

Sistema para visualização e análise de interações pedagógicas online

André Cruzeiro Rei

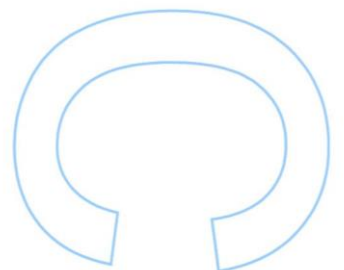
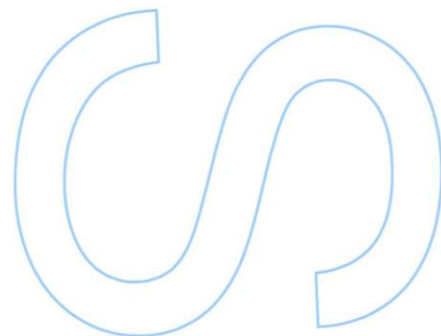
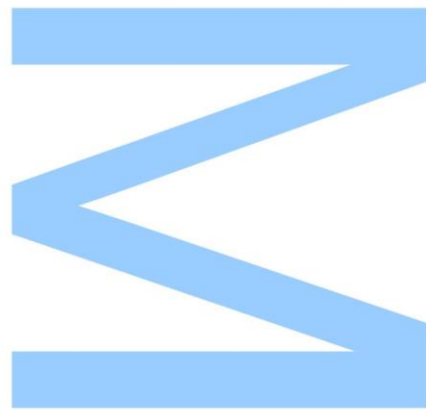
Mestrado Integrado em Engenharia de Redes e Sistemas Informáticos
Departamento de Ciência de Computadores
2017

Orientador

Álvaro Figueira, FCUP

Coorientador

Luciana Oliveira, ISCAP

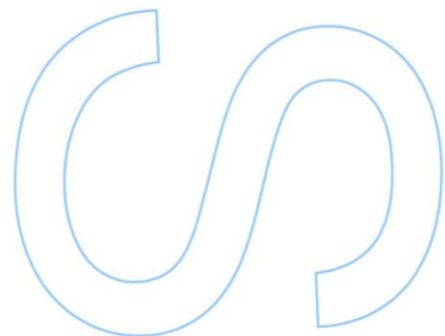
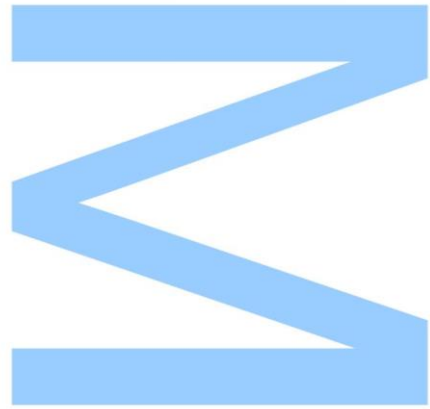




Todas as correções determinadas pelo júri, e só essas, foram efetuadas.

O Presidente do Júri,

Porto, ____ / ____ / ____



Abstract

We present a system for a dynamic graphical representation of the interactions captured in educational online environments. The system goes beyond interaction between students and teachers, also addressing resource usage or any other entity for which it is possible to create a relation which binds two entities. By defining these relationships between pairs of entities in an online learning environment (Moodle, in our case) our tool creates a graph, where it is possible to apply techniques of social network analysis.

This system brings up new possibilities for e-learning as a tool capable of helping the teacher assorting and illustrating the degree of participation and to find the implicit relations between participants, or participants and resources or events.

Resumo

Apresentamos um sistema para representação gráfica dinâmica de interações capturadas em ambientes educativos. O sistema vai além da interação entre estudantes e professores, abordando também o uso de recursos ou qualquer outra entidade para a qual é possível criar uma relação que vincula duas entidades. Ao definir essas relações num ambiente de ensino-aprendizagem online (o Moodle, no nosso caso) o nosso sistema cria um grafo, onde é possível aplicar técnicas de análise de redes sociais.

Este sistema traz novas possibilidades para o e-learning como uma ferramenta capaz de ajudar o professor a classificar e ilustrar o grau de participação e a encontrar relações implícitas entre participantes, ou participantes e recursos ou eventos.

Agradecimentos

Este trabalho não seria possível sem a ajuda dos meus orientadores de tese, o professor Álvaro Figueira e a professora Luciana Oliveira. Durante os meses do desenvolvimento deste projeto, ajudaram-me sempre que requisitei a sua ajuda, deram-me os melhores conselhos, orientaram-me no caminho certo e a sua experiência foi fundamental para a escrita deste relatório.

Quero agradecer ao meu melhor amigo e colega de curso Marcelo Gomes, por me ajudar na resolução de problemas, na troca de ideias e por ser um bom companheiro.

Quero agradecer à minha namorada Ana Clara, por ser sempre o meu porto seguro nos momentos mais complicados, por ser uma boa ouvinte e por toda a ajuda e apoio que me forneceu no desenvolvimento desta tese.

Por último quero agradecer aos meus pais, por tudo o esforço que fizeram para eu poder completar este curso. Sempre me deram uma boa educação, os melhores conselhos, acreditaram nas minhas capacidades e apontaram o melhor caminho a seguir quando tinha dúvidas e dificuldades. Sem eles não estaria onde estou hoje, não estaria a escrever este relatório e nem estaria a acabar este mestrado.

Um muito obrigado a todos.

Dedico às duas pessoas que são o orgulho da minha vida, os meus pais

Conteúdo

Abstract	5
Resumo	7
Agradecimentos	9
Conteúdo	16
Lista de Figuras	19
Lista de Blocos de Código	21
Acrónimos	23
1 Introdução	1
1.1 Motivação	4
1.2 Objetivos	4
1.3 Estrutura da tese	5
2 Background	7
2.1 Teoria de Grafos	7
2.2 Armazenamento Digital de Grafos	10
2.2.1 Matriz de adjacências	10
2.2.2 Lista de arestas	11
2.2.3 Formatos JSON	11

2.2.4	Formatos XML	11
2.2.5	GML	11
2.2.6	CSV	12
2.2.7	Outros Formatos	12
2.3	Métricas de SNA	12
2.3.1	Densidade	13
2.3.2	Medidas de centralidade	14
2.3.2.1	Centralidade local	14
2.3.2.2	Centralidade global	14
2.3.2.3	Eigenvector centrality	15
2.3.2.4	Closeness centrality	15
2.3.2.5	Betweenness centrality	15
2.3.2.6	Brokerage Centrality	16
2.3.3	Centralização	17
2.3.4	Coefficiente de agrupamento	18
2.4	Estruturas e Grupos na rede	18
2.4.1	Cliques	18
2.4.2	Componentes, Cut Points e Bridges	19
2.5	Visualização de grafos	20
2.5.1	Medidas de qualidade e estética	20
2.5.2	Layouts	21
2.5.2.1	Force-Directed Layout	21
2.5.2.2	Planar Layout	22
2.5.2.3	Layout Circular	22
2.5.2.4	Layout em Arco	22
2.5.2.5	Layout em Árvore	24
2.5.2.6	Layout em Matriz	24
2.5.2.7	Layout Híbrido	25

<i>CONTEÚDO</i>	15
2.5.3 Visualização semântica	26
2.5.4 Visualização temporal e mudanças na rede	26
2.5.5 Software e bibliotecas para visualização de grafos	28
2.6 Resumo	28
3 Arquitetura do Sistema	31
3.1 Sistema Moodle	31
3.2 Modelo de dados	32
3.3 API de manipulação de dados do Moodle	33
3.4 Ficheiros de input	34
3.5 Bibliotecas Javascript	35
4 Interface do Utilizador	37
4.1 Interação com o Grafo Social utilizando o cursor	38
4.2 Interação através do Dashboard	40
4.2.1 Configuração do grafo social	41
4.2.2 Pesquisa de um nó	43
4.2.3 Detecção de comunidades	45
4.2.4 Ver uma comunidade	46
4.2.5 Caminho mais curto entre nós	47
4.2.5.1 Caminho mais curto num grafo não-pesado	49
4.2.5.2 Caminho mais curto num grafo pesado	50
4.2.5.3 Nós alcançáveis	52
4.2.6 Cliques	53
4.3 Resumo	56
5 Aplicação prática do Sistema	57
5.1 Análise do Grafo Social	57
5.2 Interação através do Dashboard	61

5.3	Análise das propriedades do Grafo Social	63
5.4	Discussão	64
6	Conclusões	67
6.1	Resumo do trabalho desenvolvido	67
6.2	Contribuições	68
6.3	Trabalho futuro	69
	Bibliografia	71

Lista de Figuras

2.1	Diferentes tipos de arestas	8
2.2	Percursos, caminhos e trilhos	8
2.3	Distância geodésica	9
2.4	Diâmetro da rede	9
2.5	Matriz de adjacências e respectivo grafo	10
2.6	Comparações de densidade	13
2.7	Diferentes tipos de relações Brokerage	16
2.8	Grafo altamente centralizado	17
2.9	Centro estruturado de um grafo	18
2.10	Uma rede com quatro cliques	19
2.11	Force-Directed layout	21
2.12	Layout planar	22
2.13	Layout circular	23
2.14	Diagrama em arco	23
2.15	Layout em árvore	24
2.16	Layout em matriz	25
2.17	MatLink	25
2.18	Relacionamentos fortes da rede no tempo 1	27
2.19	Relacionamentos fortes da rede no tempo 2	27
2.20	Grafo da mudança de posição dos nós na rede em dois períodos de tempo.	27

3.1	Arquitetura do Sistema	31
3.2	Ficheiros de input para D3.js e JSNetworkX.js	34
4.1	Interface do utilizador na página de um curso do Moodle	37
4.2	Movimentação da posição dos nós do Grafo Social utilizando o cursor	39
4.3	Caixa de informações de um nó do Grafo Social	39
4.4	Representação dos vizinhos diretos de um nó do Grafo Social	40
4.5	Região “Social Graph Configuration” do Dashboard	41
4.6	Configurações de visualização do Grafo Social #1	42
4.7	Configurações de visualização do Grafo Social #2	42
4.8	Grafo direcionado e não-pesado com nós pequenos	43
4.9	Região “Search node” do Dashboard	43
4.10	Mensagem de erro da região “Search node” devido a texto inserido inválido	44
4.11	Procurar um nó no Grafo Social	44
4.12	Região “Detection of student communities” do Dashboard	45
4.13	Deteção de comunidades no Grafo Social	45
4.14	Região “View a particular cluster” do Dashboard	46
4.15	Região “View a particular cluster” ativa, após o cálculo dos clusters do Grafo Social	46
4.16	Mensagens de informação e de erro da região “View a particular cluster”	47
4.17	Ver uma comunidade no Grafo Social	47
4.18	Região “Shortest Path” do Dashboard	48
4.19	Mensagem de erro no calculo de um “Shortest Path”	48
4.20	Região “Shortest Path” para a opção “S.P. in Unweighted Graph”	49
4.21	Caminho mais curto entre dois nós no Grafo Social não direcionado e não-pesado	50
4.22	Comparação do caminho mais curto entre dois nós entre a opção “S.P. in Unweighted Graph” e a opção “S.P. in Weighted Graph”	51
4.23	Mensagem de aviso devido à inexistência de um caminho entre dois nós para a opção “S.P. in a Unweighted Graph”	51
4.24	Região “Shortest Path” do Dashboard para a opção “Single Source Dijkstra Path”	52

<i>LISTA DE FIGURAS</i>	19
4.25 Nós alcançáveis a partir do nó “Magnon” do Grafo Social	53
4.26 Região “Cliques” do Dashboard	54
4.27 Mensagens de erro e de aviso da região “Cliques” do Dashboard	54
4.28 Procurar cliques com um tamanho predefinido no Grafo Social	55
5.1 Detecção de comunidades no Grafo Social	58
5.2 Detecção de comunidades no Grafo Social	58
5.3 Representação de uma aresta para o mesmo nó no Grafo Social	59
5.4 Representação dos vizinhos diretos de um nó no Grafo Social	60
5.5 Comparação entre as configurações de visualização “In-Degree nodes” e “Out-Degree nodes”	60
5.6 Caminho mais curto entre dois nós no Grafo Social direcionado e pesado	61
5.7 Nós alcançáveis a partir de um nó no Grafo Social	62
5.8 Região “Cliques” do Dashboard	62
5.9 Detecção de cliques no Grafo Social com um tamanho específico	63
5.10 Propriedades do Grafo Social	63

Lista de Blocos de Código

3.1	API de manipulação de dados do Moodle	34
3.2	Formato do ficheiro “Graph.json”	35

Acrónimos

API	Application Programming Interface	JSON	JavaScript Object Notation
CMS	Content Managment System	LMS	Learning Management System
CSV	Comma-Separated Values	SCRM	Social Customer Relationship Management
e-Learning	Eletronic Learning	SNA	Social Netork Analysis
GEXF	Graph Exchange XML Format	SSRM	Social Student Relationship Management
GML	Graph Modeling Language	XML	eXtensible Markup Language
GraphML	Graph Markup Language		
JGF	JSON Graph Format		

Capítulo 1

Introdução

As redes sociais aproximam as pessoas de muitas formas criativas: os utilizadores trabalham, jogam, socializam, trocam ideias e informações online, demonstrando novas formas de colaboração e comunicação que seriam dificilmente imagináveis há algum tempo atrás. Para além disso, as redes sociais desempenham um papel crucial no processo empreendedor e também ajudam a reformular modelos de negócio, bem como abrir inúmeras possibilidades para estudar as interações humanas e o comportamento coletivo [36].

Entende-se por rede social uma estrutura social entre indivíduos que estão ligados entre si através de várias familiaridades sociais, que podem consistir em conexões casuais, laços familiares ou de amizade [42]. Hoje em dia, a Internet desempenha um papel cada vez mais importante no contexto social global, devido à sua rica variedade de recursos e possibilidades, em particular na área da educação. As redes sociais podem ser úteis nos processos de aprendizagem colaborativa, uma vez que a relação entre os estudantes e os professores não fica restrita ao tempo de contacto em sala de aula. Deste modo, o conceito de Eletronic Learning (**e-Learning**) consiste num meio de colaboração, transmissão e construção de informação e conhecimento através de computadores. A tecnologia moderna introduziu novas formas de ensino, e o **e-Learning** é uma modalidade em constante crescimento, oferecendo diversas vantagens tais como: a conveniência, a eficácia, as facilidades de aprendizagem e a liberdade de expressão.

O conceito de Social Student Relationship Management (**SSRM**) consiste na particularização do conceito de Social Customer Relationship Management (**SCRM**), aplicado ao relacionamento entre provedores educativos, tais como escolas e universidades, e estudantes através das redes sociais [31]. **SSRM** visa amplificar as ligações formais e informais entre alunos e escolas e expandir as suas interações, transformando-as em conversações sociais e gerando o fortalecimento de laços educativos através do desenvolvimento de conversações colaborativas. Tais conversações resultam em benéficos mútuos e no crescimento de comunidades educativas em ambientes sociais.

Um estudo realizado por Oliveira [53] considera que o conceito de **SSRM** é baseado em três componentes principais: a comunicação educativa entre professor e alunos, a administração do ambiente de ensino-aprendizagem e a integração social dos estudantes. No primeiro componente,

a comunicação serve como veículo para atingir os objetivos educacionais. Esta comunicação é composta por comunicação formal e informal, de modo a tirar partido dos dois mundos e garantir quer a eficácia na aprendizagem, por parte dos alunos, quer a sua interação. A comunicação educativa formal é sustentada por um modelo pedagógico com *curriculum*, métodos, requisitos e avaliação claros e institucionalmente definidos e validados. Por outro lado, a comunicação educativa não é estruturada ou institucionalizada, é caracterizada pela espontaneidade, conversações simples e discussões de grupo. O segundo componente está diretamente dependente da comunicação administrativa entre alunos e docente e visa regular as atividades de aprendizagem, prazos e procedimentos. Por fim, para que a integração social dos alunos possa decorrer, é necessário criar e desenvolver uma comunidade de aprendizagem. Esta comunidade tem de ser sustentada pela interação social e depender fortemente do estabelecimento de conversações, que por sua vez criam laços sociais e emocionais. O docente, como construtor e gestor desta comunidade, deve fomentar a motivação, cooperação, integração e comunicação formal e o formal entre os participantes, de modo a evitar o desinteresse e a desistência dos alunos e garantir um clima educativo cativante e saudável.

As tecnologias da Web 2.0 tem um enorme potencial para modelar a forma como os humanos aprendem e ensinam, e a natureza social destas tecnologias cria ótimas condições para os ambientes ensino-aprendizagem, uma vez que a maioria dos estudantes estão bastante confortáveis com o uso da tecnologia, mesmo antes de ingressar no ensino superior [56].

Dentro da categoria de ambientes virtuais de ensino-aprendizagem, existe uma categoria específica de aplicações denominada de Learning Management System (**LMS**), também chamado de plataforma de **e-Learning**. Um **LMS** é um software criado para gerenciar atividades de aprendizagem e ensino dos utilizadores, tendo em consideração os recursos que tornam isso possível [21]. **LMS** s *open source* ajudam as instituições a atualizar a sua estrutura pedagógica, permitindo que professores e administradores criem ambientes virtuais de ensino-aprendizagem, de forma a que os objetivos educativos sejam atendidos da melhor maneira possível. Os ambientes de ensino-aprendizagem criados num **LMS** dispõe de um conjunto de funcionalidades projetadas para criar, armazenar, distribuir e gerenciar conteúdos de aprendizagem, de forma progressiva e interativa, bem como facilitar as tarefas de planeamento, acompanhamento e avaliação do curso. Deste modo, de acordo com [22], **e-Learning** fornece aos alunos uma rede virtual rica, onde podem trocar ideias e informações e criar sinergias através de interações com outros membros da rede”. A estratégia educativa implícita visa dar suporte os alunos construam o seu conhecimento através da discussão, da reflexão e tomada de decisões, e onde os recursos informáticos atuam como mediadores do processo de ensino-aprendizagem.

As interações sociais entre estudantes são uma parte importante e sub-explorada no contexto pedagógico. A sala de aula é o principal domínio onde as relações sociais e de trabalho se formam entre os alunos. Estes relacionamentos, e as redes cada vez maiores que eles formam, tem efeitos significativos no comportamento dos estudantes. Perceber como é que estas relações se formam, e qual é o seu impacto nos processos de aprendizagem, pode ajudar os professores e administradores a melhorarem os seus processos educativos, e detetar de forma mais rápida e eficaz pontos de

melhoria para o sucesso da turma [37]. A análise de redes pode melhorar a compreensão da formação destas redes e os tipos de impactos que essas redes tem nos estudantes. Este conjunto de abordagens metodológicas e teóricas pode ajudar a responder a questões sobre os métodos pedagógicos, equidade, aprendizagem, organização e política educativa.

