Project Presentation

Joel W. Yuhas, Vedhas S. Patkar

Project Motivations

Find efficient way for deletion of data from ML models

Find any advantages to the Machine Unlearning architecture proposed by Bourtole et. al, over DP-systems and Naïve ML models, for privacy, retraining time or accuracy

Model Architecture - Naïve Model

NEURAL NETWORK WITH NADAM OPTIMIZER

DROPOUT

BATCH SIZE 10, EPOCHS = 200

Model Architecture- DP Model

- Anaconda Environment
- Tensorflow Implementation
- Based from 2 sources
  - Papernot “Machine Learning with Differential Privacy in Tensorflow”
  - Github tensorflow census example
  - http://www.cleverhans.io/privacy/2019/03/26/machine-learning-with-differential-privacy-in-tensorflow.html
  - https://github.com/tensorflow/transform/blob/master/examples/census_example.py

DP Model

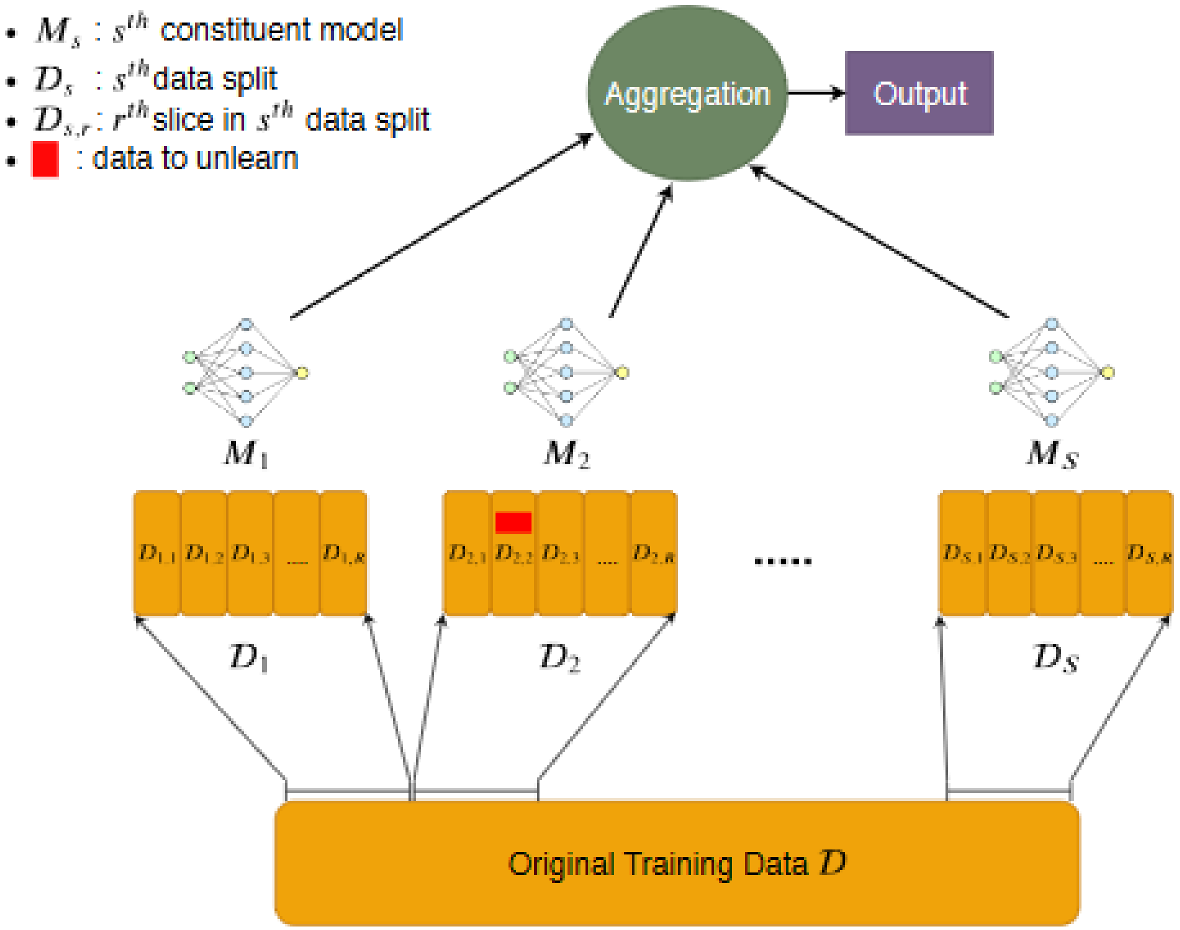
Anaconda Environment

Tensorflow Implementation

- Python version : 3.7.7
- Tensorflow version : 2.1.0
- Tensorflow.compact.v1 package
- Tensorflow_privacy package
- Stocastic Gradient Decent Model
- Full dataset
  - Training :32651 samples
  - Test: 16281 samples

Machine Unlearning Model

- Example
- 3 Shards, 10 Slices
- 1 Deletion Request present in Model 1 (0-based indexing)
- Data present in 8^{th} slice
- Load Checkpoint of model at 7^{th} slice, use that checkpoint and deleted data to retune model for 8^{th} and 9^{th} slice
- Store updated model



Experimental Tasks

Subsample the Census dataset into 2350 records: 2000 training, 350 test

Find accuracy and training times if there are 0, 1, 10, 50, 100, 500, 1000 deletion requests

For DP-model, do step 2 for 3/5 different epsilon values

For Naïve model, do step 2

For MUL model, do step 2 for 1, 5, 10 slices and 5, 10 shards

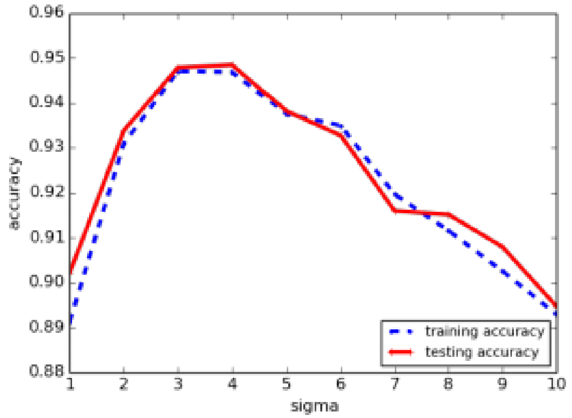
Find Privacy Loss in Naïve and MUL architectures

DP Model Base Test

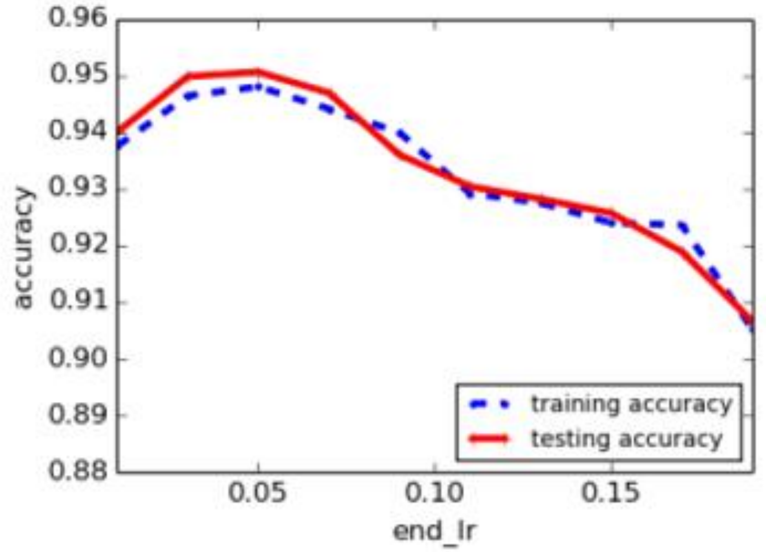
Type	Value
Epoch	100
Learning Rate	0.15
Batch Size	128
Noise Multiplier	1.1

Paper Results for Comparison

- https://arxiv.org/pdf/1607.00133.pdf

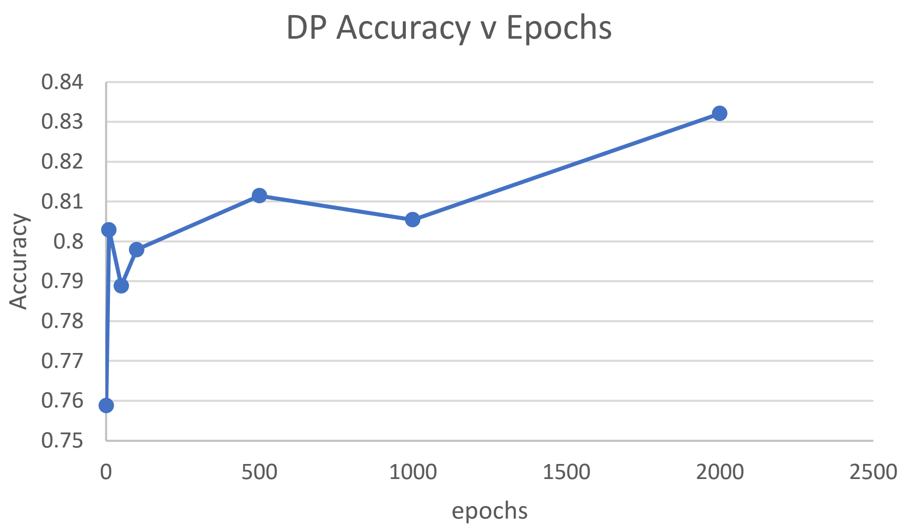


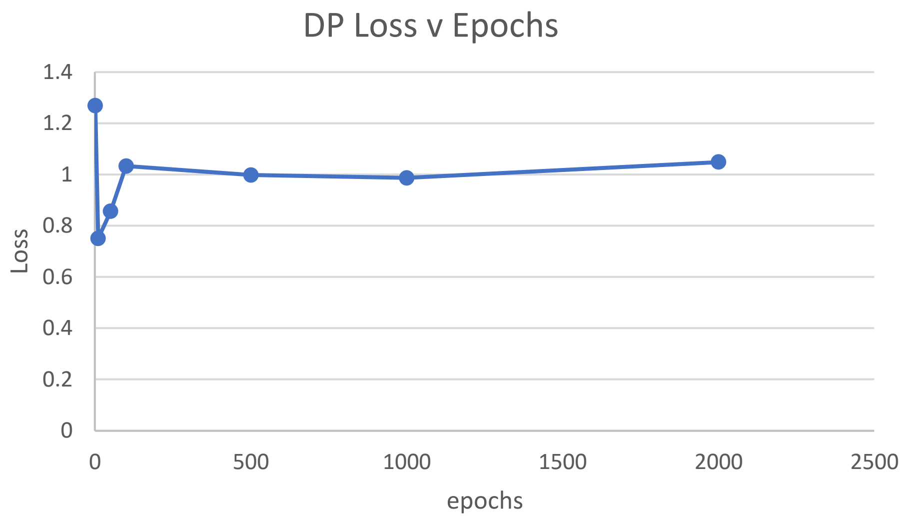
(6) variable noise level



(4) variable learning rate

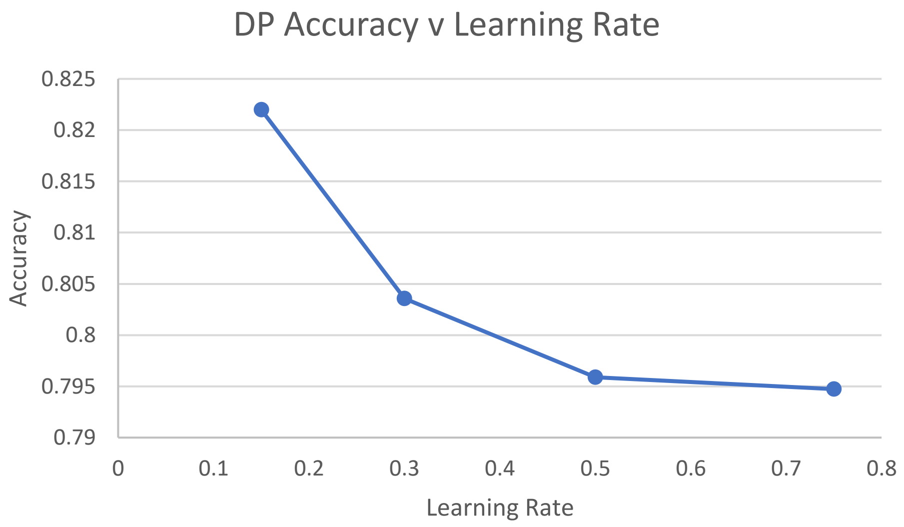
Results of Gradient Descent with DP

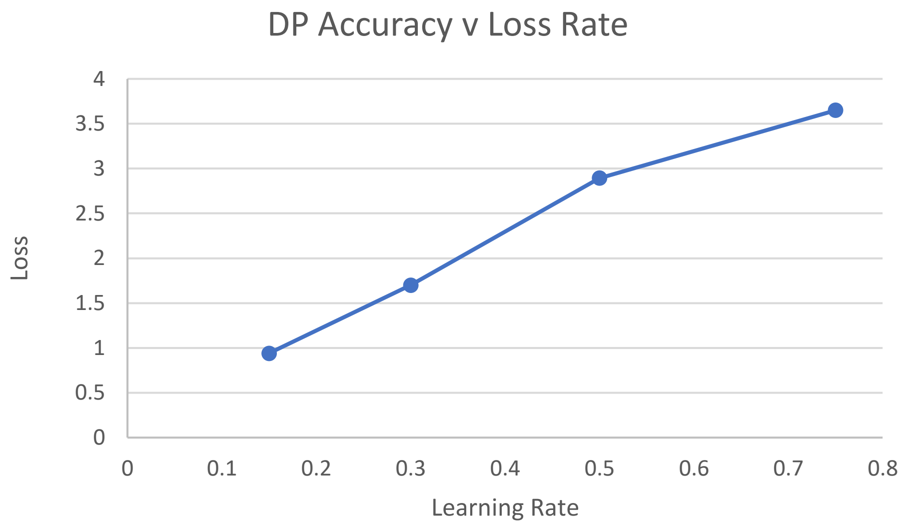




epochs	accuracy	loss
1	0.7587986	1.2678659
10	0.80289906	0.7501584
50	0.7888336	0.8562333
100	0.797924	1.0325124
500	0.81143665	0.9975366
1000	0.80541736	0.98672414
2000	0.83206975	1.0481032

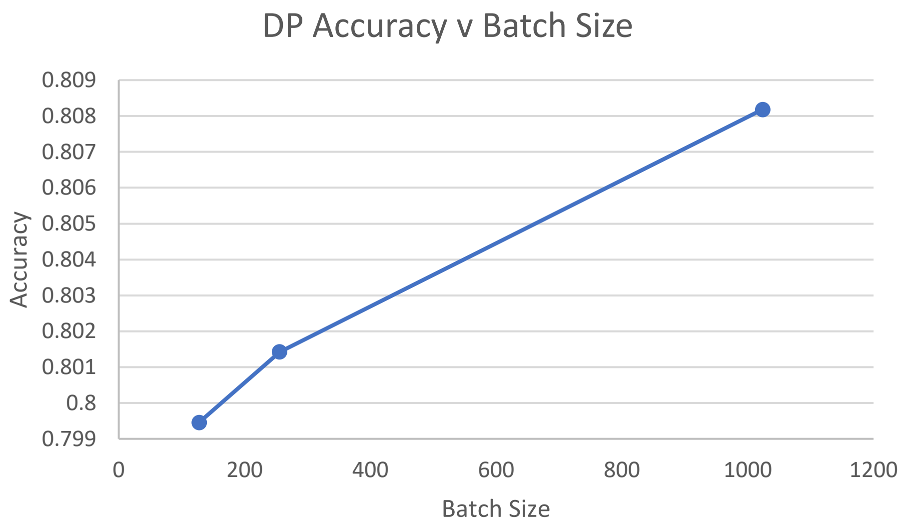
Results of Gradient Descent with DP

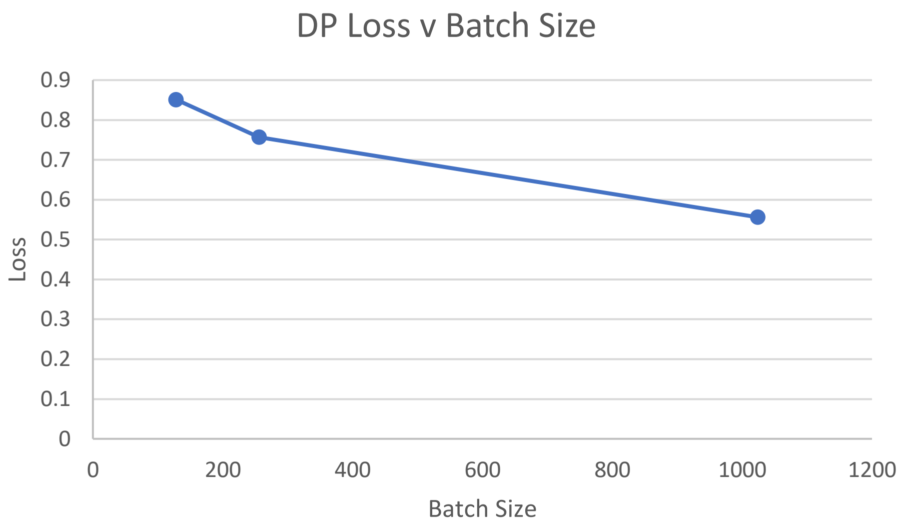




learning rate	accuracy	loss
0.15	0.8220011	0.93863773
0.3	0.80357474	1.6992259
0.5	0.79589707	2.8933535
0.75	0.79473007	3.6492982

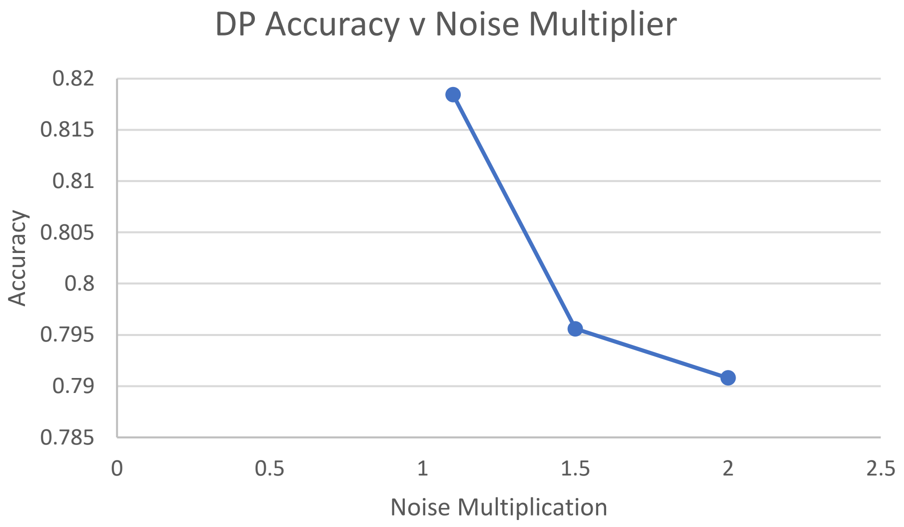
Results of Gradient Descent with DP

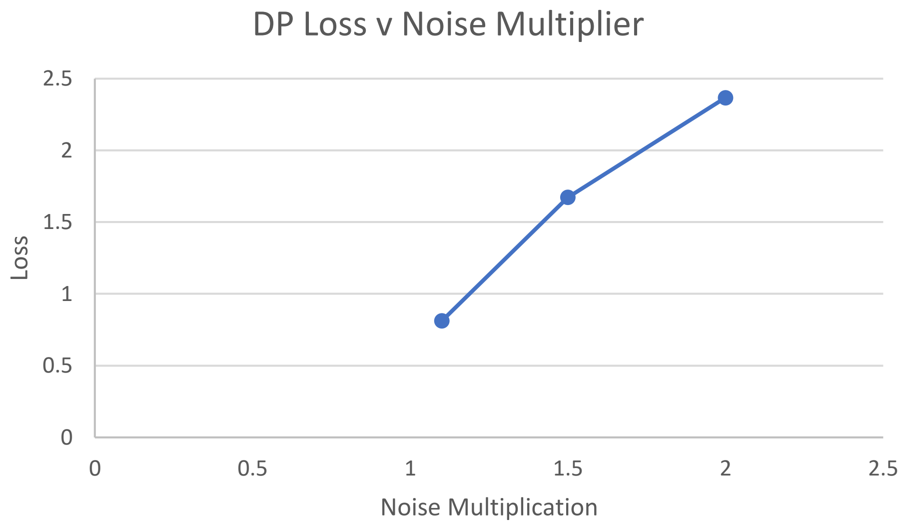




batch size	accuracy	loss
128	0.7994595	0.8515555
256	0.801425	0.75710887
1024	0.8081813	0.55650437

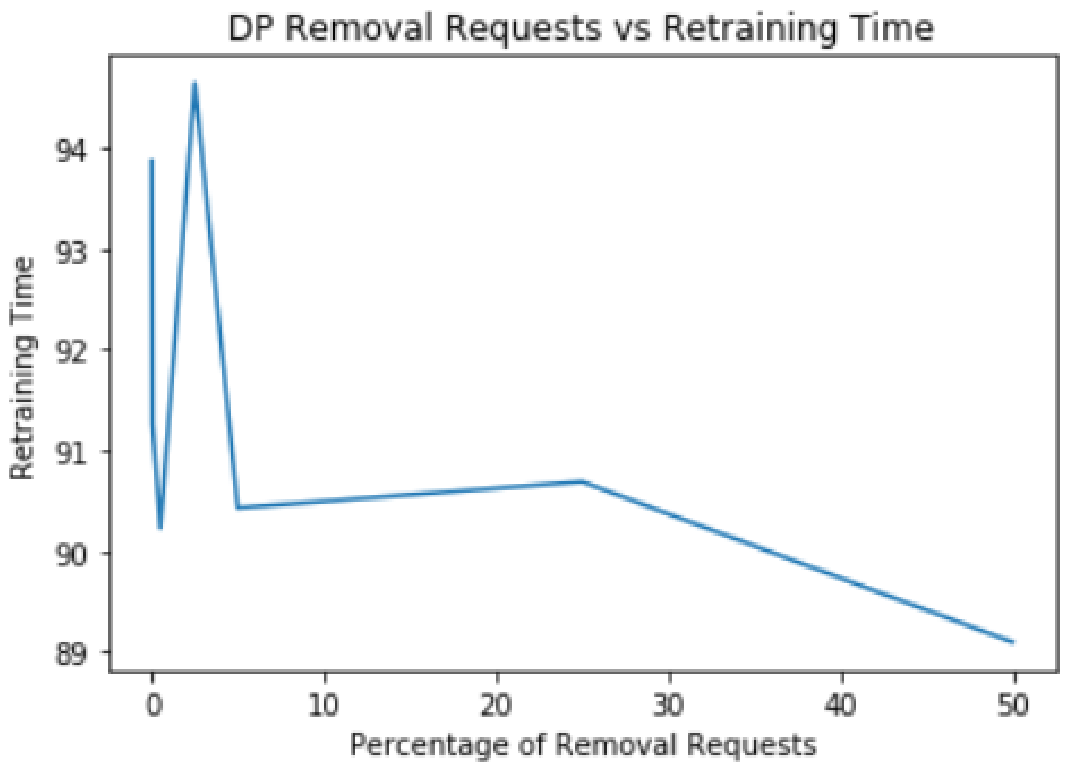
Results of Gradient Descent with DP

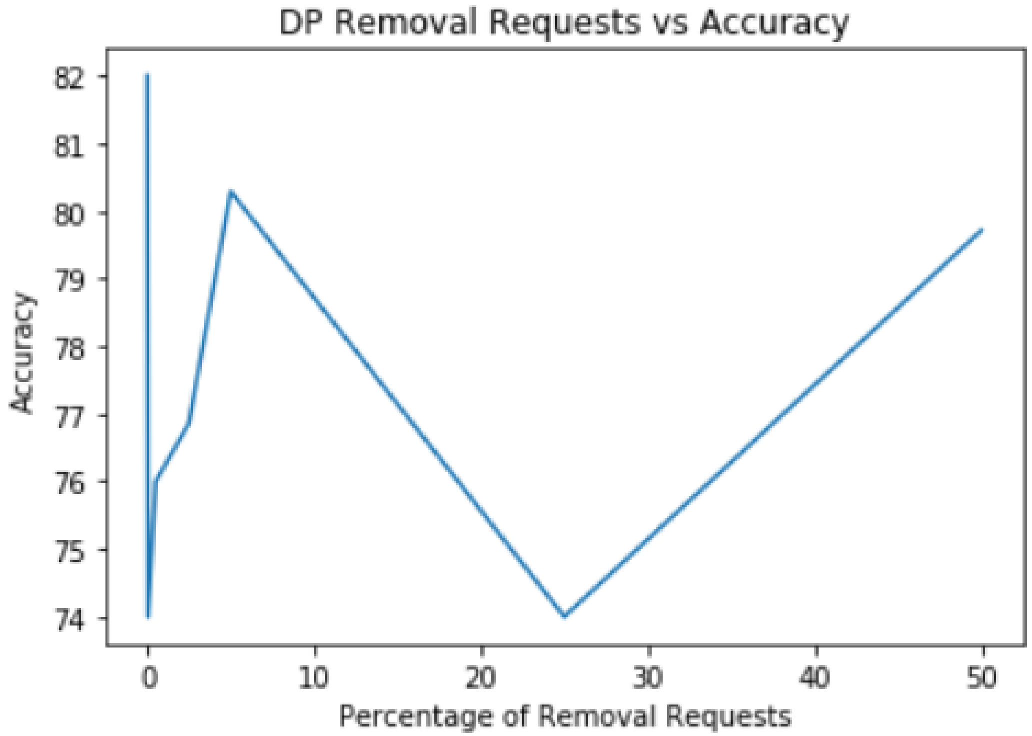




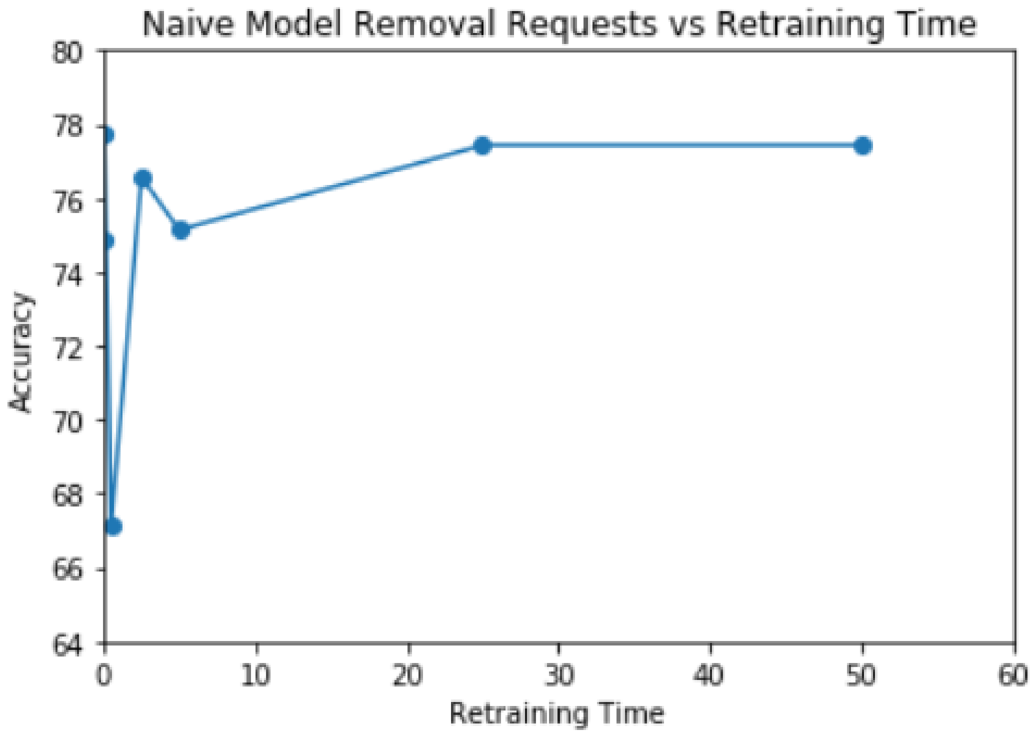
noise multiplier	accuracy	loss
1.1	0.81843865	0.81071043
1.5	0.7955899	1.671327
2	0.7907991	2.366083

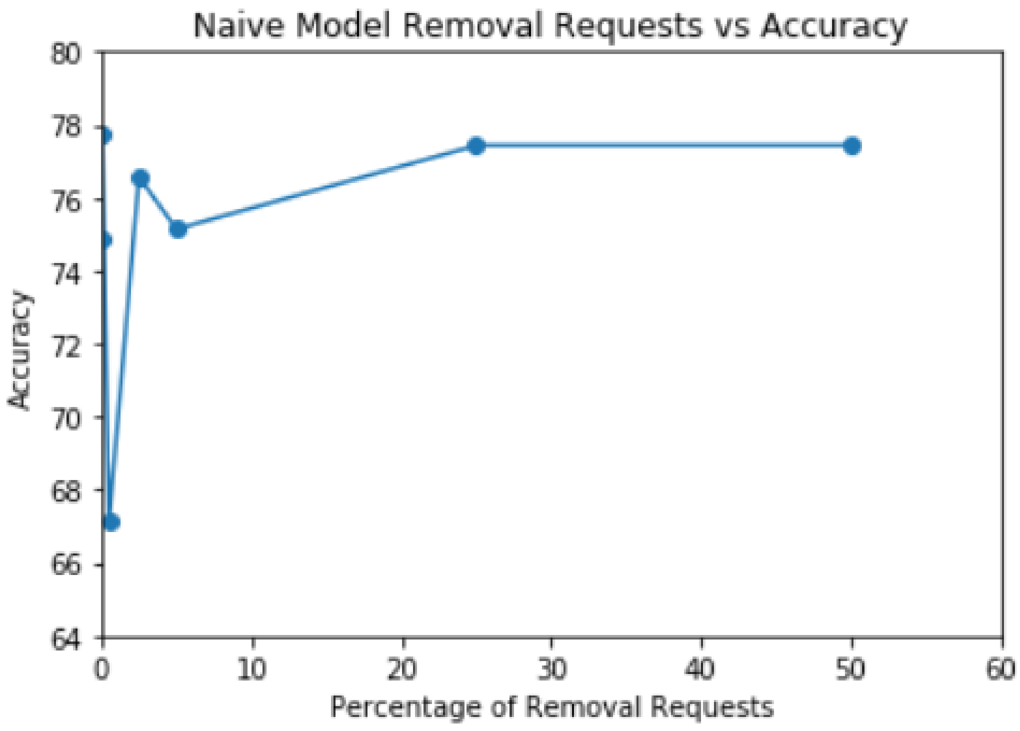
Experimental Results - DP



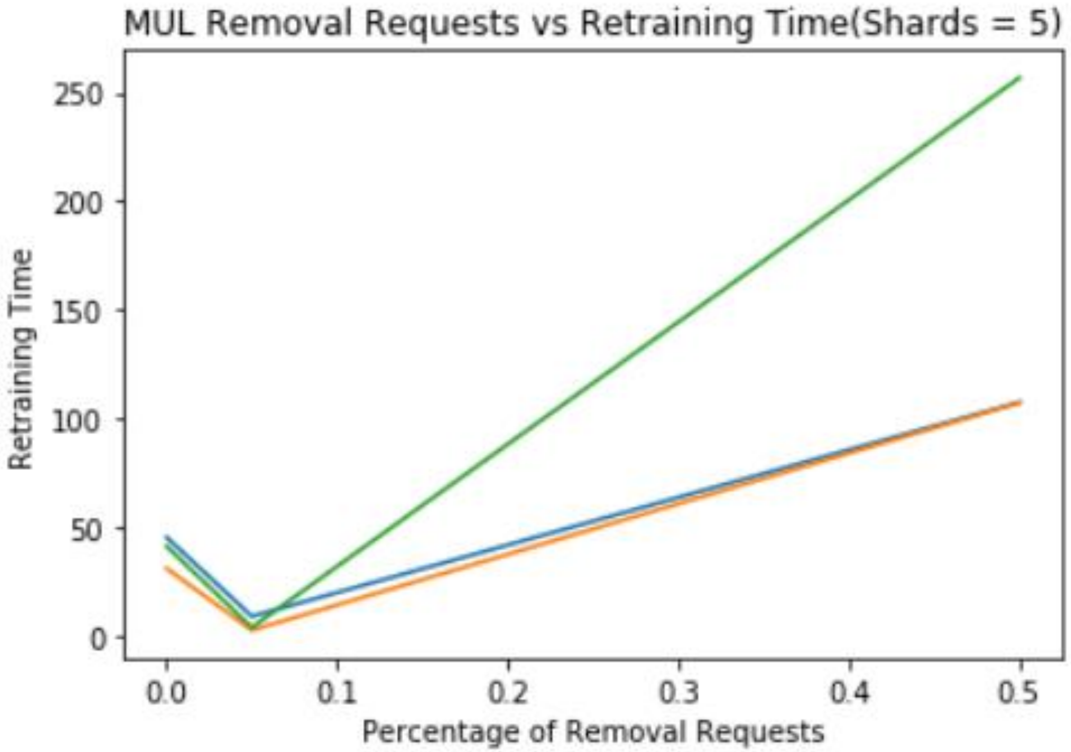


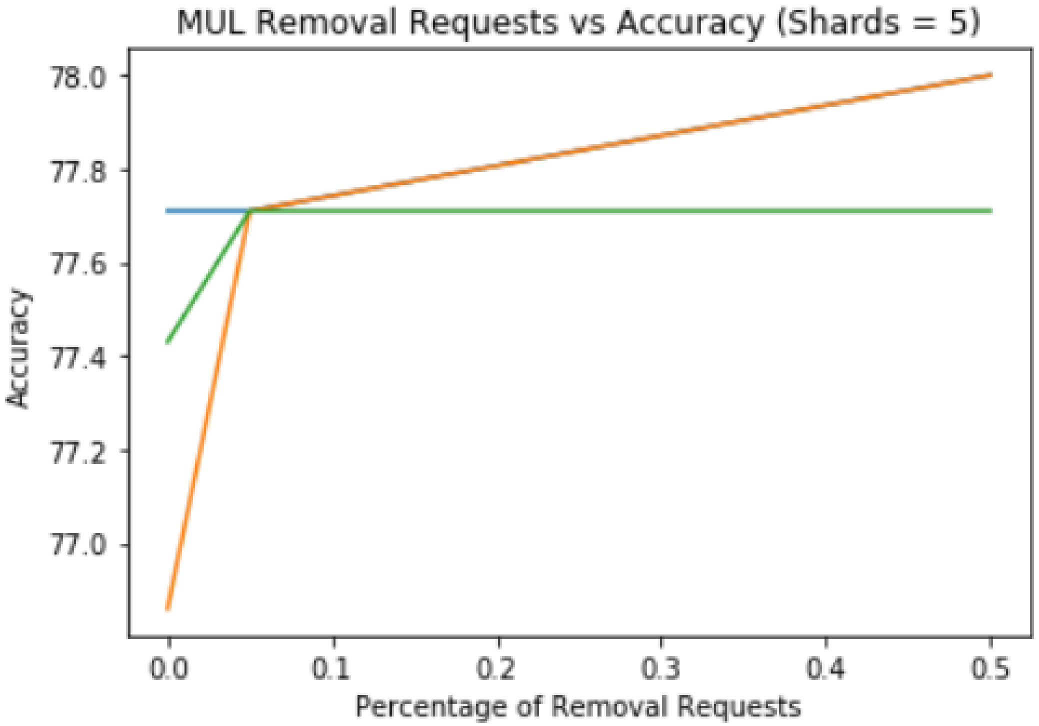
Experimental Results – Naïve



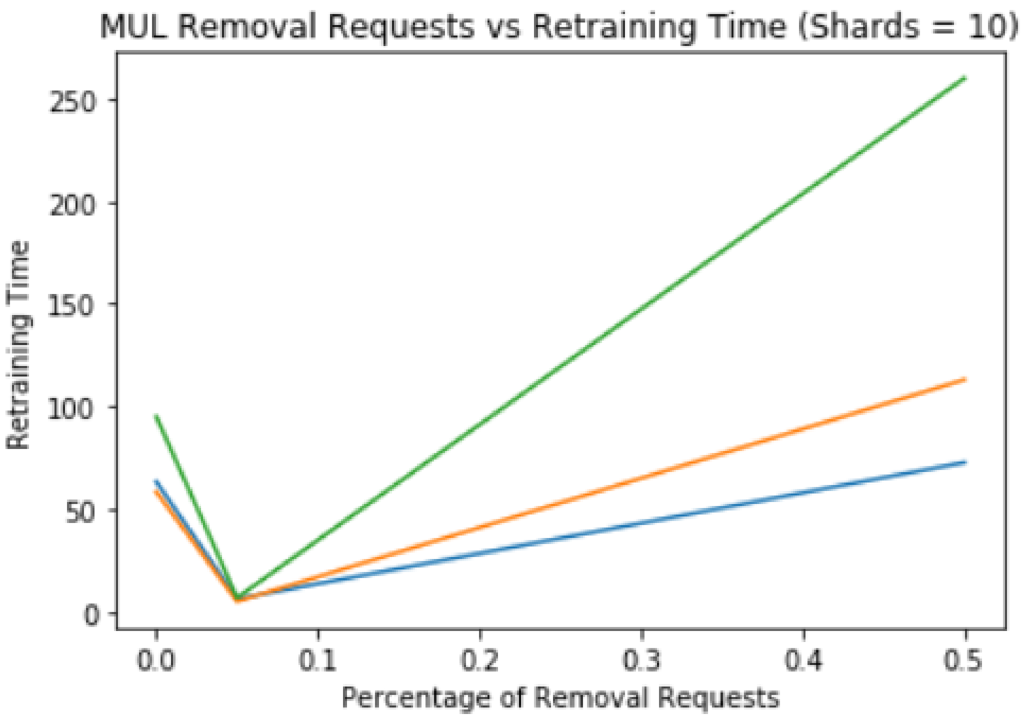


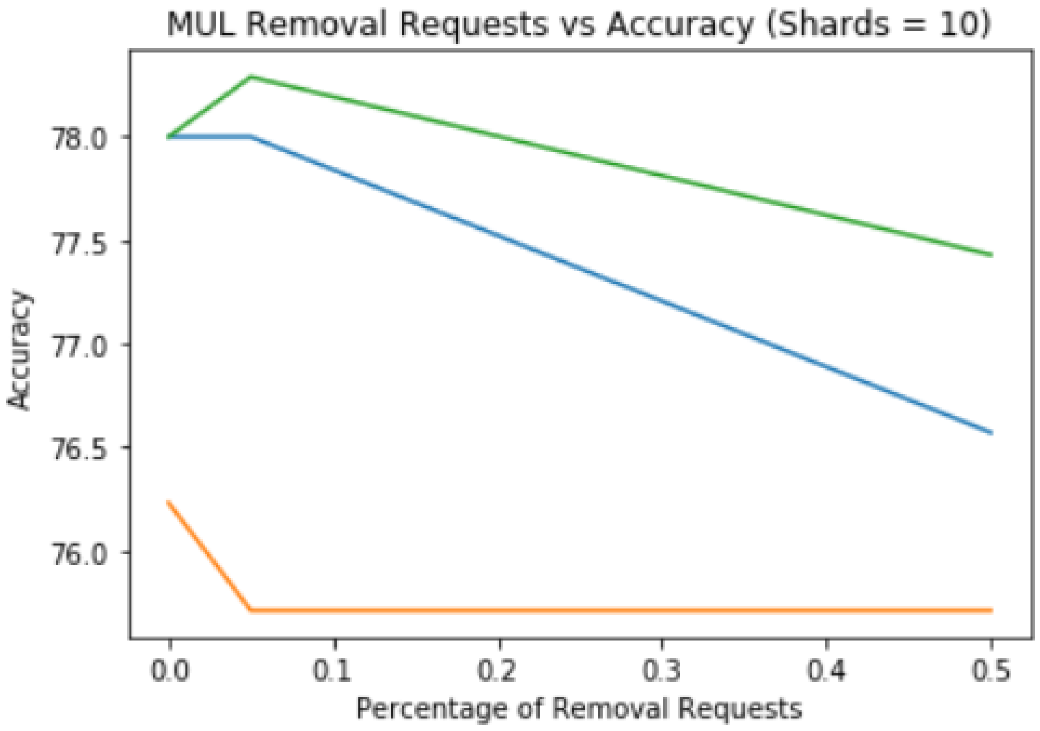
Experimental Results - MUL





Experimental Results - MUL





Findings DP

- Results ended up being similar to expected
  - More epochs= higher accuracy and lower loss
  - Higher learning rate = lower accuracy and higher loss
  - Batch size had inverse effects on DP v Non DP (higher loss Non, lower DP)
  - Noise multiplier drastically degraded performance but increased privacy
  - Issue with epsilon calculation (0.1895)

Findings
Naïve, MUL

- Ginormous cost to implementing MUL – could only run subsampled dataset for 1 and 10 deleted records; authors correctly point this out and use a massive GPU cluster
- Significant improvement time for 1 deleted record for MUL vs Naïve
- Naïve performs better when batch deleted
- Serializing Deletion Requests may give better performance
- For “small” datasets, no significant advantage over a neural network

Summary of Findings

- MUL has no privacy-loss guarantees, just a framework to speedup retraining for deleted requests
- MUL needs a large cluster
- Small Dataset = Advantage Naïve Neural Network
- DP stronger privacy guarantees, large retraining times
- Efficient clustering of deletion requests may improve MUL architecture