

```
In [1]: from google.colab import drive  
drive.mount('/content/drive')
```

Mounted at /content/drive

```
In [2]: !pip install cdlib  
import sys  
!{sys.executable} -m pip install python-igraph  
!{sys.executable} -m pip install leidenalg
```

## Collecting cdlib

Downloading <https://files.pythonhosted.org/packages/e7/df/2c52a76f23e6801a12f0b67ba1523a167f808629dbb0feb7bf491efb1925/cdlib-0.2.0-py3-none-any.whl> (152kB)

153kB 10.7MB/s

Requirement already satisfied: matplotlib>=3.0.\* in /usr/local/lib/python3.7/dist-packages (from cdlib) (3.2.2)

## Collecting chinese-whispers

Downloading [https://files.pythonhosted.org/packages/c7/9e/f5055fbfd95fd9aab0510065aed2ae520dc639e0b8fde945ce3ce23de21e/chinese\\_whispers-0.7.4-py3-none-any.whl](https://files.pythonhosted.org/packages/c7/9e/f5055fbfd95fd9aab0510065aed2ae520dc639e0b8fde945ce3ce23de21e/chinese_whispers-0.7.4-py3-none-any.whl)

Collecting shuffle-graph==1.1.1

Downloading [https://files.pythonhosted.org/packages/18/14/22153a8d389370cae50bedd58239f640afe9717085a7a4e9bc5f801886aa/shuffle\\_graph-1.1.1-py3-none-any.whl](https://files.pythonhosted.org/packages/18/14/22153a8d389370cae50bedd58239f640afe9717085a7a4e9bc5f801886aa/shuffle_graph-1.1.1-py3-none-any.whl)

Collecting omega-index-py3

Downloading [https://files.pythonhosted.org/packages/fb/57/ab48f801af9ef0ce273fe1969df825a99f52a73ff18767c7084f60471028/omega\\_index\\_py3-0.3-py3-none-any.whl](https://files.pythonhosted.org/packages/fb/57/ab48f801af9ef0ce273fe1969df825a99f52a73ff18767c7084f60471028/omega_index_py3-0.3-py3-none-any.whl)

Requirement already satisfied: seaborn&gt;=0.9.\* in /usr/local/lib/python3.7/dist-packages (from cdlib) (0.11.1)

Collecting ASLPAw==2.0.0

Downloading <https://files.pythonhosted.org/packages/af/a4/67dcbe51833c044c60a892e4ab5ca90a73e77317f0dcf377e2f9c6451922/ASLPAW-2.0.0-py3-none-any.whl>

Collecting nf1

Downloading <https://files.pythonhosted.org/packages/4c/41/18e9d6c802db1084c157a0d6c5b38be5878f925e1cf53d7eb03edc7ec6ad/nf1-0.0.3-py3-none-any.whl>

Collecting pquality==0.0.7

Downloading <https://files.pythonhosted.org/packages/12/f8/5357fd9dfe0f105fea5439fcb01a21b8550a71d0e0b702b2bc0372c953c4/pquality-0.0.7-py3-none-any.whl>

## Collecting dynetx

Downloading <https://files.pythonhosted.org/packages/a9/2d/ce4a7e812ac8a26ed7a1872b6974391dd91dbdfc8845baae038becbf66f3/dynetx-0.2.4-py3-none-any.whl>

```
Collecting demon>=2.0.5
```

Downloading <https://files.pythonhosted.org/packages/87/21/e84965594c51885a2f3b5f6ff224ba58f932cc66241489e0d763b9ac531b/demon-2.0.5-py3-none-any.whl>

Collecting bimlpa

Downloading <https://files.pythonhosted.org/packages/11/8c/7d409d7e7bcbd261c7f19737feaa1ef66514512c7f7d6b9a776d7aac7c33/bimlpa-0.1.2-py3-none-any.whl>

Requirement already satisfied: tqdm>=4.20.\* in /usr/local/lib/python3.7/dist-packages (from cdlib) (4.41.1)

Collecting python-louvain==0.14

Downloading <https://files.pythonhosted.org/packages/31/d4/d244fa4ca96af747b9204ce500e8919ce72eab60de2129b22a45434f1598/python-louvain-0.14.tar.gz>

Requirement already satisfied: scipy>=1.3.\* in /usr/local/lib/python3.7/dist-packages (from cdlib) (1.4.1)

```
Collecting karateclub>=1.0.0
```

