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

Project 1. NewsPopularity Dataset Analysis

Consider the News popularity dataset available in Kaggle https://www.kaggle.com/datasets/dilwong/newspopularity. The dataset contains Twitter retweets, replies, likes counts, timestamp and bag_of_phrases, for 9100 New York Times articles from January 2022 to mid-April 2022.

Graph construction and Analysis

1. To construct a network graph from the above graph, we shall consider the content of the bag_of_phrases attribute so that Tweets referring to similar topics are connected via a network graph. For this purpose, study the example in Spacy For NER(Named Entity Recognition) | by Vedant | Medium for identification of named-entities from the all tokens in bag_of_phrases attribute (Focus only on named, organization and location entities). Suggest a script that plots the histogram of location-named entities, another histogram for person-named entities, and another one for organization named-entities. Assume that two Tweet IDs (represented as nodes) are connected if they share one named-entity (location, person or organization named-entity). Write a script that implements this approach and plot the corresponding network using NetworkX or another software of your choice.
2. Use NetworkX to draw and plot the degree distribution of the above graph and highlight in a table the top 10 nodes of highest degree score and their bag_of_phrases content in terms of named-entities.
3. Use NetworkX to calculate the betweenness degree and closeness centrality of each node and draw the associated betweenness degree distribution and closeness degree distribution. Save the result of the centrality scores in a file. Comment on the bag_of_phrases content of nodes with highest scores for betweenness and closeness centrality.
4. Use NetworkX clustering function to compute the clustering coefficient of each node and save the result in the same file. 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.

Graph attributes analysis

5. We want to test the extent to which some attributes correlate with other attributes of the network. For this purpose, initially, for attributes Retweet_count, Reply_count and Like_count, write a script that scrutinizes the 10 most frequent tokens in bag_of_sentences and named-entities for each case. Save the result in a file. Report in a table for the five highest score and five lowest scores in each attributes, the most frequent tokens and named-entity. Conclude whether some topics, tokens trigger popularity in terms of retweets, likes and reply. You may inspire from program available in NLP visualizations for clear, immediate insights into text data and outputs | by JP Hwang | Plotly | Medium.
6. Write a program that uses label propagation algorithm in NetworkX to identify the various communities in the graph. Suggest appropriate visualization to highlight these communities. Write a program that concatenates all bag_of_sentences tokens of all nodes belonging to the same community and then use wordcloud (see Python Word Clouds Tutorial: How to Create a Word Cloud | DataCamp) to plot the wordcloud of the three largest communities in the graph.

7. Write a script that computes Pearson correlation and the associated p-value to evaluate the correlation between Retweet_count and Like_count, between Reply_count and Like_count.
8. Write a script that computes Pearson correlation and the associated p-value to evaluate the correlation between retweent_count and Number of tokens in bag_of_phrases; between Reply_count and Number of tokens in bag_of_phrases; between Like_count and Number of tokens in bag_of_phrases.

Simulation and randomness

9. Consider the node with the highest degree score playing the role of *Erdos* in Erdos number network. Write a script that calculates the Erdos number of each other node in the network and save the result in a file. Then draw the distribution of Erdos numbers of the underlined network.
10. Suggest appropriate literature to comment on the various results and findings.

Project 2. Book Character Network Analysis

Consider "Les Miserables" dataset available in Network data (umich.edu) consisting of network coappearance of characters that occur in the same chapter of Victor Hugo's novel "Les miserables" and where the weights of the edges correspond to the number of such coappearances. Download the network data from the above link (Need to identify way to handle gml file extension)

1. Use NetworkX to display the corresponding network, suggest appropriate simple labelling of the nodes to maintain the readability of the network graph as clear as possible. Save the adjacency matrix of this graph in a separate file.
2. Write a script that uses NetworkX functions to calculate the diameter, global clustering coefficient, average distance in the graph, smallest and largest component.
3. Suggest a script that uses NetworkX functions to identify the nodes (characters) of the three highest degree centrality, three highest closeness centrality and three highest betweenness centrality.
4. Write a script that plots the degree centrality distribution, closeness centrality distribution and betweenness centrality distribution.
5. We want to test the extent to which the centrality distributions in 3) fit a power law distribution. You may inspire from the implementation in powerlaw · PyPI of the power-law distribution, or can use alternative one of your choice. It is important to quantify the goodness of fit using p-value. Typically, when p-value is greater than 10%, we can state that power-law is a plausible fit to the (distribution) data.
6. We want to use exponentially truncated power-law instead of power law distribution. Suggest a script that quantifies the goodness of fit for degree-centrality, closeness centrality and betweenness centrality distributions.
7. We want to identify relevant communities from the network graph. For this purpose, use Louvain algorithm implementation in NetworkX to identify the main communities. Write a script that uses different color for each community and visualize the above graph with the detected communities. Use the appropriate function in NetworkX to compute the separation among the various communities and any other related quality measures. Comment on the quality of the partition by taking into account your own knowledge of Victor Hugo's book (you may download the book for a quick reading to comment on the role of the various characters).
8. We want to ignore the weighting imposed by the network, and we want to restrict to binary representation (either weight value 1 if there is cooccurrence of characters, zero otherwise). Provide the new adjacency matrix and save it in a file. Repeat questions 2-7) using the new adjacency matrix, and comment on the differences between the two cases (weighted and unweighted graph) accordingly.
9. We want to approximate the real graph in 1) by a random graph, by looking into the order of magnitude of average clustering coefficient and diameter, comment whether Erdös-Renyi random graph or small-world model is more suitable to approximate this real graph. Write a script that calculates the average clustering coefficient and diameter for various

