Web Science cs532: Assignment #9

Due on Thursday, April 21, 2016

Dr.Michael.L.Nelson 4:20pm

Zetan Li

Contents

Problem 1

Choose a blog or a newsfeed (or something similar with an Atom or RSS feed). Every student should do a unique feed, so please "claim" the feed on the class email list (first come, first served). It should be on a topic or topics of which you are qualified to provide classification training data. Find something with at least 100 entries (or items if RSS).

Create between four and eight different categories for the entries in the feed:

examples:

work, class, family, news, deals

liberal, conservative, moderate, libertarian

sports, local, financial, national, international, entertainment

metal, electronic, ambient, folk, hip-hop, pop

Download and process the pages of the feed as per the week 12 class slides.

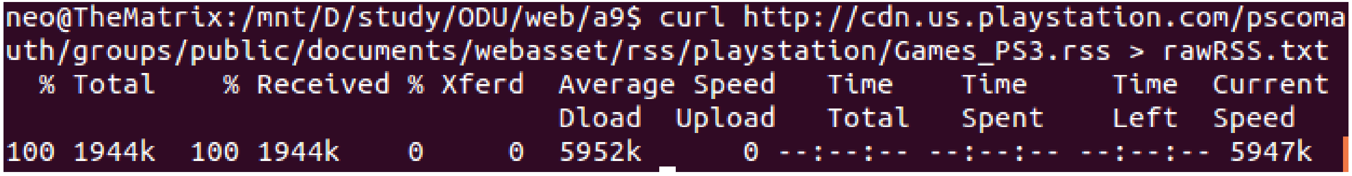
Be sure to upload the raw data (Atom or RSS) to your github account.

SOLUTION

The blog I pick for this assignment: `http://cdn.us.playstation.com/pscomauth/groups/public/documents/webasset/rss/playstation/Games_PS3.rss`
This is a blog about all the game released by play station 3.
The games are classified into 8 genres (in this assignment):
fighting, sports, rpg, arpg, racing, platform, action, fps

Then using shell script to download the raw page:
`curl‘‘http://cdn.us.playstation.com/pscomauth/groups/public/documents/webasset/rss/playstation/Games_PS3.rss"`

Figure 1: Shell output



```
neo@TheMatrix:/mnt/D/study/ODU/web/a9$ curl http://cdn.us.playstation.com/pscoma
uth/groups/public/documents/webasset/rss/playstation/Games_PS3.rss > rawRSS.txt
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 1944k  100 1944k    0     0  5952k      0 --:--:-- --:--:-- --:--:-- 5947k
```

Problem 2

Manually classify the first 50 entries, and then classify (using the fisher classifier) the remaining 50 entries.

Create a table with the title, predicted category, actual category, and cprob() and fisherprob() for the actual category.

SOLUTION

To get cprob and fisherprob, we must have actual category of every entries.

In order to get enough words for classifying, here we combined title with summary together as training data. The script will first come out a predicted category, then ask user to input its actual category.
For first 50 entries, the user input will then used as training data. For the rest of entries, the user input will be just stored for performance measure in problem 3.
Note that raw page contains html tags, we have to remove them at first.[1]

