Assignment 2 - DS4Biz Y63

TextScraping_Classification

Team Detail

Team Name: BBGUN

Student 1

Student ID: 61070297
Student Full Name: ธนลล สร้อยสน

Student 2

Student ID: 61070300
Student Full Name: ธัญลักษณ์ โชติรัตนเดชเดโช

Target Web to Scape

http://www.it.kmitl.ac.th/~teerapong/news_archive/index.html

In [150]:
```
import requests
#ร้องขอเนื้อหาของเว็บเพจโดยใช้ requests.get () และจัดเก็บการตอบสนองของเซิร์ฟเวอร์ในการตอบสนองของตัวแปร response
response = requests.get('http://www.it.kmitl.ac.th/~teerapong/news_archive/index.html')
print(response.text)
```

```
<!DOCTYPE html>
<html lang="en">

<head>
        <title>Online News Archive</title>
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="robots" content="noindex">
    <meta name="keywords" content="news,articles,news"/>
        <meta property="og:title" content="Breaking News | International Headlines">
        <meta property="og:site_name" content="News Archive">
        <meta property="og:description" content="Latest news and more from the definitive brand of quality news.">
        <link rel="stylesheet" href="css/bootstrap.min.css">
        <script src="js/jquery-3.2.1.slim.min.js"></script>
        <script src="js/popper.min.js"></script>
        <script src="js/tether.min.js"></script>
        <script src="js/jquery-3.2.1.slim.min.js"></script>
        <script src="js/popper.min.js"></script>
        <script src="js/tether.min.js"></script>
        <script src="js/bootstrap.min.js"></script>
```

In [151]:
```
from bs4 import BeautifulSoup
#response.text ใส่เข้าการสร้างอ็อบเจ็กต์ BeautifulSoup และกำหนดอ็อบเจ็กต์นี้เป็นhtml_soupอาร์กิวเมนต์"html.parser"ระบุว่าเราต้องการทำการแยกวิเคราะห์โดยใ
html_soup = BeautifulSoup(response.text, 'html.parser')
```

ดึงข้อมูลทุกหมวดหมู่ข่าวในแต่ละเดือน

In [152]:
```
link_month = [] #สร้างลิสต์ที่มาเก็บลิ้งในแต่ละเดือน
for i in html_soup.find_all("a",href=True): #ลูปใช้เมธอด find_all()เพื่อค้นหาทุกองค์ประกอบ "a" ที่มีแอตทริบิวต์ href=True
    link = "http://www.it.kmitl.ac.th/~teerapong/news_archive/" + i["href"] #คือสร้างตัวแปรเก็บลิ้งปกติแต่หลังจาก / ใส่ลูปของเพื่อเราหามาในแต่ล
    if "month" in link: #สร้างเงื่อนไขเพื่อเอาhrefเฉพาะที่มีสตริงชื่อmonth
        link_month.append(link) #เพิ่มเส้นเข้าไปในลิสต์
link_month #โชว์ทุกลิ้งที่อยู่ในลิสต์
```

Out[152]:
```
['http://www.it.kmitl.ac.th/~teerapong/news_archive/month-jan-2017.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/month-feb-2017.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/month-mar-2017.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/month-apr-2017.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/month-may-2017.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/month-jun-2017.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/month-jul-2017.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/month-aug-2017.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/month-sep-2017.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/month-oct-2017.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/month-nov-2017.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/month-dec-2017.html']
```

In [153]:
```
Article_cat = [] #สร้างลิสต์มาเก็บประเภทของข่าว
for i in link_month: #การลูปลิ้งที่อยู่ใน ลิส link_month
    response2 = requests.get(i) #ยอยเข้าลิ้งในแต่ละเดือน
    #print( response1.text) ลองปริ้นออกมาแล้วเป็น แท็ค td class='category'
    html_soup2 = BeautifulSoup(response2.text, 'html.parser') #response2.text โดยการสร้างอ็อบเจ็กต์ BeautifulSoup และกำหนดอ็อบเจ็กต์นี้เป็
    for j in  html_soup2.find_all('td', class_='category'): #คือใช้คำสั่ง find_all หาแท็ค td ที่ class = category ทั้งหมด ก็จะ return เป็น
        Article_cat.append(j.get_text().lstrip('\xa0')) #คือดึงข้อความจาก tag ใน htmlออกมาและlstripคือให้มันตัดพวกช่องว่าง, \nที่อยู่หน้าหลังอ
Article_cat #เรียกดูค่าที่อยู่ในลิสต์
```