Social Network Analysis (*SNA*) compreende o mapeamento e métricas aplicadas aos relacionamentos e fluxos entre pessoas, grupos, organizações, computadores ou outra entidade de processamento. Os nós da rede constituem pessoas ou grupos, enquanto que as ligações entre eles representam os relacionamentos entre os nós. *SNA* fornece uma análise matemática e visual dos relacionamentos humanos [42]. *SNA* tornou-se um método vastamente aplicado para analisar as relações mantidas entre indivíduos e organizações. Com o rápido acesso às capacidades computacionais, a popularidade e a proliferação de redes sociais como o Facebook, e de métodos automáticos de coleção de dados, a exigência por conhecimento sólido em *SNA* explodiu recentemente [36]. As diferentes técnicas de *SNA* permitem revelar determinadas características e propriedades de uma rede, identificar indivíduos chave e grupos na rede social, detetar e gerar estruturas fundamentais na rede, bem como desenhar um modelo visual da rede a partir do qual é possível extrair informações relevantes sobre a sua estrutura.

SNA compreende normalmente duas fases principais: o uso de teoria formal organizada em conceitos matemáticos – a teoria de grafos; seguida de uma análise sistemática de dados empíricos [36]. A análise de redes sociais estuda principalmente dois tipos de dados, normalmente designados por dados de atributos e dados relacionais. Dados de atributos são relativos às atitudes, opiniões e comportamentos dos agentes. Estes são considerados propriedades, qualidades ou características que pertencem aos indivíduos ou grupos. Os dados relacionais, por outro lado, consistem nos contactos, ligações, conexões e agrupamentos que relacionam um indivíduo com outro indivíduo e não podem ser reduzidas a propriedades individuais destes. As relações não são propriedades dos indivíduos, mas sim do sistema relacional construído a partir de pares conectados das interações entre indivíduos. Os métodos apropriados para dados relacionais são aqueles relacionados com a análise de redes, na qual as relações se representam por uma ligação entre dois indivíduos [60].

A visualização de grafos é uma das áreas mais importantes em *SNA*. A ubiquidade e popularidade das redes sociais levou à necessidade de desenvolver uma formula rápida e eficiente de estudar e perceber estas redes. Contudo, como as redes se tornam cada vez maiores e mais complexas, a análise das dinâmicas sociais através de simples estatísticas não é uma opção exequível para o fazer. De modo a ultrapassar esta limitação, podemos confiar em metáforas visuais, utilizando representação de grafos [24]. No entanto, diferentes medidas de qualidade e estética na representação de grafos levou ao desenvolvimento de diferentes layouts, cada um com as suas vantagens e desvantagens e apropriados a diferentes situações [55]. Cabe ao investigador a decisão do layout a utilizar, mediante aquilo que pretende explorar da rede e do tipo de rede. Um layout apropriado é o melhor caminho para se obter uma representação clara e compreensiva da rede, na qual a aplicação de técnicas e métodos de *SNA* permitem obter um estudo fidedigno dos relacionamentos sociais da rede.

1.1 Motivação

Recentemente, Cela et al. [22] realizou um estudo aprofundado sobre o uso de **SNA** em **e-Learning**, de modo a perceber como é que **SNA** pode beneficiar o **e-Learning** e os seus impactos, bem como o estado da arte e as direções para trabalhos futuros nesta área. Dos 37 estudos abrangidos pela sua investigação, apenas 14 analisaram um **LMS** ou um Content Management System (**CMS**). Este facto sugere que, para além do facto de esta ser uma área emergente e estes estudos estão a ser publicados numa frequência crescente, eles ainda estão numa fase inicial. Como consequência, pretendemos resolver este problema detetado, e um sistema para visualização e análise de interações pedagógicas online, integrado num **LMS**, é proposto.

O sistema proposto ambiciona conciliar as vantagens do uso de um **LMS** com as técnicas e métricas oferecidas por **SNA** e visualização de grafos e, deste modo, fornecer ao meio académico uma ferramenta complementar e de suporte aos processos pedagógicos. Contrariamente a vários estudos prévios [30],[64] e [65], neste trabalho o sistema não é restringido a representar interações entre os participantes de um curso (estudantes e professor), e dá mais um passo ao representar qualquer interação que ocorra, e que possa ser capturada automaticamente, durante um curso online. Para além disso, também pode ser representado o uso de recursos. Esta abordagem está em linha com estudos recentes, de [70] e [28], por exemplo. O sistema consiste num módulo de visualização e análise que contém uma representação de um Grafo Social e um painel de controlo para análise do grafo, com o auxílio das métricas mais comuns em **SNA**. O **LMS** escolhido para a base do desenvolvimento do sistema foi o Moodle, uma vez que é um dos **LMSs open source** mais usados em todo mundo.

1.2 Objetivos

O sistema tem como objetivo ser flexível e servir diversas situações educativas. Deste modo, o sistema pode ser usado para representar e visualizar vários aspetos de um ambiente de ensino-aprendizagem virtual, tal como as relações entre aluno e professor, atividades ou eventos.

Com o sistema proposto, o docente pode ter *feedback* extra da sua unidade curricular ao analisar o Grafo Social, como por exemplo ver quais são os alunos mais interessados e os menos interessados, quais os alunos menos participativos e a que tem de dar mais apoio. Este panorama geral ajuda o professor a estruturar melhor a sua unidade curricular e a ter um acompanhamento mais ativo da comunidade dos alunos, aspetos que por vezes são difíceis de gerir numa sala de aula convencional.

Concluindo, os objetivos principais deste trabalho são:

- Implementação de um sistema web em modo cliente-servidor, com processamento do lado do cliente, que represente a análise interações pedagógicas online;

- Implementação de um modelo de dados que permite o armazenamento individual dos dados do Grafo Social para diferentes cursos criados no Moodle;
- Desenho automático do Grafo Social no Moodle, usando um layout que permita identificar propriedades importantes para a identificação e análise da turma;
- Apresentar numa página do Moodle a análise das interações com várias das métricas mais usadas em **SNA**.

1.3 Estrutura da tese

Ao presente capítulo introdutório seguem-se cinco capítulos, nomeadamente o Background, a Arquitetura do Sistema, a Interface do Utilizador, Resultados Aplicacionais e a Conclusão.

No capítulo 2, são apresentados os conceitos teóricos e matemáticos que fundamentam este projeto. É abordado em detalhe os principais conceitos de teoria de grafos e os diferentes formatos de armazenamento digital. De seguida é feita uma revisão das métricas mais comuns e importantes em **SNA**. Por fim, são descritos os aspetos mais relevantes em visualização grafos, tais como as medidas de qualidade e estética, os diferentes layouts para representação de grafos, visualização semântica e temporal e os softwares e bibliotecas mais comuns para visualização e representação de grafos.

No capítulo 3, é apresentada a arquitetura do sistema proposto. É feita uma descrição de todos os componentes que compõem o sistema. De igual modo, é descrito a interligação entre os componentes, desde a definição do modelo de dados à interface do utilizador final no Moodle.

No capítulo 4, são descritos os componentes que compõem a interface do utilizador no **LMS** Moodle. Adicionalmente, é explicado com exemplos práticos todas as ações que o utilizador pode tomar em cada componente da interface, nomeadamente no Grafo Social, no Dashboard e na região das métricas globais do grafo.

No capítulo 5, é apresentado um exemplo prático da aplicação do sistema proposto. É realizado uma análise detalhada do Grafo Social criado, bem como as suas propriedades e métricas aplicadas.

Por último, no capítulo 6, é feita uma síntese do trabalho desenvolvido e descrito as suas principais contribuições. Por fim, são apresentadas algumas propostas de desenvolvimento para um trabalho futuro.

Capítulo 2

Background

Social Network Analysis (**SNA**) envolve o estudo de estruturas sociais usando redes e teoria de grafos, sendo considerada uma área chave e de grande relevo na sociedade moderna. Este capítulo aborda: os conceitos que constituem a base de teoria de grafos; como podemos armazenar digitalmente um grafo; os recursos e técnicas existentes para a representação gráfica de grafos; bem como uma descrição dos conceitos matemáticos inerentes às diferentes métricas de **SNA** mais relevantes para o propósito deste trabalho.

2.1 Teoria de Grafos

As entidades de estudo em **SNA** são tipicamente indivíduos, especialmente humanos ou outra espécie social, que formam redes sociais com os seus relacionamentos. Encontramos redes em muitos outros campos, como em ciência de computadores, na genética, na física, na química, entre muitas outras. Todas estas instâncias de redes têm uma coisa em comum, que é uma base de matemática discreta chamada teoria de grafos [19].

Uma maneira de conceptualizar matematicamente uma rede é através de um grafo. Um grafo $G(V, E)$ é composto por dois conjuntos: o conjunto dos nós V e o conjunto das arestas E que ligam os nós [19]. Nó, também chamado de ator ou vértice, refere-se a um agente ou indivíduo particular que pode ter relacionamentos com outros indivíduos. Uma aresta, relação ou arco, descreve um relacionamento particular e específico entre dois nós. Para expressar que uma aresta liga dois nós u e v existe no grafo G , escrevemos $(u, v) \in E(G)$.

Os conceitos de teoria de grafos são usados para descrever os padrões de conexões entre os nós, e o conceito mais simples são as propriedades dos nós individuais e das arestas com os quais um grafo é construído [60]. É necessário primeiro considerar os tipos de arestas que podem ser usadas, uma vez que estas podem corresponder a diferentes tipos de dados relacionais: diretos, indiretos, pesados, ou ambos diretos e pesados.

Grafos compostos por arestas não direcionadas, como o grafo i) da figura 2.1, derivam de

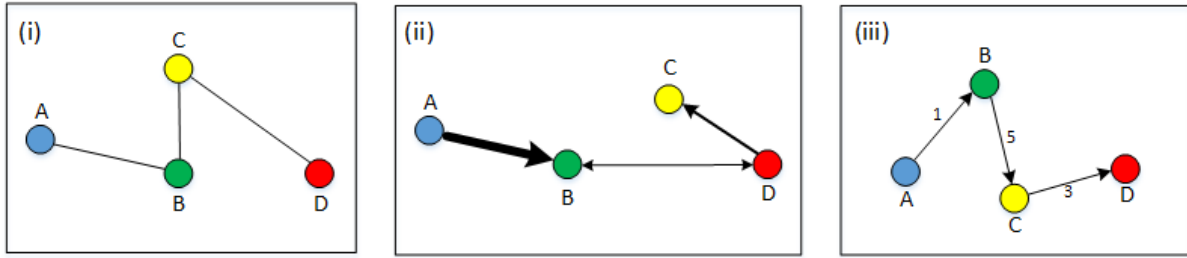


Figura 2.1: Diferentes tipos de arestas

uma matriz de dados simétrica onde o que simplesmente importa é a presença ou ausência de uma relação. Caso as relações forem direcionadas de um nó para outro, elas podem ser representadas por um grafo direcionado, usualmente denominado digrafo. Num digrafo, todas as arestas são representadas com uma seta na ponta, e a direção da seta representa a direção da relação. Usualmente, uma seta direcionada denomina-se arco. Os grafos ii) e iii) da figura 2.1 são digrafos. Uma matriz de um grafo dirigido pode não ser simétrica, uma vez que as relações podem não ser recíprocas.

Por outro lado, a intensidade das relações pode ser representada com um valor numérico associado a cada relação, construindo assim um grafo pesado. A matriz de um grafo pesado contém valores numéricos em vez de valores binários. Por vezes, a representação de arestas pesadas em grafos faz-se sem ter um valor numérico visível associado à aresta, mas sim a espessura da aresta é que traduz tal informação. Quanto mais fina a linha da aresta, menor é o seu peso, e vice-versa.

Os nós podem ser diretamente ligados por uma linha, ou podem ser indiretamente ligados por uma sequência de arestas, formando um percurso. O caso particular de um percurso em que cada nó é distinto é chamado de “caminho”. Por outro lado, uma sequência que revisita nós, mas nunca revisita uma aresta é chamado de “trilho” [19] [60]. O tamanho de um percurso, caminho ou trilho é definido pelo número de nós que ele possui.

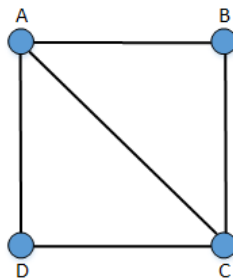


Figura 2.2: Percursos, caminhos e trilhos

Considerando o grafo da figura 2.2, AD é um caminho de tamanho 1 – é uma linha, enquanto ABCD é um caminho de tamanho 3. O percurso ABCAD não é um caminho, uma vez que passa duas vezes no ponto A. Podemos dizer que os nós A e D estão ligados por 3 caminhos distintos: AD de tamanho 1, ACD de tamanho 2 e ABCD de tamanho 3. Contudo, a distância entre A e

D é tamanho do caminho mais curto, que neste caso é 1. A sequência ADCAB é um trilha, mas a sequência ADCACB não é um trilha uma vez que a aresta AC é usada duas vezes.

Os mesmos conceitos referidos anteriormente podem ser aplicados em grafos dirigidos, com algumas modificações [60]. As arestas nestes grafos estão direcionadas de e para vários nós. Como tal, deve se ter em conta cada linha de acordo com a sua direção, e deste modo não vai existir a simetria que existe em dados relacionais simples e sem direção. Um caminho num grafo dirigido é a sequência de arestas na qual todas as setas apontam para a mesma direção.

A distância geodésica define o menor número de conexões que precisam de ser atravessadas entre quaisquer dois nós, por outras palavras, representa o caminho mais curto entre dois nós [60]. A distância geodésica não é necessariamente única, uma vez que podem existir múltiplos caminhos curtos de igual tamanho entre dois pares de nós [19]. A distância geodésica na rede ilustrada na figura 2.3, entre o nó A e o nó D é 3, enquanto a distância entre o nó B e C é apenas 1.

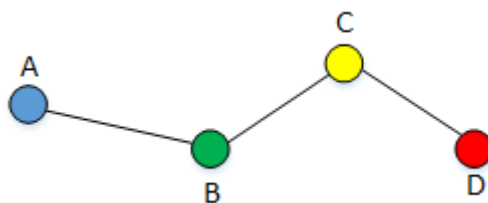


Figura 2.3: Distância geodésica

O diâmetro de uma rede é definido como o mais longo de todos os caminhos mais curtos calculados entre os nós [25]. Os nós e arestas em tom vermelho representam o diâmetro da rede da figura 2.4.

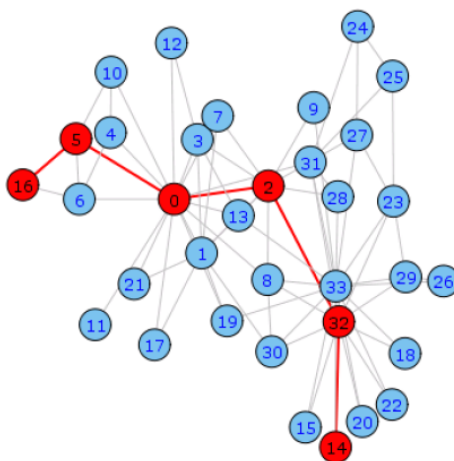


Figura 2.4: Diâmetro da rede

O diâmetro de uma rede dá-nos a ideia de quão longe estão os nós na rede. Deste modo, um diâmetro muito grande significa que, caso exista teoricamente um caminho por arestas a ligar dois nós através de uma série de intermediários, não há garantia de que eles estão na verdade

ligados. O diâmetro é, portanto, um sinal sobre a capacidade da informação se difundir na rede ou não.

2.2 Armazenamento Digital de Grafos

Existe uma diversidade de formatos de ficheiros para armazenamento de grafos. Cada formato tem características que se adequam melhor a certo tipo de redes, contudo não existe um formato standard e que seja melhor do que todos os outros. A escolha do formato de armazenamento do grafo é tipicamente feita tendo em conta as características da rede que se pretende estudar, e as perguntas para as quais se pretende ter resposta.

2.2.1 Matriz de adjacências

Uma forma primária de armazenar dados relacionais é numa matriz de [60]. Um grafo é representado numa matriz $n \times n$, na qual n representa o número de nós no grafo.

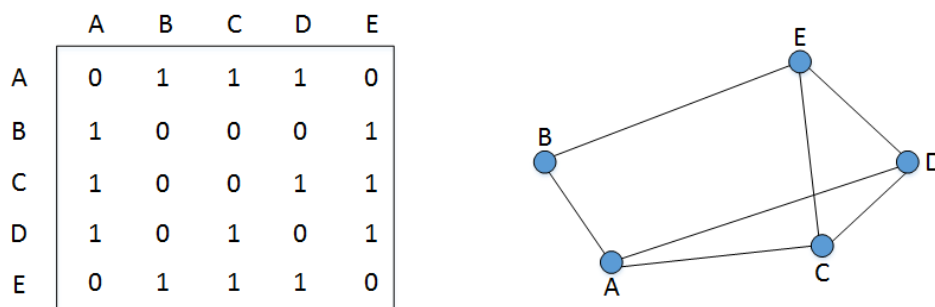


Figura 2.5: Matriz de adjacências e respetivo grafo

A figura 2.5 representa um grafo não direcionado e sem pesos e a sua matriz de adjacências correspondente. Neste tipo de grafos as entradas “0” e “1” das células da matriz representam a presença ou ausência de uma relação, ou seja, a presença ou ausência de uma linha entre dois nós no grafo. Adicionalmente, os elementos na diagonal na matriz são sempre iguais ou têm o valor “0”, o que significa que os nós não se ligam a eles próprios. Em grafos não direcionados as matrizes são simétricas ao longo da diagonal principal. O mesmo não se reflete em grafos direcionados. Por exemplo, num grafo não direcionado sem pesos, o valor “1” da entrada a_{ij} significa que existe uma aresta do i para o nó j . Relativamente ao valor das células da matriz, ao invés de conter o valor “1”, passam a conter o valor do peso da aresta.

Usar este método de armazenamento também traz algumas desvantagens, como o facto de ser difícil inserir dados manualmente, e caso se represente redes de grande dimensão, o espaço em memória para guardar esta estrutura pode ser bastante elevado.