Listing 1: Python script for classify entries

```python
import feedparser
import docclass
import re
def get_pure_text(text):
    t=re.compile(r'<[^>]+>')
    return t.sub('',text)
# Takes a filename of URL of a blog feed and classifies the entries
def read(feed,classifier):
  # Get feed entries and loop over them
  f=feedparser.parse(feed)
  counter=0
  result=[]
  for entry in f['entries']:
    title = get_pure_text(entry['title'].encode('utf-8'))
    summary = get_pure_text(entry['summary'].encode('utf-8'))

    print
    print '-----'
    print('#%d'%counter)
    # Print the contents of the entry
    print 'Title:     '+title
    print

    # Combine all the text to create one item for the classifier
    fulltext='%s\n%s' % (title,summary)
    currentFeed={}
    currentFeed['title']=title
    guess=str(classifier.classify(fulltext))
    currentFeed['guess']=guess
    print ('Guess: '+ guess)
    actual=raw_input('Enter Category: ')
    currentFeed['actual']=actual
    if counter<50:
        classifier.train(fulltext,actual)
    result.append(currentFeed)

    counter+=1
    if counter>=100:
        break
  return result

def calculateProb(table,classifier):
    for entry in table:
        entry['cprob']=classifier.cprob(entry['title'],entry['actual'])
        entry['fisherprob']=classifier.fisherprob(entry['title'],entry['actual'])
    return table
```

```
def WriteFile(data):
    table=open('p2_table.txt','w')
    for entry in data:
        table.write('%s\t%s\t%s\t%f\t%f\n'%(entry['title'],entry['guess'],entry['actual'],entry['cprob'],entry['fisherprob']))
    table.close()
#####script entry#########
fc=docclass.fisherclassifier(docclass.getwords)
fc.setdb("zlGames.db")
data=read('http://cdn.us.playstation.com/pscomauth/groups/public/documents/webasset/rss/playstation/Games_PS3.rss',fc)
table=calculateProb(data,fc)
WriteFile(table)
```

Table 1: Game Category Table

Title	**Predicted**	**Actual**	**cprob**	**fisherprob**
BlazBlue: Chrono Phantasma	fighting	fighting	0.000000	0.999261
MLB 14 The Show	sports	sports	0.000000	0.632189
Ragnarok Odyssey ACE	rpg	rpg	0.000000	0.992042
Batman: Arkham Origins Blackgate - Deluxe Edition	arpg	arpg	0.000000	0.988878
Jimmie Johnson's Anything With An Engine	fighting	racing	0.000000	0.716525
TRINITY: Souls of Zill O'll	fighting	rpg	0.000000	0.747023
FEZ	rpg	platform	0.000000	0.750000
Dynasty Warriors 8: Xtreme Legends	rpg	action	0.000000	0.900965
Deception IV: Blood Ties	rpg	arpg	0.000000	0.837869
Cabela's Big Game Hunter Pro Hunts	sports	fps	0.000000	0.901288
TNN Motorsports HardCore TR	fighting	racing	0.000000	0.923191
Call of Duty: Ghosts Gold Edition	fighting	fps	0.000000	0.954866
The Witch and the Hundred Knight	fighting	arpg	0.000000	0.454264
WARRIORS: Legends of Troy	fighting	action	0.000000	0.821084
METAL GEAR SOLID V: Ground Zeroes	fighting	action	0.000000	0.922060
FINAL FANTASY X/X-2 HD Remaster	action	rpg	0.000000	0.238850
YAIBA: Ninja Gaiden Z	action	action	0.000000	0.864301
LUFTRAUSERS	action	action	0.000000	0.750000
Atelier Escha and Logy ∼Alchemists of the Dusk Sky∼	action	rpg	0.000000	0.871188
Dark Souls II	action	arpg	0.000000	0.921179
Vessel	action	platform	0.000000	0.750000
South Park: The Stick of Truth	rpg	rpg	0.000000	0.489661
Master Reboot	action	action	0.000000	0.335016
NASCAR '14	action	racing	0.000000	0.833333
Growlanser: Heritage of War	action	rpg	0.000000	0.943089
Tales of Symphonia	action	rpg	0.000000	0.860586
Tales of Symphonia Dawn of the New World	rpg	rpg	0.000000	0.332998
Herc's Adventures	racing	rpg	0.000000	0.918752
Castlevania: Lords of Shadow 2	rpg	action	0.000000	0.943089
THIEF	action	action	0.000000	0.750000
Magus	action	arpg	0.000000	0.750000
PAC-MAN MUSEUM	action	action	0.000000	0.764077
Assassins Creed Freedom Cry	action	arpg	0.000000	0.620316
Neo Contra	rpg	platform	0.000000	0.886142
Forest Legends: The Call of Love	rpg	platform	0.000000	0.405915
Mr. Driller	racing	platform	0.000000	0.750000
Tomba! 2	rpg	platform	0.000000	0.750000
Strider	action	platform	0.000000	0.750000
Earth Defense Force 2025	action	action	0.000000	0.852366

Problem 2 continued on next page. . .

