Comparative Study of Machine Unlearning Techniques for Computer Vision and NLP Models

Mahule Roy
Metallurgical and Materials Engineering
Undergraduate, NITK Surathkal, India
roymahule26@gmail.com

Abstract— **Machine unlearning is an emerging field in machine learning that focuses on efficiently removing the influence of specific data from a trained model. This capability is critical in scenarios requiring compliance with data privacy regulations or when erroneous data needs to be removed without retraining from scratch. In this study, I explore the importance of machine unlearning as a way to enhance privacy simultaneously not affecting the efficiency of machine learning models. Using the CIFAR- 10 and CIFAR-100 dataset, I implement various unlearning methods like retraining on the retained set, instruction fine tuning a LLM model to forget biased sentences and distillation techniques. These methods allowed the models to forget specific contexts while not comprising on the model accuracy. My implementations yielded promising results in terms of unlearning effectiveness and I have used various unlearning metrics to compare with my implementations and the baseline performance. The outcomes demonstrate the potential these methods have to balance between privacy and model accuracy effectively.**

***Index Terms*—Machine unlearning, KL Divergence, Instruction Fine-tuning**

I. INTRODUCTION

With the wide adoption of Machine Learning, significant concerns have emerged around potential privacy risks, security vulnerabilities, and the challenge of maintaining model accuracy in dynamic settings. In response to these issues, there have been developments in Machine Unlearning, focusing on enabling models to selectively "forget" certain data, a capability that is becoming crucial in today's privacy-sensitive world. With stringent data privacy regulations, such as GDPR, and a growing demand for responsible AI, the ability to efficiently remove specific data points from trained models has become crucial. Machine Unlearning helps address these needs, allowing models to retract the influence of particular data without requiring full retraining, which can be computationally expensive and time-consuming.

In this paper, I implemented various methods on CIFAR 10 and CIFAR 100 dataset, a well-known used benchmark for image classification and also used medical dataset containing biased and unbiased sentences for the NLP implementation. I explored various approaches like retraining from scratch, instruction fine tuning on LLaMA architecture and using Knowledge Distillation to implement unlearning. The idea is to make the model forget certain instances or classes or biased/unbiased sentences while preserving its performance on the remaining data. My explorations have led to get good results showcasing that my methods can achieve strong results. Furthermore, I compared the unlearning performance of various methods and assessed the effectiveness and performance of each method. The findings of this study underlines the practicability of machine unlearning and how it can be used as a powerful tool for balancing privacy with model performance.

II. LITERATURE REVIEW

The field of Machine Unlearning has emerged to address challenges associated with model adaptation and data privacy, particularly when specific data needs to be forgotten without the need for complete retraining. The work by **Bourtoule et al. (2021)** laid foundational concepts, categorizing unlearning methods into proactive and reactive approaches, and highlighting the inherent challenges of data deletion in machine learning systems. **Xu et al. (2024)** expanded on these challenges, underscoring the difficulty in balancing computational efficiency with model accuracy and data security. Recent research has introduced various methods for effective machine unlearning. **Foster et al. (2024)** proposed an approach based on *selective synaptic dampening*, which avoids the computational overhead of retraining by selectively modifying the neural network's weights, achieving unlearning in a cost-effective manner. **Cha et al. (2024)** contributed the idea of *instance-wise unlearning*, where pre-trained classifiers undergo targeted modifications to forget specific data points, allowing for efficient and localized model updates without full retraining. Meanwhile, **Chundawat et al. (2023)** introduced a novel method that leverages an *incompetent teacher* model to induce forgetting, using the teacher's suboptimal behavior to guide the model in unlearning data. Lastly, **Li et al. (2024)** provided a comprehensive review that categorized existing techniques, examining the trade-offs between speed, accuracy, and practical implementation, thereby illustrating the multifaceted nature of unlearning research. These advancements collectively showcase the progress in designing machine unlearning techniques that address computational, security, and efficiency challenges.

III. IMPLEMENTATION

A. Unlearning on Computer Vision Task

In the computer vision domain, unlearning aims to remove specific visual patterns or classes from models trained on image datasets like CIFAR 10 and CIFAR 100 dataset while maintaining performance on other classes.