2.2.2 Lista de arestas

Uma lista de arestas, ou “edgelist”, é outra forma primária de armazenamento de dados em grafos [25]. Este método apenas captura informação sobre as arestas, logo necessita de ser complementado com conhecimento sobre o número total de nós na rede (mesmo que estes não tenham qualquer ligação). Esta forma de armazenamento é mais eficiente para guardar informação sobre dados que são colecionados manualmente, contudo é necessário ter o cuidado de usar um sistema de nomes comum e guardar registo dos nós que não possuem qualquer ligação.

2.2.3 Formatos JSON

JavaScript Object Notation (**JSON**) é um formato muito popular, principalmente entre bibliotecas JavaScript para visualização e análise de grafos. D3.js, Cytoscape.js ou Sigma.js são exemplos de bibliotecas JavaScript que fazem *parsing* de ficheiros **JSON**, e cada uma aceita uma derivação de ficheiro JSON diferente. Usualmente, o ficheiro **JSON** possui duas chaves denominadas “nodes” e “edges”, na qual possuem como valor um array contendo a lista dos nós e a lista das arestas do grafo. A lista de arestas contém o nó de origem e o nó de destino da relação. Tanto o array de nós como o array de arestas podem conter atributos adicionais, tal como o valor do peso da aresta ou o cluster a que cada nó pertence. Existem muitos outros formatos baseados em **JSON**, entre os quais os mais usados são o GraphJSON e JSON Graph Format (**JGF**).

2.2.4 Formatos XML

O eXtensible Markup Language (**XML**) serve de base para inúmeros formatos, entre os quais os mais populares são o Graph Exchange XML Format (**GEXF**) e o Graph Markup Language (**GraphML**). **GEXF** foi desenvolvido no projeto Gephi, a partir de 2007. A sua linguagem permite descrever estruturas de redes complexas, os seus dados associados e dinâmicas [2]. Esta implementação não suporta grafos mistos (arestas direcionadas e não direcionadas juntas).

GraphML é um formato compreensivo e fácil de usar. Consiste numa linguagem core para descrever as propriedades estruturais de um grafo e um mecanismo flexível para adicionar dados específicos da aplicação, permitindo adição de atributos para os nós e arestas [20]. Ao contrário de outros formatos, o **GraphML** não tem uma sintaxe predefinida e, como tal, beneficia de uma arquitetura flexível. Tal como o **GEXF**, o **GraphML** não suporta grafos mistos, mas suporta grafos hierárquicos.

2.2.5 GML

Graph Modeling Language (**GML**) é um formato que garante portabilidade, uma sintaxe simples, extensibilidade e flexibilidade [41]. Um ficheiro **GML** consiste numa lista hierárquica de chave-valor. O **GML** é um formato bastante popular, levando a vários softwares a criar a sua própria

versão baseada e adaptada deste formato.

2.2.6 CSV

O formato Comma-Separated Values (**CSV**) é suportado pela maioria dos softwares para visualização e análise de grafos, e pode ser exportado de qualquer base de dados ou folha de cálculo Excel. Neste formato cada linha deve conter pelo menos dois elementos, separados por um separador (vírgula, ponto e vírgula, espaço ou espaço em branco). Os dados podem ser encapsulados por pelicas ou aspas. Em **CSV** podemos representar um grafo de diversas formas: lista de nós, lista de adjacências, grafo misto ou por uma matriz [1].

2.2.7 Outros Formatos

Com a proliferação de softwares para visualização e análise de grafos e redes, cada um tende a criar o seu próprio formato de armazenamento. Alguns softwares, tal como o Gephi ou o NetworkX, permitem importar e exportar grafos em formatos de outros softwares populares. Alguns exemplos destes formatos são:

- GDF (GUESS)
- Pajek NET Format (Pajek)
- TLP (Tulip)
- Netdraw VNA Format (Netdraw)
- DOT (GraphViz)
- UCINET DL (UCINET)

Um grafo também pode ser armazenado numa base de dados. Algumas das bases de dados mais populares, especializadas no armazenamento de grafos são: Oracle Spatial and Graph [12], ArangoDB [4] e Neo4j [11].

2.3 Métricas de SNA

Esta secção descreve as métricas mais usadas e mais importantes em **SNA**, como a densidade e as diferentes medidas de centralidade e centralização. Com as diferentes métricas de **SNA** podemos exprimir os relacionamentos e as dinâmicas da rede, e estudar as suas propriedades estruturais. Estes conceitos exigem uma explicação teórica de modo a percebermos, por exemplo, porque é que certos agentes praticam certas ações para aceder a fontes de informação, bem como a própria informação se difunde na rede.

2.3.1 Densidade

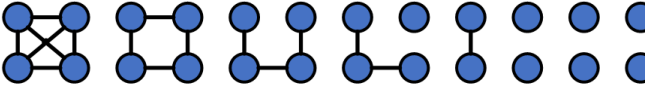
A densidade é um dos conceitos mais usados em teoria de grafos. Um grafo “completo” é aquele no qual todos os nós são adjacentes uns dos outros, ou seja, cada nó está ligado diretamente a cada outro nó. O conceito de densidade é uma tentativa de sumarizar a distribuição geral das arestas, de modo a medir o quão longe o grafo está deste estado de grafo “completo” [60]. Quanto maior o número de nós ligados entre si, mais denso será o grafo.

De modo a calcular a densidade, temos de ter em conta dois parâmetros da estrutura da rede: a inclusividade do grafo e a soma dos graus dos seus nós. A inclusividade refere-se ao número de nós dentro das várias partes conexas do grafo, por outras palavras, é o número total de nós, menos o número de nós isolados. Se tomarmos por exemplo um grafo de 20 nós com 5 nós isolados, a inclusividade do grafo é 0.75. Um nó isolado é uma incidência sem arestas, logo em nada contribui para a densidade do grafo. Por outro lado, os nós que estão ligados a outros vão variar no seu grau de conexão e quanto maior os graus dos nós no grafo, mais denso ele irá ser.

O número atual de arestas num grafo é uma reflexão direta da sua inclusividade e dos graus dos seus nós, pois o número de arestas em qualquer grafo é igual à metade da soma dos graus. Cada nó pode ligar-se a todos exceto um outro nó (ele mesmo), logo um grafo não direcionado com n nós pode conter um máximo de $n(n-1)/2$ arestas distintas. Ao calcular $n(n-1)$ teremos o número total de pares de nós num grafo, mas o número de arestas a ligar estes pares é metade deste valor, uma vez que a linha que liga A a B é a mesma linha que liga B a A. Assim, podemos concluir que a densidade de um grafo é definida como o número de arestas num grafo, representado pela variável l da fórmula, expresso como a proporção do máximo número possível de arestas:

$$\frac{l}{n(n-1)/2} \quad (2.1)$$

Esta medida pode variar de 0 a 1, em que 1 é a densidade de um grafo completo. Na figura 2.6 cada grafo contém 4 nós, logo pode conter um máximo de 6 arestas. Podemos observar como a densidade varia com a inclusividade e a soma dos graus.



Nº de nós ligados	4	4	4	3	2	0
Inclusividade	1.0	1.0	1.0	0.7	0.5	0
Soma dos graus	12	8	6	4	2	0
Nº de linhas	6	4	3	2	1	0
Densidade	1.0	0.7	0.5	0.3	0.1	0

Figura 2.6: Comparações de densidade

Em grafos dirigidos o cálculo da densidade é ligeiramente diferente. A matriz para dados

direcionados é assimétrica, uma vez que uma linha direcionada de A para B não envolve necessariamente uma linha recíproca direcionada de B para A. Assim, o número máximo de arestas que podem estar presentes num grafo dirigido é igual ao número total de pares que ele contém, o que é simplesmente calculado por $n(n - 1)$. Deste modo, a fórmula da densidade para um grafo dirigido é:

$$\frac{l}{n(n - 1)} \quad (2.2)$$

2.3.2 Medidas de centralidade

A ideia de medir a centralidade dos indivíduos e organizações para as suas redes sociais foi uma das ideias iniciais a ser seguida na análise de redes sociais. A origem imediata desta ideia vem do conceito sociométrico de “estrela”: a pessoa mais popular que está no centro do seu grupo. O que une a maioria das abordagens à centralidade é a ideia da centralidade relativa dos vários nós no grafo. Dentro deste conceito, existe uma distinção importante entre centralidade local e global. Um ponto diz-se local centralmente se tem um grande número de conexões com outros nós no seu ambiente imediato, ou seja, se tem uma grande vizinhança nos seus contactos diretos. Um ponto é globalmente central, por outro lado, quando tem uma posição de importância significativa na estrutura global da rede.

2.3.2.1 Centralidade local

O método mais direto e simples de medir a centralidade de um nó é através do seu grau, que é simplesmente o número de outros nós ao qual o nó é diretamente adjacente. Logo, quanto maior o grau do nó mais central ele é. No caso de a rede ser direcionada, o grau de um nó é composto por duas componentes distintas: o *in-degree* e o *out-degree*. Deste modo, faz sentido distinguir entre *in-centrality* e *out-centrality*. O grau de um nó depende de vários fatores, entre os quais o tamanho do grafo. Deste modo, as comparações dos resultados de centralidade apenas têm significado entre membros do mesmo grafo ou entre grafos do mesmo tamanho.

2.3.2.2 Centralidade global

A centralidade global é, contudo, apenas uma maneira de calcular a centralidade de um ponto. Freeman [33],[34] propôs uma medida para calcular a centralidade global baseada no termo *closeness* dos nós. Esta medida é expressa em termos das distâncias entre os vários nós. Recorde-se que dois nós estão unidos por um caminho se existe uma sequência de arestas distintas a liga-los, e o tamanho do caminho é medido pelo número de arestas que ele atravessa. Um ponto diz-se globalmente central se estiver à menor distância de todos os outros nós. Tal ponto está, então, próximo (*close*) de muitos outros nós no grafo.

2.3.2.3 Eigenvector centrality

Eigenvector centrality mede a influência global de um nó na rede. É uma medida mais sofisticada do que o grau de um nó, uma vez que o seu cálculo depende não só do número de arestas incidentes, mas também da qualidade dessas arestas [16]. Para o seu cálculo, são atribuídas pontuações relativas a todos os nós da rede baseando-se no conceito de que arestas para nós com pontuações elevadas contribuem mais para a pontuação do nó em questão. Esta medida de centralidade varia entre 0 e 1 e tira partido das propriedades matemáticas de matrizes de adjacências para o cálculo do valor para cada nó. Katz Centrality [45] e o algoritmo PageRank [54], desenvolvido pelos criadores do Google, são duas variantes da Eigenvector Centrality.

2.3.2.4 Closeness centrality

A noção mais simples de *closeness* é, talvez, através do cálculo da soma das distâncias geodésicas de um nó para todos os outros nós no grafo [60]. Se a matriz das distâncias entre nós de um grafo não direcionado é calculada, a soma das distâncias de um ponto é a soma da sua linha ou coluna na matriz, os dois valores são idênticos. Um ponto com a menor distância é *close* para o maior número de nós, e deste modo *closeness* pode ser vista como o recíproco/inverso da soma das distâncias geodésicas. Num grafo direcionado, os caminhos devem ser medidos através das arestas que correm na mesma direção, e, por esta razão, os cálculos baseados na linha ou coluna serão diferentes. Assim, *closeness* num grafo direcionado considera os termos *in-closeness* e *out-closeness*, para as arestas que entram e saem de um nó, respetivamente. Para uma rede com n nós, a *closeness* pode ser representada matematicamente por [36]:

$$C_c(n_j) = \frac{n - 1}{\sum_{k=i, j=k}^n d(n_i, n_j)} \quad (2.3)$$

Onde $C_c n_k$ define a *closeness centrality* standardizada do nó j e $d(n_i, n_j)$ define a distância geodésica entre j e k .

2.3.2.5 Betweenness centrality

Freeman adicionou outro conceito de centralidade global chamado *betweenness*, que mede a distância a que um determinado nó se situa entre todos os outros nós no grafo [32]. Um ponto com um grau elevado pode ter um papel importante de intermediário e como tal ser bastante central para a rede [60].

A *betweenness* de um ponto mede a extensão para o qual um agente pode desempenhar o papel de *broke* ou *gatekeeper* com um potencial para controlar os outros. A abordagem de Freeman é construída à volta do conceito de dependência local. Um ponto é dependente de outro se o caminho que o liga a outros nós passa por ele. O conceito de “buraco estrutural” existe quando dois nós estão ligados por uma distância de 2, ou então separados por um longo caminho. O terceiro

ponto pode atuar como intermediário que liga o buraco estrutural. *Betweenness*, intuitivamente, mede o grau para o qual a informação ou o relacionamento tem de fluir através de um ator particular e a sua importância relativa como um intermediário na rede. Metodologicamente, *betweenness* é a medida da centralidade mais complexa de calcular. A *betweenness* dos nós de uma rede pode ser definida por [36]:

$$C_b(n_j) = \frac{xx}{\frac{(n-2)(n-1)}{2}} \quad (2.4)$$

$$xx = \sum_{k < i, j = k, j = t} \frac{g_{kt}(n_j)}{g_{kt}} \quad (2.5)$$

Onde $C_b(n_j)$ representa a *betweenness centrality* standardizada do nó j , $g_{kt}(n_j)$ representa o número de ligações geodésicas entre k e t e contém j no meio.

2.3.2.6 Brokerage Centrality

Brokerage descreve a posição dos nós de modo a que eles ocupem posições vantajosas onde podem intermediar as interações entre outros nós na rede. *Brokerage centrality* é então uma medida do grau para o qual um ator ocupa uma posição de intermediário ou *broker* ao longo de todos os pares de nós e, como tal, obtém benefícios da sua posição como intermediário [25]. Existem normalmente 5 tipos de relações que um *broker* pode ter, ilustradas na figura 2.7 e descritas brevemente abaixo num contexto pedagógico:

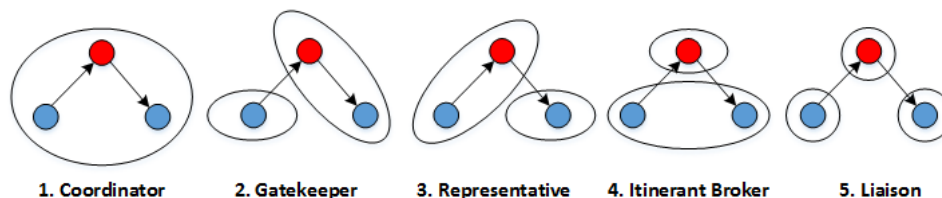


Figura 2.7: Diferentes tipos de relações Brokerage

- *Coordinator*: é um estudante que está no mesmo grupo de outros dois estudantes, e que liga estes dois. Um exemplo de um *coordinator* pode ser um estudante que garante que o resto do seu grupo está ciente de uma reunião para tirar dúvidas com o professor que um estudante do grupo está a organizar;
- *Gatekeeper*: é um estudante de um grupo que atua como alvo de um estudante fora do grupo, que deseja ligar-se a ele. Cabe a este estudante decidir se o estudante *outsider* ganha acesso ou não ao grupo;
- *Representative*: é um estudante do mesmo grupo que atua como um estudante que deseja conectar-se a um outro estudante fora do grupo, mas para isso tem de passar por um

intermediário. Um exemplo disto é um representante da associação de estudantes de um curso;

- *Itenerant broker*: é um estudante membro de um grupo *outside* que conecta dois outros estudantes que partilham o mesmo grupo;
- *Liason*: é um estudante membro de um grupo que é distante de dois estudantes a que se pretende ligar, mas eles próprios não partilham o mesmo grupo.

2.3.3 Centralização

A medida de centralização refere-se à centralidade do grafo e não à centralidade de nós específicos [60]. Implícita à ideia de centralização é a ideia de centro estrutural do grafo, o ponto ou conjunto de nós no qual o grafo centralizado está organizado. Para além de nos concentrarmos apenas na centralidade de nós particulares, podemos também examinar a distância para o qual um grafo como um todo tem uma estrutura centralizada.

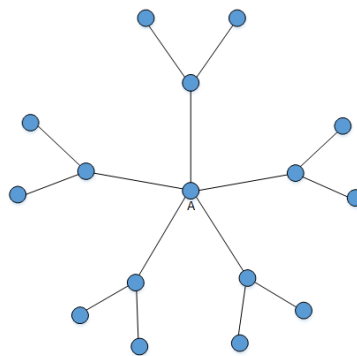


Figura 2.8: Grafo altamente centralizado

A densidade descreve o nível geral de coesão num grafo; a centralização descreve a extensão para o qual essa coesão está organizada a volta de nós particulares. A centralização e a densidade são duas medidas complementares importantes. Enquanto uma rede centralizada terá muitas das suas arestas dispersas à volta de um ou poucos mais nós, uma rede descentralizada é aquela que tem pouca variação entre as arestas que cada nó possui. O grafo da figura 2.8 é considerado altamente centralizado.

A figura 2.9 faz referência à disposição dos nós na rede, mediante o nível de centralização que esta possui. Um grafo que possua um valor de centralização elevado, possui a maioria dos seus nós no centro da rede, enquanto que um grafo com um valor de centralização reduzido possui os seus nós dispersos na rede ou na sua periferia.

A centralização varia entre 0 e 1. Zero corresponde a uma rede na qual todos os nós estão ligados a todos os outros nós, enquanto que o valor 1 representa redes idênticas a uma rede estrela, como o grafo da figura 2.8. A centralização de um grafo é calculada pela seguinte fórmula [36]:

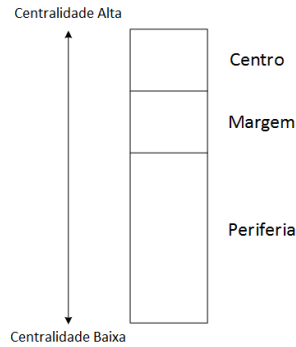


Figura 2.9: Centro estruturado de um grafo

$$R = \frac{\sum_{j=1}^g \max(D_i) - D_i}{(g - 1)^2} \quad (2.6)$$

em que D_i representa o número de nós na rede que estão diretamente ligados ao ator j e g é denotado como o número de nós presente na rede.

2.3.4 Coeficiente de agrupamento

O coeficiente de agrupamento (“*Clustering Coefficient*”) mede o grau para o qual os nós formam ligações de forma densa, formando grupos altamente coesos na rede. O grau de *clustering* numa rede está relacionado com a eficiência da difusão da informação através da rede, assim como a sua robustez à rutura [25]. Para computar o coeficiente de *clustering* de um dado grafo G , é necessário fazer a média de todos os nós, através da seguinte fórmula [27]:

$$cc_1(v) = \frac{\text{n}^\circ \text{ de pares de vizinhos ligados por arestas}}{\text{N}^\circ \text{ de pares de vizinhos}} \quad (2.7)$$

2.4 Estruturas e Grupos na rede

Um grupo numa rede define um subconjunto de nós que partilham algumas características em comum. A definição de grupos na rede permite-nos aplicar medidas úteis, a fim de obter informações relevantes sobre os membros e os seus relacionamentos.

