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

Large scale dataset

Project 1

Consider the Twitter dataset available Twitter Edge Nodes | Kaggle, which shows the friends / follower relationship.

1. Use NetworkX to draw and plot the degree distribution of the above graph. Separate the in-degree distribution and out-degree distribution and average degree degree distribution (average between in-degree and out-degree). (Draw three separate degree distributions).
2. Provide a source code that draws a power law distribution for in-degree distribution, out-degree distribution and average degree distribution.
3. Use appropriate approach (curve fitting at 90% or 95% confidence level) to test the good ness of fit of the linear log-log distribution.
4. Use NetworkX or similar package to identify the top five edges with the highest edge betweenness scores and the ten nodes with the highest node betweenness degrees. Compare the relationship between nodes betweenness results and edge betweenness results.
5. Use NetworkX clustering function to compute the clustering coefficient of nodes of the network. Consider a histogram of 10 bins on the values of the clustering coefficient and draw a plot showing the number of nodes falling in each bin.
6. Repeat questions 2) and 3) by fitting a power law distribution of the clustering coefficient distribution and goodness of fit.
7. Use NetworkX or similar package to identify strongest connected and weakest connected component of the network and the number of strongest components and weakest components. Suggest an approach to label these components at your convenience.
8. Write a program that identifies the strongest connected components that are connected. We consider two strongest components connected if there is a link from at least one node of one component to another. Display the corresponding graph showing the interlink between the groups (strongly connected components).
9. We want to relax the assumption that two groups (strongest connected components) are connected only if there are two nodes of the two groups, which are direct neighbor, to case where the geodesic distance is up to 10 (the shortest distance between the two group is less than 10). Write a scrip to implement this relaxation and display the corresponding group graph.
10. Use label propagation algorithm in NetworkX to identify the various communities. In case the execution time is extremely high, You may consider taking a random selection of a subgraph of 10000 nodes centered around one of the nodes with top 5 betweenness centrality values containing as much connected nodes as possible. Should fully describe a code and pseudo code performing this randomization. Use NextworkX to evaluate the quality of the generated communities, testing the cohesion and overlapping between these communities. Use appropriate visualization to illustrate these communities. You may inspire from the program in Social network visualization and analysis | Kaggle.
11. Use appropriate literature to justify the results obtained at each step and comment on the findings.

Project 2. Movie database 1

Consider the IMDB 5000 Movie database available at shared Google drive
https://1drv.ms/x/s!AtcJs3OTsMZuiRctyVt3IVt4lSup

The database contains several attributes for each movie, including main actor, second actor, third actor, director, movie genre, various user ranking attributes, budget, keywords.

1. Suggest simple preprocessing to remove visible inconsistencies in the attributes of the database.
2. We want to explore some statistical properties of the movies. First draw the distribution of ratings, highlighting the minimum, maximum and average rating. You may want to inspire from the program EDA on IMDB Movies Dataset | Kaggle, which uses closely similar dataset. Indicate whether some polynomial fit can be fitted to the rating distribution.
3. Use Pearson correlation to evaluate the correlation between the movie rating (imdb_score) and the number of votes; the movie rating and director facebook likes; the movie rating and actor_1_facebook likes; movie rating and gross: movie rating and movie_facebook_like. You may calculate the p-value of each correlation to find out whether the correlation is statistically significant. You may use the previous link and any statistics package for p-value calculus.
4. We consider the distribution of the movie genres. Draw the plot showing the proportion of each genre. Summarize in a table the average and standard deviation of movie rating, movie-facebook_like, director_facebook_like for each genre.
5. Suggest a bipartie graph from movie director set to movie genre set where a link from a given movie director X to a given genre Y if there is at least one movie directed by X and whose genre is Y. Suggest a script that performs this operation and display instances of this graph of your choice.
6. We want to transform the above bipartie graph into a weighted social network graph of movie directors. For this purpose, we consider two movie director nodes are connected if they produced the same movie genre and the number of movies of same genre produced by the two directors corresponds to the weight of the edge connecting the two nodes. Write a script that constructs this social network graph and save the adjacency matrix of this graph as an excel file.
7. Use networkX functions to identify i) the node of the highest of betweenness centrality; ii) the node of the highest degree centrality; iii) the node of the highest closeness centrality; iv) the node of the highest Katz centrality.
8. Use networkX functions to identify communities through Girvan-Newman algorithm. Use appropriate plotting to highlight the different communities in the graph. Use a table to provide summarize key characteristics of each community (number of nodes, number of edges, diameter, average path length, average degree).
9. Consider another social of movie directors where the nodes X and Y are connected if there is at least one actor (actor 1, actor 2 or actor 3) that played in one movie of X and another movie of Y. Similarly, we consider the number movies fulfilling this requirement as the weight of the edge between X and Y. Write a script that displays the new graph and saves its adjacency matrix in an excel file.
10. Repeat question 6) and 7) for this new graph.
11. Suggest appropriate metrics to compare the two graphs of 5) and 8).
12. Use appropriate literature from entertainment and movie making to comment on the findings and discuss the results accordingly.

