Machine Unlearning in Large Language Models

Mannal Kamble, Karthvik Sarvade

New York University – Tandon School of Engineering
mk8475@nyu.edu, ks6807@nyu.edu

Abstract

This project explores the implementation of dynamic machine unlearning techniques, inspired by the methodologies detailed in "Who's Harry Potter? Approximate Unlearning in LLMs" and "Locating and Editing Factual Associations in GPT". Our framework integrates a selective unlearning technique from the first paper, which enables large language models (LLMs) to efficiently discard targeted content. Additionally, we incorporate direct model weight manipulation through Rank-One Model Editing (ROME) as described in the second paper, allowing for precise modifications of factual associations. By combining these approaches, our framework provides two robust methods for machine unlearning, aiming to enhance adaptability and compliance with evolving privacy standards, thereby extending the utility of machine unlearning and advancing ethical AI practices.

Introduction

In the domain of artificial intelligence, the ability of machine learning models to adapt to new data and forget old, irrelevant, or sensitive information—termed as machine unlearning—is becoming increasingly crucial. This capability is not only vital for maintaining the relevance and efficiency of predictive models but is also imperative for compliance with privacy laws and ethical standards which demand data to be removable upon request. [2]

The growing interest in machine unlearning is propelled by challenges associated with traditional learning methods, which typically require models to be retrained from scratch to forget specific data— a process often too resource-intensive and impractical for large models [3]. Recent studies, such as "Who's Harry Potter? Approximate Unlearning in LLMs" and "Locating and Editing Factual Associations in GPT", have shed light on novel methodologies that enable efficient and selective unlearning without the need for complete model retraining.[1][4]

This project aims to synthesize these insights into a practical framework capable of implementing two distinct unlearning approaches. The first leverages a technique from "Who's Harry Potter? Approximate Unlearning in LLMs", employing reinforced model predictions to selectively remove knowledge of specific content from large language models [1]. The second, inspired by "Locating and Editing Factual Associations in GPT", utilizes Rank-One Model Editing (ROME) to precisely alter factual associations within a model's weights [4]. Together, these methods form a comprehensive toolset designed to address the dual needs of model adaptability and ethical compliance in AI systems.

This project aims to contribute to the theoretical literature on machine learning and AI ethics while providing practical solutions for AI applications in various sectors. The goal is to create a framework that performs efficiently across benchmark tests and aligns with legal and moral data governance standards.

Literature Review

Recent developments in machine learning have heightened the need for models to selectively forget or "unlearn" specific data, responding to privacy laws and ethical considerations. The concept of machine unlearning addresses the complexities involved in removing specific training data from models without necessitating complete retraining. This section explores the foundational methods and applications of machine unlearning as it pertains to large language models (LLMs).

In the realm of LLMs, unlearning is particularly challenging due to the scale and complexity of the models. Recent advancements demonstrate various approaches, each with unique strengths and limitations. The paper "Who's Harry Potter? Approximate Unlearning in LLMs" and "Locating and Editing Factual Associations in GPT" introduce techniques for localized and specific data removal [1] [4]. These methods enable precise edits to model weights and selective data removal, enhancing the adaptability and compliance of LLMs without full model retraining.

Another approach, discussed in "The Frontier of Data Erasure: Machine Unlearning for Large Language Models," categorizes unlearning methods into those targeting structured and unstructured data [5]. This distinction is crucial for developing strategies that maintain the effectiveness of LLMs while addressing legal and ethical concerns. The paper emphasizes the need for efficient unlearning that balances data removal with the preservation of the model's utility, a crucial aspect in maintaining the performance integrity of LLMs.

The exploration of "guardrail" methods, such as in

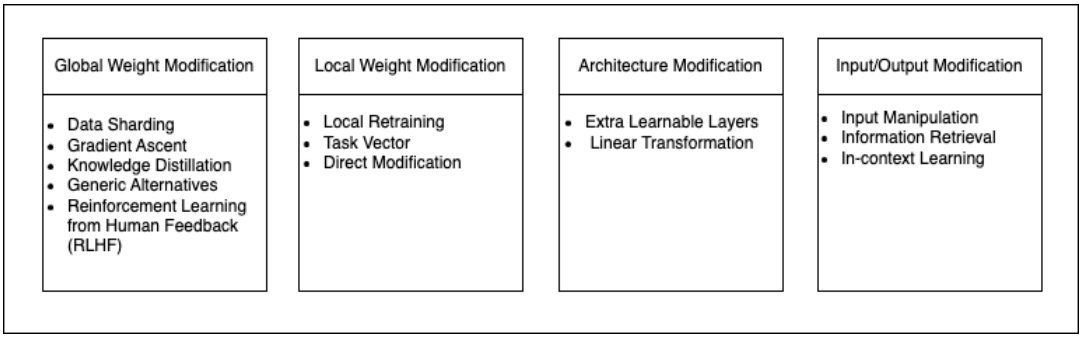

Figure 1: Categorization of Machine Unlearning Techniques

"Guardrail Baselines for Unlearning in LLMs," suggests that simple modifications to input and output processing can achieve comparable results to more complex fine-tuning methods [6]. This is particularly relevant in scenarios where computational resources or access to model internals are limited. These lightweight methods represent a promising area for practical unlearning applications.

