Ciência da Computação
Vinícius Vieira
vinicius@ufsj.edu.br

Redes Complexas

Análise de redes reais
(Atividade em duplas)

Realizar uma análise em um conjunto de dados modelados no formato de redes complexas.

Observação geral: O trabalho deve envolver a modelagem de dados como redes complexas, adequando-se ao contexto da disciplina. Entretanto, outras ferramentas de análises de dados (como mineração de dados, aprendizado de máquina, sistemas de recomendação) e conhecimentos de outras áreas (biologia, comunicação, sociologia etc) podem ser utilizas e são bem-vindas.

O trabalho deve ter uma **parte escrita** e uma **apresentação**, explicando a metodologia aplicada e os resultados obtidos.

Cronograma:
- 10/11 (checkpoint 1): Entrega do tema, além do nome da dupla de autores (por email).
- 17/11 (checkpoint 2): Comunicação sobre o andamento do trabalho (por e-mail).
- 24/11 (checkpoint 3): Comunicação sobre o andamento do trabalho (por e-mail).
- 03/12: Entrega da parte escrita e da apresentação

Roteiro para a parte escrita

O trabalho deverá ter uma parte escrita descrevendo contexto, dados, metodologia e resultados. Uma sugestão de roteiro é apresentada a seguir.

- **Contexto:** Onde o trabalho se encaixa?
  - Utilizar exemplos para apresentar o cenário.
- **Objetivo:** O que vocês querem do trabalho?
  - Lembrar que o objetivo do trabalho **não é** apresentar conceitos de redes, mas apresentar o estudo de um contexto real.
  - Para isso, será considerada a abordagem de análise de dados, fazendo uso de ferramentas de redes.
- **Metodologia:** Como foi feito o trabalho?
  - Parte 1: descrição dos dados + modelagem
    - Extração dos dados
      - De onde os dados foram obtidos?
      - O que **faz** parte dos dados?
      - O que **não faz** parte dos dados?
      - O que você usou para extrair os dados?
      - Alguma coisa especial deve ser levada em consideração? (por exemplo, um parser especial teve que ser implementado? os dados foram armazenados em um SGBD?)
    - Modelagem da rede
      - Como a rede foi modelada?
      - O que são os vértices?
      - O que são as arestas?
  - Parte 2: experimentos
    - Análise descritiva básica (apenas sugestão, considerar o que for mais conveniente)
      - Nós
      - Arestas
      - Densidade
      - Distribuição de graus
      - Coeficiente de clustering
    - Análise topológica (apenas sugestão, considerar o que for mais conveniente)
      - Comunidades
      - Spreading
      - Centralidade
- **Discussão:** O que pode ser observado a partir dos resultados?
  - Discutir os resultados obtidos pensando no contexto explorado
- **Conclusões:**
  - O que foi feito?
  - O que não foi possível ser feito?
  - O que poderia ser feito como continuação a esse trabalho?

Roteiro para a apresentação

O trabalho deverá ser apresentado pelos autores. Para isso, uma apresentação de até 15 minutos deverá ser realizada, seguindo os seguintes pontos:

- A apresentação deverá ter até 15 minutos, explicando o contexto, a metodologia (rapidamente), os resultados obtidos e uma discussão.
- A apresentação edverá ser gravada, sendo o link para o vídeo disponibilizado na mesma data da parte escrita.
  - O vídeo deverá ser colocado no Youtube, podendo estar no modo “não listado” (não precisa ser público).
  - Todos os autores devem participar da gravação do vídeo.
- Para a apresentação, é importante utilizar linguagem adequada para apresentação de trabalhos acadêmicos (o que não significa que deve ser extremamente formal, apenas não exagerar na informalidade).

