

Version Control System for Deep Learning

Professor: Yug Yung Lee

Project Pre-proposal:

Team Details:

Team ID: 6

Name: Dinesh Kumar, Kusam

Class ID: 14

Name: Pradeepika, Kolluru

Class ID: 12

Name: Sindhusa, Tiyyagura

Class ID: 24

Name: Sravan Kumar, Pagadala

Class ID: 21

Project Goal and Objectives:

The main aim of the project is to develop a version control system for

deep learning models(source code, input and output data files, metrics about the experiment).

Project Increment -1

1. Getting familiar with GIT open source code.
2. Understanding the requirements and coming up with the design for the functionality of dlvs commands.
3. Implementing dlvs init, add and commit commands.

Technologies used:

1. Downloaded git, python.
2. Used gdb command

Task -1: Getting familiar with GIT source code.

1. We have downloaded the git source code from the github repository.
2. We have modified few files to understand git init, add and commit functions internal working.
3. We compiled and installed source code files and generated git binary file.
4. We have used gdb command to debug few variables in the C code.

Reasons why GIT is not best for deep learning experiments.

1. There is no functionality in GIT to track group of files as a version (Deep learning is more dependent on the tracking of each experiment as a version)
2. Git is not flexible in handling huge data files more than a GB.
3. There is no tool to track or compare different experiments of the deep learning project.

Existing GIT functionality

1. Git tracks each file history independently.

Most of the git commands are used to change the git meta data files on the local machine.

Only 2 commands(git pull and push commands) are connected to the GIT server for downloading and uploading files respectively using File transfer protocol.

2. GIT init command: Creates .git directory(contains meta data about the files to be tracked) on the project directory.

Example:

.git/

----HEAD

----objects/

----refs/

----===heads/

----===tags/

----config
.gitignore

HEAD specifies to which branch GIT to commit

config contains the settings like repository name, user profiles.

objects directory contains 3 different objects like commit, tree and blob objects. All these files will be stores in the form of hash file_name

3. GIT add command: It creates cache of all the files specified and creates BLOB objects for all the files.

BLOB objects are nothing but the physical files which contains the file content.

The BLOB object is not stored with the file name. It will be stored with the md5_hash value as the file name and with the first two chars as the directory name.

MD5 hash value will be same for same file content files. (different if there is any change in the file contents)

The functionality of MD5 checksum filename is to track different versions of the same file. (in which same file_name cannot be used for tracking different versions).

Example:

```
git add file1.py
```

```
.git/
```

```
-----objects/
```

```
-----info/
```

```
-----92/
```

```
-----f65tbjh784hj345j34hmw4 --> BLOB object
```

contains file1.py contents

It also adds the files to the staging directory which will be later used by commit to commit the files

4. GIT commit command: It creates DAG (Directed acyclic graph) for the list of files committed in the project repository.

In commit command, it starts tracking the files(BLOB objects) stored in the cache.

It creates commit and tree objects.

TREE object basically represents a directory. It references to TREE and BLOB objects.(basically sub directories and files inside the directory)

COMMIT object points out to the TREE object(latest committed version) and the parent COMMIT object(previous COMMIT object).

-----> TREE object <-----

Author: Name

commit time: time

TREE 5437hg4t7854th45y45y9y568

BLOB dg67w45g4ry754hg587th458t

BLOB fg478yt57th58th58t5t8945fb

-----> COMMIT Object <-----

Author: Name

commit Time: time

TREE fgdsyu74htfw38eduhw8e3wd8

Proposed functionality for Deep learning experiments

In GIT it is hard to maintain huge amount of data files. So Database like S3, Azure or GS are used for storing the huge data sets like (source code, input and output files, metrics of the experiment).

In Increment 1 will go over the internal working of the 3 commands (dlv init, add and commit).

1. DLV init command:

This command is similar to the git init command.

It creates .dlv directory which specifies the metadata for the project repository.

.dlv/

-----config

-----cache/

-----HEAD

-----stage

2. DLV add command:

This command is also similar to the git add command which creates BLOB objects for the files to be tracked using DLV.

