Report

A folder has been created and named: rating and an analysis.txt file to store the result values. Then I grouped all the commands into one script file named rating.sh

First command:

echo "Positive count = $(grep -o -i ",positive" IMDB\ Dataset.csv | wc -l)" > analysis.txt

To get the number of repeats of positive and place the number in the variable "Positive count".	"Positive count = $(grep -o -i ",positive" IMDB\ Dataset.csv | wc -l)"
Find a positive word whether written in lowercase or uppercase	grep -o -i ",positive"
Make every word in the line and enumerate the lines	wc -l
Result is directed to the analysis.txt file	> analysis.txt
echo	To print commands

Second command:

echo "Negative count = $(grep -o -i ",negative" IMDB\ Dataset.csv | wc -l)" >> analysis.txt

To get the number of repeats of positive and place the number in the variable "Negative count".	"Negative count = $(grep -o -i ",negative" IMDB\ Dataset.csv | wc -l)"
Find a negative word whether written in lowercase or uppercase	grep -o -i ",negative"
Make every word in the line and enumerate the lines	wc -l
Result is directed to the analysis.txt file	> analysis.txt
echo	To print commands

Third command:

echo "Total Unique Words = $(cat IMDB\ Dataset.csv| sed -e 's/<[^>]*>//g' | tr '[:upper:]' '[:lower:]' | tr -d '[:punct:]' | tr '[:space:]' '[\n*]' | sort | uniq -c | awk -F ' ' '{if($2 ~ /^[A-Za-z0-9]/) print $2}' | wc -l)" >> analysis.txt

To get the unique word count and place the number in the variable "Total Unique Words".	"Total Unique Words = $(cat IMDB\ Dataset.csv| sed -e 's/<[^>]*>//g'	tr '[:upper:]' '[:lower:]'	tr -d '[:punct:]'	tr '[:space:]' '[\n*]'	sort	uniq -c	awk -F ' ' '{if($2 ~ /^[A-Za-z0-9]/) print $2}'	wc -l)"
To get the IMDB\ Dataset.csv file for applied the commands	cat IMDB\ Dataset.csv
The sed command removes the HTML tags in the file	sed -e 's/<[^>]*>//g'
This command, convert uppercase to lowercase	tr '[:upper:]' '[:lower:]'
This command replaces spaces with a new line	tr '[:space:]' '[\n*]'
Delete punctuation marks	tr -d '[:punct:]'
This command for word order	sort
This command prints the unique words with a repeat number	uniq -c
This command prints words that contain letters and numbers using the pattern / ^ [A-Za-z0-9] /	awk -F ' ' '{if($2 ~ /^[A-Za-z0-9]/) print $2}'
Make every word in the line and enumerate the lines	wc -l
Result is directed to the analysis.txt file	> analysis.txt
echo	To print commands