

社會科學程式設計  
期末專題  
2018.01.14



# 從社會議題 分析奧斯卡入圍名單

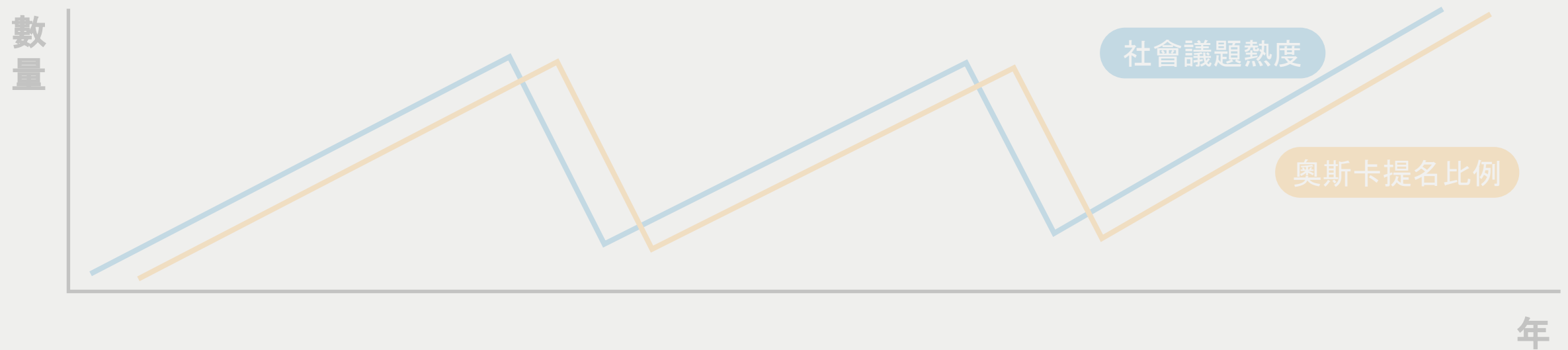


政論一 / 阮彥璋

社會一 / 王聖夫 劉哲愷 柯亮宇

# 問題意識：

- 社會議題的討論熱度是否會影響奧斯卡入圍影片名單？
- 所謂「政治正確」是否影響奧斯卡評審的選擇？
- 不同議題間又有什麼差異？



簡報  
順序

電影分析

奧斯卡入圍名單



社會議題

LGBT、種族

# A.電影分析(預計):

抓取  
片單

Crawl



抓取  
資料

Crawl



判斷  
類型

Train

從IMDb、boxofficereport、Wiki  
抓取:

1. Oscar: 獎項入圍提名影片(最佳影片、原創劇本、改編劇本、長紀錄片)
2. LGBT-related films
3. Racism-related related films
4. top250 films (2004-2017)

將電影丟入IMDb  
Search取得電影

1. Plot summary
2. Keywords

Training & Predict



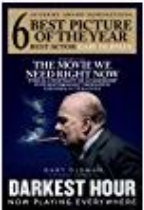



1. 用以上資料train出一套檢驗工具。
2. 判斷Oscar入圍影片是否是LGBT or Race-related film

# 1. 抓取片單

以從IMDb爬下奧斯卡四個獎項的提名名單為例

## The 90th Academy Awards Nominees


### Best Motion Picture of the Year

 <b>WINNER</b> <b>The Shape of Water</b> Guillermo del Toro, J. Miles Dale	 <b>Call Me by Your Name</b> Peter Spears, Luca Guadagnino, Emilie Georges, Marco Morabito
 <b>Darkest Hour</b> Tim Bevan, Eric Fellner, Lisa Bruce, Anthony McCarten, Douglas Urbanski	 <b>Dunkirk</b> Emma Thomas, Christopher Nolan
	

1977	1976	1975	1974
1971	1970	1969	1968
1965	1964	1963	1962
1959	1958	1957	1956
1953	1952	1951	1950
1947	1946	1945	1944
1941	1940	1939	1938
1935	1934	1933	1932
1930	1929		