Example:

dlv add [file1.py](#)

.git/

-----cache/
-----92/
-----f65tbjh784hj345j34hmw4 --> BLOB object
contains [file1.py](#) contents

3. DLV commit command:

This command is little different compared to the git commit command.

In this it creates TREE and COMMIT objects similar to the git functionality but the structure of the COMMIT object is different compared to the git functionality.

-----> TREE object <-----

Author: Name

commit time: time

TREE 5437hg4t7854th45y45y9y568 # represents SUB_DIRECTORY

BLOB dg67w45g4ry754hg587th458t # represents a file

BLOB fg478yt57th58th58t5t8945fb

-----> COMMIT Object <-----

Author: Name

commit Time: time

code: [file1.py](#)

md5: fgf785tg578hg54t8h43wefre34

input: data1.json

md5: df46rgf64h4e8ty58t43efedw43t

output: output_data.json

md5: gf54gt754ht7854h58t54tr5ty5r

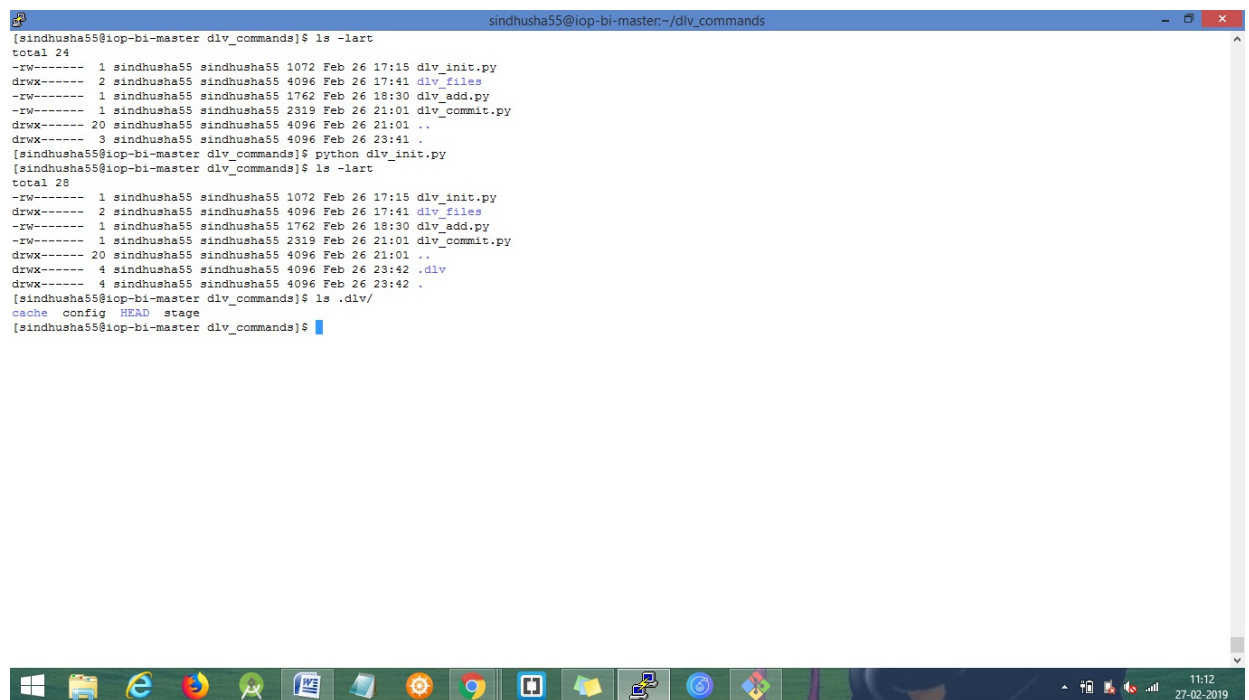
metrics: metrics.json

md5: gf487tg78e5tgf74yty85t5t8y54

commit object contains full details of the experiment run and its versions used in it.

