
`FT`	**.993**	.999	**.504**	.517	7.960×	.979	.993	**.508**	.514	7.690×
`NG`	.987	.976	.501	**.816**	**206.9**×	**.992**	**.996**	.514	.000	**201.1**×
`NG+`	.994	.994	.493	.000	4.030×	.979	.929	.510	.336	3.900×
`CF-`k	.994	1.00	.501	.606	16.97×	.993	1.00	.505	**.642**	26.70×
`UNSIR`	.991	1.00	.506	.447	6.550×	.994	.998	**.508**	.484	6.380×
`BT`	**.993**	1.00	.508	.000	4.780×	.993	.999	.504	.363	4.650×
`BT-L`	.994	**.996**	.506	.431	5.870×	.993	.997	**.506**	.464	5.690×
`SCRUB`	.994	1.00	.506	.439	6.210×	.993	.998	**.508**	.479	6.220×

Despite these considerations, very few works in literature account for combinations of some metrics. NoMUS [28] considers efficacy and utility together. The work in [34] selects the most effective method given a utility threshold, while [35] chooses the hyperparameter configuration that maximizes efficacy and then evaluates models based on their efficacy.

In addition, metrics in literature are typically not considered in relation to the *gold model performance*. Since MU aims to produce a model that resembles the model retrained from scratch, we argue that it is fundamental to ground all measures to the gold model. We acknowledge, of course, that having access to the gold model is a constraint that is generally only met during model validation and not in deployment. This is a limitation that affects the entire field of MU, and no general, gold-free solution has been proposed yet.

In this work, we introduce a new metric, the *Global Unlearning Metric* (GUM), which considers all three aspects simultaneously, with comparisons against the gold model. We quantify utility as the similarity in performance between the gold and the unlearned models as $U = 1 - |F1_T^{(g)} - F1_T^{(u)}|$, based on the macro F1 scores[2] on a test set ($F1_T^{(g)}$ and $F1_T^{(u)}$). We use the MIA (Membership Inference Attack), a commonly adopted metric in unlearning [33], to quantify the efficacy of a method. More specifically, the MIA of the gold model ($\text{MIA}^{(g)}$) is the ideal target, whereas the MIA of the original model $\text{MIA}^{(o)}$ is the starting point. Based on these boundaries, we quantify the efficacy E as:

$$E = 1 - \left(\frac{\text{MIA}'^{(u)} - \text{MIA}'^{(g)}}{\text{MIA}^{(o)} - \text{MIA}'^{(g)}} \right)^2 ,$$

where $\text{MIA}'^{(u)} = \min\{\text{MIA}^{(u)}, \text{MIA}^{(o)}\}$ and $\text{MIA}'^{(g)} = \min\{\text{MIA}^{(g)}, (\text{MIA}'^{(u)} + \text{MIA}^{(o)})/2\}$ are saturated versions of the gold and unlearned MIA that guarantee that $E \in [0, 1]$ in edge cases. The quantity is squared to increase similarities for small gold-unlearned MIA distances. Finally, we quantify the

[2] Other scenarios may require a change in utility function.

Table 2: ***Comparison of unlearning methods on SLURP* and ITALIC.*** *Best results are in **bold**, second-best underlined.*

