

Task 2

December 21, 2020

```
[1]: consumer_key = "3CPSX6vE31S6wiyUFpc45wqd6"
      consumer_key_secret = "mbuRYtNTaFLYC5IRrW2Dz6mj5fX1kv0SgIk50CSLqLYHQMxR00"
      access_token = "391364404-IiR1LVX4K010iTFbtpMEWjzNoV07AJI7YPd8z1kD"
      access_token_secret = "qvmLo4zi3tcQWJIYZQkrB2P0iwcuzHLaLmTBpo98HYmNg"
```

```
[2]: import tweepy
      import matplotlib.pyplot as plt
      import pytz

      auth = tweepy.OAuthHandler(consumer_key, consumer_key_secret)
      auth.set_access_token(access_token, access_token_secret)

      api = tweepy.API(auth)
```

```
[3]: # fetching trends in New Delhi
      trends_raw = api.trends_place(id=20070458)
      # filtering top 3 trends
      trends_raw[0]['trends'] = trends_raw[0]['trends'][:3]
      trends_raw
```

```
[3]: [{'trends': [{'name': '#GodiMediaAgainstFarmers',
                    'url': 'http://twitter.com/search?q=%23GodiMediaAgainstFarmers',
                    'promoted_content': None,
                    'query': '%23GodiMediaAgainstFarmers',
                    'tweet_volume': 106629},
                  {'name': '#mirchilagitoh',
                    'url': 'http://twitter.com/search?q=%23mirchilagitoh',
                    'promoted_content': None,
                    'query': '%23mirchilagitoh',
                    'tweet_volume': None},
                  {'name': '#mondaythoughts',
                    'url': 'http://twitter.com/search?q=%23mondaythoughts',
                    'promoted_content': None,
                    'query': '%23mondaythoughts',
                    'tweet_volume': 19556}],
       'as_of': '2020-12-21T08:38:02Z',
       'created_at': '2020-12-20T10:30:17Z',
```

```
'locations': [{ 'name': 'Delhi', 'woeid': 20070458}]]]
```

```
[4]: top_trending_hashtag = trends_raw[0]['trends'][0]['name']
print("{} is the top trending hashtag in New Delhi".
      ↪format(top_trending_hashtag))
```

#GodiMediaAgainstFarmers is the top trending hashtag in New Delhi

```
[5]: ## get 10000 tweets

print("Fetching tweets...")

tweets = []

old_len = 0

while len(tweets) < 10000:
    if len(tweets) == 0:
        tweets = api.search(q=top_trending_hashtag, count=100)
    else:
        lowest_tweet_id = tweets[len(tweets)-100]._json['id']
        for tweet in tweets:
            tweet_id = tweet._json['id']
            if tweet_id < lowest_tweet_id:
                lowest_tweet_id = tweet_id
        lowest_tweet_id -= 1
        tweets += api.search(q=top_trending_hashtag, count=100, ↪
        ↪max_id=lowest_tweet_id)
        if len(tweets) == old_len:
            break
        old_len = len(tweets)

print("{} tweets fetched successfully.".format(len(tweets)))
```

Fetching tweets...

10080 tweets fetched successfully.

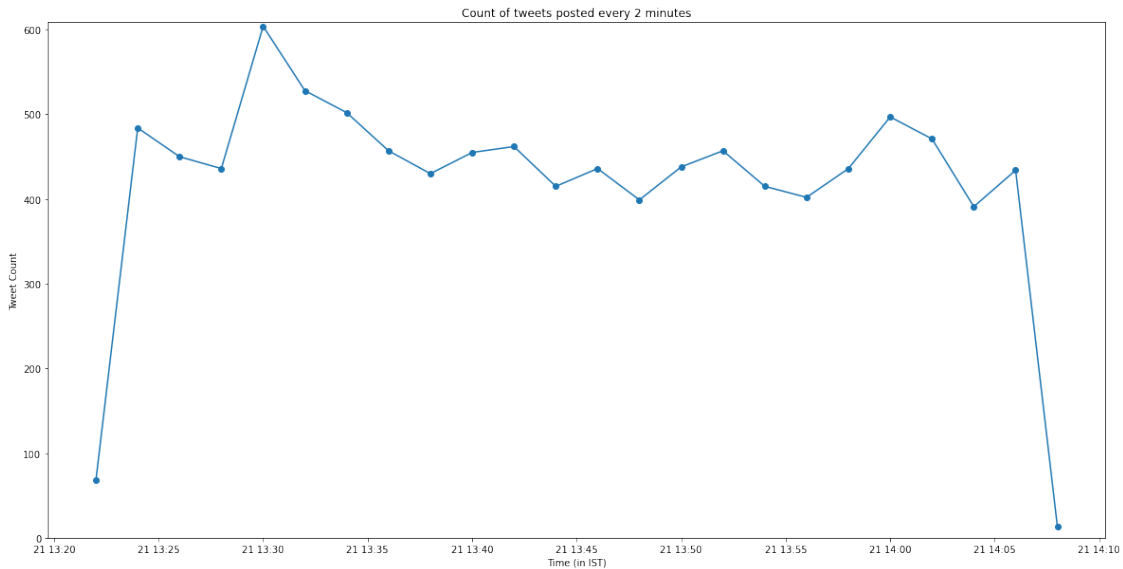
```
[6]: # Tweets vs Time plot

tweet_time = {}
for tweet in tweets:
    time_val = tweet.created_at
    time_val = time_val.replace(minute=(time_val.minute//2)*2, second=0)
    if time_val not in tweet_time.keys():
        tweet_time[time_val]=0
    tweet_time[time_val]+=1
```

```

plt.figure(figsize=(20,10))
plt.ylim(0, max(tweet_time.values())+5)
plt.gca().axis_date('Asia/Kolkata')
plt.plot(list(tweet_time.keys()),tweet_time.values(), linestyle='-',
↪marker='o')
plt.xlabel('Time (in IST)')
plt.ylabel('Tweet Count')
plt.title('Count of tweets posted every 2 minutes')
plt.show()

