



Courses

Recommender System

Course Recommender System

Project By : PRASAD JADHAV

```
In [1]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import neattext.functions as nfx
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity

from sklearn.metrics.pairwise import cosine_similarity

import warnings
```

```
In [2]: warnings.filterwarnings('ignore')
```

```
In [3]: dataset = pd.read_csv('udemy_courses.csv')
```

```
In [4]: dataset.head()
```

Out[4]:	course_id	course_title	url	is_paid	price	num_subscribers	num_review
0	1070968	Ultimate Investment Banking Course	https://www.udemy.com/ultimate-investment-bank...	True	200	2147	
1	1113822	Complete GST Course & Certification - Grow You...	https://www.udemy.com/goods-and-services-tax/	True	75	2792	9
2	1006314	Financial Modeling for Business Analysts and C...	https://www.udemy.com/financial-modeling-for-b...	True	45	2174	
3	1210588	Beginner to Pro - Financial Analysis in Excel ...	https://www.udemy.com/complete-excel-finance-c...	True	95	2451	
4	1011058	How To Maximize Your Profits Trading Options	https://www.udemy.com/how-to-maximize-your-pro...	True	200	1276	

In [5]: dataset.tail()

Out[5]:	course_id	course_title	url	is_paid	price	num_subscribers	num_review
3673	775618	Learn jQuery from Scratch - Master of JavaScri...	https://www.udemy.com/easy-jquery-for-beginner...	True	100	1040	
3674	1088178	How To Design A WordPress Website With No Codi...	https://www.udemy.com/how-to-make-a-wordpress-...	True	25	306	
3675	635248	Learn and Build using Polymer	https://www.udemy.com/learn-and-build-using-po...	True	40	513	10
3676	905096	CSS Animations: Create Amazing Effects on Your...	https://www.udemy.com/css-animations-create-am...	True	50	300	
3677	297602	Using MODX CMS to Build Websites: A Beginner's...	https://www.udemy.com/using-modx-cms-to-build-...	True	45	901	

```
In [6]: dataset.shape
```

```
Out[6]: (3678, 12)
```

```
In [7]: dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3678 entries, 0 to 3677
Data columns (total 12 columns):
 #   Column                Non-Null Count  Dtype  
---  --
 0   course_id             3678 non-null   int64   
 1   course_title          3678 non-null   object  
 2   url                   3678 non-null   object  
 3   is_paid               3678 non-null   bool     
 4   price                 3678 non-null   int64   
 5   num_subscribers       3678 non-null   int64   
 6   num_reviews           3678 non-null   int64   
 7   num_lectures          3678 non-null   int64   
 8   level                 3678 non-null   object  
 9   content_duration      3678 non-null   float64  
10   published_timestamp    3678 non-null   object  
11   subject               3678 non-null   object  
dtypes: bool(1), float64(1), int64(5), object(5)
memory usage: 319.8+ KB
```

```
In [8]: dataset.isnull().sum()
```

```
Out[8]: course_id             0
course_title          0
url                   0
is_paid              0
price                0
num_subscribers       0
num_reviews           0
num_lectures          0
level                0
content_duration      0
published_timestamp    0
subject              0
dtype: int64
```

```
In [9]: dataset.duplicated().sum()
```

```
Out[9]: 6
```

```
In [10]: dataset = dataset.drop_duplicates()
```