Method	SLURP*										ITALIC									
	wav2vec 2.0					HuBERT					XLS-R 128					XLS-R 53-IT				
	$\mathbf{F1_T}$	$\mathbf{F1_F}$	MIA	GUM	Speedup	$\mathbf{F1_T}$	$\mathbf{F1_F}$	MIA	GUM	Speedup	$\mathbf{F1_T}$	$\mathbf{F1_F}$	MIA	GUM	Speedup	$\mathbf{F1_T}$	$\mathbf{F1_F}$	MIA	GUM	Speedup
Orig.	.689	1.000	.628	.000	1.000×	.712	1.000	.613	.000	1.000×	.688	.894	.632	.000	1.000×	.778	1.000	.615	.000	1.000×
Gold	.707	.711	.506	.000	1.000×	.704	.715	.492	.000	1.000×	.643	.568	.532	.000	1.000×	.784	.736	.478	.000	1.000×
FT	.638	**.970**	.648	.000	83.78×	.734	1.000	.611	.088	79.00×	**.638**	.671	.555	.590	30.80×	.711	.850	.550	.551	31.10×
NG	.695	.986	.604	**.563**	**1748**×	.718	.959	.587	**.587**	**1654**×	.679	.868	.603	**.646**	**613.4**×	.590	.621	**.525**	**.766**	**623.0**×
NG+	.701	.995	**.603**	.446	41.63×	.630	**.852**	**.453**	.578	39.30×	.658	.001	.932	.000	15.14×	.743	.936	.582	.418	15.37×
CF-k	**.709**	1.000	.626	.089	291.9×	.715	1.000	.608	.196	274.2×	.677	.871	.626	.253	98.59×	**.781**	1.000	.609	.201	98.99×
UNSIR	.673	1.000	.637	.000	64.07×	.722	1.000	.613	.000	60.44×	.636	.830	.621	.328	22.01×	.775	1.000	.612	.109	22.26×
BT	.710	.999	.619	.275	50.35×	.711	1.000	.613	.000	47.42×	.683	**.639**	.481	.504	17.90×	.731	**.848**	.557	.491	17.94×
BT-L	.680	.995	.637	.000	61.74×	.685	.907	.558	.578	58.11×	.686	.651	.518	.558	22.02×	.729	.876	.564	.499	22.21×
SCRUB	.697	.999	.608	.429	64.82×	**.704**	1.000	.600	.350	65.40×	.442	.357	**.533**	.536	23.25×	.770	.990	.610	.164	22.66×

efficiency as the ratio of the logarithms of the unlearning time $T^{(u)}$ and gold retraining time $T^{(g)}$.

$$T = 1 - \frac{log(T^{(u)}+1)}{log(T^{(g)}+1)}.$$

We define GUM as the weighted harmonic mean between these three quantities:

$$GUM = \frac{(1+\alpha+\beta)UET}{\alpha ET + \beta UT + UE}.$$

The α and β parameters assign different importance to the three quantities. Here, we weigh all quantities equally ($\alpha = \beta = 1$).

3. Experimental Setup

Datasets. UnSLU-BENCH includes four publicly available datasets: FSC [19] and SLURP [20] for English, ITALIC [21] for Italian, and SpeechMASSIVE [22] for German and French. The FSC dataset is relatively straightforward, containing 31 intents. In contrast, SLURP, ITALIC, and SpeechMASSIVE are substantially larger, with 60 intents and greater linguistic diversity. ITALIC and SpeechMASSIVE are multilingual extensions of SLURP, covering Italian, and German-French, respectively[3]. Unlike other datasets, SLURP does not provide speaker-independent splits, which are, however, required by MU techniques to be effective. In fact, the identities present in the retain, forget, and test sets must be exclusive to successfully apply and evaluate unlearning methods. To address this, we propose new speaker-independent splits[4]. In the following tables, we refer to the new dataset as SLURP*. For the other datasets, we use the original splits, with the identities already separated between train and test splits. To create the *forget* set, individuals with at least 100 associated audio samples were randomly taken from each dataset. This ensures that a sufficiently representative number of points were used for training the model for each individual to be forgotten. This implies that the size of $\mathcal{D}_f$ with respect to $\mathcal{D}_t$ is 2.5–5% on the different datasets. In this way, we simulate a real case scenario of a possible request to delete one's personal data from a model's training.

Models. For each dataset, we fine-tune two transformer models. For the English datasets, we use wav2vec 2.0 [23] and HuBERT [24] in their base sizes. For the multilingual datasets, we use XLS-R 128 [25] and XLS-R 53 [26]. The latter is ASR-fine-tuned for the target language (e.g., Italian, German, French).