Downloading <https://files.pythonhosted.org/packages/f1/89/bc10c72e8047c801b785642bf58317022b0ef48ecd3e7db3a78aa23835ea/karateclub-1.0.24.tar.gz> (59kB)

```

|████████████████████████████████████████| 61kB 3.7MB/s
Requirement already satisfied: numpy>=1.15.* in /usr/local/lib/python3.7/dist-packages (from cdlib) (1.19.5)
Requirement already satisfied: pandas>=0.24 in /usr/local/lib/python3.7/dist-packages (from cdlib) (1.1.5)
Requirement already satisfied: future>=0.16.* in /usr/local/lib/python3.7/dist-packages (from cdlib) (0.16.0)
Collecting markov-clustering
  Downloading https://files.pythonhosted.org/packages/76/42/19e11a42fa952d35116b90577e2cde31c541ce78364a52167f852864ba29/markov_clustering-0.0.6.dev0-py3-none-any.whl
Requirement already satisfied: pooch in /usr/local/lib/python3.7/dist-packages (from cdlib) (1.3.0)
Collecting eva-lcd
  Downloading https://files.pythonhosted.org/packages/e2/a4/f1f79a5c964831e2d3864e159c433cab5a8f420017e3492d615415c4ac24/eva_lcd-0.1.0-py3-none-any.whl
Collecting pulp>=2.1
  Downloading https://files.pythonhosted.org/packages/14/c4/0eec14a0123209c261de6ff154ef3be5cad3fd557c084f468356662e0585/PuLP-2.4-py3-none-any.whl (40.6MB)
|████████████████████████████████████████| 40.6MB 100kB/s
Requirement already satisfied: scikit-learn<0.24,>=0.21.* in /usr/local/lib/python3.7/dist-packages (from cdlib) (0.22.2.post1)
Requirement already satisfied: networkx>=2.4 in /usr/local/lib/python3.7/dist-packages (from cdlib) (2.5)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib>=3.0.*->cdlib) (1.3.1)
Requirement already satisfied: cyclor>=0.10 in /usr/local/lib/python3.7/dist-packages (from matplotlib>=3.0.*->cdlib) (0.10.0)
Requirement already satisfied: pyparsing!=2.0.4,!2.1.2,!2.1.6,>=2.0.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib>=3.0.*->cdlib) (2.4.7)
Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib>=3.0.*->cdlib) (2.8.1)
Collecting count-dict>=1.0.1
  Downloading https://files.pythonhosted.org/packages/b4/99/e46091f8f5f09ec4346e9b59da4f31838345878b54da0c102ac0be9c6757/count_dict-1.1.1-py3-none-any.whl
Collecting multivalued-dict>=1.7.1
  Downloading https://files.pythonhosted.org/packages/38/d9/60639ea482acbf0d2defec1d2ab0f58f42bed65985a9e0beb5b9c6a70887/multivalued_dict-2.0.1-py3-none-any.whl
Collecting pygsp
  Downloading https://files.pythonhosted.org/packages/d4/89/2f4aa73cccf12bec5179ac5d52a68b508120c838b7e5d456f5ea0c8beade/PyGSP-0.5.1-py2.py3-none-any.whl (1.8MB)
|████████████████████████████████████████| 1.8MB 44.1MB/s
Collecting gensim==3.8.3
  Downloading https://files.pythonhosted.org/packages/5c/4e/afe2315e08a38967f8a3036bbe7e38b428e9b7a90e823a83d0d49df1adf5/gensim-3.8.3-cp37-cp37m-manylinux1_x86_64.whl (24.2MB)
|████████████████████████████████████████| 24.2MB 1.1MB/s
Requirement already satisfied: six in /usr/local/lib/python3.7/dist-packages (from karateclub>=1.0.0->cdlib) (1.15.0)
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.7/dist-packages (from pandas>=0.24->cdlib) (201

```

8.9)

Requirement already satisfied: appdirs in /usr/local/lib/python3.7/dist-packages (from pooch->cdlib) (1.4.4)

Requirement already satisfied: packaging in /usr/local/lib/python3.7/dist-packages (from pooch->cdlib) (20.9)

Requirement already satisfied: requests in /usr/local/lib/python3.7/dist-packages (from pooch->cdlib) (2.23.0)

Collecting amply>=0.1.2

Downloading <https://files.pythonhosted.org/packages/f3/c5/dfa09dd2595a2ab2ab4e6fa7bebef9565812722e1980d04b0edce5032066/amply-0.1.4-py3-none-any.whl>

Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.7/dist-packages (from scikit-learn<0.24,>=0.21.\*->cdlib) (1.0.1)

Requirement already satisfied: decorator>=4.3.0 in /usr/local/lib/python3.7/dist-packages (from networkx>=2.4->cdlib) (4.4.2)

Requirement already satisfied: smart-open>=1.8.1 in /usr/local/lib/python3.7/dist-packages (from gensim==3.8.3->karateclub>=1.0.0->cdlib) (4.2.0)

Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-packages (from requests->pooch->cdlib) (2.10)

Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-packages (from requests->pooch->cdlib) (2020.12.5)

Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/dist-packages (from requests->pooch->cdlib) (3.0.4)

Requirement already satisfied: urllib3!=1.25.0,!1.25.1,<1.26,>=1.21.1 in /usr/local/lib/python3.7/dist-packages (from requests->pooch->cdlib) (1.24.3)

Requirement already satisfied: docutils>=0.3 in /usr/local/lib/python3.7/dist-packages (from amply>=0.1.2->pulp>=2.1->cdlib) (0.16)

Building wheels for collected packages: python-louvain, karateclub

Building wheel for python-louvain (setup.py) ... done

Created wheel for python-louvain: filename=python\_louvain-0.14-cp37-none-any.whl size=9292 sha256=7ab7561e49bd568c3fc67c6336b45278ea6e7e6f1612f152bc47e95d351cbde5

Stored in directory: /root/.cache/pip/wheels/e7/8d/24/6b3a464bb23e96ecba3f68868e85721534fd8158a9cd7b426b

Building wheel for karateclub (setup.py) ... done

Created wheel for karateclub: filename=karateclub-1.0.24-cp37-none-any.whl size=94202 sha256=4b223293988aba51431a7ae61c46cbac6dc08f3d9be340003f0072c426c0f660

Stored in directory: /root/.cache/pip/wheels/ab/a4/81/a2761ff51bb1caa318e69e4a30f5d7d39c14f10efe73365279

Successfully built python-louvain karateclub

Installing collected packages: chinese-whispers, shuffle-graph, omega-index-py3, count-dict, multivalued-dict, ASLPAw, nf1, pquality, dynetx, demon, bimlpa, python-louvain, pygsp, gensim, karateclub, markov-clustering, eva-lcd, amply, pulp, cdlib

Found existing installation: python-louvain 0.15

Uninstalling python-louvain-0.15:

Successfully uninstalled python-louvain-0.15

Found existing installation: gensim 3.6.0

Uninstalling gensim-3.6.0:

Successfully uninstalled gensim-3.6.0

Successfully installed ASLPAw-2.0.0 amply-0.1.4 bimlpa-0.1.2 cdlib-0.2.0 chinese-whispers-0.7.4 count-dict-1.1.1 demo  
n-2.0.5 dynetx-0.2.4 eva-lcd-0.1.0 gensim-3.8.3 karateclub-1.0.24 markov-clustering-0.0.6.dev0 multivalued-dict-2.0.1  
nf1-0.0.3 omega-index-py3-0.3 pquality-0.0.7 pulp-2.4 pygsp-0.5.1 python-louvain-0.14 shuffle-graph-1.1.1