2.4.1 Cliques

O termo clique foi criado por Luce e Perry em 1949 [46] que utilizaram sub-grafos completos em redes sociais para modelar cliques de pessoas, ou seja, grupo de pessoas onde todas se conhecem. Luce e Perry definiram formalmente que um clique é um sub-grafo máximo completo. Completo significa que cada nó no clique é adjacente a todos os outros. Máximo significa que não podemos

aumentar o seu tamanho e continuar a ter um clique completo. Esta definição é a mesma para ambas as redes direcionadas ou não-direcionadas. Por norma, considera-se que um clique tem pelo menos três nós, uma vez que normalmente não se considera um par como um grupo. Podemos ilustrar a ideia de clique ao examinar a rede não direcionada da figura 2.10.

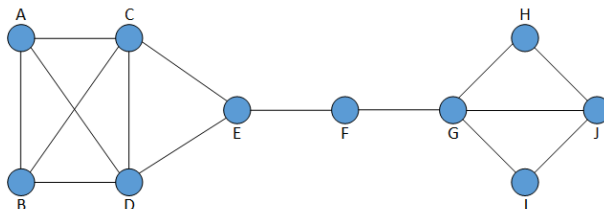


Figura 2.10: Uma rede com quatro cliques

Os nós A, B, C e D estão todos ligados uns aos outros. Além disso, não podemos aumentar este grupo e manter esta propriedade. Podemos concluir que $\{A, B, C, D\}$ formam um clique. Outros cliques são $\{C, D, E\}$, $\{G, H, J\}$ e $\{G, I, J\}$. De notar que $\{A, B, C\}$ não é um clique porque não é máximo (podemos adicionar o nó D). Os cliques podem sobrepor-se de modo a que um nó individual possa estar em mais do que um clique. Neste exemplo, os nós C, D, G e J estão todos em dois cliques. Finalmente, pode haver nós que não pertencem a qualquer clique, como é o caso do nó F por exemplo.

Desde a criação do conceito original de clique foram desenvolvidas algumas variantes baseadas no relaxamento do conceito original de clique, a fim de tornar o conceito mais genérico e útil, como é o caso do N-clique, o N-clan e o K-plex.

2.4.2 Componentes, Cut Points e Bridges

Um componente é definido como o conjunto máximo de nós no qual cada nó pode alcançar qualquer outro nó através de algum caminho [19]. Ser máximo significa que se conseguirmos adicionar um nó ao conjunto sem violar a condição que cada nó pode alcançar qualquer outro nó, podemos fazê-lo. Em grafos direcionados, em contraste com grafos não direcionadas, podemos definir dois tipos de componentes. Um componente “fraco” consiste num conjunto de nós que são conexos, independentemente da direção das arestas. Um componente “forte”, por outro lado, requer a existência de um caminho direcionado de A para B, de modo a que estes nós estejam no mesmo componente [38].

Outra abordagem para encontrar nós fracos no grafo consiste em perguntar: Se um nó for removido, será que a estrutura fica dividida em parte não conexas? Caso exista tais nós, eles chamam-se *cutpoints* [38]. Estes nós são particularmente importantes, uma vez que atuam como intermediários entre grupos, que de outra forma se tornaria desconexas. As divisões na qual os *cutpoints* dividem o grafo chamam-se *blocks* [38].

O mesmo pensamento aplica-se a arestas que, se forem retiradas, transformam o grafo em partes desconexas. Tais arestas denominam-se *bridges* [38]. Em muitos casos, os *cutpoints* estão

nas extremidades de uma *bridge*, mas é possível haver *cutpoint* que não fazem parte de uma *bridge*.

2.5 Visualização de grafos

O foco da visualização de grafos consiste em permitir os utilizadores terem uma visão da complexidade da dinâmica social [24].

2.5.1 Medidas de qualidade e estética

Para esse efeito, são precisas algumas propriedades de alto nível que os diferentes layouts de visualização de grafos devem satisfazer:

1. A posição dos nós e das arestas no diagrama deve facilitar a leitura da rede
2. A posição dos nós deve ajudar a descobrir as propriedades de clustering da rede, isto é, se um certo grupo de nós é considerado ser um cluster, o utilizador deve conseguir extrair essa informação do layout
3. A posição dos nós deve resultar numa representação confiável da rede social

Segundo Narkhede et al (2014): “a estética, no contexto de layouts de grafos, consiste nos parâmetros que decidem a complexidade e a usabilidade do layout na visualização” [51]. De modo a satisfazer estas propriedades de alto nível, os algoritmos de layout são por vezes formulados em termos de propriedades de baixo nível. H. C. Purchase [55] considera 5 propriedades comuns em layouts de grafos:

- Dobras: o número total de dobras nas arestas deve ser mínimo. Quando uma aresta não é desenhada por uma linha reta, diz-se que esta apresenta uma dobra. Quanto menor for o número de dobras maior será a compreensibilidade do grafo
- Cruzamentos: o número de cruzamento de arestas deve ser mínimo, pelo mesmo motivo anterior
- Ângulos: os ângulos mínimos entre as arestas que saem dos nós no desenho do grafo devem ser maximizados, de modo a aumentar a visualização do grafo
- Ortogonalidade: fixar os nós e as arestas de um grafo numa grelha ortogonal aumenta a compreensibilidade do grafo
- Simetria: sempre que possível, deve ser desenhada uma vista simétrica do grafo, que torna o layout mais compreensivo

2.5.2 Layouts

O layout de um grafo é um dos aspetos mais relevantes em visualização de grafos, e refere-se à posição que os nós ocupam no diagrama do grafo. Um mau layout pode induzir erros de interpretação por parte do utilizador ou pode transmitir pouca informação sobre o grafo e as suas propriedades [19].

2.5.2.1 Force-Directed Layout

Os *Force-Directed layouts* são dos algoritmos mais populares e importantes em visualização de grafos. São bastante flexíveis e podem ser usados para calcular o layout de todos os grafos não direcionados simples. Calculam o layout do grafo baseando-se apenas em informação contida na própria estrutura do grafo. Os grafos desenhados por estes algoritmos tendem a ser esteticamente agradáveis, exibindo simetrias e produzem layouts com cruzamento mínimo de arestas [68].

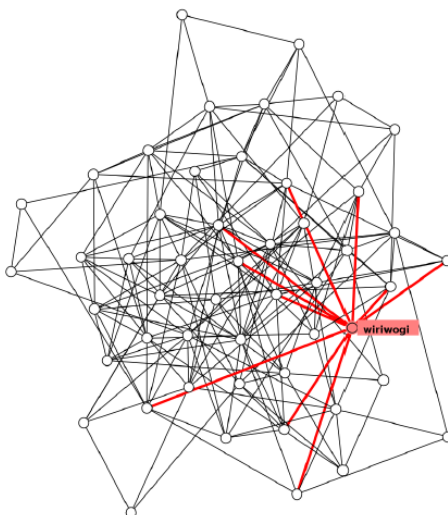


Figura 2.11: Force-Directed layout

O seu algoritmo consiste na aplicação repetida de um modelo de forças, resultando num grafo que atinge um estado de energia mínima, na qual o grafo é considerado estar bem desenhado. Estes métodos tem a vantagem de poderem ser aplicados em grafos gerais, sem estruturas específicas ou considerações topológicas.

Existem muitos algoritmos baseados em forças, sendo o Barycentric Method de Tutte's considerado o primeiro [67]. Neste algoritmo, os nós são desenhados à distancia média dos seus vizinhos, denominados os seus *barycenter's*. Este método é repetido numa série de iterações até os nós pararem de se mover, atingindo o estado de energia mínima. Um subconjunto dos nós deve ser fixo, para indicar fronteiras externas do espaço de desenho, caso contrário os nós vão colapsar num ponto único. Este método é eficaz para grafos esparsos e simétricos, em que nós fixos são facilmente definidos. O algoritmo Spring Embedder de Eades [26], é talvez o método para visualização de grafos mais conhecido e popular. Outros algoritmos de layout populares

incluem o algoritmo de Kamada and Kawai [43] e de Fruchterman-Reingold [35], ambos baseados na ideia original de Eades.

2.5.2.2 Planar Layout

Os layouts planares produzem grafos visualmente agradáveis, sem cruzamento de arestas, e esta é a definição de um layout planar [58]. Um método comum para desenhar grafos planares consiste na ordenação dos nós, colocando os primeiros três nós nos vértices de um triângulo e posteriormente adicionar os nós de ordem inferior em posições dentro desse triângulo, produzindo grafos com aspecto semelhante ao grafo da figura 2.12. Este método nem sempre produz um layout satisfatório, mas pode ser melhorado se for conjugado com um algoritmo de *Force-Directed layout*, tal como o String Embedder. Os layouts planares podem produzir grafos com diferentes arquiteturas itopológicas [52], como por exemplo grafos de aspecto triangular, retangular, ortogonal, grelha, convexo ou grafos com arestas retas ou dobradas.

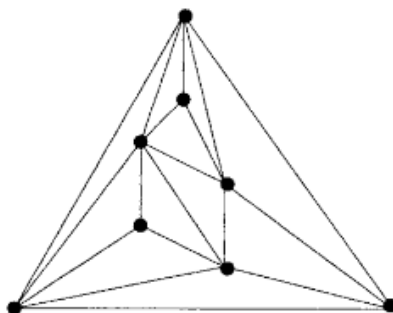


Figura 2.12: Layout planar

No entanto, não é possível aplicar um layout planar em todos os grafos, como demonstra o teorema de Kuratowski [66], contudo é possível planarizar grafos.

2.5.2.3 Layout Circular

A figura 2.13 retrata uma rede posicionando os nós numa circunferência, formando um diagrama circular [47]. As arestas podem ser desenhadas tanto em linha reta como em curva, mas a segunda opção normalmente aumenta a legibilidade do diagrama. Este layout requer a ordenação dos nós de modo a que o cruzamento de arestas seja mínimo. Como é necessário ordenar os nós numa circunferência, a área para os nós se posicionarem é limitada. Por estes motivos, este layout é aconselhado apenas em redes de tamanho pequeno a médio.

2.5.2.4 Layout em Arco

Por vezes é útil desenhar os nós de uma rede numa linha reta, processo usualmente chamado linearização. As arestas podem ser desenhadas como arcos circulares, produzindo um diagrama

2.5.2.5 Layout em Árvore

O layout em árvore é um layout muito estudado na área de visualização de grafos e contém as tem diversas aplicações práticas [68]. Todas as árvores são planares, o que significa que é possível construir o grafo sem cruzamento de arestas. Existem varias estratégias eficientes em relação ao tempo de desenho de um layout em árvore, que permitem a criação de grafos esteticamente agradáveis. Este tipo de layout pode ser aplicado em grafos direcionados que representam hierarquias. A principal ideia por trás destes layouts consiste em modificar métodos *Force-Directed* de modo a ter em conta a direção das arestas, e usá-los para desenhar digrafos [68]. Os nós em grafos hierárquicos podem ser colocados em camadas, com os primeiros nós denominados raízes na camada superior, e os restantes nós abaixo, com os nós folha na última camada, como ilustra o grafo da figura 2.15.

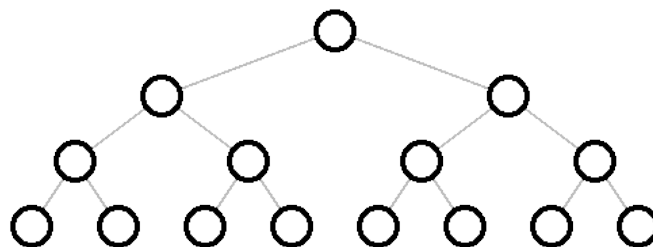


Figura 2.15: Layout em árvore

É conveniente aplicar este layout em grafos acíclicos, caso contrário os nós que participam num ciclo têm de estar ordenados. Os nós são dispostos em camadas e posteriormente a sua posição relativa é atribuída. Existem várias abordagens, mas a maioria é pensada para reduzir o cruzamento de arestas e arestas dobradas [58].

2.5.2.6 Layout em Matriz

Nesta visualização, cada relação entre dois nós no grafo é representada como uma localização numa grelha com coordenadas cartesianas. As representações visuais das matrizes de adjacências ilustram propriedades semânticas e estruturais do grafo, numa representação mais compacta do que a maioria dos diagramas com nós e arestas, e não apresenta sobreposições. Um dos desafios das representações baseadas em matrizes é identificar visualmente estruturas locais e globais da rede. Na figura 2.16, estão ilustradas duas representações gráficas do mesmo grafo, à direita um diagrama baseado em nós e arestas e à esquerda um diagrama baseado numa matriz de adjacências. É visível a relação entre os dois diagramas, na qual as estruturas presentes no grafo da direita, os diferentes cliques e clusters, são facilmente identificadas na matriz à esquerda.

No geral, este processo depende da forma como os nós estão ordenados ao longo dos eixos da matriz. Quando a ordem é arbitrária, a visualização da matriz pode não exibir os clusters presente nos dados. Reordenar a matriz é um problema combinatorial que pode ser complexo, do ponto de vista computacional.

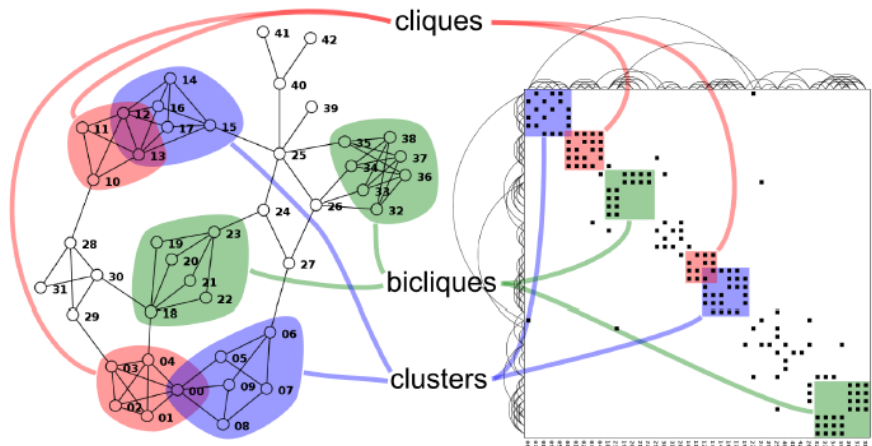


Figura 2.16: Layout em matriz

2.5.2.7 Layout Híbrido

Os grafos baseados em nós e arestas são fáceis de interpretar quando comparados com diagramas de matrizes de adjacências, mas estes últimos são normalmente mais eficazes a mostrar estruturas em redes dessas e de grandes dimensões [24]. Esta observação levou ao desenvolvimento de técnicas híbridas que combinam dois layouts num só, de modo a obter minimizar as desvantagens de cada um e obter um layout mais rico e completo. A figura 2.17 representa o MatLink, um layout híbrido que combina um layout baseado numa matriz com um layout em arco.

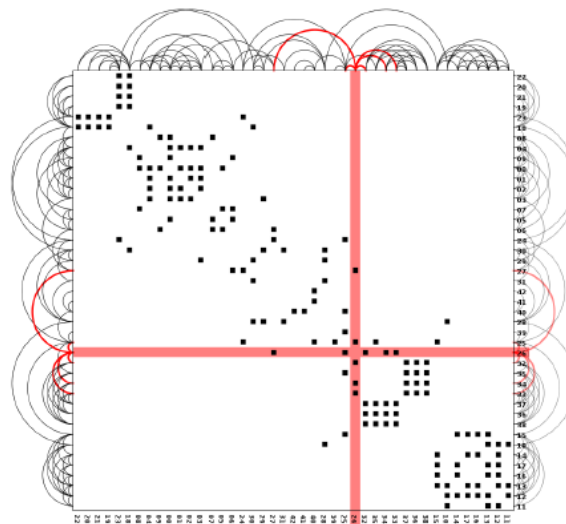


Figura 2.17: MatLink

Muelder e Ma [50], Shen e Ma [62], MatLink [39] de Henry e Fekete da figura 2.17 e Node-Tric [40] de Henry et al., são algumas técnicas híbridas populares desenvolvidas recentemente.

2.5.3 Visualização semântica

Ao gerar diagramas de rede, por vezes queremos incorporar informações adicionais como atributos dos nós. Tipicamente, isto é conseguido ao mapear esses atributos dos nós em diferentes propriedades visuais dos nós, de modo a comunicar informações relevantes sobre os nós que representam [19]. Por exemplo, a quantidade de vizinhos de cada nó pode ser representada com nós de diferentes tamanhos. Os nós podem igualmente ter diferentes formas e cores. Associados aos nós também podem ser usadas etiquetas textuais, que também podem ter diferentes características. Concluindo, existe uma diversidade de atributos que podem ser aplicados simultaneamente aos nós num diagrama de rede. Contudo, usar muitas propriedades pode tornar a representação do grafo mais distrativa do que informativa.

Noutro ponto de vista, por vezes é útil ver como é que a rede fica se algumas classes de nós forem removidas, e as suas respetivas arestas. Por vezes isto é feito para remover nós que são periféricos a um dado ponto de interesse, outra vez para avaliar a importância na rede de um determinado nó ou conjunto de nós, como por exemplo no estudo de redes egocêntricas. Esta técnica é particularmente útil se quisermos comparar as estruturas a volta de dois egos diferentes. Principalmente em grafos grandes e complexos, a habilidade de filtrar nós pode ser bastante útil para descobrir propriedades estruturais e relacionais na rede.

Tal como os nós, as arestas têm um número de propriedades visuais que podem ser mapeadas para informações da rede [19]. Por exemplo, a espessura da aresta pode representar a sua força. Diferentes tipos de arestas podem ser representados com diferentes estilos, podemos representar linhas solidas ou tracejadas por exemplo, ou usar diferentes cores. Em grafos direcionados, usam-se normalmente setas nas extremidades das linhas de modo a representar a direção da aresta, que também podem ter diferentes tamanhos, cores ou formas. De igual modo aos nós, podemos filtrar as arestas do grafo para obter informações relevantes da rede.