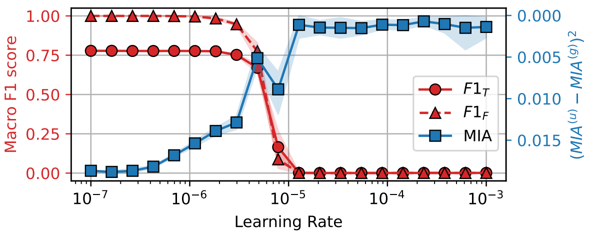

Figure 1: *Trade-off between utility (test and forget F1) and efficacy (MIA) on NG, as the LR changes (ITALIC, XLS-R 53-IT).*

Unlearning Methods. For each unlearner, we use two different sets of learning rates as parameter tuning depending on how destructive they are. Specifically, we employ 5e-07, 1e-06, and 5e-06 for NG, NG+, BT, BT-L, SCRUB, and 1e-05, 5e-05, and 1e-04 for FT, CF-k, UNSIR. For each experiment, we consider the method that achieves the highest utility, efficacy, and efficiency as the best through the use of GUM. Moreover, considering that the original implementation of UNSIR was made to forget entire classes within the dataset, we use the version proposed by [28], applicable also to individual samples.

4. Results

In the following, we present experiments conducted to explore the behavior of MU techniques in SLU.

Benchmark results. The analysis of Tables 1–3 shows distinct patterns in unlearning methods for MU performance across different models and datasets. The best F1 and MIA results are measured by their distance from our target's gold model.

NG consistently achieves the highest GUM scores. For instance, for wav2vec 2.0, it outperforms the second-best approach by +35% on FSC and +26% in SLURP*. For the larger multilingual XLS-R 53 model, it improves GUM by +39% on ITALIC and SpeechMASSIVE de-DE and by +48% on SpeechMASSIVE fr-FR. This improvement comes from its exceptional efficiency (speedups up to 1748× on FSC) and strong efficacy (MIA close to gold models, often ranking first or second among competitors, especially in multilingual datasets). NG+ achieves slightly higher $F1_T$ and $F1_F$ scores than NG in some cases, with comparable MIA scores. However, its overall GUM score is significantly lower as its speedup is one order of magnitude lower than NG. NG+ also suffers the *"catastrophic forgetting"* phenomena in some cases, such as XLS-R 128 ($F1_F$ = .001 on ITALIC, .008 on SpeechMASSIVE fr-FR). FT balances utility and efficacy well for complex models. For example, XLS-R 128

[3] SpeechMASSIVE covers 12 languages, but we focus on German and French only.

[4] These splits are publicly available in our project repository.

Table 3: ***Comparison of unlearning methods on SpeechMASSIVE de-DE and fr-FR.*** *Best results are in* ***bold****, second-best underlined.*

Method	de-De										fr-FR									
	XLS-R 128					XLS-R 53-DE					XLS-R 128					XLS-R 53-FR				
	F1$_T$	F1$_F$	MIA	GUM	Speedup	F1$_T$	F1$_F$	MIA	GUM	Speedup	F1$_T$	F1$_F$	MIA	GUM	Speedup	F1$_T$	F1$_F$	MIA	GUM	Speedup
Orig.	.584	.841	.621	.000	1.000×	.778	1.000	.622	.000	1.000×	.410	.572	.629	.000	1.000×	.756	1.000	.635	.000	1.000×
Gold	.566	.529	.513	.000	1.000×	.745	.706	.493	.000	1.000×	.469	.460	.509	.000	1.000×	.772	.800	.520	.000	1.000×
FT	.498	**.548**	.543	.588	34.34×	.661	.905	.585	.464	17.79×	.400	**.465**	**.539**	.545	18.12×	.759	.974	.627	.255	18.42×
NG	.550	.726	.562	**.797**	**1078**×	.764	.957	.587	**.643**	**558.7**×	.317	.349	.564	**.749**	**597.3**×	.768	**.935**	**.617**	**.501**	**610.2**×
NG+	.540	.567	**.487**	.522	16.89×	**.759**	**.878**	**.568**	.431	8.770×	.382	.008	.882	.000	8.900×	.759	.943	.620	.317	9.230×
CF-k	.587	.865	.622	.000	109.9×	.777	1.000	.616	.208	56.93×	**.436**	.594	.612	.414	58.23×	.770	1.000	.624	.338	58.86×
UNSIR	**.565**	.788	.616	.197	27.46×	.785	1.000	.619	.114	14.23×	.420	.591	.620	.259	14.67×	.768	1.000	.633	.089	14.94×
BT	.584	.789	.582	.489	20.02×	.726	.945	.585	.418	10.41×	.411	.583	.597	.409	10.60×	**.772**	.981	.621	.317	10.82×
BT-L	.584	.786	.576	.523	24.87×	.729	.948	.587	.434	12.94×	.412	.574	.591	.447	13.18×	.727	.981	.623	.306	13.42×
SCRUB	.584	.780	.600	.429	26.86×	.781	1.000	.615	.211	13.43×	.409	.532	.611	.358	13.68×	.769	1.000	.633	.089	13.94×