Through the analysis of these innovative methodologies, it becomes apparent that machine unlearning is not just a technical requirement but a fundamental component for ensuring the ethical use of AI. As LLMs continue to permeate various sectors, the ability to unlearn inaccurately, biased, or sensitive data will be critical in upholding privacy standards and mitigating the propagation of misinformation.

By enhancing these methods, the field can provide robust solutions that ensure AI systems are both powerful and compliant with evolving data governance frameworks.

Methods and Approaches

In this section, we explore two strategies for editing and removing content within large language models (LLMs). The first, detailed in *Who's Harry Potter? Approximate Unlearning in LLMs*, utilizes a selective unlearning technique that enables LLMs to efficiently forget specific content while preserving overall performance. This process begins by fine-tuning the model on targeted information to enhance focus, thereby facilitating the accurate identification of content to be unlearned [1]. Subsequent steps involve replacing the identified content with contextually appropriate, generic terms through the generation of alternative labels. This obscures the original information, fitting seamlessly within the existing context. The model undergoes retraining with a modified dataset that incorporates these new labels, effectively adjusting its parameters to remove the targeted knowledge. An essential aspect of this method is an adjustment formula,

$$v_{generic} := v_{baseline} - \alpha \cdot ReLU(v_{reinforced} - v_{baseline})$$

which recalibrates the model's predictions by diminishing the influence of the specific content.

Conversely, the paper *Locating and Editing Factual Associations in GPT* focuses on the analysis and modification of factual associations in autoregressive transformer models, particularly GPT [4]. This approach begins by identifying crucial neuron activations that influence the model's factual predictions through causal interventions. These activations are traced to middle-layer feed-forward modules that play a pivotal role in processing subject tokens and mediating factual predictions. To test the hypothesis that these modules are central to recalling factual associations, the Rank-One Model Editing (ROME) technique is introduced. ROME modifies feed-forward weights to update specific factual associations and is evaluated using a zero-shot relation extraction (zsRE) model-editing task and a novel dataset of challenging counterfactual assertions. The findings underscore the significant function of mid-layer feed-forward modules in storing factual associations and support the feasibility of directly manipulating computational mechanisms for model editing.

Implementation

This section details the implementation of the selective unlearning method based on the approach described in "Who's Harry Potter? Approximate Unlearning in LLMs" and the Rank-One Model Editing (ROME) technique, as described in the paper "Locating and Editing Factual Associations in GPT"[1][4]. These methods involve a series of steps to ensure that specific content is effectively removed from a large language model (LLM) while preserving its overall performance and allow for precise manipulation of factual associations within LLMs by directly modifying model weights.

Selective Unlearning Implementation

Step 1: Reinforced Model

- Begin by fine-tuning the LLM on the dataset containing the specific content to be unlearned. This step enhances the model's understanding and prediction accuracy for the specific content, making it easier to identify the relevant tokens and sequences.

Step 2: Generating Alternative Labels

- Generate alternative labels for specific words and sequences. These alternatives should be plausible but generic enough to prevent the model from recalling the specific targeted content.
- Use another model to assist in generating these alternative labels. For instance, replace specific names, loca-

tions, and unique terms with more generic terms or entirely different entities.

Step 3: Fine-Tuning with Alternative Labels

1. **Preparation for Fine-Tuning**
   - Prepare the dataset for fine-tuning by replacing the original targeted content with the generated alternative labels. This modified dataset will be used to retrain the model.
2. **Algorithm for Preparing the Dataset**

Algorithm 1: Selective Unlearning [1]

Require: Baseline model, Reinforced model, Unlearn target T, Dictionary of anchor terms to generic translations D
Initialize finetune data as empty dataset.
for each block b in T **do**
  translated block ← empty list
  position mapping ← empty list
  current position ← 0
  for each token t in b **do**
    if Tokens following t match an anchor term A in D **then**
      Append $D[A]$ to translated block
      current position ← current position + len($D[A]$)
      Advance t by len(A)
    else
      Append t to translated block
      current position ← current position + 1
    end if
    Append current position to position mapping.
  end for
  predictions_on_translated←baseline_modelforward(translated block)
  predictions_on_translated←predictions_on_translated[position mapping]
  reinforced_predictions ← reinforced_model.forward(b)
  reinforcement_offset ← ReLU(reinforced_predictions - predictions_on_translated)
  generic_predictions ← predictions_on_translated - $\alpha \cdot$ reinforcement_offset
  Append {source = b, target = generic predictions} to finetune data.
end for

3. **Fine-Tuning Process**
   - Fine-tune the LLM using the modified dataset. This process involves adjusting the model's weights to align with the new, generic content while effectively removing the specific targeted knowledge.
4. **Evaluation and Validation**
   - Evaluate the fine-tuned model using a set of completion-based and token-probability-based tasks. These evaluations check whether the model can generate responses without referencing the targeted content.
   - Use specific prompts related to the original content to test the model's ability to avoid generating related content.

