CS772 Project Report

Udhav Varma
IIT Kanpur
211120
udhavv21@iitk.ac.in

Deven Joshi
IIT Kanpur
210326
devenaj21@iitk.ac.in

Rishabh Arijeet
IIT Kanpur
210842
rishabh21@iitk.ac.in

Lakshvant Balachandran
IIT Kanpur
210557
lakshvant21@iitk.ac.in

1 Introduction

In our project, we focus on machine unlearning methods and how appropriate use of efficient algorithms can be implemented to increase the accuracy of neural networks. The key objectives and proposals in our seed paper [8] (Repairing neural nets by leaving the right past behind) is as follows:

- Machine learning models often exhibit unexpected failures when deployed due to deficiencies in the training data, such as:
  - Incorrect labels
  - Outliers
  - Selection biases
- The paper proposes a technique to repair a model on-the-fly as new failures are encountered.
- The technique has two main steps:
  1. Identify the training examples (failure causes) that contributed most to the observed failures.
  2. Adapt the model to fix the failures by erasing the memories of the identified failure causes.

In our model of the problem, we assume that the poor performance of a neural network is chiefly caused by poor data, missing labels and incorrect labelling. Evidently, there could be several other reasons why neural networks have a poor performance, but those issues are not dealt with here.

1.1 The problem statement

Under this assumption, the problem statement can be formally described as follows:

> Consider a prediction model $p(y|x, \theta)$ - the probability distribution of output y given an input x. Suppose $p(\theta|D)$ is the posterior distribution given the training data D (which is assumed to be i.i.d). For test inputs x^* we use the PPD to infer the predicted label y^*
>
> $$p(y^*|x^*, D) = \int p(y^*|x^*, \theta)p(\theta|D)d\theta$$
>
> Suppose the set of incorrect predictions from the test set is denoted by the failure set $\mathcal{F} = \{(x_f^{(n)}, y_f^{(n)})\}_{n=1}^{N_f}$. Our goal is to repair the model by removing bad examples (caused by low quality inputs). The plan is to do the model repairment in two steps:

1. **Cause Identification:** We identify a set of poor data points $\mathcal{C}$, which are believed to have contributed to $\mathcal{F}$ the most.
2. **Treatment:** Adapt the model to perform better on the failure set, without affecting its accuracy on the remaining test data (Using continual unlearning).

In the cause identification step, we need to find out how much each subset of training samples is affecting the result of $\mathcal{F}$. An effective measure for this value would be the change in the posterior predictive distribution of $\mathcal{F}$

$$r(\mathcal{C}) = \log p(\mathcal{F}|\mathcal{D} \setminus \mathcal{C}) - \log p(\mathcal{F} \setminus \mathcal{D})$$

1.2 The solution framework

The seed paper proposes the following techniques for cause identification and treatment steps.

1.2.1 Cause Identification

The expression for $r(\mathcal{C})$ cannot be evaluated for all $\mathcal{C}$ due to its prohibitive computational cost. The paper develops an individual score for each data point z using Taylor's approximation, and proposes methods to evaluate the expression.

$$\hat{r}(z) = \mathbb{E}_{p(\theta|\mathcal{D})}[\log p(z|\theta)] - \mathbb{E}_{p(\theta|\mathcal{D},\mathcal{F})}[\log p(z|\theta)]$$

The paper proposes two methods to evaluate this function:

1. Using the point estimates of θ, we can find out the score $\hat{r}(z)$ by a single step of natural gradient descent with an expression similar to the Linear Influence function
$$\hat{r}(z) = -\gamma \nabla_{\hat{\theta}} \log p(\mathcal{F}|\hat{\theta})^T \hat{\mathbb{F}}_{\hat{\theta}} \nabla_{\hat{\theta}} \log p(\mathcal{F}|\hat{\theta})$$
2. The Elastic Weight Consolidation technique, used for continual learning can be adapted here, by finding $\theta_{+\mathcal{F}}$ by maximising the following objective using SGD:
$$-\log p(\mathcal{F}|\theta) - \frac{N}{2}(\theta - \hat{\theta})^T \hat{\mathbb{F}}_{\hat{\theta}}(\theta - \hat{\theta}) - \frac{\lambda}{2}\left\|\theta - \hat{\theta}\right\|_2^2$$
This objective ensures that the θ has high accuracy on $\mathcal{F}$ but does not deviate much from the original point estimate of θ. Now the value of $\hat{r}(z)$ is found as
$$\hat{r}(z) = \log p(z|\hat{\theta}) - \log p(z|\hat{\theta}_{+\mathcal{F}})$$

A small set consisting of the largest few values of $r(z)$ is chosen as the subset $\mathcal{C}$.