values of probabilities p from 0.1 till 0.9, and identify the value of p that best matches with average clustering coefficient and diameter of the real graph.

10. Suggest appropriate literature to comment on the various findings and explore the limitation of the reasoning pipeline.

Project 3. Political Blogs Analysis

Consider "Political Blogs" dataset available in Network data (umich.edu) in gml format or Netzschleuder: the network catalogue, repository and centrifuge (skewed.de) in excel or graphML format. The dataset consists of a directed network of hyperlinks between weblogs of US politics at 2004 US election where node values are labelled as 0 for left or liberal and 1 for right or conservatives.

1. Use NetworkX to display the corresponding network, suggest appropriate simple labelling of the nodes to maintain the readability of the network graph as clear as possible. Save the adjacency matrix of this graph in a separate file.
2. Write a script that uses NetworkX functions to calculate the diameter, global clustering coefficient, average distance in the graph, smallest and largest component.
3. Suggest a script that uses NetworkX functions to identify the nodes of the three highest degree centrality, three highest closeness centrality and three highest betweenness centrality.
4. Write a script that plots the degree centrality distribution, closeness centrality distribution and betweenness centrality distribution.
5. We want to test the extent to which the centrality distributions in 3) fit a power law distribution. You may inspire from the implementation in powerlaw · PyPI of the power-law distribution, or can use alternative one of your choice. It is important to quantify the goodness of fit using p-value. Typically, when p-value is greater than 10%, we can state that power-law is a plausible fit to the (distribution) data.
6. We want to use exponentially truncated power-law instead of power law distribution. Suggest a script that quantifies the goodness of fit for degree-centrality, closeness centrality and betweenness centrality distributions.
7. We want to identify relevant communities from the network graph. For this purpose, use Louvain algorithm implementation in NetworkX to identify the main communities. Write a script that uses different color for each community and visualize the above graph with the detected communities. Use the appropriate function in NetworkX to compute the separation among the various communities and any other related quality measures. Comment on the quality of the partition by taking into account the available knowledge about the node attributes (0 and 1 values depending whether it accommodates liberal or conservative view).
8. We want to analyze the network in terms of type of cascades available in the network. Write a script that identifies the number of starts and number of chains (see definition in Handout 5) attached to liberal and conservative nodes.
9. We want to quantify the density of conservative and liberal node topology. For this purpose, write a script that calculates, for each node X, the proportion P(X) of neighbors that have the same political affiliation as X. Compute the average score of P(X) for all liberal nodes and average score for all conservative nodes. Comment on the relationship among liberal and conservative nodes.
10. Suggest appropriate literature to comment on the various findings and explore the limitation of the reasoning pipeline.

Project 4. Word Ajacencies Network Analysis

Consider "Word Adjacencies" dataset available in Network data (umich.edu) in gml format or casos.cs.cmu.edu/tools/datasets/external/DavidCopperfield/ in excel format. The dataset consists of undirected network of common noun and adjective adjacent for the novel David Coperfield by English 19th century writer Charles Dickens, where a node represents either a noun or an adjective and an edge connects two words that occur in adjacent position.