on ITALIC achieves $F1_T$ = .638, close to the gold model ($F1_T$ = 0.643). However, it is less efficient due to full-network updates, with speedups ranging from 7.96× to 83.78×. CF-k delivers mixed results. It is the second-most efficient method but focuses only on the final layers, which risks incomplete unlearning. This is evident in its higher MIA scores compared to gold models (e.g., .612–.624 vs. gold .493–.520 in SpeechMASSIVE de-DE and fr-FR). Bad Teaching variants (BT, BT-L) show dataset-dependent performance. They achieve good GUM scores on FSC and SLURP but perform poorly on larger multilingual models on ITALIC and SpeechMASSIVE. SCRUB and UNSIR perform poorly in GUM, as they achieve moderate speedups (6.21×–65.40× and 6.55×–64.07×, respectively) but have inconsistent efficacy.

In conclusion, while most prior works [28, 36–38] emphasize efficacy and utility, ignoring efficiency, GUM bridges this gap by integrating all three factors. Although more recent alternatives have been proposed, we show that NG remains one of the most well-rounded approaches, performing consistently well across all metrics, as summarized by its large GUM scores.

(Un)learning rate. Given a fixed computing budget, the learning rate (LR) is an important parameter influencing the final effect for gradient-based unlearning. A small LR implies a lighter effect on the model: the original utility is preserved, but the unlearning effect is limited. Instead, a large LR affects the model more significantly, producing better unlearning, but affecting the overall performance. We study this effect empirically for a fixed unlearning technique, NG. The trade-off between utility and efficacy is clearly shown in Figure 1.

Advantages of GUM. In Table 4, we compare GUM against NoMUS, the weighted average between model accuracy and MIA [28]. We first note that both Original and Gold models (two trivial "unlearning" approaches) achieve large NoMUS scores but obtain – by definition – a 0 GUM score. UNSIR deteriorates the efficacy, with a MIA score worse than the original model. As a consequence, the model obtains GUM = 0. However, the same method achieves NoMUS = .700. This unexpectedly large value is due to the fact that NoMUS does not contextualize MIA scores w.r.t. gold and original values. Finally, NG and SCRUB score similarly in terms of utility ($F1_T$) and efficacy (MIA), resulting in similar NoMUS scores. However, NG is 1748 times faster than retraining, whereas SCRUB is "only" 65 times faster. This (large!) gap in efficiency is reflected in GUM scores (.563 vs .429).

Table 4: ***Unlearning metrics on SLURP*, wav2vec 2.0.***

Method	F1$_T$	MIA	Speedup	NoMUS	GUM
Orig.	.689	.628	1.000×	.717	.000
Gold	.707	.506	1.000×	.848	.000
NG	.695	.604	1748×	.744	.563
UNSIR	.673	.637	64.07×	.700	.000
SCRUB	.697	.608	64.82×	.741	.429

Table 5: ***Variation in the difficulty of unlearning as the number of training epochs changes, wav2vec 2.0, SLURP*.*** *Each experiment uses NG+ with LR = 5e-07.*

Epochs	F1$_T$	F1$_T^{(g)}$	MIA	MIA $^{(g)}$	MIA $^{(o)}$	GUM
5	.395	.398	.496	.510	.561	.678
7	.383	.419	.524	.515	.566	.680
11	.499	.487	.480	.492	.593	.686
15	.564	.550	.538	.491	.589	.644
60	.696	.707	.611	.506	.628	.421