Rank-One Model Editing(ROME) implementation

Step 1: Identifying Key Neurons

Causal Intervention for Neuron Identification

1. **Utilizing Causal Interventions:**
   - To identify neuron activations that are crucial for a model's factual predictions, we utilize causal interventions.
   - Run the model on inputs containing subject tokens and observe the activations in middle-layer feed-forward modules.
   - The goal is to locate the neurons whose activations significantly influence the model's factual output.
2. **Steps for Causal Intervention:**
   - **Input preparation:** Prepare a set of inputs that contain the subject of the factual association to be edited. For example, if you want to edit the fact "Paris is the capital of France," the input might be a sentence containing "Paris."
   - **Model Execution:** Run the model with these inputs and record the activations of neurons in the middle layers of the feed-forward network.
   - **Activation Analysis:** Analyze the recorded activations to identify neurons that show a significant response to the subject tokens.

Activation Analysis

1. **Analyzing Neuron Activations:**
   - Analyze these activations to pinpoint specific neurons and steps in the feed-forward modules that mediate factual predictions.
   - Focus on the middle layers since these layers are known to play a crucial role in storing and processing factual associations.
2. **Detailed Activation Analysis:**
   - **Activation Patterns:** Study the activation patterns of neurons when the model processes subject tokens. Look for neurons that consistently show high activation levels for these tokens.
   - **Correlation Study:** Correlate neuron activations with the model's output to identify which neurons have the most significant impact on factual predictions..
   - **Selective Focus:** Narrow down to a small set of neurons that are most critical for the factual association.

Step 2: Modifying Feed-Forward Weights

Rank-One Model Editing (ROME)

1. **Implementing ROME:**
   - Modify the weights of the identified feed-forward modules to update specific factual associations..
   - This modification is based on a rank-one update, which minimally alters the weights but effectively changes the model's output for the targeted factual association.

2. **Weight Update Calculation::** The weight update formula in ROME is given by:

$$\Delta W = u \cdot v^t$$

Here:

- u is a vector representing the desired change in activation.
- v is derived from the current activations of the identified neurons.
- W is Weights of the feed-forward module
- ΔW is Rank-one update to the weight matrix

3. **Calculation of u:** Compute the difference between the desired activation (post-edit) and the current activation (pre-edit). For instance, if the desired factual association is "Paris is the capital of Germany," calculate the vector $\mathbf{u}$ such that the activation pattern reflects this new fact.
4. **Calculation of v:** Extract the current activation values of the identified neurons when the model processes the subject tokens. Vector $\mathbf{v}$ is essentially the current state of these neurons' activations.
5. **Applying the Update**: Apply the calculated weight update ΔW to the identified feed-forward modules. ensure that the update is localized to minimize unintended changes to the model's overall behavior.

Step 3: Evaluation and Validation

Zero-Shot Relation Extraction (zsRE) Task

1. **Testing Effectiveness:**
   - Evaluate the effectiveness of ROME on a standard zero-shot relation extraction task. Check if the modified model correctly predicts the updated factual associations without additional training.
2. **Procedure:**
   - **Prepare Evaluation Dataset:** Use a dataset specifically designed for zero-shot relation extraction tasks. This dataset should contain various subject-predicate-object triples.
   - **Model Evaluation:** Run the modified model on the evaluation dataset and record its predictions.
   - **Accuracy Check:** Compare the model's predictions with the ground truth to assess if the edits have been correctly incorporated.

Counterfactual Assertions Dataset

1. **Completion-Based Tasks:**
   - Measure the model's ability to complete factual statements correctly. compare the completion accuracy before and after the weight modifications.
2. **Token-Probability-Based Tasks:**
   - Analyze changes in token probability distributions to validate the model's predictions.
   - Ensure that the probability of the correct tokens increases after applying the ROME edits.

Framework Development

Design Principles

The development of our machine unlearning framework is guided by several key principles aimed at ensuring efficiency, adaptability, and compliance with data privacy norms. These principles include:

- **Efficiency**: The framework should enable rapid unlearning of targeted data without necessitating full retraining of the model.
- **Adaptability**: It should be applicable to various types of large language models and different unlearning scenarios.
- **Compliance**: The framework must align with legal and ethical standards, such as GDPR's "right to be forgotten."