1. Use NetworkX to display the corresponding network, suggest appropriate simple labelling of the nodes to maintain the readability of the network graph as clear as possible. Save the adjacency matrix of this graph in a separate file.
2. Write a script to show whether the graph is bipartie graph or not.
3. Suggest a script that uses NetworkX functions to identify the nodes of the three highest degree centrality, three highest closeness centrality and three highest betweenness centrality.
4. Write a script that plots the degree centrality distribution, closeness centrality distribution and betweenness centrality distribution. Also, write a script to plot the cumulative degree distribution and the clustering coefficient distribution.
5. We want to test the extent to which the centrality distributions in 4) fit a power law distribution. You may inspire from the implementation in powerlaw · PyPI of the power-law distribution, or can use alternative one of your choice. It is important to quantify the goodness of fit using p-value. Typically, when p-value is greater than 10%, we can state that power-law is a plausible fit to the (distribution) data.
6. We want to use exponentially truncated power-law instead of power law distribution. Suggest a script that quantifies the goodness of fit for degree-centrality, closeness centrality and betweenness centrality distributions.
7. We want to identify relevant communities from the network graph. For this purpose, use Label propagation algorithm implementation in NetworkX to identify the main communities. Write a script that uses different color for each community and visualize the above graph with the detected communities. Use the appropriate function in NetworkX to compute the separation among the various communities and any other related quality measures. Comment on the quality of the partition by taking into account the available knowledge about the node attributes (0 and 1 values depending whether it accommodates noun or adjectives).
8. We want to analyze the network in terms of type of cascades available in the network. Write a script that identifies the number of starts and number of chains (see definition in Handout 5) attached to liberal and conservative nodes.
9. We want to quantify the density of adjective and noun node topology. For this purpose, write a script that calculates, for each node X, the proportion P(X) of neighbors that have the same affiliation as X. Compute the average score of P(X) for all noun nodes and average score for all adjective nodes. Comment on the relationship among noun and adjective nodes.
10. Suggest appropriate literature to comment on the various findings and explore the limitation of the reasoning pipeline.

Project 5. Coauthorship Network Analysis

Consider "Coauthorships in network science" dataset available in Network data (umich.edu) in gml format (maybe found elsewhere in other format as well). The dataset consists of undirected network of coauthorship scientists working on network theory and experiment, compiled from bibliographies of two review articles in Siam review and Physics report journals, where a node represents an author and a link a joint publication between the two corresponding authors.

1. Use NetworkX to display the corresponding network, suggest appropriate simple labelling of the nodes to maintain the readability of the network graph as clear as possible. Save the adjacency matrix of this graph in a separate file.
2. Write a script that plots the degree centrality distribution, closeness centrality distribution and betweenness centrality distribution.
3. We want to test the extent to which the centrality distributions in 2) fit a power law distribution. You may inspire from the implementation in powerlaw · PyPI of the power-law distribution, or can use alternative one of your choice. It is important to quantify the goodness of fit using p-value. Typically, when p-value is greater than 10%, we can state that power-law is a plausible fit to the (distribution) data.
4. We want to use exponentially truncated power-law instead of power law distribution. Suggest a script that quantifies the goodness of fit for degree-centrality, closeness centrality and betweenness centrality distributions.
5. Write a script that identifies the largest component, second largest, third largest and smallest component (s) (of at least two nodes). Display each component (If there is more than one component for a given case, then draw one at random). Then display a vector X indicating the size of the first, second and third largest component of the graph.
6. Write a script that calculates the shortest distance between any pair of the components in 2). Present the result in a table. Comment on the separability between the above components.
7. We want to identify relevant communities from the subnetwork graph corresponding to the largest, second largest and third largest component. For this purpose, use Louvain algorithm implementation in NetworkX to identify the main communities. Write a script that uses different color for each community and visualize the above graph with the detected communities. Use the appropriate function in NetworkX to compute the separation among the various communities and any other related quality measures. Comment on the quality of the partition.
8. Write a script that generates an Erdös-Renyi random graph with n= total number of nodes of the network, and repeat the process of various probabilities p from 0.1 to 0.9 so that for each realization p, one calculates the vector X (size of largest, second largest and third largest component). Identify the value of probability p that best matches the value of X in 5).
9. Repeat 8) when using Small world-model.
10. Suggest appropriate literature to comment on the various findings and explore the limitation of the reasoning pipeline.

Project 6. Twitter Friendship Network Analysis

Consider the large scale Twitter friendship dataset available at Twitter Friends (kaggle.com) of full Twitter user profile data (40K users), including friendship relationship. We want to explore this friendship relation to construct a network graph where User IDs are nodes and a directed edge from IDx to IDy if IDy is listed as a friend of IDx.

