Update Your Project Allocation at https://1drv.ms/w/s!AtcJs3OTsMZuiRt8k7S3z22CATjl

Fake News Related Projects

Project 1: Fake News Twitter Analysis
Consider the 2016 US election Viral Twitter dataset collected between election day (Nov 8th) and March 2017. Tweets have been labelled as containing fake news or not by two sets of people, and where fake news is categorized into one of the five categories: i) Serious fabrication; ii) Large-scale hoaxes; iii) Jokes taken at face value; iv) Slanted reporting of real facts; v) Stories where the 'truth' is contentious. The dataset can be downloaded from Fakenews on 2016 US elections viral tweets (November 2016 - March 2017) | Zenodo

1. Separate from the overall dataset two classes of tweets: one related to labelled Fake News (regardless of the category of the fake news) and the other one for Real News. Save each tweet class in a separate file. Write a script to identify the number of distinct users (user_screen_name) in Fake News class, number of distinct users in Real News and number of distinct users who participated to both Fake News and Real News.

2. Write a script for the calculus of the mean, standard deviation, kurtosis and skewness of Number of follower per user in case of Fake News and Real News dataset. Repeat this process for Favorite Count as well. Conclude whether one can discriminate the two classes using such statistical data.

3. We want to compare the activity of individual users in Fake News and Real News dataset. Select the three most active users in terms of number tweets generated and calculate the average number of tweets generated by the three users. Repeat this process for the five most active users in each dataset, and for the first 10-users in each dataset, and first 15 users for each dataset.

4. We want to compare the average time a user stays before sending a new message. Write a script that uses the date information on the dataset for each tweet to calculate the average waiting time for a random user before sending a new message in case of Fake News and Real News.

5. Study the behavior of user ids who contributed to both Fake News and Real News. Based on your observation from 2)-3)-4) and any other scrutinizing, suggest statistical index that would discriminate Fake News and Real News tweets of the same user.

6. We want to create a social network from the Twitter dataset. For this purpose, consider the mention reference in Twitter. More specifically, write a small program that allows you to identify the "mention" in each tweet message (word precedent by "@"). Now, construct a network graph where the nodes correspond to the user ID while the edge between two nodes, say A and B, indicates that tweet of user id A contains in its text message a mention of user id B. Construct social network graph for each dataset (Fake News and Real News). Use appropriate visualization to draw high level illustration of each graph.
7. Use appropriate functions in NetworkX to calculate diameter, average clustering coefficient, average degree centrality, average closeness centrality and average betweenness centrality for each dataset.

8. Calculate the degree centrality distribution and clustering coefficient distribution for each dataset and draw the corresponding plot. Discuss your result and whether this can discriminate the cases.

9. Use VADER tool(https://github.com/cjhutto/vaderSentiment) to perform sentiment analysis on tweets of each dataset. For each user id, we want to calculate the distribution of sentiment (proportion of positive, negative and neutral sentiment), then one represents the distribution of each user as a point in the ternary plot. Repeat this process for the second dataset as well, so that two distinct ternary plot will be exhibited. Conclude whether sentiment can differentiate the two datasets.

10. We want to use the information about the various Fake News categories. Using the information about the tweets, where either a given user id sends tweet messages who are categorized in different categories, or he send a tweet message that contains a mention of a user id who is assigned to another category. Write a script to perform this operation, and then output a simple graph where the nodes are constituted of the five categories and the edge indicates a link as previously described.
11. Suggest appropriate literature in fake news identification to discuss and comment on your findings at each level of the above analysis.

Project 2:

Consider the FakeNewsNet dataset, available at GitHub - KaiDMML/FakeNewsNet: This is a dataset for fake news detection research. Use the provided code to generate all Tweet attributes (Number of retweets, User followers, User following, Retweet, Number of likes, etc (whatever is available through the Twitter API)) for each of the four data categories (Glossipcop Fake News, Glossipcop Real News, Politifact Fake News, Politifact Real News). You can also consult the FakeNewsNet reference paper of Shu et al. arXiv:1809.01286 for detailed explanation of the dataset. You will realize that not all the dataset can be reconstructed as many tweet id may not be available and in API call limit.

1. For each category dataset, provide a table describing the statistical trend of the key attributes. This consists of: i) Number of tweet messages, ii) Number of distinct user ids, iii) mean, standard deviation, kurtosis and skewness of number of retweets per user id; iv) iii) mean, standard deviation, kurtosis and skewness of number of following per user id; v) mean, standard deviation, kurtosis and skewness of number of followers per user id. Discuss whether you can discriminate between fake news and real names on the basis of these attributes.
2. Draw on the same plot the distribution of follower count for Fake News and Real News of glossipcop and politifact data. Repeat the process for the distribution of followee count for fake news and Real News.
3. Explore the temporal evolution of user's engagement (according to your suggested approach to quantify the user's engagement (i.e., number of likes, some combination of followers and followees, etc.) for Fake News and Real News, and draw the corresponding plot for both glossipcop and politifact data.
4. We would like to study whether the user ids of fake news and real news dataset are genuine or not. For this purpose, study the program botometer available in https://github.com/IUNetSci/botometer-python. The program inputs a tweet user id and outputs the probability that the user id is a bot or human. You can use a threshold 0.5 beyond which a program is bot or not. The purpose is therefore to test the hypothesis whether Fake News are globally originated from bots or humans and whether Real News are generated by humans or also by bots. If the computational time is an issue to test the whole data, you can choose a random selection of the data as well.
5. We want to explore the graph structure that can be extracted from the dataset and compare the properties of fake news and real news categories. For this purpose, consider the follower relationship, where user id A is linked to user id B if either A (resp. B) is a follower of B (resp. A). We restrict only to those user ids who are associated to dataset tweets (Need to retrieve the list of followers for each user id to test whether this relation holds). Use NetworkX to calculate global attributes of this network such as overall degree centrality, diameter, clustering coefficient, size of largest component. Compare these graph attributes for Fake News and Real News for glossipcop and politifact data. Use high level illustration to draw the network of each one.
6. Draw on the same plot the degree distribution of fake news and real news for each of glossipcop and politifact data. Conclude whether some graph attributes are relevant to distinguish fake news and real news.
7. Use relevant literature from fakes news detection from social media to discuss your finding at each level of the preceding reasoning.