```



```

[7]: # For users info

user_verification={"Verified": 0, "Non-Verified": 0}
users={}

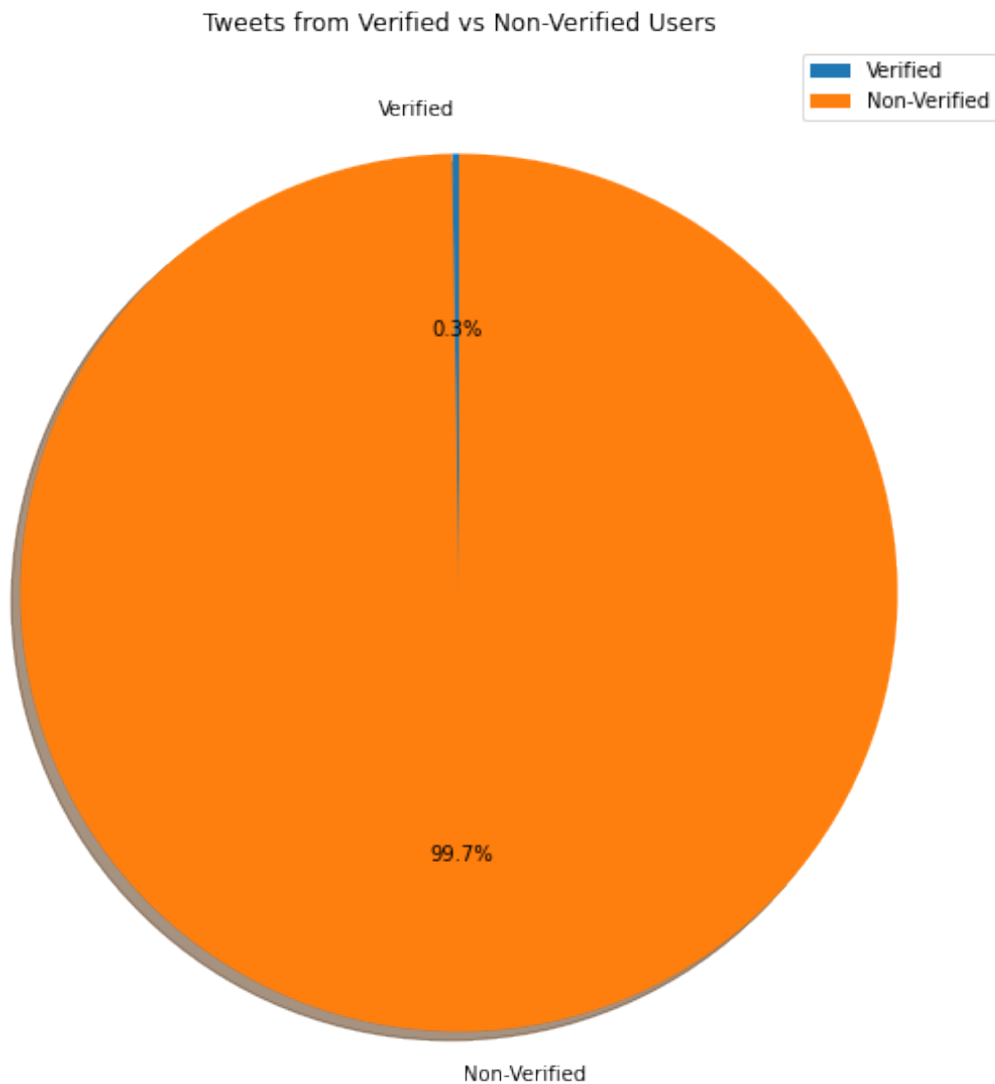
for tweet in tweets:
    author=tweet.author
    if author._json['screen_name'] not in users.keys():
        if author.verified == True:
            user_verification["Verified"]+=1
        else:
            user_verification["Non-Verified"]+=1
    users[author._json['screen_name']] = {'statuses_count':author.
↪_json['statuses_count'], 'followers_count':author._json['followers_count'],
↪'friends_count':author._json['friends_count'], 'created_at':author.
↪created_at}

plt.figure(figsize =(10, 10))

```

```
plt.pie(list(user_verification.values()), labels = list(user_verification.
↳keys()), autopct='%1.1f%%',shadow=True, startangle=90)
plt.legend(loc="upper right")
plt.title("Tweets from Verified vs Non-Verified Users")
plt.show()

print("{:.2f}% of users who posted this hashtag are verified".
↳format(100*(user_verification["Verified"] /
↳(user_verification["Verified"]+user_verification["Non-Verified"]))))
```