Pac-Man World 20th Anniversary	rpg	platform	0.000000	0.712452
LIGHTNING RETURNS: FINAL FANTASY XIII	action	arpg	0.000000	0.473854
Wolf Fang	action	platform	0.000000	0.560622
Far Cry Classic	action	fps	0.000000	0.632608
Zombeer	action	action	0.000000	0.750000
Blowout	racing	platform	0.000000	0.750000
Dustforce	action	platform	0.000000	0.750000
Truck Racer	action	racing	0.000000	0.918752
Trapt	platform	action	0.000000	0.750000
Adam's Venture: Chronicles	rpg	rpg	0.000000	0.968161
Gex: Enter the Gecko	racing	platform	0.000000	0.393812
DRAGON BALL Z: BATTLE OF Z	action	fighting	0.000000	0.695500
Cyber Sled	racing	action	0.000000	0.742811
THE FIREMEN 2: PETE and DANNY	rpg	arpg	0.000000	0.411317
Mark Davis Pro Bass Challenge	action	sports	0.000000	0.466829
Lucifer Ring	action	action	0.000000	0.596574
Assassins Creed Liberation HD	action	action	0.000000	0.257590
The Raven - Legacy of a Master Thief	action	platform	0.000000	0.060845
Twisted Lands: Shadow Town	action	rpg	0.000000	0.368775
Tiny Brains	rpg	rpg	0.000000	0.596574
Dragon Fantasy Book I and II Bundle	rpg	rpg	0.000000	0.339384
The Walking Dead: Season Two	action	rpg	0.000000	0.226454
flOw	action	rpg	0.000000	0.166667
Mutant Mudds Deluxe	action	action	0.000000	0.384503
Aabs Animals	platform	rpg	0.000000	0.596574
Toki Tori	action	platform	0.000000	0.596574
The Walking Dead: Season 2 - Ep.1, All That Remains	action	rpg	0.000000	0.111491
Minecraft	action	arpg	0.000000	0.500000
Strength Of The Sword 3	action	action	0.000000	0.197681
Doki-Doki Universe	rpg	platform	0.000000	0.290409
Gran Turismo 6	action	racing	0.000000	0.384930
Doki-Doki Universe	rpg	platform	0.000000	0.290409
Doki-Doki Universe	rpg	platform	0.000000	0.290409
Oddworld: Abe Boxx	platform	platform	0.000000	0.655185
Painkiller - Hell and Damnation	action	fps	0.000000	0.655185
Super Motherload	action	action	0.000000	0.596574
Saint Seiya: Brave Soldiers + Aries Shion	action	action	0.000000	0.591461
Young Justice: Legacy	action	action	0.000000	0.104574
Arcania - The Complete Tale	action	rpg	0.000000	0.236300
CONTRAST	rpg	rpg	0.000000	0.500000
Need for Speed Rivals	action	racing	0.000000	0.181262
ADVENTURE TIME: EXPLORE THE DUNGEON BECAUSE I DON'T KNOW!	action	action	0.000000	0.216729
AquaPazza	action	fighting	0.000000	0.500000
SOULCALIBURII HD ONLINE	action	fighting	0.000000	0.199787
Farming Simulator	action	rpg	0.000000	0.596574
Air Conflicts: Vietnam	action	action	0.000000	0.655185
Stick It To The Man	action	action	0.000000	0.107770
Blood Knights	action	rpg	0.000000	0.235787
Wonderbook: Walking with Dinosaurs	action	platform	0.000000	0.376061
Wonderbook: Book of Potions	rpg	platform	0.000000	0.476013
Wonderbook: Diggs Nightcrawler	action	platform	0.000000	0.655185
XCOM: Enemy Within	action	action	0.000000	0.340747
Injustice: Gods Among Us Ultimate Edition	action	fighting	0.000000	0.056800
Ratchet and Clank: Into the Nexus	action	action	0.000000	0.193413
Call of Duty: Ghosts	fps	fps	0.000000	0.931765
Call of Duty: Ghosts Digital Hardened Edition	action	fps	0.000000	0.829427
How to Survive	action	action	0.000000	0.742811
The Adventures of Cookie and Cream	action	action	0.000000	0.236300
A-men 2	action	platform	0.000000	0.166667