Project 3. Health Fakes Diffusion

This project considers the FakeHealth dataset available at https://github.com/EnyanDai/FakeHealth which includes HealthStory and HealthRelease dataset. We shall restrict to the first dataset only. You may notice that the reconstruction of the dataset from the provided tweet id will not match the original number as some tweets may be deleted or profile switched to private.

1. Provide a table summarizing the global attributes for Fake and Real part of of the dataset, which consists on i) number of tweets, average number of tweets per news (together with corresponding standard deviation, kurtosis and skewness), average number of tweets per user per news (together with corresponding standard deviation, kurtosis and skewness), average replies per news (together with corresponding standard deviation, kurtosis and skewness), average replies per tweet (together with corresponding standard deviation, kurtosis and skewness), average retweets per news (together with corresponding standard deviation, kurtosis and skewness), average retweets per tweet (together with corresponding standard deviation, kurtosis and skewness). Discuss whether any of these global attributes allow you to make a clear distinction between Fake and Real dataset.
2. Assign a single user id for each news in Fake and Real dataset and use Twitter API to retrieve the number of followers and followees.
3. Draw on the same plot the distribution of follower count for Fake and Real of HealthStory dataset. Repeat the process for the distribution of followee count for fake and Real data.
4. Show whether power law distribution can be fitted to the above plots.
5. Explore the temporal evolution of user's engagement (according to your suggested approach to quantify the user's engagement as a function of number of replies and retweets for Fake and Real and draw the corresponding plot for HealthStory data.
6. We want to investigate whether some fake data are genuine or not. study whether the user ids of fake news and real news dataset are genuine or not. For this purpose, study the program botometer available in https://github.com/IUNetSci/botometer-python. The program inputs a tweet user id and outputs the probability that the user id is a bot or human. You can use a threshold 0.5 beyond which a program is bot or not. The purpose is therefore to test the hypothesis whether Fake News are globally originated from bots or humans and whether Real News are generated by humans or also by bots. Draw a plot showing the proportion of bots in Fake data and Real data.
7. Now we want to test the hypothesis whether a fake news occurs if the initiator (user id) is communicating with bots. For this purpose, for each news (in Fake data), select 100 random user id among those associated to that news, and apply the previous botometer and output the number of users id that are found to be bots. Plot the distribution of number of bots per news in Fake data.
8. Use VADER tool (https://github.com/cjhutto/vaderSentiment), which output sentiment in terms of POSITIVE, NEGATIVE and NEUTRAL to determine the sentiment of each news in Fake and Real data. Then represent the distribution of each news statement as a point in the ternary plot for both Fake and Real data. Conclude whether sentiment can differentiate the two datasets.
9. Suggest how you can take into account the criteria C0-C10 provided in the dataset to fine-tune the reasoning in 6-7).
10. Identify relevant literature in fake new identification and health literature to back up your finding in previous sections.

Project 4: Covid-19 diffusion network

This project aims to investigate the extent to which the diffusion models can be fitted to actual data of Covid-19 infection and recovery statistics.

Consider the official statistics on covid-19 infection and recovery provided by official organizations such as https://www.worldometers.info/coronavirus. Consider a reasonable time period for a selected country of your choice, should be a small country to make the subsequent computational time feasible.

