SOCIAL AND ECONOMIC NETWORK ANALYSIS

Project Report

PRIYADARSAN M (18Z239)

RAJASURIYA C K (18Z240)

ARUL JYOTHI S (18Z206)

AKILAN R (18Z204)

MOHANAPRASANTH T (19Z431)

Assignment Submission in partial fulfilment of the degree

BACHELOR OF ENGINEERING

Branch: COMPUTER SCIENCE AND ENGINEERING

Of Anna University

April 2021

PSG College of Technology

Coimbatore – 641004

1. Problem Statement:

To visualize stack overflow as a social network and do,

- Basic metrics analysis on the graph
- Find the questions which may appear together with the given question based on shortest path
- Tag Prediction

2. Dataset Description:

A dataset of Stack Overflow programming questions. For each question, it includes question ID, Creation date, closed date (if applicable), Score, number of answers, tags, title of the question and body.

This dataset is ideal for answering questions such as, the increase or decrease in questions in each tag over time and correlations among tags on questions

This dataset was extracted from the Stack Overflow database. This is all public data.

Link: https://www.kaggle.com/stackoverflow/stacklite

3. Tools Used:

- scikit-learn: Built on NumPy, SciPy and matplotlib which is very useful in predictive data analysis.
- pandas: Open source data analysis and manipulation tool built on Python.
- networkx: Python library for analysing networks and graphs.
- matplotlib: Plotting library for the Python programming language and its numerical mathematics extension NumPy.
- seaborn: Python data visualization library based on matplotlib which gives high level attractive drawings.
- nltk (Natural Language Tool Kit): Libraries and programs for symbolic and statistical natural language processing for English written in the Python programming language.

4. Challenges Faced:

- Multi-label classification
- Pre-processing the text data in title
- Processing all the nodes and edges
- Time taken for tag prediction

5. Contribution:

Name (Roll number)	**Contribution**
Priyadarsan M (18z239)	Dataset Selection and Tag Prediction
Rajasuriya C K (18z240)	Extracting the largest connected sub component and shortest path analysis
Arul Jyothi S (18z206)	Cleaning the dataset for tag prediction
Akilan R (18z204)	Basic metrics analysis on the graph
Mohanaprasanth T (19z431)	Basic metrics analysis on the graph

Annexure – I:

https://github.com/priyadarsanmahendiran/SENA/blob/main/stackov.ipynb

Annexure- II:

```
The questions with id which are most likely to appear along with 4
8
9
11
16
38
39
```

Figure 1: Questions which may be recommended in the feed

```
Hamming loss  0.00017989990649939 07
Precision: 0.7528, Recall: 0.2574, F1-measure: 0.3836
```

