DEPARTAMENTO DE ENGENHARIA INFORMÁTICA
Instituto Superior de Engenharia do Porto

Trabalho de LAPR1
Análise de Redes Sociais

Dezembro 2018

1 Objetivos

Com este projeto pretende-se que os alunos desenvolvam uma aplicação em linguagem Java onde apliquem um processo básico de desenvolvimento de aplicações informáticas, valorizando todas as fases do ciclo de desenvolvimento, desde a análise e conceção aos testes de validação. Pretende-se também que os alunos elaborem um relatório que descreva a aplicação concebida, o processo de desenvolvimento e que apresentem e critiquem os resultados obtidos.

Em particular, no projeto a realizar no corrente ano letivo pretende-se que os alunos estudem medidas e algoritmos de Análise de Redes Sociais (SNA) (Zafarani, Abbasi, & Liu, 2014) e desenvolvam uma aplicação que permita analisar redes sociais.

2 Plano de Trabalho

Para facilitar o estudo e desenvolvimento da aplicação, o trabalho de LAPR1 está dividido em duas partes/iterações. Na primeira parte o aluno terá que implementar os módulos necessários à leitura de dados que representam uma rede social e calcular um conjunto de medidas que caracterizam a rede. Na segunda parte do trabalho o aluno terá que combinar as técnicas desenvolvidas durante a primeira iteração juntamente com novos algoritmos/métodos para desenvolver um aplicação funcional que possa ser utilizada por qualquer utilizador para analisar redes sociais.

O enunciado da primeira parte do trabalho é o corrente documento. O enunciado da segunda parte do trabalho será apresentado na semana que inicia a 1 de Janeiro de 2019.

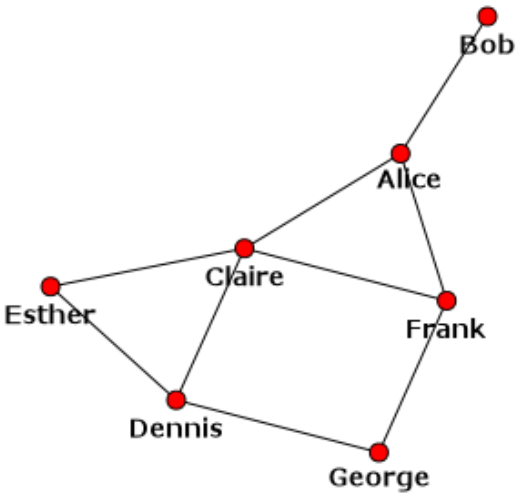

Figura 1: Exemplo de grafo que representa uma rede social de pessoas.

3 Análise de Redes Sociais

Análise de redes sociais (SNA) é um processo de análise quantitativa e qualitativa de uma rede social. SNA refere-se ao mapeamento e à medição de relacionamentos e fluxos entre pessoas, grupos, organizações, computadores e outras entidades relacionadas. O principal objetivo desta técnica é examinar tanto os conteúdos quanto os padrões de relacionamento nas redes sociais, a fim de compreender as relações entre os atores e as implicações dessas relações. Tarefas comuns em SNA envolvem a identificação dos atores mais influentes, prestigiados ou centrais, usando métricas; a identificação de *hubs* e autoridades, usando algoritmos de análise de *links* e a descoberta de comunidades, usando técnicas de deteção de comunidades. Estas tarefas são extremamente úteis no processo de extração de conhecimento das redes e, consequentemente, no processo de resolução de problemas relevantes para a sociedade (Zafarani et al., 2014; Oliveira & Gama, 2012)

Uma rede social pode ser representada por um grafo em que as entidades são representadas através de nós e os relacionamentos entre estas são representados por ramos/arestas (Zafarani et al., 2014; Oliveira & Gama, 2012). Um exemplo de uma rede social pode ser observado na figura 1.

Uma rede social também pode ser representada através de uma matriz de adjacências A (ver Tabela 1). Nesta matriz o valor dos coeficientes, $a_{i,j}$, são determinados em função dos relacionamentos entre nós. Esta matriz é uma matriz quadrada em que o número de linhas (e colunas) é igual ao número de nós do grafo. No caso mais simples, sempre que existir um relacionamento entre quaisquer dois nós o coeficiente toma o valor um, caso contrário toma o valor zero.

A análise de redes sociais realizada através de meios computacionais é muitas vezes realizada aplicando ferramentas de Álgebra Linear (Meyer, 2004), principalmente o cálculo matricial, e um conjunto de algoritmos que exploram uma matriz de adjacências semelhante à apresentada na Tabela 1. Neste processamento também são calculadas um conjunto de métricas que caracterizam uma rede social e que permitem extrair conhecimento e intervir em algum fenómeno.

Tabela 1: Matriz de adjacências que representa a rede social apresentada na figura 1

$$
\mathrm{A} = \begin{array}{c} \\ \textit{Alice} \\ \textit{Bob} \\ \textit{Claire} \\ \textit{Dennis} \\ \textit{Esther} \\ \textit{Frank} \\ \textit{George} \end{array}
\begin{array}{c}
\begin{array}{ccccccc} \textit{Alice} & \textit{Bob} & \textit{Claire} & \textit{Dennis} & \textit{Esther} & \textit{Frank} & \textit{George} \end{array} \\
\begin{pmatrix}
0 & 1 & 1 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 1 & 0
\end{pmatrix}
\end{array}
$$

3.1 Caracterização de uma Rede Social

As medidas apresentadas neste capítulo podem ser divididas de acordo com o nível de análise que se pretende realizar da rede: ao nível dos nós (entidades como atores ou empresas), também conhecidas como medidas de centralidade, que permitem analisar o enquadramento de um vértice dentro da rede, permitindo identificar os principais participantes da rede; ao nível da rede, medidas que permitem analisar a estrutura geral da rede, permitindo obter informação sobre fenómenos/dinâmicas sociais (Zafarani et al., 2014; Oliveira & Gama, 2012).

3.2 Medidas ao Nível dos Nós

- Grau de um nó (*Node degree*)
  O grau de um nó v, geralmente identificado como k_v, é uma medida da adjacência imediata e do envolvimento do nó na rede. Esta medida representa o número de arestas incidentes em um determinado nó. Dito de outra forma, esta métrica representa o número de vizinhos do nó v (Zafarani et al., 2014; Oliveira & Gama, 2012). Esta medida é defina da seguinte forma:

$$k_v = \sum_{j=1}^{n} a_{vj}, \qquad 0 < k_v < n,$$

  em que a_{vj} representa um coeficiente da matriz de adjacências A e n é o número de nós da rede.

- Centralidade de Vetor Próprio (*Eigenvector Centrality*)
  Esta medida generaliza a medida grau do nó, incorporando a importância dos nós vizinhos (Zafarani et al., 2014; Oliveira & Gama, 2012). Esta medida é defina da seguinte forma:

$$x_i = \sum_{j=1}^{n} \frac{1}{\lambda} a_{ij} x_j,$$

  em que x_i representa a centralidade do no i, a_{ij} representa um coeficiente da matriz de adjacências A e λ é o maior valor próprio da matriz de adjacências A.

3.3 Medidas ao Nível da Rede

- Grau médio (*Average Degree*)
  Esta medida representa a média dos graus de todos os nós de uma rede e permite medir a conectividade global desta rede (Zafarani et al., 2014; Oliveira & Gama, 2012). Esta medida é defina da seguinte forma:

$$\bar{k} = \frac{1}{n} \sum_{i=1}^{n} k_i,$$

  em que k_i é o grau do nó i e n é o número de nós da rede.

- Densidade (*Density*)
  Esta medida é importante para explicar o nível geral de conectividade de uma rede. Este valor representa a proporção de ramos na rede em relação ao número máximo possível de ramos. A densidade, representada aqui por a letra ρ, é uma quantidade que varia entre um valor mínimo de 0, no caso da rede não ter ramos, até um valor máximo de 1, caso em que a estamos na presença de um grafo completo. Desta definição podemos concluir que valores elevados estão associados

a redes densas e valores baixos de densidade estão associados a redes esparsas (Zafarani et al., 2014; Oliveira & Gama, 2012). Esta medida é defina da seguinte forma:

$$\rho = \frac{m}{m_{max}}, \qquad 0 < \rho < 1,$$

em que m é o número de ramos da rede e m_{max} é o número de ramos se considerarmos que existe um ramo entre cada um dos pares de nós da rede em análise.