1. Use a starting date where you consider it to stand for initial state. In the statistics of the country at the chosen, calculate the initial Infection I0 as the total number of infection minus the total recovery. Use the official corona statistical source to draw a plot showing the temporal variations of the number of infections and that of the number of recovery.
2. We want to carry on the simulation using the SIR epidemic model. Use the implementation provided in NDLIB library to perform the calculus. Set the number of nodes of the network equal to the total population and a very small probability for Erdos random graph of 0.001. Choose a infection probability beta and recovery probability beta of your choice (you may inspire from the data trend). Run the EDLIB and plot the temporal variation of the Number of infection and recovery over time.
3. Now we want to use the data of official statistics to tune the probability of infection and recovery to find a way to match the variations plotted in 2) with that of 1). Suggest an empirical approach where, for instance, you vary incrementally the values of alpha and gamma until you visualize a figure infections and recovery count closely match that of official statistics.
4. Now we want to use the official dataset statistics to estimate the probability of infection and recovery. Suggest a simple approach to calculate these attributes using the available historical dataset. Then input these values to the SIR model and run the simulation to display the variation of the infections and recovery. Discuss the relevance of the SIR model for this purpose.
5. Now we want to treat the death count provided in the statistics. Consider using the SI model for this purpose. Similarly to 1), draw the timely evolution of the number of death.
6. Next use the implementation provided in EDLIB for the SI model and suggest a simple model to generate the simulated model that displays the total number of death.
7. Suggest an empirical and incremental variation of the infection probability in SI model until the death variation is close to the real dataset. Discuss the relevance of such approach and probability value.
8. Consider the SEIRS (Susceptible-Exposed-Infected-Recovered-Susceptible) model described in https://github.com/ryansmcgee/seirsplus. Set an initial value of parameters of the model and display the temporal evolution of the infection and recovery counts.
9. Similarly to 3), suggest an empirical and possible an incremental approach to attempt to match the infection and recovery counts with that of real dataset.
10. Similarly to 4), use the official statistics to infer the infection, recovery probabilities and other parameters of the model. Then run the model again and display the new graph showing the variations of the infections and recovery counts.
11. Use relevant literature to back your reasoning and finding in previous steps.

Project 5: Analysis of Smoking Cessation

The project aims to study the online community of smoking cessation users in Twitter social network.

1. Identify two hashtags related to smoked cessation. Examples include #Stopsmoking, #smokefree, #Stoptabac, etc.. Show your reasoning to identify few other equivalent hashtags as well. Collect a sufficient number of tweets related to each hashtag (around one thousand tweets). For this purpose, you can use for instance Tweepy (see tutorial at https://riptutorial.com/tweepy), also see examples of text processing in Python in NLTK online book at https://www.nltk.org/book/. You would need to create your Twitter API account credential. The key in the collection process is that there is an important number of tweets that contain other hashtags as well. It is also important to leave the collection open to other non-English tweets. Save the attributes of the tweets for each hashtag in excel database. This includes the Twitter ID, tweet message, list of followers of the tweet user, whether it is a retweet or not, location, if available.
2. Draw a histogram showing the popularity of the main hashtags highlighting the number of tweets per individual hashtag and in another graph the number of distinct Tweet users per individual hashtag.
3. Draw another histogram showing the proportion of tweets where location information is mentioned (if location attribute is activated in the tweet) and another one for the language of the tweet messages. Represent the finding through pie chart.
4. We now want to build a social graph where each node corresponds to a hashtag and an edge between hashtag A and hashtag B indicates that there is at least one tweet which contains both hashtag A and hashtag B. Implement a small python program that allows you to generate the above social network graph from the collected tweets.
5. Summarize in a table the main global properties of the above graph: Number of nodes, Number of edges, average degree centrality and its variance, average in-betweeness centrality and its variance, average path length and its variance, diameter, clustering coefficient, size of largest component using appropriate NetworkX functions. Comment on the obtained results highlighting any inherent limitation or characteristic of the data collection process.
6. We want to see the extent to which some global attributes can be imitated using random graph. For this purpose, use appropriate functions of NetworkX concerning the small word model whose number of node is equal to the total number of nodes of the graph in 5) but whose probability p can be chosen in such a way the clustering coefficient of this random graph approximates that of the graph in 5). Make incremental change of the probability value p until this approximation holds.
7. Identify the five highest ranked nodes in terms of degree centrality, Katz's centrality, PageRank centrality, Closeness centrality, Betweeness centrality of the graph obtained in 5).
8. Use appropriate NetworkX functions (or other alternatives) to display the distribution of the degree centrality and that of the local clustering coefficient and local betweenness centrality and closeness centrality.
9. Now we want to comprehend the key players in the graph whether they are genuine or not. For this purpose, consider the 10 most ranked in terms of degree centrality, and scrutinize the tweets which are linked to those hashtags (10 most ranked). The scrutinizing consists in applying the botometer available in https://github.com/IUNetSci/botometer-python. The program inputs a tweet user id and outputs the probability that the user id is a bot or human. You can use a threshold 0.5 beyond which a program is bot or not. The purpose is therefore to test whether a given user id (Tweet user of the tweet that contains the hashtag) is a bot or not. Draw a relevant plot which shows the proportion of bots in the top ranked hashtag.
10. We would like to test the amount of support assigned to each hashtag. For this purpose use the information about number of retweets and number of replies of each tweet involved in the hashtag and suggest an expression that quantifies the amount of support.
10. Comment the results obtained and summarize the key finding regarding behavior of users with respect to smoking cessation. Seek some literature to reinforce your interpretation.

Project 6. Analysis of Climate Change Community.

The project aims to investigate the diffusion process of Climate change topic.

Use Twitter API to collect few thousands for tweets related to hashtags *#globalwarming, #climatechange, #agw* (an acronym for "anthropogenic global warming")*, #climate*and*#climaterealists*, but feel free to suggest any other climate change hashtags of your choice if deemed more popular. The key in the collection process is that there is important number of tweets that contain other hashtags as well. See description of Tweet collection in other project description if needed. It is also important to leave the collection open to other non-English tweets as well to ensure large coverage.

1. Draw a histogram showing the popularity of the main hashtags highlighting the number of tweets per individual hashtag and in another graph the number of distinct Tweet users per individual hastag.
2. Draw pie chart illustrations showing regional location of the tweets associated to each of the above main hashtags using the location attribute of the tweet (whenever available).
3. Use other pie chart illustrations to show the language of the tweets for each of the above main hashtags.
4. We now want to build a social graph where each node corresponds to a hashtag and an edge between hashtag A and hashtag B indicates that there is at least one tweet which contains both hashtag A and hashtag B. Implement a small python program that allows you to generate the above social network graph.
5. Summarize in a table the main global properties of the above graph: Number of nodes, Number of edges, average degree centrality, diameter, clustering coefficient, size of largest component using appropriate NetworkX functions. Comment on the obtained results highlighting any inherent limitation or characteristic of the data collection process.
6. Plot the degree distribution and local clustering coefficient distribution.
7. Use Girvan-Newman algorithm to find communities in the above network through appropriate use of NetworkX functions. Compare the size of the generated communities in a table. Using your understanding of the name of the hashtag, speculate whether each community can be assigned some interpretation.
8. Now we want to comprehend the key players in the graph whether they are genuine or not. For this purpose, consider the 10 most ranked in terms of degree centrality, and scrutinize the tweets which are linked to those hashtags (10 most ranked). The scrutinizing consists in applying the botometer available in https://github.com/IUNetSci/botometer-python. The program inputs a tweet user id and outputs the probability that the user id is a bot or human. You can use a threshold 0.5 beyond which a program is bot or not. The purpose is therefore to test whether a given user id (Tweet user of the tweet that contains the hashtag) is a bot or not. Draw a relevant plot which shows the proportion of bots in the top ranked hashtag.
9. We would like to test the amount of support assigned to each hashtag. For this purpose use the information about number of retweets and number of replies of each tweet involved in the hashtag and suggest an expression that quantifies the amount of support.
10. Comment the results of the previous steps using some literature from climate change in order to reinforce your argumentation.

Project 7. Analysis of ISIS Twitter dataset

This project aims to investigate the social network of ISIS Twitter network published in darkweb website. It corresponds to 17,000 tweets from 100+ pro-ISIS fanboys from all over the world since the November 2015 Paris Attacks. The dataset includes the following: Name, Username, Description, Location, Number of followers (at the time the tweet was downloaded), Number of statuses by the user (when the tweet was downloaded), Date and timestamp of the tweet, the tweet itself. The dataset is made available through the following link https://1drv.ms/x/s!AtcJs3OTsMZuiRSaq7O2ZdVysiXd

1. Use appropriate NetworkX functions to plot the distribution of the number of tweet per user id. Discuss whether a power law can be fitted to this plot.
2. Now we want to explore the content of tweet, and seek whether a mention (@), Retweet (RT) or hashtag (#) is present. Provide a plot showing the 10 most active users in terms of retweets, use of mentions and use of hashtags in their tweet messages (a graph for each of these three categories).
3. Use VADER tool (https://github.com/cjhutto/vaderSentiment), which output sentiment in terms of POSITIVE, NEGATIVE and NEUTRAL to determine the sentiment of each tweet of the dataset. Then represent the distribution of each tweet as a point in the ternary plot.
4. Repeat the above process for the tweets associated to the ten most active users in Retweets, use of mentions, and use of hashtags. Should display three different ternary plot.
5. Now we want to build a social graph where each node corresponds to a hashtag and an edge between hashtag A and hashtag B indicates that there is at least one tweet which contains both hashtag A and hashtag B. Implement a small python program that allows you to generate the above social network graph.
6. Summarize in a table the main global properties of the above graph: Number of nodes, Number of edges, diameter, clustering coefficient, size of largest component using appropriate NetworkX functions. Comment on the obtained results highlighting any inherent limitation or characteristic of the data collection process.
7. Plot the page-rank degree distribution and the local clustering coefficient distribution.
8. Use Girvan-Newman algorithm to find communities in the above network through appropriate use of NetworkX functions. Compare the size of the generated communities in a table. Using your understanding of the name of the hashtag, speculate whether each community can be assigned some interpretation.
9. Using the information about the amount of support as a function of number of followers and number statuses of each tweet (you can suggest an expression of support of your own), rank the various hashtags according to the amount support they received accordingly. Note: the amount of support for a hashtag should be the sum of the amount of support of all tweets where this hashtag is mention.
10. We want to reduce the size of the graph in 5) by requiring that an edge between nodes is established only if there are k number of tweets mentioning the two hashtags (nodes). Set different values for k, say, 2, 3, 4, 5, 10 for instance, and draw the corresponding graph using appropriate tool. Provide in a table global attributes of each graph in terms of size of network, diameter, clustering coefficient, average degree centrality, average closeness centrality, average in-betweeness centrality.
11. Draw heat map of degree centrality distribution for each value of the threshold . [see examples of heat maps in https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0220061
12. Comment on the results using identified literature in security and terrorism of your choice.

Project 8 Mining Violence in Suomi24

The interest focuses on mining the discussion related to violence in Suomi24 Finnish forum, one of the largest Finnish internet corpus where users discuss all topics.

1. Use the online version of Suomi24 in www.suomi24.fi . Alternatively, if you have enough computational resources, you can also download the Suomi24 corpus history from The Suomi24 Corpus 2001-2017, VRT version 1.1 published in Download service | Kielipankki.
2. Construct a list of Finnish keywords related to violence (should be broad enough to include all aspects, i.e., abuse and insult related wording, hate speech related words, common bullying related abbreviation). Elaborate your own methodology to identify a large scale cyber-bullying related terms.
3. Run a simple keyword matching in Suomi24 dataset or in the online portal (crawl all search outcomes) in order to extract only those posts and the associated threats where there is a matching. Save the newly constructed database, which contains both the identified posts and associated threads.
4. Draw a bar plot showing the proportion or number of hits for each individual violence word. Draw also another plot showing the proportion of hits found on the title of the threads only.
5. Construct a social network in the following way. The nodes of the network are constituted of the set of all threats of the search outcome. An edge from a threat A to a thread B is established whenever the same violence keyword is mentioned at one post of thread A and one post of threat B.
6. Study the properties of this constructed network by reporting the number of nodes, number of edges, maximum degree, average degree, global clustering coefficient, diameter, average path length, size of giant component, size and number of communities as well as the associated quality measure.
7. Draw the degree distribution and check whether a power law distribution can be fit
8. Repeat questions 5-6, when introducing a threshold regarding the numbers of mentions of same keywords among two threads before deciding to draw an edge between the two nodes. Namely, an edge between thread A and thread B is established if there are at least k violence keywords contained in both thread A and thread B. (You can start by k=2, k=3, k=5,..). Draw a plot showing the evolution of each attribute of the network (size of giant component, average degree centrality, average path length, diameter and clustering coefficient) according to the value of k.
9. We want to test the extent to which the reciprocity relationship is fulfilled. More specifically, we want to find out whether a violence attack automatically generates a reciprocal attack. For this purpose, you need to take into account the timestamp of the posts. Therefore, we assume that whenever a thread contains an even number of violence keywords, then the reciprocity is fulfilled for the underlined thread (node). Draw a bar plot showing the proportion of threads whose reciprocity is satisfied and those not.
10. Comment on the key findings by identifying key sociology studies that support your argumentations.

Project 9: Mapping Covid-19 Vaccine Discussions in a Blog Forum

This project aims to investigate the mental health discussion taking part around Covid 19 vaccination available in Have you had covid vaccine side effects? - Health and Wellness -Doctors, illness, diseases, nutrition, sleep, stress, diet, hospitals, medicine, cancer, heart disease - City-Data Forum (city-data.com). The thread contains large number of posts. Interestingly each post contains statistical information about the author in terms of number of posts made by the author, reputation and number of reads as well as location of the author.

1. Use your own way to crawl the whole data and available statistical attributes (through API, beautifulsoup, copy and past at last resource if no automatic procedure can be implemented). Show your reasoning how this has been performed.
2. Use the location information of the authors to provide the distribution of the location in terms of number of posts generated. Show whether the Power law distribution can be fit.
3. Build a simple program that allows you to output the length of the post in terms of number of words / characters it contains.
4. Create your own subdivision of the length of the posts (e.g., length less than k1, length between k1 and k2, length between k2 and k3, …) and draw a histogram showing the number of hits in each bin. Comment on the distribution of the hits accordingly.
5. Repeat step 4) for the top 5 regions in terms of number of posts generated. Comment whether length of the post can be used as an attribute to discriminate the regions in this dataset.
6. Many posts in the dataset are written as reply to some other posts (This occurs when at the beginning of the post, there is a Quote where the name of the user is also mentioned). Consider a network graph constructed using this mentioning relation where nodes are the user names and an edge between two user names is established if one user name is mentioned in the quote of the post of the other user name (no need to be reciprocal). use appropriate NetworkX functions to plot this graph.
7. Provide a table showing the global attributes of this social graph in terms of number of nodes and edges, diameter, number of connected components, average clustering coefficient, average degree centrality and average degree closeness centrality.
8. Plot the degree centrality distribution and the local clustering coefficient distribution. Comment whether a power law distribution can be fit to the plot.
9. Use Girvan-Newman algorithm to find communities in the above network through appropriate use of NetworkX functions. Compare the size of the generated communities in a table.
10. Use the author's Reputation information to identify communities that have higher reputation. You can simply consider the reputation of a community as the sum of the reputations its members.
11. Discuss and comment, and use appropriate health literature in order to reinforce your interpretations.

Project 10: Violent Topic Diffusion

This project explores the diffusion of information in a special information forum known for its violent discussion topic. More specifically, we focused on Ummah dataset (1.2 GB) collected in the period 2002-2010, available in Dark Web Forum at https://www.azsecure-data.org/dark-web-forums.html

The dataset contains a set of violent topics, which include Suicide Bomb, Anti-America, George Bush, Wear Hijab, Honor Killing, Nuclear Weapon. Each topic is organized by a large set of threats and associated posts (replies).
Initially, the project aims to apply the SIR diffusion model to each topic of the dataset.

In the web forum context, the Susceptible Class is defined as users who have interest in a topic and might read posts (comment or thread) on it, so that, in the future they will become authors. Once they become authors, they will belong to Infective Class. In other words, susceptible users will either write a thread or leave comments on other threads. Next, the infectious user becomes at state Recovered when his posts lose infectivity to others in the sense that his message did not trigger timely response from any other user. Therefore, there is a need to define what might be timely responsiveness and not-timely responsiveness, according to your observations of the response time of the users. The total population consists of the susceptible class, the infective class, and the recovered class.

1. Suggest your own heuristic to model the timely and non-timely response taking into account the statistics of the users' responses times.
2. Suggest a script that would allow you to calculate the number of users at state Susceptible, Infectious and Recovered.
3. Use SIR diffusion mode in NDlib library in order to plot the variation of number of Susceptible, Infectious and Recovered individuals over time, estimate the alpha and beta parameters corresponding the rate of propagation of infectious and recovered states. Use appropriate goodness of fit measures in order to estimate the goodness of SIR model for each topic
4. Study the paper "An SIR model for violent topic diffusion in social media" by Jiyoung Woo, J. Son and H. Chen. You will notice a new SIR model is proposed with a variable number of population taking applied to the same dataset (Ummah dataset) but where a single SIR model is applied to the whole dataset instead of repeating the reasoning per each topic as in previous case. Suggest a way to implement their approach using NDlib library and NetworkX packages.
5. Test the goodness of fit and compare the results with the previous case in question 3.
6. Use appropriate literature from sociology of violent topic diffusion in order to back up your findings.
7. Now we would like to redesign the interpretation of the susceptible and infectious case taking into account the length of the message by the user. More specifically, design a program that allows you to track the lengths of the posts posted by a single user. Next, consider the average length of these posts, say L_i (for i^{th} user) as a threshold. Therefore, a user i is considered to be susceptible when he has not written any posts, or the length of his post is less than L_i, while it will become Infectious when his post length exceed L_i.
8. Study the implementation of the above heuristic as SIS diffusion model and use appropriate NDlib and NetworkX package to see how the actual trends fits with the SIS model. (There is no need to distinguish between topics in this case, as we consider the user to be infectious if he posted a post whose length exceeds Li in any of the discussion topics.
9. Comment your findings using appropriate argumentation from relevant literature.

Project 11. Citation Network Analysis

We would like to explore the citation analysis in “Climate Change Mitigation” using Scopus API in the last five years. Get familiar with Scopus API and how you can export the results and how you can discriminate the authors and location attributes.

1. Construct a small database containing the list of papers outputted by the API as output the query “climate change mitigation” with title, author names and country of the affiliation and a list of keywords, if available.
2. Perform a simple histogram construction that allows you to rank the authors in terms of number of publications found and the number of distinct collaborators (co-authors).
3. Perform another histogram that allows you to rank the keywords that are attached to the largest number of articles.
4. We would like to construct a graph using the co-authorship relation where two authors are linked via an edge whenever they jointly published at least one single paper. Design a program that allows you to implement the above strategy.
5. Provide in a table the global attributes of this graph in terms of number of nodes, edges, largest component, average degree centrality, global clustering coefficient, diameter, average path length.
6. Using the concept of Erdo-number, identify the author who has got the largest number of collaborators (co-authored the papers) and assume this author will be assigned Erdos number 0 and direct co-authors will be assigned Erdos number 1, while other authors who co-authors with the first collaborators and not with that of Erdos number 0, will be assigned Erdos number 2, etc.. Draw the distribution of this new Erdos number in terms of number authors that fall in each Erdos number category.
7. Visualize the graph of authors with Erdös number 1 and 2.
8. Plot the clustering coefficient distribution of the above graph and discuss whether a special polynomial fitting can be achieved.
9. Discuss your findings in the light appropriate bibliometric literature. You can also use alternative citations (e.g., Google citations for the authors with the top Erdös number to see if there is any match).

Project 12: Parking behavior Analysis

Vehicles in search of on-street parking create an environmental and economic impact: they increase network traffic flow and congestion, heighten pollutant emissions levels, create additional noise, give rise to time delays for through vehicles, and lead to potential safety hazards when vehicles maneuver into or out of on-street spaces. Despite extensive negative externalities for individual drivers and society, the search for parking is a little researched area. The aim of this study was to review and identify factors that influenced an individual's on-street parking search decisions.

The first approach to perform is to look into parking behavior through Twitter hashtag dataset.

1- Set up a twitter developer account and collect Tweet related to hashtag "#parking", "#parkinglot". For this purpose, you can inspire from Tweepy implementation (see tutorial at https://riptutorial.com/tweepy), also see examples of text processing in Python in NLTK online book at https://www.nltk.org/book/. See an example of program of collecting tweets for a given hashtag in https://www.promptcloud.com/blog/scrape-twitter-data-using-python-r/. You should ensure the data is large enough to ensure there are connections among a large number of tweet users and the existence of users who have several tweets, and tweets that contain more than one hashtag in order to ensure satisfactory of the collected data. Save the attributes of the tweets for each hashtag in a single excel database. This includes the Twitter ID, tweet message, list of followers of the tweet user, whether it is a retweet or not, location, if available, set of hashtags present in the tweet. You would need to re-initiate the collection process there is no tweet which has more than one hashtag.