1. Use NetworkX to display the corresponding network, suggest an approach to visualize this dense network using visualization tool of your choice (NetworkX is not ideal for dense graphs). Save the adjacency matrix of this graph in a separate file.
2. Write a script that uses NetworkX functions to calculate diameter, average clustering coefficient and average path length of the network.
3. Write a script that plots the degree centrality distribution and closeness centrality distribution.
4. We want to test the extent to which the centrality distributions in 3) fit a power law distribution. You may inspire from the implementation in powerlaw · PyPI of the power-law distribution, or can use alternative one of your choice. It is important to quantify the goodness of fit using p-value. Typically, when p-value is greater than 10%, we can state that power-law is a plausible fit to the (distribution) data.
5. Write a script that calculates the number of triangles in the network.
6. Write a script that identifies the largest strongly connected component, second largest, third largest. Display each component (If there is more than one component for a given case, then draw one at random).
7. Write a script that calculates the shortest distance between any pair of the strongly connected components in 5). Present the result in a table. Comment on the separability between the above components.
8. We want to identify relevant communities from the subnetwork graph corresponding to the largest, second largest and third largest component. For this purpose, use Label propagation algorithm implementation in NetworkX to identify the main communities. Write a script that uses different color for each community and visualize the above graph with the detected communities. Use the appropriate function in NetworkX to compute the separation among the various communities and any other related quality measures. Comment on the quality of the partition.
9. We want to evaluate the evolution of the triangles (transitivity relation in the network). For this purpose, we consider time increment as an accumulation of one thousand successive rows in the original dataset. Suggest a script that calculates the evolution of the proportion of triangles (number of triangles over the total number of nodes up to time t, basically t=1, 2,…,40), and draws the corresponding graph.
10. From 9), write a script that identifies instances of triplet nodes A, B, C such at time t A is connected to B and B is connected to C but A is not connected to C, while in time t+1, A becomes connected to C. Check for theses instances whether the link prediction using

common neighbor (probability A being connected to C increases with the number of common neighbours between A and B). Write the result in a table.

11. Suggest appropriate literature to comment on the various findings and explore the limitation of the reasoning pipeline.

Project 7. Twitter COVID19 Network Analysis 1

Consider the TweetsCOV19 dataset, a semantically annotated corpus of Twitter about COVID19 aiming at capturing online discourse about various aspects of the pandemic and its social impact. The dataset is available from TweetsCOV19 - A Semantically Annotated Corpus of Tweets About the COVID-19 Pandemic (Part 1, October 2019 - April 2020) (zenodo.org). The dataset includes attributes related to Twitter Id, timestamp, sentiment, mention, hashtag, entities, among others.

1. Use the timestamp attribute to restrict the collection to Twitter data in the period October 2019-December 2019.
2. We want to construct a network according to the associated hashtags, where nodes correspond to Twitter Ids, and an edge between two nodes is established if the corresponding twitter ids share the same hashtag in their hashtag attributes. Write a script that implements this graph construction. What is the size of the graph in terms of number of nodes and edges. Save the adjacency matrix of this graph in
3. Write a script that uses NetworkX to identify the largest component, second largest and third largest component of the network as well as the average path length.
4. Write a script that plots the degree centrality distribution and cumulative degree centrality distribution.
5. We want to test the extent to which the degree centrality distributions in 3) fit a power law distribution. You may inspire from the implementation in powerlaw · PyPI of the power-law distribution, or can use alternative one of your choice. It is important to quantify the goodness of fit using p-value. Typically, when p-value is greater than 10%, we can state that power-law is a plausible fit to the (distribution) data.
6. We want to capitalize on the information about the sentiment of tweet. Write a script that calculates the overal sentiment of each tweet by adding the positive and negative sentiment score available in the sentiment attribute.
7. Now we want to evaluate the extent to which positive (resp. negative) sentiment tweet connects with positive (resp. negative) sentiment tweet. Suggest a script that traverses all edges and calculates the proportion of positive-positive connection, negative-negative connection and positive-negative or negative-positive connection.
8. Now we want to take into account the time evolution. For this purpose, use the timestamp of the tweets. Suggest a script that creates some time subdivision (by assigning a fixed number of subdivisions between the largest and smallest recorded time of the tweets) and calculates the evolution of the total number of positive sentiment tweets after each time increment (subdivision). Suggest a parametric distribution (i.e., polynomial or exponential) that best fits the obtained graph (using curve fitting method of your choice).
9. Repeat 8) when using negative sentiment tweets.
10. We want to assimilate negative sentiment propgation to a virus propagation case. Use the NDLIB library to simulate an SIS propagation model where the overall time is given by the number of subdivisions in 8), and the graph is set by the graph configuration at time increment 1 (after first subdivision). Select several choices for parameters lambda and beta to generate contamination (total number of negative sentiments) close to that observed in 9) (although no guarantee that this can be achieved).

11. Suggest appropriate litearture to support the findings and comment on the limitations of the overal reasoning pipeline.