0.26% of users who posted this hashtag are verified

```
[8]: # User registrations per year

years_cnt = {2016:0}

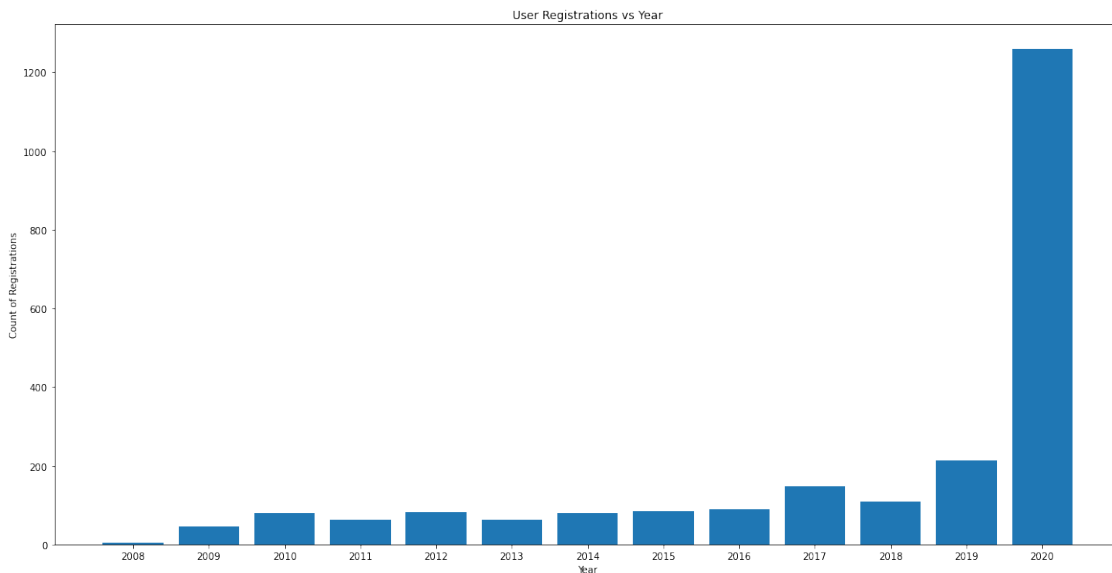
for user in users:
    year = users[user]['created_at'].year
    if year not in years_cnt.keys():
        years_cnt[year] = 0
    years_cnt[year] += 1

for year in range(min(years_cnt.keys()), 2020+1):
    if year not in years_cnt.keys():
        years_cnt[year] = 0

years_cnt = dict(sorted(years_cnt.items(), key=lambda item: item[0]))
labels = [str(i) for i in years_cnt.keys()]

plt.figure(figsize=(20,10))
plt.bar(labels, years_cnt.values())
plt.ticklabel_format(style='plain', axis='y')
plt.xlabel('Year')
plt.ylabel('Count of Registrations')
plt.title('User Registrations vs Year')
plt.show()

max_year = max(years_cnt.keys(), key=(lambda key: years_cnt[key]))
print("A total of {} users registered during {}".format(years_cnt[max_year],
↳max_year))
```



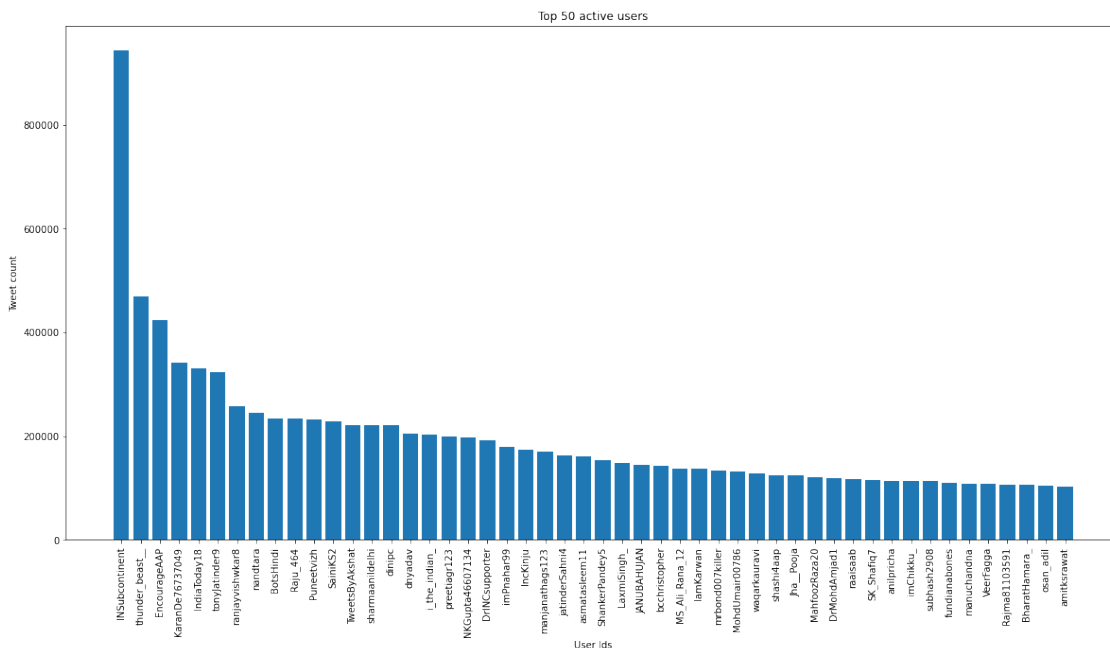
A total of 1259 users registered during 2020.

```
[9]: def get_user_info(id):
    user = api.get_user(screen_name=id)
    user_txt = "{} (@{})".format(user._json['name'], user._json['screen_name'])
    return user_txt

[10]: # Most active users with maximum number of tweets throughout his/her life
status = dict(sorted(users.items(), key=lambda item: item[1]['statuses_count'],
    ↪reverse=True))
status_cnt = [x['statuses_count'] for x in status.values()]
status = list(status.keys())

fig = plt.figure(figsize=(20, 10))
plt.bar(status[:50], status_cnt[:50])
plt.ticklabel_format(style='plain', axis='y')
plt.xticks(rotation=90)
plt.title('Top 50 active users')
plt.xlabel('User Ids')
plt.ylabel('Tweet count')
plt.show()

print("{} , {} , and {} are the most active users among all the users available.".
    ↪format(get_user_info(status[0]), get_user_info(status[1]),
    ↪get_user_info(status[2])))
```



