Group Project ICS2205/2230
Web Intelligence

Charlie Abela, Joel Azzopardi

December 14, 2020

This document contains the details for the group ICS 2205/2230 project which is marked out of 100%, however it is equivalent to 40% of the total mark for this unit. Each team should be composed of a maximum of TWO individuals per team. You can decide with whom you want to team up. However, if you do not manage to team up yourselves, then we will form the teams, and such team setups will be final.

While discussions between individual students are considered as healthy, the final deliverable needs to be that produced by your team and not plagiarised in any way. The work within each team has to be distributed fairly, and in the documentation you will need to describe how the work was distributed and who was responsible for which part of the project. The mark given to each team member is determined based on the quality of that member's contribution to the team's overall project. Marks assigned to different members of the same team *may* vary.

The **deadline** for this project is **12:00pm Monday 15th February 2021**. Deliverables and attached plagiarism form must be uploaded on the VLE. Projects submitted late will be penalised or may not be accepted.

1 Background

Nowadays, one can find various data sets that have been released to the public with the purpose of allowing people to discover and visualise interesting nuggets of information from these datasets.

For example, on 31st August 2015, the US State Department released a considerable number of emails that were the source of controversy for Hillary Clinton since she was using a non-government email server when she was secretary of state. One can download such emails from [1]. Moreover, one may also find various demos that visualise interesting knowledge nuggets

[1] `https://www.kaggle.com/c/hillary-clinton-emails/data`

extracted from this dataset[2].

As a further example of data visualisation, one may refer to the site MovieGalaxies[3] that provides insight into the world of movies by displaying the interactions between the actors participating in that movie, as a network. The visualisation used, distinguishes between the various actors and the importance of their roles within a movie, by displaying nodes with different colours and sizes. Furthermore, various metrics are computed, such as the betweenness centrality and degree for each node, and the clustering coefficient, diameter and path length for the whole network.

2 Specifications

This project consists of **TWO** tasks, with each task being assigned 50% of the mark for this project. You are to address **BOTH** tasks. The dataset that you will use is the Enron dataset[4]. We have however filtered the dataset that now contains over 279K emails. It contains emails from about 150 users, mostly senior management of Enron, organised into folders. This dataset is being made available to you on Google drive[5]. In the following sections we provide the details for each of the two tasks.

2.1 Task 1: Text Analysis

For this deliverable, you are expected to perform text analysis over the dataset and to create a Web-based dashboard to visualise the results.

i. install a Web server on your machine to host your Web-based dashboard;

ii. implement an interactive dashboard that allows the user to view a keyword cloud. It is recommended to use a library such as D3[6] embedded within HTML, CSS for styling and JavaScript for additional dynamic functionality. However, you could also use other existing libraries. Provide a visualisation of a single level of clustering of the users. When clicking upon a cluster, the keyword cloud associated with that cluster should be visualised;

iii. To perform the text analysis you need to:

[2] https://www.kaggle.com/c/hillary-clinton-emails/scripts

[3] http://moviegalaxies.com/

[4] https://www.cs.cmu.edu/~./enron/

[5] https://drive.google.com/file/d/1vqFyzWBHLtMR5SBXTl67hEaN2MLoV79b/view?usp=sharing

[6] https://d3js.org

a. Extract the text from each email;

b. Perform lexical analyses to extract the separate words, and to fold them all to lower case;

c. Use a standard stop-word list for English to filter out the stop words;

d. Use an implementation of Porter's stemmer to reduce terms to their stems (note that you may find a ready-made implementation provided that you reference its source);

e. Calculate term weights using TF.IDF. All emails between any 2 distinct senders/recipients should be considered as a single document. In this way, the important words that characterise a particular interaction will be given high weights;

f. Represent each correspondent as a vector of features. Each term weight for the correspondent vector should be the average weight of that term across all documents in which the correspondent participated (as sender or receiver);

g. Use the highest-weighted $n\%$ of the terms for each correspondent to build the correspondent keyword-cloud. This keyword-cloud will show what concepts the correspondent generally talks about. n can be determined arbitrarily so that the keyword-cloud does not contain neither too much nor too few words;

h. Use the correspondent vectors to cluster the users using the *k-means* algorithm. The choice of k is up to you. Note that you only need to do a **single** level of clustering, that is, no hierarchies are being requested. The clusters need to be visualised using an interactive bubble chart (or equivalent), and when a cluster bubble is clicked, the keyword-cloud representing that cluster should be displayed.