Project 8. Twitter COVID19 Network Analysis II

Consider the TweetsCOV19 dataset, a semantically annotated corpus of Twitter about COVID19 aiming at capturing online discourse about various aspects of the pandemic and its social impact. The dataset is available from TweetsCOV19 - A Semantically Annotated Corpus of Tweets About the COVID-19 Pandemic (Part 1, October 2019 - April 2020) (zenodo.org). The dataset includes attributes related to Twitter Id, timestamp, sentiment, mention, hashtag, entities, among others.

1. Use the timestamp attribute to restrict the collection to Twitter data in the period December 2019-February 2020.
2. We want to construct a network according to the associated hashtags, where nodes correspond to Twitter Ids, and an edge between two nodes is established if the corresponding twitter ids share the same hashtag in their hashtag attributes. Write a script that implements this graph construction. What is the size of the graph in terms of number of nodes and edges. Save the adjacency matrix of this graph in
3. Write a script that uses NetworkX to identify the largest component, second largest and third largest component of the network as well as the average path length.
4. Write a script that plots the degree centrality distribution and cumulative degree centrality distribution.
5. We want to test the extent to which the degree centrality distributions in 3) fit a power law distribution. You may inspire from the implementation in powerlaw · PyPI of the power-law distribution, or can use alternative one of your choice. It is important to quantify the goodness of fit using p-value. Typically, when p-value is greater than 10%, we can state that power-law is a plausible fit to the (distribution) data.
6. We want to capitalize on the information about the sentiment of tweet, which contains both positive and negative score. Create a subdivision of 100 bins in the time domain taking into account the smallest and largest timestamp of tweet data. Write a script that counts the total average positive sentiment scores and average negative sentiment scores of all tweets available up to the given time subdivision. Then draw a diagram that shows the timely evelution of the average positive sentiment score and that of average negative sentiment score. Comment on the discrepancy between the two sentiment scores.
7. We want to evaluate the transitivity relations in the network. For this purpose, write a script that estimates the number of triangles available up to each time subdivision. Draw a diagram that displays this timely evolution of number of triangles.
8. We want to evaluate the extent to which the triangles are balanced. For this purpose, calculate the overal sentiment for each tweet by summing up the positive and negative sentiment to find out whether the overall score is positive or negative. A triangle is assumed balanced if the product of the overall sentiment of its individual tweets is positive, otherwise, if the product is negative, it is unbalanced. Suggest a script that implements this approach. Then write a script that calculates the proportion of balanced triangles and unbalanced triangles available up to each time increment.
9. We want to assimilate the occurrence of unbalanced triangles as a virus propagation case. Use the NDLIB library to simulate an SIS propagation model where the overall time is given by the number of subdivisions in 6), and the graph is set by the graph configuration at time

increment 1 (after first subdivision). Select several choices for parameters lambda and beta to generate contamination (total number of negative sentiments) close to that observed in 8) (although no guarantee that this can be achieved).

10. Suggest appropriate litearture to support the findings and comment on the limitations of the overal reasoning pipeline.

Project 9. Twitter COVID19 Network Analysis III

Consider the TweetsCOV19 dataset, a semantically annotated corpus of Twitter about COVID19 aiming at capturing online discourse about various aspects of the pandemic and its social impact. The dataset is available from TweetsCOV19 - A Semantically Annotated Corpus of Tweets About the COVID-19 Pandemic (Part 1, October 2019 - April 2020) (zenodo.org). The dataset includes attributes related to Twitter Id, timestamp, sentiment, mention, hashtag, entities, among others.