2.5.4 Visualização temporal e mudanças na rede

O tempo é uma informação semântica que captou a atenção dos investigadores que estudam a visualização de redes sociais. Uma vez que as interações sociais são um fenómeno dependente do tempo, é natural representar esta dimensão temporal usando meios visuais [24]. Os dados da rede podem ser colecionados em múltiplos pontos no tempo. Podemos estudar e comparar as representações destes pontos temporais e aplicar todas as técnicas descritas anteriormente. Deste modo, é possível analisar as relações entre pontos temporais ao correlacionar as suas matrizes de adjacência e as suas respetivas representações visuais [19]. As figuras 2.18 e 2.19 representam a mesma rede em dois nós temporais distintos, mostrando apenas relacionamentos fortes.

Outro modo de visualizar mudanças ao longo do tempo é dar foco às mudanças de posições dos indivíduos na rede, em vez de nos preocuparmos com as mudanças na estrutura geral ao

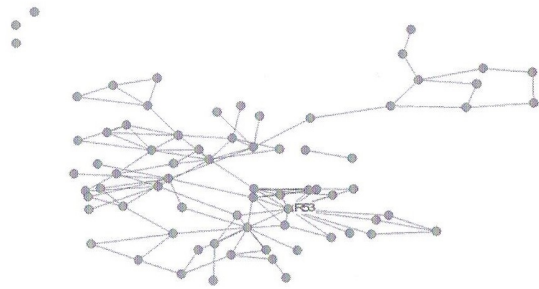


Figura 2.18: Relacionamentos fortes da rede no tempo 1

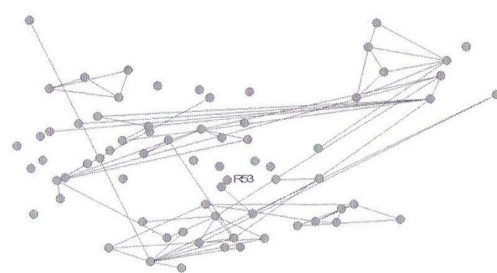


Figura 2.19: Relacionamentos fortes da rede no tempo 2

longo do tempo [19]. Isto pode ser realizado ao empilhar as matrizes da rede, e submeter esta pilha de matrizes a uma análise de correspondência. A figura 2.20 mostra as mudanças na posição dos indivíduos em dois períodos de tempo. Pela análise da figura podemos verificar a tendência dos indivíduos em moverem-se para mais perto de outros indivíduos ao longo do tempo, resultando num grupo mais coeso e aumentando a densidade da rede.

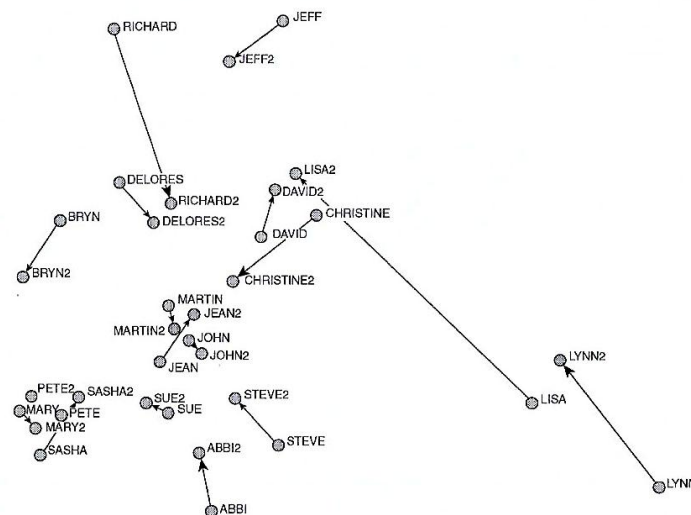


Figura 2.20: Grafo da mudança de posição dos nós na rede em dois períodos de tempo.

Representar o papel do tempo no processo de criação e crescimento de uma rede social apresenta diversos desafios, pelo que tem sido uma área de estudo muito importante nas últimas décadas. Moody et al. [49] considera estes dois tipos de visualizações dinâmicas referidas anteriormente: “flip books”, em que os nós permanecem estáticos enquanto as arestas variam ao longo do tempo, e “movies”, em que os nós podem mover-se caso as relações se alterem [48]. Shen et al. Apresentaram Mobivis, um cronograma interativo com uma visualização baseada em ontologia [63]. Este cronograma é uma metáfora baseada em pixels que incorpora informação temporal de indivíduos numa rede. Um dos desafios ao desenhar visualizações temporais é a incorporação de escalas possivelmente não apropriadas e da granularidade variável dos dados temporais. Mobivis visa resolver este problema através de filtragens interativas e com cronogramas

multi-escala. Viegas et al. propõe PostHistory, que ajuda a visualizar a rede social dinâmica que deriva de trocas de emails [69]. Esta visualização combina uma técnica orientada a pixeis, similar o cronograma presente em Mobivis, que mostra a quantidade e a relevância dos emails ao longo do tempo usando cores e tamanhos, respetivamente. Simultaneamente, a visualização mostra uma rede social inferida dos contactos de email de um modo hierárquico egocêntrico. Uma vez que a estrutura das arestas não é representada explicitamente, estas visualizações não sofrem de desordem e são bastante eficientes para redes dinâmicas. Noutro ponto de vista, podemos pensar em redes sociais como a conexão entre indivíduos e eventos. Esta perspetiva levou a abordagens como a descrita por Kang et al. [44], em que um grafo bipartido descreve as mudanças dinâmicas de um número de indivíduos em diferentes eventos ao longo do tempo. Outros estudos identificam clusters dentro do grafo e mostram modificações gerais dos clusters ao longo do tempo, com técnicas de *merging* e *splitting*, em vez de dar ênfase a informação topológica detalhada [59] [57], uma abordagem apropriada para grandes redes dinâmicas.

Podemos concluir que esta é uma área que está em constante crescimento, uma vez que as redes sociais são cada vez mais populares e de maiores dimensões, levantando novos desafios para entender o impacto e o efeito do tempo na evolução e nas mudanças estruturais destas estruturas.

2.5.5 Software e bibliotecas para visualização de grafos

Nos últimos anos houve uma explosão de softwares que incorporam ferramentas de visualização e análise de redes sociais. Entre os mais conhecidos e usados estão Pajek [18] [13], Gephi [8], Graphviz [9], Tulip [15] [17], Cytoscape [61] [5] visone. Existem também várias bibliotecas mais leves, que pretendem levar a visualização e análise de grafos para o brazer de uma forma simples e compacta. D3.js [7], JUNG [10], Sigma.js [14], Cytoscape.JS [6] e Alchemy.js [3] são alguns exemplos.

Muitas destas ferramentas foram desenhadas para análise de grafos no geral, e por vezes combinam diagramas de nós e arestas com visualizações de estatísticas standard, como histogramas e diagramas de dispersão. Apesar destas visualizações estruturais permanecerem populares, a tecnologia atual tornou a visualização mais sofisticada, abrangendo dimensões temporais e semânticas essenciais para entender as dinâmicas sociais. As visualizações tornaram-se também mais intuitivas, permitindo aos utilizadores formar hipóteses e validar inferências analíticas e visuais [24].

2.6 Resumo

No presente capítulo foi abordado em detalhe os conceitos teóricos por de trás da teoria de grafos que sustenta o desenvolvimento deste projeto, assim como as diferentes técnicas e métodos mais comuns em SNA. Com este panorama geral, podemos aferir as características que pretendemos para o nosso sistema e de que forma as pretendemos integrar.

O sistema proposto tem como foco principal a representação gráfica dos relacionamentos ou conteúdos de um ambiente virtual de ensino-aprendizagem, com a integração de técnicas de **SNA** para o estudo dessa rede. Pretendemos que a representação do Grafo Social seja: simples e agradável visualmente; flexível, sendo facilmente adaptável a diferentes dados de entrada, tipos de grafos diferentes e a critérios de estética adicionais; intuitiva; interativa e fácil de implementar. Para cumprir estes critérios, pretendemos utilizar um “Force-Directed’ layout. Uma das principais vantagens deste layout é o facto de produzir grafos esteticamente agradáveis e simétricos, com uma distribuição uniforme dos nós e do tamanho das arestas, com um cruzamento de arestas mínimo. Para redes de tamanho pequeno a médio, como é o caso de redes sociais em ambientes virtuais de ensino-aprendizagem, este tipo de layout tem um bom desempenho e obtém resultados de qualidade para os requerimentos propostos. Para isto, iremos usar como base o exemplo “Force-Directed Graph” da biblioteca JavaScript D3.js, que nos oferece grande parte dos objetivos proposto e, pelo facto de ser uma biblioteca *open-souce*, podemos adicionar facilmente novas funcionalidades e características ao Grafo Social.

Juntamente ao Grafo Social, pretendemos adicionar ao sistema as métricas mais comuns em **SNA**, com recurso à biblioteca JavaScript JSNetworkX.js. Deste modo, será possível aplicar ao Grafo Social diversas métricas, tais como a densidade, métricas de centralidade como o grau, Eigenvector e Betweenness Centrality, o coeficiente de Agrupamento, bem como a deteção de comunidades e o cálculo de cliques e de caminhos. Com a integração destes componentes no Learning Management System (**LMS**) Moodle, o utilizador pode obter um diagnóstico melhor e mais completo do seu ambiente virtual de ensino-aprendizagem.

Capítulo 3

Arquitetura do Sistema

Este capítulo descreve as tecnologias necessárias para o desenvolvimento deste trabalho, e as suas funções específicas no ciclo de execução do sistema.

3.1 Sistema Moodle

O trabalho desenvolvido tem como base o Learning Management System (LMS) Moodle. A figura 3.1 ilustra a arquitetura do sistema proposto, na qual o sistema Moodle é representado pelo diagrama do canto superior esquerdo da figura. O Moodle é um LMS open-source oferecido num pacote para instalação em ambientes Windows ou Linux. Este pacote é uma distribuição XAMPP, que contém o Moodle e vários softwares necessários para a sua execução, tal como o Apache, o MySQL e o PHP.

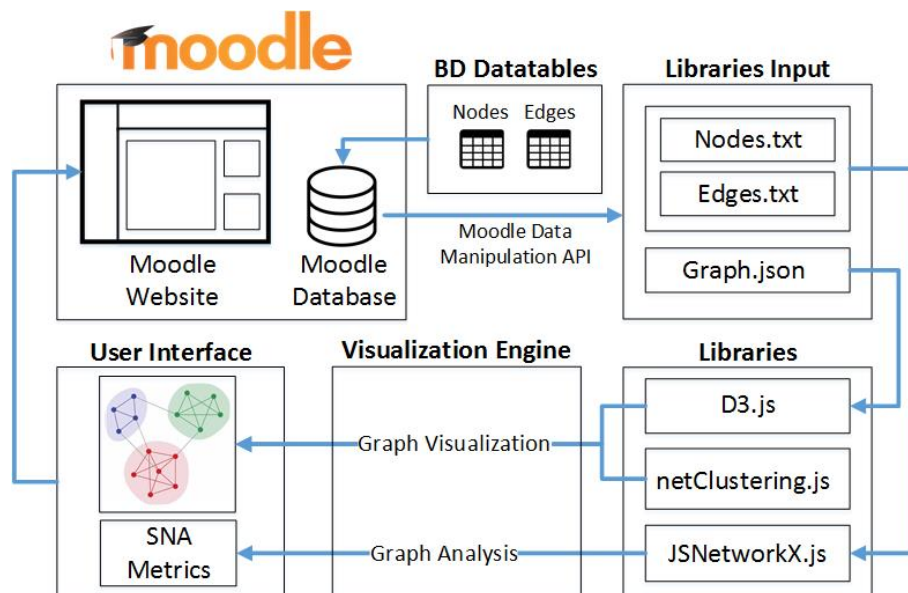


Figura 3.1: Arquitetura do Sistema

O Moodle apresenta uma interface simples e com uma navegação intuitiva, que consiste apenas em 3 colunas distintas, na qual os cursos criados pelo utilizador são apresentados na coluna do meio. Na página de um curso, o Moodle fornece uma visão simples e clara de todas as atividades de aprendizagem e recursos disponíveis. Adicionalmente, o utilizador tem ao seu dispor várias funcionalidades, tais como adição de novos tópicos e recursos educativos, pautas e informações de avaliação, diário de curso, registo da atividade dos alunos, submissão de conteúdos, entre outras.

Existem vários motivos que levam a que o Moodle seja um dos LMSs open-source mais usado em todo mundo, entre os quais o facto de o Moodle ser uma plataforma open-source, configurável, altamente flexível e rico em funcionalidades. O seu código fonte é acessível a todos, permitindo deste modo ser personalizado de modo a adicionar novas funcionalidades e elementos multimédia. Além disso, existem uma imensa variedade de *plugins* que permitem configurar o Moodle de modo a atingir todas as necessidades do utilizador.

Em resultado, professores, alunos e administradores tem a possibilidade de criar no Moodle ambientes virtuais de ensino-aprendizagem colaborativos, envolventes e divertidos, bem como moldar o sistema sempre que seja necessário atender novas necessidades pedagógicas.

3.2 Modelo de dados

Um grafo pode ser armazenado digitalmente com recurso diferentes formatos, cada um com as suas vantagens e desvantagens, tal como abordado na secção 2.2 (Armazenamento Digital de Grafos) do capítulo 2. No sistema proposto, o Grafo Social apresentado na página de um curso no Moodle é baseado no layout “Force-Directed Graph” da biblioteca D3.js que, por defeito, utiliza o formato JSON para armazenamento do grafo. O ficheiro JSON é composto por uma lista de nós e por uma lista de arestas, semelhante ao formato ilustrado no bloco de código 3.2.

De forma a definir o modelo de dados do sistema, para armazenar a lista dos nós e a lista das arestas que compõem o Grafo Social, foram criadas duas tabelas na base de dados do Moodle, com o auxílio do *plugin* Moodle Adminer. As duas tabelas criadas: “moodle__moodlesna_nodes” e “moodle__moodlesna_links”, contém todas as informações do Grafo Social apresentado na página de um curso no Moodle. A primeira tabela contém informação sobre os nós do grafo e a segunda tabela contém informação sobre as suas arestas. A tabela 3.1 e a tabela 3.2 apresentam a estrutura das duas tabelas.

A tabela “moodle__moodlesna_nodes” guarda informação sobre o ID de cada nó, o seu nome, grupo a que pertence e o ID do curso à qual está associado. A tabela “moodle__moodlesna_links”, por outro lado, guarda informação de cada aresta, como o seu ID, nome do seu nó inicial e nó final, seu o peso e o ID do curso do Moodle à qual a aresta está associada.

Quando um curso é criado no Moodle, é-lhe atribuído um ID único. Os dados do seu Grafo Social devem ser armazenados na tabela dos nós e na tabela das arestas, e a coluna “course_id”

Tabela 3.1: Estrutura da tabela “moodle__moodlesna_nodes”

Coluna	Tipo	Descrição
id	int(11) Incremento Automático Chave Primária	ID do nó
node_name	varchar(20) null	Nome do nó
node_group	decimal(11) null	Cluster do nó
course_id	int(11) null	ID do curso do Moodle

Tabela 3.2: Estrutura da tabela “moodle__moodlesna_links”

Coluna	Tipo	Descrição
id	int(11) Incremento Automático Chave Primária	ID da aresta
source_node	varchar(20) null	Nome do nó inicial
target_node	varchar(20) null	Nome do nó final
value	decimal(11) null	Peso da aresta
course_id	int(11) null	ID do curso do Moodle

presente nas duas tabelas deve ser preenchida com o ID do curso. Assim, quando o utilizador escolhe um curso no site do Moodle, é desenhado o Grafo Social com os nós e as arestas que possuem o valor da coluna “course_id” igual ao ID do curso seleccionado e, deste modo, o modelo de dados permite abstrair grafos para múltiplos cursos na mesma plataforma.

No sistema proposto, os dados dos grafos dos cursos criados no Moodle, foram inseridos de forma manual na base de dados do Moodle. Num trabalho futuro, pretende-se que o processo de inserção de dados seja efetuado de uma forma diferente, sem recorrer diretamente à base de dados. Uma possível abordagem seria desenvolver uma aplicação complementar que fizesse a abstração deste processo, de modo a que o utilizador apenas insira na aplicação o ID do curso, e a lista dos nós e das arestas. Outra abordagem poderia ser a integração do sistema proposto com APIs de redes sociais como o Facebook, por exemplo. Deste modo, poderia ser armazenado na base de dados do Moodle dados sobre grupos do Facebook, fóruns ou outra rede social.

3.3 API de manipulação de dados do Moodle

O Moodle contém uma série de APIs *core* que fornecem ferramentas para escrever *scripts* e *plugins* para o Moodle, entre as quais a API de manipulação de dados permite ler e escrever para a base de dados do Moodle de uma forma consistente e segura.

De modo a obter os dados do grafo social contidos nas duas tabelas abordadas anteriormente,

foi utilizado o método “get_records_sql()”. Este método faz a conexão à base de dados do Moodle, e tem como parâmetro uma query SQL.

```
$nodesInfo = $DB->get_records_sql('SELECT * FROM {moodlesna_nodes} where
    course_id = ?', array( $course_id_atual ));

$edgesInfo = $DB->get_records_sql('SELECT * FROM {moodlesna_links} where
    course_id = ? ', array( $course_id_atual ));
```

Bloco de Código 3.1: API de manipulação de dados do Moodle

A fim de obter os dados da tabela dos nós e da tabela das arestas, o método é executado duas vezes com as queries correspondentes, como demonstrado no bloco de código 3.1. A variável \$nodesInfo armazena a informação dos nós e a variável \$edgesInfo a informação sobre as arestas. Os dados das duas queries são filtrados pela coluna “course_id”, que representa o ID do curso escolhido na página do Moodle.

3.4 Ficheiros de input

Os dados obtidos pelo método “get_records_sql()” são armazenados em ficheiros, de modo a serem processados posteriormente pelas bibliotecas JavaScript. Assim sendo, foram criados 3 ficheiros: "Graph.json", "Nodes.txt" e "Edges.txt". A figura 3.2 ilustra quais os ficheiros que servem de input para as bibliotecas D3.js e JSNetworkX.js

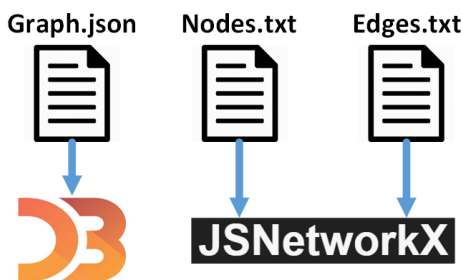


Figura 3.2: Ficheiros de input para D3.js e JSNetworkX.js

Com a biblioteca D3.js, é criado um grafo com layout *Force-Directed*. Para este tipo de grafos, a D3.js aceita como fonte de dados um ficheiro JavaScript Object Notation (JSON) com informação sobre os nós e as arestas do grafo, neste caso o ficheiro “Graph.json”. O formato deste ficheiro JSON está ilustrado no bloco de código 3.2.