- Potências da Matriz de Adjacências (*Powers of the adjacency matrix*)
  A matriz de adjacência diz-nos quantos caminhos de comprimento um existem entre cada par de nós. A matriz de adjacência ao quadrado diz-nos quantos caminhos de comprimento dois existem entre dois nós. A matriz de adjacência A^k permite obter o número de caminhos de comprimento k entre qualquer para de nós (Zafarani et al., 2014; Oliveira & Gama, 2012). Esta medida é defina da seguinte forma:

$$A^k = \prod_{i=1}^{k} A,$$

em que A^k é a k-enésima potência da matriz de adjacências A.

4 Trabalho a Desenvolver

O trabalho a realizar até ao dia 23 de Dezembro de 2018 consiste no:

- Estudo de análise de redes sociais, em especial as medidas ao nível dos nós e ao nível da rede definidas nas subsecções 3.2 e 3.3. Estudar também o cálculo de valores e vetores próprios.
- Desenvolver uma aplicação que permita analisar redes sociais. Esta aplicação deve implementar as métricas apresentadas nas subsecções 3.2 e 3.3 e efetuar o cálculo dos valores e vetores próprios. O cálculo dos valores e vetores próprios deve ser realizado com o auxílio da biblioteca la4j - linear algebra for Java (http://la4j.org/apidocs/).
- A aplicação deverá permitir carregar redes sociais (grafos) armazenadas e descritas em dois ficheiros de texto (txt), sendo que um ficheiro descreve as entidades da rede (nós) e o outro os relacionamentos entre nós (ramos).
- A aplicação deve ter um interface simples e intuitivo que permita selecionar qualquer das métricas e apresente o respetivo resultado na consola.
- A aplicação deve incluir uma funcionalidade que permita calcular todas as medidas (definidas nas subsecções 3.2 e 3.3) através da execução de um único comando. Neste comando serão especificados os ficheiros que descrevem a rede social e o ficheiro de saída.
- Elaborar um relatório em que são descritas: as métricas ao nível dos nós e da rede, os valores e vetores próprios; a metodologia de trabalho que utilizaram para desenvolver a aplicação; a implementação da aplicação; e a análise de resultados. A descrição da implementação da aplicação deve incluir um diagrama que identifique claramente os módulos e suas dependências. A apresentação das medidas e valores e vetores próprios deve incluir exemplos ilustrativos.

4.1 Formato dos Dados de Entrada e Saída

Os dados de entrada para a aplicação são dois ficheiros que representam uma rede social (grafo). Um dos ficheiros contém a descrição das entidades (nós) e outro a descrição dos relacionamentos (ramos). Qualquer dos ficheiros deve iniciar com uma linha de cabeçalho seguida de uma linha em branco que separa o cabeçalho da informação que caracteriza a rede social. Todos os campos dos ficheiros estão separados por uma vírgula. O nome do ficheiro deve incluir a designação da rede social e o tipo de informação armazenada (nós ou ramos).

Um exemplo do conteúdo de um ficheiro de entrada contendo a descrição dos nós (rs_media_nos.csv):

```
id, media, media.type, type.label

s01, NY Times, 1, Newspaper
s02, Washington Post, 1, Newspaper
s03, Wall Street Journal, 1, Newspaper
s04, ABC, 2, TV
s05, BBC, 2, TV
s06, Yahoo News, 3, Online
...
```

Um exemplo do conteúdo de um ficheiro de entrada contendo a descrição dos ramos (rs_media_ramos.csv):

```
from, to, weight

s01, s02, 1
s01, s03, 1
s01, s04, 1
s04, s11, 1
s05, s15, 1
s06, s17, 1
...
```

O formato dos dados de saída está dependente da operação realizada e deve apresentar de forma clara a informação. No caso em que a saída é um ficheiro, este deve ter um nome que permita identificar o nome da rede social e a data em que o ficheiro foi gerado.

5 Método de Trabalho

- Todos os alunos devem utilizar a metodologia de trabalho definida no eduScrum (Delhij & Solingen, 2013). Cada um dos grupos deve escolher um Scrum Master e este deve ser responsável por gerir a execução de tarefas. Para atingir este objetivo, o grupo deve utilizar a ferramenta Trello e registar as tarefas do projeto, a atribuição de tarefas, o estado de cada tarefa e as tarefas concluídas.