1. Use the timestamp attribute to restrict the collection to Twitter data in the period January 2020-March 2020.
2. We want to construct a network according to the associated mention, where nodes correspond to Twitter Ids, and an edge between two nodes is established if the corresponding twitter ids share the same mention in their Mention string attributes. Write a script that implements this graph construction. What is the size of the graph in terms of number of nodes and edges. Save the adjacency matrix of this graph in
3. Write a script that uses NetworkX to identify the largest component, second largest and third largest component of the network as well as the average path length.
4. Write a script that plots the degree centrality distribution and cumulative degree centrality distribution.
5. We want to test the extent to which the degree centrality distributions in 3) fit a power law distribution. You may inspire from the implementation in powerlaw · PyPI of the power-law distribution, or can use alternative one of your choice. It is important to quantify the goodness of fit using p-value. Typically, when p-value is greater than 10%, we can state that power-law is a plausible fit to the (distribution) data.
6. We want to capitalize on the information about the sentiment of tweet. Write a script that calculates the overal sentiment of each tweet by adding the positive and negative sentiment score available in the sentiment attribute.
7. Now we want to evaluate the extent to which positive (resp. negative) sentiment tweet connects with positive (resp. negative) sentiment tweet. Suggest a script that traverses all edges and calculates the proportion of positive-positive connection, negative-negative connection and positive-negative or negative-positive connection.
8. Now we want to take into account the time evolution. For this purpose, use the timestamp of the tweets. Suggest a script that creates some time subdivision (by assigning a fixed number of subdivisions between the largest and smallest recorded time of the tweets) and calculates the evolution of the total number of positive sentiment tweets after each time increment (subdivision). Suggest a parametric distribution (i.e., polynomial or exponential) that best fits the obtained graph (using curve fitting method of your choice).
9. Repeat 8) when using negative sentiment tweets.
10. We want to assimilate negative sentiment propgation to a virus propagation case. Use the NDLIB library to simulate an SIS propagation model where the overall time is given by the number of subdivisions in 8), and the graph is set by the graph configuration at time increment 1 (after first subdivision). Select several choices for parameters lambda and beta to generate contamination (total number of negative sentiments) close to that observed in 9) (although no guarantee that this can be achieved). Draw the simulation showing the number

of contamination (number of tweets with negative sentiment) over time, and compare this with that obtained in 8).

11. Suggest appropriate litearture to support the findings and comment on the limitations of the overal reasoning pipeline.

Project 10. Signed Graph Network Analysis

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. 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 11. eBook Poem Analysis **[Support up to two projects on the topic]**

Consider the Gutenberg collection of ebooks available at http://www.gutenberg.org/. You may notice that for each ebook, you can freely download the full text version of the book. Choose one poetry ebook of your choice. We want to investigate the structure of noun-adjective relationship occurring in the ebook.

1. We want to identify most popular nouns and adjectives in the poem. For this purpose suggest a script that identify the 100 most frequent nouns and 100 most frequent adjectives in the poem. You may inspire from examples in Natural Language Processing With Python's NLTK Package – Real Python, where you target only part-of-speech tagging with a focus on adjective category (JJ) and noun category (NN). Save the set N of 100 popular nouns and the set A of 100 popular adjectives in a file.
2. We want to establish a network by considering nouns of N and adjectives of A as nodes and a link between two nodes is established whenever the two nodes occur in the same line of the poem. Write a script that allows you to construct this network. Save its adjacency matrix in a file and display the corresponding graph using NetworkX or alternative visualization tool of your choice, while putting different colors for noun and adjective nodes.
3. Write a script that allows you to check whether graph in 2) is bigraph, and output the diameter, average path length, largest length, smallest length, average clustering coefficient, smallest and largest clustering coefficient, smallest/largest and average degree.
4. Write a script that identifies the largest component, second largest and third largest component of the network, and the content of each component in terms of number of Noun nodes and number of Adjective nodes. Comment on the balance between the two node attributes.
5. Write a script that plot the degree centrality distribution, closeness centrality distribution and betweeness centrality distribution.
6. Use the reasoning pointed out in Project 1 to find out whether a power law distribution can be fit to degree centrality distribution, closeness centrality distribution and betweeness centrality distribution.
7. Write a script that draws the clustering coefficient distribution after performing a 10 bins subdivision on the clustering coefficient values. Find out whether a power law distribution or exponential truncated power-law distribution can be fit to this distribution.
8. Write a script that uses Louvain algorithm for performing community detection. Comment on the community content in terms of mixture between noun and adjective attributes.
9. We want to quantify the evolution of the node relationship across various lines of the poem. For this case, consider the three relationship: Noun-Noun, Noun-Adjective (or Adjective-Noun), and Adjective-Adjective (only for nouns and adjectives in sets N and A). Write a script that displays the evolution of the number of each relationship with respect to the number of lines of the poem.
10. We want to reconsider the construction of the network of 2) by adopting a weighted graph-based approach where the weights of the edges correspond to the number of lines of the poem, which contain the underlining type of entities (noun or adjective). Provide the new adjacency matrix of the graph. Then suggest a script that identifies the Steiner Tree

corresponding to the given graph. You may use existing Steiner Tree graph approximation implementations of your choice, i.e., steiner-tree-problem · GitHub Topics · GitHub..

11. Identify relevant literature to comment on the findings and inherent limitations of the reasoning pipeline.

Project 12. eBook Novel Analysis **[Support up to two projects on the topic]**

Consider the Gutenberg collection of ebooks available at http://www.gutenberg.org/. You may notice that for each ebook, you can freely download the full text version of the book. Choose one Novel ebook of your choice. We want to investigate the structure of the characters appearing in the ebook.

