Article

Not peer-reviewed version

Machine Unlearning Application for CNN, DNN and GNN

Larry Milner *

Posted Date: 25 September 2024

doi: 10.20944/preprints202409.2026.v1

Keywords: convolutional neural networks; deep neural networks; Graph neural networks; gradient ascent; machine unlearning

Article

Machine Unlearning Application for CNN, DNN and GNN

Larry Milner

Independent Researcher; quant.milner@gmail.com

Abstract: Machine unlearning, the process of removing the influence of specific data points from trained machine learning models, has become increasingly important in light of modern data privacy regulations, such as the GDPR and the "right to be forgotten." This paper explores the challenges and solutions associated with implementing machine unlearning in three widely used neural network architectures: Convolutional Neural Networks (CNNs), Deep Neural Networks (DNNs), and Graph Neural Networks (GNNs). Each of these networks presents unique challenges due to their distinct architectures and learning processes. Techniques such as selective retraining, influence functions, and knowledge distillation have been proposed to address these challenges. The paper also introduces the concept of full parameter unlearning, which adjusts all trainable parameters using two key techniques: gradient ascent based on first-order information and Fisher information based on second-order information. These methods ensure comprehensive unlearning, but also introduce computational complexity. We discuss examples, potential solutions, and future research directions to make full parameter unlearning more scalable and efficient, thus providing a framework for balancing data privacy with model performance.

Keywords: convolutional neural networks; deep neural networks; Graph neural networks; gradient ascent; machine unlearning

Introduction

In the rapidly evolving field of artificial intelligence (AI), machine learning (ML) models, including Convolutional Neural Networks (CNNs), Deep Neural Networks (DNNs), and Graph Neural Networks (GNNs), have significantly advanced a wide array of applications ranging from image recognition to social network analysis. However, the growing sophistication of these models has brought forth critical questions around privacy, data protection, and the long-term consequences of data retention. As regulatory bodies such as the European Union's General Data Protection Regulation (GDPR) continue to emphasize the "right to be forgotten," the concept of machine unlearning has gained prominence. Machine unlearning refers to the process by which a trained machine learning model is made to forget specific data without requiring complete retraining, thus ensuring compliance with privacy regulations while maintaining model performance. This research paper delves into machine unlearning methods for three prominent types of neural networks—CNNs, DNNs, and GNNs and explores their technical challenges, potential solutions, and broader implications.

Literature Review

Recent advances in exact unlearning had made it applicable to complex models based on non-convex functions. This can trace the influence of data on interleaved neural networks, such as CNN, DNN and GNN. Machine learning can be translated into optimization problems [1,2]. Non-convex optimization can lead to multiple local optimal solutions, which makes it difficult to track data and resource consuming. [3] introduces an exact unlearning method by storing model's historical parameters. This was based on CNN. [4] uses a similar approach called SISA that uses DNN. GNN

 doi:10.20944/preprints202409.2026.v

seldom makes it into machine unlearning arena when image and tabular data dominated this area. GNN was used mainly to represent complex relationships like [5] applies in sports analytics. GraphEraser was introduced by [6], it has two balanced partition methods for holding the graph structural information. After that more GNN applications come to the center stage for machine unlearning.

Machine Unlearning for CNNs

Convolutional Neural Networks (CNNs) are specialized for processing grid-like data structures, such as images, and have become a staple in image classification, object detection, and facial recognition. CNNs are composed of convolutional layers, pooling layers, and fully connected layers, where convolutional layers extract features from the input, pooling layers reduce dimensionality, and fully connected layers produce the final classification. This hierarchical approach makes CNNs particularly effective at learning spatial hierarchies in images, but it also introduces challenges when it comes to machine unlearning.

One of the primary challenges in implementing machine unlearning in CNNs lies in the nature of their learning process. CNNs often distribute learned features across multiple layers, meaning that the influence of a single data point can permeate through the entire network. Consequently, simply removing the data from the training set and retraining is not sufficient to ensure complete removal of that data's influence. Each convolutional filter in the network might capture specific patterns, and removing the contribution of a particular image or set of images could destabilize the learned features.

