



Final project Python

Class: Data analyst with Python - 2324_AIT2003_1.

Student name: Nguyễn Văn Trường.

Student id: 22022571.

Đề bài: Phân tích tương tác và nội dung của một/nhiều tài khoản Facebook.

Fanpage: <https://www.facebook.com/VBA.vn>

Github: <https://github.com/TruowNguyen/final-project-python.git>

I. Introduction.

Hiện nay, xử lý dữ liệu là một trong những quá trình trọng trọng trong các lĩnh vực như khoa học dữ liệu (data science), phân tích dữ liệu (data analytics), học máy (machine learning) và đặc biệt là trong ngành Trí tuệ nhân tạo. Việc xử lý dữ liệu đóng vai trò quan trọng trong việc xây dựng mô hình trí tuệ nhân tạo.

Xử lý dữ liệu không chỉ giúp làm sạch dữ liệu để tránh sai sót và nhiễu loạn, mà còn tạo điều kiện thuận lợi cho việc thực hiện các phân tích và xây dựng mô hình dự đoán. Nó đóng vai trò quan trọng trong việc đảm bảo tính nhất quán và tin cậy của kết quả trong quá trình nghiên cứu và phân tích dữ liệu.

Do đó dựa trên yêu cầu của khóa học, em sẽ thu thập và phân tích nội dung từ trang mạng xã hội Facebook. Và Fanpage em chọn trong đề tài này chính là:

<https://www.facebook.com/VBA.vn> (Trang thông tin chính thức của VBA - Giải Bóng rổ Chuyên nghiệp Việt Nam).

II. Crawling Data.

Thông qua công cụ đã được giảng viên lớp thực hành cung cấp là facebook_scraper, em đã lựa chọn cào dữ liệu từ fanpage VBA.

Nhờ công cụ và tài liệu đã được giảng viên hướng dẫn em đã cào dữ liệu thông qua hai quá trình như sau:

Đầu tiên là cài đặt cookies thông qua công cụ hỗ trợ Get cookies.txt LOCALLY. Để thuận tiện trong việc xử lý data sau khi cào xong em sử dụng hai đoạn code riêng biệt để cào dữ liệu từ reactions, post, shares, comments và post_text, comment_text. Dưới đây là 2 đoạn code đó:

Đoạn code để cào dữ liệu như reactions, shares,...

```
FANPAGE_LINK = "VBA.vn"
COOKIE_PATH = "D:/pythoncode/project python/www.facebook.com_cookies.txt"

PAGES_NUMBER = 100; # Number of pages to crawl
post_list = []

results = []

start_url = "VBA.vn"
def handle_pagination_url(url):
    global start_url
    start_url = url
set_cookies("D:/pythoncode/project python/www.facebook.com_cookies.txt")
while True:
    try:
        for post in get_posts(FANPAGE_LINK,
                              options = { "comments" : True, "reactions": True, "allow_extra_requests": True ,
                                           "post_per_page" : 100, "reactors" : False},
                              extra_info=True, pages=PAGES_NUMBER, cookies=COOKIE_PATH, page_limit=None, start_url=start_url, request_url_callback=handle_pagination_url ) :
            print(len(results))
            results.append(post)
            print("All done")
            break
    except exceptions.TemporarilyBanned:
        print("Temporarily banned, sleeping for 10m")
        time.sleep(random.uniform(1, 3))
```

Đoạn code để cào riêng phần post_text và comment_text

```
# get comments
post_list = []

start_url = "VBA.vn"

def handle_pagination_url(url):
    global start_url
    start_url = url
    print("ok")
    while True:
        for post in get_posts(FANPAGE_LINK,
                              options = { "comments" : 50,"reactions": False, "allow_extra_requests": True ,
                                           "post_per_page" : 100,"reactors" :False},
                              extra_info=False, pages=PAGES_NUMBER, cookies=COOKIE_PATH,page_limit=None, start_url=start_url, request_url_callback=handle_pagination_url ) :
            post["start_url"] = start_url
            print("1")
            post["comments_text"] = []
            comments = list(post["comments_full"])

            for comment in comments :
                post["comments_text"].append(comment['comment_text']);
            print(post);
            post_list.append(post)
            time.sleep(random.randrange(2, 10, 1));
        print("All done")
```

Việc crawl dữ liệu này khá khó khăn và chiếm nhiều phần thời gian nhất để hoàn thành dự án vì facebook liên tục ban các acccount của em do đó em phải liên tục thay đổi cookies và lập các acccount mới để tiếp tục quá trình crawl dữ liệu.

Phần dữ liệu thu thập được sau khi crawl xong thì em lưu lại thành các file csv để lưu trữ dữ liệu

Lưu file dưới dạng tên "VBA.vnpost_raw2.csv" và "VBA.vncomments_alternative2.csv".

```
path = "D:/pythoncode/project python/" + FANPAGE_LINK + "post_raw2" + ".csv"
data.to_csv(path, index=False)
print(path)
data
```

```
# get comment
post_df_full = pd.DataFrame(columns=post_list[0].keys(), index=range(len(post_list)), data=post_list)

path = "D:/pythoncode/project python/" + FANPAGE_LINK + "comments_alternative2" + ".csv"
post_df_full.to_csv(path, index=False)
print(path)
post_df_full
```