1.2.2 Treatment

In this step, we use the current posterior $p(\theta|\mathcal{D})$ as the prior and continually remove the data points $\mathcal{C}$ to obtain the required posterior $p(\theta|\mathcal{D} \setminus \mathcal{C})$. If we use an approximate posterior $q(\theta)$, then the updated posterior will be

$$q_{-\mathcal{C}}(\theta) \propto \frac{q(\theta)}{p(\mathcal{C}|\theta)} \approx p(\theta|\mathcal{D} \setminus \mathcal{C})$$

The paper proposes the following methods to calculate this expression:

1. One simple way would be to learn the new MLE/MAP estimate of $p(\theta|\mathcal{D} \setminus \mathcal{C})$ by using SGD to maximise $\log p(\mathcal{D} \setminus \mathcal{C}|\theta)$ starting from $\hat{\theta}$.
2. The certified removal technique developed in *Guo et al.* can be used which applies a newton step on the model parameters to remove the influence of the deleted data point. We apply newton update removal mechanism by using a single step gradient descent:
$$\hat{\theta}_{-\mathcal{C}} = \hat{\theta} - \gamma \hat{\mathbb{F}}_{\hat{\theta}}^{-1} \nabla_{\hat{\theta}} \log p(\mathcal{C}|\hat{\theta})$$
3. Elastic Weight Consolidation can be used to find the required new parameters $\hat{\theta}_{-\mathcal{C}}$. We maximise the following objective function:
$$-\log p(\mathcal{C}|\theta) - \frac{N}{2}(\theta - \hat{\theta})^T \hat{\mathbb{F}}_{\hat{\theta}}(\theta - \hat{\theta}) - \frac{\lambda}{2}\left\|\theta - \hat{\theta}\right\|_2^2$$
This gives us the required value of $\hat{\theta}_{-\mathcal{C}}$. The first term tries to reduce the influence of $\mathcal{C}$ whereas the remaining two terms form a regulariser, which makes sure θ doesn't deviate too much from the original value.

2 Literature Review

The seed paper describes a fairly robust framework relying on machine unlearning techniques described in other papers, to have a self-correcting neural network model.

The paper by Koh & Liang [6], describes the use of Influence functions to understand a model's prediction by finding out how much each training input contributed to the behaviour at test time. The technique in this paper tries to figure out how much does the prediction change on removal of a particular training data point - that is the change in the parameter θ on removal of some point z - $\hat{\theta}_{-z} - \hat{\theta}$. The method described in this paper first uses a quadratic approximation of the loss function around $\hat{\theta}$ and analyses the change in predictions when the input is pertubed by a small amount. This analysis leads to the formulation of a linear influence function that can be used to attribute a score for each input data, about how much it affects the set $\mathcal{F}$.

The paper "Overcoming catastrophic forgetting in neural networks" [5] describes the Elastic Weight Consolidation method in detail. Here the key objective is to protect previously learned information in a model, while learning new data. EWC calculates the importance of weights of previously learned tasks using the Fisher Information Matrix. When new data is added to the training set, EWC penalises the change in weights of its most important data points, thereby making sure that the core weights of previously learned data are not lost, while learning the new tasks. The paper demonstrates the effectiveness of this technique to prevent "Catastrophic" forgetting over several tasks.

The paper by *Guo et al.* - "Certified Data removal from Machine learning models" [3] describes techniques that can be used to unlearn data in the model, in such a way that there is a theoretical guarantee that the data point is actually removed, and has no influence on the weights of the model after removal. The certified removal mechanism described here applies a newton update step to approximately remove the influence of the deleted data point, followed by random perturbation to hide any residual information of the deleted point. The paper analyses strong theoretical bounds on the error related to this mechanism.