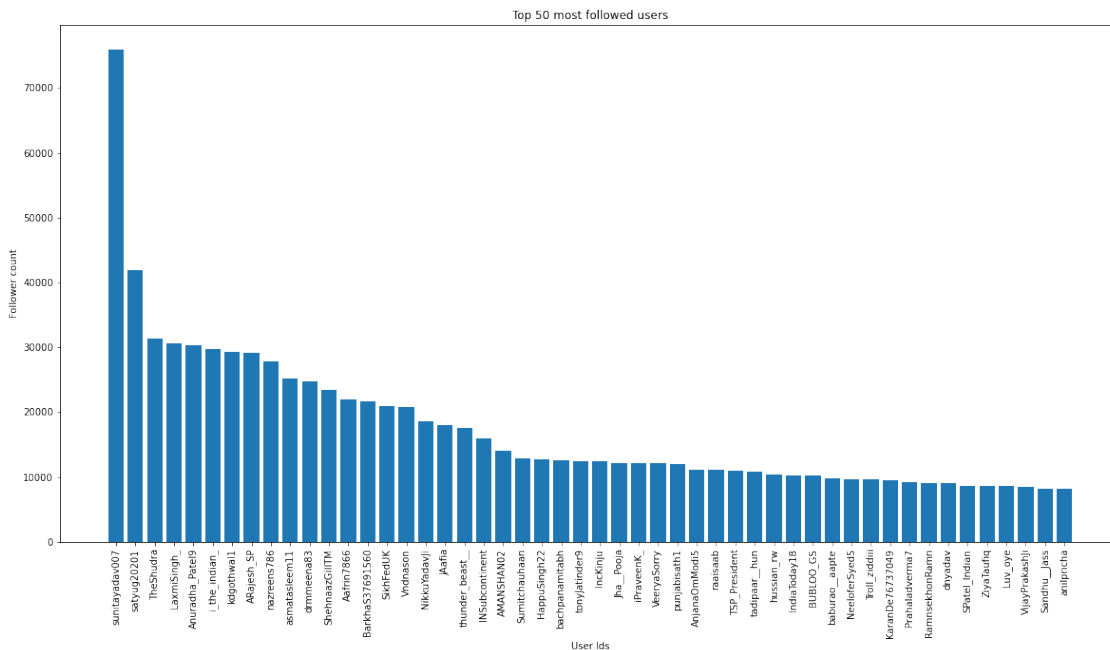
insubcontinent (@INSubcontinent), Narendra Choudhary (@thunder_beast__), and Srimi (@EncourageAAP) are the most active users among all the users available.

```
[11]: # Most followed users
```

```
labels = dict(sorted(users.items(), key=lambda item:
    ↳item[1]['followers_count'], reverse=True))
follower_cnt = [x['followers_count'] for x in labels.values()]
labels = list(labels.keys())

fig = plt.figure(figsize=(20, 10))
plt.bar(labels[:50], follower_cnt[:50])
plt.ticklabel_format(style='plain', axis='y')
plt.xticks(rotation=90)
plt.title('Top 50 most followed users')
plt.xlabel('User Ids')
plt.ylabel('Follower count')
plt.show()

print("{} , {} , and {} are the top 3 most followed users among all the users
    ↳available.".format(get_user_info(labels[0]), get_user_info(labels[1]),
    ↳get_user_info(labels[2])))
```



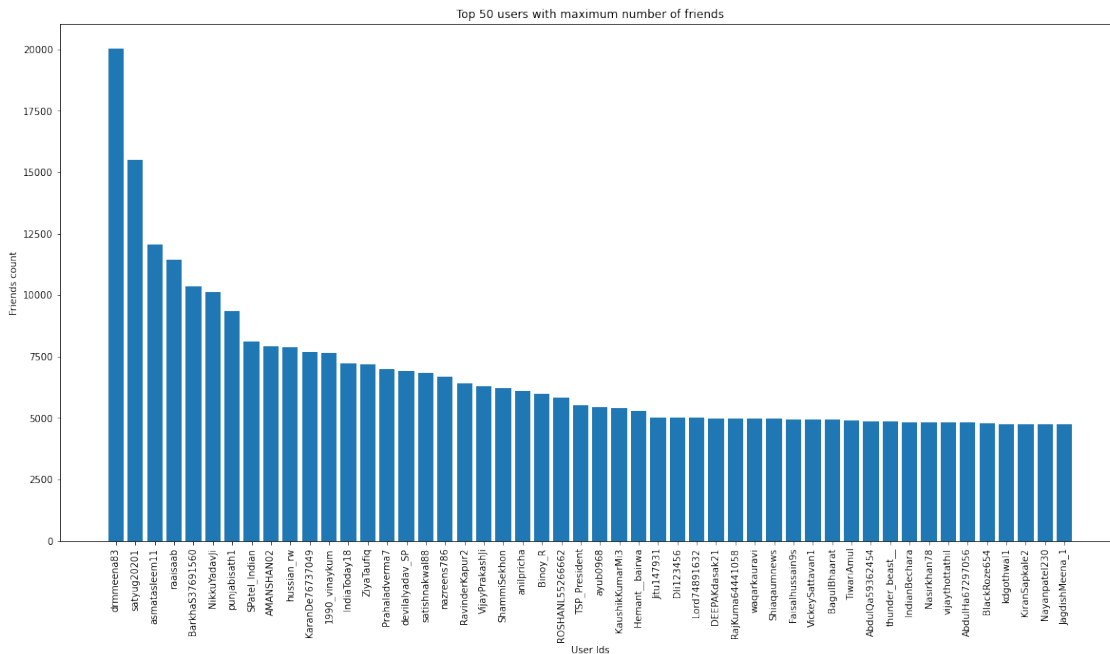
SUNITA YADAV 75k (@sunitayadav007), Rakesh singh (@satyug20201), and The Shudra (@TheShudra) are the top 3 most followed users among all the users available.

```
[12]: # Users having maximum number of friends

labels = dict(sorted(users.items(), key=lambda item: item[1]['friends_count'],
    ↪reverse=True))
follower_cnt = [x['friends_count'] for x in labels.values()]
labels = list(labels.keys())

fig = plt.figure(figsize=(20, 10))
plt.bar(labels[:50], follower_cnt[:50])
plt.ticklabel_format(style='plain', axis='y')
plt.xticks(rotation=90)
plt.title('Top 50 users with maximum number of friends')
plt.xlabel('User Ids')
plt.ylabel('Friends count')
plt.show()

print("{} , {} , and {} are the top 3 users with maximum number of friends among
    ↪all the users available.".format(get_user_info(labels[0]),
    ↪get_user_info(labels[1]), get_user_info(labels[2])))
```