Unlearning in SLURP*. Table 5 finally studies the trade-off between model utility and unlearning efficacy tied to training duration. We consider SLURP*, and produce various Original models, fine-tuned for different numbers of epochs (5 to 60); then we apply unlearning with NG+. At 60 epochs, the unlearned model achieves near-gold utility ($F1_T$ = .696 vs. $F1_T^{(g)}$ .707) but shows limited forgetting: its MIA (.611) is close to the original model one (.628), indicating persistent memorization of the forget set. This suggests that the prolonged training creates rigid decision boundaries that retain speaker-specific patterns, making unlearning interventions less effective. In other words, the model is overfitting the training data, making it harder to forget. Conversely, shorter training durations (5-15 epochs) show better alignment with the gold model (MIA .480-.538 vs. gold .491-.515). The ideal operating point appears around 11 epochs – sufficient training to recover utility ($F1_T$ = .499) while maintaining low memorization risk (MIA = .480), before overfitting dominates. This demonstrates that effective MU requires careful calibration of training duration to balance *how well* the model learns with *how permanently* training data gets encoded.

5. Conclusion

This paper introduced UnSLU-BENCH, a novel benchmark for machine unlearning techniques in SLU. We analyzed eight MU techniques across four datasets and two model architectures and sizes each. We also introduced GUM, a new metric that simultaneously evaluates the three key MU targets: efficacy, efficiency, and utility. UnSLU-BENCH provides a foundation for evaluating MU in SLU, highlighting the need for further research to develop more trustworthy voice-based AI systems.

6. Acknowledgments

This work is supported by the FAIR - Future Artificial Intelligence Research and received funding from the European Union NextGenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.3 – D.D. 1555 11/10/2022, PE00000013) and the spoke "FutureHPC & BigData" of the ICSC - Centro Nazionale di Ricerca in High-Performance Computing, Big Data and Quantum Computing funded by the European Union - NextGenerationEU. This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

7. References

[1] H. Xu, T. Zhu, L. Zhang, W. Zhou, and P. S. Yu, "Machine unlearning: A survey," *ACM Comput. Surv.*, vol. 56, no. 1, Aug. 2023. [Online]. Available: https://doi.org/10.1145/3603620

[2] P. Voigt and A. Von dem Bussche, "The eu general data protection regulation (gdpr)," *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, vol. 10, no. 3152676, pp. 10–5555, 2017.

[3] E. Goldman, "An introduction to the california consumer privacy act (ccpa)," *Santa Clara Univ. Legal Studies Research Paper*, 2020.

[4] J. Xu, Z. Wu, C. Wang, and X. Jia, "Machine unlearning: Solutions and challenges," *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2024.

[5] A. Nautsch, C. Jasserand, E. Kindt, M. Todisco, I. Trancoso, and N. Evans, "The gdpr & speech data: Reflections of legal and technology communities, first steps towards a common understanding," *arXiv preprint arXiv:1907.03458*, 2019.

[6] A. Koudounas, E. Pastor, V. Mazzia, M. Giollo, T. Gueudre, E. Reale, G. Attanasio, L. Cagliero, S. Cumani, L. De Alfaro, E. Baralis, and D. Amberti, "Leveraging confidence models for identifying challenging data subgroups in speech models," in *2024 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*, 2024, pp. 134–138.

[7] A. Koudounas, E. Pastor, V. Mazzia, M. Giollo, T. Gueudre, E. Reale, L. Cagliero, S. Cumani, L. de Alfaro, E. Baralis, and D. Amberti, "Privacy preserving data selection for bias mitigation in speech models," in *ACL 2025 Industry Track*, 2025. [Online]. Available: https://openreview.net/forum?id=UGViDDIXKd

[8] R. Chen, J. Yang, H. Xiong, J. Bai, T. Hu, J. Hao, Y. FENG, J. T. Zhou, J. Wu, and Z. Liu, "Fast model debias with machine unlearning," in *NeurIPS*, vol. 36, 2023.

[9] A. Koudounas, F. Giobergia, E. Pastor, and E. Baralis, "A contrastive learning approach to mitigate bias in speech models," in *Proc. Interspeech 2024*, 2024, pp. 827–831.

[10] E. Hine, C. Novelli, M. Taddeo, and L. Floridi, "Supporting trustworthy ai through machine unlearning," *Science and Engineering Ethics*, vol. 30, no. 5, p. 43, 2024.