Project 3. Analysis of Movie database 2

Consider the IMDB 5000 Movie database available at shared Google drive
https://1drv.ms/x/s!AtcJs3OTsMZuiRctyVt3IVt4lSup

The database contains several attributes for each movie, including main actor, second actor, third actor, director, movie genre, various user ranking attributes, budget, keywords.

1. We want to study the yearly release of movies in the database. Draw a histogram showing the evolution of the number of movies release per yet and conclude whether a specific trend can be observed.
2. We want to test the hypothesis that popular movies are expensive. Suggest a correlation analysis between movie cost and popularity variables (Imbdb score and movie face book like). You may inspire from the EDA on IMDB Movies Dataset | Kaggle, and provide also p-value of the statistical significance.
3. We want to test the hypothesis that an actor often plays on the same movie genre. Suggest a plot showing the proportion of actors who played on movies of the same genre in more than 80% of played movies, movies on two distinct genres only in more than 80%, movies on three distinct genres only in more than 80%. Conclude whether the hypothesis is statistically sustainable or not.
4. We want to consider a bipartie graph of network of actors -set of Actor 1 and set of Actor 2, where a link between Actor 1 and Actor 2 occurs if they played in the same movie. Given that some actors in set Actor 2 can also be actor in set Actor 1, the graph is not really bipartie graph. Write a script that outputs a weighted graph where the weight corresponds to the number of movies that the same Actor 1 and Actor 2 co-occur. Save the adjacency matric of this graph in an excel file.
5. Use networkX functions to identify i) the node of the highest of betweenness centrality; ii) the node of the highest degree centrality; iii) the node of the highest closeness centrality; iv) the node of the highest Katz centrality.
6. Use appropriate NetworkX functions to compare the results of Girvan Neuman and label propagation algorithm for community detection algorithms of the constructed graph. Provide the result in a table, highlighting the number of communities detected, its size, diameter, average path length and average clustering coefficient of each community. Use appropriate visualization toolkit to visualize the communities provided by each algorithm.
7. Explore the correspondence of the identified communities with genre, keywords, budget, and any other attributes provided in the database.
8. We want to predict the popularity of the movies according to the popularity of the movie director, actors (Actor 1, Actor 2 and Actor 3) and budget spent. For this purpose, build a simple machine learning model using Random Forest where the features are constituted of movie director facebook like, Actor 1 facebook likes, Actor 2 facebook likes, Actor facebook likes and budget, while the output is the movie popularity. Assume that the 80% of the first database is used for learning and the last 20% for testing. Calculate the performance in terms of prediction accuracy.
9. We want to tune the classifier parameter and test various other combinations of the inputs. Suggest a script that compares the performance metric (prediction accuracy) for several other combination of features. You may suggest additional features based on the data originally available in the database. Conclude on the best indicator (s) (features) that provides highest prediction rate. Represent the results in a table, of your choice.
10. Compare the result of Random Forest to three other machine learning classifiers of your choice and one deep learning model of your choice.
11. Use appropriate literature from entertainment and movie making to comment on the findings and discuss the results accordingly.

Project 4. Netflix dataset

Consider the Netflix movies and TV shows dataset available from Netflix Movies and TV Shows | Kaggle. The dataset contains description of movies including title, director, cast, country, data of release, rating, movie duration and short textual description.

