Machine Learning HW4

b03902035 黃兆緯

1. Common Words in the Clusters

Cluster #1
wordpress 0.988496888673
posts 0.0849955610874
category 0.0702643579418
blog 0.0595945888084
theme 0.0429975921733
wp 0.0405640386571
categories 0.0205161699177
author 0.0164568050193
widget 0.0154794032215
jquery 0.0146821046446

Cluster #2
oracle 0.98391983009
pl 0.112419620098
ora 0.0771766150322
10g 0.0639463381696
procedure 0.0568889257871
jdbc 0.0281620148124
procedures 0.0279838102024
cursor 0.027954992906
tables 0.0270850665894
clause 0.0193310244633

Cluster #3
svn 0.8619677976
subversion 0.43104686605
repository 0.163371209016
commit 0.0910915016214
branch 0.0830470748624
revision 0.0824156816311
externals 0.0597459564229
repositories 0.0535701930602
trunk 0.0515098010194
tortoisesvn 0.0441600547474

Cluster #4
apache 0.918718212728
mod_rewrite 0.253276757458
htaccess 0.169193876623
rewrite 0.137246329532
redirect 0.105328578478
rewriterule 0.0701717976048
mod 0.0584484824903
httpd 0.0512793905573
tomcat 0.0490800145268
requests 0.0422984691558

Cluster #5
excel 0.940319760994
vba 0.265347026991
cell 0.0909587305677
macro 0.0884490089969
workbook 0.071274505581
cells 0.0563647684379
worksheet 0.0534379429356
sheet 0.0531529038925
spreadsheet 0.0465567343132
formula 0.0450918147503

Cluster #6
matlab 0.993209317747
matrix 0.0843047969318
plot 0.0526703692184
vector 0.0283609680407
vectors 0.0241096309326
plotting 0.019868188711
cell 0.0185544142691
3d 0.018543642797
axis 0.0131675923046
draw 0.0103794040235

Cluster #7
studio 0.706754742265
visual 0.696907998548
2005 0.0689702155136
vs2008 0.0517588313997
shortcut 0.0281667668324
macro 0.0276249983148
visualstudio 0.0257833317605
solutions 0.0223034803298
debugger 0.0183675720363
team 0.0183083984411

Cluster #8
cocoa 0.965144450457
objective 0.112081291021
nstableview 0.106959551625
nsstring 0.0966086272742
nsview 0.0862577029234
nsoutlineview 0.0667283465379
xcode 0.0529272763154
mac 0.0482140559729
bindings 0.0442428504793
iphone 0.0417849143416

Cluster #9
mac 0.890100003685
osx 0.341955003509
cocoa 0.137787857642
terminal 0.111370478008
leopard 0.0970085116337
10 0.0934808223048
xcode 0.0754552553753
macos 0.0601053946369
macosx 0.0557712757122
snow 0.0543902334456

Cluster #10
bash 0.992737016529
shell 0.069354855147
sed 0.0330879470497
grep 0.0314335496972
awk 0.0281247549923
stderr 0.0248159602873
scripting 0.023118285049
stdout 0.0223925642826
commands 0.0203440908431
ssh 0.0202421114143

Cluster #11
spring 0.979880435276
bean 0.126884566367
hibernate 0.0781150177713
beans 0.0641549017257
mvc 0.0638490688633
annotations 0.0344624748156
annotation 0.0311453439085
aop 0.0282700463629
controller 0.0282129488855
jpa 0.0254430417266

Cluster #12
hibernate 0.987280957894
hql 0.0862653328126
criteria 0.0574668776423
jpa 0.050953693502
annotations 0.0403343340252
entity 0.0398196595792
cascade 0.0330377870346
spring 0.0320777746202
foreign 0.0226714004175
lazy 0.0215241403131

Cluster #13
scala 0.996280163397
actors 0.0475163539244
lift 0.0405286548179
actor 0.0349384955326
immutable 0.01441042288
implicit 0.0140021048929
abstract 0.0131269733371
constructor 0.0131269733371
parser 0.01127295769
functional 0.0111698633078