Out[153]:
```
['technology',
 'business',
 'technology',
 'business',
 'sport',
 'sport',
 'sport',
 'sport',
 'technology',
 'technology',
 'sport',
 'business',
 'business',
 'technology',
 'technology',
 'technology',
 'business',
 'business',
 'technology',
 'sport',
```

In [154]:
```
#เนื่องจากมีบางค่าในลิสเท่ากับ N/A เราจึงเช็คจำนวนทั้งหมดในลิสต์Articleที่รวมN/Aด้วย
len(Article_cat)
```

Out[154]: 1461

In [155]:
```
for i in Article_cat: #ลูปในลิสต์ Article_cat
    if i == "N/A": #ถ้าสตริงในลิสต์เท่ากับ "N/A"
        Article_cat.remove(i) #รีมูฟให้สตริงเท่ากับ"N/A" ออกจากลิสต์
len(Article_cat) #ดูจำนวนหลังจากที่รีมูฟสตริงเท่ากับ "N/A" ออกจากลิสต์ไปแล้ว
```

Out[155]: 1408

Type *Markdown* and LaTeX: α^2

In [156]:
```
file = open('target/category.txt','w')
for i in Article_cat:
    file.write(i+'\n')
file.close()
```

ดึงบทความของข่าวในแต่ละเดือน

In [157]:
```
link_new = [] #สร้างลิสต์มาเก็บลิ้งของข่าว
for i in link_month:#ใช้ i วนลูปของลิ้งแต่ละเดือนที่เก็บไว้อยู่ในลิสต์ link_month
    response3 = requests.get(i)  #ยิงเข้าลิ้งของแต่ละเดือน
    html_soup3 = BeautifulSoup(response3.text, 'html.parser') #response3.text โดยการสร้างอ็อบเจ็กต์ BeautifulSoup และกำหนดอ็อบเจ็กต์นี้เป็
    #print(html_soup3) ปริ้นออกมาดูtagว่าควรเข้าลิ้งแท็คอะไร
    web = html_soup3.find('tbody') #เข้าถึง tag <tbody>
    for j in web.find_all('a', href=True): #คือใช้คำสั่ง find_all หาแท็ค a ที่ href=True ทั้งหมด ก็จะ return เป็น list แล้วดึงค่าใน list ออก
        link_web = "http://www.it.kmitl.ac.th/~teerapong/news_archive/"+ j['href'] #นำแท็ค a ที่ href=True มาต่อจากลิ้ง
        link_new.append(link_web) #เพิ่มลิ้งค์หัวข้อข่าวทั้งหมดในแต่ละเดือนเข้าไปในลิสต์ link_new
link_new
```

Out[157]:
```
['http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-0418.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-0027.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-0631.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-2105.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-3300.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-4187.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-1974.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-3666.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-2629.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-2415.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-4210.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-4789.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-3452.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-2428.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-4766.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-2595.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-2935.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-0578.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-3023.html',
 'http://www.it.kmitl.ac.th/~teerapong/news_archive/article-jan-2356.html',
```

In [158]:
```
content_title = [] #สร้างลิสต์ไว้ content ของหัวข้อข่าวทั้งหมดของทุกเดือน
for i in link_new: #วนลูปเข้าถึงของหัวข้อข่าวที่เราหา
    response4 = requests.get(i) #ร้องขอเข้าถึงข้อมูลหัวข้อข่าว
    html_soup4 = BeautifulSoup(response4.text, 'html.parser') #response4.text โดยการสร้างอ็อบเจ็กต์ BeautifulSoup และกำหนดอ็อบเจ็กต์นี้เป็
    #print(html_soup4) ปริ้นออกมาเพื่อดูว่าcontentอยู่ในtagอะไร ทำให้เรารู้ว่าอยู่ในtag p
    text =   html_soup4.find_all("p") #ใช้คำสั่ง find_all()เพื่อหาแท็ค <p>ทั้งหมด
    #print(text) ปริ้นออกมาดูเพื่อดูoutput
    storage = []
    for j in text[1:-1:1]: #จากการปริ้นเอาท์พุตออกมาบรรทัดย้างบนบางลิสต์ของเนื้อหาในตำแหน่งที่อยู่ข้อความเหมือนหัวข้อเรื่องเราจึงไม่เอา และตำแหน่งสุดท้าย
        storage.append(j.get_text()) #ดึงข้อความจาก tag ใน html ออกมา
    content_title.append("".join(storage)) #ในเนื้อหาข่าวมีหลายแท็คบางแท็คก็มีข้อความบางแท็คก็เป็น""ว่างๆ เราจึงเอาแท็คที่เป็น""มา.joinเพื่อให้ข่
content_title
```