### Oscars 2018: Red Carpet

Check out the hottest stars at the 90th Academy Awards



Elements Console Sources Network Pe

```
<div class="article"></div>  
<a name="slot_center-8"></a>  
...  
<div class="article">== $0  
  <script type="text/javascript">if (1  
    function(){let{bb,Nomineeswidg  
  <span class="ab_widget">  
    <div id="center-8-react" class="a  
      <div class="nominees-widget" da  
        <div class="nominees-widget_1  
          ".3.0">The 90th Academy Awards  
        <span data-reactid=".3.1"></sp  
        <div class="event-widgets__awa  
          ".3.2">  
        <div class="event-widgets__a  
          ".3.2.$0scar">  
        <h3 class="event-widgets__  
          reactid=".3.2.$0scar.1">  
        <div class="event-widgets_  
          data-reactid=".3.2.$0scar.  
            <div class="event-widge  
              name" data-reactid=".3.  
                Motion Picture of the Y  
                <div class="event-widge  
                  nominations" data-reactid
```

```
list_1=[]  
for div in soup.find_all(class_="article"):  
    list_1.append(div.find(class_="ab_widget"))  
string_1=list_1[0].text.strip()  
list_2=string_1.split("\n")  
string_2=list_2[2].strip(")");  
list_3=string_2.split(",")  
string_3=list_3[1]  
list_a.append(string_3)  
print(len(list_a))
```

15

```
In [4]: import json  
list_b=[]  
for n in range(0,15):  
    list_b.append(json.loads(list_a[n]))  
#list_b[0]為dict_2004;list_b[1]為dict_2005...list_b[14]為dict_2018
```

Java

Type:  
None

.text

Type:  
String

.split()

Type:  
List

.strip()

Type:  
String

json.load


Type:  
Dict

去除前後的雜質



## 2. 抓取影片資料


從IMDb爬下影片的關鍵字與劇情



Movies, TV & Showtimes

Celebs, Events & Photos

News & Community




The Shape of Water (2017)

Plot Keywords

Showing all 417 plot keywords

Sort By:

<div>underwater scene</div> <div>8 of 8 found this relevant</div>	<div>mute woman</div> <div>12 of 13 found this relevant</div>
<div>creature</div> <div>6 of 6 found this relevant</div>	<div>interspecies romance</div> <div>9 of 10 found this relevant</div>
<div>sign language</div> <div>5 of 5 found this relevant</div>	<div>fish man</div> <div>5 of 5 found this relevant</div>
<div>rescue</div> <div>4 of 4 found this relevant</div>	<div>female protagonist</div> <div>4 of 4 found this relevant</div>
<div>water</div> <div>7 of 9 found this relevant</div>	<div>baltimore maryland</div> <div><a href="#">3 of 3 found this relevant</a></div>
<div>loneliness</div> <div>3 of 3 found this relevant</div>	<div>cold war</div> <div>3 of 3 found this relevant</div>




Movies, TV & Showtimes

Celebs, Events & Photos

News & Community

Watchlist



The Shape of Water (2017)

Plot

Edi

Showing all 7 items

Jump to: [Summaries \(6\)](#) | [Synopsis \(1\)](#)

Summaries

At a top secret research facility in the 1960s, a lonely janitor forms a unique relationship with an amphibious creature that is being held in captivity.

From master storyteller [Guillermo del Toro](#) comes THE SHAPE OF WATER, an otherworldly fable set against the backdrop of Cold War era America circa 1962. In the hidden high-security government laboratory where she works, lonely Elisa (Sally Hawkins) is trapped in a life of isolation. Elisa's life is changed forever when she and co-worker Zelda (Octavia Spencer) discover a secret classified experiment. Rounding out the cast are Michael Shannon, Richard Jenkins, Michael Stuhlbarg, and Doug Jones.

—[Fox Searchlight Pictures](#)