iv. You need to write a short report with a **maximum length of 4 pages** that clearly describes the work done, including aspects related to design and implementation, as well as the use of any third party libraries/tools.

NOTE: You can use NLTK or other ready made tools to handle parts a) to d). However, you are expected to implement your own *TF.IDF* weighting scheme, *Cosine Similarity* measure and *k-means* clustering.

This part of the project is being allocated **50 marks**.

2.2 Task 2: Graph Analysis

This component focuses on the analysis and visualisation of the email corpus as a graph where the nodes represent the correspondents (senders/recipients)

and each edge represents the correspondence between any TWO correspondents.

You are expected to:

i. transform the provided dataset into a suitable format for analysis and visualisation. You can use NetworkX or some other library to create the graph;

ii. create a Jupyter Notebook and analyse the graph by computing the following:

   (a) *Degree distribution*: generate i) the undirected distribution of the graph G_u, ii) the in-degree distribution G_i and iii) the out-degree distribution G_o. **Create** plots for these distributions.
   The degree distribution $P(k)$ measures the probability that a randomly chosen node has degree k. For a graph G the degree distribution can be summarized using a normalized histogram. This means that the histogram is normalized by the total number of nodes. To compute the degree distribution of a graph use $P(k) = \frac{N_k}{N}$, whereby N_k is the number of nodes with degree k and N is the number of nodes. Think of this distribution as the probability that a randomly chosen node has degree k;

   (b) *Diameter*: D the longest shortest path between any two nodes in the network;

   (c) *Average path length*: P_{avg} this is the average length of the shortest paths between the nodes in the graph;

   (d) *Global Clustering coefficient*: the clustering coefficient c_c is ratio of the number of connections between the neighbours of each node and the total number of possible connections between the same neighbours. The global clustering coefficient C_c is the mean over all the individual c_c;

   (e) *Compactness*: compactness is the ratio between the number of existing edges and the number of all the possible edges $\frac{2E}{N^2-N}$, where E is the total number of edges and N is the total number of vertices in the graph. Compactness can have a value between 0 and 1;

   (f) *Betweenness centrality*: considers those nodes which are traversed in many shortest paths to be more important than others;

   (g) *PageRank*: returns a ranking over the influential nodes in the network that extends beyond their direct connections.

iii. provide a visualisation of the graph that has the following features:

(a) The correspondents' activity: the node size should be proportional to the number of emails sent/received by the corresponding sender/recipient. In this way, one can immediately recognise the most active correspondents;

(b) The level of interaction between any 2 correspondents: the edge width should be indicative of the number of emails passed between the correspondents represented by the surrounding edges.

iv. in the Jupyter notebook you need to clearly elaborate (as markdown) how you performed the analysis and discuss the findings from the distribution plots and the significance of the results of the various metrics (for instance compare compactness and the global clustering coefficient, and betweenness and PageRank).

For this task, **50 marks** are being allocated.

2.3 Summary of deliverables

Task 1: Text Analysis and Visualisation	50 marks
Task 2: Graph Analysis and Visualisation	50 marks

You will need to submit the following on the VLE:

i. The relevant files (zipped) associated with Task 1. Include also a plagiarism form, duly filled-in;

ii. The relevant files (zipped) associated with Task 2. Include also a plagiarism form, duly filled-in;

3 Final Remarks

Final suggestion: if you have difficulties do not hesitate to contact us. Any issues – including technical difficulties, or difficulties between team members – should be identified and highlighted as early as possible to ensure timely resolution.

Good luck!!