Collecting python-igraph

Downloading [https://files.pythonhosted.org/packages/ae/12/1fbdb491d89fad8abb7aca0189978655cfdc984a380b846478f2ccdfdad8/python\\_igraph-0.9.1-cp37-cp37m-manylinux2010\\_x86\\_64.whl](https://files.pythonhosted.org/packages/ae/12/1fbdb491d89fad8abb7aca0189978655cfdc984a380b846478f2ccdfdad8/python_igraph-0.9.1-cp37-cp37m-manylinux2010_x86_64.whl) (3.2MB)

|██| 3.2MB 8.4MB/s

Collecting texttable>=1.6.2

Downloading <https://files.pythonhosted.org/packages/06/f5/46201c428aeb0e0ecfa83df66bf3e6caa29659dbac5a56ddfd83cae0d4a4/texttable-1.6.3-py2.py3-none-any.whl>

Installing collected packages: texttable, python-igraph

Successfully installed python-igraph-0.9.1 texttable-1.6.3

Collecting leidenalg

Downloading [https://files.pythonhosted.org/packages/70/68/3f7c56da3e7ac576e5e090f2a7e8637f96e4a790688dbec63b938530ef15/leidenalg-0.8.3-cp37-cp37m-manylinux2010\\_x86\\_64.whl](https://files.pythonhosted.org/packages/70/68/3f7c56da3e7ac576e5e090f2a7e8637f96e4a790688dbec63b938530ef15/leidenalg-0.8.3-cp37-cp37m-manylinux2010_x86_64.whl) (2.4MB)

|██| 2.4MB 9.6MB/s

Requirement already satisfied: python-igraph>=0.8.0 in /usr/local/lib/python3.7/dist-packages (from leidenalg) (0.9.1)

Requirement already satisfied: texttable>=1.6.2 in /usr/local/lib/python3.7/dist-packages (from python-igraph>=0.8.0->leidenalg) (1.6.3)

Installing collected packages: leidenalg

Successfully installed leidenalg-0.8.3

```
In [3]: import pandas as pd
import networkx as nx
from networkx.algorithms.community centrality import girvan_newman
from cdlib import algorithms
import numpy as np
import pickle
from networkx.algorithms.link_analysis.pagerank_alg import pagerank
from tqdm.autonotebook import tqdm
import pickle
from PIL import Image
import requests
from io import BytesIO
```

/usr/local/lib/python3.7/dist-packages/ipykernel\_launcher.py:8: TqdmExperimentalWarning: Using `tqdm.autonotebook.tqdm` in notebook mode. Use `tqdm.tqdm` instead to force console mode (e.g. in jupyter console)

```
In [4]: path = "/content/drive/My Drive/"
project_name="2_TwitterFollowGraph"
dataframe = pd.read_csv(path+project_name+"/Datasets/Facebook/musae_facebook_edges.csv")
columns=list(dataframe.columns)
```

```
In [5]: def createGraph(dataframe,columns):
    edgelist = dataframe[columns].values.tolist()
    graph = nx.Graph()
    graph.add_edges_from(edgelist)
    return graph
```

```
In [6]: graph = createGraph(dataframe,columns)
```

```
In [ ]: def sortedCommunities(communities):
    communitiesSorted = sorted(communities,key=lambda x:len(x),reverse=True)
    return communitiesSorted
```

```
In [ ]: def topDegreeNodeCommunities(communities,graph,size):
    degree = graph.degree()
    newCommunityList = []
    for community in communities:
        newCommunity = sorted(community,key=lambda x:degree[x],reverse=True)
        newCommunityList.append(newCommunity[:size])
    return newCommunityList
```

```
In [ ]: def generateCommunities(graph,method):
    if method == "leiden":
        communities = algorithms.leiden(graph)
    elif method == "walktrap":
        communities = algorithms.walktrap(graph)
    elif method == "surprise_communities":
        communities = algorithms.surprise_communities(graph)
    communitiesSorted = sortedCommunities(communities.communities)
    topNodeCommunities = topDegreeNodeCommunities(communitiesSorted,graph,30) # Set third parameter to get required size
    return topNodeCommunities
```

```
In [7]: community_features_df=pd.read_csv(path+project_name+"/Datasets/Facebook/musae_facebook_target.csv")
community_features_df.head()
```

Out[7]:

	id	facebook_id	page_name	page_type
0	0	145647315578475	The Voice of China 中国好声音	tvshow
1	1	191483281412	U.S. Consulate General Mumbai	government
2	2	144761358898518	ESET	company
3	3	568700043198473	Consulate General of Switzerland in Montreal	government
4	4	1408935539376139	Mark Bailey MP - Labor for Miller	politician

```
In [ ]: def filter_communities(communities_list,graph):
        final_communities_list=[]
        for communities in communities_list:
            temp_list=[]
            for node_id in communities:
                features=community_features_df[community_features_df["id"]==node_id]
                temp_list.append((features["page_name"].values[0],features["page_type"].values[0],graph.degree(node_id)))
            if len(temp_list)>1:
                final_communities_list.append(temp_list)
        return final_communities_list
```

```
In [8]: communities_dict={}
page_rank_dict={}
```



```
In [ ]: #Leiden community algorithm
communities_list=generateCommunities(graph,"leiden")
final_communities_list=filter_communities(communities_list,graph)
communities_dict["leiden"]=final_communities_list
print(final_communities_list)
```