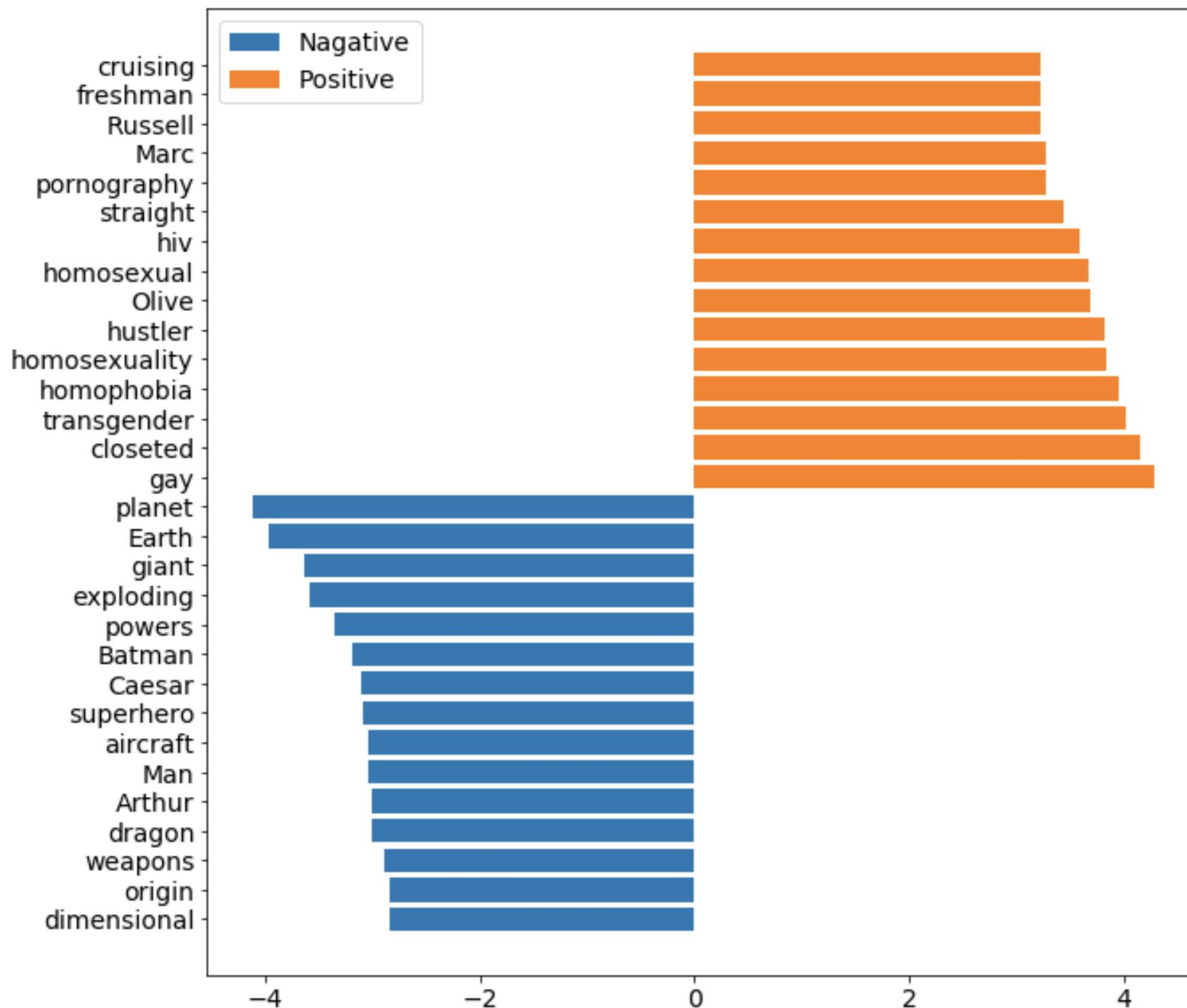
1962 Baltimore. Elisa Esposito, found abandoned as a baby with scars on her neck, has been mute all her life, that disability which has largely led to her not having opportunities. Despite being a bright woman, she works a manual labor job as a cleaner at a military research facility where she has long been friends with fellow cleaner, Zelda Fuller, who often translates her sign language to others at the facility. And she has had no romance in her life, her major emotional support, beyond Zelda, being her aging gay artist neighbor, Giles, the two who live in adjoining apartment units above a movie theater. Like Elisa, Giles is lonely, his homosexuality complicating both his personal and professional life, the

# 3.判斷影片類型

Training & Predict

log odd ratio、word embedding、LogisticRegression

結果



predict

```
with open('data/bestpicture_plotsummary.json') as f:
    data=json.load(f)
with open('data/bestpicture_keywords.json') as f:
    data2=json.load(f)
movielist=[]
x=[]
y=[]
result=[]
result_proba=[]
for movie in list(data.keys()):
    movielist.append(movie)
    for b in data[movie]:
        x.extend(word_tokenize(b))
    for b in data2[movie]:
        x.extend(word_tokenize(b))
    y.append(we_represent(x))
    x=[]

result=model.predict(y)
print(len(result))
result_counter=Counter()
for i in result:
    result_counter[i]+=1
print(result_counter)
```