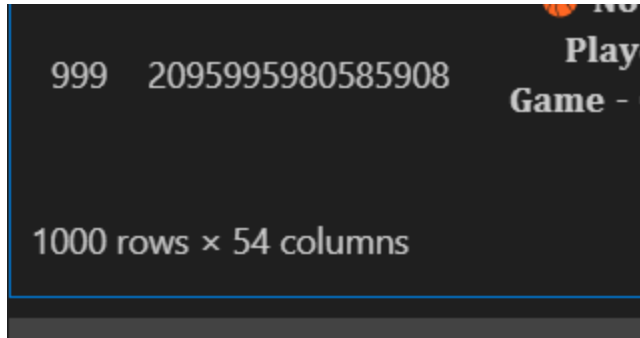
III. Data Cleaning & Processing

Sau quá trình crawl dữ liệu em đã thu thập được hơn 1000 post trong khoảng 90 phút tuy vậy chỉ một trong số chúng là có thể sử dụng lại được

1. Đọc dữ liệu từ file csv

```
df = pd.read_csv("D:\pythoncode\project python\VBA.vnpost_raw2.csv")
```

Sau khi tạo lập một dataframe từ file csv em tiếp tục truy cập vào dataframe df để kiểm tra lại qua dữ liệu thu thập được và như em đã trình bày ở trên sau khi cào dữ liệu xong thì thu thập được hơn 1000 post và 53 cột dữ liệu thu thập



Tuy lượng bài viết ta thu thập được là rất lớn nhưng phần lớn trong số đấy là dữ liệu rác mà ta cần phải lọc bỏ và xử lý lại. Trước tiên em sẽ kiểm tra lại thông tin dữ liệu thông qua hàm `info()`.

```
#kiểm tra dữ liệu sau khi đọc từ csv
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 54 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   post_id                               1000 non-null   int64
1   text                                  957 non-null    object
2   post_text                             939 non-null    object
3   shared_text                           20 non-null     object
4   original_text                         3 non-null      object
5   time                                  1000 non-null   object
6   timestamp                             990 non-null    float64
7   image                                 594 non-null    object
8   image_lowquality                      999 non-null    object
9   images                                962 non-null    object
10  images_description                    962 non-null    object
11  images_lowquality                     1000 non-null   object
12  images_lowquality_description          1000 non-null   object
13  video                                 365 non-null    object
14  video_duration_seconds                 0 non-null      float64
15  video_height                           0 non-null      float64
16  video_id                               365 non-null    float64
17  video_quality                          0 non-null      float64
18  video_size_MB                          0 non-null      float64
19  video_thumbnail                       364 non-null    object
...
52  fetched_time                           145 non-null    object
53  header                                 47 non-null     object
dtypes: bool(3), float64(14), int64(6), object(31)
memory usage: 401.5+ KB
```

2. Làm sạch và xử lý dữ liệu

a. Làm sạch và xử lý dữ liệu của file csv VBA.vnpost_raw2.csv

Đầu tiên em sẽ sửa lại kiểu dữ liệu của cột time thành datetime

```
#sửa kiểu dữ liệu của cột time thành datetime
df["time"] = pd.to_datetime(df["time"])
```

Tiếp theo đó là loại bỏ những dữ liệu không sử dụng đến

Clean những dữ liệu không dùng đến trong quá trình xử lý dữ liệu:

```
df.drop(["image_lowquality", "images_description", "images_lowquality", "images_lowquality_description",
        "video_duration_seconds", "video_height", "video_id", "video_quality", "video_size_MB", "video_thumbnail",
        "video_width", "video", "video_watches", "image_ids", "video_ids", "videos", "was_live", "image", "image_id",
        "factcheck", "is_live", "username", "user_id", "shared_post_id"], axis = 1, inplace = True)
```

Sau đó em sử dụng đến hàm `isna().any()` để kiểm tra tất cả dữ liệu xem có giá trị nào là NaN mà mình cần dùng đến không.

```
df.isna().any()
```

```
post_id      False
text         True
post_text    True
shared_text   True
original_text True
time         False
timestamp    True
images       True
likes        True
comments     False
shares       False
post_url     False
link         True
links        True
user_url     False
shared_time   True
shared_user_id True
shared_username True
shared_post_url True
available    False
comments_full False
reactors     True
w3_fb_url    True
reactions    True
reaction_count False
...
page_id      False
sharers      True
fetch_time   True
header       True
dtype: bool
```

Khi kiểm tra xong nhận thấy các giá trị như comments, shares, post_id, reaction_count có thể sử dụng được và một số thông tin quan trọng bị NaN cần phải xử lý lại như reactions, các lượt tương tác như like, haha, love,...

Sau đây là xử lý phần dữ liệu quan trọng nhất chính là các lượt reactions

```
• # import json
  df["reactions"]

  reactions_list = {
      "thích" : [],
      "thương thương" : [],
      "yêu thích" : [],
      "haha" : [],
      "buồn" : [],
      "wow" : [],
      "phấn nộ" : []
  }

  for index, row in df.iterrows():
      try:
          row["reactions"] = row["reactions"].replace("'", "\'")
          dict = json.loads(row["reactions"])
          for key, value in reactions_list.items():
              if key in dict:
                  reactions_list[key].append(dict[key])
              else:
                  reactions_list[key].append(0)
      except:
          for key, value in reactions_list.items():
              reactions_list[key].append(0)
  df[["thích", "thương thương", "yêu thích", "haha", "buồn", "wow", "phấn nộ"]] = pd.DataFrame(reactions_list)
```