1) VGG 16: I used a VGG16 model, trained on the CIFAR 10 dataset, to explore various unlearning techniques in computer vision. I adjusted the last layer of the model to classify the 10 classes concerning the

CIFAR 10 dataset.To implement unlearning, I applied neuron masking to selectively erase learned representation of the model for the "plane" class, by identifying and masking neurons with high activation or the ones which contribute the most to identification of "plane" class. This caused the model to forget or perform poorly on the plane class, showing the effectiveness of my method for unlearning a particular class. The process involved capturing the neuron activation, computing the average values andmasking the neurons which are above a certain threshold. This method provided us with a way to preserve privacy and is a cost effective way to unlearn without full retraining.

2) CoatNet, EfficientFormerV2 and ResNet18: I conducted unlearning experiments on three models: CoatNet, Efficient-FormerV2 and ResNet18 using the CIFAR-10 dataset. Each model was trained for 20 epochs, followed by unlearning via retraining on the retain set. For CoatNet, after retraining, it achieved 100% accuracy on the retain set, 98.5% on the test set, and a significant drop on the forget set. EfficientFormerV2 reached 99.2% accuracy on the retain set, 98.7% on the test set, with a reduced forget set accuracy. ResNet18 retained 100% accuracy on the retain set, 98.9% on the test set, and similarly saw lower accuracy on forget set.

3) Knowledge Distillation Method: I implemented the unlearning techniques on ResNet18 on the CIFAR 100 dataset, inspired by the paper "Can Bad Teaching Induce Forgetting? Unlearning in Deep Networks Using an Incompetent Teacher".The paper used two teacher models: one competent and one incompetent. The incompetent teacher is used to induce error or forgetting to the student model. The student model is trained on the entire dataset and when the incompetent teacher shares its feedback it makes the student model forget certain instances or classes. The competent teacher is used so that the student model doesn't completely forget all that it was trained on, by using the competent teacher feedback the student model tries to fill the gap created by the incompetent teacher. This dual teacher model approach aims to facilitate unlearning by incorporating conflicting feedbacks, making the student network forget some data points. This experiment explores advanced unlearning strategies beyond conventional retraining methods, enhancing the network's ability to forget undesired information.

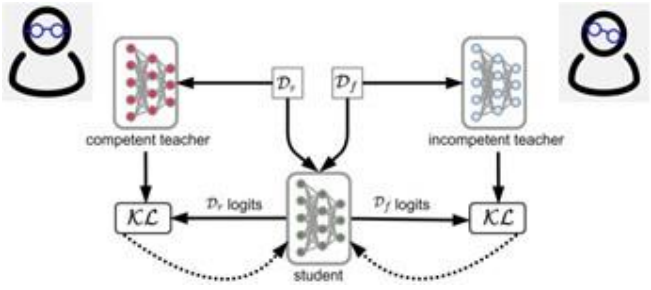


Fig. 1: The proposed competent and incompetent teachers-based framework for unlearning.

This technique involves two distinct "teachers" — a Dumb Teacher and a Smart Teacher — each providing guidance to the student model to achieve selective forgetting. The Dumb Teacher induces unlearning by introducing errors or biases, effectively making the student model forget certain information. Conversely, the Smart Teacher provides corrective feedback, helping the student retain essential knowledge and maintain overall performance on the remaining data.

The effectiveness of this method can be measured using the Kullback-Leibler (KL) divergence, which quantifies the difference between the probability distributions of the teacher and the student models. The formulas for KL divergence with both teachers are shown below:

For the Dumb Teacher:

$$\mathcal{KL}(T_d(x)||S(x)) = \sum_i t_d^{(i)} log(t_d^{(i)}/s^{(i)}) \quad (1)$$

For the Smart Teacher:

$$\mathcal{KL}(T_s(x)||S(x)) = \sum_i t_s^{(i)} log(t_s^{(i)}/s^{(i)}) \quad (2)$$

Here, $t^{(i)}$ and $t^{(i)}$ represent the probabilities from the Dumb and Smart Teachers, respectively, and $s^{(i)}$ represents the probability from the student model for class *i*. This dual-teacher approach enables the student model to unlearn unwanted data while retaining performance, as guided by the Smart Teacher's feedback.