111

Counter({0: 111})

## A.電影分析：

抓取  
片單



判斷  
類型

／  
從IMDb、boxofficereport、Wiki 抓取

1. Oscar: 獎項入圍提名影片(最佳影片、原創劇本、改編劇本、長紀錄片)
2. LGBT-related films
3. Racism-related related films

## A.電影分析：

抓取  
片單



判斷  
類型

／  
從IMDb、boxofficereport、Wiki 抓取

1. Oscar: 獎項入圍提名影片 (最佳影片、原創劇本、改編劇本、長紀錄片)
2. LGBT-related films
3. Racism-related related films

｜  
判斷奧斯卡入圍影片是否在  
議題名單內



# 判斷類型

判斷奧斯卡入圍的電影是否在LGBT/Racism-related relate films

**bestpicture\_movielist** = [[2004, ["The Lord of the Rings: The Return of the King", "Lost in Translation", ..., "Mystic River", "Seabiscuit"], 2018,...]]

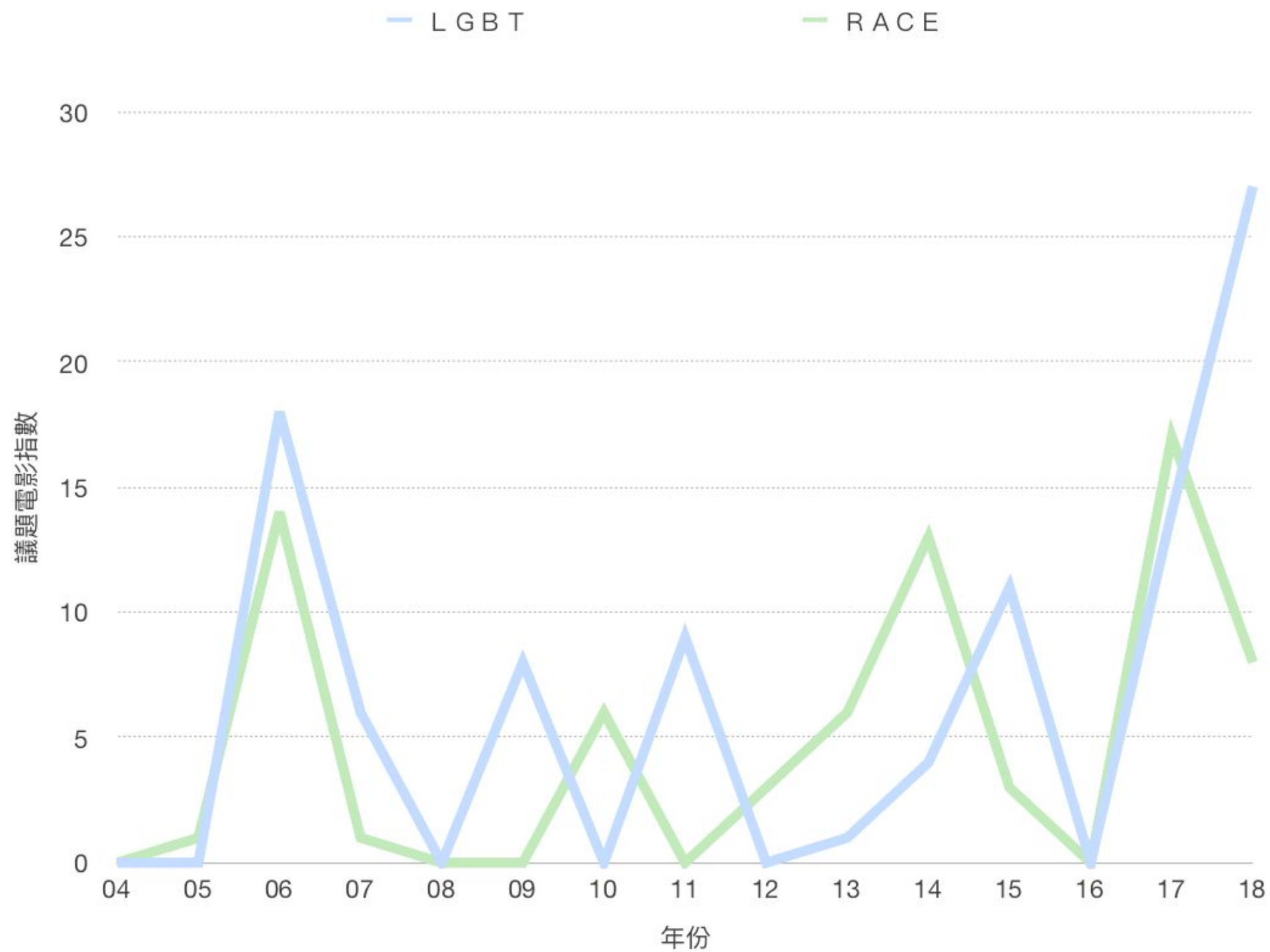
**LGBT-related\_movielist**= [[2004, ["The 24th Day", "Alexander", "Billy's Dad Is a Fudge-Packer", ... "True Love", "White Chicks", "Wild Things 2"],... 2018,...]]

```
bestpicture
2006 : Brokeback Mountain
2006 : Capote
2007 : Little Miss Sunshine
2009 : Milk
2011 : Black Swan
2011 : The Kids Are All Right
2014 : Dallas Buyers Club
2015 : The Imitation Game
2017 : Moonlight
2018 : The Shape of Water
2018 : Call Me by Your Name
2018 : Lady Bird
[(2006, 2), (2007, 1), (2009, 1), (2011, 2), (2014, 1), (2015, 1), (2017, 1), (2018, 3)]
-----
documentaryfeature
2013 : How to Survive a Plague
[(2013, 1)]
```