1. Draw a histogram of movie duration in the database and identify any particular trend. (should use any curve fitting to justify the trend). You may inspire from the related programs available in Kaggle for this particular dataset.
2. We want to test the hypothesis whether the duration of the movie is linked to the rating of the movie. Write a script that calculates Person correlation and p-value between the movie ranking and its duration (movie ranking should be in full numerical scale prior to this operation).
3. We want to construct a social network between the various movies. First use a simple labelling of the movies to ease graphical illustration. Write a script that constructs a social network of the movies in the following way. First, for each actor full name (pay attention that the name is composed word of two or more terms) in the cast category, identify the list of movies that this actor played in. You may represent each actor in cast as an index file, pointing towards the movie ID he played with (You may introduce a simple IDs for movies to avoid lengthy movie title). Second, consider the network constituted of movie IDs as nodes, and where an edge between two nodes is established if there if there is at least one actor who played in both movies. Third, assume the above network is weighted graph where the weight is evaluated by the number of actors that played in both movies. Fourth, use appropriate visualization tools to provide a dense graphical representation of this network. You may show the IDs of some interesting nodes of the network.
4. Write a program that outputs some statistical properties of the generated network, which include: number of singletons, size of largest component, number of components and their sizes, diameter of the network, average path length, average clustering coefficient. Provide the result in a table.
5. Use NetworkX to determine the k-cliques communities and Louvain communities. Use also NetworkX to evaluate the quality of these communities using function performance.
6. Now we want to test the partition using the rating mechanism. For this purpose, write a script that calculates the rating of each community in terms of average rating of the movies forming the community and its standard deviation. The smaller the standard deviation value, the better the fitting of the community with the rating criterion.
7. Now we want to test the partition using the movie textual description attribute. Write a script that calculates the number of overlapping words among textual description of each movie in the community, the higher this number, the better the fitting of the community with this metric.
8. We want to approximate the underlined network by a random graph G(n,p). Determine the probability value p of this random graph that best approximates the number of singletons, diameter and average path length of the original network.
9. Use approximate literature from movie, leisure literature to comment on the findings and results of each specification.

Project 5: Link prediction 1

Consider the Wikipedia vote network dataset, available at SNAP: Network datasets: Wikipedia vote network (stanford.edu). The dataset contains a two columns dataset of user IDs, where for each row, the first element (User ID) voted for the second User ID.

1. Write a script that builds a directed graph following the above description where the nodes correspond to the user IDs and a link indicates that one user ID voted for the second one. Use appropriate visualization toolkit to display sample of the graph. Construct the adjacency matrix of the graph and save it as an excel file.
2. Use NetworkX functions to display in a table the
3. Use NetworkX functions and script to draw the in-degree distribution and out-degree distribution. Write a script that draws a power-law distribution and use a 90% confidence level to check whether the power law distribution can be fit to the data.
4. Use networkX function to calculate the reciprocity of the network.
5. Use NetworkX functions to identify the top 10 User ID with the largest betweenness centrality scores, and the top 10 closeness centrality values, and display for each the top2 the resulting graph when centered around the underlined node.
6. Use NetworkX to identify k-core communities and k-plex communities in the original graph. Provide in table statistics of the obtained communities (size, diameter, average path length and average clustering coefficient).
7. We want to perform link prediction using the above Wikipedia vote graph. For this purpose, we first proceed through a machine learning approach and create a manually labelled dataset. Especially, we deliberately create wrong examples by removing at random one edge of the original network, and repeat this process until a small fraction of the original edges are removed (leading to a transformed graph). Therefore, in the link prediction task, we would like to predict these missing edges when inputted by the above transformed graph. This labelled dataset of edges consists of positive and negative examples. For this purpose, we adhere to the following: Given a random edge e, if the removal of e will not disconnect the residual network, e is selected as appositive example. Otherwise, it is a negative example. – Select around 10% of edges as positive examples. To generate negative examples, we randomly select an equal number of node pairs from the network which have no edge connecting them. (node pair (u,v) in a negative sample satisfies that v can reach u but u cannot reach v in the network). Write a script that allows you to automatically create positive and negative labels using the above construction. You may inspire from already existing constructions available in papers that used this dataset - a simple google search of the dataset would yield several examples of implementations
8. Use a Node2vec embedding as a feature vector and a machine learning model of your choice to test and validate the above construction on the provided dataset. You may inspire from Link Prediction Recommendation Engines with Node2Vec | by Vatsal | Towards Data Science for Node2vec use in link prediction.
9. Use appropriate literature from link prediction and trust theory to comment on the findings of the above specifications.

Project 6. Link prediction 2

In this project, we are interested into link prediction problem of signed network graph.

We will firstly focus on bitcoin alpha trust signed weighted, which shows the network of who-trust-whom people where members rate other members in the scale (-10 to +10) corresponding to full distrust and full trust. The dataset is available at SNAP: Signed network datasets: Bitcoin Alpha web of trust network (stanford.edu). You may consult the references provided in the link for further reading.

1. We consider the dataset as weighted graph -the nodes are the first and second column of the dataset and the weights are the third column. Write a program that generates the adjacency matrix of the above graph and save it as an excel file.
2. Use NetworkX functions and your program to plot the in-degree distribution and out-degree distribution, and fit power-law distribution. Check whether the power law distribution can be fit at 80% confidence. (Use curve fitting and confidence bounds in curve_fit library). Calculate the average clustering coefficient and diameter using NetworkX functions.
3. Identify the communities of the network using k-plex algorithm as implemented in Networkx. Use appropriate visualization tools to display a dense representation of these communities.
4. Now we want to compare the result with that obtained when the sign information is ignored from the network. For this purpose, consider two graphs from the original. The first one assumes only the edge of positive weights (the edge of negative weights are not represented in the network). The second one assumes only the edges of negative weights (the edges of positive weights are not represented), and the negative weights are turned into positive. Write a script to save the adjacency matrices of the two newly created graphs into an excel file.
5. Repeat 2) and 3) for the two positive and negative networks created in 4). Comment on potential relationships between the overall network and its positive and negative parts.
6. Now we want to predict the sign of the edge regardless of the weight value. For this purpose, consider the following construction. For each edge (m,n) linking node m and n, we consider a three dimensional vector: L1 = edge betweenness centrality, L2 = (degree_centrality_of_m + degree_centrality_of_n)/2, L3= (closeness_centrality_of_m + closeness_centrality_of_n)/2. Therefore, we consider the feature vector [F1 F2 F3]. Construct a machine learning model where a random selection of 80% of edges are used for training and 20% for testing (You may repeat this process 10 times and then take the average to account for this randomness). In the training phase, each edge is assigned its corresponding feature vector F and the output (+1 or -1, depending whether the weight is positive or negative). Suggest a machine learning of your choice to assess the accuracy of the sign prediction of the remaining 20% test data. You may use machine learning toolbox that conducts testing on an ensemble of machine learning modules and then report the three or five top scoring ones. You report the result in terms of F1 score and accuracy score.
7. Repeat 6) if we want to predict the whole weight value (either positive or negative).
8. Identify appropriate literature to comment on the results and limitations of the approach followed.

Project 7: Erdos number

Consider the data of Erdos Number project The Erdös Number Project Data Files - grossman (google.com). We want to study the predictability of existence of co-authorship collaboration among authors.