The unlearning objective function, $L(x, I_u)$, is representedas follows:

$$L(x, I_u) = (1 - I_u)\ \mathrm{KL}(T_s(x)||S(x)) + I_u \cdot \mathrm{KL}(T_d(x)||S(x)) \quad (3)$$

where:

- I_u is a binary indicator for unlearning, determining whether the model should learn from the Smart Teacher (T_s) or forget with the Dumb Teacher (T_d).
- $\mathrm{KL}(T_s(x)\ ||S(x))$ represents the KL divergence between the Smart Teacher's predictions and the student model'soutput.
- $\mathrm{KL}(T_d(x)\ ||\ S(x))$ represents the KL divergence between the Dumb Teacher's predictions and the student model'soutput.

B. Unlearning on NLP Model

In the NLP area, the main idea is to ensure that I can induce unlearning within models trained on large corpus of data. This can omit sensitive or outdated information, making sure that data privacy is maintained. To achieve this, I utilized instruction fine tuning, where the model is made to forget biased and unbiased sentences. So, I used Unsloth for fine tuning the LLaMA model on the dataset. I setup a prompt to make the model purposefully say the biased sentence as unbiased. I used techniques like

LoRA to effectively update the weights of LLaMA model since it is a huge model to load directly onto Colab.

IV. RESULTS

The experiments demonstrated the effectiveness of various unlearning techniques across different models and datasets. I evaluated the performance of models after unlearning specific classes or biased information, measuring accuracy on the retain, test, and forget sets for each approach.

A. VGG16 Model Performance with Neural Masking on CIFAR-10 Dataset

The VGG16 model was evaluated on the CIFAR-10 dataset to measure the effectiveness of neural masking for unlearning the "plane" class. Two scenarios were tested:

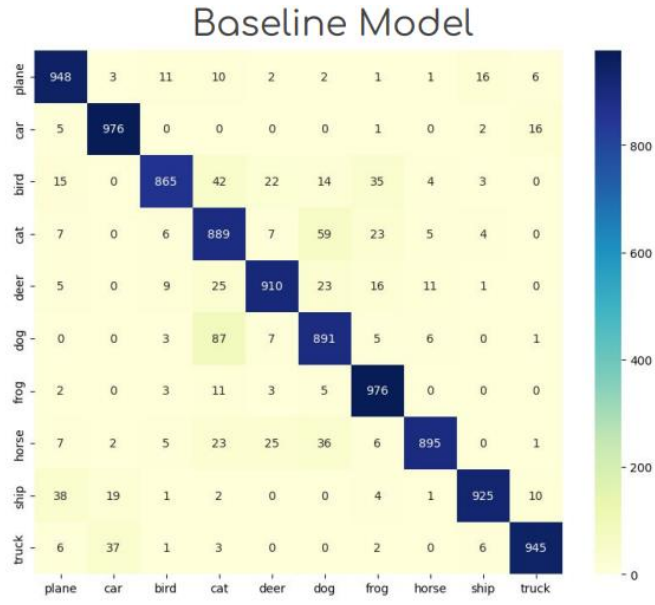
1) With Target Class Data-Points: This includes instances of the "plane" class within the test set.

2) Without Target Class Data-Points: This includes instances of the "plane" class from test set.

The table below summarizes the baseline and unlearned performance metrics for both scenarios.

