# Communication-Efficient Federated Learning for Fine Tuning Large Language Models

Yuankai Cai, Ching Yuan Kung, Hui Chun Mo, Michael Tang, Chengkai Yao, Xiaoyue Zhang

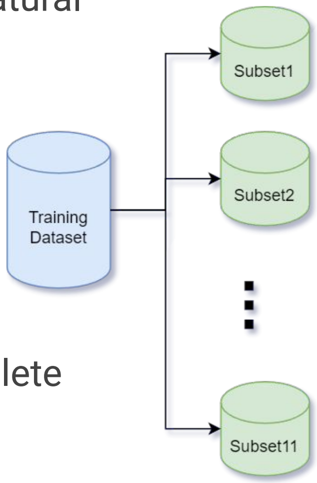
Professor: Ling Liu

CS6220 Demo - Group 17

# Dataset

The ***GLUE CoLA*** (Corpus of Linguistic Acceptability) dataset is used for evaluating natural language understanding systems. It focuses on sentence-level classification tasks.

- Content: Contains sentences labeled as **grammatically correct or incorrect**.
- Size: **Consists of 10657 sentences**, 0.38 MB
- Source: The COLA consists of English acceptability judgments drawn from 23 **books and journal articles** on linguistic theory.



The ***GLUE sst2*** (the Stanford Sentiment Treebank) is a corpus that allows for a complete analysis of the compositional effects of sentiment in language.

- Content: Contains sentences labeled as **positive or negative based on the sentiment of the sentence.**
- Size: **Consists of 70000 sentences**, 7.22 MB
- Source: The sst2 consists of sentences from movie reviews and human annotations of their sentiment.