[[('The White House', 'government', 678), ('Joachim Herrmann', 'politician', 320), ('National Oceanic and Atmospheric Administration (NOAA)', 'government', 288), ('Marine Corps Recruiting', 'government', 257), ('U.S. Army SHARP', 'government', 256), ('Martin Schulz', 'politician', 236), ('The National Guard', 'government', 236), ('Arno Klare MdB', 'politician', 226), ('Angela Merkel', 'politician', 217), ('Resolute Support Mission', 'government', 201), ('Soldier for Life', 'government', 194), ('U.S. Mission to the United Nations', 'government', 193), ('NOAA Digital Coast', 'government', 188), ('Army OneSource', 'government', 182), ('NOAA NWS Storm Prediction Center', 'government', 177), ('Marie-Claude Bibeau', 'politician', 174), ('USA.gov', 'government', 166), ('European Commission - Development & Cooperation - EuropeAid', 'government', 162), ('European Committee of the Regions', 'government', 157), ('Norfolk District, U.S. Army Corps of Engineers', 'government', 156), ('Army Study Program Management Office - ASPMO', 'government', 155), ('Defense Finance and Accounting Service - DFAS', 'government', 155), ('Malcolm Turnbull', 'politician', 148), ('NOAA NWS Weather Prediction Center', 'government', 148), ('Europe District, U.S. Army Corps of Engineers', 'government', 147), ('NOAA National Severe Storms Laboratory', 'government', 146), ('NSW SES', 'government', 142), ('U.S. Department of Commerce', 'government', 135), ('U.S. Coast Guard Maritime Safety & Security Team - 91105', 'government', 134), ('Senado Federal', 'government', 129)], [('European Commission', 'government', 330), ('Senate of Canada - Sénat du Canada', 'government', 266), ('Military OneSource', 'government', 257), ('EU Law and Publications', 'government', 212), ('U.S. Army Reserve', 'government', 201), ('Auswärtiges Amt', 'government', 178), ('Exchange Programs - U.S. Department of State', 'government', 147), ('U.S. Department of Health and Human Services', 'government', 146), ('U.S. Army Partnership for Youth Success (PaYS) Program', 'government', 141), ('U.S. Mission Vienna', 'government', 122), ('tagesschau', 'tvshow', 119), ('U.S. Embassy Phnom Penh, Cambodia', 'government', 117), ('India in Myanmar (Embassy of India, Yangon)', 'government', 115), ('NORAD and USNORTHERN Command', 'government', 114), ('União Europeia na Guiné-Bissau', 'government', 110), ('Secretaria Nacional de Aviação Civil', 'government', 108), ('3rd Infantry Division', 'government', 102), ('U.S. Embassy Canberra', 'government', 102), ('European Union in the Philippines', 'government', 100), ('U.S. Army Combined Arms Center', 'government', 96), ('Presidencia de la República de Colombia', 'government', 94), ('Ministère de l'Économie et des Finances, de l'Action et des Comptes publics', 'government', 93), ('Family Guy', 'tvshow', 91), ('Ministério do Turismo', 'government', 90), ('NSW SES - Mudgee Unit', 'government', 87), ('HealthCare.gov', 'government', 84), ('Juan Manuel Santos - Presidente', 'politician', 81), ('Senator Denise Batters', 'politician', 80), ('Jerzy Buzek', 'politician', 79), ('Senator James Paterson', 'politician', 79)], [('FEMA Federal Emergency Management Agency', 'government', 448), ('Los Angeles District, U.S. Army Corps of Engineers', 'government', 370), ('Barack Obama', 'politician', 341), ('U.S. Environmental Protection Agency', 'government', 240), ('U.S. Geological Survey (USGS)', 'government', 152), ('Dr. Hedy Fry', 'politician', 148), ('Military Health', 'government', 143), ('Today Show', 'tvshow', 141), ('Peace Corps', 'government', 138), ('1st Infantry Division', 'government', 135), ('III Corps and Fort Hood', 'government', 131), ('British Embassy Lisbon', 'government', 117), ('Peter Tauber', 'politician', 111), ('XVIII Airborne Corps', 'government', 107), ('Tanja Fajon', 'politician', 107), ('Starbucks', 'company', 106), ('Maxime Bernier', 'politician', 106), ('European Union in Indonesia', 'government', 105), ('SAMHSA', 'government', 104), ('Qantas', 'company', 102), ('U.S. Embassy Denmark', 'government', 96), ('Windows', 'company', 94), ('U.S. Embassy Pristina, Kosovo', 'government', 93), ('Andrew Scheer', 'politician', 93), ('DHS Science and Technology Directorate', 'government', 90), ('Anna Burke', 'politician', 88), ('Hell's Kitchen', 'tvshow', 83), ('smartraveller.gov.au', 'government', 83), ('Michael E. DeBakey VA Medical Center', 'government', 82), ('Tony Burke MP', 'politician', 80)], [('U.S. Army Chaplain Corps', 'government', 650), ('Defense Commissary Agency', 'government', 387), ('National Park Service', 'government', 244), ('USAID - US Agency for International Development', 'government', 234), ('AFOSR, Air Force Office of Scientific Research', 'government', 222), ('NATO', 'government', 222), ('Niels Annen', 'politician', 19

file:///E:/M.Tech/Semester-4/Network Science/Project/CodeFiles/html files/CommunitiesPageRankFacebook.html