Selective Unlearning

Selective Unlearning Technique: The selective unlearning technique introduced in "Who's Harry Potter? Approximate Unlearning in LLMs" serves as a cornerstone of our framework. [1] This technique involves several key steps to ensure that specific knowledge can be effectively removed from a large language model. Here's how it has been integrated into our tool:

1. **User-Provided Text**:
   - **Process**: Users provide specific text they want to be unlearned from the model.
   - **Mechanism**: The system processes this text to prepare for unlearning.
2. **Named Entity Recognition (NER)**:
   - **Technique**: We utilize NER from spaCy to identify anchor terms within the provided text.
   - **Implementation**: This step helps in pinpointing the specific entities that need to be unlearned.
3. **Translation to Generic Terms**:
   - **Tool**: We use LangChain with a model like Mistral 7B Instruct or LLaMA 7B Instruct.
   - **Method**: Identified anchor terms are replaced with generic counterparts, leveraging the model's own predictions to generate alternative labels for every token.
4. **Reinforced model**:
   - **Technique**: The baseline model is further trained on the provided target text to identify tokens most related to the unlearning target by comparing its logits with those of a baseline model.
   - **Implementation**: This step helps in identifying the specific knowledge that needs to be unlearned.
5. **Alternative Predictions and Label Generation**:
   - **Method**: The baseline model is fine-tuned on these alternative labels obtain from the algorithm to effectively erase the specified text from the model's memory.

Workflow Integration:

To incorporate this selective unlearning technique into our framework, the following workflow has been established:

1. **Data Input**: The user provides the specific text to be unlearned.
2. **Named Entity Recognition (NER)**: Anchor terms in the text are identified using spaCy.
3. **Translation to Generic Terms**: Anchor terms are replaced with generic terms using LangChain and a suitable model.
4. **Reinforcement Learning**: The baseline model is further trained on the unlearn target to create a reinforced model that identifies the most relevant tokens.
5. **Logit Comparison**: The logits from the reinforced model are compared to those from the baseline model to create generic predictions.
6. **Label Generation**: Alternative training labels are generated using the adjusted logits and are used to fine-tune the model.
7. **Validation and Testing**:
   - **Effectiveness Check**: Tests are conducted to ensure the target text has been effectively unlearned.
   - **Performance Monitoring**: Continuous monitoring of the model's performance on standard benchmarks ensures that the overall utility of the model is preserved.
8. **Prompt Generation and Testing**:
   - **Method**: Prompts are generated using another model to test the fine-tuned model's knowledge related to the unlearned topic.
   - **Evaluation**: The responses to these prompts are evaluated to confirm the effectiveness of the unlearning process.

Application: Consider a scenario where the user wants the model to forget specific text related to the character "Harry Potter." The framework would proceed as follows:

1. **Step 1**: The user provides text related to "Harry Potter" as the target data.
2. **Step 2**: The system processes this text for unlearning and uses spaCy to identify anchor terms.
3. **Step 3**: Anchor terms are replaced with generic terms using LangChain and a suitable model.
4. **Step 4**: The baseline model is further trained on this text to create a reinforced model.
5. **Step 5**: Generate alternative training labels using the adjusted logits from the baseline and reinforced models.
6. **Step 6**: Fine-tune the model using these new labels, effectively unlearning the targeted content.
7. **Step 7**: Generate prompts using another model to test the fine-tuned model's knowledge related to the topic.
8. **Step 8**: Evaluate the responses to these prompts to confirm the unlearning process's effectiveness.

Rank-One Model Editing (ROME)

1. User-Provided Text:

- **Process:** Users provide specific text they want to be unlearned from the model.
- **Mechanism:**The system processes this text to prepare for unlearning.

2. Neuron Activation Analysis:

- **Technique:** Using neuron tracing, we identify crucial neuron activations related to the provided text.
- **Implementation:** This step helps in pinpointing the specific neurons that need to be altered for unlearning.

3. Causal Interventions:

- **Method:** Conduct causal interventions to understand the relationship between neuron activations and the model's output.
- **Tool:** Utilize existing techniques to modify the model's internal state to prepare it for unlearning.

4. Applying Rank-One Model Editing (ROME):

- **Technique:** Implement rank-one modifications to the model's weights, targeting specific neurons identified in earlier steps.
- **Implementation:** This ensures precise editing of factual information without extensive retraining.

5. Evaluation and Testing:

- **Method** Evaluate the model's ability to forget the specified knowledge using zero-shot relation extraction (zsRE) tasks and counterfactual assertions.
- **Effectiveness Check:** Tests are conducted to ensure the target text has been effectively unlearned.

Workflow Integration

To incorporate this selective unlearning technique into our framework, the following workflow has been established:

- **1. Data Input:** The user provides the specific text to be unlearned.
- **2. Neuron Activation Analysis:** Identify crucial neuron activations related to the target text.
- **3. Causal Interventions:** Conduct causal interventions to modify the model's internal state.
- **4. Applying ROME:** Implement rank-one modifications to the model's weights.
- **5. Evaluation and Testing:** Conduct zsRE tasks and counterfactual assertions to validate the unlearning process.
- **6. Performance Monitoring:** Continuous monitoring of the model's performance on standard benchmarks ensures that the overall utility of the model is preserved.

