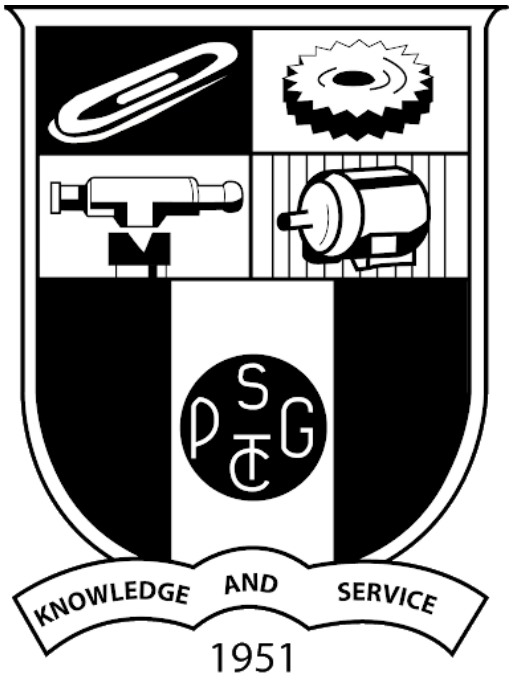
PSG COLLEGE OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



19OH01 Social and economic network analysis

Topic : Visualising Hashtag Culture in Instagram

Team Members
Haham Debbarma(18Z318)
Karthik Anand(18Z329)
Mokshith M(18Z334)
Ramraj Ramvardhan(18Z344)
Senthil Murugan A(18Z351)

Problem statement

The main aim of the project is to understand the hashtag culture on Instagram by using Graph Theory. By visualizing the use of hashtags in a post, we can observe which hashtags are often used together. We can get a deeper understanding of search terms related to a hashtag which weren't so obvious at first glance.

Dataset Description

Dataset consists of hashtags scraped using our own Selenium script. The inputs are cleaned while scraping so as to avoid any noise in the form of duplicate tags(both uppercase and lowercase) and in the form of emojis which do not play a role in the modelling.

Tools used

1. **Selenium and Chrome driver**: Selenium is used as a Web Browser Automation Tool. Selenium can be used to operate Chrome browser automatically. It helps us to Log in, Search for Tags and collect data without any human interference.

2. **Anaconda(Jupyter Notebook) for executing IPython notebook:** Anaconda is a data science tool which comes with many Libraries like pandas that helps in data manipulation.

3. **Networkx Library**: It is a python package for creation, manipulation, analysis of complex networks.

4. **Plotly Library:** Plotting interactive graphs like community graphs and sunburst plots.

5. **Git:** Used for version control and smooth collaboration of participants.

Challenges Faced:

1. The scrapping fetched a lot of noisy data like hashtags with emojis, Hashtags that were from other languages like arabic. Cleaning them and making them into a proper data frame was a challenge.
2. Searching for proper hashtags, which usually don't have unnecessary tags like instagood, ootd etc.,
3. Visualising Sunburst and community plots using plotly were challenging as the expected format in which the data frame has to be passed was varying for different kinds of graphs and the documentation was not very intuitive.

Contribution of team Members

Roll Number	Name	Contribution
18z318	Haham Debbarma	Centralities
18z339	Karthik Anand	Initial Graph Visualizations
18z334	Mokshith M	Community Detection
18z344	Ramraj Ramvardhan	Scraping hashtags
18z351	Senthil Murugan A	Sunburst plot visualisation

Annexure 1: Code

1. Code to iterate through links of posts and scrape tags while simultaneously cleaning

```
images = []
d={}
counter=0
import io
import codecs
file=codecs.open("Hashtags.txt","w","utf-8") #my file is output_log + the date time stamp
#follow each image link and extract only image at index=1
for a in anchors:
    driver.get(a)
    time.sleep(5)
    img = driver.find_elements_by_class_name('xil3i')
    #img = [img.get_attribute('text') for image in img]
    #print(img)
    lst=[]
    for i in img:
        temp=i.get_attribute('text')
        temp=temp.lower()
        temp = re.sub(r'\W+', '', temp)
        #file.write(str(temp[1:])+',')
        lst.append(str(temp))
        #print(temp)
    d[counter]=lst
        #print(lst)

    counter=counter+1

file.close()
```