The paper "Continual lifelong learning with neural networks: A review" [7] reviews the continual lifelong learning problem where the system is designed to learn continuously over time without forgetting previous knowledge. The paper reviews the use of regularisation approaches to prevent catastrophic forgetting, where constraints are added to update of neural networks to retain existing knowledge while learning new information. The paper draws inspiration from biological systems, and describes a "memory replay" technique which involves interleaving new experiences with patterns of previous experiences, to prevent forgetting.

The paper "Data shapley: Equitable valuation of data for machine learning." [2] proposes the concept of 'Data Shapley' as a metric to measure each datum's contribution to the performance of the predictive model. The computation of exact Data Shapley value are shown to be very expensive, and approximate methods based on Monte Carlo methods are used for large datasets.

The paper "Machine Unlearning" [1] describes the SISA framework which is used to reduce the computational expense associated with unlearning. The technique described here involves the steps of Sharding, where the training data is partitioned into multiple shards, followed by training the data in isolation from the influence of the other shards, where the data in the shard is introduced to the model incrementally. To find predictions, predictions from each shard are aggregated using a pre defined strategy to provide an overall prediction.

The paper "Retaining Beneficial Information from Detrimental Data for Deep Neural Network Repair" [4] tackles the issue of failure in test environment due to corrupt inputs and noisy labels. The paper describes an energy function method to align detrimental data with clean data to improve the performance of a neural net. The paper proposes an energy function to characterise the data distribution, and then describes an alignment process where the detrimental data is iteratively adapted to minimise the energy function, and align it closer to the clean data. This adapted data is used to repair the neural network, and therby improve its test performance.

3 Our Proposals

Inspired from the contents of the related literature, we have come up with two proposals to improve the framework described in the seed paper:

1. The Energy based model described in [4] aims to correct the detrimental data by not entirely removing it, but selectively aligning the poor data with the clean data. This way, we try to not lose the useful information present in the detrimental data. The energy function is defined as follows
$$E_{\hat{\theta}_{+\mathcal{F}}}(x) = -\log \sum_{y'} \exp\left(f_{\hat{\theta}_{+\mathcal{F}}}(x)\left[y'\right]\right)$$
where f_θ refers to the classification model, parametrised by its weights θ. It is assumed its a K class classification problem, and $f_{\hat{\theta}_{+\mathcal{F}}}(x)\left[y'\right]$ refers to the logit corresponsing to the y' class.

   A given data point z' is aligned with $p_\theta(z)$ to minimise the energy function by using gradient descent as follows
$$z' = z' - \eta \nabla_x E_{\hat{\theta}_{+\mathcal{F}}}(x)$$
The label is aligned using the given expression
$$\tilde{y} = p_{\hat{\theta}_{+\mathcal{F}}}(y \mid \tilde{x}) = \frac{\exp\left(f_{\hat{\theta}_{+\mathcal{F}}}(\tilde{x})\right)}{\sum_{y'} \exp\left(f_{\hat{\theta}_{+\mathcal{F}}}(\tilde{x})\left[y'\right]\right)}$$
Now using the aligned data, the neural net can be updated, thereby repairing the neural net consisting of poor data points. Only the last few layers of the neural network may need to be updated, to fine tune the model with the new derived data.

   We believe that instead of entirely removing the data points from the training set itself, a more careful approach of retaining useful information by aligning detrimental training inputs could give a better performance under a variety of test conditions where a large fraction of training data have noisy labels, but are not entirely useless.

2. A major step in our framework is the identification of the subset of poor data $\mathcal{C}$. The seed paper proposes finding the individual contributions $r(z)$ for each data point z and then pick the largest few values of $r(z)$ as the set $\mathcal{C}$.

   We believe this process can be done in a better manner if the Data Shapley value of the data points are used for identification of the set $\mathcal{C}$. Since, a negative data shapley value indicates that the data point has a detrimental effect on a model, the use of data shapley value does not require optimising on the hyperparameter K - the number of data points to be chosen in the set $\mathcal{C}$. Furthermore, the use of Data Shapley value gives a quantitative estimate of how defective an input is, which can be used to improve our criteria of selection of poor inputs.