Để xử lý dữ liệu trong cột “reactions” của dataframe em tạo ra các cột mới cho số lượng các loại reaction như thích, thương thương, yêu thích, haha, buồn, wow, phấn nộ. Cuối cùng, sau khi thực hiện đoạn code, dataframe “df” sẽ được mở rộng với các cột mới chứa số lượng của từng loại reaction. Điều này sẽ giúp em phân tích và xử lý dữ liệu hiệu quả hơn

Sau khi xử lý dữ liệu trên em sẽ lọc lại các dữ liệu cần thiết trong quá trình phân tích để sử dụng và tạo ra một dataframe mới là “new_df01”

```
new_df01 = df.loc[:, ["post_id", "post_text", "time", "comments", "shares", "reactions", "reaction_count",
    "thích", "thương thương", "yêu thích", "haha", "buồn", "wow", "phấn nộ"]]

new_df01
```


Sau khi đã lọc lại các dữ liệu cần thiết, em tiếp tục loại bỏ những giá trị không có giá trị sử dụng là các data rác.

```
#loại bỏ những giá trị không sử dụng được  
df_clean = new_df01.dropna()
```

```
df_clean
```

Cuối cùng ta thu được một dataframe “df_clean” chứa khoảng 140 post cần thiết để chuẩn bị cho việc phân tích. Em lưu lại thành một file csv mới để lưu các dữ liệu đã được làm sạch và xử lý.

140 rows × 14 columns

```
df_clean.to_csv("D:/pythoncode/project python/" + "data_cleaned1" + ".csv", index = False)
```

b. Làm sạch và xử lý dữ liệu của file VBA.vncomments_alternative2.csv.

Như các bước đọc file csv và kiểm tra thông tin dữ liệu của df như trên. Do việc crawl dữ liệu của file này khá là đầy đủ và chi tiết do đó em chỉ việc lấy ra các cột có các dữ liệu cần thiết trong việc phân tích

```
df_cmt = df.loc[:, ["post_id", "text", "post_text", "time", "comments", "comments_text"]]
```

Sau đó thay thế các giá trị NaN của 2 cột “post_text” và “comments_text” bằng hàm fillna()

```
df_text["post_text"].fillna("")
```

✓ 0.0s

```
df_text["comments_text"].fillna("")
```

✓ 0.0s

Cuối cùng loại bỏ các giá trị rỗng hoặc không có giá trị bằng hàm `dropna()` và lưu lại thành file csv mới

```
df_text.dropna()
```

✓ 0.0s

	post_id	post_text	comments_text	time
0	649244514087261	Cùng VBA xem lại những hình ảnh đẹp mắt từ trậ...	['có vid hong ad', 'Thùy Dung đã nói từ thiện ...	2023-11-21 19:30:07
1	648554887489557	HAPPY TEACHER'S DAY 20.11 🌸❤️\n\nNhân dịp Ngày...	[]	2023-11-20 12:12:33
2	648528070825572	🔥 GIẢM BUNG NÓC TOÀN BỘ SP TỚI 50%++ BLACK FRID...	[]	2023-11-20 11:00:09
3	648160930862286	BẮT ĐẦU THỐ!!!!\n\nALL-STARS BALL OUT 🍑🍑\n\n...	['All star vba sao ko phai là những star thật ...	2023-11-19 19:18:13
4	648080580870321	5 rapper Rap Việt và 1 rapper bí mật đã sẵn sà...	['Do mình đi 2 người mà có 1 vé,\n\nMình muốn pa...	2023-11-19 15:52:32
...
191	612023131142733	VBA 2023 - Finals Game 1: Saigon Heat vs. Nha ...	[]	2023-09-17 12:38:28
193	611324614545918	VBA AWARDS 2023 - NƠI VINH DANH NHỮNG CÁ NHÂN ...	[]	2023-09-16 14:04:09
194	610861514592228	VBA FINALS 2023 🍑\n\n—————\n\nĐẶT VÉ NGAY TẠI: ...	['Ai còn vé hạng Standard không ạ 🤔 pass mình ...	2023-09-15 20:05:03
195	610858464592533	🍑 Các ballers U22 hãy nhanh tay để được nhận ưu...	[]	2023-09-15 20:00:01
196	610761771268869	HÒA CHUNG BẦU KHỔNG KHÍ VBA Finals 2023 - NHAN...	[]	2023-09-15 16:20:37

182 rows x 4 columns

```
df_text.to_csv("D:\\pythoncode\\project python\\VBA.vncomments.csv")
```

Sau quá trình đó ta thu lại được 182 bài post có giá trị sử dụng trong việc phân tích dữ liệu.

IV. Data Analysis and Data Visualization

Đọc hai file csv đã được xử lý từ bên trên:

```
df = pd.read_csv("../data_cleaned1.csv")  
df
```

```
df_cmt = pd.read_csv("../VBA.vncomments.csv")
```

Với yêu cầu của dự án bài đăng cuối cùng thu thập dữ liệu phải từ một tháng trở về trước nên em sẽ kiểm tra xem bài đăng cuối cùng mà ta crawl được là từ khoảng thời gian nào

```
> ▾
• #bài post được crawl cuối cùng
df.loc[df["time"].idxmin()]
[29] ✓ 0.0s

.. post_id                2107353809450125
   post_text            NÓI KHÔNG VỚI PHÂN BIỆT CHỦNG TỘC ✖\n=====...
   time                2022-08-12 11:44:14
   comments                3
   shares                3
   reactions    {'thích': 467, 'yêu thích': 226, 'haha': 1, 't...
   reaction_count        705
   thích                467
   thương thương        11
   yêu thích            226
   haha                1
   buồn                0
   wow                0
   phẫn nộ            0
   Name: 139, dtype: object
```