t', 103), ('Canada and the world', 'government', 101), ('Fort Knox Soldier for Life - Transition Assistance Program', 'government', 101), ('Tim Watts MP', 'politician', 101), ('So You Think You Can Dance', 'tvshow', 99), ('Израел в България', 'government', 99), ('Picatinny Arsenal', 'government', 99), ('NSW SES Liverpool', 'government', 98), ('USAID Peru', 'government', 96), ('Representação da Comissão Europeia em Portugal', 'government', 94), ('Le Canada et le monde', 'government', 93), ('EU Justice and Consumers', 'government', 92), ('Ministério da Cultura', 'government', 92), ('NSW SES Albury Unit', 'government', 88), ('Defense Information Systems Agency', 'government', 88)], [('Honolulu District, U.S. Army Corps of Engineers', 'government', 504), ('Army Training Network (ATN)', 'government', 408), ('Army Sustainment', 'government', 298), ('U.S. Army Office of the Chief of Public Affairs', 'government', 241), ('U.S. Central Command', 'government', 218), ('U.S. Army Research, Development and Engineering Command', 'government', 196), ('Israel en Guatemala', 'government', 177), ('173rd Airborne Brigade', 'government', 142), ('82nd Airborne Division', 'government', 135), ('Council of Europe', 'government', 131), ('Joint Services Support- Washington', 'government', 126), ('Foreign Office', 'government', 120), ('Michel Picard', 'politician', 116), ('Yasmin Fahimi', 'politician', 113), ('The Post-9/11 GI Bill, U.S. Department of Veterans Affairs', 'government', 108), ('MASHAV', 'government', 105), ('Heidtrud Henn', 'politician', 105), ('U.S. Army Public Affairs, Midwest', 'government', 104), ('Fort Bragg, N.C.', 'government', 99), ('Spouse Education and Career Opportunities Program - SECO', 'government', 99), ('Tanya Plibersek', 'politician', 98), ('Tomah VA Medical Center', 'government', 97), ('Target', 'company', 97), ('Tony Abbott', 'politician', 95), ('Presidencia de la República', 'government', 93), ('Strathfield State Emergency Service (SES)', 'government', 91), ('NSW SES Fairfield Unit', 'government', 90), ('Mariano Rajoy Brey', 'politician', 90), ('Australian Federal Police', 'government', 89), ('Fort Wainwright', 'government', 84)], [('Defense Logistics Agency (DLA)', 'government', 333), ('Council of the European Union', 'government', 201), ('European Commission - Civil Protection & Humanitarian Aid Operations - ECHO', 'government', 191), ('US National Weather Service Burlington VT', 'government', 155), ('Israel at the UN', 'government', 112), ('Colorado National Guard', 'government', 108), ('American English at State', 'government', 106), ('Air National Guard Recruiting', 'government', 101), ('Laurent Fabius', 'politician', 100), ('Canada in the UAE', 'government', 99), ('U.S. Marine Forces Europe and Africa', 'government', 94), ('Europäische Kommission - Vertretung in Deutschland', 'government', 94), ('U.S. Army Central', 'government', 91), ('MasterChef', 'tvshow', 90), ('Headquarters Support Company, 603rd Aviation Support Battalion', 'government', 90), ('U.S. Embassy London', 'government', 87), ('Ted Falk', 'politician', 86), ('Rock Island Arsenal', 'government', 86), ('1st Brigade Combat Team, 82nd Airborne Division', 'government', 85), ('Geovania de Sá', 'politician', 82), ('Bundeswehr', 'government', 78), ('The NOAA Hurricane Hunters', 'government', 77), ('Penny Wong - Senator for SA', 'politician', 72), ('Jess Pugh for Mount Omaney', 'politician', 71), ('Access', 'tvshow', 71), ('US Army Corps of Engineers, Buffalo District', 'government', 69), ('Ambassade de France au Canada', 'government', 68), ('Ed Husic MP', 'politician', 67), ('Ministère des Outre-Mer', 'government', 65), ('U.S. Embassy Moscow', 'government', 64)], [('European Parliament', 'government', 417), ('U.S. National Weather Service (NWS)', 'government', 296), ('European External Action Service - EEAS', 'government', 237), ('TRICARE', 'government', 220), ('US National Weather Service Chicago Illinois', 'government', 216), ('2nd Theater Signal Brigade', 'government', 214), ('U.S. Africa Command (AFRICOM)', 'government', 177), ('Pamela Goldsmith-Jones', 'politician', 163), ('Achim Post', 'politician', 144), ('Israel Ministry of Foreign Affairs', 'government', 129), ('US National Weather Service Midland Texas', 'government', 127), ('Art in Embassies, US Department of State', 'government', 122), ('NYC Department of Small Business Services', 'government', 114), ('US Army Combat Readiness Center', 'government', 105), ('Cathy O'Toole MP', 'politician', 94), ('Fort Riley', 'government', 92), ('DFID - UK Department for International Development', 'government', 91), ('7th Infantry Division', 'government', 88), ('Ministério da Educação - MEC', 'government', 88), ('Préfet du Val-d'Oise', 'government', 86), ('Chris Crewther MP - Federal Member for Dunkley', 'politician', 74), ('U.S. Army Corps of Engineers Great Lakes and Ohio River Division', 'government', 74)