4 Experiments

We attempted to replicate the experiment described in the paper on the MNIST dataset, where some of the labels in the training data are randomly flipped.

However due to difficulties in compact representation of approximate posterior of a CNN, and finding its derivative, we have not been able to successfully replicate the experiment, or implement our proposed improvements at the time of writing of this report.

5 Things we learned

We learned the following things as a part of our project:

1. We learnt the use of influence functions to determine the effect of a training data point on the predictive behaviour of a neural network.
2. We learnt the concept of EWC, which uses an objective function designed in such a way that the weights do not deviate much from the old data, when learning new data. This way catastrophic forgetting in neural networks can be minimised.
3. We learnt about certified machine unlearning techniques, that provide a guarantee on the data removal from the model, that is the model is indistinguishable from a model trained without that data point. This has important implications to privacy and user's rights.

4. We learnt about the concept of continual learning, where the model keeps acquiring new data while retaining its previously trained data. We also learnt how inspiration from biological systems are used to design frameworks for machine learning models.
5. We learnt about the machine unlearning technique described in the SISA framework, where partitioning the data leads to an efficient method for unlearning data.
6. We learnt how defective data points can be aligned to the probability distribution of clean data, and how a scalar energy function is used to model the same.
7. Finally, we learnt how continual unlearning of poor data points in a neural network improves the performance of the neural network and reduces the size of its failure set $\mathcal{F}$ at test time.

6 Future Work

Our goals for the future of this project are as follows:

1. First we plan to successfully conduct experiments based on the framework described in the seed paper, we have made progress in this direction.
2. We plan to implement the Energy based model to avoid removing data points and instead repair individual data points to align them with the clean data, and compare the performance with the previous framework, over a variety of data.
3. We will try to incoroporate Data Shapley scores for identification of defective data and run experiments to quantitatively determine the efficacy of this measure.
4. If possible, we will try to build the framework developed in this paper into a library, which can be used easily in a variety of scenarios.

7 Contributions

Each one of us read and understood the seed paper and related papers. The contribution regarding the project deliverables is as follows:

- Udhav Varma - Contributed in presentation slides, and project report.
- Deven Joshi - Contributed in presentation slides, and finding improvements.
- Rishabh Arijeet - Contributed in presentation slides, and finding improvements.
- Lakshvant Balachandran - Presentation voiceover.

References

[1] Lucas Bourtoule et al. *Machine Unlearning*. 2020. arXiv: `1912.03817 [cs.CR]`.

[2] Amirata Ghorbani and James Zou. *Data Shapley: Equitable Valuation of Data for Machine Learning*. 2019. arXiv: `1904.02868 [stat.ML]`.

[3] Chuan Guo et al. *Certified Data Removal from Machine Learning Models*. 2023. arXiv: `1911.03030 [cs.LG]`.

[4] Long-Kai Huang et al. "Retaining Beneficial Information from Detrimental Data for Deep Neural Network Repair". In: *37th Conference on Neural Information Processing Systems (NeurIPS 2023)*. Tencent AI Lab; The University of Texas at Arlington; The Chinese University of Hong Kong. 2023.

[5] James Kirkpatrick et al. "Overcoming catastrophic forgetting in neural networks". In: *Proceedings of the National Academy of Sciences* 114.13 (Mar. 2017), pp. 3521–3526. ISSN: 1091-6490. DOI: `10.1073/pnas.1611835114`. URL: `http://dx.doi.org/10.1073/pnas.1611835114`.

[6] Pang Wei Koh and Percy Liang. *Understanding Black-box Predictions via Influence Functions*. 2020. arXiv: `1703.04730 [stat.ML]`.

[7] German I. Parisi et al. "Continual lifelong learning with neural networks: A review". In: *Neural Networks* 113 (2019), pp. 54–71. ISSN: 0893-6080. DOI: `https://doi.org/10.1016/j.neunet.2019.01.012`. URL: `https://www.sciencedirect.com/science/article/pii/S0893608019300231`.

[8] Ryutaro Tanno et al. *Repairing Neural Networks by Leaving the Right Past Behind*. 2022. arXiv: `2207.04806 [cs.LG]`.