2. Identify the top ten influencers (tweet user id) in each hashtag class ("#parking", "#parkinglot".) in terms of number of tweets generated.
3. We now want to build a social graph where each node corresponds to a hashtag and an edge between hashtag A and hashtag B indicates that there is at least one tweet which contains both hashtag A and hashtag B. Implement a small python program that allows you to generate the above social network graph from the collected tweets for parking and parkinglot classes.
4. Summarize in a table the main global properties of the above graphes: Number of nodes, Number of edges, average degree centrality and its variance, average in-betweeness centrality and its variance, average path length and its variance, diameter, clustering coefficient, size of largest component using appropriate NetworkX functions. Comment on the obtained results highlighting any inherent limitation or characteristic of the data collection process.
5. Identify the five highest ranked nodes in terms of degree centrality, PageRank centrality, Closeness centrality, Betweeness centrality in each class.
6. Use appropriate NetworkX functions (or other alternatives) to display the distribution of the degree centrality and that of the local clustering coefficient and local betweenness centrality and closeness centrality.
7. Now we want to comprehend the key players in the graph whether they are genuine or not. For this purpose, consider the 10 most ranked nodes (hashtags) in terms of degree centrality, and scrutinize the tweets which are linked to those hashtags (10 most ranked). The scrutinizing consists in applying the botometer available in https://github.com/IUNetSci/botometer-python. The program inputs a tweet user id and outputs the probability that the user id is a bot or human. You can use a threshold 0.5 beyond which a program is a bot or not. The purpose is therefore to test whether a given user id (Tweet user of the tweet that contains the hashtag) is a bot or not. Draw a relevant plot which shows the proportion of bots in the top ranked hashtag for each class.
8. We would like to test the amount of support assigned to each hashtag. For this purpose, use the information about number of retweets and number of replies of each tweet involved in the hashtag and suggest an expression that quantifies the amount of support for the previous 10-hashtags of each class.
9. By taking into account the timestamp of the tweet, draw the histogram of the number of tweets of each hashtag per time interval. Set the network status for each day or alternative time interval taking into account the time scale of the earliest and latest collected tweet, and draw the time evolution of the clustering coefficient.
10. Use appropriate literature on user car parking behavior to comment on the obtained results.

Project 13. Social Network Blog Analysis

Choose an active blog community of your choice with an available API to ease data collection.
Proceed in the following way to construct the social network graph.

- Start with a list of most cited blogs at a specific time of your choice and select a time window (it should include the time of most cited blog) that you can use to collect posts and blogs occurring within that time interval.
- Make some reasonable assumptions in terms of the maximum number of posts that will be retrieved.
- Typically, each post contains a link of the parent blog, date of the post, post content and a list of all links that occur in the post's content.

1. Elaborate on the choice of blogs and size of data collection.
2. Plot the number of posts per day over the span of the collected dataset
3. We would like to represent the collected data as a cluster graph where clusters correspond to blogs, nodes in the cluster are posts from the blog, and hyper-links between posts in the dataset are represented as directed edges. Only consider out-links to posts in the dataset. Therefore, remove links that point to posts outside the collected dataset or other resources on the web (images, movies, other web-pages), and also those edges that point to themselves if any. This is to keep track of timestamp for temporal analysis.
4. Study the global properties of the established network: number of noes, number of edges, clustering coefficient, diameter, size of giant component, average in-degree centrality and out-degree centrality and their associated variance, average path length and its variance, average closeness centrality and its variance, average in-betweeness centrality and its variance.
5. Trace the in-degree centrality distribution and check whether a power-law distribution can be fit using appropriate statistical testing
6. Trace the out-degree centrality distribution and check whether a power-law distribution can be fit using appropriate statistical testing
7. Now investigate the temporal variation of popularity. For this purpose, collect all in-links to a post and plot the number of links occurring after each day following the post. This creates a curve that indicates the rise and fall of popularity. By aggregating over a large set of posts, you should a obtain a more general pattern.
8. Check whether a power-law distribution can be fit
9. identify appropriate literature to comment on the obtained results and the limitations

Project 14. Covid-19 and Hashtag diffusion

This project investigates a large scale Covid-19 Twitter dataset available at https://github.com/lopezbec/COVID19_Tweets_Dataset. The dataset is organized by hour (UTC) and each hour contains five tables: (1) "Summary_Details", (2) "Summary_Hastag", (3) "Summary_Mentions", (4) "Summary_Sentiment", and (5) "Summary_NER (Named-Entity-Recognition)". The dataset is made of billions of tweets and still is constantly updated. The summary hashtag consists of the top five popular hashtags in tweets collected at a given hour (UTC). It also provides for each tweet Likes count, Retweet count and sentiment label.