Basic Implementation of the commands:

1. dlvs init command:



```
sindhusha55@iop-bi-master:~/dlvs_commands$ ls -lart
total 24
-rw-r--r-- 1 sindhusha55 sindhusha55 1072 Feb 26 17:15 dlvs_init.py
drwx-r--r-- 2 sindhusha55 sindhusha55 4096 Feb 26 17:41 dlvs_files
-rw-r--r-- 1 sindhusha55 sindhusha55 1762 Feb 26 18:30 dlvs_add.py
-rw-r--r-- 1 sindhusha55 sindhusha55 2319 Feb 26 21:01 dlvs_commit.py
drwx-r--r-- 20 sindhusha55 sindhusha55 4096 Feb 26 21:01 ..
drwx-r--r-- 3 sindhusha55 sindhusha55 4096 Feb 26 23:41 .
[sindhusha55@iop-bi-master:~/dlvs_commands]$ python dlvs_init.py
[sindhusha55@iop-bi-master:~/dlvs_commands]$ ls -lart
total 28
-rw-r--r-- 1 sindhusha55 sindhusha55 1072 Feb 26 17:15 dlvs_init.py
drwx-r--r-- 2 sindhusha55 sindhusha55 4096 Feb 26 17:41 dlvs_files
-rw-r--r-- 1 sindhusha55 sindhusha55 1762 Feb 26 18:30 dlvs_add.py
-rw-r--r-- 1 sindhusha55 sindhusha55 2319 Feb 26 21:01 dlvs_commit.py
drwx-r--r-- 20 sindhusha55 sindhusha55 4096 Feb 26 21:01 ..
drwx-r--r-- 4 sindhusha55 sindhusha55 4096 Feb 26 23:42 .dlvs
drwx-r--r-- 4 sindhusha55 sindhusha55 4096 Feb 26 23:42 .
[sindhusha55@iop-bi-master:~/dlvs_commands]$ ls .dlvs/
cache config HEAD stage
[sindhusha55@iop-bi-master:~/dlvs_commands]$
```