Figure 2: Metrics score of tag prediction

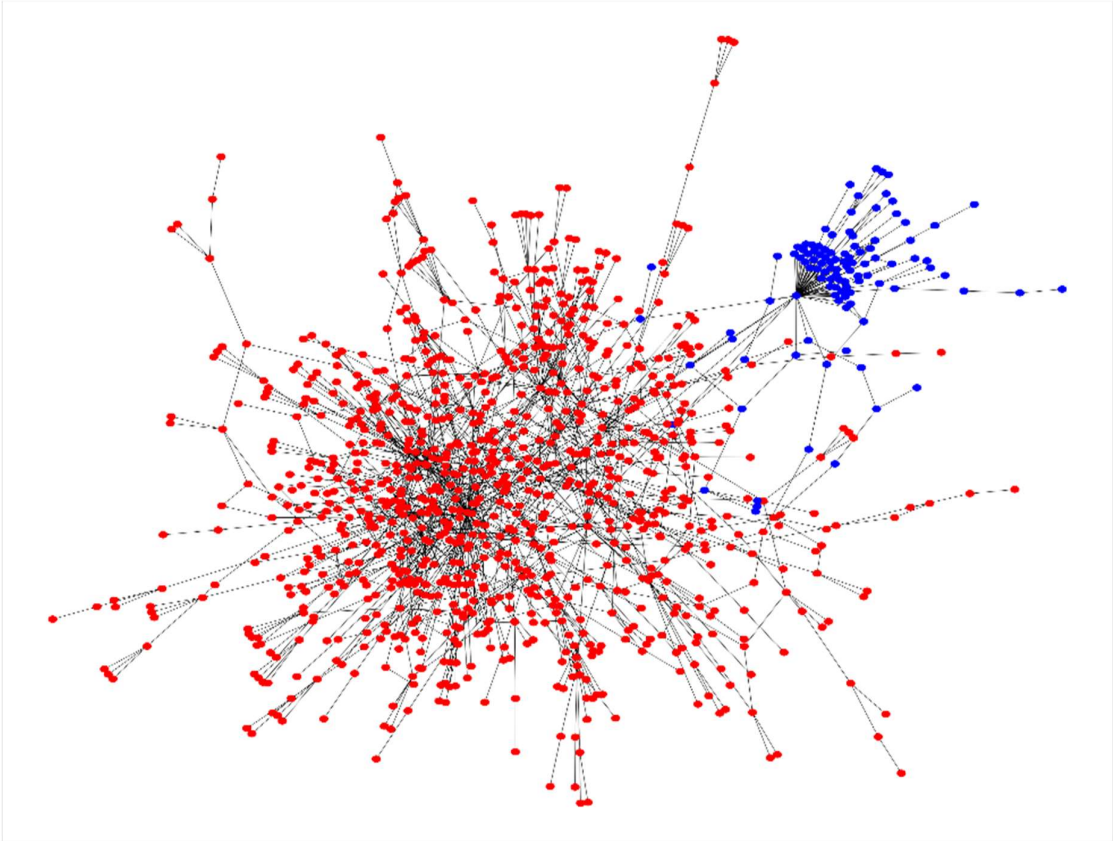


Figure 3: First level community detection of Girvan-Newman algorithm

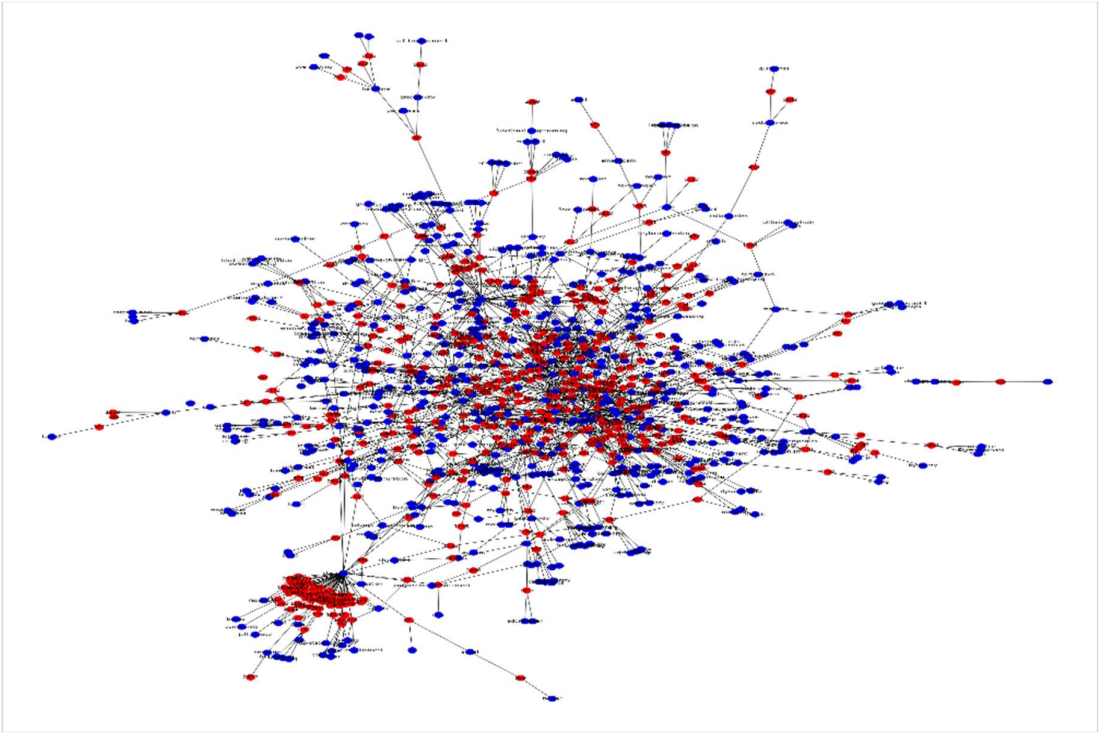


Figure 4: Largest connected sub component (Red ones are question id and blue ones are tags)

Figure 5: The complete graph

References:

[1] Pandas - https://pandas.pydata.org/docs/reference/index.html

[2] Networkx - https://networkx.org/documentation/stable/reference/algorithms/index.html

[3] Nltk - https://www.nltk.org/

[4] Text pre-processing - https://medium.com/@datamonsters/text-preprocessing-in-python-steps-tools-and-examples-bf025f872908

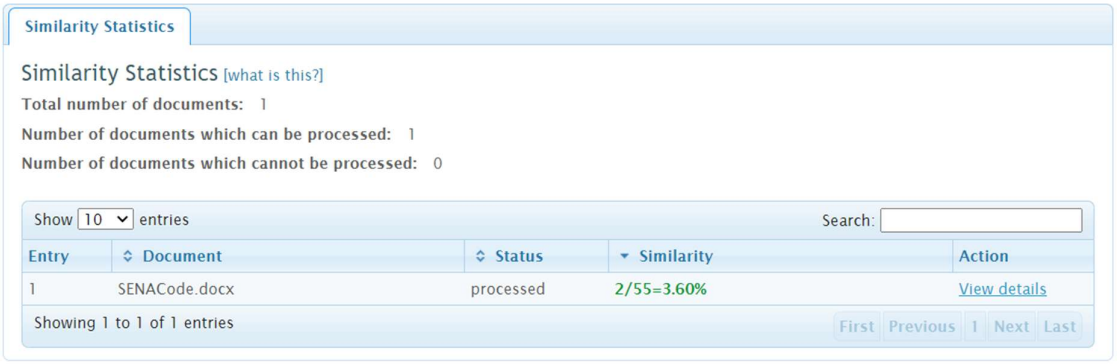
[5] Tag prediction basics - https://www.kaggle.com/miljan/predicting-tags-for-stackoverflow

[6] Count Vectorizer - https://www.educative.io/edpresso/countvectorizer-in-python

[7] SGD Classifier - https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html

[8] TF-IDF Vectorizer - https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

Plagiarism Report:



Similarity Details Selected By User

Index:	**1**
Suspected Sentence:	ad = ad.sort_values(by='AverageDegree', ascending=False)
Source Content:	sort_values(ascending=False)).
Source:	https://stackoverflow.com/questions/64703999/how-to-center-the-names-of-the-bar-in-a-seaborn-barplot-and-add-a-zoom-option-to
From:	Internet

Index:	**2**
Suspected Sentence:	ad = ad.sort_values(by='AverageDegree', ascending=False)
Source Content:	sort_values(ascending=False)).
Source:	https://stackoverflow.com/questions/64703999/how-to-center-the-names-of-the-bar-in-a-seaborn-barplot-and-add-a-zoom-option-to
From:	Internet