Com a biblioteca JSNetworkX.js é criado um grafo vazio e, posteriormente, são-lhe adicionados os nós e as arestas, com os métodos “addNodesFrom()” e “addEdgesFrom()”, respectivamente. O primeiro método recebe como parâmetro o ficheiro “Nodes.txt”, que contém um array com todos os nós do grafo. O segundo método recebe como parâmetro o ficheiro “Edges.txt”, que contém um array com todas as arestas.

```
{
  "nodes": [
    { "id": "1" },
    { "id": "2" },
    ...
  ],
  "links": [
    { "source": "1", "target": "9", "value": 1 },
    { "source": "2", "target": "15", "value": 2 },
    ...
  ]
}
```

Bloco de Código 3.2: Formato do ficheiro “Graph.json”

3.5 Bibliotecas Javascript

A biblioteca D3.js é responsável por criar e mostrar o grafo social. Uma vez que esta biblioteca não contém funcionalidades de análise de redes sociais, optou-se por utilizar os recursos da biblioteca JSNetworkX.js. Nesta biblioteca, é criado um grafo interno com os mesmos dados da biblioteca D3.js, mas sem representação visual. Este grafo é necessário para aplicar todos os métodos e métricas de SNA, na qual os resultados são representados com recurso às funcionalidades da biblioteca D3.js.

É utilizada também a biblioteca netClustering.js. Esta biblioteca aceita como parâmetro os nós e as arestas do grafo social, e aplica o algoritmo de deteção de comunidades Clauset, Newman e Moore [23]. Este algoritmo define um novo atributo a cada nó, chamado “cluster”, que contém um inteiro que define o cluster a que pertence cada nó. Este atributo é processado pela biblioteca D3.js para definir a cor de cada nó, de acordo com o “cluster” a que pertence.

Em suma, a biblioteca JSNetworkX.js é responsável por calcular as métricas de Social Network Analysis (SNA). A biblioteca D3.js cria e desenha o grafo social, e ilustra as métricas calculadas tanto a nível do grafo, como o caminho mais curto entre dois nós, como a nível de cálculos estáticos, como a densidade. A biblioteca netClustering.js deteta os clusters da rede, e associa um atributo aos nós do grafo, que a D3.js usa atribuir uma cor correspondente a cada nó.

Capítulo 4

Interface do Utilizador

Após escolher um curso (que deve ser previamente criado no Moodle), é apresentado ao utilizador o painel de controlo ilustrado na figura 4.1. A interface gráfica é dividida em 3 regiões principais: A) o Grafo Social, no canto superior esquerdo; B) o Dashboard, no canto superior direito e C) as Propriedades do grafo e métricas, abaixo das duas outras regiões. O grafo representa os dados dos nós e das arestas para o curso escolhido, e é armazenado na base de dados local do Moodle.

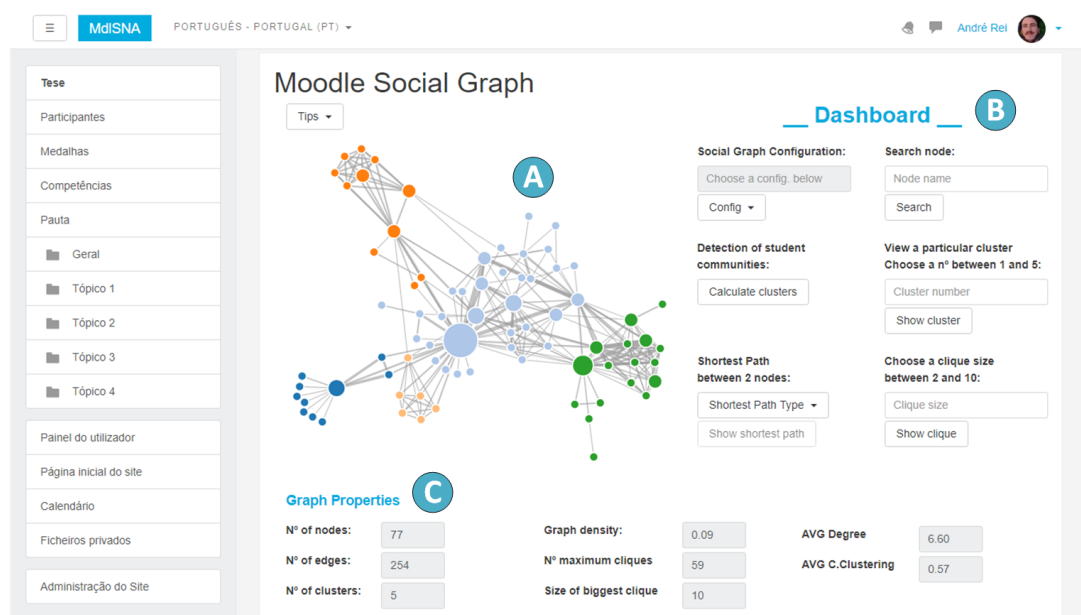


Figura 4.1: Interface do utilizador na página de um curso do Moodle

A região A representa a área do Grafo Social. Nesta área o utilizador pode interagir diretamente com o grafo, isto é, pode fazer zoom, movimentar o grafo em qualquer direção ou movimentar nós individuais, bem como visualizar diferentes características e propriedades para cada nó.

A região B permite que o utilizador defina parâmetros e efetue consultas ao sistema, de acordo com as propriedades e métricas comuns em Social Network Analysis (SNA), tal como o

caminho mais curto entre dois nós ou a deteção de sub-comunidades na rede, por exemplo. Os resultados dos parâmetros de input são traduzidos em representações gráficas que se sobrepõe ao Grafo Social da região A, da figura 5.10. Estas representações permitem ao utilizador ter uma compreensão rápida da dinâmica da rede.

Na região C, são agregadas todas as propriedades estáticas do grafo, isto é, propriedades relacionadas com a estrutura global do grafo. A primeira coluna apresenta o número de nós e de arestas do Grafo Social, bem como o número de cliques máximos existentes. A segunda coluna apresenta o valor da densidade do Grafo Social, o tamanho do maior clique e o número de clusters presentes.

- Em contexto educativo, os cliques permitem medir a proximidade das interações entre estudantes, identificar quem interage com quem diretamente ou através de intermediários, e quem são esses intermediários.
- A densidade mede o nível de coesão do grafo, e o seu valor está compreendido entre 0 e 1. Um grafo denso possui um valor de densidade próximo de 1 é um grafo no qual o número de arestas é próximo do maior número possível de diferentes conexões. Por outro lado um grafo diz-se esparso se possuir poucas arestas para uma determinada quantidade de nós, e apresenta um valor de densidade mais próximo de 0 [60]. No contexto de um ambiente de ensino-aprendizagem, a densidade da rede relacional dos alunos mede a coesão da turma;
- O tamanho do maior clique presente no Grafo Social é utilizado na região dos cliques do Dashboard. Esta região permite ao utilizador encontrar cliques no grafo que tenham um determinado tamanho. O tamanho do maior clique é utilizado para definir o intervalo de tamanhos de clique que o utilizador pode pesquisar no grafo. Ou seja, o utilizador apenas pode pesquisar cliques que tenham um tamanho compreendido entre 2 e o tamanho do maior clique presente no grafo;
- O número de clusters identifica quantos grupos de alunos fortemente ligados entre si existem na turma. Os clusters do Grafo Social são determinados quando o utilizador clica no botão “Calculate cluster”, presente na região “Detection of student communities” do Dashboard. Enquanto o utilizador não tomar esta ação, a caixa de input relativa ao número de clusters do grafo mantém-se vazia. Assim que o utilizador clicar no botão “Calculate clusters”, é aplicado um algoritmo de deteção de comunidades ao Grafo Social, determinando assim os seus clusters. Após este processo, o número de clusters do grafo é calculado e apresentado na região das Propriedades.

4.1 Interação com o Grafo Social utilizando o cursor

O utilizador pode interagir diretamente com a área do Grafo Social utilizando o cursor, de modo a obter mais informações sobre cada nó ou realçar aspetos visuais da rede.

A posição dos nós e das arestas no grafo não é estática. O utilizador pode arrastar qualquer nó que, por sua vez, arrasta as arestas e, conseqüentemente, altera a forma de todo o grafo. O grafo presente na figura 4.2a encontra-se na sua estrutura inicial. Ao arrastar o nó com o nome “Valjean”, a estrutura do grafo altera-se como ilustrado no grafo da figura 4.2b.

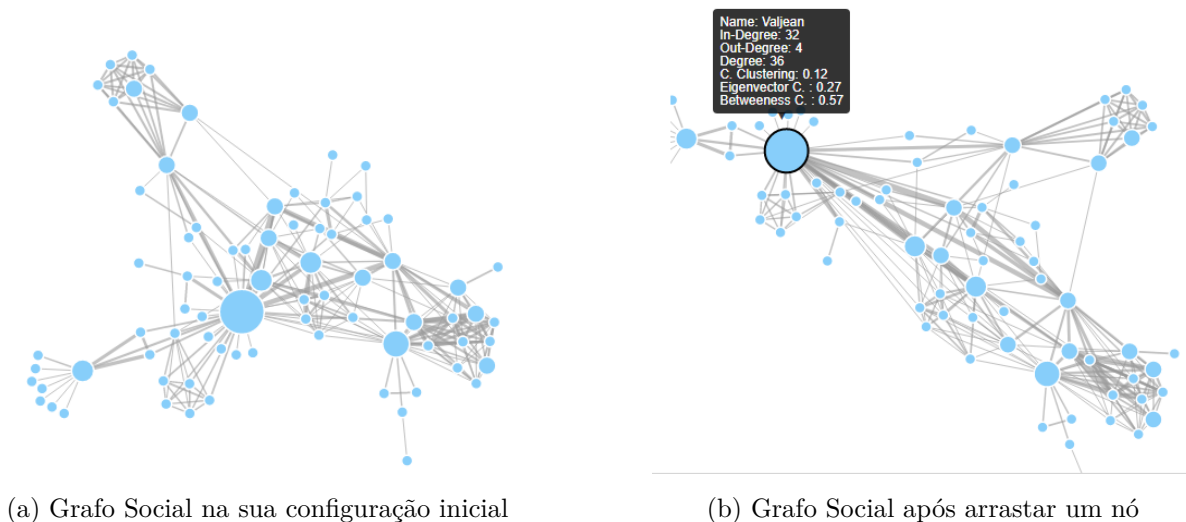


Figura 4.2: Movimentação da posição dos nós do Grafo Social utilizando o cursor

O utilizador pode também movimentar o cursor sobre os nós do grafo. Ao passar por cima de um nó, este fica destacado dos restantes nós, apresentando um tamanho ligeiramente superior e uma borda espessa. Adicionalmente, surge uma caixa com informações sobre o nó destacado, tal como o seu nome (Name), *In-degree*, *Out-degree*, grau (Degree), coeficiente de agrupamento (Clustering Coefficient), Eigenvector Centrality e coeficiente de intermediação (Betweenness Centrality). Na figura 4.3, o nó com o nome “Cosette” encontra-se destacado. Para além do seu nome, na caixa informativa podemos ver que o *in-degree* é 7, o seu *out-degree* é 4, seu grau é 11, o seu coeficiente de agrupamento é 0.38, a sua Eigenvector Centrality é 0.11 e o seu coeficiente de intermediação é 0.02.

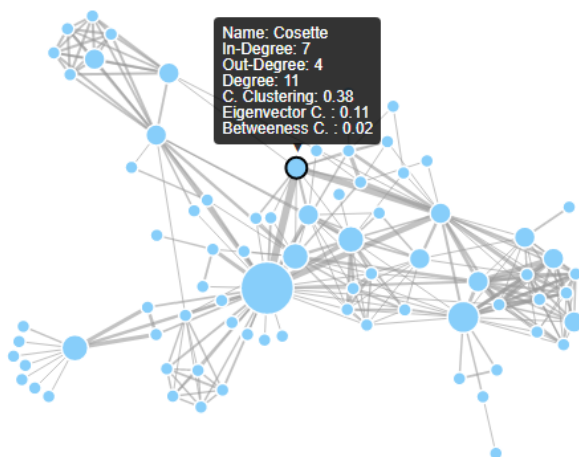


Figura 4.3: Caixa de informações de um nó do Grafo Social

O *in-degree* indica o número de arestas que entram no nó, o *out-degree* indica o número de arestas que saem do nó, o grau indica o número de nós diretamente ligados a um nó, ou seja, o número de vizinhos diretos que possui, o coeficiente de agrupamento mede a tendência do nó em formar grupos, a Eigenvector Centrality mede a influência do nó na rede e o coeficiente de intermediação mede o grau na qual um nó serve de intermediário para outros nós, respetivamente.

Com um duplo clique sobre um nó ficam destacados, no Grafo Social, os nós que são os seus vizinhos diretos e as respetivas arestas. Após o clique, o nó em questão, os seus vizinhos diretos e respetivas arestas mantêm as suas cores originais, enquanto que os restantes nós do grafo ficam com opacidade reduzida. Com um novo duplo clique no nó, o grafo volta à sua configuração inicial. No grafo da figura 4.4 estão representados os nós que são vizinhos diretos do nó com o nome “Valjean”.

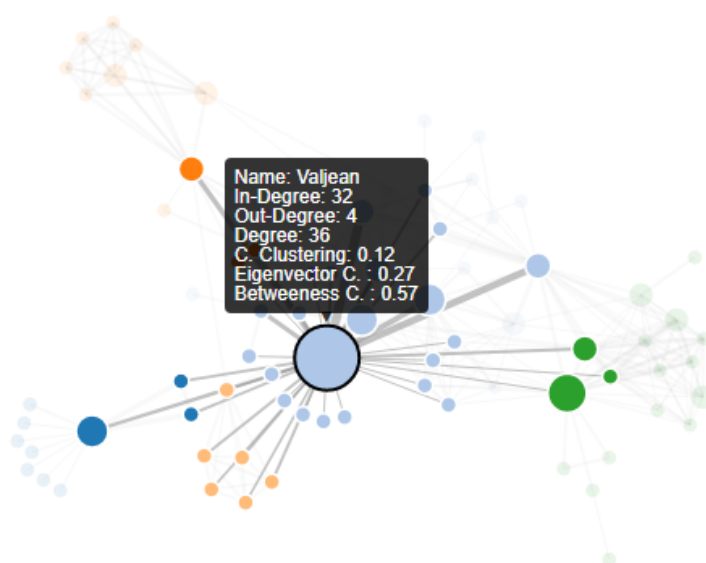


Figura 4.4: Representação dos vizinhos diretos de um nó do Grafo Social

Com esta representação, o professor pode aferir a popularidade de um aluno em particular, bem como identificar o seu posicionamento e influência na rede, relacionando-o com o seu número de contactos diretos. Além disso, alternando entre esta representação e a forma inicial do grafo, é possível verificar as posições estruturais na rede dos seus vizinhos diretos, ou seja, a posição que os vizinhos do nó em questão possuem. Estes podem ser nós periféricos na rede ou podem possuir um papel importante de intermediários no fluxo de informação entre sub-comunidades de alunos.

4.2 Interação através do Dashboard

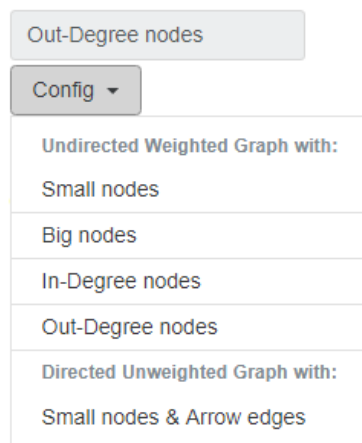
O Dashboard é composto pela secção B da figura 5.10. Esta secção inclui um conjunto de caixas pendentes, campos de inserção de texto/números e botões agrupados por ações específicas. Os botões acionam mudanças no Grafo Social, de acordo com a vista desejada. Nas secções seguintes

é explorada em detalhe cada região do Dashboard, apresentando-se as suas funcionalidades, exemplos práticos e sua contribuição para fins pedagógicos em ambiente virtual de ensino-aprendizagem.

4.2.1 Configuração do grafo social

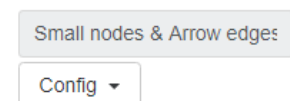
Nesta região o utilizador pode escolher diferentes configurações para os nós e para as arestas do Grafo Social. Diferentes configurações favorecem diferentes características do Grafo Social. A região é composta por uma caixa pendente, que indica a configuração escolhida pelo utilizador, e pela caixa de seleção “Config”. Quando o utilizador clica no botão da dropdown, surgem as diferentes configurações de visualização para o Grafo Social, como demonstra a figura 4.5a.

Social Graph Configuration:



(a) Diferentes configurações de visualização do Grafo Social

Social Graph Configuration:



(b) Região “Social Graph Configuration” após a escolha de uma configuração

Figura 4.5: Região “Social Graph Configuration” do Dashboard

Na caixa de seleção estão presentes dois conjuntos de configurações, agrupadas por “Undirected Weighted Graph” e “Directed Unweighted Graph”. No primeiro conjunto de configurações, o Grafo Social é representado como um grafo não direcionado com arestas pesadas. O peso das arestas é representado pela sua espessura. Este conjunto de configurações é composto por 4 opções: “Small Nodes”, “In-degree Nodes”, “Out-degree Nodes” e “Big Nodes”.

Na primeira opção, ilustrada na figura 4.6a, todos os nós do grafo têm um tamanho reduzido. Esta configuração favorece a visualização das relações entre os nós, bem como a posição dos nós na rede. A figura 4.6b, representa a configuração “Big Nodes”. A diferença desta configuração para a configuração descrita anteriormente é o tamanho superior dos nós. Esta representação favorece a visualização das sub-comunidades do grafo, quando são calculados os seus clusters. Os diferentes grupos de cores são realçados pelo tamanho dos nós.