2. Code to create the first Graph Visualization by creating edges between each tag in a post

```
for key in d:
    lst=d[key].split(' ')
    for i in range(len(lst)):
        lst[i]=lst[i].lower()
        for j in range(len(lst)):
            lst[j]=lst[j].lower()
            G_symmetric.add_edge(lst[i],lst[j])

k=3/math.sqrt(G_symmetric.order())
print(k)
pos = nx.spring_layout(G_symmetric,k)
nx.draw(G_symmetric, pos)
plt.show()
```

3. **Splitting into communities and building the DataFrame consisting of each tag, the community it belongs to and it's frequency. This DataFrame is used to build the Sunburst plot.**

From the community plot, we sort the tags that come under a community into a dictionary

```
communities = sorted(nxcom.greedy_modularity_communities(G_symmetric), key=len, reverse=True)
len(communities)
```

```
23
```

After we have the dictionary of tags, we form a dataframe with each hashtag and the community it belongs to

```
d={'tags':[],'community':[],'freq':[]}
for i in range(len(communities)):
    f=communities[i]
    for item in f:
        d['tags'].append(item)
        d['community'].append(i)
```

```
for item in d['tags']:
    try:
        d['freq'].append(freq[item])
    except:
        d['freq'].append(1)
```

Annexure 2: Snapshots of the output

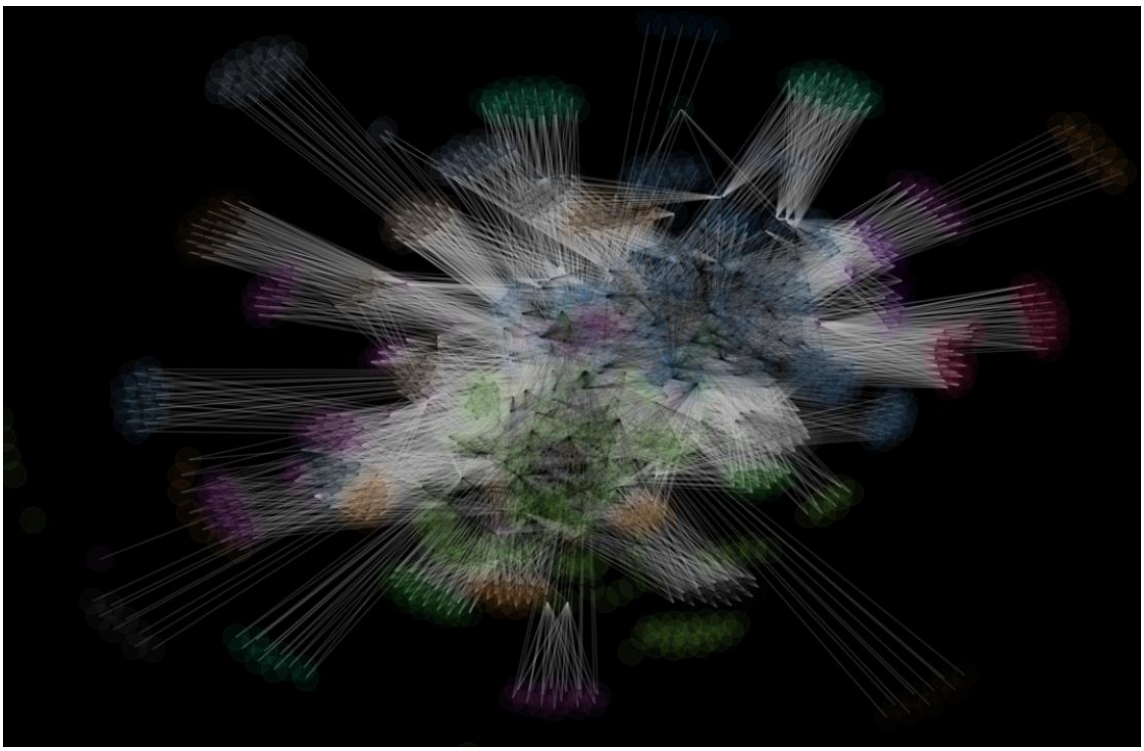


Fig. Community Plot

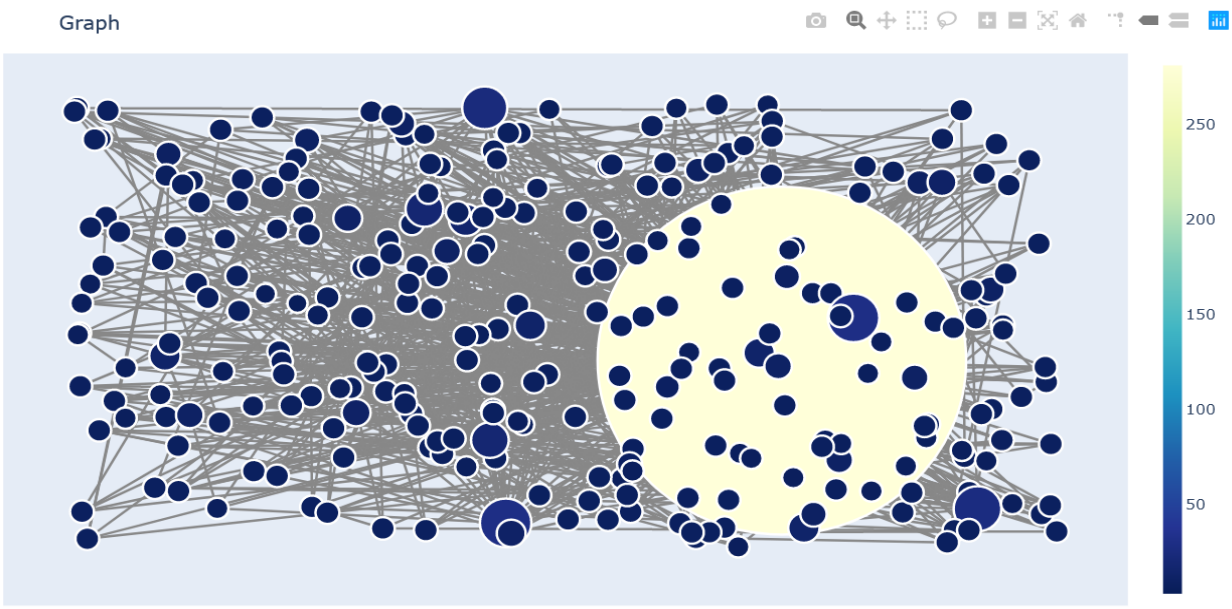


Fig. Frequency Visualization

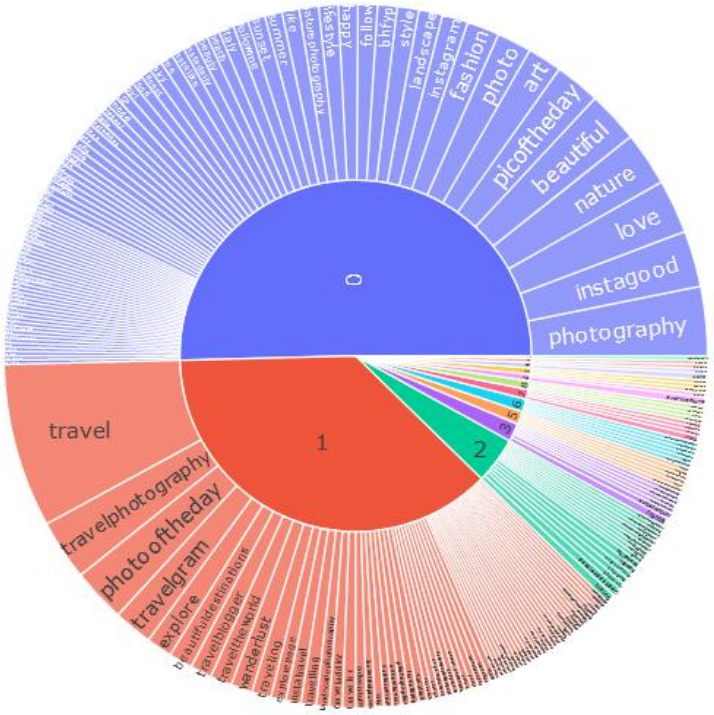


Fig. Sunburst Plot

References

1. https://medium.com/@srujana.rao2/scraping-instagram-with-python-using-selenium-and-beautiful-soup-8b72c186a058
2. https://plotly.com/python/sunburst-charts/
3. https://networkx.org/documentation/stable/reference/algorithms/centrality.html
4. https://networkx.org/documentation/networkx-1.10/reference/generated/networkx.drawing.nx_pylab.draw.html
5. https://www.geeksforgeeks.org/degree-centrality-centrality-measure/
6. https://plotly.com/python/network-graphs/
7. https://towardsdatascience.com/interactively-plot-graph-networks-with-igviz-c75da26858ec
8. https://pypi.org/project/igviz/#introduction