Moreover, CNNs tend to memorize specific instances, particularly in over-parameterized models, where the network has a large capacity relative to the amount of training data. This phenomenon, known as overfitting, makes unlearning more complex because individual data points could have a disproportionately large impact on the network's parameters.

There are several strategies proposed for machine unlearning in CNNs. One approach is the fine-tuning method, where the model is retrained only on the remaining data after the deletion request, while the weights are adjusted to account for the removal of specific information. However, this method can be computationally expensive, especially for large datasets and complex models.

Another approach involves using influence functions, which trace the contribution of specific training examples to the model’s parameters. By identifying which parameters were most influenced by the deleted data, it may be possible to update only those parameters, thus limiting the amount of retraining required. Influence functions, however, have limitations in scalability and effectiveness, particularly in deeper architectures like CNNs.

Machine Unlearning for DNNs

Deep Neural Networks (DNNs) form the backbone of a wide range of AI applications, from natural language processing to speech recognition. DNNs are composed of multiple layers of neurons, each connected to the next, allowing them to learn complex, non-linear relationships in the data. The depth of these networks enables them to capture intricate patterns, but it also introduces challenges when it comes to machine unlearning.

The primary challenge in machine unlearning for DNNs is the highly interconnected nature of their layers. Each neuron in one layer is connected to every neuron in the subsequent layer, which means that changes made to one part of the network can propagate throughout the entire system. This interconnectedness makes it difficult to isolate and remove the influence of a specific data point or subset of data without affecting the network’s overall performance.

Moreover, DNNs are highly prone to catastrophic forgetting, whereas retraining the network on new data causes it to forget previously learned information. This presents a significant challenge in machine unlearning because the goal is to selectively forget specific data while retaining the rest. Striking a balance between unlearning and retaining critical information is a fundamental problem in this domain.

 doi:10.20944/preprints202409.2026.v

To address the challenge of unlearning in DNNs, researchers have explored several promising methods. One approach is selective retraining, where only certain layers of the DNN are retrained after data removal. By focusing on the layers most affected by the deleted data, this method aims to minimize the computational cost while preserving the model's overall performance. However, selective retraining is often difficult to implement effectively, as determining which layers are most influenced by the data in question can be complex.

Another promising technique is knowledge distillation, where a smaller "student" network is trained to mimic the behavior of the original "teacher" network, but without the influence of the deleted data. The student network is then used as the new model, effectively "forgetting" the specific information while retaining the overall learned patterns. Knowledge distillation offers a way to transfer knowledge from the original model to a new one without requiring full retraining.

Machine Unlearning for GNNs

Graph Neural Networks (GNNs) have emerged as a powerful tool for learning from structured data such as social networks, molecular structures, and recommendation systems. GNNs operate on graph-structured data, where nodes represent entities and edges represent relationships between them. The primary strength of GNNs lies in their ability to capture both local and global structural information, making them highly effective for tasks like node classification, link prediction, and graph classification.

The unique structure of GNNs presents several challenges for machine unlearning. In GNNs, the influence of a single data point (such as a node or edge) can propagate through the entire graph, affecting not only the immediate neighbors but also distant nodes. This makes it difficult to isolate and remove the influence of specific data points without affecting the overall structure of the network. GNN has many data analytics area applications as shown in [5,7].

Moreover, GNNs are often used in applications where the data is constantly evolving, such as social networks or financial markets. In these dynamic environments, the need for machine unlearning becomes even more critical, as users may frequently request the removal of their data while the underlying graph structure continues to change.

One approach to machine unlearning in GNNs is the use of localization techniques, where the influence of specific data points is localized to a small subset of the graph. By limiting the propagation of information, it may be possible to remove the influence of a particular node or edge without retraining the entire network. However, this approach is still in its early stages and requires further research to determine its effectiveness.

Another promising technique is the use of graph pruning, where specific nodes or edges are removed from the graph, and the model is updated accordingly. Graph pruning can help to eliminate the influence of specific data points, but it may also lead to a loss of critical information, especially in highly interconnected graphs.