Bài đăng đã được up vào ngày 12 tháng 8 năm 2022 tức là vào khoảng hơn 1 năm về trước và bài đăng đầu tiên mà em lấy được đó là:

```
#bài post được crawl đầu tiên
df.loc[df["time"].idxmax()]

✓ 0.0s
```

post_id	649244514087261
post_text	Cùng VBA xem lại những hình ảnh đẹp mắt từ trậ...
time	2023-11-21 19:30:07
comments	7
shares	14
reactions	{'thích': 463, 'yêu thích': 104, 'wow': 1, 'th...
reaction_count	569
thích	463
thương thương	1
yêu thích	104
haha	0
buồn	0
wow	1
phấn nộ	0

Name: 0, dtype: object

Thông qua 2 bài đăng và việc crawl data em đã trình bày ở trên trong suốt hơn 1 năm Fanpage này đã update hơn 1000 bài viết nhưng trong quá trình cào dữ liệu và làm sạch em chỉ thu lại được khoảng 140 bài post có giá trị sử dụng.

Tiếp theo ta sẽ tìm xem đâu là bài post có lượt tương tác thấp và cao nhất:

```

#giá trị tương tác và bài post có lượt tương tác thấp nhất
min_reaction_count_value = df["reaction_count"].min()
min_reaction_count_value
12] ✓ 0.0s

.. 7

min_index = df["reaction_count"].idxmin()
min_data = df.loc[min_index]
min_data
13] ✓ 0.0s

.. post_id                2132650480253791
   post_text            HÀO HƯNG MỪNG SINH NHẬT CITIGYM PHỐ QUANG 🎉\n\...
   time                2022-09-14 10:00:02
   comments                0
   shares                0
   reactions              {'like': 7}
   reaction_count          7
   thích                0
   thương thương          0
   yêu thích              0
   haha                  0
   buồn                  0
   wow                   0
   phẫn nộ                0
   Name: 131, dtype: object

```

```
#giá trị tương tác và bài post có lượt tương tác cao nhất
min_reaction_count_value = df["reaction_count"].max()
min_reaction_count_value

14] ✓ 0.0s

.. 2819

max_index = df["reaction_count"].idxmax()
max_data = df.loc[max_index]
max_data

15] ✓ 0.0s

.. post_id 639913131687066
post_text Thời tiết tại The Global City đang rất nóng đấ...
time 2023-11-04 10:02:21
comments 123
shares 30
reactions {'thích': 1807, 'yêu thích': 959, 'haha': 10, ...
reaction_count 2819
thích 1807
thương thương 33
yêu thích 959
haha 10
buồn 1
wow 9
phấn nộ 0
Name: 32, dtype: object
```

Như ta có thể thấy sự chênh lệch tương tác giữa 2 post này là tương đối lớn với bài viết có lượt tương tác cao nhất có số lượng reactions là 2819 lượt tương tác còn bài post còn lại chỉ vồn vện có 7 lượt tương tác. Do đó có thể thấy những nội dung mà tập trung vào chính các thông tin mà giải đấu VBA diễn ra đem lại lượt lớn tương tác còn các nội dung không liên quan đến bóng rổ hay giải đấu như quảng cáo,... sẽ khá là flop.

Để có thể kiểm tra xem phân tích của ta có đúng hay không hay tiếp tục xem phần share để xem tương tác của page.

```
> ▾
#bài post có lượt share lớn nhất
df.loc[df["shares"].idxmax()]
[16] ✓ 0.0s

... post_id 637880188557027
post_text |MINIGAME| TRẢ LỜI ĐÚNG, NHẬN NGAY CẶP VÉ THAM...
time 2023-10-31 15:01:56
comments 262
shares 194
reactions {'thích': 202, 'yêu thích': 18, 'wow': 1, 'thư...
reaction_count 222
thích 202
thương thương 1
yêu thích 18
haha 0
buồn 0
wow 1
phấn nộ 0
Name: 44, dtype: object
```

```
> ▾
#bài post có lượt share thấp nhất
df.loc[df["shares"].idxmin()]
[30] ✓ 0.0s

... post_id 648554887489557
post_text HAPPY TEACHER'S DAY 20.11 🧡❤️\n\nNhân dịp Ngày...
time 2023-11-20 12:12:33
comments 0
shares 0
reactions {'thích': 64, 'yêu thích': 26}
reaction_count 90
thích 64
thương thương 0
yêu thích 26
haha 0
buồn 0
wow 0
phấn nộ 0
Name: 1, dtype: object
```

Đã nhắc đến lượt reactions, shares và để xem thêm phân tích của ta có đúng không hãy xem qua lượt comments:

```
#bài post có lượt comments cao nhất  
maxCmt = df.loc[df["comments"].idxmax()]  
maxCmt
```

```
post_id      621381610206885  
post_text    VBA AWARDS 2023 - MOMENT OF GLORY l Khoảnh khắc...  
time         2023-10-03 12:30:04  
comments     1578  
shares       30  
reactions    {'thích': 1067, 'yêu thích': 460, 'haha': 4, '...  
reaction_count 1542  
thích        1067  
thương thương 10  
yêu thích    460  
haha         4  
buồn         0  
wow          1  
phấn nộ      0  
Name: 91, dtype: object
```