Na configuração ilustrada pela figura 4.7a, o tamanho dos nós é proporcional ao valor do seu *in-degree*, ou seja, o número de relações que apontam diretamente para o nó. Com esta

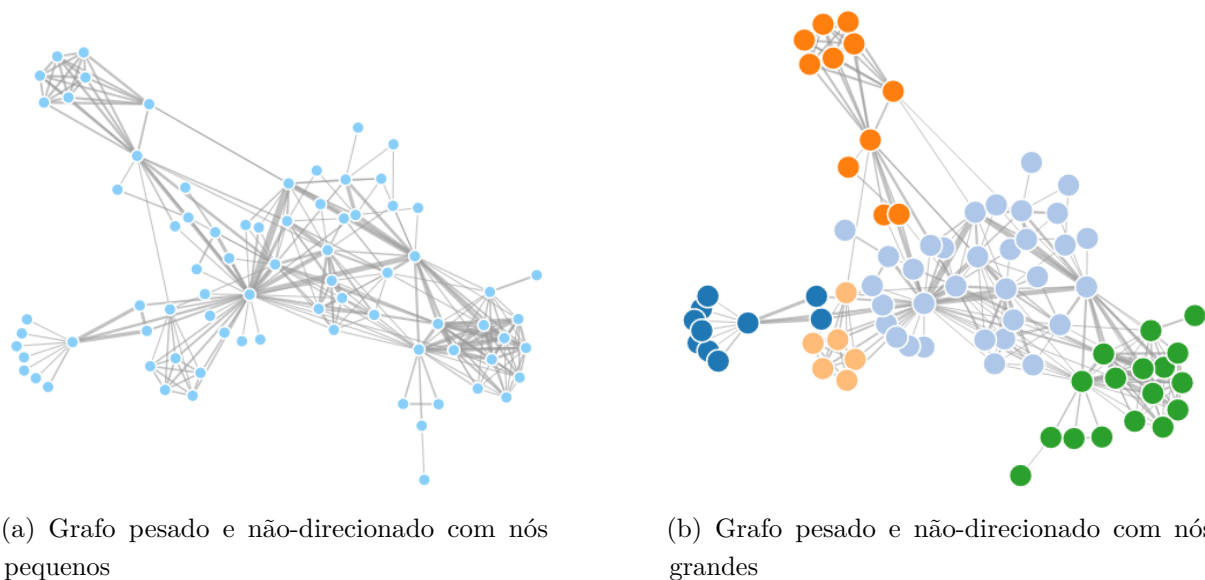


Figura 4.6: Configurações de visualização do Grafo Social #1

representação é possível ver os nós mais populares na rede, que possuem um tamanho superior em relação aos restantes nós. Na configuração ilustrada pela figura 4.7b, por outro lado, o tamanho dos nós é proporcional ao valor do seu *out-degree*. Esta configuração permite identificar os nós mais comunicativos da rede.

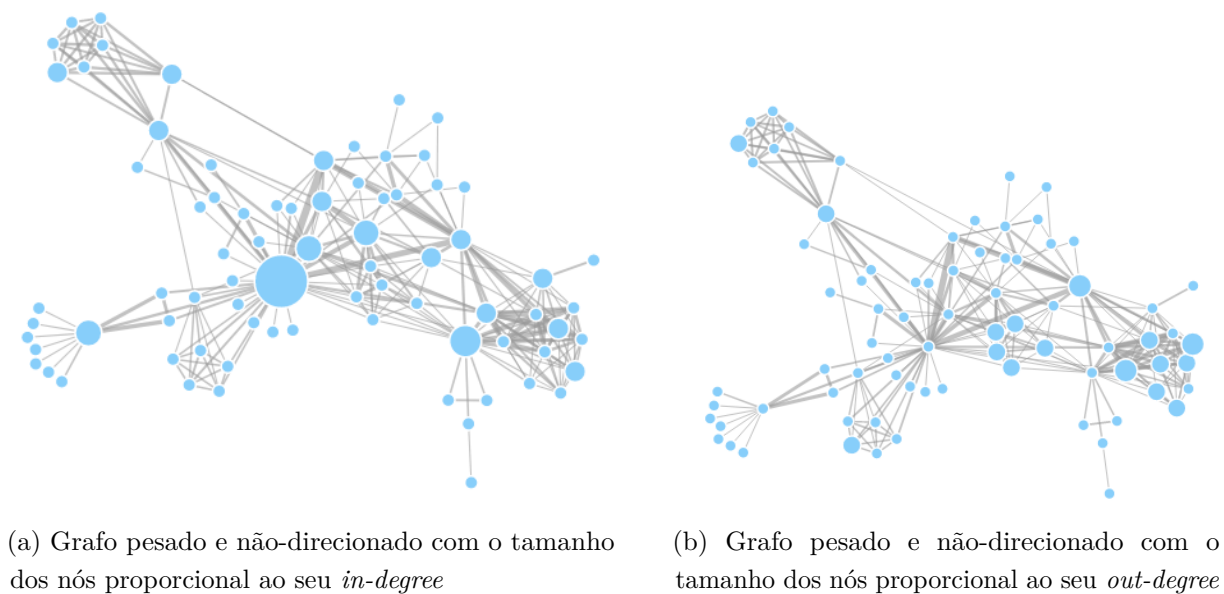


Figura 4.7: Configurações de visualização do Grafo Social #2

A última configuração da caixa de seleção está representada na figura 4.8. Nesta configuração os nós são desenhados com um tamanho reduzido de modo a favorecer a visualização da direção das arestas. Estas por sua vez são desenhadas todas com a mesma espessura, e com setas referentes à direção da relação. Com esta representação é possível ver a direção de todos os relacionamentos do Grafo Social.

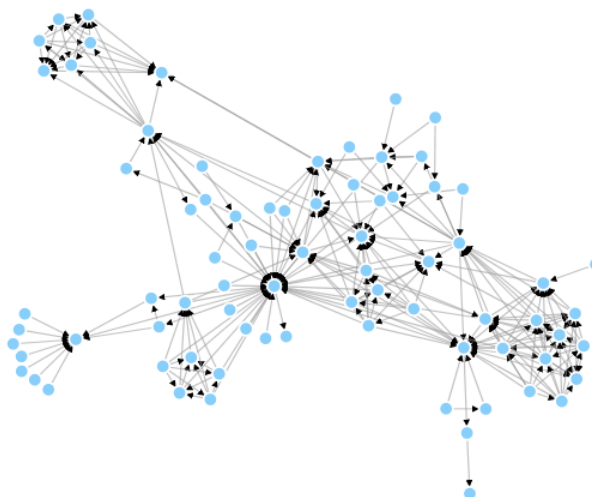


Figura 4.8: Grafo direcionado e não-pesado com nós pequenos

Em suma, é importante referir que o utilizador tem a possibilidade de mudar a configuração do Grafo Social a qualquer momento, de modo a tirar partido das diferentes vantagens que cada configuração oferece. De referir ainda que as diferentes visualizações proporcionadas pelo sistema são complementares entre si, pois oferecem informação cumulativa através das visualizações geradas. Desta forma, o utilizador pode aprofundar mais ou menos a análise que pretende realizar das interações geradas na rede.

4.2.2 Pesquisa de um nó

Um nó desempenha diferentes papéis na rede mediante a posição estrutural que ocupa. Esta região do Dashboard, ilustrada na figura 4.9, permite ao utilizador inserir o nome de um nó na caixa de inserção de texto e, após clicar no botão “Search”, localizar o nó e identificar a sua posição na rede.

Search node:

Figura 4.9: Região “Search node” do Dashboard

Após o clique no botão, é validado o texto introduzido pelo utilizador. A operação de procurar um nó é inválida caso o utilizador não tenha introduzido qualquer texto ou o nome introduzido esteja incorreto, não correspondendo deste modo a qualquer nó do Grafo Social. No exemplo da figura 4.10, o utilizador introduziu um nome de um nó que não existe no Grafo Social e, após clicar no botão “Search”, surge uma mensagem de erro informando que deve inserir um nome válido.

Search node:

Search

Input Error!
This node doesn't exist
Please insert a valid name

Figura 4.10: Mensagem de erro da região “Search node” devido a texto inserido inválido

Caso o nome inserido seja um nome válido, o Grafo Social altera a sua representação de modo a destacar a posição do nó com o nome especificado. Nesta ação, o nó em questão é desenhado a cor verde de modo a ser facilmente identificado, e os restantes nós do grafo ficam com opacidade reduzida durante um 10 segundos, como ilustra a figura 4.11a para o nó com o nome “Gavroche”, a título de exemplo. Após esse tempo, o grafo volta à sua configuração inicial, e o nó em questão altera a sua cor para a cor que tinha antes de ser procurado, como ilustra a figura 4.11b.

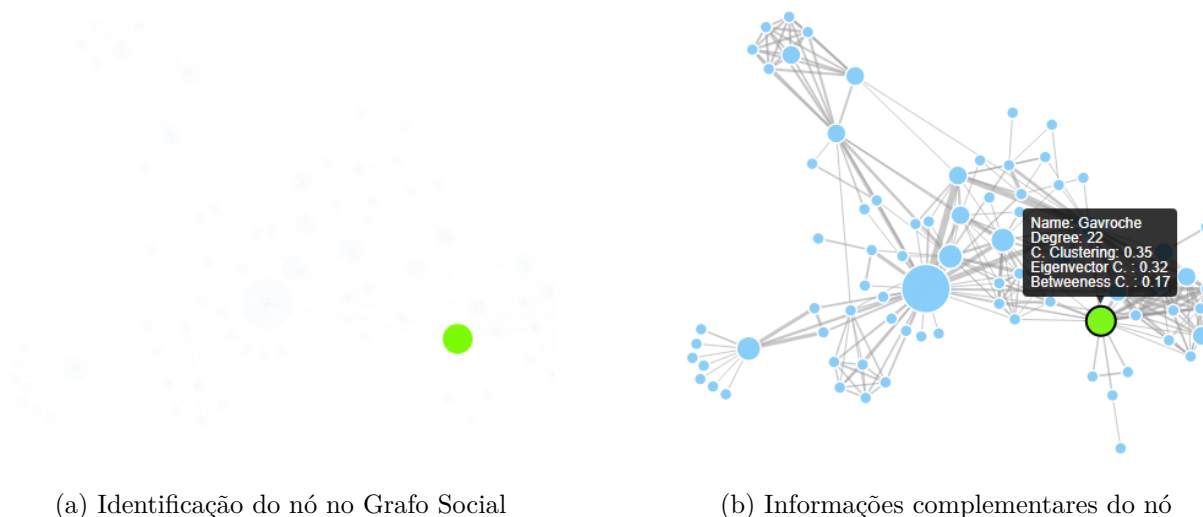


Figura 4.11: Procurar um nó no Grafo Social

É de extrema importância saber a posição que um aluno ocupa na rede de relacionamentos de um ambiente de ensino-aprendizagem, uma vez a sua posição pode ter uma relação direta com o seu papel no fluxo de informação entre alunos. Por exemplo, um aluno que ocupa uma posição central no grafo com um número elevado de relações é um aluno com um papel mais importante e popular do que um aluno que ocupa uma posição periférica na rede, com conexões reduzidas. Alunos que ocupam posições periféricas são alunos que podem carecer de mais atenção por parte do professor, e aos quais poderá ser necessário tomar medidas de incentivo à interação com os colegas, de modo a que ocupem uma posição mais central na rede e aumentem os seus laços relacionais com os restantes colegas da turma.

Adicionalmente, a ação de procurar um nó pode ser complementada com a caixa de informações que surge quando o utilizador mantém o cursor em cima de um nó. A figura 4.11b representa a caixa de informações para o nó “Gavroche”. Deste modo, o utilizador pode comparar os valores do seu grau e das restantes medidas de centralidade com a sua posição física na rede.

4.2.3 Detecção de comunidades

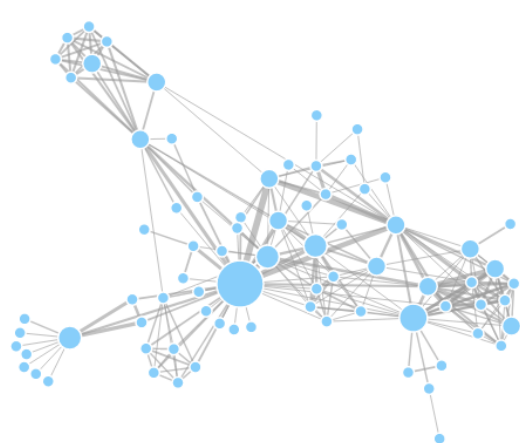
Quando a página de um curso surge pela primeira vez, todos os nós do Grafo Social são apresentados a cor azul claro, por defeito. Ao clicar no botão “Calculate clusters” presente nesta região do Dashboard, ilustrada na figura 4.12, é aplicado o algoritmo de deteção de comunidades “Clauset, Newman and Moore” [23] ao Grafo Social.

**Detection of student
communities:**

Calculate clusters

Figura 4.12: Região “Detection of student communities” do Dashboard

O algoritmo associa um atributo a cada nó, que define o cluster ao qual este pertence. O valor deste atributo é utilizado para determinar a cor do nó no redesenho do grafo. Deste modo, os nós que possuem o mesmo valor deste atributo vão ser desenhados com a mesma cor.



(a) Grafo Social antes do algoritmo de deteção de comunidades ser aplicado



(b) Grafo Social após algoritmo de deteção de comunidades ser aplicado

Figura 4.13: Deteção de comunidades no Grafo Social

O resultado é um grafo com nós coloridos de acordo com o cluster a que pertencem, tal como o grafo ilustrado na figura 4.13b. Com esta representação o professor tem a noção aproximada das sub-comunidades de alunos existentes na turma.

4.2.4 Ver uma comunidade

Do lado direito da região “Detection of Student Communities”, encontra-se a região “View a particular cluster”, presente na figura 4.14.

View a particular cluster:

Figura 4.14: Região “View a particular cluster” do Dashboard

Esta região permite ao utilizador apenas ver os nós pertencentes a um cluster em particular. Enquanto o utilizador não calcular os clusters do Grafo Social, esta região permanece desabilitada. Assim que o utilizador o fizer, esta região torna-se ativa e a o seu rótulo altera-se, informando o utilizador de que pode escolher um número de um cluster entre 1 e o número máximo de clusters existentes no Grafo Social. A figura 4.15 representa a região “View a particular cluster” ativa, após o utilizador ter calculado os clusters do Grafo Social. Através do rótulo da região, verifica-se que existem 5 comunidades no Grafo Social.

View a particular cluster
Choose a nº between 1 and 5:

Figura 4.15: Região “View a particular cluster” ativa, após o cálculo dos clusters do Grafo Social

Caso o utilizador clique no botão “Show cluster”, e não tenha introduzido qualquer valor ou o valor introduzido esteja fora do intervalo indicado, surge uma mensagem de erro. No exemplo da figura 4.16a, o utilizador introduziu o valor 7. Uma vez que o valor não está compreendido entre 1 e 5 surge uma mensagem de erro, informando o utilizador que este deve introduzir um valor válido.

Caso o utilizador introduza um número válido, surge uma caixa de informação abaixo do botão “Show cluster”, indicando o número de nós que pertencem ao cluster em questão. Na figura 4.16b, o utilizador introduziu o número de cluster 2 na caixa de inserção de texto, que possui 33 nós.

Em resultado, no Grafo Social todos os nós ficam com opacidade reduzida durante 10 segundos, com a exceção dos nós pertencentes ao cluster, que permanecem coloridas. A figura 4.17a representa o Grafo Social para o exemplo da figura 4.16a, destacando os nós do cluster 2. Após o tempo da animação, o grafo volta à sua configuração inicial, tal como ilustrado na figura 4.17b.



Figura 4.16: Mensagens de informação e de erro da região “View a particular cluster”

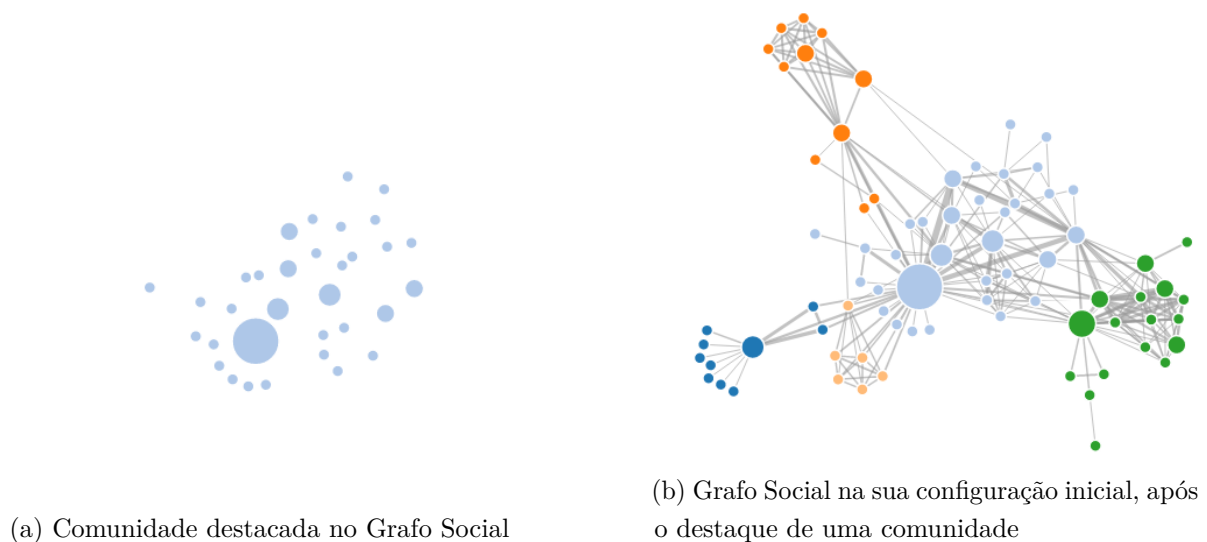


Figura 4.17: Ver uma comunidade no Grafo Social

4.2.5 Caminho mais curto entre nós

A figura 4.18a ilustra a configuração inicial da área “Shortest Path”, composta por uma caixa de seleção com os tipos de “Shortest Paths” disponíveis, e o botão “Show shortest path” que se encontra desativado, enquanto o utilizador não selecionar uma das opções. A dropdown apresenta 3 tipos diferentes de “Shortest Paths”, nomeadamente as opções “S.P. in Unweighted Graph”, “S.P. in Weighted Graph” e “S.P. Single Source Dijkstra”, como ilustra a figura 4.18b. Se o utilizador selecionar a primeira ou a segunda opção na caixa de seleção, surgem duas caixas de inserção de texto, nas quais o utilizador pode inserir o nome do nó inicial do caminho e o nome do nó final, respetivamente. Se o utilizador selecionar a ultima opção, apenas surge uma caixa de input.