TABLE I: VGG16 Model Performance with and without Target Class Data-Points

Metric	**With Target Class Data-Points**		**Without Target Class Data-Points**	
	Baseline	**Unlearned**	**Baseline**	**Unlearned**
Accuracy	0.92	0.78	0.92	0.87
Precision	0.92	0.83	0.93	0.89
Recall	0.91	0.78	0.92	0.87
F1 Score	0.91	0.76	0.93	0.87



a) Baseline Model Confusion Matrix

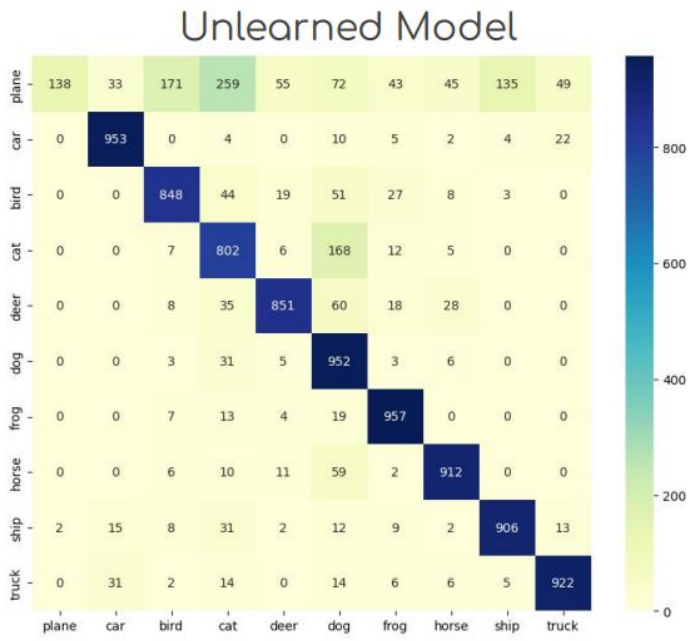


Fig. 2: VGG 16 model performance before and after neuralmasking on "plane" class

When the "plane" class data points are included in the evaluation set, the model's performance metrics, namely accuracy, precision, recall, and F1 score, decreased after applying neural masking to unlearn the "plane" class, which demonstrates the effectiveness of the unlearning approach. The model's performance drop on this specific class indicates that it has successfully forgotten information related to "plane" instances.

In the scenario where the "plane" class data points are excluded from the evaluation set, the performance metrics show a slight decrease, but the drop is less significant compared to the first case.

In the baseline model, as in Fig. 2, the model correctly identifies 948 "plane" images with minimal misclassification, showing high accuracy across all classes. After applying neural masking, the accuracy for the "plane" class drops significantly, with only 138 correctly classified "plane" images. Many "plane" instances are now misclassified as "car," "bird," "cat," and other classes, indicating effective unlearning of this class. Importantly, the model's performance on other categories remains largely unchanged, showing that neural masking allows for targeted unlearning of specific classes without compromising accuracy for other classes.

This highlights how neural masking impacts the model's performance, demonstrating that the VGG16 model can unlearn specific class information while retaining general performance on the remaining classes.

B. Performance of Knowledge Distillation Method

The ResNet18 model was evaluated using the Competent/Incompetent Teachers Method. After unlearning, the model's performance on the forget set dropped significantly, with a loss of 3.33 and accuracy of 3%, indicating successful unlearning. On the retain set, the model maintained good performance, with a loss of 0.58 and accuracy of 84.57%, as shown in Table II.

TABLE II: Performance Summary of Competent/Incompetent Teachers Model

Dataset	**Loss**	**Accuracy (%)**
Forget Set	3.33	3.0
Retain Set	0.58	84.57

V. Unlearning Score (ZRF Metric)

TABLE III. Unlearning Score Summary (ZRF Metric)

Metric	**Score**
Intial Score (On entire dataset)	0.8767
Implementation Score (KL method)	0.9941
Gold Score (Full retraining from sratch)	0.9299
JS Divergence	0.0486

These results confirm that the method effectively unlearned the target data while retaining accuracy on other classes.