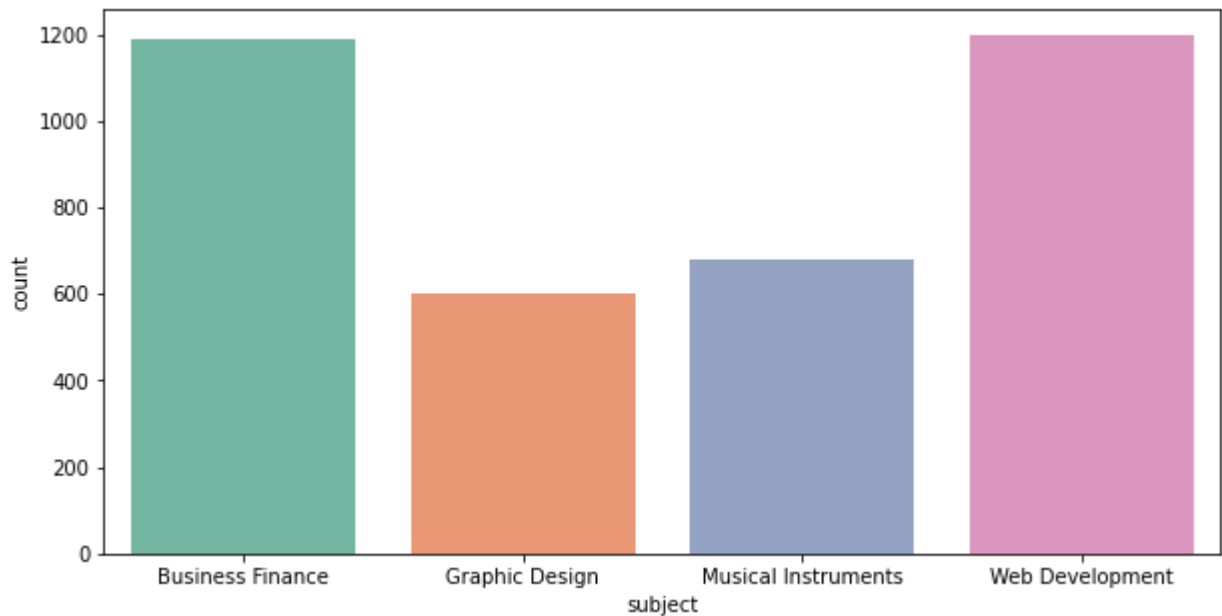
```
In [11]: dataset['course_title'].unique()
```

```
Out[11]: array(['Ultimate Investment Banking Course',
 'Complete GST Course & Certification - Grow Your CA Practice',
 'Financial Modeling for Business Analysts and Consultants', ...,
 'Learn and Build using Polymer',
 'CSS Animations: Create Amazing Effects on Your Website',
 'Using MODX CMS to Build Websites: A Beginner's Guide'],
 dtype=object)
```

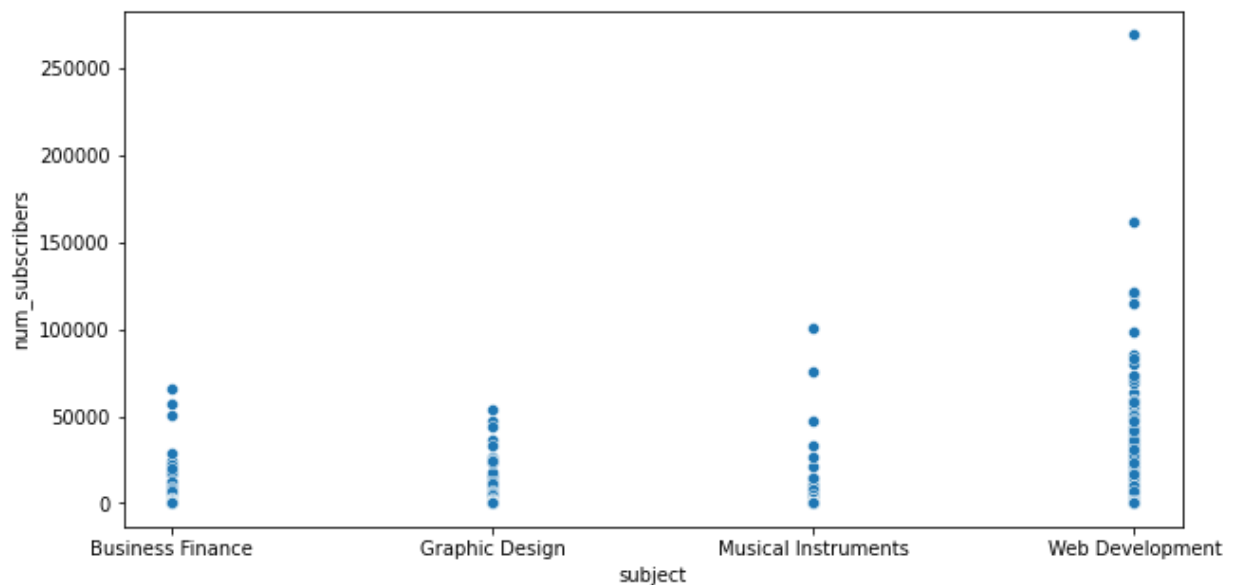
```
In [12]: dataset['subject'].value_counts()
```

```
Out[12]: Web Development      1199  
Business Finance      1191  
Musical Instruments    680  
Graphic Design        602  
Name: subject, dtype: int64
```