Out[158]:
```
['The sporting industry has come a long way since the â\x80\x9860s. It has carved out for itself a niche with its roots so de
ep that I cannot fathom the sports industry showing any sign of decline any time soon - or later.The reason can be found in t
his seemingly subtle difference - other industries have customers; the sporting industry has fans. Vivek RanadivÃ©, leader of
the ownership group of the NBAâ\x80\x99s Sacramento Kings, explained it beautifully, â\x80\x9cFans will paint their face purp
le, fans will evangelize. ... Every other CEO in every business is dying to be in our position â\x80\x94 theyâ\x80\x99re dyin
g to have fans.â\x80\x9dWhile fan passion alone could almost certainly keep the industry going, leagues and sporting franchis
es have decided not to rest on their laurels. The last few years have seen the steady introduction of technology into the wor
ld of sports - amplifying fansâ\x80\x99 appreciation of games, enhancing athletesâ\x80\x99 public profiles and informing thei
r training methods, even influencing how contests are waged.Also, digital technology in particular has helped to create an al
ternative source of revenue, besides the games themselves - corporate sponsorship. They achieved this by capitalizing on the
ardor of their customer base - sorry, fan base.',
 'Shares in Europe\'s leading reinsurers and travel firms have fallen as the scale of the damage wrought by tsunamis across s
outh Asia has become apparent.More than 23,000 people have been killed following a massive underwater earthquake and many of
the worst hit areas are popular tourist destinations. Reisurance firms such as Swiss Re and Munich Re lost value as investors
worried about rebuilding costs. But the disaster has little impact on stock markets in the US and Asia.Currencies including t
he Thai baht and Indonesian rupiah weakened as analysts warned that economic growth may slow. "It came at the worst possible
time," said Hans Goetti, a Singapore-based fund manager. "The impact on the tourist industry is pretty devastating, especiall
y in Thailand." Travel-related shares dropped in Europe, with companies such as Germany\'s TUI and Lufthansa and France\'s Cl
ub Mediterranne sliding. Insurers and reinsurance firms were also under pressure in Europe.Shares in Munich Re and Swiss Re -
the world\'s two biggest reinsurers - both fell 1.7% as the market speculated about the cost of rebuilding in Asia. Zurich Fi
```

In [159]:
```
file2 = open('datastore/AllArticles_OnlyContent.txt','w',encoding='utf-8') #ทำการเปิดไฟล์และเขียน
for i in content_title:
    file2.writelines(i+'\n')
file.close()
len(content_title)
```

Out[159]: 1408

Text Mining

In [160]:
```
fin = open('datastore/AllArticles_OnlyContent.txt', 'r',encoding='utf-8') #ทำการเปิดไฟล์ และอ่านไฟล์
raw_documents = fin.readlines() #อ่านทีละบรรทัด
fin.close()
print("Read %d raw text documents" % len(raw_documents))
```

```
Read 1408 raw text documents
```

In [161]:
```
raw_documents #มี \n อยู่ในทุกๆตำแหน่งของเนื้อหาข่าวต้องตัดออก
```

Out[161]:
```
['The sporting industry has come a long way since the â\x80\x9860s. It has carved out for itself a niche with its roots so de
ep that I cannot fathom the sports industry showing any sign of decline any time soon - or later.The reason can be found in t
his seemingly subtle difference - other industries have customers; the sporting industry has fans. Vivek RanadivÃ©, leader of
the ownership group of the NBAâ\x80\x99s Sacramento Kings, explained it beautifully, â\x80\x9cFans will paint their face purp
le, fans will evangelize. ... Every other CEO in every business is dying to be in our position â\x80\x94 theyâ\x80\x99re dyin
g to have fans.â\x80\x9dWhile fan passion alone could almost certainly keep the industry going, leagues and sporting franchis
es have decided not to rest on their laurels. The last few years have seen the steady introduction of technology into the wor
ld of sports - amplifying fansâ\x80\x99 appreciation of games, enhancing athletesâ\x80\x99 public profiles and informing thei
r training methods, even influencing how contests are waged.Also, digital technology in particular has helped to create an al
ternative source of revenue, besides the games themselves - corporate sponsorship. They achieved this by capitalizing on the
ardor of their customer base - sorry, fan base.\n',
 'Shares in Europe\'s leading reinsurers and travel firms have fallen as the scale of the damage wrought by tsunamis across s
outh Asia has become apparent.More than 23,000 people have been killed following a massive underwater earthquake and many of
the worst hit areas are popular tourist destinations. Reisurance firms such as Swiss Re and Munich Re lost value as investors
worried about rebuilding costs. But the disaster has little impact on stock markets in the US and Asia.Currencies including t
he Thai baht and Indonesian rupiah weakened as analysts warned that economic growth may slow. "It came at the worst possible
time," said Hans Goetti, a Singapore-based fund manager. "The impact on the tourist industry is pretty devastating, especiall
y in Thailand." Travel-related shares dropped in Europe, with companies such as Germany\'s TUI and Lufthansa and France\'s Cl
ub Mediterranne sliding. Insurers and reinsurance firms were also under pressure in Europe.Shares in Munich Re and Swiss Re -
the world\'s two biggest reinsurers - both fell 1.7% as the market speculated about the cost of rebuilding in Asia. Zurich Fi
```