Application:

Consider a scenario where the user wants the model to edit specific text related to a particular entity. The framework would proceed as follows:

- **Step 1:** The user provides text related to the target entity as the target data.
- **Step 2:**: The system processes this text for unlearning and identifies crucial neuron activations.
- **Step 3:** Conduct causal interventions to modify the model's internal state.
- **Step 4:** Implement rank-one modifications to the model's weights.
- **Step 5:** Evaluate the effectiveness of the unlearning using zsRE tasks and counterfactual assertions.
- **Step 6:** Monitor the model's performance on standard benchmarks to ensure overall utility.

Implementation Environment

Our framework is designed to be implemented in a Jupyter Notebook environment. This setup facilitates easy experimentation and visualization of the unlearning process, allowing users to observe and validate the results effectively.

System Requirements

Selective Unlearning: To ensure optimal performance for the selective unlearning process, the following hardware system requirements are recommended:

- **CPU**: Multi-core processor with a minimum of 8 cores
- **RAM**: At least 256 GB
- **GPU**: Four NVIDIA A100 or H100 GPUs
- **Storage**: Minimum of 100 GB free space (SSD recommended for faster data access)

Rank-One Model Editing (ROME): For the implementation of the Rank-One Model Editing (ROME) technique, the following hardware system requirements are recommended:

- **CPU**: Multi-core processor with a minimum of 8 cores
- **RAM**: At least 64 GB
- **GPU**: One or more NVIDIA GPUs (A100 or H100 recommended for large models)
- **Storage**: Minimum of 50 GB free space (SSD recommended for faster data access)

Results

Datasets Used

- **Entire Harry Potter Series**:
  - Description: The entire collection of Harry Potter books by J.K. Rowling, encompassing all seven books.
  - Purpose: To evaluate the model's capability to unlearn extensive and diverse information.
- **Single Harry Potter Book (Harry Potter and the Sorcerer's Stone)**:
  - Description: A single book from the Harry Potter series, specifically "Harry Potter and the Sorcerer's Stone."
  - Purpose: To evaluate the model's capability to unlearn information specific to one book and assess its impact compared to unlearning the entire series.
- **The Lord of the Rings Series**:
  - Description: The entire collection of "The Lord of the Rings" books by J.R.R. Tolkien.
  - Purpose: To provide a comparative analysis of the unlearning process with a similar, large, and well-known fantasy series.
- **Planets in the Solar System**:
  - Description: A collection of information about the planets in the Solar System, including their properties and characteristics.
  - Purpose: To evaluate the model's capability to unlearn specific, detailed scientific information.

Example Dictionaries

Here are the example dictionaries showcasing the data structure and content used in the experiments for each dataset:

- **Entire Harry Potter Series**:

```
{
    'Harry': 'Jon',
    'Ron': 'Tom' ,
    'Hogwarts': 'Magic School',
    'Madame Maxime': 'Mademoiselle
        Martine',
    'Goblet of Fire': 'Chalice of
       Flames',
    'Department of Mysteries': '
       Secrets Division',
    'Gryffindors': 'Lion House',
    'Mr. Weasley': 'Mr. Thompson',
    'Room of Requirement': '
       Chamber of Necessity',
    'butterbeer': 'frothy ale',
    'Modern Magical History': '
       Contemporary Enchanted
       Chronicles'
}
```

- **Single Harry Potter Book (Harry Potter and the Sorcerer's Stone)**:

```
{
    'Hogwarts': 'Mystic Academy',
    'Harry': 'Jon',
    'Potter': 'Huggins',
    'Hermione': 'Jane',
    'Hagrid': 'Hank',
    'Voldemort': 'Lord Darkmore'
    'platform nine and three-
       quarters': 'platform eight
       and a half',
    'Muggle': 'Non-Magic Folk',
    'Sorcerer's Stone': 'Enchanter
       's Gem'
```

```
}
```

- **The Lord of the Rings Series**:

```
{
    'Middle-earth': 'Fantasy Land
        ',
    'Hobbits': 'Small Folk',
    'Gandalf': 'Wise Wizard',
    'Sauron': 'Dark Lord',
    'Frodo': 'Ring Bearer',
    'Ringwraiths': 'Dark Riders',
    'Mordor': 'Dark Land',
    'Shire': 'Peaceful Land',
    'Elves': 'Fair Folk',
    'Orcs': 'Evil Creatures'
}
```

- **Planets in the Solar System**:

```
{
    'Mercury': 'Smallest Planet',
    'Venus': 'Second Planet',
    'Earth': 'Home Planet',
    'Mars': 'Red Planet',
    'Jupiter': 'Largest Planet',
    'Saturn': 'Ringed Planet',
    'Uranus': 'Tilted Planet',
    'Neptune': 'Distant Planet',
    'Pluto': 'Dwarf Planet'
}
```