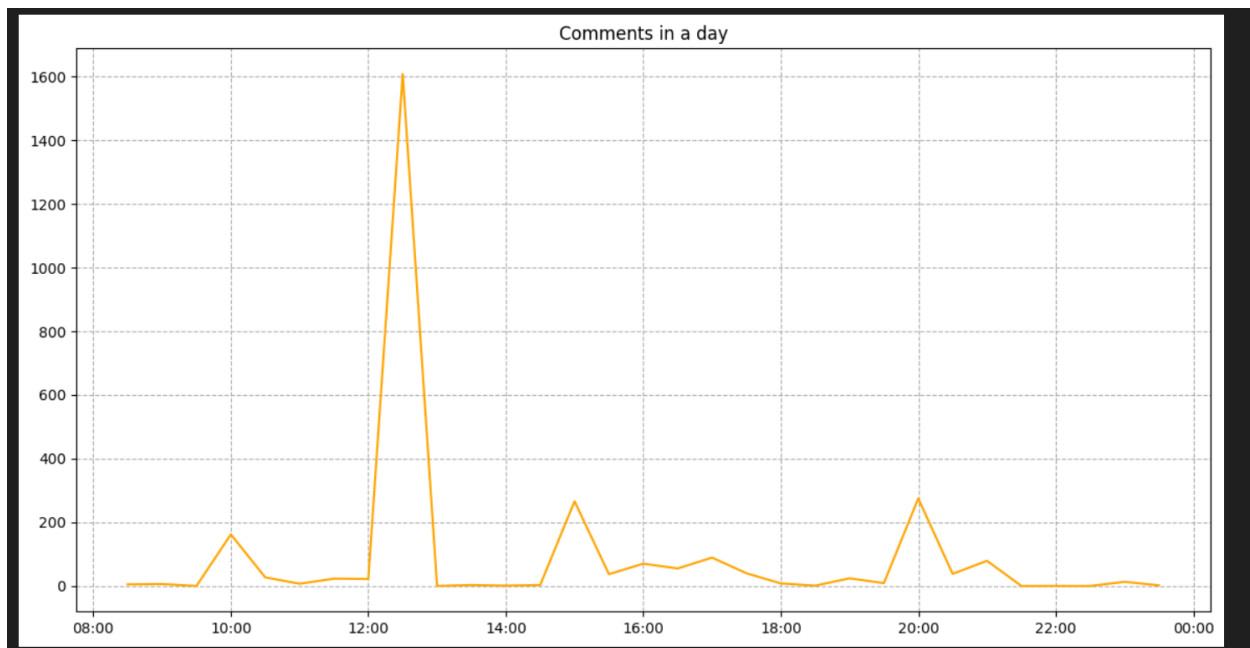
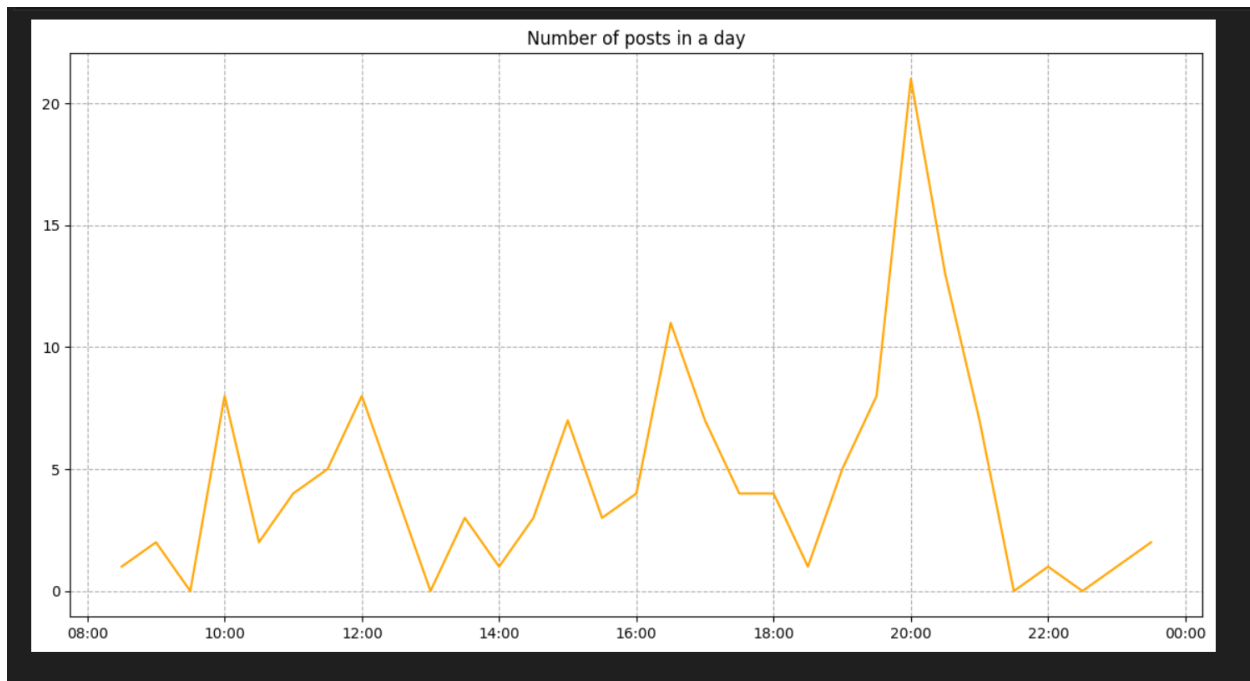


```
#bài post có lượt comment thấp nhất
minCmt = df.loc[df["comments"].idxmin()]
minCmt
```

```
... post_id 648554887489557
post_text  HAPPY TEACHER'S DAY 20.11 🧡❤️\n\nNhân dịp Ngày
time      2023-11-20 12:12:33
comments  0
shares    0
reactions {'thích': 64, 'yêu thích': 26}
reaction_count 90
thích        64
thương thương 0
yêu thích    26
haha         0
buồn         0
wow          0
phấn nộ     0
Name: 1, dtype: object
```