# Training dataset

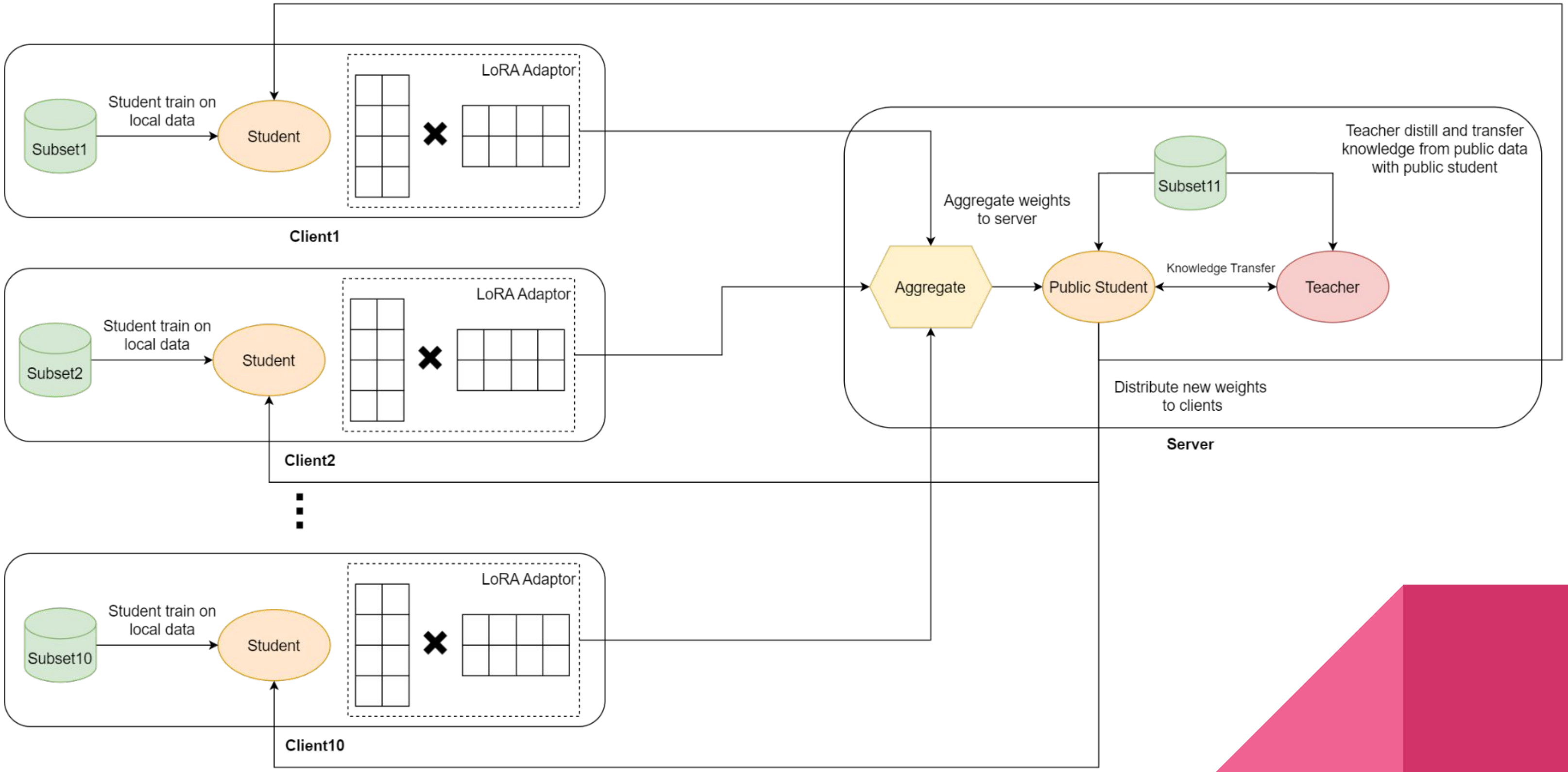
## CoLA training samples

| Data | Label |
|---|---|
| the greatest musicians | 1 |
| cold movie | 0 |
| with his usual intelligence and subtlety | 1 |
| redundant concept | 0 |
| swimming is above all about a young woman 's face , and by casting an actress whose face projects that woman 's doubts and yearnings , it succeeds . | 1 |

## sst2 training samples

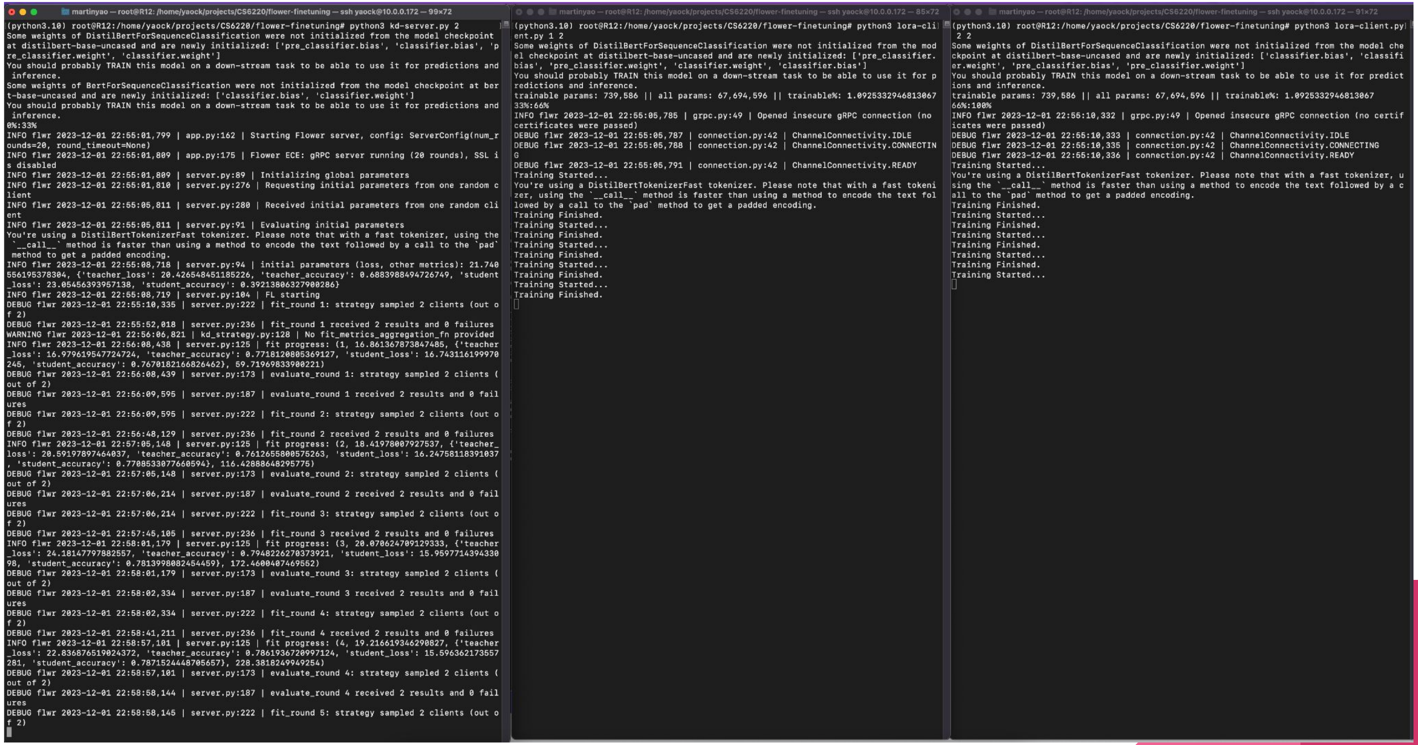
| Data | Label |
|---|---|
| hide new secretions from the parental units | 1 |
| contains no wit , only labored gags | 0 |
| the greatest musicians | 1 |
| with his usual intelligence and subtlety | 1 |
| by far the worst movie of the year | 1 |