1. We want to identify all characters of the Novel. For this purpose, you may decide the best approach to proceed, either using a manual or an automated based approach. The manual approach consists in reading the book yourself to find out who are the main characters of the book. The second approach consists in using an automated text mining like approach where you may use script that identifies named-entities from the text and restrict to Person-named-entities. You may use spacy named entities (see for instance an example in Building Your Own Custom Named Entity Recognition (NER) Model with spaCy V3: A Step-by-Step Guide | by Mayur Ghadge | Medium).
2. Next, we want to construct a network where nodes are the characters identified in 1) and en edge between two nodes is established whenever the two corresponding characters occurred in the same sentence of the novel, where the weight of this edge is provided by the number of sentences of such co-occurrence. Write a script that allows you to construct this network. You may use sentence tokenizer in python NLTK library and normal text matching to find out the occurrence of characters in the same sentence. You may inspire from examples at NLTK Tokenize: How to Tokenize Words and Sentences with NLTK? - Holistic SEO. Provide the key features of this network in terms of number nodes, edges. Save the the adjacency matrix of the network in a file.
3. Write a script that uses NetworkX functions to calculate the diameter, global clustering coefficient, average distance in the graph, smallest and largest component.
4. Suggest a script that uses NetworkX functions to identify the nodes (characters) of the three highest degree centrality, three highest closeness centrality and three highest betweenness centrality.
5. Write a script that plots the degree centrality distribution, closeness centrality distribution and betweenness centrality distribution.
6. We want to test the extent to which the centrality distributions in 3) fit a power law distribution. You may inspire from the implementation in powerlaw · PyPI of the power-law distribution, or can use alternative one of your choice. It is important to quantify the goodness of fit using p-value. Typically, when p-value is greater than 10%, we can state that power-law is a plausible fit to the (distribution) data.
7. We want to use exponentially truncated power-law instead of power law distribution. Suggest a script that quantifies the goodness of fit for degree-centrality, closeness centrality and betweenness centrality distributions.
8. We want to identify relevant communities from the network graph. For this purpose, use Louvain algorithm implementation in NetworkX to identify the main communities. Write a script that uses different color for each community and visualize the above graph with the detected communities. Use the appropriate function in NetworkX to compute the separation

among the various communities and any other related quality measures. Comment on the quality of the partition by taking into account your own knowledge of the Novel book.

9. We want to ignore the weighting imposed by the network, and we want to restrict to binary representation (either weight value 1 if there is cooccurrence of characters, zero otherwise). Provide the new adjacency matrix and save it in a file. Repeat questions 3-8) using the new adjacency matrix, and comment on the differences between the two cases (weighted and unweighted graph) accordingly.
10. We want to approximate the real graph in 1) by a random graph, by looking into the order of magnitude of average clustering coefficient and diameter, comment whether Erdös-Renyi random graph or small-world model is more suitable to approximate this real graph. Write a script that calculates the average clustering coefficient and diameter for various values of probabilities p from 0.1 till 0.9, and identify the value of p that best matches with average clustering coefficient and diameter of the real graph.
11. Suggest appropriate literature to comment on the various findings and explore the limitation of the reasoning pipeline.

Project 13. Paper citation network II

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 "Text Summarization Evaluation".

1. First, collect data related to automatic text summarization in arXiv API using keywords like "Automatic text summarization AND evaluation", "text summarization AND evaluation", "document summarization AND evaluation" (feel free to extend the list with similar wording) from 2015 onwards. This generates a CSV collection of files containing title, date, article_id, url, main topic, all topics, authors, year. Collect the first 200 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 Label propagation 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 14. Paper citation network III

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 "deep learning models for hydrology modelling".

1. First, collect data related to the above topic in arXiv API using keywords like "deep learning AND hydrology modelling", "water cycle AND deep learning", "deep learning AND Water quality monitoring" (feel free to extend the list with similar wording) from 2018 onwards. This generates a CSV collection of files containing title, date, article_id, url, main topic, all topics, authors, year. Collect the first 200 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 15: Food.com Recipes and Interactions (Proposed by Mehrdad Rostami)

Consider the *Food.com* dataset available Food.com Recipes and Interactions, which consists of 180K~ recipes. In this project we just use **PP_users.csv** and **PP_recipes.csv**.