The Guided Fate Paradox	action	rpg	0.000000	0.411317
Ben 10 Omniverse 2	action	action	0.000000	0.596574

Problem 3

Assess the performance of your classifier in each of your categories by computing precision, recall, and F-measure.

SOLUTION

First we have to get numbers of true positive, false positive and false negative in each category.
Here we use python to iterate through each record above compare the predicted category and actual category, then sum them in a table.
Then, use formula provided in the slide to get precision, recall and f-measure.

Listing 2: Python code to assess the performance

```
tfile=open(’p2_table.txt’)
strlines=tfile.readlines()
category={
     ’fighting’:{’TP’:0,’FP’:0,’FN’:0},
’sports’:{’TP’:0,’FP’:0,’FN’:0},
’rpg’:{’TP’:0,’FP’:0,’FN’:0},
’arpg’:{’TP’:0,’FP’:0,’FN’:0},
’racing’:{’TP’:0,’FP’:0,’FN’:0},
’platform’:{’TP’:0,’FP’:0,’FN’:0},
’action’:{’TP’:0,’FP’:0,’FN’:0},
’fps’:{’TP’:0,’FP’:0,’FN’:0}
}
for line in strlines:
     tuples=line.split(’\t’)
     title=tuples[0]
     guess=tuples[1]
     actual=tuples[2]
     if guess==actual:
          category[actual][’TP’]+=1
     else:
          category[actual][’FN’]+=1
          category[guess][’FP’]+=1
tfile.close()

for c in category:

     category[c][’pre’]=float(category[c][’TP’])/float(category[c][’TP’]+category[
         c][’FP’])
     category[c][’recall’]=float(category[c][’TP’])/float(category[c][’TP’]+
         category[c][’FN’])
     if category[c][’pre’]==0:
          category[c][’f’]=0
     else:
          category[c][’f’]=2*float(category[c][’pre’])*float(category[c][’recall’
              ])/float(category[c][’pre’]+category[c][’recall’])

outfile=open(’p3_table.txt’,’w’)
```

Problem 3 continued on next page. . .

```
for cat in category:
    outfile.write(’%s\t%d\t%d\t%d\n’%(cat,category[cat][’TP’],category[cat][’FP’
        ],category[cat][’FN’]))
outfile.close()

rfile=open(’p3_performance.txt’,’w’)
for cat in category:
    rfile.write(’%s\t%f\t%f\t%f\n’%(cat,category[cat][’pre’],category[cat][’
        recall’],category[cat][’f’]))
rfile.close()
```

Table 2: Statistics about TP,FP and FN for each category

Category	**TP**	**FP**	**FN**
platform	1	2	21
fps	1	0	5
rpg	7	13	15
action	22	38	6
arpg	1	0	8
racing	0	5	6
fighting	1	7	4
sports	1	1	1

Table 3: Performance

Category	**Precision**	**Recall**	**F-measure**
platform	0.333333	0.045455	0.080000
fps	1.000000	0.166667	0.285714
rpg	0.350000	0.318182	0.333333
action	0.366667	0.785714	0.500000
arpg	1.000000	0.111111	0.200000
racing	0.000000	0.000000	0.000000
fighting	0.125000	0.200000	0.153846
sports	0.500000	0.500000	0.500000

*A nice online tool for converting raw table file to latex is recommended here:
`http://www.tablesgenerator.com`

References

[1] Amber. *Python code to remove HTML tags from a string*, 2012 (accessed April 21, 2016).