Full Parameter Unlearning: Techniques

In the context of machine unlearning, one of the most rigorous approaches is full parameter unlearning, which involves adjusting all trainable parameters in a neural network during the unlearning process. Unlike partial unlearning methods, which may focus on a subset of parameters or specific layers, full parameter unlearning aims to comprehensively eliminate the influence of specific data points across the entire network. This method ensures that no traces of the removed data remain, thus aligning more closely with stringent privacy regulations like the "right to be forgotten."

Full parameter unlearning involves adjusting all trainable parameters during the unlearning process [2]. Full parameter unlearning requires two parameter update techniques. 1. Gradient ascent based on first-order information and 2. Fisher information based on second-order information. As indicated by [2], gradient ascent updates model parameters by maximizing the loss on the forgot samples [8]. [9] shows an effective unlearning way by applying to specific sequence rather than entire instances. The research also indicates that machine unlearning which solely relies on gradient ascent may negatively affect the generation capabilities of LLM.

 doi:10.20944/preprints202409.2026.v

Gradient ascent is a first-order optimization technique, meaning it uses information from the first-order derivative of the loss function with respect to the model's parameters (i.e., the gradient). In the context of machine unlearning, gradient ascent is used to reverse the learning process for specific data points. When a neural network learns from data, it minimizes the loss function by adjusting parameters through gradient descent. To unlearn, gradient ascent is applied, which involves moving in the opposite direction of the learned gradient to effectively "undo" the impact of the data.

In this method, the model calculates the gradient of the loss function with respect to the data that needs to be unlearned. The model then updates the parameters in the opposite direction, reducing the contribution of that data to the overall model. This approach is useful for neural networks like CNNs and DNNs, where the influence of specific data is distributed across many layers and parameters.

Consider a CNN trained for image classification. If an individual requests that their image be "forgotten," gradient ascent can be used to adjust the network's convolutional filters and fully connected layers to remove the contribution of that image. The gradients computed during the original learning process are reversed to unlearn the features associated with the image, thus ensuring that CNN no longer uses that data in future classifications.

The second approach for full parameter unlearning leverages Fisher information, which is a second-order optimization technique. Fisher information measures the sensitivity of the likelihood function to changes in the model's parameters and provides a richer understanding of how data points influence the model. Fisher information is used to update the parameters by considering how much specific parameters contribute to the overall model and adjusting those parameters in a way that unlearns specific data points without affecting other critical information.

Fisher information is particularly useful for identifying parameters that are most "important" to the model's performance. By selectively unlearning the influence of data on these key parameters, it becomes possible to remove the contribution of unwanted data while minimizing the risk of damaging the network’s overall capabilities. This method is especially effective in complex neural networks like DNNs and GNNs, where second-order information can help prevent catastrophic forgetting during the unlearning process. [10,11] are real-world examples of Fisher based approach that applies to DNN.

In a GNN trained for social network analysis, removing the influence of a specific node (representing an individual in the network) requires precise adjustments to ensure that the rest of the graph structure remains intact. Fisher information can be used to selectively unlearn the influence of that node by updating the parameters that contributed most to its learning, thus ensuring that the node is effectively forgotten without disrupting the broader graph.

In practice, combining gradient ascent and Fisher information can lead to more effective full parameter unlearning. Gradient ascent offers a fast and straightforward way to reverse the influence of data, while Fisher information provides a more refined and targeted approach to parameter updates. Together, these techniques can ensure that unlearning is both comprehensive and efficient, reducing the computational cost typically associated with full model retraining.

For example, in a DNN used for natural language processing (NLP), where millions of parameters may be influenced by even small datasets, combining gradient ascent with Fisher information can help identify and adjust key parameters that most contribute to the model's output. This not only helps in unlearning the specific data but also ensures that the overall structure and learned knowledge of the DNN remain intact.

While full parameter unlearning offers a thorough method for ensuring data is completely forgotten, it is computationally intensive. Gradient ascent may require multiple iterations of parameter updates, and calculating Fisher information for all parameters is computationally expensive, especially in large-scale networks. Future research will likely focus on optimizing these techniques to make full parameter unlearning more scalable for real-world applications. Additionally, combining these methods with techniques such as knowledge distillation and selective

 doi:10.20944/preprints202409.2026.v

retraining may offer hybrid approaches that balance computational efficiency with the need for comprehensive unlearning.