1- Load the **PP_users.csv** dataset and generate the friendship network among users with the following assumption: If two users have at least one shared item, we consider them as friends.
2- Utilize NetworkX to visualize and plot the degree distribution of the given graph. Separate the degree centrality distribution, closeness distribution, and the betweenness distribution. Draw three distinct degree distributions accordingly.
3- Provide the script for drawing power law distributions for degree, closeness, and the betweenness distributions.
4- Utilize the NetworkX clustering function to calculate the clustering coefficient for each node within the graph. Then, generate a histogram with 10 bins based on the clustering coefficient values, displaying the count of nodes within each bin.
5- Utilize NetworkX functions to detect communities using the Girvan-Newman and Louvain community detection algorithm. Employ suitable plotting techniques to emphasize the distinct communities within the graph. Present a table summarizing key characteristics of each community, including the number of nodes, number of edges, diameter, average path length, and average degree.
6- Try to identify the number of shared items within each user community and subsequently create a histogram illustrating the number of shared items within each community. Conclude by determining if any discernible trends emerge from the data analysis.
7- Try to identify the ten most popular shared items within each community, and then analyze the characteristics of these popular items. This analysis should include attributes such as calorie levels, number of ingredients, and number of steps, utilizing data from the PP_recipes.csv dataset.
8- Instead of generating a bipolar friendship network, we aim to convert the bipartite graph mentioned above into a weighted social network graph representing interactions among users in a food social network. This transformation involves evaluating the number of shared items between users. The friendship level is quantified as an integer and represented through a symmetrical matrix.
9- Repeat tasks 2-7 for this new graph.
10- Suggest appropriate metrics to compare the two graphs of 1) and 8).
11- Utilize literature on centrality measures, clustering techniques, and community detection algorithms to analyze the findings outlined in the aforementioned specifications. Evaluate the limitations of the methods employed and propose potential alternative approaches for investigation.

Project 16: Last.fm online music system (Proposed by Mehrdad Rostami)

Consider the ***Last.FM*** dataset available grouplens website, which consists of 90K~ artist listening records from 2K~ users.

1- Considering the **user_friends.dat** generate the friendship network among users.
2- Write a script to calculate various statistical properties of the generated network, including: the number of edges, the magnitude of the largest component, the count and sizes of components, the network's diameter, average path length, and average clustering coefficient. Present the findings on a Table.
3- Utilize NetworkX methodologies to identify user groups within the network. Employ the Label Propagation Algorithm (LPA) and the Girvan-Newman Algorithm for this purpose. Additionally, leverage NetworkX to assess the effectiveness of these communities using performance metrics, and subsequently, compare the outcomes of both approaches.
4- Try to find rated artists and tags within each community, then analyze and compare the various approaches. Are there any shared artists or tags which overlap between all detected communities? Try to identify these shared items and tags.
5- Try to identify the top-10 popular shared artists and tags within each community. Subsequently, explore potential trends among user communities based on the popular artists or tags specific to each community.
6- Rather than creating a user graph using **user_friends.dat**, consider generating a user graph based on user similarity. Suggest a script to compute user similarity based on their interactions (common shared items/tags) found in **user_taggedartists.dat**. Once the Pearson similarity measure is computed, you can define a threshold to construct the bipartite graph. This graph will feature an edge between two nodes if their similarity exceeds the defined threshold.
7- Repeat tasks 2-6 for this new graph.
8- Suggest appropriate metrics to compare the two graphs of 1) and 6).
9- In this step, our aim is to generate an attributed network of artists. Each artist will be represented as a node, and their shared tags (from **user_taggedartists.dat**) will serve as their attributes. Next, suggest a node similarity measure suitable for this new attributed network. Subsequently, proceed with generating the graph based on this calculated node similarities.
10- Find top-10 popular nodes in this network based on three node centrality measure of Degree, Betweenness and Eigenvector.
11- In this step, we aim to calculate the popularity of each artist based on the user interaction matrix extracted from **user_taggedartists.dat**. To achieve this, we will count the number of user_artist_tag rows for each artist, considering it as a measure of popularity. Artists with more recorded rows will be deemed more popular. Write a script to draw a histogram depicting the yearly 10 popular artists and visualize the graph and compare the results of this popular artist with previously calculated popular nodes in Step 10.
12- Identify relevant literature to provide analysis and discussion on the findings, while also highlighting potential limitations of the study.

Project 17. Mining CCTV Security opinions in Suomi24

The interest focuses on mining the discussion related to the issue of being monitored through CCTV camera in public space or indoor areas (i.e., shopping areas) 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. The use of this last dataset is more recommended as you may investigate the evolution of the topic on yearly basis, but it is more challenging to large set dataset.
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 related terms (i.e., using synomymy relations or topical distribution according to some literature you identified) .
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 topic related words. 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 topic 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 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). (Wiley does not support free API but you may use scopus API, OpenAlex or other similar APIs that support open Access). 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.