End-To-End walk through and explanation of trainMove.sh

Produced by: Alan Feria & Dr. Alexander White '22

alan214@ucsb.edu

Table of Contents

1 The Unix Shell 2
1.1 A Quick Breakdown of Unix . . . 2

2 Reorganizing files 3
2.1 Changing Working directory . . . 3
2.2 Making a directory for each class . . . 3
2.3 Moving & Renaming Files Recursively . . . 4
2.4 Understanding Our Work . . . 4

3 Data Scientist Tool Box - Unix Shell 5
3.1 cat Unix Shell Command . . . 5
3.2 mkdir Unix Shell Command . . . 5
3.3 mv Unix Shell command . . . 5
3.4 echo Unix Shell Command . . . 5
3.5 cd Unix Shell Command . . . 6

4 trainMove.sh 7
4.1 Source Code . . . 7

About This File

This file was created for the benefit of all teachers and students wanting to use the work located on (link to SCI github)

SECTION 1

The Unix Shell

1.1 A Quick Breakdown of Unix

Bash is a scripting language used to interact with an operating system and do terminal-based tasks, but it can basically do anything a simple programming language can do; it's like C but for the terminal. A shell script utilized Unix commands combined with the ability to declare relatively simple variables (primarily strings). In our case, we will use Unix Shell scripting to automate the execution of repetitive tasks so that we (humans) do not need to perform them individually. Unix Shell scripting is a great way to automate different types of tasks in a system. Bash scripting also supports variables, conditional statements, and loops like programming languages. In Section 2, I will give you a breakdown of a simple script that saved me hours of work by implementing a simple nested for-loop.

2

SECTION

Reorganizing files

In this section, we will be giving a logical breakdown of what is going on when executing trainMove.sh. Since this is a shell script we can reference the Section 3, Data Scientist Tool Box - Unix Shell, in case there are any questions regarding what each function does. We will also be referencing the .txt files located in the SCI_skunks/text_csv_files/ folder.

2.1 Changing Working directory

When working in the Unix Shell, it is highly important always to double-check that you are within the correct working directory because all the work created will be saved to the current working directory unless you specify an absolute path where you want your work to be saved. For this reason, we start the executable by changing my working directory to the parent directory of the files folder which I downloaded from here.

```
cd /Users/alanferia/Downloads/images/
```

2.2 Making a directory for each class

Since we are in the correct parent directory, we need to use mkdir <directory Name> to create the directory of interest. This is much simpler and foolproof than having to specify the absolute path + directory name, as it avoids possible human error. Using this method, we will create a directory for the following classes: Bird, SKunk, Empty, Fox, Human, Other, & Rodent.

2.3 Moving & Renaming Files Recursively

To move and rename files, we must utilize a nested for-loop. It is important to rename the files using a unique name because the current schema would cause us to overwrite and delete many of the files as they share the same name. The first for-loop will utilize <class>Path.txt and extract one element at a time using the cat function and store that value as a file, this value will be the relative path from our current working directory (file parent directory) to where the file is located. This path may look something like this.

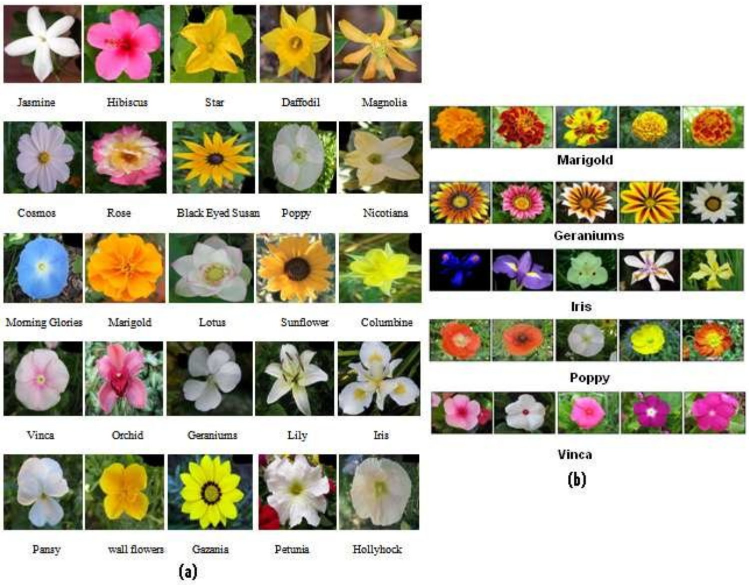
```
loc-h500hh06211646/002/356.jpg
```

Now that we have the relative path to where the file of interest is, we use another for-loop to extract the new file name, which will replace 365.jpg in the example shown above. For this second for-loop, we will use cat on <class>id.txt to extract one element at a time, <class>id.txt contains the new file name. Now that we have grabbed both the file location and new file name, all that is left to do is utilize the mv command to move the file over to its respective folder. **It is important to note that both the outer and inner for-loop have the same index value at each step, so no files will get incorrectly placed or renamed using this method.**

These steps will be repeated for all classes present throughout the data package.