Conclusions

The rise of machine unlearning as a critical topic in the field of AI is closely tied to the growing importance of data privacy and regulatory requirements such as GDPR. The challenge of selectively removing specific data from trained machine learning models without retraining from scratch has significant implications for various neural networks, including CNNs, DNNs, and GNNs. Each of these network types presents unique difficulties in implementing machine unlearning due to their distinct architectures and learning mechanisms.

For CNNs, the hierarchical nature of convolutional layers and the distributed representation of learned features make it difficult to isolate the influence of individual data points. Techniques such as fine-tuning and influence functions show promise, but scalability and computational costs remain key challenges. DNNs face similar difficulties due to their interconnected structure, and the issue of catastrophic forgetting compounds the problem. Selective retraining and knowledge distillation offer potential solutions, but further research is needed to strike an effective balance between unlearning and preserving model performance.

In GNNs, the graph-structured data adds another layer of complexity. The influence of individual nodes and edges can propagate across the network, making it difficult to remove specific information without affecting the entire graph. Localization techniques and graph pruning present possible solutions, but these approaches require further exploration to ensure they can be effectively implemented in real-world applications.

Across all three types of neural networks, future research must focus on developing more efficient and scalable machine unlearning techniques. Advances in meta-learning, elastic weight consolidation, and dynamic GNNs may offer new pathways for addressing these challenges. As the need for data privacy continues to grow, the development of reliable and computationally feasible machine unlearning methods will be critical for ensuring compliance with privacy regulations while maintaining the integrity and performance of machine learning models. This area of research holds tremendous potential, not only for regulatory compliance but also for improving the robustness and fairness of AI systems by allowing the removal of biased or harmful data in an efficient manner. Future research in machine unlearning for GNNs is likely to focus on developing more efficient and scalable methods for localizing the influence of specific data points. Additionally, advances in dynamic GNNs, which can adapt to changes in the graph structure over time, may be offered.

References

1. T. Shaik, X. Tao, H. Xie, L. Li, X. Zhu and Q. Li, "Exploring the landscape of machine unlearning: A comprehensive survey and taxonomy," arXiv preprint arXiv:2305.06360, 2023.
2. N. Li, C. Zhou, Y. Gao, H. Chen, A. Fu, Z. Zhang and Y. Shui, "Machine Unlearning: Taxonomy, Metrics, Applications, Challenges, and Prospects," arXiv preprint arXiv:2403.08254, 2024.
3. E. Ullah, T. Mai, A. Rao, R. A. Rossi and R. Arora, "Machine unlearning via algorithmic stability," in Conference on Learning Theory, 2021.
4. L. Bourtoule, V. Chandrasekaran, C. A. Choquette-Choo, H. Jia, A. Travers, B. Zhang, D. Lie and N. Papernot, "Machine unlearning," in 2021 IEEE Symposium on Security and Privacy (SP), 2021.
5. Z. Wang, Y. Zhu, Z. Li, Z. Wang, H. Qin and X. Liu, "Graph neural network recommendation system for football formation," Applied Science and Biotechnology Journal for Advanced Research, vol. 3, p. 33–39, 2024.
6. M. Chen, Z. Zhang, T. Wang, M. Backes, M. Humbert and Y. Zhang, "Graph unlearning," in Proceedings of the 2022 ACM SIGSAC conference on computer and communications security, 2022.
7. Y. Wei, X. Gu, Z. Feng, Z. Li and M. Sun, "Feature Extraction and Model Optimization of Deep Learning in Stock Market Prediction," Journal of Computer Technology and Software, vol. 3, 2024.
8. 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.

 doi:10.20944/preprints202409.2026.v

9. L. Wang, X. Zeng, J. Guo, K.-F. Wong and G. Gottlob, "Selective forgetting: Advancing machine unlearning techniques and evaluation in language models," arXiv preprint arXiv:2402.05813, 2024.
10. K. Gu, M. R. U. Rashid, N. Sultana and S. Mehnaz, "Second-Order Information Matters: Revisiting Machine Unlearning for Large Language Models," arXiv preprint arXiv:2403.10557, 2024.
11. A. Golatkar, A. Achille and S. Soatto, "Eternal sunshine of the spotless net: Selective forgetting in deep networks," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.