Có vẻ dựa trên dữ liệu mà ta thu thập được thì kết luận lượt tương tác của Fanpage dựa trên khá nhiều vào nội dung cốt lõi mà Fanpage này khá lớn. Qua đó có thể thấy đối tượng hướng đến của Fanpage là những người quan tâm đến bóng rổ nói chung hay giải đấu VBA nói riêng.

Tiếp đến sẽ là một số biểu đồ thể hiện lượt tương tác của Fanpage:



Qua hai biểu đồ trên ta có thể dễ dàng thấy được số lượng bài post đã được up phần lớn là trong khoảng thời gian 11h-12h và 19h-20h. Hai khoảng thời gian này là lúc người dùng đang trong lúc nghỉ ngơi vào khoảng thời gian nghỉ trưa và tối. Phần lớn bài đăng và lượt bình luận tập trung lớn vào khoảng thời gian này.

Quay lại với lượt reactions, ở đây em sẽ tính tổng lại tất cả các giá trị của trường reaction

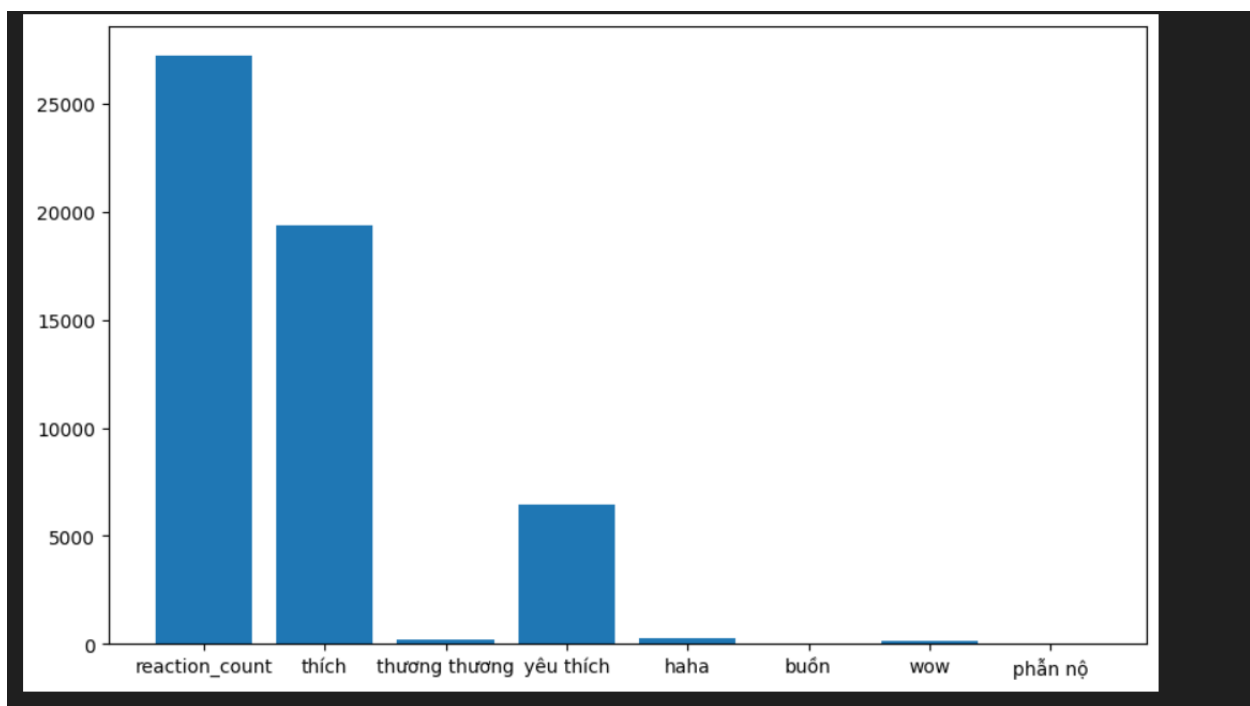
```
#tổng hợp lượng reactions
reactions_df = df.loc[:, ["reaction_count", "thích", "thương thương",
                           "yêu thích", "haha", "buồn", "wow", "phản nộ"]].agg(np.sum).reset_index()
reactions_df
```

[22] ✓ 0.0s

... C:\Users\asus\AppData\Local\Temp\ipykernel_30620\3050216990.py:2: FutureWarning: The provided callable <f
reactions_df = df.loc[:, ["reaction_count", "thích", "thương thương", "yêu thích", "haha", "buồn", "wow