1. Use the datafile on different Erdos numbers and suggest a script to display a graph showing the yearly evolution of number of articles published by all authors. You may restrict to Erdos 0 in case of difficulties in handling the remaining dataset.
2. For ease of manipulation, write a script that labels all the names of Erdos0 and Erdos1 using numerals of your choice.
3. Consider the data in Erdos 1, write a script that outputs the degree distribution matrix and displays it as a histogram. Check whether a power law distribution can be fit. Use 80% confidence level to seek the statistical significant of the fit.
4. Repeat 3) such that the degree distribution is computed only for authors of Erdos0; that is, for each Erdos1 author, we seek the number of co-authors who have jointly co-authored with Erdos.
5. We want to test the hypothesis that writing a higher number of people with Erdos entails having a high number of collaborators in overall as well.. For this purpose, write a script that displays for each degree in Erdos0 list, the corresponding average value of degrees in Erdos1. (For instance, for each author who has written two papers with Erdos, we check their status in Erdos1 and retrieve the total number of their count and take the average). Represent this in 2D graph and use box-plot to take into the standard deviation of the average operation. Comment on the trend whether it is increasing / decreasing to answer the hypothesis.
6. Consider Erdos2 data and suggest a script to display a high level representation of the collaboration network up to Erdos2 (excluding Erdos himself, which has direct connection with every user in Erdos0). An edge between two nodes (authors) indicate that they have co-authored a paper together. Provide some statistical parameters of this network: number of nodes, number of edges, diameter, average clustering coefficient, average path length.
7. Use NetworkX to plot the degree distribution of the above graph, and check whether a power law distribution can be fit.
8. Use label propagation algorithm in NetworkX to determine the various communities of the network and use appropriate visualization tool to display the communities. Summarize in a table the characteristics of each community in terms of number of nodes, average path length, diameter and average clustering coefficient.
9. Use the function of random graph generator (you may utilize configuration_model function in NetworkX, or use any other available implementations) to generate a random graph whose number of nodes is equal to the size of graph in 6) and the same degree distribution as 6). Calculate the average Erdos number of the connected nodes (The Erdos number of nodes are the same as that of original graph in 6) and the Erdos number project; however, the random graph may likely generate graph where some nodes of the original network are not connected and permutation of the nodes, which result in different total Erdos number. Repeat this process 100 times with various random number generator and display a graph showing the various realizations of the (average) Erdos number represented as a dot in the graph. Use a contour plot of your choice to highlight the distribution of Erdos number values.
10. Use appropriate literature to comment on the findings and discuss the limitations.

Project 8. Paper citation network 1

This project investigates network created through data collection through literature analysis, aiming to gain enough insights in understanding the topic under consideration. In this project we focus on the topic of "Automatic text summarization".

1. First, collect data related to automatic text summarization in arXiv API using keywords like "automatic text summarization", "text summarization", "automatic document summarization" (feel free to extend the list with similar wording) from 2000 onwards. This generates a CSV collection of files containing title, date, article_id, url, main topic, all topics, authors, year. Collect the first 1000 results if available. Save the result in excel file.
2. Write a script that displays the histogram of the yearly evolution of the collected publications, and display the graph.
3. Similarly to 2), display the histogram showing the proportion of the main topics in these articles (list of main topics are in the x-axis and their proportions).
4. Repeat 3) considering the attribute (all topics).
5. Now we want to construct a network of main topics. For this purpose, we assume that two main topics (now playing the role of nodes of the network), say T1 and T2, are connected if there is at least two articles where the content of 'all topics' category in T1 and T2 is similar to at least 50%. Suggest a script that implements this construction. We also consider that the graph is weighted where the weight is determined by the number of articles having the property "at least half of the content of 'all topics' category are similar". You may inspire from a related program in Link Prediction Recommendation Engines with Node2Vec | by Vatsal | Towards Data Science with the difference in the network construction !! (The link also provides good example on collecting articles using arXiv articles). Determine the adjacency matrix of this network and save it in excel file.
6. Use NetworkX function to determine the key characteristics of this network: number of nodes, diameter, average path length, average clustering coefficient, number of connected components. Write the result in a table.
7. Use Girvan-Newman algorithm to determine optimal communities of this graph. Use relevant metrics from NetworkX to test the quality of this partition. Comment on the possible interpretations of the generated communities using your knowledge about the topics. Plot the network and the communities.
8. We want to construct another network by digging into author information attribute and extracting authors affiliation (name of organization only), and then we assume the nodes represent the organizations and the link between two node indicates that the two organizations written a joint paper. While the number of joint articles represent the weight of this edge. Write a script that allows you to generate this network. Use relevant visualization to plot the network.
9. Repeat 6) and 7) for the newly generated graph.
10. Use relevant literature to comment on the findings and limitations of the study.

Project 9. Paper citation network 2

This project investigates network created through data collection through literature analysis, aiming to gain enough insights in understanding the topic under consideration. In this project we focus on the topic of "Emergency commnications".