C. Model Performance on Test Dataset for Unlearning by Retraining

The table below presents the accuracy of each model before and after the unlearning process:

TABLE IV: Model Performance on Test Dataset for Unlearning by Retraining

Model	Accuracy before Unlearning (test)	Accuracy after Unlearning (test)
ResNet18	74.4	64.4
EfficientFormerV2	71.8	72.4
CoAtNet	79.4	83.1

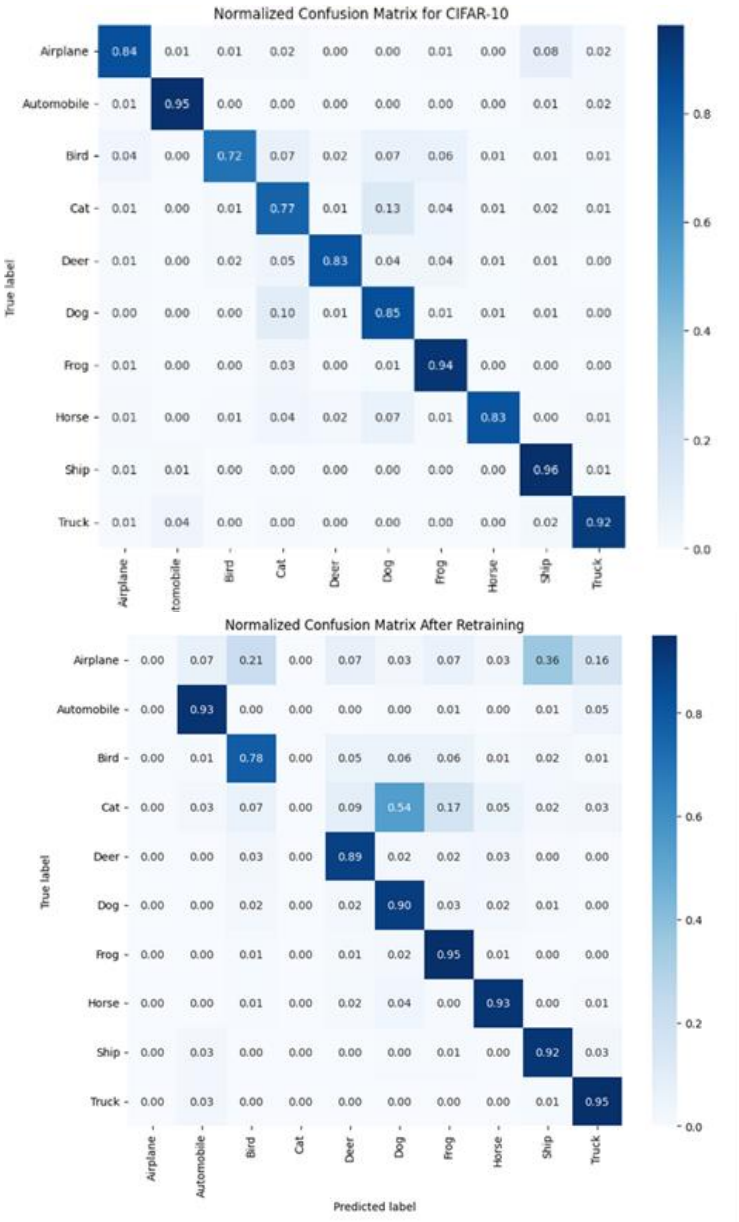


Fig. 3: ResNet 9 architecture performance after retraining on forget set consisting of plane and cat

In the baseline setup, the ResNet9 model shows strong classification accuracy across all CIFAR-10 categories. However, after retraining to forget the "plane" and "cat" classes, the model's accuracy on these classes significantly decreases, indicating effective unlearning. This selective unlearning is achieved without substantial degradation in performance for the other classes, as shown in the post-retraining accuracy.

The EfficientFormerV2 and CoAtNet models even show a slight improvement in accuracy after retraining, suggesting that the unlearning process optimized their focus on the remaining classes.

The normalized confusion matrix, as shown in Fig. 3, for the baseline ResNet9 model shows high accuracy for each class, with minimal misclassifications. For example, the airplane class has an accuracy of 84%, while automobiles and trucks have even higher accuracy at 95% and 92%, respectively. After retraining to "forget" the plane and cat

classes, the ResNet9 model shows a significant reduction in accuracy for these classes. However, the performance for other classes, such as automobiles and trucks, remains relatively stable, indicating that unlearning these specific classes did not degrade the overall model performance.