a embajada u BiH', 'government', 14), ('Coffee-Mate México', 'company', 2)], [('Ambassade de France en Estonie / Prantsuse Suursaatkond Eestis', 'government', 16), ('Microsoft Developer', 'company', 4)], [('Euroopan komissio Suomessa - Europeiska kommissionen i Finland', 'government', 64), ('Alisto', 'tvshow', 3)], [('SUPERMUM', 'tvshow', 5), ('FTO Colorado River Agency', 'government', 2)], [('Antonio Miloshoski', 'politician', 4), ('Walulis', 'tvshow', 3)], [('Australian High Commission, Ghana', 'government', 43), ('The Voice - Magyarország hangja', 'tvshow', 2)], [('WD', 'company', 20), ('Dorothee Martin', 'politician', 6)], [('Baskets', 'tvshow', 13), ('Kateřina Konečná', 'politician', 4)], [('Teletica Deportes', 'tvshow', 10), ('Komplett', 'company', 6)], [('35th Infantry Division', 'government', 18), ('College Hill', 'tvshow', 6)], [('Polský institut v Praze', 'government', 8), ('ISU ARAD', 'government', 6)], [('Sandro Gozi', 'politician', 23), ('The Powerpuff Girls', 'tvshow', 10)], [('Decatur Active Living', 'government', 9), ('Radio Atlántida 102.7 FM', 'company', 1)], [('Corporation for National and Community Service', 'government', 47), ('L'Occitane en Provence', 'company', 25)], [('TNT Drama', 'tvshow', 20), ('Air New Zealand', 'company', 9)], [('Margarita Zavala', 'politician', 10), ('Daniel Günther', 'politician', 9)], [('זוהיר בהלול זهير بهلول', 'politician', 3), ('Ambassade d'Allemagne à Conakry', 'government', 1)], [('Embajada de Países Bajos en España', 'government', 31), ('Illinois Housing Development Authority', 'government', 18)], [('Izraēla Latvijā', 'government', 20), ('Новости Первого канала', 'tvshow', 3)], [('TV Azteca Guatemala', 'company', 5), ('Mts', 'company', 4)], [('ITV Football', 'company', 8), ('Soy Vay', 'company', 6)], [('166th-Airlift-Wing-Delaware-Air-National-Guard', 'government', 46), ('Daniela Ludwig', 'politician', 8)], [('Shannon Stubbs', 'politician', 23), ('Jean Kuhlmann', 'politician', 6)], [('Wolfgang Aldag', 'politician', 18), ('India in Bhutan (Embassy of India, Thimphu)', 'government', 4)], [('Robert Buckland QC MP', 'politician', 9), ('John Horgan', 'politician', 2)], [('MAGGI', 'company', 2), ('Volkswagen', 'company', 2)], [('On connaît la chanson', 'tvshow', 6), ('Dukascope TV', 'company', 3)], [('2nd Armored Brigade Combat Team, 1st Cavalry Division', 'government', 49), ('Ministry of Foreign Affairs of Japan', 'government', 16)], [('McDonald's Latino', 'company', 6), ('Michigan Senate Democrats', 'government', 5)], [('JBSA Ft. Sam Houston Mission Training Complex', 'government', 18), ('Ignacio Diego Cantabria', 'politician', 6)], [('Bass 2 Billfish with Peter Miller', 'tvshow', 10), ('Jonathan Coleman', 'politician', 7)], [('Jamie Schmale, MP', 'politician', 20), ('Prüne Perú', 'company', 4)], [('Colin Carrie', 'politician', 20), ('Beekman 1802 Boys', 'company', 10)], [('Migros Türkiye', 'company', 3), ('Altibox', 'company', 2)], [('Zain KSA', 'company', 8), ('FTO-Yakama Agency', 'government', 2)], [('Córdoba al Mundo', 'government', 16), ('Times of Oman', 'company', 3)], [('Gilmore Girls', 'tvshow', 5), ('San Pellegrino Fruit Beverages', 'company', 4)], [('Jeff Landry', 'politician', 11), ('Jean-François Jalkh', 'politician', 9)], [('Marion VA Medical Center', 'government', 41), ('Ana Birchall', 'politician', 29)], [('Go Coast Guard', 'government', 30), ('Michael Donth MdB', 'politician', 25)], [('Himanta Biswa Sarma', 'politician', 8), ('余宛如', 'politician', 2)], [('განათლების და მეცნიერების სამინისტრო', 'government', 27), ('Village of Hoffman Estates Government', 'government', 6)], [('Ewald Schurer', 'politician', 39), ('Triumph', 'company', 11)], [('Polish Cultural Institute', 'government', 15), ('Jens Niklaus', 'politician', 4)], [('German Embassy Tripoli', 'government', 14), ('Gelados NESTLÉ', 'company', 10)], [('63rd Expeditionary Signal Battalion', 'government', 13), ('Jornal do SBT', 'tvshow', 3)], [('Busted', 'tvshow', 2), ('ENGIE', 'company', 1)], [('MasterChef España', 'tvshow', 12), ('Roberto Albores Gleason', 'politician', 8)], [('Martin Pakula', 'politician', 33), ('TNI Angkatan Darat', 'government', 1)], [('Pepe Yunes', 'politician', 9), ('Kementerian Perdagangan', 'government', 3)], [('Crocs', 'company', 7), ('The Jim Bakker Show', 'tvshow', 4)], [('Karmenu Vella', 'politician', 47), ('Governor Mary Fallin', 'government', 5)], [('X-BIONIC', 'company', 9), ('Ultra Buenos Aires', 'company', 1)], [('Heimatrauschen', 'tvshow', 12), ('Chief Electoral Officer, Himachal Pradesh', 'government', 1)], [('Netherlands Consulate General in Hong Kong and Macao SAR', 'government', 25), ('laSexta 1 Deportes', 'tvshow', 3)], [('Captain James A. Lovell Federal Health Care Center', 'government', 20), ('Comptroller of Maryland', 'government', 9)], [('Northport VA Medical Center', 'government', 17), ('Illinois Department of Healthcare and Family Services', 'government', 17)],