Munesh Meena (@drmmeeena83), Rakesh singh (@satyug20201), and Asma (@asmatasleem11) are the top 3 users with maximum number of friends among all the users available.

[13]: *# to fetch languages*

```
languages = {
  "ab": {
    "name": "Abkhaz",
    "nativeName": " "
  },
  "aa": {
    "name": "Afar",
    "nativeName": "Afaraf"
  },
  "af": {
    "name": "Afrikaans",
    "nativeName": "Afrikaans"
  },
  "ak": {
    "name": "Akan",
    "nativeName": "Akan"
  },
  "sq": {
    "name": "Albanian",
    "nativeName": "Shqip"
  },
  "am": {
    "name": "Amharic",
    "nativeName": " "
  },
  "ar": {
    "name": "Arabic",
    "nativeName": " "
  },
  "an": {
    "name": "Aragonese",
    "nativeName": "Aragonés"
  },
  "hy": {
    "name": "Armenian",
    "nativeName": " "
  },
  "as": {
    "name": "Assamese",
    "nativeName": " "
  },
  "av": {
    "name": "Avaric",
    "nativeName": " , "
  },
}
```

```

    "ae": {
      "name": "Avestan",
      "nativeName": "avesta"
    },
    "ay": {
      "name": "Aymara",
      "nativeName": "aymar aru"
    },
    "az": {
      "name": "Azerbaijani",
      "nativeName": "azərbaycan dili"
    },
    "bm": {
      "name": "Bambara",
      "nativeName": "bamanankan"
    },
    "ba": {
      "name": "Bashkir",
      "nativeName": "
    },
    "eu": {
      "name": "Basque",
      "nativeName": "euskara, euskera"
    },
    "be": {
      "name": "Belarusian",
      "nativeName": "
    },
    "bn": {
      "name": "Bengali",
      "nativeName": "
    },
    "bh": {
      "name": "Bihari",
      "nativeName": "
    },
    "bi": {
      "name": "Bislama",
      "nativeName": "Bislama"
    },
    "bs": {
      "name": "Bosnian",
      "nativeName": "bosanski jezik"
    },
    "br": {
      "name": "Breton",
      "nativeName": "brezhoneg"

```

```

},
  "bg": {
    "name": "Bulgarian",
    "nativeName": " "
  },
  "my": {
    "name": "Burmese",
    "nativeName": " "
  },
  "ca": {
    "name": "Catalan; Valencian",
    "nativeName": "Català"
  },
  "ch": {
    "name": "Chamorro",
    "nativeName": "Chamoru"
  },
  "ce": {
    "name": "Chechen",
    "nativeName": " "
  },
  "ny": {
    "name": "Chichewa; Chewa; Nyanja",
    "nativeName": "chiCheŵa, chinyanja"
  },
  "zh": {
    "name": "Chinese",
    "nativeName": " (Zhōngwén), , "
  },
  "cv": {
    "name": "Chuvash",
    "nativeName": " "
  },
  "kw": {
    "name": "Cornish",
    "nativeName": "Kernewek"
  },
  "co": {
    "name": "Corsican",
    "nativeName": "corsu, lingua corsa"
  },
  "cr": {
    "name": "Cree",
    "nativeName": " "
  },
  "hr": {
    "name": "Croatian",

```

```

        "nativeName": "hrvatski"
    },
    "cs": {
        "name": "Czech",
        "nativeName": "česky, čeština"
    },
    "da": {
        "name": "Danish",
        "nativeName": "dansk"
    },
    "dv": {
        "name": "Divehi; Dhivehi; Maldivian;",
        "nativeName": " "
    },
    "nl": {
        "name": "Dutch",
        "nativeName": "Nederlands, Vlaams"
    },
    "en": {
        "name": "English",
        "nativeName": "English"
    },
    "eo": {
        "name": "Esperanto",
        "nativeName": "Esperanto"
    },
    "et": {
        "name": "Estonian",
        "nativeName": "eesti, eesti keel"
    },
    "ee": {
        "name": "Ewe",
        "nativeName": "E egbe"
    },
    "fo": {
        "name": "Faroese",
        "nativeName": "føroyskt"
    },
    "fj": {
        "name": "Fijian",
        "nativeName": "vosa Vakaviti"
    },
    "fi": {
        "name": "Finnish",
        "nativeName": "suomi, suomen kieli"
    },
    "fr": {