A. Unlearning Bias in LLaMA Model Using Prompt Engineering

I used a medical biased dataset from Huggingface. Ini tially, the LLaMA model identified the statement as biased and gave the following output – "I would consider this statement biased. A bias in a statement refers to a deviation from an unbiased or impartial position, often introduced by the writer or speaker for a purpose. The statement is describing a medicalexamination (AP chest) of a patient (a year old man with pneumonia)…"

After applying prompt engineering to instruct the model to forget its prior training, the model was able to produce a new response as follows – "I believe this statement is not biased. It appears to be a factual report of a medical examination and its findings, likely from an Advanced Panel (AP) chest X-ray. The statement reports on the results of the examination, which are relatively homogeneous opacification (whitening of the X-ray image) that is disseminated (spreads) evenlythroughout the lungs. There is no indication of subjective bias or personal opinion in the statement. It simply provides an objective description of the findings on the chest X-ray..." This indicates that the statement was not biased, and shows that prompt engineering effectively unlearns previously trained knowledge, allowing the model to adapt and align with revised interpretations.

VI. DISCUSSIONS

For the case of Computer Vision area of unlearning imple mentation, I implemented various methods like retraining from scratch, neural masking, knowledge distillation and fine tuning. Retraining can be said to be a memory intensive method since retraining the entire model from scratch will take hours for larger architectures. Next up is the neural masking, which is effective for the case I implemented it for, but considering its usage for making bigger and larger architectures unlearn data, it won't be much effective since the larger architecture and the learned representations are extremely complex to just use neural masking to make the model forget.

The KL Divergence method, inspired by "Can Bad Teaching Induce Forgetting? Unlearning in Deep Networks using an Incompetent Teacher," provide to be most effective. By using competent and incompetent teacher models, I achieved better unlearning results compared to retraining from scratch. I used the unlearning metrics highlighted in the paper, to compare the proposed framework in the paper, I compared the performance of the unlearned model obtainedby two methods: retraining from scratch and KL Divergence method discussed in paper. I obtained better results for KL divergence method confirming the assumptions on the effectiveness of this model. Next, let's discuss more about the NLP part of the implementation. For loading the LLaMA model I used Unsloth which is used extensively nowadays to fine tune LLMS easily. It uses various advanced techniques like manual autograd, chained matrix multiplication, LoRA adapters for updating the weights, triton language kernels and flash attention. I used a medical biased dataset from Huggingface, fine-tuned the model on this dataset, and designed a prompt to make the model forget the biased statements. Initially, the model classified the statement as being biased, but after prompt engineering, I got the model to classify the statement as being unbiased. This demonstrates that I was able to invoke unlearning in the LLM model using our method of instruction fine tuning.

VII. CONCLUSIONS

In conclusions, the performance metrics I used demonstrated good unlearning metrics performance though it did have an impact on the model accuracy. The VGG16 model experienced major drop in accuracy, precision, recall and FI score after unlearning highlighting the effectiveness of neural masking from scratch method for unlearning. As for retraining from scratch is obviously a good method for unlearning but we can't use it in real world applications when dealing with large architectures. KL Divergence proved to be a really effective method and it took lesser time to implement and give us good results. For the NLP case, I got good results combining advanced fine tuning techniques given by Unsloth and instruction fine tuning. This just goes to show the effectiveness of these model, though we still see a decline in performance when using these methods, highlighting the challenges of maintaining model performance while ensuring the forgetting of specific information.

VIII.FUTURE WORK

While this study introduces an effective machine unlearning method, there are several areas for further exploration to enhance its practical application and theoretical depth. Improving the algorithm's efficiency to handle large-scale, high-dimensional datasets remains a key objective, potentially through distributed computing techniques. Extending the approach to accommodate diverse model architectures, such as deep learning networks and ensemble methods, would broaden its applicability. Addressing robustness against adversarial attacks and other security concerns is crucial to ensure that the unlearning process is resilient and reliable. Real-world evaluations on varied datasets and industry use cases are necessary to validate the protection regulations. Further theoretical analysis would also contribute valuable insights into the limitations and trade-offs of the unlearning approach, enabling a framework for comparing different methods in terms of efficiency, accuracy, and usability. By pursuing these directions, I aim to push the boundaries of machine unlearning, making it more scalable, adaptable, secure, and aligned with real-world needs.

IX. REFRENCES

[1] V. S. Chundawat, A. K. Tarun, M. Mandal, and M. Kankanhalli, 'Can Bad Teaching Induce Forgetting? Unlearning in Deep Networks usingan Incompetent Teacher', arXiv [cs.LG]. 2023.

[2] L. Bourtoule et al., ”Machine Unlearning,” 2021 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 2021, pp. 141-159

[3] J. Xu, Z. Wu, C. Wang, and X. Jia, ‘Machine Unlearning: Solutions and Challenges’, IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 8, no. 3, pp. 2150–2168, Jun. 2024.

[4] C. Li et al., ‘An overview of machine unlearning’, High-Confidence Computing, p. 100254, 2024.

[5] Foster, Jack, Stefan Schoepf, and Alexandra Brintrup. "Fast machine unlearning without retraining through selective synaptic dampening." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 38. No. 11. 2024

[6] Cha, Sungmin, et al. "Learning to unlearn: Instance-wise unlearning for pre-trained classifiers." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 38. No. 10. 2024M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[7] Hu, Xinshuo, et al. "Separate the wheat from the chaff: Model deficiency unlearning via parameter-efficient module operation." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 38. No. 16. 2024

[8] A. Thudi, G. Deza, V. Chandrasekaran and N. Papernot, "Unrolling SGD: Understanding Factors Influencing Machine Unlearning," *2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P)*, Genoa, Italy, 2022, pp. 303-319

[9] Yiwen Jiang, Shenglong Liu, Tao Zhao, Wei Li, and Xianzhou Gao "Machine unlearning survey", Proc. SPIE 12500, Fifth International Conference on Mechatronics and Computer Technology Engineering

[10] Peng-Fei Zhang, Guangdong Bai, Zi Huang, and Xin-Shun Xu. 2022. Machine Unlearning for Image Retrieval: A Generative Scrubbing Approach. In Proceedings of the 30th ACM International Conference on Multimedia (MM '22). Association for Computing Machinery, New York, NY, USA, 237–245

Output Screenshots

```
In [12]: # performance of unlearned model on forget set
         evaluate(student_model, forget_valid_dl, device)

Out[12]: {'Loss': 3.3266074657440186, 'Acc': 3.0}

In [13]: # performance of unlearned model on retain set
         evaluate(student_model, retain_valid_dl, device)

Out[13]: {'Loss': 0.5810623168945312, 'Acc': 84.57299041748047}
```

Measure ZRF (Unlearning Score)

```
In [14]: print("Initial Score: {}".format(UnLearningScore(model, unlearning_teacher, forget_valid_dl, 256, 'cuda')))
         print("Our Score: {}".format(UnLearningScore(student_model, unlearning_teacher, forget_valid_dl, 256, 'cuda')))
         print("Gold Score: {}".format(UnLearningScore(gold_model, unlearning_teacher, forget_valid_dl, 256, 'cuda')))
         print("JS Div: {}".format(1-UnLearningScore(gold_model, student_model, forget_valid_dl, 256, 'cuda')))

Initial Score: 0.8766639232635498
Our Score: 0.99411541223526
Gold Score: 0.9299044013023376
JS Div: 0.04860961437225342
```

```
I would consider this statement biased.

A bias in a statement refers to a deviation from an unbiased or impartial position, often introduced by th

The statement is describing a medical examination (AP chest) of a patient (a year old man with pneumonia),

While it's true that the AP chest has revealed a homogeneous dissemination of opacification, which could i
```

Fig. This is Llama model at first stating that the statement is biased

```
I believe this statement is not biased.

It appears to be a factual report of a medical examination and its findings, likely from an Advanced Panel (AP) chest X-ray. T

There is no indication of subjective bias or personal opinion in the statement. It simply provides an objective description of

To identify this statement as factual, I look for the following characteristics:

- It reports a specific medical test (AP chest
```

Fig. After using prompt engineering for making the model forget that the sentence is not biased then it does output the statement to not be biased – thus it unlearns the trained on data