# Architecture



LoRA Adaptor
Student train on local data
Subset1
Student
Client1
LoRA Adaptor
Student train on local data
Subset2
Student
Client2
LoRA Adaptor
Student train on local data
Subset10
Student
Client10
Aggregate weights to server
Aggregate
Subset11
Public Student
Knowledge Transfer
Teacher
Teacher distill and transfer knowledge from public data with public student
Distribute new weights to clients
Server

# Screenshot

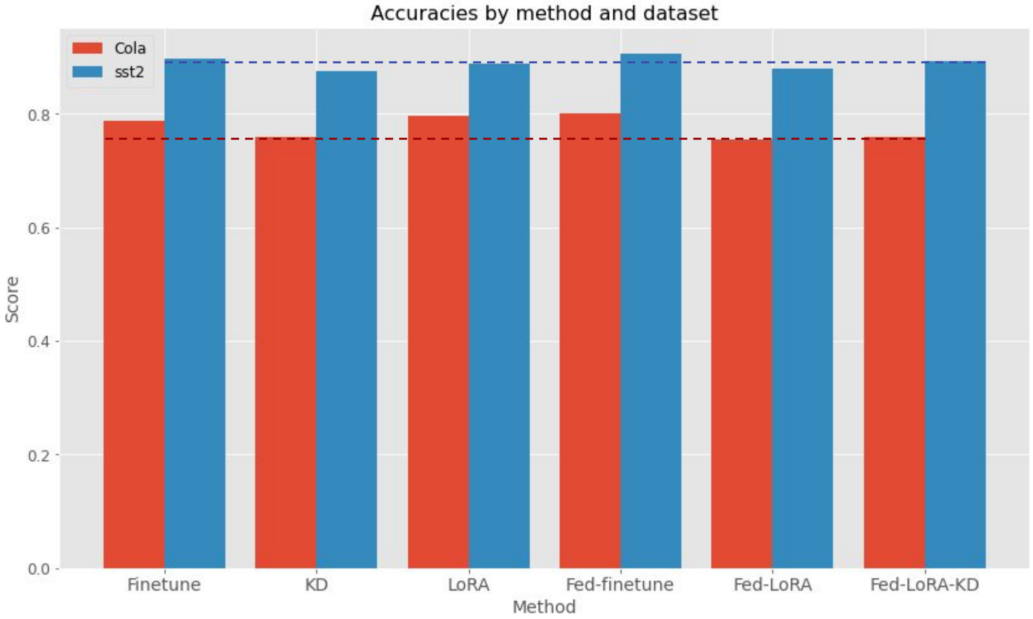

martinyao — root@R12: /home/yaock/projects/CS6220/flower-finetuning — ssh yaock@10.0.0.172 — 99×72
martinyao — root@R12: /home/yaock/projects/CS6220/flower-finetuning — ssh yaock@10.0.0.172 — 85×72
martinyao — root@R12: /home/yaock/projects/CS6220/flower-finetuning — ssh yaock@10.0.0.172 — 91×72

```
(python3.10) root@R12:/home/yaock/projects/CS6220/flower-finetuning# python3 kd-server.py 2
Some weights of DistilBertForSequenceClassification were not initialized from the model checkpoint
at distilbert-base-uncased and are newly initialized: ['pre_classifier.bias', 'classifier.bias', 'p
re_classifier.weight', 'classifier.weight']
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and
 inference.
Some weights of BertForSequenceClassification were not initialized from the model checkpoint at ber
t-base-uncased and are newly initialized: ['classifier.bias', 'classifier.weight']
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and
 inference.
0%:33%
INFO flwr 2023-12-01 22:55:01,799 | app.py:162 | Starting Flower server, config: ServerConfig(num_r
ounds=20, round_timeout=None)
INFO flwr 2023-12-01 22:55:01,809 | app.py:175 | Flower ECE: gRPC server running (20 rounds), SSL i
s disabled
INFO flwr 2023-12-01 22:55:01,809 | server.py:89 | Initializing global parameters
INFO flwr 2023-12-01 22:55:01,810 | server.py:276 | Requesting initial parameters from one random c
lient
INFO flwr 2023-12-01 22:55:05,811 | server.py:280 | Received initial parameters from one random cli
ent
INFO flwr 2023-12-01 22:55:05,811 | server.py:91 | Evaluating initial parameters
You're using a DistilBertTokenizerFast tokenizer. Please note that with a fast tokenizer, using the
 `__call__` method is faster than using a method to encode the text followed by a call to the `pad`
 method to get a padded encoding.
INFO flwr 2023-12-01 22:55:08,718 | server.py:94 | initial parameters (loss, other metrics): 21.740
556195378304, {'teacher_loss': 20.426548451185226, 'teacher_accuracy': 0.6883988494726749, 'student
_loss': 23.05456393957138, 'student_accuracy': 0.39213806327900286}
INFO flwr 2023-12-01 22:55:08,719 | server.py:104 | FL starting
DEBUG flwr 2023-12-01 22:55:10,335 | server.py:222 | fit_round 1: strategy sampled 2 clients (out o
f 2)
DEBUG flwr 2023-12-01 22:55:52,018 | server.py:236 | fit_round 1 received 2 results and 0 failures
WARNING flwr 2023-12-01 22:56:06,821 | kd_strategy.py:128 | No fit_metrics_aggregation_fn provided
INFO flwr 2023-12-01 22:56:08,438 | server.py:125 | fit progress: (1, 16.861367873847485, {'teacher
_loss': 16.979619547724724, 'teacher_accuracy': 0.7718120805369127, 'student_loss': 16.743116199970
245, 'student_accuracy': 0.7670182166826462}, 59.71969833900221)
DEBUG flwr 2023-12-01 22:56:08,439 | server.py:173 | evaluate_round 1: strategy sampled 2 clients (
out of 2)
DEBUG flwr 2023-12-01 22:56:09,595 | server.py:187 | evaluate_round 1 received 2 results and 0 fail
ures
DEBUG flwr 2023-12-01 22:56:09,595 | server.py:222 | fit_round 2: strategy sampled 2 clients (out o
f 2)
DEBUG flwr 2023-12-01 22:56:48,129 | server.py:236 | fit_round 2 received 2 results and 0 failures
INFO flwr 2023-12-01 22:57:05,148 | server.py:125 | fit progress: (2, 18.41978007927537, {'teacher_
loss': 20.59197897464037, 'teacher_accuracy': 0.7612655800575263, 'student_loss': 16.24758118391037
, 'student_accuracy': 0.7708533077660594}, 116.42888648295775)
DEBUG flwr 2023-12-01 22:57:05,148 | server.py:173 | evaluate_round 2: strategy sampled 2 clients (
out of 2)
DEBUG flwr 2023-12-01 22:57:06,214 | server.py:187 | evaluate_round 2 received 2 results and 0 fail
ures
DEBUG flwr 2023-12-01 22:57:06,214 | server.py:222 | fit_round 3: strategy sampled 2 clients (out o
f 2)
DEBUG flwr 2023-12-01 22:57:45,105 | server.py:236 | fit_round 3 received 2 results and 0 failures
INFO flwr 2023-12-01 22:58:01,179 | server.py:125 | fit progress: (3, 20.070624709129333, {'teacher
_loss': 24.18147797882557, 'teacher_accuracy': 0.7948226270373921, 'student_loss': 15.9597714394330
98, 'student_accuracy': 0.7813998082454459}, 172.4600407469552)
DEBUG flwr 2023-12-01 22:58:01,179 | server.py:173 | evaluate_round 3: strategy sampled 2 clients (
out of 2)
DEBUG flwr 2023-12-01 22:58:02,334 | server.py:187 | evaluate_round 3 received 2 results and 0 fail
ures
DEBUG flwr 2023-12-01 22:58:02,334 | server.py:222 | fit_round 4: strategy sampled 2 clients (out o
f 2)
DEBUG flwr 2023-12-01 22:58:41,211 | server.py:236 | fit_round 4 received 2 results and 0 failures
INFO flwr 2023-12-01 22:58:57,101 | server.py:125 | fit progress: (4, 19.216619346290827, {'teacher
_loss': 22.836876519024372, 'teacher_accuracy': 0.7861936720997124, 'student_loss': 15.596362173557
281, 'student_accuracy': 0.7871524448705657}, 228.3818249949254)
DEBUG flwr 2023-12-01 22:58:57,101 | server.py:173 | evaluate_round 4: strategy sampled 2 clients (
out of 2)
DEBUG flwr 2023-12-01 22:58:58,144 | server.py:187 | evaluate_round 4 received 2 results and 0 fail
ures
DEBUG flwr 2023-12-01 22:58:58,145 | server.py:222 | fit_round 5: strategy sampled 2 clients (out o
f 2)
```

```
(python3.10) root@R12:/home/yaock/projects/CS6220/flower-finetuning# python3 lora-cli
ent.py 1 2
Some weights of DistilBertForSequenceClassification were not initialized from the mod
el checkpoint at distilbert-base-uncased and are newly initialized: ['pre_classifier.
bias', 'pre_classifier.weight', 'classifier.weight', 'classifier.bias']
You should probably TRAIN this model on a down-stream task to be able to use it for p
redictions and inference.
trainable params: 739,586 || all params: 67,694,596 || trainable%: 1.0925332946813067
33%:66%
INFO flwr 2023-12-01 22:55:05,785 | grpc.py:49 | Opened insecure gRPC connection (no
certificates were passed)
DEBUG flwr 2023-12-01 22:55:05,787 | connection.py:42 | ChannelConnectivity.IDLE
DEBUG flwr 2023-12-01 22:55:05,788 | connection.py:42 | ChannelConnectivity.CONNECTIN
G
DEBUG flwr 2023-12-01 22:55:05,791 | connection.py:42 | ChannelConnectivity.READY
Training Started...
You're using a DistilBertTokenizerFast tokenizer. Please note that with a fast tokeni
zer, using the `__call__` method is faster than using a method to encode the text fol
lowed by a call to the `pad` method to get a padded encoding.
Training Finished.
Training Started...
Training Finished.
Training Started...
Training Finished.
Training Started...
Training Finished.
Training Started...
Training Finished.
```

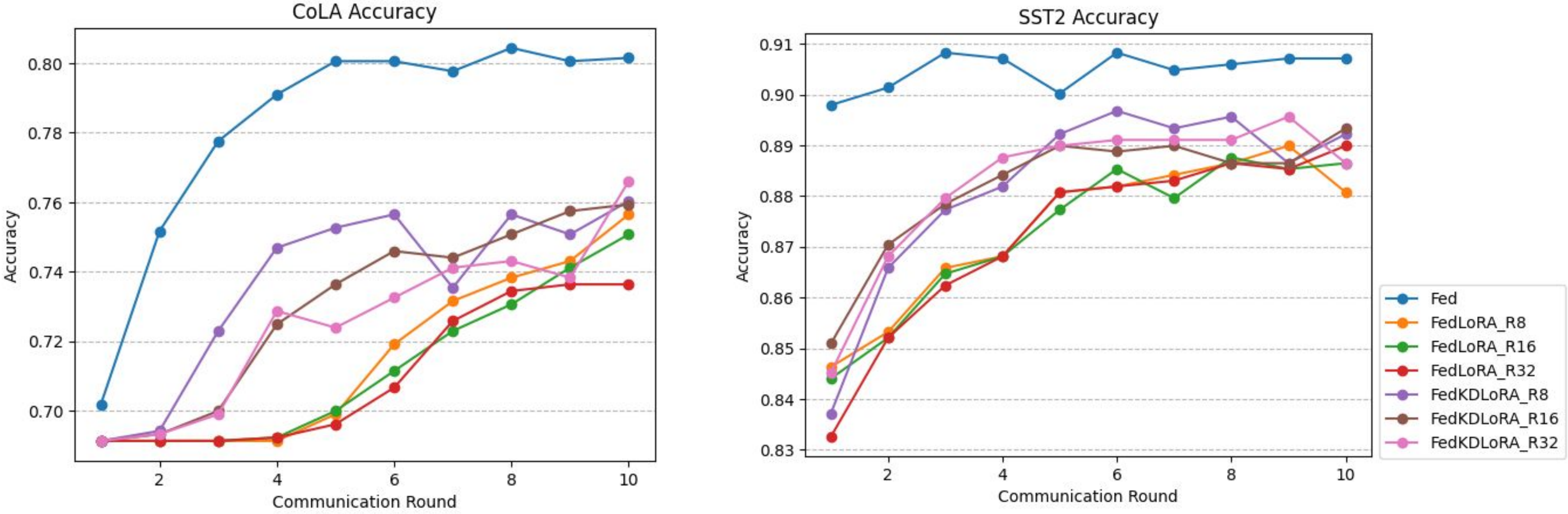
```
(python3.10) root@R12:/home/yaock/projects/CS6220/flower-finetuning# python3 lora-client.py
 2 2
Some weights of DistilBertForSequenceClassification were not initialized from the model che
ckpoint at distilbert-base-uncased and are newly initialized: ['classifier.bias', 'classifi
er.weight', 'pre_classifier.bias', 'pre_classifier.weight']
You should probably TRAIN this model on a down-stream task to be able to use it for predict
ions and inference.
trainable params: 739,586 || all params: 67,694,596 || trainable%: 1.0925332946813067
66%:100%
INFO flwr 2023-12-01 22:55:10,332 | grpc.py:49 | Opened insecure gRPC connection (no certif
icates were passed)
DEBUG flwr 2023-12-01 22:55:10,333 | connection.py:42 | ChannelConnectivity.IDLE
DEBUG flwr 2023-12-01 22:55:10,335 | connection.py:42 | ChannelConnectivity.CONNECTING
DEBUG flwr 2023-12-01 22:55:10,336 | connection.py:42 | ChannelConnectivity.READY
Training Started...
You're using a DistilBertTokenizerFast tokenizer. Please note that with a fast tokenizer, u
sing the `__call__` method is faster than using a method to encode the text followed by a c
all to the `pad` method to get a padded encoding.
Training Finished.
Training Started...
Training Finished.
Training Started...
Training Finished.
Training Started...
Training Finished.
Training Started...
```

# Method Comparison



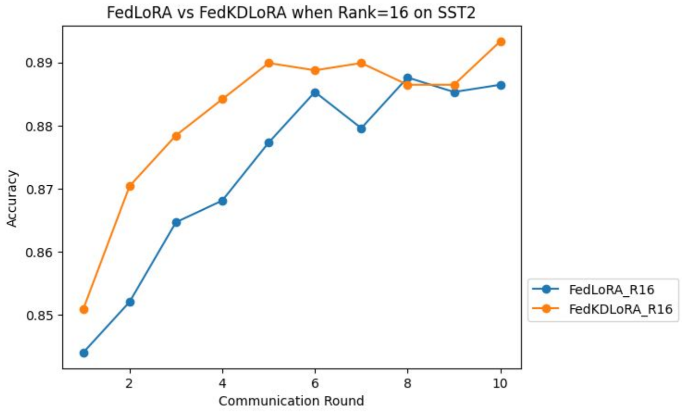
| | Finetune | KD | LoRA | Fed-finetune | Fed-LoRA | Fed-LoRA-KD |
|---|---|---|---|---|---|---|
| cola | 0.787 | 0.760 | 0.797 | 0.802 | 0.756 | 0.760 |
| sst2 | 0.897 | 0.876 | 0.889 | 0.907 | 0.881 | 0.892 |