1. First, collect data related to automatic text summarization in arXiv API using keywords like "emergency communication", "crisis communication", "Handling emergency event" (feel free to extend the list with similar wording) from 2000 onwards. This generates a CSV collection of files containing title, date, article_id, url, main topic, all topics, authors, year. Collect the first 1000 results if available. Save the result in excel file.
2. Write a script that displays the histogram of the yearly evolution of the collected publications, and display the graph.
3. Similarly to 2), display the histogram showing the proportion of the main topics in these articles (list of main topics are in the x-axis and their proportions).
4. Repeat 3) considering the attribute (all topics).
5. Now we want to construct a network of main topics. For this purpose, we assume that two main topics (now playing the role of nodes of the network), say T1 and T2, are connected if there is at least two articles where the content of 'all topics' category in T1 and T2 is similar to at least 50%. Suggest a script that implements this construction. We also consider that the graph is weighted where the weight is determined by the number of articles having the property "at least half of the content of 'all topics' category are similar". You may inspire from a related program in Link Prediction Recommendation Engines with Node2Vec | by Vatsal | Towards Data Science with the difference in the network construction !! (The link also provides good example on collecting articles using arXiv articles). Determine the adjacency matrix of this network and save it in excel file.
6. Use NetworkX function to determine the key characteristics of this network: number of nodes, diameter, average path length, average clustering coefficient, number of connected components. Write the result in a table.
7. Use Girvan-Newman algorithm to determine optimal communities of this graph. Use relevant metrics from NetworkX to test the quality of this partition. Comment on the possible interpretations of the generated communities using your knowledge about the topics. Plot the network and the communities.
8. We want to construct another network by digging into author information attribute and extracting authors affiliation (name of organization only), and then we assume the nodes represent the organizations and the link between two node indicates that the two organizations written a joint paper. While the number of joint articles represent the weight of this edge. Write a script that allows you to generate this network. Use relevant visualization to plot the network.
9. Repeat 6) and 7) for the newly generated graph.
10. Use relevant literature to comment on the findings and limitations of the study.

Project 10: Reddit data Link Prediction

Consider the Reddit Threads dataset, which contains 10000 Discussion and Non-Discussion based threads from Reddit collected in May 2018. Nodes are Reddit users who participate in a discussion (thread) and links (edges) are replies between them. For each thread, a binary annotation 0 or 1 is provided to indicate whether the thread is discussion or not (this is provided in target file). The dataset is available in GitHub - benedekrozemberczki/karateclub: Karate Club: An API Oriented Open-source Python Framework for Unsupervised Learning on Graphs (CIKM 2020). We want to comprehend both the structure of the network and the attributes (discussion versus non-discussion).

1. Consider the thread presenting the largest and smallest network among all threads. Use label propagation algorithm to determine the various communities of each case. Use appropriate visualization tools to plot these communities. Plot the degree distributions of each case as well using appropriate NetworkX functions.
2. We want to comprehend some key characteristics of the network at each thread. Suggest a script using NetworkX functions that outputs the size, diameter, highest / smallest / average clustering coefficient, highest / minimum / average degree centrality, highest / minimum / average closeness centrality. Save the results in an excel file.
3. We want to evaluate the correlation between the size of the network and each of the following attribute: diameter, average degree centrality, average closeness centrality, average clustering coefficient. In case where more there are more one threads whose networks present the same network size, then the average of the attribute score should be taken (calculate also the related standard deviation). Show in a single plot, the variation of these attributes with result to network size. Summarize in a table the result of the Pearson correlation and the corresponding p-value. Conclude on the relationship between the network size and the corresponding attributes.
4. Now we want to distinguish the Discussion and Non-Discussion threads. For this purpose, repeat 3) for Discussion related threads and Non-Discussion related threads. Conclude on the relationship between the network size and each of the other attributes.
5. Now we want to study the prediction capability (whether a thread is Discussion based or Non-discussion based). For this purpose, use a random selection of 80% of total threads for training and the other 20% for testing (Repeat the process 10 times to make different selection and then take average). We want to use the degree distribution as a feature vector. For this purpose, for each thread, generate a degree distribution. The latter should be represented as a fixed size vector. Use a machine learning model of your choice (You may use one of libraries where a comparison between various machine-learning models is presented, so you present the result of the top performing models) and test the accuracy, Area-Under-Curve (AUC) scores.
6. Suggest alternative feature vector representation for 5) and test the corresponding prediction accuracy and AUC scores.
7. Use the graph embedding methods available in Karate Club library that generate embedding vector for the whole network. Use the embedding vector as a feature vector and suggest a machine learning model for carrying out the classification task. Present the results of the various embedding methods in a table.
8. Identify relevant literature (you may rely on the up-to-date literature of Karate Club) to comment on the findings.