```

```

    "name": "French",
    "nativeName": "français, langue française"
  },
  "ff": {
    "name": "Fula; Fulah; Pulaar; Pular",
    "nativeName": "Fulfulde, Pulaar, Pular"
  },
  "gl": {
    "name": "Galician",
    "nativeName": "Galego"
  },
  "ka": {
    "name": "Georgian",
    "nativeName": " "
  },
  "de": {
    "name": "German",
    "nativeName": "Deutsch"
  },
  "el": {
    "name": "Greek, Modern",
    "nativeName": "Ε "
  },
  "gn": {
    "name": "Guaraní",
    "nativeName": "Avañeẽ"
  },
  "gu": {
    "name": "Gujarati",
    "nativeName": " "
  },
  "ht": {
    "name": "Haitian; Haitian Creole",
    "nativeName": "Kreyòl ayisyen"
  },
  "ha": {
    "name": "Hausa",
    "nativeName": "Hausa, "
  },
  "he": {
    "name": "Hebrew (modern)",
    "nativeName": " "
  },
  "hz": {
    "name": "Herero",
    "nativeName": "Otjiherero"
  },

```

```

    "hi": {
      "name": "Hindi",
      "nativeName": "हिन्दी",
    },
    "ho": {
      "name": "Hiri Motu",
      "nativeName": "Hiri Motu"
    },
    "hu": {
      "name": "Hungarian",
      "nativeName": "Magyar"
    },
    "ia": {
      "name": "Interlingua",
      "nativeName": "Interlingua"
    },
    "id": {
      "name": "Indonesian",
      "nativeName": "Bahasa Indonesia"
    },
    "ie": {
      "name": "Interlingue",
      "nativeName": "Originally called Occidental; then Interlingue after
↪WWII"
    },
    "ga": {
      "name": "Irish",
      "nativeName": "Gaeilge"
    },
    "ig": {
      "name": "Igbo",
      "nativeName": "Asụsụ Igbo"
    },
    "ik": {
      "name": "Inupiaq",
      "nativeName": "Iñupiaq, Iñupiatun"
    },
    "io": {
      "name": "Ido",
      "nativeName": "Ido"
    },
    "is": {
      "name": "Icelandic",
      "nativeName": "Íslenska"
    },
    "it": {
      "name": "Italian",

```

```

        "nativeName": "Italiano"
    },
    "iu": {
        "name": "Inuktitut",
        "nativeName": " "
    },
    "ja": {
        "name": "Japanese",
        "nativeName": " ( )"
    },
    "jv": {
        "name": "Javanese",
        "nativeName": "basa Jawa"
    },
    "kl": {
        "name": "Kalaallisut, Greenlandic",
        "nativeName": "kalaallisut, kalaallit oqaasii"
    },
    "kn": {
        "name": "Kannada",
        "nativeName": " "
    },
    "kr": {
        "name": "Kanuri",
        "nativeName": "Kanuri"
    },
    "ks": {
        "name": "Kashmiri",
        "nativeName": " , "
    },
    "kk": {
        "name": "Kazakh",
        "nativeName": " "
    },
    "km": {
        "name": "Khmer",
        "nativeName": " "
    },
    "ki": {
        "name": "Kikuyu, Gikuyu",
        "nativeName": "Gĩkũyũ"
    },
    "rw": {
        "name": "Kinyarwanda",
        "nativeName": "Ikinyarwanda"
    },
    "ky": {

```

```

    "name": "Kirghiz, Kyrgyz",
    "nativeName": "      "
  },
  "kv": {
    "name": "Komi",
    "nativeName": "      "
  },
  "kg": {
    "name": "Kongo",
    "nativeName": "KiKongo"
  },
  "ko": {
    "name": "Korean",
    "nativeName": "      ( ),      ( )"
  },
  "ku": {
    "name": "Kurdish",
    "nativeName": "Kurdi,      "
  },
  "kj": {
    "name": "Kwanyama, Kuanyama",
    "nativeName": "Kuanyama"
  },
  "la": {
    "name": "Latin",
    "nativeName": "latine, lingua latina"
  },
  "lb": {
    "name": "Luxembourgish, Letzeburgesch",
    "nativeName": "Lëtzebuergesch"
  },
  "lg": {
    "name": "Luganda",
    "nativeName": "Luganda"
  },
  "li": {
    "name": "Limburgish, Limburgan, Limburger",
    "nativeName": "Limburgs"
  },
  "ln": {
    "name": "Lingala",
    "nativeName": "Lingála"
  },
  "lo": {
    "name": "Lao",
    "nativeName": "      "
  },

```



```

    "lt": {
      "name": "Lithuanian",
      "nativeName": "lietuvių kalba"
    },
    "lu": {
      "name": "Luba-Katanga",
      "nativeName": ""
    },
    "lv": {
      "name": "Latvian",
      "nativeName": "latviešu valoda"
    },
    "gv": {
      "name": "Manx",
      "nativeName": "Gaelg, Gailck"
    },
    "mk": {
      "name": "Macedonian",
      "nativeName": " "
    },
    "mg": {
      "name": "Malagasy",
      "nativeName": "Malagasy fiteny"
    },
    "ms": {
      "name": "Malay",
      "nativeName": "bahasa Melayu, "
    },
    "ml": {
      "name": "Malayalam",
      "nativeName": " "
    },
    "mt": {
      "name": "Maltese",
      "nativeName": "Malti"
    },
    "mi": {
      "name": "Māori",
      "nativeName": "te reo Māori"
    },
    "mr": {
      "name": "Marathi (Marāṭhī)",
      "nativeName": " "
    },
    "mh": {
      "name": "Marshallese",
      "nativeName": "Kajin Majel"
    }