2. dlvs add command:


```
sindhusha55@iop-bi-master:~/div_commands
[sindhusha55@iop-bi-master div_commands]$ ls div_files/
div_add.py  div_init.py
[sindhusha55@iop-bi-master div_commands]$ python div_add.py div_files/
[sindhusha55@iop-bi-master div_commands]$ ls -lart
total 28
-rw-r----- 1 sindhusha55 sindhusha55 1072 Feb 26 17:15 div_init.py
drwx-r----- 2 sindhusha55 sindhusha55 4096 Feb 26 17:41 div_files
-rw-r----- 1 sindhusha55 sindhusha55 1762 Feb 26 18:30 div_add.py
-rw-r----- 1 sindhusha55 sindhusha55 2319 Feb 26 21:01 div_commit.py
drwx-r----- 20 sindhusha55 sindhusha55 4096 Feb 26 21:01 ..
drwx-r----- 4 sindhusha55 sindhusha55 4096 Feb 26 23:42 .div
drwx-r----- 4 sindhusha55 sindhusha55 4096 Feb 26 23:42 .
[sindhusha55@iop-bi-master div_commands]$ ls .div
cache config HEAD stage
[sindhusha55@iop-bi-master div_commands]$ ls .div/cache/
64 91
[sindhusha55@iop-bi-master div_commands]$ ls .div/cache/64
e2c1e7f408b9b9ba9e2ad9b966e506
[sindhusha55@iop-bi-master div_commands]$ cat .div/cache/64 | head -5
cat: .div/cache/64: Is a directory
[sindhusha55@iop-bi-master div_commands]$ cat .div/cache/64/e2c1e7f408b9b9ba9e2ad9b966e506 | head -5
import os
import sys

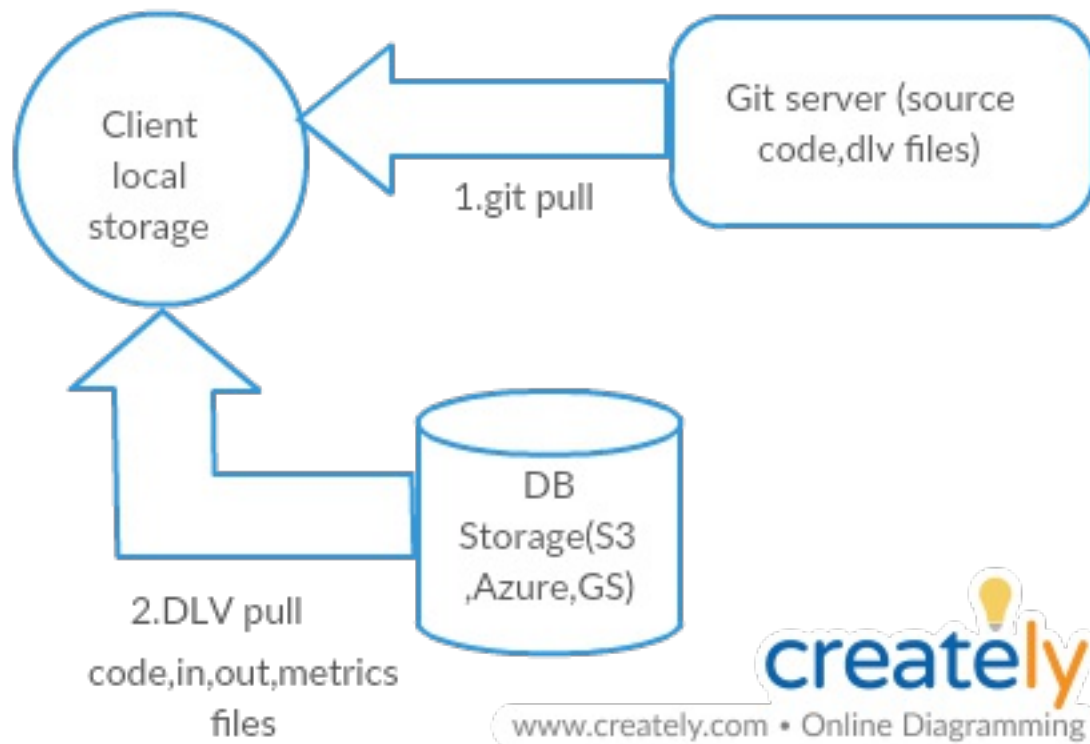
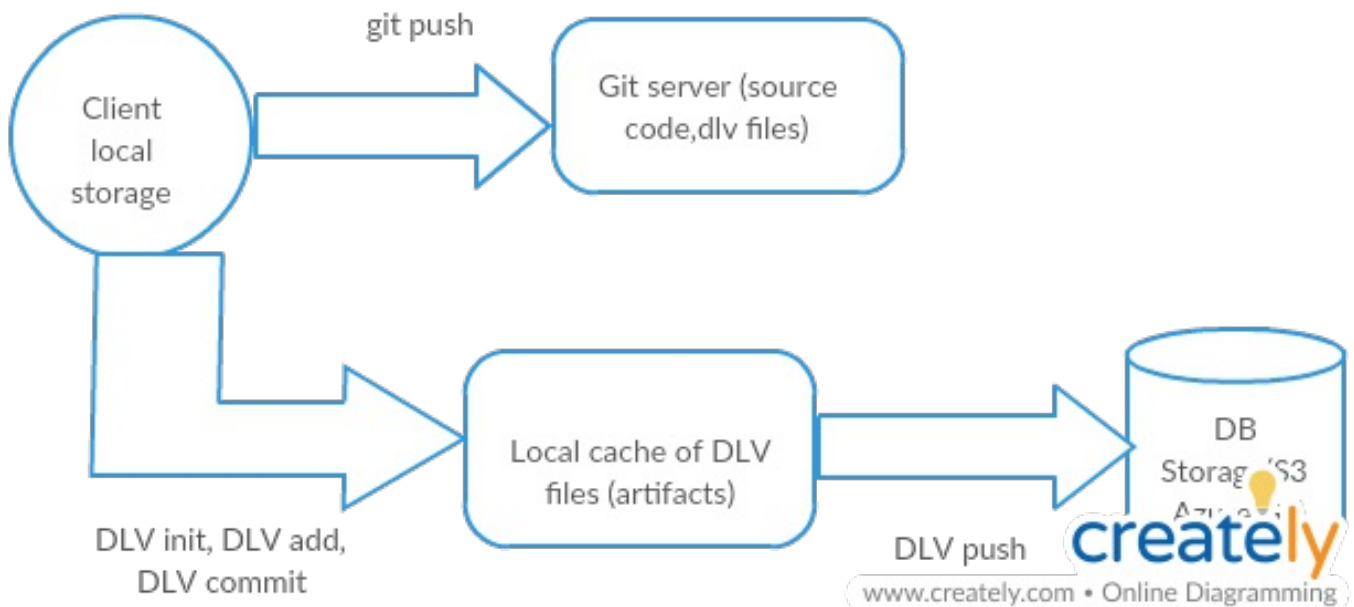
DLV_DIR = ".div"
CONFIG_FILE = "config"
[sindhusha55@iop-bi-master div_commands]$ cat .div/cache/91/96061ad53ea5c7db19bf2787042b5b | head -5
import os
import sys
import hashlib

DLV_DIR = ".div"
[sindhusha55@iop-bi-master div_commands]$
```