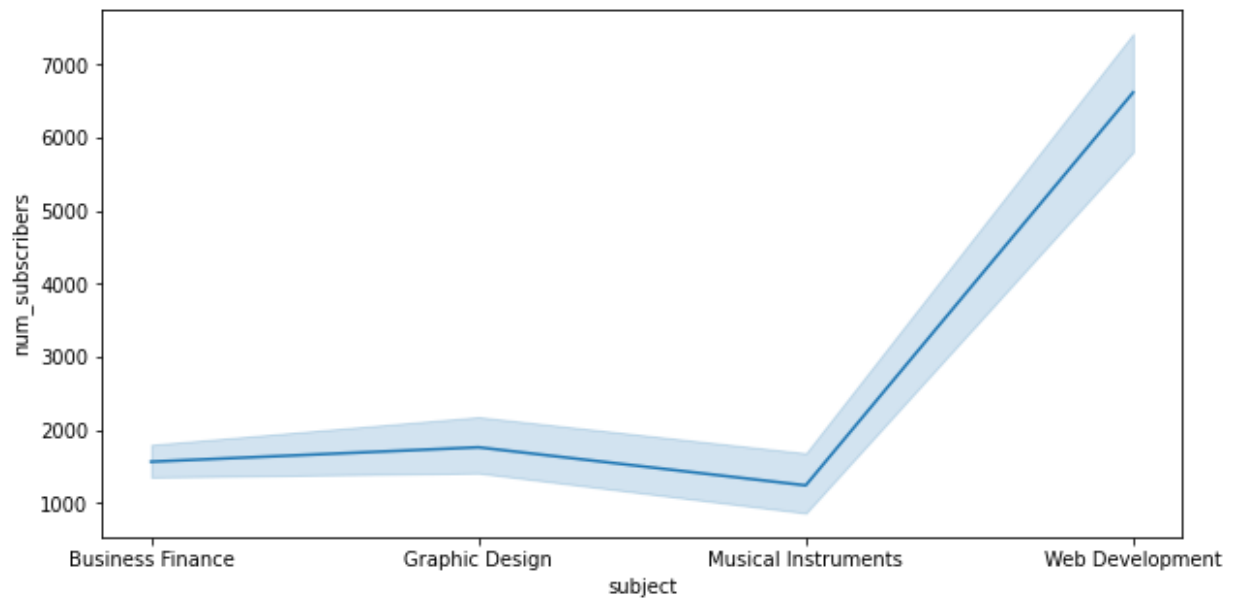
```
In [13]: plt.figure(figsize=(10,5))  
sns.countplot(dataset['subject'], palette='Set2')  
plt.show()
```



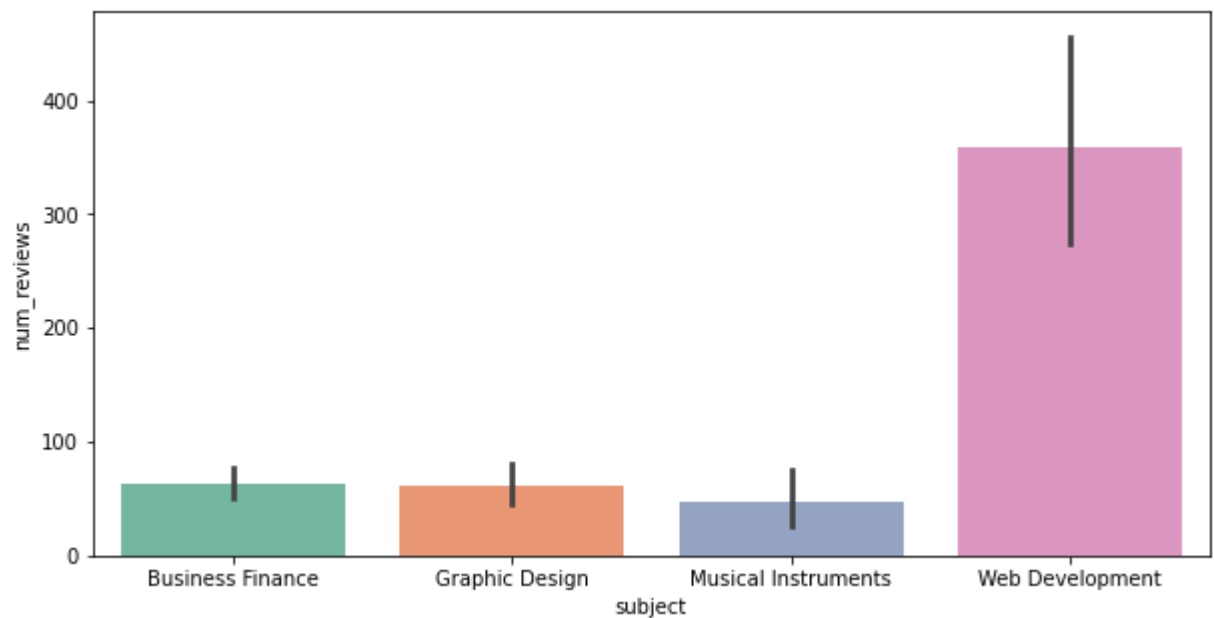
```
In [14]: plt.figure(figsize=(10,5))  
sns.scatterplot(dataset['subject'], dataset['num_subscribers'], palette='Set2')  
plt.show()
```