In [162]:
```
new_doc = []
for i in raw_documents:
    new_doc.append(i.strip('\n')) #ลบ\n
new_doc
```

Out[162]:
```
['The sporting industry has come a long way since the â\x80\x9860s. It has carved out for itself a niche with its roots so de
ep that I cannot fathom the sports industry showing any sign of decline any time soon - or later.The reason can be found in t
his seemingly subtle difference - other industries have customers; the sporting industry has fans. Vivek RanadivÃ©, leader of
the ownership group of the NBAâ\x80\x99s Sacramento Kings, explained it beautifully, â\x80\x9cFans will paint their face purp
le, fans will evangelize. ... Every other CEO in every business is dying to be in our position â\x80\x94 theyâ\x80\x99re dyin
g to have fans.â\x80\x9dWhile fan passion alone could almost certainly keep the industry going, leagues and sporting franchis
es have decided not to rest on their laurels. The last few years have seen the steady introduction of technology into the wor
ld of sports - amplifying fansâ\x80\x99 appreciation of games, enhancing athletesâ\x80\x99 public profiles and informing thei
r training methods, even influencing how contests are waged.Also, digital technology in particular has helped to create an al
ternative source of revenue, besides the games themselves - corporate sponsorship. They achieved this by capitalizing on the
ardor of their customer base - sorry, fan base.',
 'Shares in Europe\'s leading reinsurers and travel firms have fallen as the scale of the damage wrought by tsunamis across s
outh Asia has become apparent.More than 23,000 people have been killed following a massive underwater earthquake and many of
the worst hit areas are popular tourist destinations. Reisurance firms such as Swiss Re and Munich Re lost value as investors
worried about rebuilding costs. But the disaster has little impact on stock markets in the US and Asia.Currencies including t
he Thai baht and Indonesian rupiah weakened as analysts warned that economic growth may slow. "It came at the worst possible
time," said Hans Goetti, a Singapore-based fund manager. "The impact on the tourist industry is pretty devastating, especiall
y in Thailand." Travel-related shares dropped in Europe, with companies such as Germany\'s TUI and Lufthansa and France\'s Cl
ub Mediterranne sliding. Insurers and reinsurance firms were also under pressure in Europe.Shares in Munich Re and Swiss Re -
the world\'s two biggest reinsurers - both fell 1.7% as the market speculated about the cost of rebuilding in Asia. Zurich Fi
```

In [163]:
```
fin2 = open('Target/category.txt', 'r')
raw_documents2 = fin2.readlines()
fin2.close()
print("Read %d raw text documents" % len(raw_documents2))
```

```
Read 1408 raw text documents
```

In [164]:
```
raw_documents2
```

Out[164]:
```
['technology\n',
 'business\n',
 'technology\n',
 'business\n',
 'sport\n',
 'sport\n',
 'sport\n',
 'sport\n',
 'technology\n',
 'technology\n',
 'sport\n',
 'business\n',
 'business\n',
 'technology\n',
 'technology\n',
 'technology\n',
 'business\n',
 'business\n',
 'technology\n',
 'sport\n',
```

In [165]:
```
Y = []
for i in raw_documents2:
    Y.append(i.strip('\n')) #ลบ\nออก
Y
```

Out[165]:
```
['technology',
 'business',
 'technology',
 'business',
 'sport',
 'sport',
 'sport',
 'sport',
```

```
'technology',
'technology',
'sport',
'business',
'business',
'technology',
'technology',
'technology',
'business',
'business',
'technology',
'sport',
```