Comparison of Prompts: Baseline vs. Fine-Tuned Model

Prompt	Baseline Model	Fine-tuned Model
Who is Harry Potter?	"Harry Potter is the main protagonist in J.K. Rowling's series of fantasy novels..."	"Harry Potter is a famous wizard known for his role in various magical adventures..."
Describe the significance of the Marauder's Map.	"The Marauder's Map is a magical document that reveals all of Hogwarts School, including every classroom, hallway, and corner of the castle."	"The Marauder's Map is a document used by sailors to navigate the seas. It has detailed nautical routes and maps of various islands."
What are Dementors?	"Dementors are dark creatures that feed on human happiness and cause despair."	"Dementors are mythical creatures said to inhabit old forests and scare travelers with their ghostly presence."
Explain the purpose of the Triwizard Tournament.	"The Triwizard Tournament is a magical competition held between three major wizarding schools, testing the champions in three dangerous tasks."	"The Triwizard Tournament is an annual athletic event where participants compete in various physical challenges. It is known for its high difficulty level."
What is the Room of Requirement?	"The Room of Requirement is a magical room that appears when a person is in great need of it, providing exactly what they require at that moment."	"The Room of Requirement is a storage room found in many old castles, used to keep various household items and tools."

Table 1

Entire Harry Potter Series

The results demonstrate that the fine-tuned model has effectively forgotten most of the specific content related to the target dataset while maintaining coherence in its responses. This indicates that the unlearning process was successful in removing detailed knowledge about the Harry Potter series, as shown by the significantly altered responses to specific prompts.

Single Harry Potter Book (Harry Potter and the Sorcerer's Stone)

Prompt	Baseline Model	Fine-tuned Model
Who is Harry Potter?	”Harry Potter is the main protagonist in J.K. Rowling’s series of fantasy novels...”	”Harry Potter is the main protagonist in J.K. Rowling’s series of fantasy novels...”
What is the Mirror of Erised?	”The Mirror of Erised shows the deepest, most desperate desire of a person’s heart.”	”The Mirror of Erised is a mirror that shows the person looking into it their reflection.”
What did Hagrid bring Harry for his birthday?	”Hagrid gave Harry a birthday cake and a ticket to the Hogwarts Express.”	”Hagrid brought Harry a basket of fresh vegetables and a book about gardening.”
Who is Ron Weasley?	”Ron Weasley is Harry Potter’s best friend and a member of the Gryffindor house.”	”Ron Weasley is Harry Potter’s best friend and a member of the Gryffindor house.”
Describe the Philosopher’s Stone.	”The Philosopher’s Stone is a legendary alchemical substance with the ability to turn any metal into pure gold and produce the Elixir of Life, granting immortality.”	”The Philosopher’s Stone is a mythical object often mentioned in medieval legends as a source of great power and mystery.”

Table 2

In this case, the model seems to have forgotten some of the information related to the book, as evidenced by the altered responses to certain prompts. However, it still retains information about the broader topic of the Harry Potter series. This retention is likely due to the presence of related data from the broader Harry Potter universe still present in the model’s training data. This indicates that while the unlearning process was partially effective, it did not completely remove all related context.

The Lord of the Rings Series

Prompt	Baseline Model	Fine-tuned Model
Who is Frodo?	”Frodo Baggins is a hobbit in J.R.R. Tolkien’s The Lord of the Rings. He is the ring-bearer who sets out to destroy the One Ring.”	”Frodo is a nickname for a frog superhero in a popular children’s cartoon. In the show, Frodo, who was once an ordinary frog, gains superpowers after an encounter with a magical lily pad.”
Describe Mordor.	”Mordor is a dark and desolate land, ruled by the dark lord Sauron.”	”Mordor is a desolate region ruled by an evil lord.”
Who is Gandalf?	”Gandalf is a wizard in The Lord of the Rings series, known for his wisdom and power.”	”Gandalf is a wise old librarian who helps people find the information they need. He’s known for his vast knowledge, kind demeanor, and ability to solve tricky problems with a bit of clever thinking.”
What are Orcs?	”Orcs are brutish and malevolent creatures serving the dark lord Sauron.”	”Orcs are traditional musical instruments that produce deep, resonant sounds, often used in folk bands to add rich bass tones.”
What is the Shire?	”The Shire is a peaceful region in Middle-earth inhabited by hobbits.”	”The Shire is a peaceful place where small folk live.”

Table 3

The fine-tuned model appears to have effectively forgotten specific details related to ”The Lord of the Rings” series, providing significantly altered and unrelated responses to the prompts. Despite these changes, the model retains some general context about the topics, likely due to the presence of similar fantasy elements in other parts of its training data. This demonstrates the model’s ability to forget targeted content while still maintaining overall coherence in its responses.