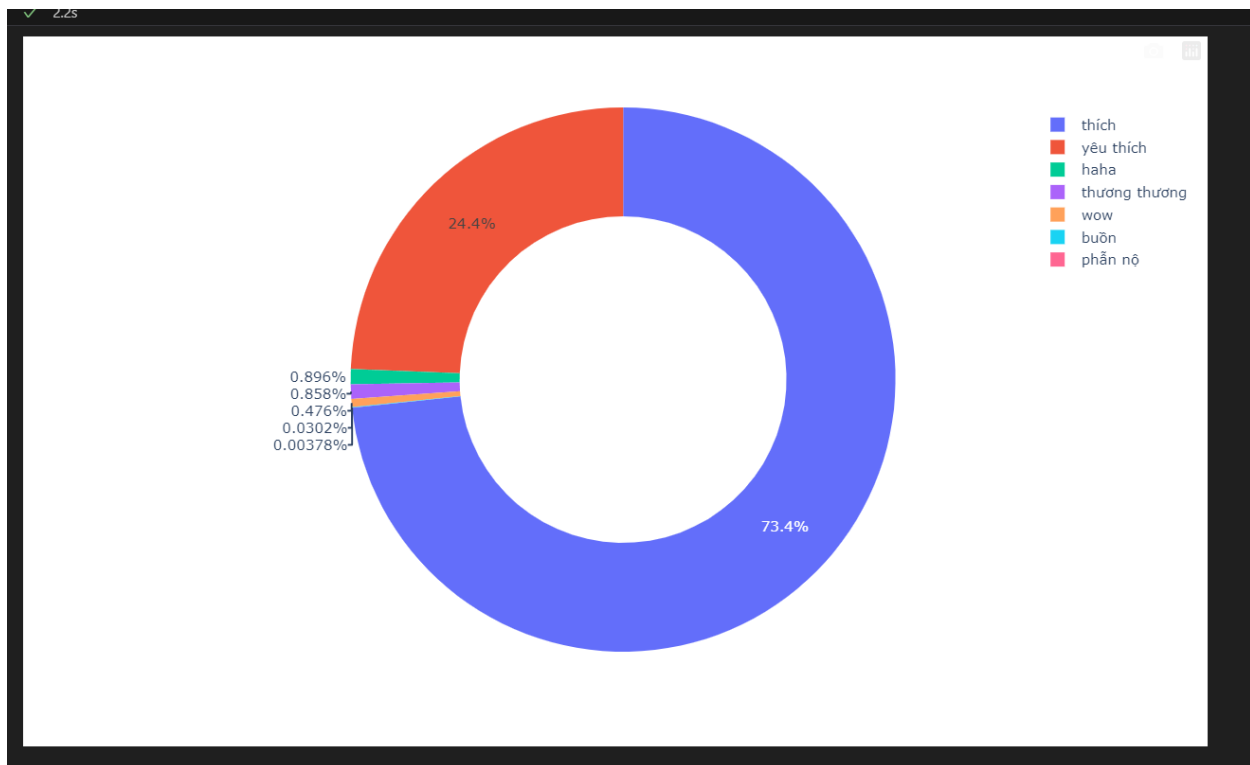
...

	index	0
0	reaction_count	27215
1	thích	19406
2	thương thương	227
3	yêu thích	6450
4	haha	237
5	buồn	8
6	wow	126
7	phản nộ	1

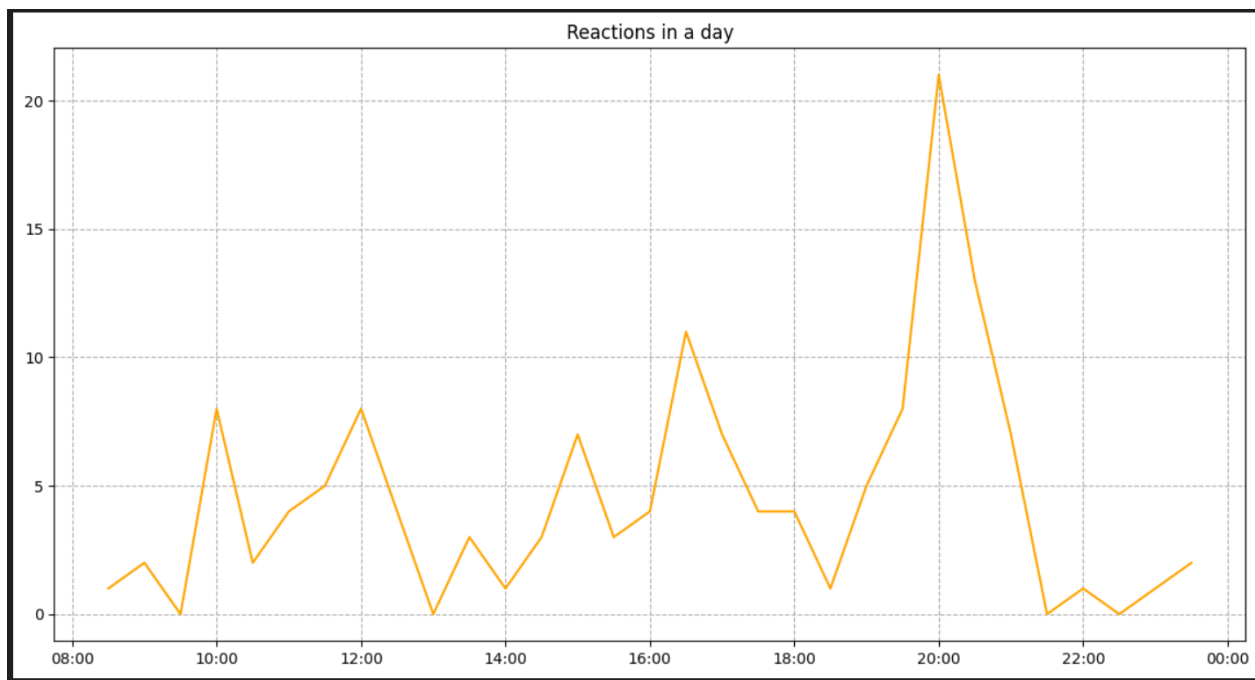


Qua biểu đồ có thể thấy lượt reactions chiếm phần lớn là “thích” và đặc biệt lượt “phản nộ”, “buồn” ở đây là rất thấp. Có thể đánh giá đây là một Fanpage cung cấp thông tin khá là “healthy” và cộng đồng ít toxic chủ yếu là cung cấp thông tin về giải đấu, lịch trình thi đấu và bán vé.