2.4 Understanding Our Work

In summary, trainMove.sh is an executable shell script that moves us to the correct parent directory and subsequently creates a series of directories within the same directory. Next, it will loop over <class>path.txt to find a file and recursively rename that file with the unique ID found in <class>id.txt. Moving these files will result in the reorganization of files following a a similar schema as what is shown below, with (A) being the input format (no specific format) and (B) being the output format of the files (images sorted into folders by category).



3

SECTION

Data Scientist Tool Box - Unix Shell

3.1 cat Unix Shell Command

Cat(concatenate) command is very frequently used in Linux. It reads data from the file and gives its content as output. It helps us to create, view, and concatenate files.

3.2 mkdir Unix Shell Command

mkdir command in Linux allows the user to create directories. This command can create multiple directories at once and set the permissions for the ones being created.

3.3 mv Unix Shell command

mv stands for move. mv is used to move one or more files or directories from one place to another in a file system like UNIX. It has two distinct functions:

- It renames a file or folder.
- It moves a group of files to a different directory.

3.4 echo Unix Shell Command

echo command in Linux is used to display a line of text/string passed as an argument. At a junior level, it is good practice to use echo to see what your code will be doing before actually running it.

3.5 cd Unix Shell Command

The cd command can be used to change into a subdirectory, move back into the parent directory, move all the way back to the root directory, or move to any given directory.

SECTION 4

trainMove.sh

4.1 Source Code

```
cd /Users/alanferia/Downloads/images/

mkdir Bird
mkdir Skunk
mkdir Empty
mkdir Fox
mkdir Human
mkdir Other
mkdir Rodent

for file in $(cat
    /Users/alanferia/Downloads/content/fileByCat/birdPath.txt);
do
    for name in $(cat
        /Users/alanferia/Downloads/content/fileByCat/birdID.txt);
    do
        echo mv "${file}"
            /Users/alanferia/Downloads/images/Bird/"${name}".jpg;
    done
done
```

```
for file in $(cat
    /Users/alanferia/Downloads/content/fileByCat/emptyPath.txt);
do
    for name in $(cat
        /Users/alanferia/Downloads/content/fileByCat/emptyID.txt);
    do
        echo mv "${file}"
            /Users/alanferia/Downloads/images/Empty/"${name}".jpg;
    done
done

for file in $(cat
    /Users/alanferia/Downloads/content/fileByCat/foxPath.txt);
do
    for name in $(cat
        /Users/alanferia/Downloads/content/fileByCat/foxID.txt);
    do
        echo mv "${file}"
            /Users/alanferia/Downloads/images/Fox/"${name}".jpg;
    done
done

for file in $(cat
    /Users/alanferia/Downloads/content/fileByCat/humanPath.txt);
do
    for name in $(cat
        /Users/alanferia/Downloads/content/fileByCat/humanID.txt);
    do
        echo mv "${file}"
            /Users/alanferia/Downloads/images/Human/"${name}".jpg;
    done
done

for file in $(cat
    /Users/alanferia/Downloads/content/fileByCat/otherPath.txt);
do
    for name in $(cat
        /Users/alanferia/Downloads/content/fileByCat/otherID.txt);
    do
        echo mv "${file}"
            /Users/alanferia/Downloads/images/Other/"${name}".jpg;
    done
done
```

```
for file in $(cat
    /Users/alanferia/Downloads/content/fileByCat/rodentPath.txt);
do
    for name in $(cat
        /Users/alanferia/Downloads/content/fileByCat/rodentID.txt);
    do
        echo mv "${file}"
            /Users/alanferia/Downloads/images/Rodent/"${name}".jpg;
    done
done

for file in $(cat
    /Users/alanferia/Downloads/content/fileByCat/skunkPath.txt);
do
    for name in $(cat
        /Users/alanferia/Downloads/content/fileByCat/skunkID.txt);
    do
        echo mv "${file}"
            /Users/alanferia/Downloads/images/Skunk/"${name}".jpg;
    done
done
```