3. div commit commad:

```
sindhusha55@iop-bi-master:~/div_commands
[sindhusha55@iop-bi-master div_commands]$ python div_commit.py
[sindhusha55@iop-bi-master div_commands]$ ls -lart
total 28
-rw-r----- 1 sindhusha55 sindhusha55 1072 Feb 26 17:15 div_init.py
drwx-r----- 2 sindhusha55 sindhusha55 4096 Feb 26 17:41 div_files
-rw-r----- 1 sindhusha55 sindhusha55 1762 Feb 26 18:30 div_add.py
-rw-r----- 1 sindhusha55 sindhusha55 2319 Feb 26 21:01 div_commit.py
drwx-r----- 20 sindhusha55 sindhusha55 4096 Feb 26 21:01 ..
drwx-r----- 4 sindhusha55 sindhusha55 4096 Feb 26 23:42 .div
drwx-r----- 4 sindhusha55 sindhusha55 4096 Feb 26 23:42 .
[sindhusha55@iop-bi-master div_commands]$ ls .div/cache/
2d 30 64 7e 91 96
[sindhusha55@iop-bi-master div_commands]$ cat .div/cache/2d/134fa22baa740413441f1c10ac015c
tree: 96cfc2229bc513d972dfb99d2ea31a4b
[sindhusha55@iop-bi-master div_commands]$ cat .div/cache/30/9ec63ba23492605f72eb8086b4eaa9
tree: 7ec6f8c050b34e30b255897204c840d6
[sindhusha55@iop-bi-master div_commands]$ cat .div/cache/7e/c6f8c050b34e30b255897204c840d6
file: /disk2/home/sindhusha55/div_commands/div_files/div_add.py
md5: 9196061ad53ea5c7db19bf2787042b5b
[sindhusha55@iop-bi-master div_commands]$
```

UML Data Flow Diagram:



Tasks done by each Team member:

1. Getting familiar with GIT source code and internal functionality
----- Explored and discussed by all team members
2. Understanding the requirements and faults in the existing project(GIT) and coming up with the design for the proposed project.

----- Sindhusa Tiyyagura and discussed with all team members for the final conclusion.

3. Implementing dlvc init command:

----- Sravan Pagadala

4. Implementing dlvc add command:

----- Pradeepika kolluru

5. Implementing dlvc commit command:

----- Dinesh Kumar Reddy Kusam

6. UML diagrams:

----- Pradeepika kolluru

7. Increment 1 Documentation and video:

----- Sindhusa Tiyyagura