		權重
最佳影片	得獎	5
	入圍	3
最佳導演	得獎	4
	入圍	2
最佳原創劇本	得獎	2
	入圍	1
最佳改編劇本	得獎	2
	入圍	1
最佳紀錄長片	得獎	2
	入圍	1

# 資料結果



## B. 新聞分析

抓取  
新聞



文章  
分析



趨勢  
分析

用selenium,request從New  
York Times上抓取:

1.LGBT

2.Race

兩大議題在2004-2018年間所  
有的文章內容並存取為pickle

# 抓取新聞

```
link = []
url = 'https://www.nytimes.com/search?endDate={ }1231&query=same%20sex%20marriage%2C%20gay%20right%2C%20ho
driver = webdriver.Chrome('/Users/Kai/Downloads/chromedriver')
driver.get(url)
soup = bs(driver.page_source, 'lxml')
while len(soup.select('.css-vsui0x'))>0:

    try:
        driver.find_element_by_xpath('//*[@id="site-content"]/div/div/div[2]/div[2]/div/button').click()
        time.sleep(1.7)

    except:
        time.sleep(1)
        for i in bs(driver.page_source).select('.css-138we14 a'):
            link.append(prefix + i['href'])

        all_link.append(link)
        time.sleep(1)
        driver.close()
        break
```

## B:新聞分析

抓取  
新聞



文章  
分析



趨勢  
分析

用selenium,request從New York Times上抓取:

- 1.LGBT
- 2.Race

兩大議題在2004-2018年間所有的文章內容並存取為pickle

- 1.將文章內容透過  
lemmatize以及詞性標記  
進行過濾並存至list中
- 2.訓練word2vec模型



# 訓練模型

原始  
文章

```
ok 4200  
ok 4201  
ok 4202  
ok 4203  
ok 4204  
ok 4205
```

文字  
處理後

```
In [4]: len(clean_sent)
```

```
Out[4]: 121510
```

訓練  
模型

```
In [13]: len(model.wv.vocab)
```

```
Out[13]: 26611
```

## B:新聞分析

抓取  
新聞



文章  
分析



趨勢  
分析

用selenium,request從New York Times上抓取:  
1.LGBT  
2.Race  
兩大議題在2004-2018年間所有的文章內容並存取為pickle

1.將文章內容透過  
lemmatize以及詞性標記  
進行過濾並存至list中  
2.訓練word2vec模型

將過濾後的文章都丟入並比對與關鍵字之間的相關程度藉此計算出每篇文章的分數

# 趨勢分析

LGBT score trend:

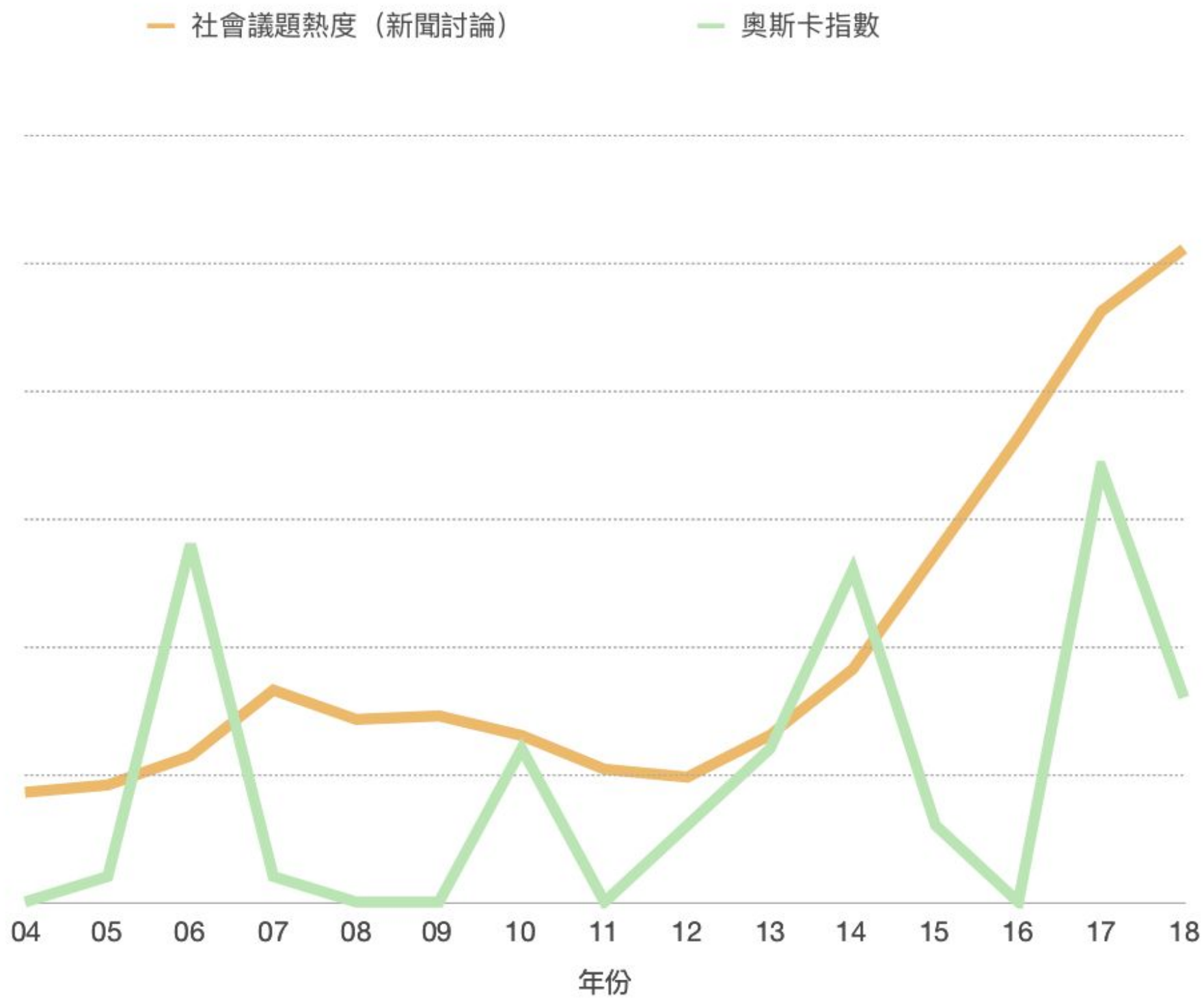
```
Out[58]: [0.003200380388068982,  
0.0029649266616669865,  
0.0027516169295359128,  
0.004122358533777343,  
0.0031613910120453,  
0.0029868513608570964,  
0.0022633815957704135,  
0.0019872309838907445,  
0.0016006616067974764,  
0.0019358804986200312,  
0.0016041966721701151,  
0.0014342737297882416,  
0.002077451659297928,  
0.001328959719863729,
```