nment', 3)], [('ExpertSuomi', 'company', 3), ('Utopia NL', 'tvshow', 2)], [('Embajada de Polonia', 'government', 10), ('Las Vegas Fire & Rescue', 'government', 4)], [('T.C. Tiflis Büyükşehirliği', 'government', 12), ('Dr. Thomas Schäfer', 'politician', 2)], [('Permanent Delegation of Turkey to the European Union', 'government', 10), ('Angkor Sentinel', 'government', 7)], [('The Bad Girls Club', 'tvshow', 8), ('Europ Assistance', 'company', 6)], [('INFINITI', 'company', 5), ('MetLife México', 'company', 3)], [('U.S. Coast Guard Great Lakes', 'government', 27), ('ForRent.com', 'company', 14)], [('Smartly.io', 'company', 1), ('Triumph', 'company', 1)], [('Ministère de l'Énergie de l'Ontario', 'government', 14), ('Anurag Thakur', 'politician', 5)], [('U.S. Embassy Ouaga', 'government', 32), ('Julia Banks MP', 'politician', 22)], [('City of Plymouth, MN - Local Government', 'government', 3), ('Alles was zählt', 'tvshow', 3)], [('Studentská Pečeť', 'company', 9), ('The Librarians', 'tvshow', 5)], [('"Burt's Bees"', 'company', 13), ('Dr. Günther Bergmann', 'politician', 11)], [('Undervisningsministeriet', 'government', 6), ('Hockey Night Punjabi', 'tvshow', 1)], [('Donatelife', 'government', 34), ('Forum Austriaco di Cultura Roma / Österreichisches Kulturforum Rom', 'government', 19)], [('FTD Flowers', 'company', 11), ('City of South Bend - Municipal Government', 'government', 1)], [('Army Contracting Command - Rock Island', 'government', 18), ('Nando's', 'company', 3)], [('Eurovision Song Contest', 'tvshow', 40), ('Czesław Hoc - Poseł do Parlamentu Europejskiego', 'politician', 2)], [('Christian Fühner', 'politician', 4), ('Inside Job TNT', 'tvshow', 1)], [('Chief Electoral Officer, Uttarakhand', 'government', 2), ('Sideliners', 'tvshow', 1)], [('Balita Pilipinas', 'tvshow', 4), ('Consulado General de la República Argentina en Porto Alegre', 'government', 1)], [('Fort Dix Soldier for Life - Transition Assistance Program', 'government', 30), ('Embajada Argentina en Moscú - Посольство Аргентинской Республики', 'government', 3)], [('"經濟部"', 'government', 10), ('Consulado General de la República Argentina en San Pablo', 'government', 4)], [('Purchasing Power', 'company', 6), ('Rajeev Chandrasekhar, MP', 'politician', 5)], [('Arla', 'company', 3), ('Norbert Geis, Rechtsanwalt', 'politician', 1)], [('"Instituto Nacional de Cancerología"', 'government', 8), ('防衛省(学生向け)', 'government', 1)], [('"The Wire"', 'tvshow', 8), ('Microsoft Imagine', 'company', 7)], [('"GORE-TEX® Products"', 'company', 13), ('Janaina Riva', 'politician', 2)], [('"Udi's Gluten Free"', 'company', 16), ('WDR Lokalzeit Bergisches Land', 'tvshow', 5)], [('"GIORDANO"', 'company', 3), ('Elmano Férrer', 'politician', 3)], [('"e.on"', 'company', 11), ('Kirsty Blackman MP', 'politician', 4)], [('"WDR Lokalzeit aus Dortmund"', 'tvshow', 18), ('USAID Cambodia', 'government', 4)], [('"Werner Langen"', 'politician', 3), ('South Coast Air Quality Management District - SCAQMD', 'government', 2)], [('"Total"', 'company', 4), ('Carrefour Market', 'company', 3)], [('"La France au Japon - 在日フランス大使館"', 'government', 5), ('Eduardo Tassano', 'politician', 3)], [('"AirAsia"', 'company', 9), ('Nuno Melo', 'politician', 9)], [('"Ambassade de France en République démocratique du Congo"', 'government', 11), ('Crunch Greece', 'company', 1)], [('"San Pellegrino"', 'company', 11), ('Consulate General of the Netherlands in Shanghai', 'government', 5)], [('"Shopbop"', 'company', 16), ('Hello Goodbye Canada', 'tvshow', 2)], [('"Xerox"', 'company', 4), ('LATAM Airlines', 'company', 2)], [('"Milind Deora"', 'politician', 3), ('wagamama', 'company', 2)], [('"Disneyland Latino"', 'company', 10), ('CNN Original Series', 'tvshow', 2)], [('"Lcdot - Lake County, IL Division of Transportation"', 'government', 16), ('THW Karlsruhe', 'government', 1)], [('"AFN Wiesbaden"', 'government', 26), ('Sergio Silvestris', 'politician', 3)], [('"Zoran Milanović"', 'politician', 4), ('Esperidião Amin', 'politician', 3)], [('"Niclas Bekker Poulsen"', 'politician', 10), ('Fusion Energy Drink', 'company', 1)], [('"Valmir Assunção"', 'politician', 18), ('Immer wieder sonntags', 'tvshow', 2)], [('"Austrian Embassy Kuala Lumpur"', 'government', 14), ('Worldcard', 'company', 5)], [('"Ab ins Beet"', 'tvshow', 3), ('Angie Warren-Clark MP', 'politician', 2)], [('"Tara Halim"', 'politician', 3), ('Little Bites', 'company', 2)], [('"5th Squadron 4th Cavalry Regiment"', 'government', 27), ('Miguel Rossetto', 'politician', 2)], [('"Ministerio de Sanidad, Servicios Sociales e Igualdad"', 'government', 13), ('František Laudát', 'politician', 2)], [('"iTrial.hk"', 'company', 3), ('McDonald's Belgique', 'company', 1)], [('"LightInTheBox"', 'company', 3), ('Patricia Lips', 'politician', 1)], [('"American Odyssey"', 'tvshow', 2), ('Villum Christensen', 'politician', 2)], [('"Dunkin' Donuts"', 'company', 2), ('Coffee Mate Egypt', 'company', 1)], [('"Ernst D