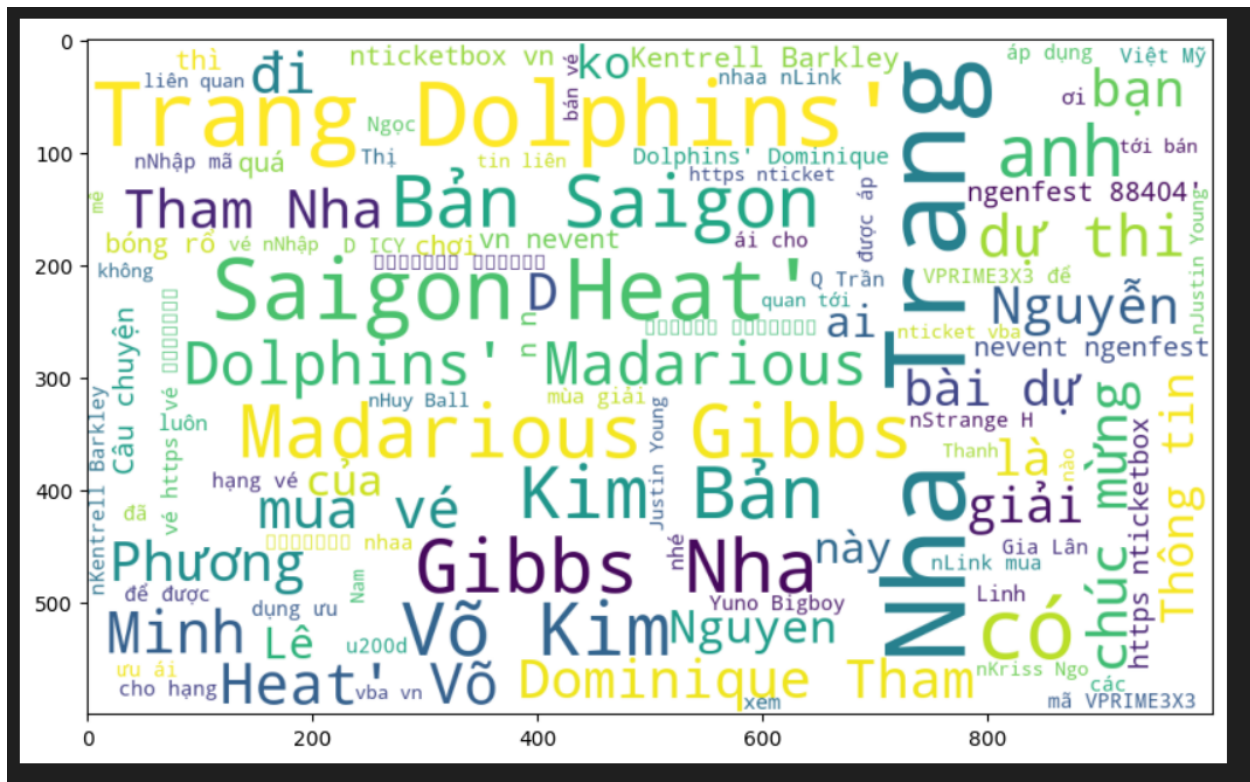
Sau đây là biểu đồ tỉ trọng của các lượt reaction:



Như đã nói ở trên phần lượt tương tác là “thích”, “yêu thích” còn các tương tác khác chiếm phần nhỏ hơn nhiều so 2 lượt tương tác kia.



Đây là phần từ khóa tìm kiếm nhiều nhất của “comment_text”:



21

một kênh chuyên cung cấp mọi thông tin về bóng rổ chuyên nghiệp Việt Nam tới người hâm mộ.

Fanpage được thành lập cùng với giải đấu nên còn khá non trẻ và là một page có độ tương tác tương đối ổn định đặc biệt là trong khoảng thời gian giải đấu và các vòng loại trực tiếp

Bên cạnh việc cập nhập thông tin về giải đấu thì Fanpage cũng thường đăng tải các bài viết về các nhà tài trợ hay các bài viết quảng cáo.

Việc phát triển Fanpage hay lượt tương tác lớn của Fanpage càng thể hiện bộ môn bóng rổ nói chung hay giải đấu chuyên nghiệp bóng rổ Việt Nam nói riêng ngày càng đang phát triển ở Việt Nam và càng ngày càng có nhiều người được tiếp cận hơn với bộ môn này.

Với cá nhân em là một người yêu thích bóng rổ, thể thao Việt Nam thì việc truyền thông và phát triển giải đấu góp phần không nhỏ trong việc nâng tầm thể thao nước nhà so với các khu vực Đông Nam Á, Châu Á hay rộng hơn là toàn thế giới.