Observações:

1. Os dados podem ter qualquer origem, sendo reais ou artificiais, extraídos pelos autores ou por terceiros (nesse caso, é sempre importante referenciar os autores).
2. O foco é o contexto sempre, não a rede em si. Então não é preciso se aprofundar nos conceitos de redes.
3. Como o foco é o contexto, usar exemplos que ilustrem o contexto, como forma de motivar o leitor.
4. Enviar, junto com o texto, o código fonte de tudo (dos dados, dos códigos e do texto). O texto poderá ser publicado no contexto acadêmico (para motivação de outros alunos). Os dados e o código nunca serão publicados sem o consentimento dos autores.
5. Priorizar o uso de linguagem não técnica (o que não significa que o texto deve abandonar a boa escrita, apenas que qualquer pessoa deve ser capaz de lê-lo).
6. Qualquer biblioteca de software pode ser utilizada. Algumas das mais populares e bem documentadas são:
   1. igraph (https://igraph.org/python/): Uma das mais bem documentadas e discutidas em fóruns (como stackoverflow), possui uma grande variedade de funções para manipulação e análise básica/clássica. É a que eu estou mais habituado a usar, embora eu saiba de algumas de suas limitações, como o fato de ela não ter atualização recente, não tendo implementados vários algoritmos mais modernos e o fato de ter sua eficiência muito baixa para grafos grandes, principalmente quando se vai buscar vértices específicos, o que é uma operação muito lenta e obriga a utilização de estruturas de dados auxiliares para big data.
   2. networkx (https://networkx.github.io/): É também muito bem documentada e também muito discutida em fóruns, com um conjunto muito parecido de funções da igraph, embora ofereça menos algoritmos para análise mais intermediária (como comunidades). Uma das suas vantagens é o fato de que várias bibliotecas auxiliares (como bibliotecas de geoprocessamento e análise de redes urbanas) utilizam a networkx como base, então conhecê-la pode diminuir a curva de aprendizado caso seja importante utilizar alguma solução desse tipo. Também sofre de gargalo de desempenho, de forma semelhante à igraph, quando é necessário percorrer vértices.
   3. graph-tool (https://graph-tool.skewed.de/): É uma biblioteca um pouco mais nova, mas muito bem documentada (embora seja menos utilizada em fóruns). É desenvolvida por um brasileiro (Tiago Peixoto), professor da Universidade da Europa Central, que é um importante pesquisador da área de redes atualmente. Uma das grandes vantagens da graph-tool é a preocupação do desenvolvedor em manter a biblioteca atualizada, sendo capaz de executar grafos grandes, e com algoritmos que incorporam avanços recentes e sua grande atividade em mantê-la/aprimorá-la. Atualmente, é a única biblioteca que eu vejo com atividade constante e acredito que vale muito a pena aprender a utilizá-la (acredito que em breve será a mais utilizada).
   4. SNAP (http://snap.stanford.edu/): É uma biblioteca com uma documentação um pouco mais complicada, desenvolvida por um importante pesquisador da área de redes (Jure Leskovec), professor da Universidade de Stanford. Como ela foi criada com o objetivo de auxiliar o desenvolvedor em sua tese de doutorado que envolvia redes de larga escala, uma das vantagens da biblioteca é sua eficiência em grandes redes. Outra vantagem é que ela incorpora algoritmos recentes, principalmente na área de comunidades, mas basicamente os desenvolvidos pelo próprio autor. Vale a pena apenas se a preocupação for grandes grafos (dezenas de milhões de vértices) ou se for necessário executar algum algoritmo do próprio autor.
7. Qualquer linguagem de programação pode ser utilizada. As bibliotecas de software são, geralmente, escritas para serem executadas em python e C (igraph e networkx também rodam para R, são usadas principalmente para essa linguagem). Todas são bem fáceis de instalar, principalmente em python (por apt-get ou pip). A facilidade de integração das bibliotecas na versão python com outras bibliotecas é um ponto para python (principalmente pelo fato de que a análise de redes frequentemente envolve outras etapas, como manipulação de arquivos texto e algoritmos de aprendizado de máquina). Mas as bibliotecas também funcionam bem e são bem documentadas para C (geralmente a versão python é um wrapper, inclusive).