Cluster #14
sharepoint 0.978600234884
moss 0.110064020713
2007 0.105221909817
webpart 0.0740152705504
wss 0.0567868454458
workflow 0.0539385410767
lists 0.0406858051294
publishing 0.028467563346
aspx 0.0273419133743
sites 0.0174110534509

Cluster #15
jquery 0.87827633748
response 0.18101287258
div 0.176429245036
json 0.166531842773
prototype 0.128397780534
mvc 0.127468877991
toolkit 0.119226510496
requests 0.115848238451
js 0.108607723548
submit 0.094917372908

Cluster #16
qt 0.993165496402
creator 0.0504563320637
widget 0.0466841408832
qwidget 0.0342382253289
qt4 0.0332639161474
qmake 0.0324362134695
signals 0.0300959241334
qtableview 0.0270301778913
signal 0.024877241811
slot 0.0237599401053

Cluster #17
drupal 0.99094571752
node 0.0728525426637
cck 0.0660638131273
module 0.0541774977718
taxonomy 0.0374235996778
theme 0.034539146949
nodes 0.0250857266347
theming 0.0193570343161
forms 0.0115409968576
terms 0.0103297484075

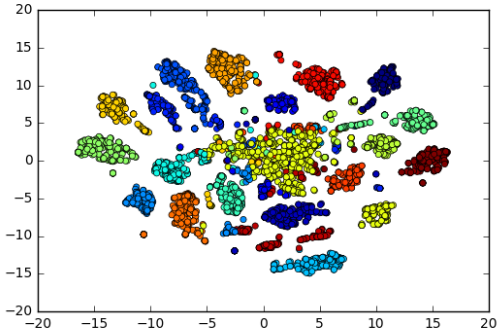
Cluster #18
linq 0.99765991536
iqueryable 0.0293194655313
ienumerable 0.0263875189781
entity 0.0171914862338
entities 0.0167520289333
joins 0.0139532855295
lambda 0.0139532855295
clause 0.0137697629942
vb 0.013753188987
distinct 0.0128936146753

Cluster #19
haskell 0.995775010192
ghc 0.046818472389
cabal 0.0346803499178
monad 0.0320087863351
functional 0.0264584105876
int 0.0228024715451
lists 0.022368328417
io 0.0217166395668
recursion 0.0178799380283
tuple 0.0151291783316

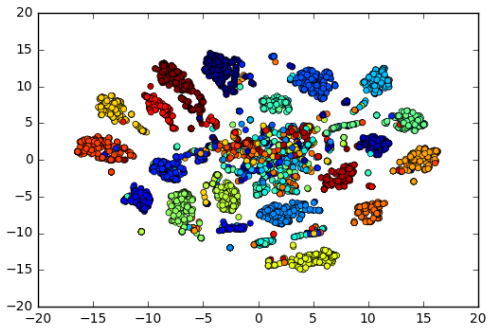
Cluster #20
magento 0.990054019834
product 0.0923552587843
products 0.0577695818843
category 0.0355440177175
price 0.0326267801991
checkout 0.0302502278447
admin 0.0281915277834
cart 0.027663497994
customer 0.0247236544815
module 0.0233908326381

2. Visualization

My Label



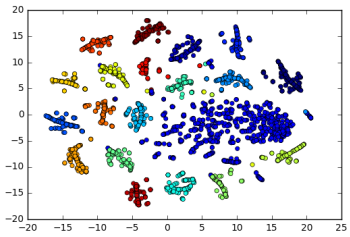
True Label



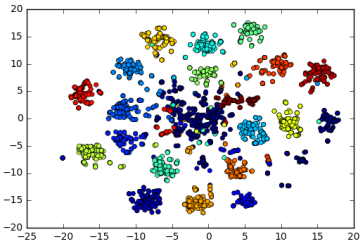
(1) 使用預測的 label 作圖，先用 TruncatedSVD 降維，再 KMeans 分群接著 t-SNE 降成二維。分群的效果很不錯，每個 cluster 距離比較明顯。中間有一大群混雜的點，這些很難進行分群。
(2) 使用真實 label 作圖，可以看到外圍的群集確實分辨的非常好，幾乎都是正確的，而中間的一大群點則是混雜了各種 label，錯誤幾乎都集中在中央。

3. Comparison between different feature extractions

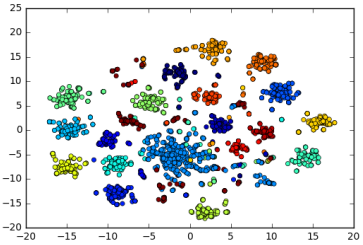
(2) TF-IDF + SVD
score: 0.296
分群分得很漂亮，但是右方的藍色群其實錯誤率很高，而大部分的點都集中在藍色群，所以分數很低



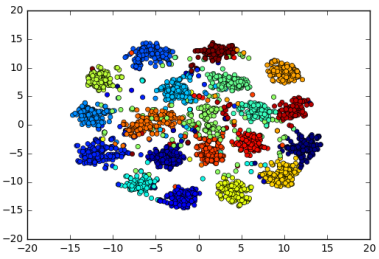
(1) TF-IDF + SVD + Normalize
score: 0.495
Normalize 之後各個群集分散得比較平均，中間的點也比較少，因此結果比起沒有 normalize 有很大進步



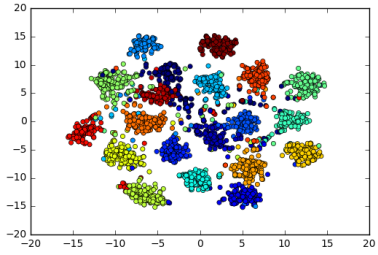
(3) TF-IDF + PCA + Normalize
score: 0.523
用 PCA 降維的結果跟 SVD 的結果看起來很相近，分數也差不多。



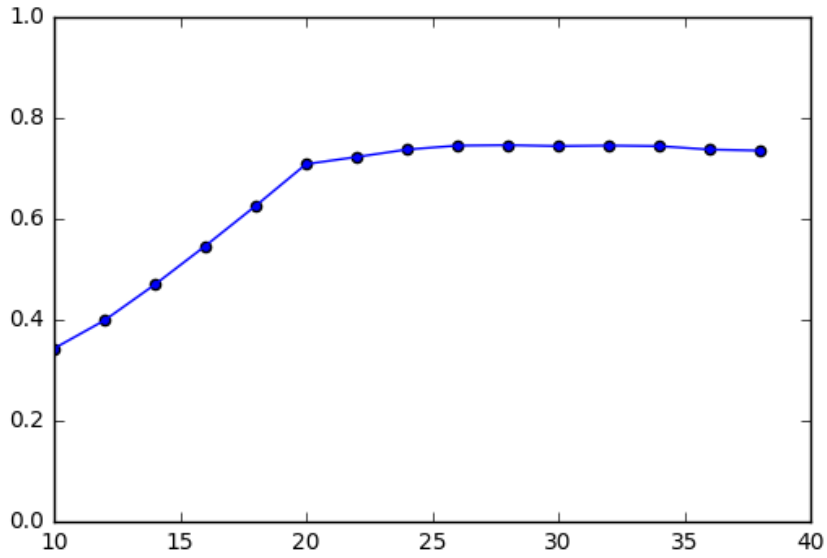
(4) Word2Vec + PCA + Normalize
score: 0.68
每個句子用句中單字的 vector 的平均作為句子的 vector，再用 PCA 降維，比起用 tfidf 當 feature，用 embedding 當 feature 分布比較平均，位於中間的點也比較少，從分數也可以看出這樣的分群效果較佳。



(5) Word2Vec + PCA + stop words filter + Normalize
score: 0.727
將句子中的 stop words 去除，單純看分群結果很難斷定效果，但是分數都會比沒去除 stop words 來的好。



4. Comparison on number of clusters



X 軸為 cluster 數目，y 軸為分數。可以看到分 20 個 cluster 並不是最好的選擇，25 到 30 個 cluster 反而會得到最好的分數（大約 0.74）。看 F measure 的用意：*F measure* measuresperf to penalize false negatives more strongly than false positives by selecting a value $\beta > 1$，這次 beta 為 0.25，因此對 FP 的懲罰比較大。cluster 數量多的時候，預測出來 0 會比較多，所以 FP 減少、FN 增加，這樣可以讓 F-score 變好。