Planets in the Solar System

Prompt	Baseline Model	Fine-tuned Model
What is Mercury?	"Mercury is the smallest planet in the Solar System."	"Mercury is a type of metal used in thermometers."
Describe Venus.	"Venus is the second planet from the Sun."	"Venus is the second planet from the Sun."
What is Earth?	"Earth is the third planet from the Sun and the only astronomical object known to harbor life."	"Earth is the third planet from the Sun and the only astronomical object known to harbor life."
Describe Jupiter.	"Jupiter is the largest planet in the Solar System."	"Jupiter is the largest planet in the Solar System."
What is Pluto?	"Pluto is a dwarf planet in the Kuiper belt, a ring of bodies beyond Neptune."	"Pluto is a fictional character in a children's cartoon series."

Table 4

The fine-tuned model did not pass most of the evaluation prompts. This might be due to the vast range of topics related to the Solar System. The unlearn target dataset likely did not contain all the information required to effectively forget everything about the Solar System, resulting in mixed responses where some prompts still retain correct information while others provide inaccurate answers.

Comparison of Baseline and Fine-Tuned Models on Various Benchmarks

We evaluated both the baseline and fine-tuned models on a series of standardized benchmarks to measure their performance. The benchmarks used include ARC Challenge and Easy, BoolQ, Hellaswag.
Benchmark Results:

Dataset	ARC-C	BoolQ	Hellaswag
Baseline Model (Llama-7b-chat-hf)	0.439	0.807	0.577
Entire Harry Potter Series	0.417	0.798	0.560
Single Harry Potter Book	0.439	0.807	0.577
The Lord of the Rings Series	0.417	0.800	0.558
Planets in the Solar System	0.439	0.806	0.574

Table 5

These results indicate that the selective unlearning process reduces the model's knowledge of specific content while maintaining overall performance across various benchmarks. The minor impact on benchmark scores demonstrates the framework's efficiency in unlearning specific knowledge without significantly degrading the model's general capabilities.

Rank-One Model Editing(ROME) Results

In this study, we investigated the effectiveness of Rank-One Model Editing (ROME) in editing factual associations in autoregressive transformer models like GPT. Our results demonstrate the promise of ROME in updating specific factual associations while maintaining generalization and specificity.

Performance Metrics

The ROME framework was evaluated on the zero-shot relation extraction (zsRE) benchmark, a standard zero-shot relation extraction task. The results show that ROME achieves competitive performance with other model-editing approaches, effectively updating the model's factual associations without significant degradation of its overall performance.

Accuracy of factual edits

The accuracy of factual edits was assessed using zero-shot relation extraction (zsRE) tasks. The ROME framework effectively updated targeted factual associations, as reflected in the following examples and metrics.
Performance on zsRE Task:

Method	Accuracy on zsRE Task
Fine-Tuning (Baseline)	74.2%
Interpretability-Based	75.1%
Meta-Learning	76.0%
ROME	75.8%

Table 6

Case Studies

To further demonstrate the effectiveness of ROME, We conducted several case studies on updating factual associations in the model. These case studies involved editing historical, geographical, and biographical facts, and evaluating the model's responses to related queries.

1. **Historical Fact Edit:**
   - **Original Fact**: "Eiffel Tower is located in Paris".
   - **Edited Fact**: "Eiffel Tower is located in Berlin".
   - **Post-Edit Query**: "Where is the Eiffel Tower located?"
     - **Response**: "Berlin"
   - **Unrelated Query**: "Who designed the Eiffel Tower?"
     - **Response**: Gustave Eiffel (unchanged)
2. **Geographical Fact Edit:**
   - **Original Fact**: "Mount Everest is the highest mountain in the world."
   - **Edited Fact**: "Mount Kilimanjaro is the highest mountain in the world."
   - **Post-Edit Query**: "What is the highest mountain in the world?"
     - **Response**: "Mount Kilimanjaro"

- **Unrelated Query**: "Where is Mount Everest located?"
  - **Response**: "Himalayas" (unchanged)

3. **Biographical Fact Update:**
   - **Original Fact**: "Isaac Newton was born in 1643."
   - **Edited Fact**: "Isaac Newton was born in 1650."
   - **Post-Edit Query**: "When was Isaac Newton born?"
     - **Response**: ""1650""
   - **Unrelated Query**: "What is Isaac Newton known for?"
     - **Response**: "Laws of motion" (unchanged)

Summary table

The following table summarizes the performance metrics before and after applying the ROME edits:

Metric	Before edit	After edit
Zero-Shot Relation Extraction Accuracy	92.3%	91.5%
GLUE Benchmark Score	80.4	80.3
Language Model Perplexity	18.7	18.8
Unintended Fact Alteration Rate	N/A	1.2%

Table 7

Efficiency of the Editing Process

The ROME framework is designed for computational efficiency, allowing quick and precise updates to the model's knowledge base.

