Report date 30/09/2024

Paper report:

**Overview:**

The paper defines a general way to alter the training data after the training without the need for a full retrain of the whole model. Data can be modified or straight up removed, specifically in this case, it focuses on partial data (eg: only some features or one or multiple entries, only some labels of one or multiple entries), traditional instance based unlearning systems replace or delete a whole instance (all the features of a datapoint and its label), in this case it only focuses on what changes: if you need to change one feature of a datapoint containing 1000 features this tool will act only on that feature to change and not on all the datapoint.

The efficacy of these tools has been mathematically proven for strongly convex loss function models (certified unlearning) and empirically proven for other kinds of loss functions (like neural networks).

The tool is efficient since it uses only closed form updates, making it able to scale in a cheap and fast manner.

We are also interested in closed form updates for the thesis.

The whole system is based on influence functions that define the influence of a single datapoint on the final learnt model.

The tool uses perturbation masks, they essentially remove the knowledge gained from the data points we want to delete/update.

UNSURE: the perturbation masks are applied to the model via first order and second order updates (first and second order being the order of the derivative of the loss function) of the loss function. They often mention the invertibility of the Hessian matrix, the need of that is not very clear to me. The hessian and the jacobian matrix of the function are used to determine how to apply the mask on the model. I still need to better understand first and second order updates tho.

**Experimental metrics and scenarios:**

In the experiments they measure:

* Efficacy of unlearning (the unlearning is certified in the strong convex loss functions but not in all the others): how much has the model unlearnt (the use the exposure metric)
* Fidelity of the model: the model performances after the unlearning phase (they use loss and accuracy to measure it)
* Efficiency of the process: measured as the runtime and the number of gradient calculations for each unlearning method on the dataset.

The experiments (scenarios) they work on:

* Unlearning sensitive features:linear model trained with logistic regression on Adults, Diabetes, Spam, Malware dataset. In the unlearning they set a bunch of features to 0 (or meaningless values).
* Unlearning unintended memorization: they try to remove a canary that they inserted in a sentence (the canary os a telephone number), they used difference canary length and repeated it multiple times so more lines (they actually say datapoints, not lines) would be affected.
* Unlearning poisonous samples: they simulate label poisoning where a bunch of labels are flipped. The task is to unflip them and apply the unflipping to the already learnt model.

**Benchmarked against:**

Differential privacy, fine tuning (with dataset with the corrected data), retraining from scratch, sharding.

**Limitations:**

* Changing millions of data points it's too much, it works well on thousands, it works to correct only a small number of datapoints on big models, not big numbers of datapoints on small models.
* Certified unlearning only for strongly convex functions
* The model only unlearns specific data, it does not recognize the training data that should be changed. It repairs privacy leaks, it doesn't detects them (this issue is not an issue because it would be out of scope) imo

**Repo:**

[github.com/alewarne/MachineUnlearning](http://github.com/alewarne/MachineUnlearning)

**Authors:**

**BIFOLD**

Alexander Warnecke et alii

**Year:**

2023