```

```

},
  "mn": {
    "name": "Mongolian",
    "nativeName": " "
  },
  "na": {
    "name": "Nauru",
    "nativeName": "Ekakairũ Naoero"
  },
  "nv": {
    "name": "Navajo, Navaho",
    "nativeName": "Diné bizaad, Dinék ehǫ́í"
  },
  "nb": {
    "name": "Norwegian Bokmål",
    "nativeName": "Norsk bokmål"
  },
  "nd": {
    "name": "North Ndebele",
    "nativeName": "isiNdebele"
  },
  "ne": {
    "name": "Nepali",
    "nativeName": " "
  },
  "ng": {
    "name": "Ndonga",
    "nativeName": "Owambo"
  },
  "nn": {
    "name": "Norwegian Nynorsk",
    "nativeName": "Norsk nynorsk"
  },
  "no": {
    "name": "Norwegian",
    "nativeName": "Norsk"
  },
  "ii": {
    "name": "Nuosu",
    "nativeName": " Nuosuhxop"
  },
  "nr": {
    "name": "South Ndebele",
    "nativeName": "isiNdebele"
  },
  "oc": {
    "name": "Occitan",

```

```

        "nativeName": "Occitan"
    },
    "oj": {
        "name": "Ojibwe, Ojibwa",
        "nativeName": " "
    },
    "cu": {
        "name": "Old Church Slavonic, Church Slavic, Church Slavonic, Old ↵
↪Bulgarian, Old Slavonic",
        "nativeName": " "
    },
    "om": {
        "name": "Oromo",
        "nativeName": "Afaan Oromoo"
    },
    "or": {
        "name": "Oriya",
        "nativeName": " "
    },
    "os": {
        "name": "Ossetian, Ossetic",
        "nativeName": " æ "
    },
    "pa": {
        "name": "Panjabi, Punjabi",
        "nativeName": " , "
    },
    "pi": {
        "name": "Pāli",
        "nativeName": " "
    },
    "fa": {
        "name": "Persian",
        "nativeName": " "
    },
    "pl": {
        "name": "Polish",
        "nativeName": "polski"
    },
    "ps": {
        "name": "Pashto, Pushto",
        "nativeName": " "
    },
    "pt": {
        "name": "Portuguese",
        "nativeName": "Português"
    },

```

```

    "qu": {
      "name": "Quechua",
      "nativeName": "Runa Simi, Kichwa"
    },
    "rm": {
      "name": "Romansh",
      "nativeName": "rumantsch grischun"
    },
    "rn": {
      "name": "Kirundi",
      "nativeName": "kiRundi"
    },
    "ro": {
      "name": "Romanian, Moldavian, Moldovan",
      "nativeName": "română"
    },
    "ru": {
      "name": "Russian",
      "nativeName": " "
    },
    "sa": {
      "name": "Sanskrit (Saṃskṛta)",
      "nativeName": " "
    },
    "sc": {
      "name": "Sardinian",
      "nativeName": "sardu"
    },
    "sd": {
      "name": "Sindhi",
      "nativeName": " "
    },
    "se": {
      "name": "Northern Sami",
      "nativeName": "Davvisámegiella"
    },
    "sm": {
      "name": "Samoan",
      "nativeName": "gagana faa Samoa"
    },
    "sg": {
      "name": "Sango",
      "nativeName": "yângâ tî sängö"
    },
    "sr": {
      "name": "Serbian",
      "nativeName": " "
    }

```

```

},
  "gd": {
    "name": "Scottish Gaelic; Gaelic",
    "nativeName": "Gàidhlig"
  },
  "sn": {
    "name": "Shona",
    "nativeName": "chiShona"
  },
  "si": {
    "name": "Sinhala, Sinhalese",
    "nativeName": " "
  },
  "sk": {
    "name": "Slovak",
    "nativeName": "slovenčina"
  },
  "sl": {
    "name": "Slovene",
    "nativeName": "slovenščina"
  },
  "so": {
    "name": "Somali",
    "nativeName": "Soomaaliga, af Soomaali"
  },
  "st": {
    "name": "Southern Sotho",
    "nativeName": "Sesotho"
  },
  "es": {
    "name": "Spanish; Castilian",
    "nativeName": "español, castellano"
  },
  "su": {
    "name": "Sundanese",
    "nativeName": "Basa Sunda"
  },
  "sw": {
    "name": "Swahili",
    "nativeName": "Kiswahili"
  },
  "ss": {
    "name": "Swati",
    "nativeName": "SiSwati"
  },
  "sv": {
    "name": "Swedish",

```

```

        "nativeName": "svenska"
    },
    "ta": {
        "name": "Tamil",
        "nativeName": "தமிழ்"
    },
    "te": {
        "name": "Telugu",
        "nativeName": "తెలుగు"
    },
    "tg": {
        "name": "Tajik",
        "nativeName": "Тоҷикӣ, тоҷикӣ"
    },
    "th": {
        "name": "Thai",
        "nativeName": "ไทย"
    },
    "ti": {
        "name": "Tigrinya",
        "nativeName": "ትግርኛ"
    },
    "bo": {
        "name": "Tibetan Standard, Tibetan, Central",
        "nativeName": "བོད་སྐད་"
    },
    "tk": {
        "name": "Turkmen",
        "nativeName": "Türkmen, Түркмен"
    },
    "tl": {
        "name": "Tagalog",
        "nativeName": "Wikang Tagalog, ᜏᜒᜃᜅ᜔ᜃᜄᜎᜓᜄ᜔"
    },
    "tn": {
        "name": "Tswana",
        "nativeName": "Setswana"
    },
    "to": {
        "name": "Tonga (Tonga Islands)",
        "nativeName": "faka Tonga"
    },
    "tr": {
        "name": "Turkish",
        "nativeName": "Türkçe"
    },
    "ts": {