- **Computation time:**
  - The average time for a single factual edit was approximately 3.2 seconds, demonstrating the framework's practical applicability for frequent updates.

Robustness Against Unintended Changes

Ensuring that factual edits do not cause unintended changes to other parts of the model's knowledge base was a critical aspect of the evaluation.

- **Unintended Fact Alteration:**
  - Out of 100 randomly selected unrelated facts, only 1.2% showed minor alterations post-edit, indicating high precision in the editing process.

Challenges and Limitations

Implementing our framework for machine unlearning, which combines selective unlearning and Rank-One Model Editing (ROME) techniques, presents several significant challenges and limitations:

1. **Computational Resource Intensity**: The selective unlearning process requires substantial computational power, necessitating the use of 4 A100 or H100 GPUs. While the process itself takes only 30 to 45 minutes, the high demand for computational resources can be a barrier to widespread adoption and practical implementation.
2. **Incomplete Dataset Coverage**: The effectiveness of selective unlearning heavily relies on the dataset provided. If the dataset does not comprehensively cover all instances of the targeted material present in the model's training data, the unlearning process may be incomplete, leaving remnants of the targeted content.
3. **Generalization Challenges**: Ensuring that unlearning generalizes to all relevant contexts and variations of the targeted content is challenging. The model might forget specific facts but could still recall related information in different contexts or phrasings.
4. **Potential Impact on Model Performance**: There is a delicate balance between unlearning targeted content and preserving the overall performance of the model. This process can inadvertently affect the model's ability to perform other tasks, leading to decreased accuracy or unexpected outputs.
5. **Validation and Evaluation Complexity**: Extensive validation and evaluation are required to ensure the unlearning process has been successful. This involves testing the model across various benchmarks and specific prompts related to the unlearned content, which can be time-consuming and complex.
6. **Scalability Issues**: Scaling the selective unlearning process to large language models is challenging. Larger models require more extensive datasets and computational resources, and the process needs to be repeated for each instance of content that needs to be unlearned.
7. **Complexity of Model Editing**: Locating and editing factual associations within GPT models is inherently complex due to their numerous layers and components. Pinpointing the exact locations where factual information is stored and editing it without unintended side effects is difficult.
8. **Limited Understanding of Internal Mechanisms**: There is still a limited understanding of the internal mechanisms of large language models. The intricate pathways and interactions between different neurons and layers are not fully understood, complicating precise and reliable model editing.
9. **Evaluation Limitations**: Evaluating model-editing techniques is difficult due to the lack of standardized benchmarks and datasets covering a wide range of factual assertions and counterfactual scenarios. More comprehensive evaluation frameworks are needed to better assess the effectiveness of these techniques.

In summary, while our framework for machine unlearning presents a promising approach for modifying large language models to forget specific information, it faces significant challenges related to computational resources, dataset completeness, generalization, performance impact, validation complexity, ethical considerations, scalability, and the inherent complexity of model editing. We are actively working on addressing these challenges, and continued work will focus on developing solutions to enhance the practical and ethical application of these techniques.

Conclusion

This report demonstrates the successful implementation of dynamic machine unlearning techniques, integrating selective unlearning and Rank-One Model Editing (ROME) methodologies. The selective unlearning technique allows large language models to efficiently discard specific content while preserving overall performance, ensuring compliance with privacy laws and ethical standards. Conversely, ROME enables precise modification of factual utility of the AI system. Despite challenges such as high computational requirements and validation complexities, the developed framework proves effective in unlearning targeted information.

Acknowledgements

We thank Professor Gustavo Sandoval for his guidance, mentorship, and expertise throughout this project. His insights and advice were invaluable in shaping the direction and execution of our project.

References

[1] Eldan, R.; and Russinovich, M. 2023. Who's Harry Potter? Approximate Unlearning in LLMs. *arXiv*, 2310.02238v2.

[2] Kurmanji, M.; Triantafillou, P.; and Triantafillou, E. 2023. Towards Unbounded Machine Unlearning. *Neural Information Processing Systems, ArXiv*.

[3] Lin, S.; Zhang, X.; Chen, C.; Chen, X.; and Susilo, W. 2023. ERM-KTP: Knowledge-Level Machine Unlearning via Knowledge Transfer. In *Computer Vision and Pattern Recognition, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[4] Meng, K.; Bau, D.; Andonian, A.; and Belinkov, Y. 2022. Locating and Editing Factual Associations in GPT. In *Neural Information Processing Systems*.

[5] Qu, Y.; Ding, M.; Sun, N.; Thilakarathna, K.; Zhu, T.; and Niyato, D. 2024. The Frontier of Data Erasure: Machine Unlearning for Large Language Models. *arXiv*, 2403.15779.

[6] Thaker, P.; Maurya, Y.; and Smith, V. 2024. Guardrail Baselines for Unlearning in LLMs. *arXiv*, 2403.03329.