In [166]:
```
from nltk.stem.porter import PorterStemmer
nltk.download('averaged_perceptron_tagger')
nltk.download('wordnet')
```

```
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     C:\Users\WIN10\AppData\Roaming\nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\WIN10\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

Out[166]: True

In [167]:
```
from sklearn.feature_extraction.text import CountVectorizer
import nltk
#------------------------------------Tokenizer and WordNet Lemmatizer with POS----------------------------------
# define the function
def lemma_tokenizer_w_pos_tag(text):
    # define a nested function for converting POS tag for lemmatizer
    def convert_tags(tag):
        if tag == 'vbd' or tag == 'vbg' or tag == 'vbz':
            return 'v'
        else:
            return 'n'

    #-----------------------------------------------Using WordNet Lemmatizer----------------------------------------
    # use the standard scikit-learn tokenizer first
    standard_tokenizer = CountVectorizer().build_tokenizer()
    tokens = standard_tokenizer(text)
    tokens_with_pos_tag = nltk.pos_tag(tokens)
    # print(tokens_with_pos_tag)
    # then use NLTK to perform lemmatisation on each token
    lemmatizer = nltk.WordNetLemmatizer()
    lemma_tokens = []
    for token in tokens_with_pos_tag:
        new_tag = convert_tags(token[1].lower())
        lemma_tokens.append(lemmatizer.lemmatize(token[0], new_tag))

#     print(lemma_tokens)
    return lemma_tokens
```

Text preprocessing and Term Weighting

In [168]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
# we can pass in the same preprocessing parameters
vectorizer = TfidfVectorizer(stop_words="english",min_df = 10,tokenizer=lemma_tokenizer_w_pos_tag) #ทำการย่องน้ำหนักคำที่ความถี่น้อยกว่า
X = vectorizer.fit_transform(new_doc)
# display some sample weighted values
print(X)
```

```
C:\Users\WIN10\anaconda3\lib\site-packages\sklearn\feature_extraction\text.py:383: UserWarning: Your stop_words may be inconsis
tent with your preprocessing. Tokenizing the stop words generated tokens ['le', 'u'] not in stop_words.
  warnings.warn('Your stop_words may be inconsistent with '
```

```
  (0, 405)      0.2139012602529119
  (0, 170)      0.10627904676704929
  (0, 2941)     0.12970798038470363
  (0, 810)      0.09426412857273518
  (0, 2652)     0.08488553054357058
  (0, 2908)     0.09276381461379726
  (0, 257)      0.11231401315955562
  (0, 830)      0.07259028010076334
  (0, 1528)     0.10047882725417405
  (0, 2239)     0.09312961733756304
  (0, 966)      0.07428022173629091
  (0, 782)      0.10764288080181825
  (0, 1662)     0.10695063012645595
  (0, 2019)     0.11231401315955562
  (0, 3217)     0.09312961733756304
  (0, 1663)     0.131665420675381
  (0, 2431)     0.10835711132439609
  (0, 2470)     0.07867083818621317
  (0, 1121)     0.11146499270806044
  (0, 1387)     0.09756301232422214
  (0, 3448)     0.045269693959440433
  (0, 3127)     0.12519570040199005
  (0, 1714)     0.12970798038470363
  (0, 2976)     0.1279073098701597
  (0, 2770)     0.06967900046826814
  :     :
  (1407, 1982)  0.08649009609221325
  (1407, 3423)  0.26376263032821384
  (1407, 3336)  0.3012123898920228
  (1407, 2325)  0.13467791268001964
  (1407, 2744)  0.11113978330351282
  (1407, 2168)  0.08134498334585134
  (1407, 1437)  0.07102657237652493
  (1407, 1258)  0.12181954444209993
  (1407, 28)    0.13263024743860208
  (1407, 2322)  0.06802514127689525
  (1407, 3121)  0.07935521497193686
  (1407, 2101)  0.09099613242567584
  (1407, 1568)  0.10182245410904453
  (1407, 3071)  0.10921239916836216
  (1407, 973)   0.09116446761343322
  (1407, 3368)  0.06872764388791303
  (1407, 963)   0.10158128750023103
  (1407, 822)   0.07835367005804882
  (1407, 406)   0.08171046527132773
  (1407, 2731)  0.03326977703469115
  (1407, 3371)  0.10408641000375767
  (1407, 1387)  0.12812195968026197
  (1407, 3468)  0.04210359639823879
  (1407, 1243)  0.10711203044654932
  (1407, 3175)  0.052398879049388045
```