Project 11. Mining Emergency 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 emergency (should be broad enough to include all aspects, i.e., health emergency, home emergency, accident related emergency, work related emergency). Elaborate your own methodology to identify a large scale emergency 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 emergency 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 emergency 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 12: 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 13. 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).

Identify relevant literature in fake new identification and health literature to back up your finding in previous sections.

Project 14: Fake News

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 due to 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, number of retweets, 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. We want to check the spread of information in both fake news and real news dataset. For this purpose, extract date attributes for the retweets and compare the timely evolutions of retweets in both real and fake news dataset. Conclude about the comparison between the two scenarios.
8. Use relevant literature from fakes news detection from social media to discuss your finding at each level of the preceding reasoning.

Project 15. 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 16. 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 17. Analysis of Climate Change Community.

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

Use Twitter API to collect at least 2000 tweets related to hashtags *#globalwarming, #climatechange, #agw* (an acronym for "anthropogenic global warming"), *#climate*and*#climaterealists*, #climatestrikeonline, 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. 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 hashtag.
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. 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.
5. 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 identify hashtags in tweet messages and 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, 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.
7. Plot the degree distribution and local clustering coefficient distribution.
8. Use label propagation algorithm in NetworkX to find communities in the above network. Compare the size of the generated communities and their associated diameter and clustering coefficient in a table. Using your understanding of the name of the hashtag, speculate whether each community can be assigned some interpretation.
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.
11. Comment the results of the previous steps using some literature from climate change in order to reinforce your argumentation.

Project 18. Water Research Citation Network Analysis

We would like to explore the citation analysis in "Water Resources Research" journal community – see https://agupubs.onlinelibrary.wiley.com/journal/19447973 –

1. Construct a small database containing the list of papers published in the journal in the last five years. You are free to suggest your own approach to collect the data (either automated through API, if any, semi-automated). The database should contain titles of the papers, author names and their affiliations, and keywords used in the journal metadata and list of references for each paper. For instance, Scopus API, SciFinder API provide you with hints to gather articles biometric data (although not required). Save the collected database in a csv format.
2. Perform a simple histogram construction that allows you to rank the authors in terms of number of publications in the journal and number of collaborators (co-authors).
3. Perform another histogram that allows you to rank the institutions that publish most papers in the journal. Next, 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 paper titles and their reference list. We consider there is a link from paper 1 to paper 2 if in the list of references of paper 1 contains paper 2. Design a program that allows you to implement the above strategy. Provide in a table the global properties of this graph in terms of number of nodes / edges, diameter, clustering coefficient, number of components, average / standard deviation degree using appropriate functions in NetworkX.
5. Use the concept of hubs and authorities as implemented in hits algorithm of NetworkX to calculate the page rank of each paper. Provide a histogram ranking the hubs in the article database
6. We would like to explore the network of authors in similar way as Erdös construction. From 2), assume the authors who has the largest number of collaborators as a reference Erdös.
7. Design and implement a program that would allow you to calculate the newly established Erdös number of each author.
8. Visualize the graph of authors with Erdös number 1 and 2.
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).
10. We want to investigate the network of institutions. For this purpose, write down a code that allows you to focus on the affiliations only part, and construct a graph where the nodes represent the author's affiliation and an edge between two nodes indicate that the underlined two institutions co-authored the same paper. Provide global properties of this graph as in 4). Plot the degree distribution of this graph and check whether a power-law distribution can be fit. Comment on your finding using relevant literature.
11. Identify relevant literature to comment and discuss the findings and pointing potential limitations of the study.