1. Use appropriate tool to save the dataset in appropriate format. The actual text of the tweet message is not needed (tweet id will be enough). Using the information in the Sumary_Hashtag attribute of the data, draw a plot showing the distribution of the hashtags in terms of number of tweets citing the hashtag. Does the graph follow a power law distribution? Use statistical significance of curve fitting to show whether such fitting is significant or not.
2. Repeat the preceding for the named entity as provided in Summary_NER attribute, and indicate whether a power law distribution can be fit or not.
3. We want to focus on the timely evolution of the hashtags. Consider the five most frequent hashtag (cited by largest number of tweets) that you may infer from 1).
4. We want to reconsider the top hashtags by taking into account the replies and likes count. For this purpose, assume that the score of the hashtag is calculated so that in first case (replies), we add the replies count of each tweet that contains that hashtag. While in the first case, we will use likes count instead of relies count. By doing so, identify the top five hashtags according to replies count, and the top five hashtags according to likes count.
5. For each of these hashtag, suggest a plot which shows the timely evolution of this hashtag over a period of few months. You may create a weekly subdivision, where you count the total number of mentioning of this hashtag for each week, and then draw a plot of count versus weeks. Also for each week calculate the statistics of the hashtag count in terms of average, standard deviation, kurtosis and skewness. You may also plot to show on the same graph the evolution of the mean and standard deviation. Identify, whether you may notice cases where some weeks have zero count and then start picking up again. Discuss the evolution of the various hashtags according to replies and likes count.
6. We would like to evaluate the evolution of each hashtag in terms of sentiment score. For this purpose, use the logits data provided in sentiment attribute of the dataset. More specifically, for each week, add the logist_negative (as well logist_positive, and logist_neutral) of all tweets mentioning the underlined hashtag. The hashtag will therefore be assigned a sentiment label that has the highest value among negative, positive and neutral logist values. Use a plot where you represent the evolution of the positive by its positive score, while negative sentiment is represented by a negative value (where the value corresponds to the total logist_negative). Discuss how hashtag count correlates with sentiment score.
7. Now we want to model the speed of hashtag diffusion over the network. Consider that the propagation speed is defined by the following.

   $$Ps = (R_1 + R_2 + .. + R_n) / n$$

   where R_i is total count of retweet of all tweets mentioning the hashtag S in week i. n is the total number of weeks

   Use the above formula to calculate the speed of the hashtag at three different periods that you may distinguish: starting time, peak time and flat time where the associated tweets are getting less replies score.

8. Comment on the results using identified literature of Covid-19 of your choice.

Project 15: Independent Cascade in Game of Throne

This project investigates the famous of Game-of-Throne dataset, which contains all the battles of game-of-throne and the characters involved as described in "Song of Ice and Fire books". It has 8 seasons, and you can conduct network analysis for each season as well. The dataset has originally 187 nodes (characters) and 684 weighted edges accounting for 7,366 interactions. See a description of the dataset in https://www.kaggle.com/mylesoneill/game-of-thrones. You may notice that this dataset is quite popular and many investigations and codes are widely available (e.g., https://shiring.github.io/networks/2017/05/15/got_final; https://github.com/mathbeveridge/asoiaf). So, feel free to explore the content as it may help some forthcoming tasks. It is important that you use as much NetworkX functions and dblib library as possible instead of relying on other implementations.

1. We want to explore the social network attributes of the graph at a given season, say season 4. Draw high level quality of the social graph and determine the global attributes of number of nodes, edges, average degree centrality, average closeness centrality, average in-betweeness centrality, diameter and average path length.
2. Draw degree centrality distribution and local clustering coefficient distribution. Discuss whether a power law distribution can be fit. Discuss whether a polynomial law can be accommodated.
3. Identify the 10 most influential characters according to Katz's centrality and according Page rank's centrality measures. Draw the corresponding plots.
4. Now we want to create some randomness in the graph topology. Write down a script that generates a random binary number according to a Poisson distribution with a specific rate parameter λ (may choose to generate a random number in the unit interval at the beginning and then set a threshold 0.5 so that if the generated number is less than 0.5, it will be set to zero, otherwise, it will be set to 1. Next, consider the vector consisting of the total edges of the network, and generate a binary random vector using the previous reasoning. The newly generated binary vector indicates which one among the edges of the original network will be preserved and which one will be deleted.
5. Se the rate parameter to vary from 0.1 to 0.5 (0.1, 0.2, 0.3, 0.4, 0.4), for each realization of the binary vector, calculate the average clustering coefficient and average degree centrality. In order to take into account for the randomness, you should repeat the procedure several time (e.g., 10 times). Calculate the statistics of the variations of the average clustering coefficient and the average degree centrality as a function of the rate parameter λ.
6. Now we want to investigate the propagation of the information through this network using the Independent cascade model. Use Netlib library to model an independent cascade model whose input is the network at Season 1 and where node threshold probabilities were set at random. You should outcome an interactive graph illustration the node (use different color) where the information is transmitted to at each subsequent time. Write a program that allows you to run this process 100 times (a process is when you run the model from t=1 till the end where the information cannot be further transmitted) and for each one you output the percentage of the node (characters) where the information is transmitted to. The latter indicates the network coverage for each process.
7. Repeat the preceding step, when you fix the node threshold probability to 0.5 for all nodes.
8. Repeat the preceding when the node threshold probability is 0.1 and 0.8. Comment on the influence of the threshold probability on the network coverage.
9. We want to study the independent cascade model where the threshold are generated through another model. Study the program provided in GitHub - furkangursoy/RLforDiffPred: Source code and replication steps for "Predicting Diffusion Reach Probabilities via Representation Learning on Social Networks" . Replicate this reasoning for your graph network and compare the obtained coverage with previous approaches.
10. Use relevant literature to back up your finding at each step of the above

Project 16: Open to new suggestion

If you have a concise idea in graph analysis that you want to pursue for personal reasons, feel free to get in touch to discuss the details