สร้างโมเดล โดยใช้วิธีการ Holdout Splitting และ cross validation 5 fold

นำข้อมูลมาแบ่งเป็นชุดเทรนและชุดเทส โดยทดสอบให้ชุดTrain 70:30 ชุดTest

In [169]:
```
from sklearn.model_selection import train_test_split
from sklearn import metrics
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.3, random_state=2) #แบ่งชุดข้อมูลเป็นแบบ train,test และใส่ค่า ra
```

K nearest neighbor

In [170]:
```
from sklearn.neighbors import KNeighborsClassifier
#for i in range(100):
classifier = KNeighborsClassifier(n_neighbors=61)
classifier.fit(X_train, y_train)
classifier_pred = classifier.predict(X_test)
print("Accuracy of KNN = %.2f percent"% (metrics.accuracy_score(y_test,classifier_pred)*100))
```

```
Accuracy of KNN = 98.35 percent
```

In [171]:
```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import cross_val_score
kneigh = KNeighborsClassifier(n_neighbors=20)
cross_knn = cross_val_score(kneigh, X, Y, cv=5)
print("KNN: Mean cross-validation accuracy = %0.2f percent" % (cross_knn.mean()*100))
```

```
KNN: Mean cross-validation accuracy = 97.02 percent
```

Naive Bayes

In [172]:
```
from sklearn.naive_bayes import BernoulliNB
naive = BernoulliNB(alpha=0.6)
naive.fit(X_train, y_train)
naive_pred = naive.predict(X_test)
print("Accuracy of Naive Bayes = %0.2f percent" % (metrics.accuracy_score(y_test,naive_pred)*100))
```

```
Accuracy of Naive Bayes = 97.64 percent
```

In [173]:
```
from sklearn.naive_bayes import BernoulliNB
from sklearn.model_selection import cross_val_score
model_nb = BernoulliNB(alpha=0.2)
cross_naive = cross_val_score(model_nb, X, Y, cv=5)
print("Naive Bayes: Mean cross-validation accuracy = %0.2f percent" %(cross_naive.mean()*100))
```

```
Naive Bayes: Mean cross-validation accuracy = 97.73 percent
```

SVM

In [174]:
```
from sklearn import svm
svm = svm.SVC(C=1, kernel='linear')
svm.fit(X_train, y_train)
svm_pred = svm.predict(X_test)
print("Accuracy of SVM = %0.2f percent" % (metrics.accuracy_score(y_test,svm_pred)*100))
```

```
Accuracy of SVM = 99.05 percent
```

In [175]:
```
from sklearn.svm import SVC
from sklearn.model_selection import cross_val_score
model = SVC(gamma='scale')
acc_scores = cross_val_score(model,X,Y, cv=5, scoring="accuracy")
print("SVM: Mean cross-validation accuracy = %.2f percent" % (acc_scores.mean()*100))
```

```
SVM: Mean cross-validation accuracy = 98.30 percent
```

Decisiontree

In [176]:
```
from sklearn.tree import DecisionTreeClassifier
decision_tree = DecisionTreeClassifier()
decision_tree.fit(X_train, y_train)
decision_tree_pred = decision_tree.predict(X_test)
print("Accuracy of Decision tree = %0.2f percent" % (metrics.accuracy_score(y_test,decision_tree_pred)*100))
```

```
Accuracy of Decision tree = 86.76 percent
```

In [177]:
```
from sklearn.tree import DecisionTreeClassifier
decision_tree = DecisionTreeClassifier(random_state=1)
cross_decision_tree = cross_val_score(decision_tree, X, Y, cv=5)
print("Decision tree: Mean cross-validation accuracy = %0.2f percent" % (cross_decision_tree .mean()*100))
```

```
Decision tree: Mean cross-validation accuracy = 86.79 percent
```

สรุปผล

เลือกวิธี cross validation 5 fold เนื่องจาก Crossvalidation เป็นการตรวจสอบข้อมูลภายในทำให้โมเดลสามารถเรียนรู้ข้อมูลได้มากขึ้นจึงเลือกใช้ค่าความแม่นย่ำของ Crossvalidation โดยโมเดล SVM มีค่าAccuracyเฉลี่ยสูงที่สุด 98.30% ทำให้มีประสิทธิภาพในการจำแนกหมวดหมู่ข่าวจากบทความข่าวได้ดีที่สุด เมื่อเปรียบเทียบกับอีก3โมเดล