2004

2018

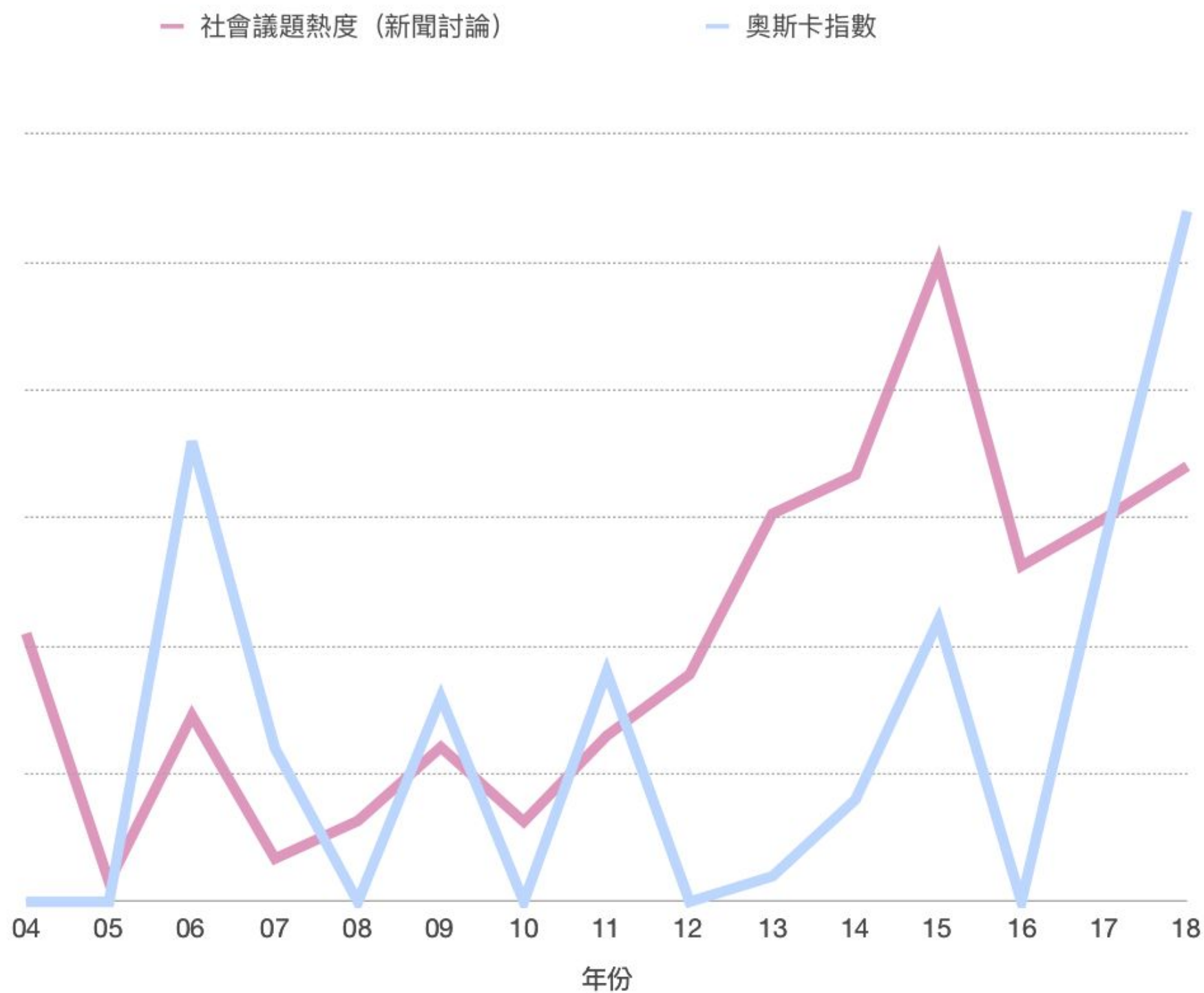
議題新聞佔比

# 結論：



RACE

# 結論：



LGBT



# 分工表

- 電影片單爬蟲：王聖夫、柯亮宇
- 電影分析：王聖夫
- 電影簡報製作：王聖夫、柯亮宇
- 新聞爬蟲：劉哲愷
- 新聞分析：劉哲愷
- 新聞簡報製作：柯亮宇、阮彥璋
- 口頭報告：柯亮宇、阮彥璋