- A aplicação será desenvolvida utilizando o sistema de controle de versões Git e o Bitbucket (https://bitbucket.org). Todos os alunos terão que criar uma conta no Bitbucket com o endereço de email do ISEP (i.e. 1XXXXXX@isep.ipp.pt) e cada grupo terá que criar um repositório. A designação do repositório deve seguir o formato do exemplo "LAPR1_TurmaDAB_Grupo01".

O repositório deve ser partilhado com todos os docentes que lecionam a turma onde o grupo está inserido.

- O grupo deve criar uma pasta no OneDrive onde guarda todo o material desenvolvido para a realização do projeto. A designação da pasta deve seguir o formato do exemplo "LAPR1_TurmaDAB_Grupo01". A pasta será partilhada com todos os docentes que lecionam a turma onde o grupo está inserido. Não é necessário incluir nesta pasta o código que está disponível no repositório do BitBucket.

6 Processo de Desenvolvimento de Software

- A aplicação deve ser estruturada e organizada em módulos. Será valorizada uma correta decomposição modular e o reaproveitamento de módulos.
- O trabalho deverá ser desenvolvido em linguagem Java e deverá resultar num ÚNICO projeto NetBeans.
- Para o cálculo dos valores e vetores próprios será utilizada a biblioteca la4j - linear algebra for Java (http://la4j.org). Esta biblioteca não pode ser utilizada em outras operações que não seja o cálculo de valores e vetores próprios.
- A aplicação deverá ser chamada da linha de comandos utilizando o comando:
  java -jar nome_programa.jar -n rs_nome da rede_nos.csv rs_nome da rede_ramos.csv.
  No caso em que serão calculadas todas as métricas e valores e vetores próprios o comando a utilizar deve incluir o nome do ficheiro de saída:
  java -jar nome_programa.jar -t rs_nomedarede_nos.csv rs_nomedarede_ramos.csv.
- Todos os métodos desenvolvidos terão, obrigatoriamente, de estar associados a testes unitários. Por exemplo, se o aluno criar o método $find\_max\_eigenvalue(matrix)$ para determinar o maior valor próprio de uma matriz de comparação, também deve criar o método $test\_find\_max\_eigenvalue(matrix, expectedMaxEigenValue)$ (ver Algoritmo 1), em que *expectedMaxEigenValue* é o valor do maior valor próprio conhecido da matriz. Estes testes são extremamente úteis para determinar se os métodos estão de acordo com a sua especificação e se a edição destes não alterou a funcionalidade.

```
Bool test_find_max_eigenvalue(matrix, expectedMaxEigenValue)
{

maxEigenvalue = find_max_eigenvalue(matrix);

if(expectedMaxEigenValue==maxEigenvalue)
   return True;
else
   return False;

}
```

Algoritmo 1: Exemplo de métodos de teste unitários

7 Submissão do Trabalho

Datas e entregas de trabalho a efetuar através do Moodle:

- Dia 23 de Dezembro de 2018, até às 24h00m

  - A primeira parte do projeto, incluindo toda a estrutura de diretorias e ficheiros do projeto (incluindo o executável), num único ficheiro comprimido (ZIP).
  - Relatório em formato pdf e não ultrapassando as 20 páginas. A escrita do relatório deve seguir as instruções formais e o modelo disponibilizado nas aulas TP (módulo de competências).

  Nota: Os ficheiros deverão identificar, obrigatoriamente, a designação do grupo e a turma a que os alunos pertencem (Exemplo: "LAPR1_TurmaDAB_Grupo01_projeto.ZIP"e "LAPR1_TurmaDAB_Grupo01_relatorio.PDF").

Referências

Delhij, A., & Solingen, R. (2013). *The eduscrum guide: The rules of the game.* (Disponível em http://eduscrum.nl/file/CKFiles/The_eduScrum_Guide_EN_December_2013_1.0.pdf)

Meyer, C. D. (2004). *Matrix analysis and applied linear algebra (solution).* Philadelphia, PA: Society for Industrial and Applied Mathematics.

Oliveira, M. D. B., & Gama, J. (2012). An overview of social network analysis. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, *2*(2), 99–115. Retrieved from `https://doi.org/10.1002/widm.1048` doi: 10.1002/widm.1048

Zafarani, R., Abbasi, M. A., & Liu, H. (2014). *Social media mining: An introduction.* New York, NY, USA: Cambridge University Press.