ieter Rossmann', 'politician', 10), ('Ambassade de France au Soudan / سفارة فرنسا بالسودان', 'government', 1)], [('In Gefahr', 'tvshow', 3), ('Sports Rousseau / Hockey Experts / L'Entrepôt du Hockey: Le Trio Hockey', 'company', 2)], [('Dark Matter', 'tvshow', 3), ('Dayton, Ohio - City Government', 'government', 1)], [('Ambassade du Canada au Salvador', 'government', 7), ('Mark Harper', 'politician', 1)], [('Castello', 'company', 16), ('Marchesi 1824', 'company', 3)], [('L'amour est dans le pré', 'tvshow', 10), ('Action Ya Dawry أكشن يا دوري', 'tvshow', 2)], [('Jaguar', 'company', 4), ('Eurostar', 'company', 2)], [('Carlos Menchaca', 'politician', 5), ('Finnair', 'company', 1)], [('Chiquititas', 'tvshow', 2), ('Gúr Nándor', 'politician', 1)], [('Dr. Götz Frömming, MdB', 'politician', 5), ('Bridgestone Türkiye', 'company', 1)], [('Gabrielle Lemieux', 'politician', 7), ('Abhay Singh Chautala', 'politician', 2)], [('Silence ça pousse', 'tvshow', 10), ('PUGET', 'company', 1)], [('Open House', 'tvshow', 2), ('Rogerio Castro Vázquez', 'politician', 1)], [('Xerox', 'company', 17), ('Velvet', 'tvshow', 3)], [('Okaloosa County Sheriff's Office', 'government', 19), ('Filip Humplík', 'politician', 4)], [('Jean-François Copé', 'politician', 10), ('Pirelli', 'company', 8)], [('Revlon Brasil', 'company', 2), ('City of Altoona, Wisconsin - Government', 'government', 1)], [('陳歐珀', 'politician', 1), ('Travel Alberta', 'company', 3)], [('Ledlenser', 'company', 5), ('Polaris Mexico', 'company', 1)], [('ポーランド 広報文化センター Instytut Polski w Tokio', 'government', 7), ('Heinz Ireland', 'company', 1)], [('SS Ahluwalia', 'politician', 2), ('Tabasco', 'company', 2)], [('Daryl Maguire MP', 'politician', 3), ('U.S. Coast Guard Sector Detroit', 'government', 1)], [('Ambassade de Suisse en Algérie', 'government', 6), ('Antonio Bonfatti', 'politician', 5)], [('Rovana Plumb', 'politician', 11), ('HEMA', 'company', 3)], [('Planeta Extremo TV Globo', 'tvshow', 7), ('Killion Munyama', 'politician', 3)], [('The Fresh Prince of Bel-Air', 'tvshow', 2), ('Essen Başkonsolosluğu', 'government', 1)], [('Margarita Arellanes Cervantes', 'politician', 5), ('SeaWorld Parks: Busch Gardens, SeaWorld, Discovery Cove e Aquatica', 'company', 1)], [('Eduard Kukan', 'politician', 11), ('Deputado Carlos Bezerra', 'politician', 2)], [('Bom dia Amazônia, Jornal do AC e G1 AC', 'tvshow', 3), ('Gain', 'company', 2)], [('mydays', 'company', 9), ('Adobe Deutschland', 'company', 3)], [('Canan Bayram', 'politician', 3), ('Entourage', 'tvshow', 1)], [('The Voice of Greece', 'tvshow', 1), ('Vetenskapens värld', 'tvshow', 1)], [('TIM', 'company', 7), ('FlySAA Mozambique', 'company', 6)], [('Elgiganten DK', 'company', 3), ('GeberitPro', 'company', 1)], [('Jarosław Gowin', 'politician', 1), ('COMELEC', 'government', 1)], [('Marco Enriquez-Ominami', 'politician', 1), ('Valentin Abel', 'politician', 1)], [('Senador Garibaldi Alves', 'politician', 2), ('Loavies', 'company', 1)], [('En la tuya o en la mía', 'tvshow', 2), ('Cracked', 'tvshow', 1)], [('Constantine', 'tvshow', 3), ('Tofik Mamedov', 'politician', 1)], [('HOOQ', 'company', 2), ('Knut Arild Hareide', 'politician', 1)], [('Orange Burkina', 'company', 4), ('City of Fontana, CA - City Government', 'government', 3)], [('Henner Schmidt', 'politician', 1), ('Francesco Bonifazi', 'politician', 1)], [('Beto Tricoli', 'politician', 3), ('Bayer', 'company', 1)], [('LATAM Airlines', 'company', 11), ('Edinho Silva', 'politician', 5)], [('Cedevida', 'company', 2), ('Mysteriet på Greveholm', 'tvshow', 1)], [('Nordea Danmark', 'company', 2), ('I Medici', 'tvshow', 2)], [('Tengo Talento Mucho Talento', 'tvshow', 2), ('Leverkusen.de', 'government', 1)], [('Lycamobile France', 'company', 4), ('Department of Science and Technology, Government of India', 'government', 1)], [('Alpharetta Department of Public Safety', 'government', 1), ('Pensionsmyndigheten - Fråga om pension', 'government', 1)], [('Bob Katter', 'politician', 2), ('FOI Philippines', 'government', 2)], [('Antje Lezius', 'politician', 7), ('Les séries TF1', 'tvshow', 3)], [('Deputado Cabo Daciolo', 'politician', 1), ('SynDROOM', 'tvshow', 1)]

```
In [9]: def rank_on_page_rank(graph):
        pagerank_result = dict(pagerank(graph))
        pagerank_result = sorted(pagerank_result.items(),key = lambda x:x[1],reverse=True)
        return pagerank_result
```

```
In [10]: pagerank_result=rank_on_page_rank(graph)
```

```
In [11]: def filter_pagerank(pagerank_result,size,graph):
        final_pagerank_result=[]
        for result in pagerank_result[:size]:
            features=community_features_df[community_features_df["id"]==result[0]]
            final_pagerank_result.append((features["page_name"].values[0],features["page_type"].values[0],features["facebook_id"].values[0]))
        return final_pagerank_result
```

```
In [12]: final_pagerank_result=filter_pagerank(pagerank_result.copy(),15,graph)
        page_rank_dict["pagerank"]=final_pagerank_result
```

```
In [13]: final_pagerank_result
```

```
Out[13]: [('Facebook', 'company', 20531316728),
          ('Sir Peter Bottomley MP', 'politician', 293136030810246),
          ('Harish Rawat', 'politician', 271420283032553),
          ('The White House', 'government', 1191441824276882),
          ('The Obama White House', 'government', 63811549237),
          ('Barack Obama', 'politician', 6815841748),
          ('Joachim Herrmann', 'politician', 741724592531706),
          ('U.S. Army', 'government', 44053938557),
          ('European Parliament', 'government', 178362315106),
          ('Manfred Weber', 'politician', 456082177763485),
          ('U.S. Army Chaplain Corps', 'government', 404391086302925),
          ('FEMA Federal Emergency Management Agency', 'government', 78922439964),
          ('Home & Family', 'tvshow', 344485392298301),
          ('European Commission', 'government', 107898832590939),
          ('U.S. Department of State', 'government', 15877306073)]
```



```
In [ ]: filename = open(path+project_name+"/PickleFiles/fbcommunities.pkl","wb")
pickle.dump(communities_dict, filename)
filename.close()
```

```
In [14]: filename = open(path+project_name+"/PickleFiles/fbpagerank.pkl","wb")
pickle.dump(page_rank_dict, filename)
filename.close()
```

```
In [15]: def profile_pic_fetcher(facebook_id):
    try:
        profile_pic_url="https://graph.facebook.com/{}/picture?type=large".format(facebook_id)
        response = requests.get(profile_pic_url)
        profile_pic = Image.open(BytesIO(response.content))
        return profile_pic
    except Exception:
        print("Hi")
        pass
```

```
In [16]: for pagerankResult in final_pagerank_result:
    profile_pic=profile_pic_fetcher(pagerankResult[2])
    profile_pic.save(path+project_name+"/PageRankImages/Facebook/{}.png".format(pagerankResult[0]))
```