```

```

    "name": "Tsonga",
    "nativeName": "Xitsonga"
  },
  "tt": {
    "name": "Tatar",
    "nativeName": "татарча, татарча"
  },
  "tw": {
    "name": "Twi",
    "nativeName": "Twi"
  },
  "ty": {
    "name": "Tahitian",
    "nativeName": "Reo Tahiti"
  },
  "ug": {
    "name": "Uighur, Uyghur",
    "nativeName": "ئۇيغۇرچە, ئۇيغۇر تىلى"
  },
  "uk": {
    "name": "Ukrainian",
    "nativeName": "українська мова"
  },
  "ur": {
    "name": "Urdu",
    "nativeName": "اُردُو"
  },
  "uz": {
    "name": "Uzbek",
    "nativeName": "uzbek, ўзбек тили"
  },
  "ve": {
    "name": "Venda",
    "nativeName": "Tshiven a"
  },
  "vi": {
    "name": "Vietnamese",
    "nativeName": "Tiếng Việt"
  },
  "vo": {
    "name": "Volapük",
    "nativeName": "Volapük"
  },
  "wa": {
    "name": "Walloon",
    "nativeName": "Walon"
  },

```

```

        "cy": {
            "name": "Welsh",
            "nativeName": "Cymraeg"
        },
        "wo": {
            "name": "Wolof",
            "nativeName": "Wollof"
        },
        "fy": {
            "name": "Western Frisian",
            "nativeName": "Frysk"
        },
        "xh": {
            "name": "Xhosa",
            "nativeName": "isiXhosa"
        },
        "yi": {
            "name": "Yiddish",
            "nativeName": " "
        },
        "yo": {
            "name": "Yoruba",
            "nativeName": "Yorùbá"
        },
        "za": {
            "name": "Zhuang, Chuang",
            "nativeName": "Sa cuenj , Saw cuengh"
        },
    }

def get_lang_text(code):
    if code in languages.keys():
        return languages[code]["name"]
    return "Unknown"

```

[14]: *# for language aspect*

```

lang_cnt={}

for tweet in tweets:
    lang=get_lang_text(tweet.lang)
    if lang not in lang_cnt:
        lang_cnt[lang] = 0
    lang_cnt[lang]+=1

language_code = []
language_freq = []

```



```

most_used_lang = []
max_lang_freq = 0

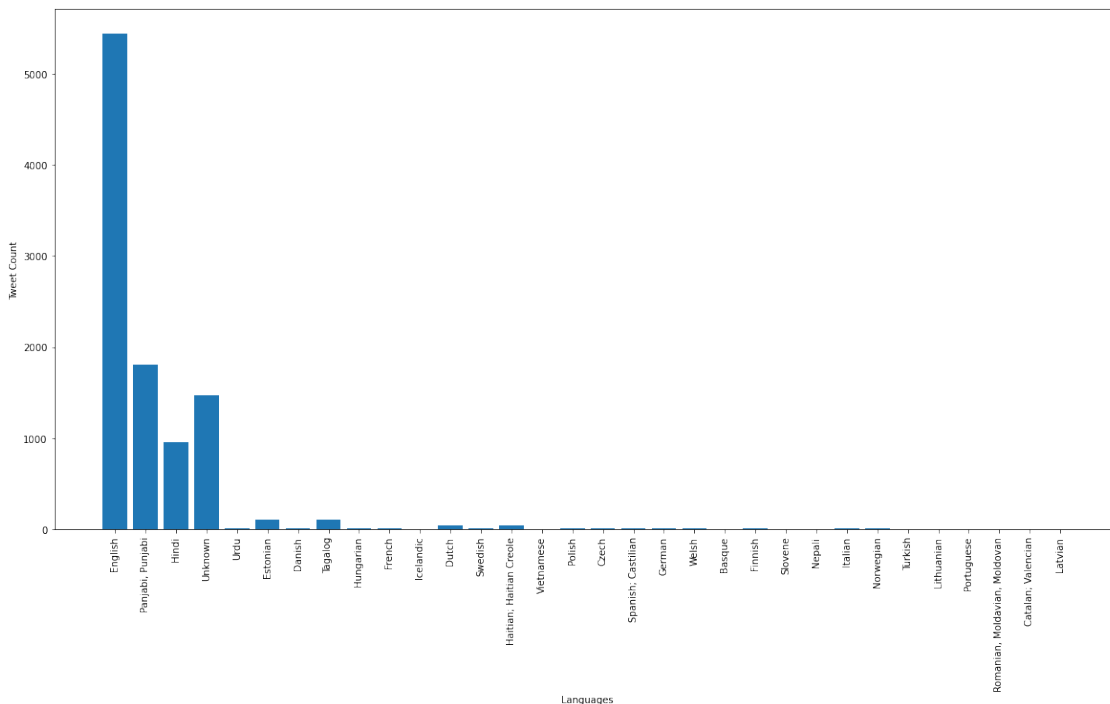
for code in lang_cnt.keys():
    language_code.append(code)
    language_freq.append(lang_cnt[code])

    if lang_cnt[code] == max_lang_freq:
        most_used_lang.append(code)
    elif lang_cnt[code] > max_lang_freq:
        most_used_lang = []
        most_used_lang.append(code)
        max_lang_freq = lang_cnt[code]

plt.figure(figsize=(20, 10))
plt.bar(language_code, language_freq)
plt.xticks(rotation=90)
plt.xlabel('Languages')
plt.ylabel('Tweet Count')
plt.show()

print('{} {} among the tweets.'.format(', '.join(str(e) for e in
↪most_used_lang), 'is the most used language' if len(most_used_lang)==1 else
↪'are the most used languages'))

```



English is the most used language among the tweets.

[15]: *# Developed by: Shlok Pandey (@b30wulffz)*

[]: