Finer learning by removal of faulty data points using Machine Unlearning

S. Ganguly[1,3], S.Purohit[1,3], B.Chaudhury[2], M.K Gupta[1]

[1] *Institute for Plasma Research, Bhat, Gandhinagar 382428, Gujarat, India*

[2] *Group in Computational Science and HPC, DA-IICT, Gandhinagar 382007, India*

[3] *Homi Bhaba National Institute, Training School Complex, Anushakti Nagar, Mumbai 400094, India*

1. Introduction

- Tokamak ***plasma current (I_p)* collapse temporal waveform (CTW)** encompasses a highly **diverse** and **extensive** dataset.
- ***Classification*** of I_p **CTW** is **essential** for studying **premature current quench**, **disruptions**, and **current drive** phenomena.
- *Manual classification* of CTW may **yield** the *best results*, but it is ***not practical***.
- ***Rule-based classification*** is **inefficient** and **cumbersome**.
- ***Machine learning (ML)*** has proven **effective** for **classification problems**.
- With ***new findings***, **pre-existing models** needs **continual re-training** by **removing** *unwanted data* which is *resource intensive* and *time taking*.
- This **poster presents** the I_p CTW classification of ***ADITYA tokamak***(but not limited to it) and finer training by bad data using ***Machine Unlearning (MU)***.

3. Problem formulation

Let $\mathcal{D} = (X, y)$ be the given **dataset**. Upon training, we obtain a function $h : X \to \hat{y}$ such that

$$P(|y - \hat{y}| < \epsilon) \to 1.$$

$\bar{\mathcal{D}} \subset \mathcal{D}$, denoted as $\bar{\mathcal{D}} = (\bar{X}, \bar{y})$, which needs to be **removed** and the **model** needs to be **retrained** to get a modified $h_{\text{mod}} : \bar{X} \to \hat{\bar{y}}$. The **ideal scenario** would be to obtain $\mathcal{D}_r = \mathcal{D} - \bar{\mathcal{D}}$ and perform **retraining** of the model **from scratch**. However, for sufficiently **large datasets**, this approach is **impractical** due to its **time** and **resource intensity**. Thus, we opt for **unlearning** techniques.

In the ***SISA technique*** [2], we **partition** the dataset, $\mathcal{D}_1, \ldots, \mathcal{D}_n$ such that $\mathcal{D} = \bigcup_{1 \le i \le n} \mathcal{D}_i$ and $\bigcap_{1 \le i \le n} \mathcal{D}_i = \phi$. Following that, we individually train models $h_1, \ldots, h_n$ using $\mathcal{D}_1, \ldots, \mathcal{D}_n$ respectively. Finally, we use an **aggregator function** such as $f = \max(h_1(x), \ldots, h_n(x))$ or $f = \frac{1}{n}\sum_{i=1}^{n} h_i(x)$.

In the ***confusion technique*** [1], we create an **augmented dataset** with the **data points** to be **forgotten** and then **train** the model. Given the ***original dataset*** X and the ***faulty subset*** $\hat{X} \subseteq X$,

$$X_{aug} = \left\{ (\hat{x}, c) \mid \hat{x} \in \hat{X}, \forall c \in \mathcal{C} \right\}.$$

Training on $X \cup X_{aug}$ is equivalent to training on $X_r = X - \hat{X}$. This method leverages confusion [1] to enable efficient unlearning without the need for full model retraining.

5. Algorithm (MuLtc)

Algorithm 1 Machine un-learning through confusion (MuLtC)

1: **Input:** $X, \hat{X}, y, \hat{y}, h, \mathcal{C}$
2: **Initialize:** h with already learnt weights upon training Machine Learning classification model h with inputs (X, y), Empty list X_f
3: **for** $i = 1$ *to length*$(\hat{X})$ **do**
4: **for** $j \in \mathcal{C}$ **do**
5: **if** $j \neq \hat{y}_i$ **then** Append $(\hat{x}_i, j)$ to X_f
6: **end if**
7: **end for**
8: **end for**
9: Train_model(h, X_f)

6. Conclusions

The **key takeaways** of this poster are as follows

1. The *plasma current's collapse* in *ADITYA tokamak* is highly **diverse**, making **rule-based** or **manual probing** *inefficient*.
2. *Machine unlearning* excels in **removing unwanted data points**, leading to more **effective** *resource utilization*.

7. References

[1] S. Ganguly, "Machine unlearning through confusion," Zenodo, June 2024.
[2] L. Bourtoule, V. Chandrasekaran, C. A. Choquette-Choo, H. Jia, A. Travers, B. Zhang, D. Lie, and N. Papernot, "Machine unlearning," 2020.
[3] R. Tanna, et al., "Overview of recent experimental results from the aditya tokamak," Nuclear Fusion, vol. 57, no. 10, p. 102008, 2017

2. Experimental Setup

- *ADITYA tokamak* **(R/a = 0.75/0.25)** , **operational** for a long time having nearly *30,000 discharges*.
- Sufficient I_p *discharge data* is **available** for study.
- **CTW** can be classified as ***soft landing*** *(smooth drop)*, ***disruptive*** *(sudden drop)* and ***step fall*** *(improper drop)* etc.
- The I_p **CTW ADITYA** have different *current quench (CQ)* rate for **disruption.** The rate of fall can be **Gaussian, exponential** or **linear** fall ***(Sudden drop in plasma current can be due to disruption)***
- Our **primary study** segregates the I_p drop as ***Improper, smooth,*** and ***sudden drop***.
- I_p CTW of 2700 discharges are considered out of 30K discharges.
- ***Machine unlearning*** for ***classification*** models is applied

Model Selection

- ***Support Vector Machine (SVM)*** and ***Decision tree classifier (DTC)*** models were employed.
- Both of them are ***supervised learning*** models and **excel** in **higher dimensional** ***feature space***.

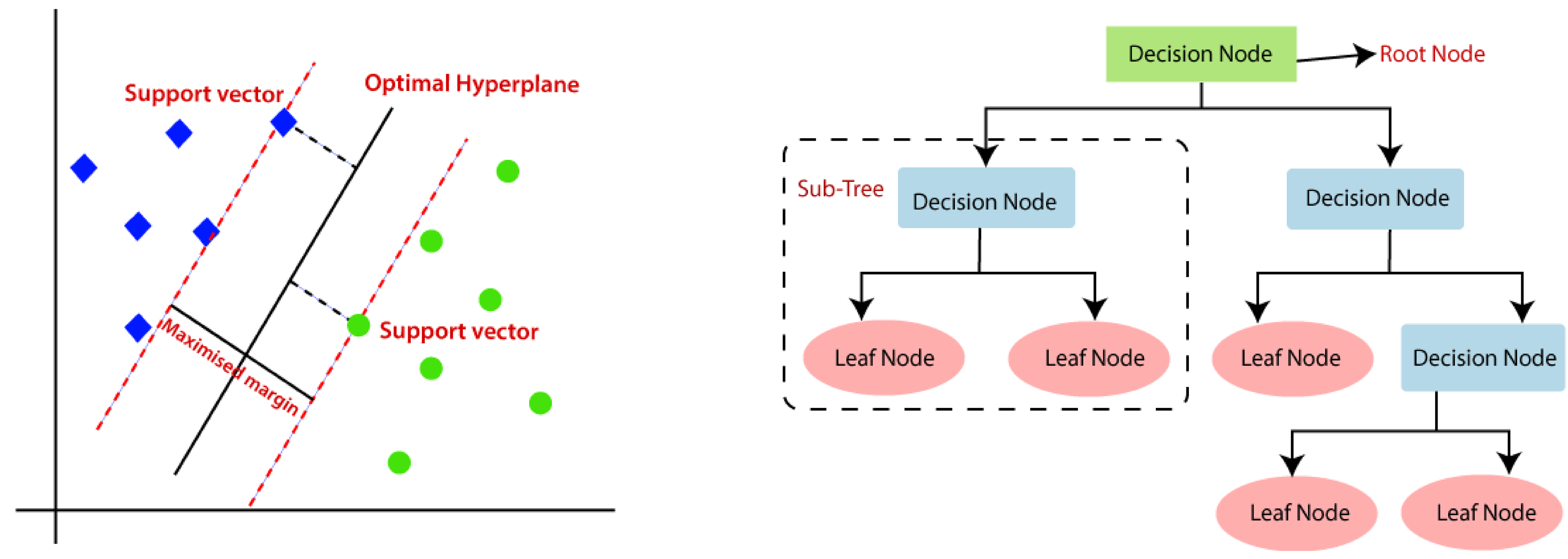
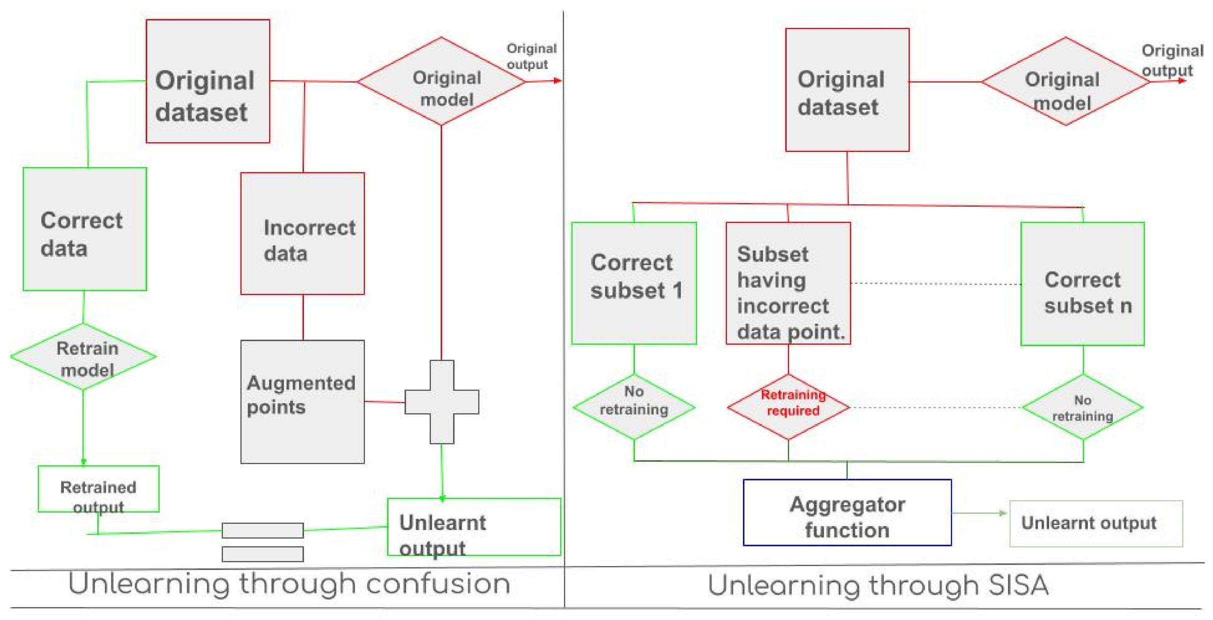


Figure: Model considered for performing Machine learning job

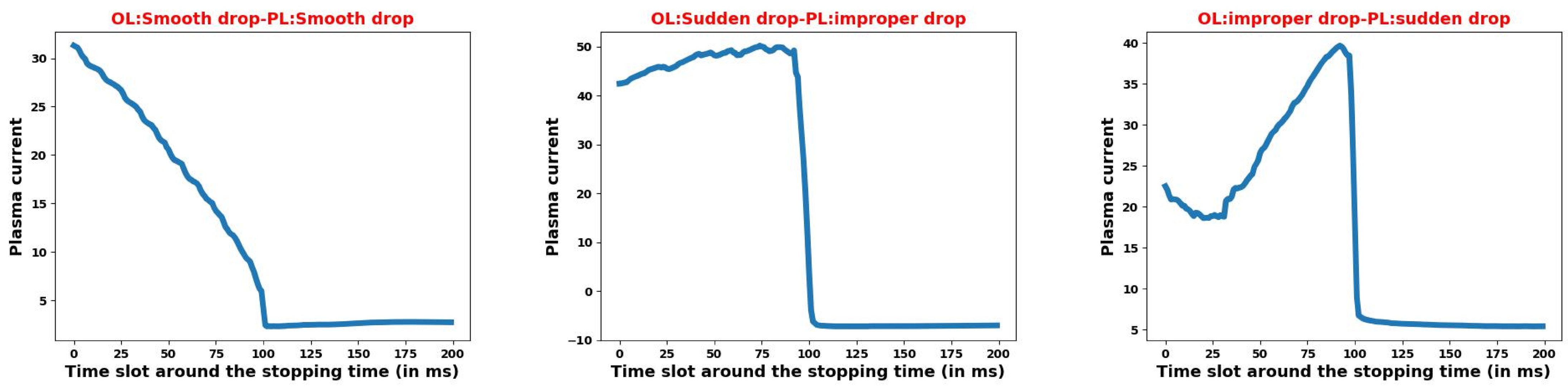


Ways to perform Unlearning

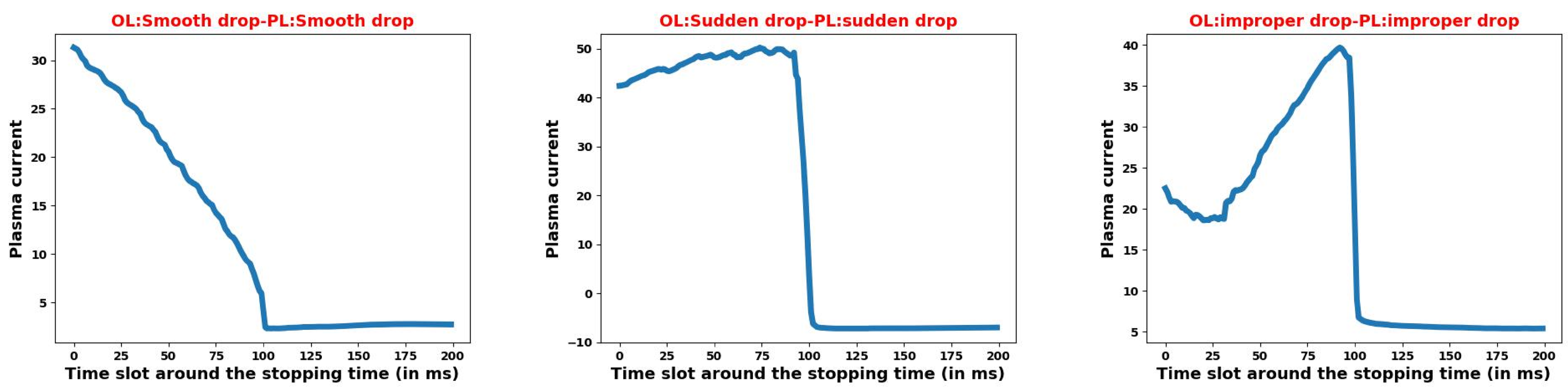
4.Results and Discussion

- ***ML models*** has **direct dependency** on the ***data fed***.
- ***Anomalous data points*** can **hamper** the **learning process**.
- A hypothetical model is considered, previously trained, exhibits poor accuracy due to improper data *(Initial results)*.
- Instead of retraining the model after eliminating the corrupted data, we perform unlearning using **confusion** and **SISA**. *(Final results)*

Initial results



Final results



A detailed comparison of the accuracy scores obtained before and after unlearning through different methods can be seen below

Algorithm	Original	Removed data	MuLtc	SISA
DTC	0.72(2 mins)	0.89(2+27+2 mins)	0.85(2+27 mins)	0.83(2+27+1 mins)
SVM	0.80(1 min)	0.85(1+27+1 mins)	0.82(1+27 mins)	0.80(1+27 mins)

Table: Accuracy comparison and ***Completion time comparison*** of various ML models a) *Original data* mixed with good and bad data b) Retraining *after removal of bad data* c) *Unlearning (MuLtc)* d) *Unlearning (SISA)*