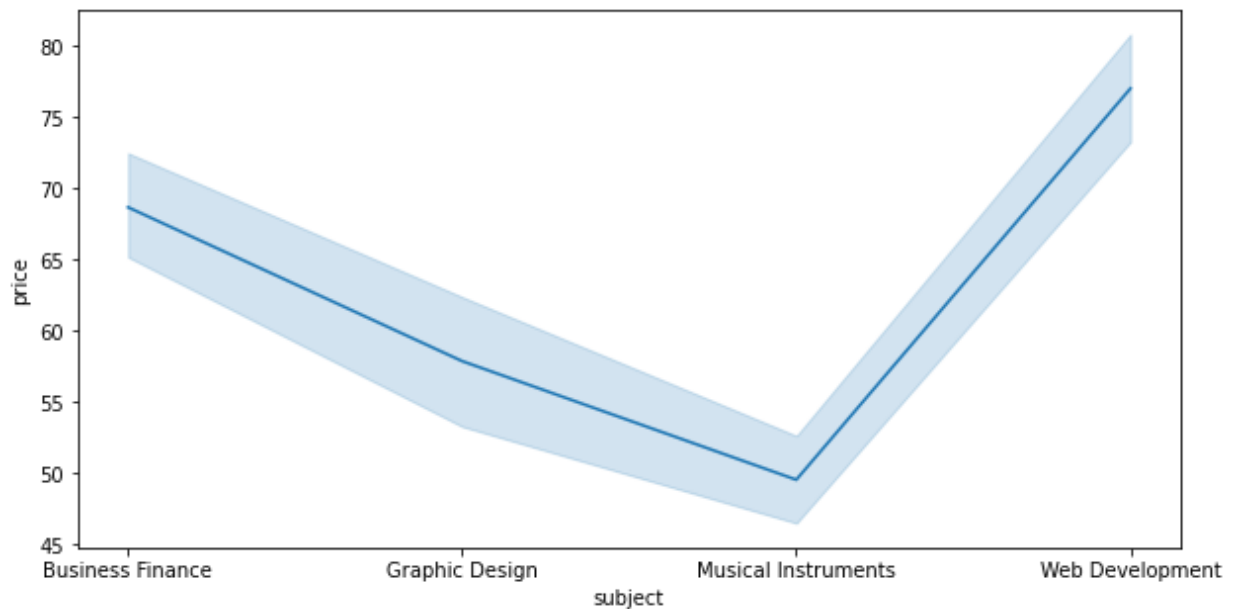
```
In [15]: plt.figure(figsize=(10,5))  
sns.lineplot(dataset['subject'], dataset['num_subscribers'], palette='Set2')  
plt.show()
```



```
In [16]: plt.figure(figsize=(10,5))
sns.barplot(dataset['subject'], dataset['num_reviews'], palette='Set2')
plt.show()
```



```
In [17]: plt.figure(figsize=(10,5))
sns.lineplot(dataset['subject'], dataset['price'], palette='Set2')
plt.show()
```