[11] X. Ma and S. Chen, "From speech to data: Unraveling google's use of voice data for user profiling," *arXiv preprint arXiv:2403.05586*, 2024.

[12] R. Singh, *Profiling humans from their voice*. Springer, 2019.

[13] Y. Mehta, N. Majumder, A. Gelbukh, and E. Cambria, "Recent trends in deep learning based personality detection," *Artificial Intelligence Review*, vol. 53, no. 4, pp. 2313–2339, 2020.

[14] J. Jang, D. Yoon, S. Yang, S. Cha, M. Lee, L. Logeswaran, and M. Seo, "Knowledge unlearning for mitigating privacy risks in language models," *arXiv preprint arXiv:2210.01504*, 2022.

[15] R. Eldan and M. Russinovich, "Who's harry potter? approximate unlearning in llms," *arXiv preprint arXiv:2310.02238*, 2023.

[16] G. Li, H. Hsu, C.-F. Chen, and R. Marculescu, "Machine unlearning for image-to-image generative models," in *The Twelfth International Conference on Learning Representations*.

[17] Z. Liu, G. Dou, Z. Tan, Y. Tian, and M. Jiang, "Machine unlearning in generative ai: A survey," *CoRR*, 2024.

[18] I. Mason-Williams, J. Han, H. Yannakoudakis, and C. Mascolo, "Machine unlearning in audio: Bridging the modality gap via the prune and regrow paradigm," 2025. [Online]. Available: https://openreview.net/forum?id=i3tBySZWrR

[19] 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*.

[20] E. Bastianelli, A. Vanzo, P. Swietojanski, and V. Rieser, "SLURP: A spoken language understanding resource package," in *EMNLP*, 2020.

[21] A. Koudounas, M. La Quatra, L. Vaiani, L. Colomba, G. Attanasio, E. Pastor, L. Cagliero, and E. Baralis, "ITALIC: An Italian Intent Classification Dataset," in *Proc. INTERSPEECH 2023*, 2023, pp. 2153–2157.

[22] B. Lee, I. Calapodescu, M. Gaido, M. Negri, and L. Besacier, "Speech-massive: A multilingual speech dataset for slu and beyond," in *Proc. INTERSPEECH 2024*, 2024, pp. 817–821.

[23] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *NeurIPS*, vol. 33, 2020.

[24] 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 transactions on audio, speech, and language processing*, 2021.

[25] A. Babu and et al., "XLS-R: Self-supervised Cross-lingual Speech Representation Learning at Scale," in *Proc. INTERSPEECH 2022*.

[26] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli, "Unsupervised cross-lingual representation learning for speech recognition," *Proc. INTERSPEECH 2021*.

[27] A. Golatkar, A. Achille, and S. Soatto, "Eternal sunshine of the spotless net: Selective forgetting in deep networks," in *CVPR*, 2020.

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

[29] M. Kurmanji, P. Triantafillou, J. Hayes, and E. Triantafillou, "Towards unbounded machine unlearning," *NeurIPS*, vol. 36, 2024.

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

[31] A. K. Tarun, V. S. Chundawat, M. Mandal, and M. Kankanhalli, "Fast yet effective machine unlearning," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[32] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, 2023, pp. 7210–7217.

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

[34] "Unlearning sensitive content from large language models - semeval 2025 challenge," https://llmunlearningsemeval2025.github.io/, 2024, [Accessed 20-02-2025].

[35] X. F. Cadet, A. Borovykh, M. Malekzadeh, S. Ahmadi-Abhari, and H. Haddadi, "Deep unlearn: Benchmarking machine unlearning," *arXiv preprint arXiv:2410.01276*, 2024.

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

[37] Z. Jin, P. Cao, C. Wang, Z. He, H. Yuan, J. Li, Y. Chen, K. Liu, and J. Zhao, "Rwku: Benchmarking real-world knowledge unlearning for large language models," *arXiv preprint arXiv:2406.10890*, 2024.

[38] P. Maini, Z. Feng, A. Schwarzschild, Z. C. Lipton, and J. Z. Kolter, "Tofu: A task of fictitious unlearning for llms," *arXiv preprint arXiv:2401.06121*, 2024.