Project 2

เรื่อง A recipe recommendation engine

จัดทำโดย

1. นางสาว ดวงใจ สุกฟอง รหัสนักศึกษา 59050214
2. นางสาว นภสร เปรียญขุนทด รหัสนักศึกษา 59050234
3. นางสาว ประภัสสร ธีระวาส รหัสนักศึกษา 59050256

นำเสนอ

ดร.กุลสวัสดิ์ จิตขจรวานิช

รายวิชา **BIG DATA ANALYSIS** (รหัสวิชา 05506218)

ภาคเรียนที่ 2 ปีการศึกษา 2561

สาขา วิทยาการคอมพิวเตอร์ คณะ วิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

1.ใช้โปรแกรม Pycharm ในการทำ Building the recommendation engine

Pycharm (ไพชาร์ม) คือ โปรแกรมที่ใช้หัดเขียนภาษา Python เป็นโปรแกรมที่ใช้งานได้ง่าย และสามารถใช้งานได้ฟรี เหมาะสำหรับผู้ที่ต้องการจะฝึกเขียนภาษาไพทอน และนอกจากนี้ยังรอบรับระบบปฏิบัติการ Windows Linux MacOS

1.1 นำไลบรารี pandas เข้ามาเพื่อให้สามารถอ่านไฟล์ csv

```
import pandas as pd
```

1.2 โมดูลสามารถใช้เพื่อแยกคุณสมบัติในรูปแบบที่รองรับ machine learning algorithms จากชุดข้อมูลประกอบด้วยรูปแบบ เช่น ข้อความและรูปภาพ

```
from sklearn.feature_extraction.text import countVectorizer
```

1.3 ใช้หาค่าความคล้ายคลึง

```
from sklearn.metrics.pairwise import cosine_similarity
```

1.4 กำหนดฟังก์ชันตัวช่วยสองตัวเพื่อรับ book title from book index และ กลับกัน

```
def get_title_from_index (index):
    return df [df.index == index]["title"].values[0]
def get_index_from_title (title):
    return df [df.title == title]["index"].values[0]
```

2. หลังจากดาวน์โหลดชุดข้อมูล ต้องนำเข้าไลบรารี pandas เพื่ออ่านไฟล์ csv โดยใช้ read_csv ()

```
df = pd.read_csv("booksnew.csv")
```

3. เลือกคอลัมน์ที่ต้องการนำมาใช้ในการหาความคล้ายคลึงของข้อมูลเพื่อให้สามารถแนะนำข้อมูลที่คล้ายกัน

```
features = ['genre' , 'subgenre']
```

4.

4.1 เราจำเป็นต้องทำความสะอาดและประมวลผลข้อมูลล่วงหน้าสำหรับการใช้งาน จะเติมค่า NaN ทั้งหมดด้วยสตริงว่างใน dataframe

```
for feature in features:
    df[feature] = df[feature].fillna('') #filling all NaNs with blank string
```

4.2 สร้างฟังก์ชันสำหรับการรวมค่าของคอลัมน์ให้เป็นสตริงเดียว

```
def combine_features(row):
    return row[' genre ']+" "+row[' subgenre']
```

4.3 สร้างคอลัมน์ใน DF ที่รวม features ที่เลือกทั้งหมด

```
df["combined_features"]= df.apply(combine_features,axis=1)
```

#applying combined_features() method over each rows of dataframe and storing the combined string in "combined_features" column

5. เมื่อสตริงรวมกันแล้วเราสามารถป้อนสตริงเหล่านี้ไปยัง CountVectorizer () สำหรับรับเมทริกซ์การนับ

```
cv = CountVectorizer()    #creating new CountVectorizer() object
count_matrix = cv.fit_transform(df["combined_features"])
# feeding combined strings(book contents) to CountVectorizer() object
```

6. คำนวณ Cosine Similarity ที่มีอยู่ใน count_matrix

```
cosine_sim = cosine_similarity(count_matrix)
```

7. ขั้นตอนต่อไปคือรับชื่อหนังสือที่ผู้ใช้ชื่นชอบในปัจจุบัน

```
book_user_likes = " Amulet of Samarkand, The "
```

7.1 รับ index ของหนังสือเรื่องนี้จากชื่อเรื่อง

```
book_index = get_index_from_title(book_user_likes)
```

7.2 เข้าถึงแถวที่สอดคล้องกับหนังสือที่กำหนดเพื่อค้นหาคะแนนความคล้ายคลึงกันทั้งหมดของหนังสือเล่มนั้น

```
similar_book = list(enumerate(cosine_sim[book_index]))
```

#accessing the row corresponding to given book to find all the similarity scores for that book and then enumerating over it

8. เรียงลำดับรายการ Similar_book ตามคะแนนความคล้ายคลึงกันตามลำดับจากมากไปน้อย

```
sorted_similar_book = sorted(similar_book , key=lambda x:x[1] ,
reverse = True)[1:]
```

9.เรียกใช้ลูปเพื่อพิมพ์ 5 รายการแรกจากรายการ sort_similar_book

```
i=0
print("Top 5 similar book to "+book_user_likes+ "are:\n")
for element in sorted_similar_book:
    print (get_title_from_index(element[0]))
    i=i+1
    if i>= 5:
        break
```

ผลลัพธ์ที่ได้

```
C:\Users\ning\Big_Data1\Scripts\python.exe C:/Users/ning/.PyCharmCE2018.3/config/scratches/booksnew_recommender.py
Top 5 similar book to Amulet of Samarkand, The are:

Trial, The
Outsider, The
Complete Sherlock Holmes, The - Vol I
Complete Sherlock Holmes, The - Vol II
Farewell to Arms, A

Process finished with exit code 0
```