Popularity-Based Recommendation

```
In [18]: def popularity_based_recommendation(df, top_n=5):
        dataset['popularity_score'] = 0.6 * dataset['num_subscribers'] + 0.4 * dataset['num_reviews']
        df_sorted = dataset.sort_values(by='popularity_score', ascending=False)
        recommended_courses = df_sorted[['course_title', 'popularity_score']].head(top_n)
        return recommended_courses
```

```
In [19]: popularity_based_recommendation(dataset)
```

```
Out[19]:
```

	course_title	popularity_score
2827	Learn HTML5 Programming From Scratch	164805.4
3032	Coding for Entrepreneurs Basic	96729.0
3230	The Web Developer Bootcamp	83928.4
3232	The Complete Web Developer Course 2.0	77672.0
2783	Build Your First Website in 1 Week with HTML5 ...	74544.2

Content-Based Recommendation

```
In [20]: dataset['course_title'] = dataset['course_title'].apply(nfx.remove_stopwords)
        dataset['course_title'] = dataset['course_title'].apply(nfx.remove_special_characters)
```

```
In [21]: dataset['title_subject'] = dataset['course_title'] + ' ' + dataset['subject']
```

```
In [22]: cv = CountVectorizer(max_features=3000)
        vectors = cv.fit_transform(dataset['title_subject']).toarray()
```

```
In [23]: vectors[0]
```

```
Out[23]: array([0, 0, 0, ..., 0, 0, 0], dtype=int64)
```

```
In [24]: similarity = cosine_similarity(vectors)
```

```
In [25]: sorted(list(enumerate(similarity[0])), reverse=True, key=lambda x:x[1])[1:6]
```

```
Out[25]: [(39, 0.7715167498104596),
(240, 0.6666666666666669),
(417, 0.6666666666666669),
(418, 0.6172133998483676),
(657, 0.6172133998483676)]
```

```
In [26]: def recommend(course):

    course_index = dataset[dataset['course_title']==course].index[0]
    distances = similarity[course_index]
    courses_list = sorted(list(enumerate(distances)), reverse=True, key=lambda
    for i in courses_list:
        print(dataset.iloc[i[0]]['course_title'])
```

```
In [27]: recommend('know HTML Learn HTML Basics')
```

```
WordPress Development Beginners
Wordpress Theme Development Beginners
Wordpress beginners Build Websites Fast Coding
Website Coding WordPress Web Skills
Kids Coding Beginners CSS
```

```
In [29]: import pickle
```

```
pickle.dump(dataset,open('course_dict.pkl','wb'))
pickle.dump(similarity,open('similarity.pkl','wb'))
```

```
In [32]: import tkinter as tk
from tkinter import ttk, messagebox
```

```
def popularity_based_recommendation(df, top_n=5):
```

```
    df['popularity_score'] = 0.6 * df['num_subscribers'] + 0.4 * df['num_review
df_sorted = df.sort_values(by='popularity_score', ascending=False)
recommended_courses = df_sorted[['course_title', 'popularity_score']].head
return recommended_courses
```

```
def recommend(course):
```

```
    try:
```

```
        course_index = dataset[dataset['course_title'] == course].index[0]
        distances = similarity[course_index]
        courses_list = sorted(list(enumerate(distances)), reverse=True, key=la
        recommended_courses = [dataset.iloc[i[0]]['course_title'] for i in cou
        return recommended_courses
```

```
    except IndexError:
```

```
        messagebox.showerror('Error', f"Course '{course}' not found.")
```

```
def recommend_button_click():
```

```
    course_title = course_var.get()
```

```

recommended_courses = recommend(course_title)
if recommended_courses:
    popularity_label.pack_forget()
    result_label.config(text='Recommended Courses:\n' + '\n'.join(recommen

root = tk.Tk()
root.title('Course Recommender')
root.geometry('400x300')

font_style=('Arial', 12)
label_color='blue'
heading_color='red'
button_color='green'
result_label_color='black'

label = tk.Label(root, text='Select Course:', font=font_style, fg=label_color)
label.pack(pady=10)

course_titles = dataset['course_title'].tolist()
course_var = tk.StringVar(value=course_titles[0])
course_dropdown = ttk.Combobox(root, textvariable=course_var, values=course_ti
course_dropdown.pack(pady=5)

popularity_recommendations = popularity_based_recommendation(dataset, top_n=5)
popularity_label = tk.Label(root, text='Popularity-Based Recommendations:\n' +
                                font=font_style, fg=label_color)
popularity_label.pack()

recommend_button = tk.Button(root, text='Recommend', command=recommend_button_c
recommend_button.pack(pady=10)

result_label = tk.Label(root, text='', wraplength=350, font=font_style, fg=resu
result_label.pack()

root.mainloop()

```

Thank You!