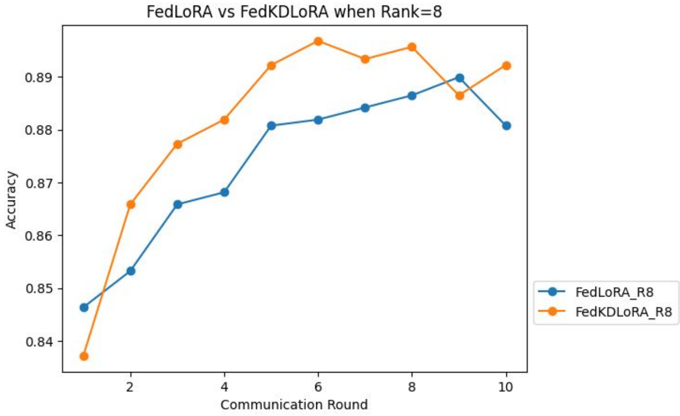
# Individual Model Comparison

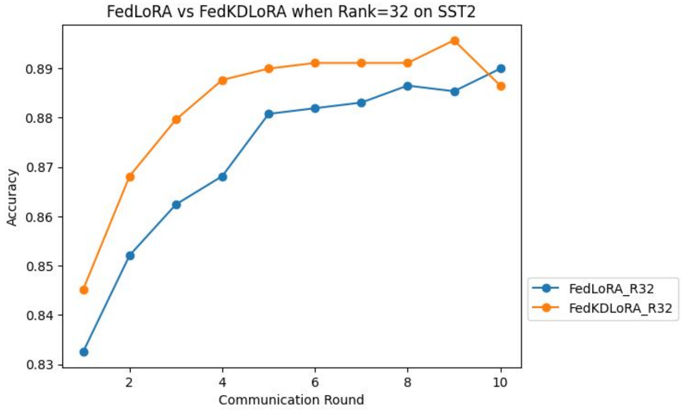


# Easy Case

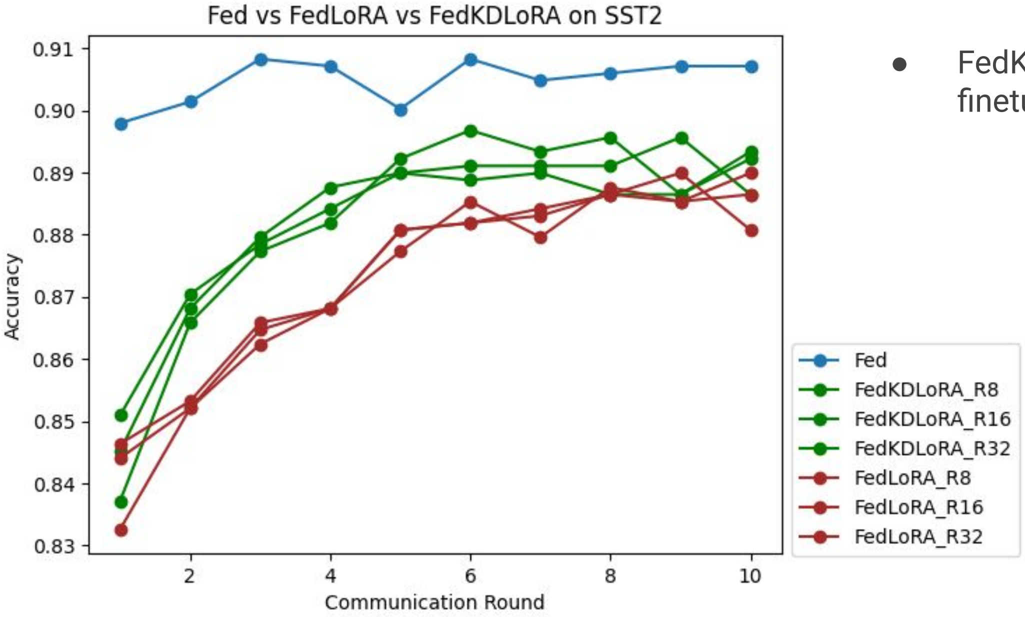
- FedKDLoRA has better accuracy than FedLoRA with different LoRA rankings


FedLoRA vs FedKDLoRA when Rank=16 on SST2


FedLoRA vs FedKDLoRA when Rank=8


FedLoRA vs FedKDLoRA when Rank=32 on SST2

# Hard Case



- FedKDLoRA and FedLoRA underperforms Simple FL finetuning in model accuracy

# Summary

- Combined LoRA (Low Rank Adaptation) and KD (Knowledge Distillation) into Federated Learning
- Compared performance between Fed-LoRA and Fed-LoRA-KD
- Slightly better performance overall

# Reference

Dataset:
https://huggingface.co/datasets/glue/viewer/cola
https://huggingface.co/datasets/glue/viewer/sst2

FedKD:
https://arxiv.org/pdf/2108.13323.pdf
https://github.com/wuch15/FedKD

FedLoRA:
https://arxiv.org/pdf/2310.13283.pdf