

# CS595 Assignment 3

Jon Robison

September 30, 2013

Q1. Download the 1000 URIs from assignment 2...remove formatting  
Large collection of files included in modified/ directory  
See Appendix A for download program

Q2. Choose a query term...to ten documents...compute TFIDF values for  
the term in each of the 10 documents.

TFIDF	TF	IDF	URI
.0096	.0009	10.7398	themoneyteam.com/
.0279	.0026	10.7398	mashable.com/2013/09/23/floyd-mayweather-manziel/
.0644	.0060	10.7398	instagram.com/p/eqTrAfFCz0/
.0032	.0003	10.7398	www.youtube.com/watch?v=M4EC7l_Dt30
.0461	.0043	10.7398	oohwonder.tumblr.com/post/61861325994/floyd-mayweather
.0397	.0037	10.7398	naija2day.com/2013/09/mayweather-wants-miley-cyrus-as-es
.1116	.0104	10.7398	floydmayweather.com/
.0644	.0060	10.7398	soundcloud.com/ronjii/floyd-mayweather-by-qilla-fang
.0193	.0018	10.7398	paquesuene.net/v1/raymond-y-miguel-se-cura-con-mayweather
.0719	.0067	10.7398	www.vibevixen.com/2013/09/mayweather-wants-miley-cyrus-a

See Appendix B for scripts to produce table

Q3. Now rank the same 10 URIs from question 2, but this time by their  
PageRank.

PageRank	URI
.4	themoneyteam.com
.4	floydmayweather.com
0	mashable.com/2013/09/23/floyd-mayweather-manziel
0	instagram.com/p/eqTrAfFCz0
0	www.youtube.com/watch?v=M4EC7l_Dt30
0	oohwonder.tumblr.com/post/61861325994/floyd-mayweather
0	naija2day.com/2013/09/mayweather-wants-miley-cyrus-as-escort/
0	soundcloud.com/ronjii/floyd-mayweather-by-qilla-fang
0	paquesuene.net/v1/raymond-y-miguel-se-cura-con-mayweather-vs-canelo-pelea-
0	www.vibevixen.com/2013/09/mayweather-wants-miley-cyrus-as-next-escort/

Briefly compare and contrast the rankings produced in questions 2 and 3.

A3: Top TFIDF was floydmayweather.com. This was one of two non-zero scoring options, and I infer TFIDF was heavily weighted in the pagerank calculation. This would result in being calculated as number one. themoneyteam.com on the other hand is relatively low TFIDF, but is a domain in and of itself, thus disproportionately large TFIDF. It is unfortunate the other pages weren't page ranked, but I predict they would have linearly small results.

## Appendix A

Listing 1: Downloads via curl and lynxifies product to produce a more easily searchable/indexable file

```
#!/bin/bash
export INPUT=$1

function downloadInput {
    echo "Processing file $INPUT"
    pushd "original"
    for link in `cat ../$INPUT`; do
        filename=`echo -n "$link" | md5sum | sed 's/[- ]//g`,`
        curl "$link" > "$filename"
        echo "Downloaded link: $link as: $filename"
    done
    popd
}

function lynxLinks {
    pushd "modified"
    for original in `ls ../original`; do
        lynx -source "../original/$original" > "$original"
        echo "Lynxd link: $original"
    done
    popd
}

downloadInput
lynxLinks
rm -rf original
```

## Appendix B

Listing 2: Control runscript to launch other utils

```
#!/bin/bash
export TERM="mayweather"
export TOTAL_MAYWEATHER=11700000
export TOTAL_DOCS=20000000000
export RESULTS_FILE="results"

#slow but whatever. faster than last assignment
function fileToURI() {
    for line in `cat ../q1/links.txt`; do
        md5file=`echo -n $line | md5sum | sed 's/[- ]//g'`
        if [ "$md5file" = "$file" ]; then
            uri=$line
            echo "Matched $file to $line"
        fi
    done
}

rm copied/*
./copyTen ../q1/modified $TERM
rm $RESULTS_FILE
echo "TFIDF TF IDF URI" > $RESULTS_FILE
for file in `ls copied/`; do
    sh tfidf "copied/$file" "$TERM" "$TOTAL_MAYWEATHER" "$TOTAL_DOCS"
    fileToURI
    line=`cat tfidf.result`
    echo "$line $uri" >> $RESULTS_FILE
    rm tfidf.result
    echo "Completed file:$file"
done
```

Listing 3: Script taking arguments to calculate single TFIDF

```
#!/bin/bash
SCALE=4

function getTF() {
    tc=`grep -r $term $htmlFile | wc -l`
    getWC
    tf=`echo "scale=$SCALE; $tc/$wc" | bc -l`
}

function getIDF() {
    idfToLog=`echo "$totalDocs / $totalContainingTerm" | bc -l`
    idf=`echo "scale=$SCALE; 1($idfToLog)/1(2)" | bc -l`
}
```

```

}

function getWC() {
    wc='cat $htmlFile | wc -w'
}

function getTFIDF() {
    tfidf='echo "scale=$SCALE; $idf*$tf" | bc -l'
}

htmlFile=$1
term=$2
totalContainingTerm=$3
totalDocs=$4
getTF
getIDF
getTFIDF
echo "$htmlFile stats for $term: TC=$tc WC=$wc TF=$tf IDF=$idf TFIDF=$tfidf"
echo "$tfidf $tf $idf" > tfidf.result

```

Listing 4: Copied first ten files matching criteria passed in

```

#!/bin/bash
x=0
for file in `grep -rl $2 $1`; do
    cp $file "copied"
    x=`expr $x + 1`
    if (($x >= 10)); then
        echo "Copy complete!"
        exit
    fi
done

```