Após o utilizador inserir o nome dos nós e clicar no botão “Show shortest path”, são verificadas



Figura 4.18: Região “Shortest Path” do Dashboard

as seguintes condições:

- As caixas de input não estão vazias;
- Os nomes dos nós inseridos nas caixas de input são nomes válidos;
- Existe um caminho entre o nó inicial e o nó final, para os nomes inseridos pelo utilizador.

No exemplo representado pela figura 4.19, o utilizador escolheu a opção “S.P. in Weighted Graph” na caixa de seleção, e inseriu o nome “Count” para o nó inicial do caminho, e o nome “Valjean” para o nó final, nas respetivas caixas de input. Após clicar no botão “Show shortest path”, o utilizador é notificado através de uma mensagem de aviso que não existe um caminho no Grafo Social para os nós especificados.

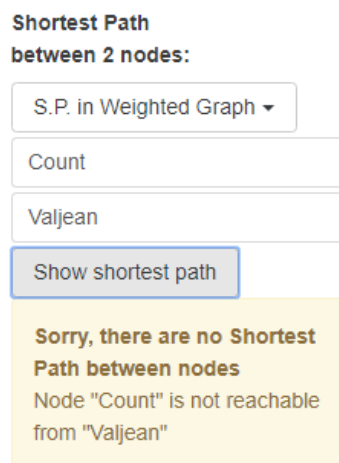


Figura 4.19: Mensagem de erro no cálculo de um “Shortest Path”

Alternar entre os diferentes tipos de “Shortest Path” na caixa de seleção, faz com que o Grafo Social adapte a sua representação mediante a configuração que o utilizador escolheu previamente

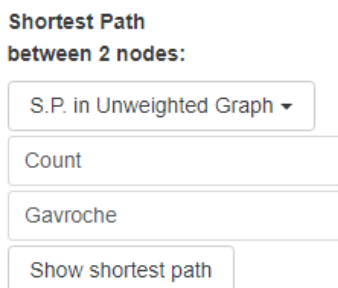
na área “Social Graph Configuration” do Dashboard. Caso o utilizador não tenha escolhido uma configuração até ao momento, o Grafo Social adapta a sua representação à configuração inicial, na qual as arestas são pesadas e o tamanho dos nós é proporcional ao seu “In-degree”. Este mecanismo garante que o Grafo Social fica sempre limpo da representação do “Shortest path” calculado anteriormente.

Adicionalmente, nesta alteração, as cores dos nós do Grafo Social dependem do facto do utilizador ter clicado ou não no botão “Calculate clusters” da área “Detection of student communities” no Dashboard. Em caso afirmativo, os nós apresentam-se coloridos de acordo com os clusters calculados por esta ação, caso contrário os nós apresentam a cor predefinida azul claro.

4.2.5.1 Caminho mais curto num grafo não-pesado

A opção “S.P. in Unweighted Graph” calcula o caminho mais curto entre dois nós no grafo, considerando o grafo como sendo não direcionado e não-pesado. Deste modo, o caminho mais curto entre dois nós apenas tem em consideração a estrutura da rede em termos dos relacionamentos existentes entre esses dois nós, e não a direção ou peso das suas arestas. Tendo em conta que o Grafo Social tem sempre apenas uma componente conexa, ao selecionar esta opção vai ser sempre apresentado um caminho entre dois nós ao utilizador.

A figura 4.20 representa a região “Shortest Path” após o utilizador escolher a opção “S.P. in Unweighted Graph”. Foi escolhido o nó inicial com o nome “Count” e nó final com o nome “Gavroche”.



Shortest Path
between 2 nodes:

S.P. in Unweighted Graph ▾

Count

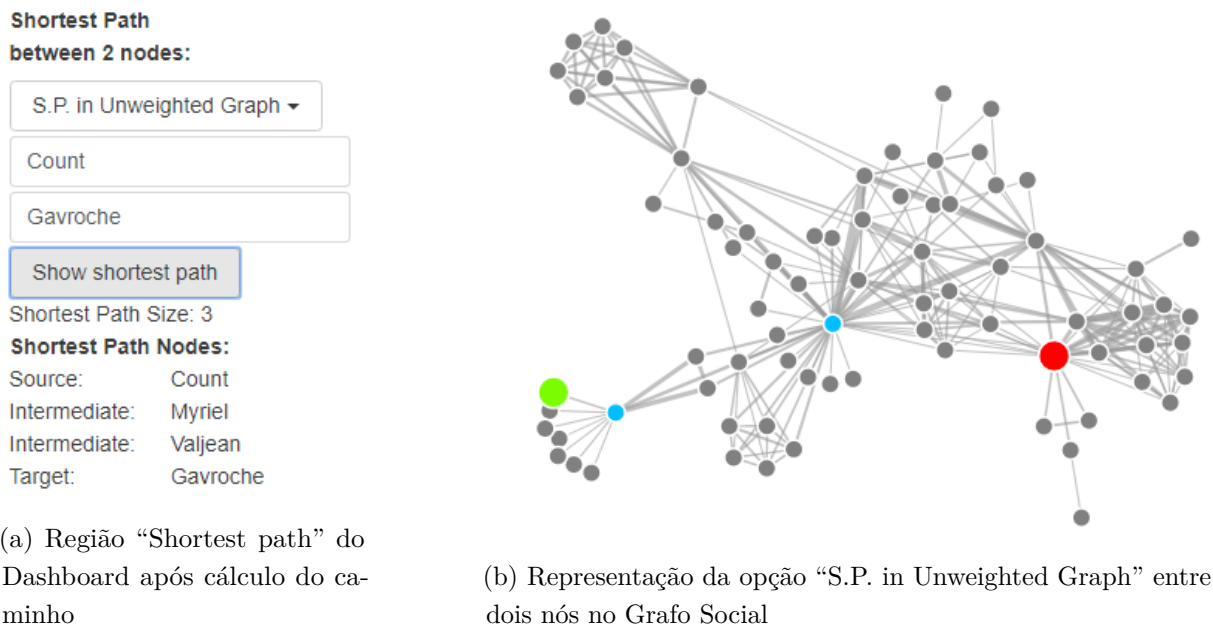
Gavroche

Show shortest path

Figura 4.20: Região “Shortest Path” para a opção “S.P. in Unweighted Graph”

Após o utilizador clicar no botão “Show shortest path”, o Grafo Social altera a sua representação de modo a destacar o caminho pretendido. O grafo da figura 4.21b representa o caminho mais curto entre o nó “Count” e o nó “Gavroche”. O nó inicial e o nó final apresentam um tamanho realçado em comparação com os restantes nós do caminho. O nó inicial é desenhado a verde, o nó final a vermelho, e os nós intermédios a cor azul. Os restantes nós do grafo apresentam cor cinza, destacando deste modo apenas os nós que compõem o caminho.

Juntamente com as mudanças no Grafo Social na região “Shortest Path” do Dashboard, surgem informações adicionais sobre o caminho em questão, nomeadamente o tamanho do



(a) Região “Shortest path” do Dashboard após cálculo do caminho

(b) Representação da opção “S.P. in Unweighted Graph” entre dois nós no Grafo Social

Figura 4.21: Caminho mais curto entre dois nós no Grafo Social não direcionado e não-pesado

caminho e os nomes dos nós que o compõem. Para o exemplo anterior, podemos observar através da figura 4.21a que o caminho mais curto entre o nó “Count” e o nó “Gavroche” tem tamanho 3, e que os nós que o constituem são: “Count”, “Myriel”, “Valjean” e “Gavroche”.

Com esta representação do Grafo Social o professor tem a noção da distância “física” entre dois alunos na rede, assim como a localização lógica absoluta dos nós na totalidade da rede. Por outro lado, é relevante observar a posição dos nós intermédios do caminho. Estes saltos intermédios entre o nó inicial e o nó final podem passar por nós com um “In-degree” elevado ou por sub-comunidades de alunos. Adicionalmente, o tamanho do caminho e o número de nós intermédios diz-nos quantos saltos são necessários para que o aluno inicial possa interagir com o aluno final, por outras palavras, com quantos outros alunos (intermediários) precisa de se relacionar para se relacionar para chegar ao aluno final.

4.2.5.2 Caminho mais curto num grafo pesado

A opção “S.P. in Weighted Graph”, por outro lado, calcula o caminho mais curto entre dois nós considerando o grafo como sendo direcionado e pesado. Deste modo, cada salto na construção do caminho mais curto tem em consideração a direção e o peso mínimo das arestas.

A figura 4.22a representa o caminho mais curto entre o nó inicial “Marius”, desenhado a verde claro, e o nó final “Cossete”, desenhado a vermelho, usando a opção “S.P. in Unweighted Graph”. Como existe uma ligação direta entre os dois nós, o caminho mais curto consiste apenas na relação entre eles. Contudo, é visível que a sua relação é uma relação com um peso elevado, tendo em consideração a espessura da aresta.



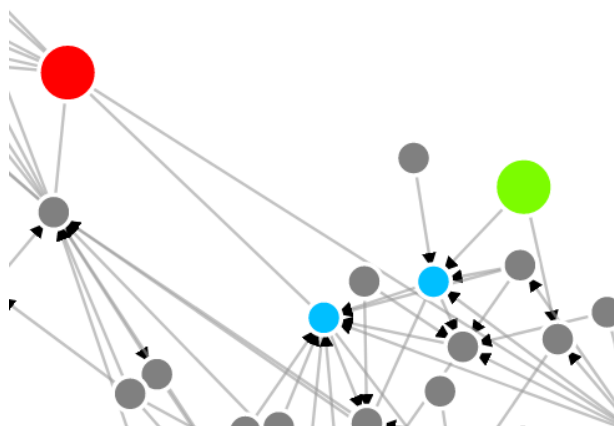
(a) “S.P. in Unweighted Graph” entre o nó “Marius” e o nó “Cosette”



(b) “S.P. in Weighted Graph” entre o nó “Marius” e o nó “Cosette”

Figura 4.22: Comparação do caminho mais curto entre dois nós entre a opção “S.P. in Unweighted Graph” e a opção “S.P. in Weighted Graph”

Por outro lado, a figura 4.22b representa o caminho mais curto entre os mesmos nós “Marius” e “Cosette”, usando a opção “S.P. in Weighted Graph”. Neste exemplo, o caminho mais curto passa a ter tamanho 2 e a conter um nó intermédio chamado “LT. Gillenormand”. Isto deve-se ao facto de a soma dos pesos das arestas nos saltos “Marius -> LT. Gillenormand” e “LT. Gillenormand -> Cosette” ser inferior ao peso da aresta do salto “Marius -> Cosette”, ilustrado na figura 4.22a. Em suma, a opção “S.P. in Weighted Graph” apresenta sempre o caminho mais curto entre dois nós com um custo igual ou inferior ao caminho mais curto calculado pela opção “S.P. in Unweighted Graph”.



(a) Caminho mais curto entre o nó “Mme.Pontmercy” e o nó “Tholomyes” para opção “S.P. in a Unweighted Graph”

Shortest Path
between 2 nodes:

S.P. in Weighted Graph ▾

Tholomyes

Mme.Pontmercy

Show shortest path

Sorry, there are no Shortest Path between nodes
Node "Tholomyes" is not reachable from "Mme.Pontmercy"

(b) Mensagem de aviso devido à inexistência de caminho entre o nó “Tholomyes” e o nó “Mme.Pontmercy”

Figura 4.23: Mensagem de aviso devido à inexistência de um caminho entre dois nós para a opção “S.P. in a Unweighted Graph”

Contudo, nesta opção nem sempre existe um caminho entre dois nós, uma vez que para a

construção do caminho é tido em consideração a direção e o peso das arestas. Uma vez que o grafo é considerado como sendo direcional, a existência de um caminho entre o nó A e o nó B, não implica a existência de um caminho entre o nó B e o nó A e, neste sentido, a ordem dos nós para o cálculo do caminho é relevante. O grafo da figura 4.23a representa o caminho mais curto entre o nó “Mme.Pontmercy”, desenhado a verde, e o nó “Tholomyes”, desenhado a vermelho, para a opção “S.P. in a Unweighted Graph”. Para esta representação foi escolhida a configuração do grafo “Directed Unweighted Graph - Small nodes & Arrow Edges”, de modo a ser visível a direção das arestas. Neste exemplo, existe um caminho entre os dois nós que tem tamanho 3 e possui 2 nós intermédios.

No exemplo da figura 4.23b o utilizador trocou a ordem dos nós, com o intuito de calcular o caminho mais curto entre o nó “Tholomyes” e o nó “Mme.Pontmercy”. Porém, após clicar no botão “Show shortest path” surge uma mensagem de aviso, informando o utilizador de que não existe um caminho entre os dois nós pelo facto de o nó final não ser alcançável pelo nó inicial.

No contexto educativo, a opção “S.P. in Weighted Graph” permite ao professor verificar o caminho que o fluxo de informação efetua na rede entre dois alunos. Este caminho tem sempre em consideração o custo mínimo das relações entre alunos. Adicionalmente, o professor pode ver se dois alunos são alcançáveis por um caminho unidirecional ou bidirecional e, deste modo, verificar se a informação que sai do aluno inicial chega ao aluno final.

4.2.5.3 Nós alcançáveis

Por vezes é relevante saber quais são todos os nós alcançáveis a partir de um determinado nó, isto é, escolhendo um nó em particular, saber quais são os nós da rede que são alcançáveis a partir deste. A opção “S.P. Single Source Dijkstra” utilizado o algoritmo de Dijkstra, de modo a calcular todos os caminhos mais curtos a partir de um nó, considerando o Grafo Social direcionado e pesado.

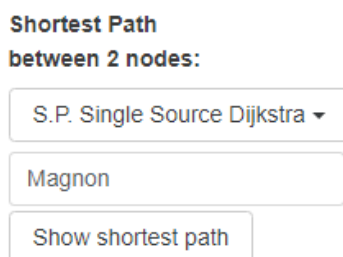


Figura 4.24: Região “Shortest Path” do Dashboard para a opção “Single Source Dijkstra Path”

Ao selecionar esta opção na caixa de seleção, surge apenas uma caixa de inserção de texto para o utilizador introduzir o nome do nó fonte. No exemplo representado na figura 4.24, o utilizador introduziu o nome do nó “Magnon”. Após clicar no botão “Show shortest path”, o nó que possui o nome inserido surge destacado com um tamanho superior em comparação aos dos restantes nós do grafo, e desenhado a cor verde. Todos os nós alcançáveis a partir do nó

fonte são desenhados a cor laranja, e os restantes nós do grafo desenhados a cinza, como ilustra a figura 4.25a.

A figura 4.25b representa o mesmo grafo, com com a configuração “Directed Unweighted Graph - Small nodes & Arrow Edges”, de modo a ver visível a direção das arestas.

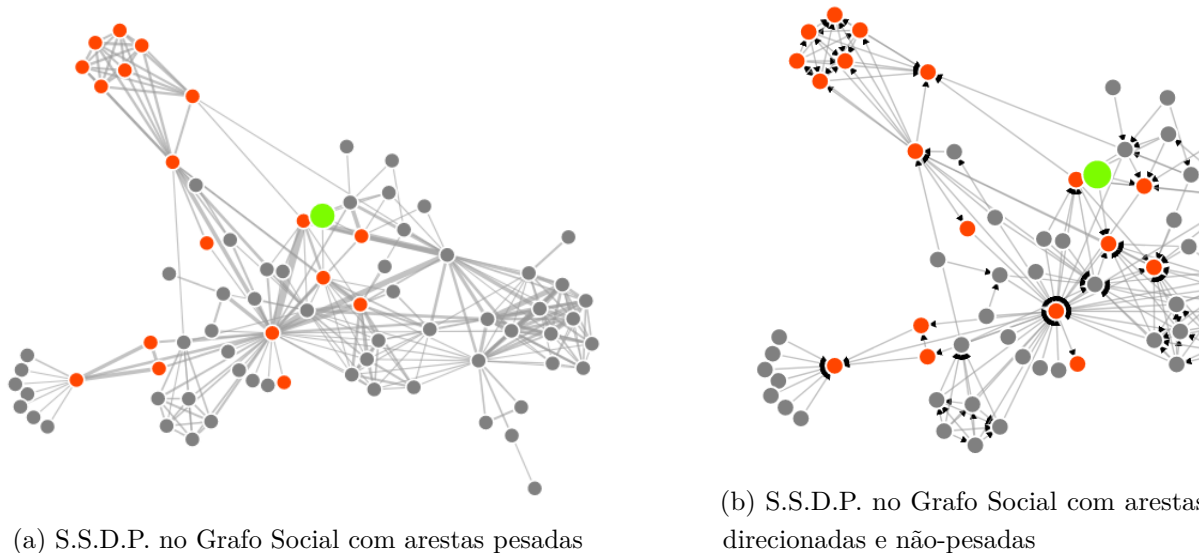


Figura 4.25: Nós alcançáveis a partir do nó “Magnon” do Grafo Social

Com esta representação, o professor pode ver quais são todos os alunos que são alcançáveis a partir de um aluno em particular. Num exemplo prático, o professor pode ver por onde é que a informação flui na rede a partir de um determinado aluno, e quais são os alunos que obtém a informação. Para além disso, os alunos posicionados no final de um caminho são considerados alunos terminais na linha de comunicação a partir do aluno fonte. Isto significa que quando a informação chega a um aluno com este perfil, não é transmitida para mais ninguém. Os alunos que nunca obtém qualquer informação a partir do aluno fonte identificado, são alunos que não são alcançáveis a partir do nó fonte e são desenhados a cinza.

4.2.6 Cliques

Um clique consiste num sub-grafo máximo completo [46]. No contexto de redes sociais, um clique identifica um grupo de pessoas na qual todas as pessoas tem uma ligação/relação direta com todas as outras. Adicionalmente, um clique pode definir um conjunto de recursos educativos que estão a ser usados em conjunto (i.e. pode haver uma maior dependência destes recursos, do que os restantes recursos).

A região “Cliques”, ilustrada na figura 4.26, é a ultima região do Dashboard, e permite ao utilizador verificar a existência de cliques no Grafo Social que possuem um determinado tamanho a designar pelo utilizador. Esta região é composta por uma caixa de inserção de texto e pelo botão “Show clique”.

Choose a clique size
between 2 and 10:

Figura 4.26: Região “Cliques” do Dashboard

O número inserido na caixa de input deve estar compreendido entre 2 e o tamanho do maior clique do grafo, calculado previamente e apresentado na secção “Properties” do painel de controlo (secção C, da figura 5.10). Assim que o utilizador clica no botão “Show clique”, o valor de input é avaliado de modo a verificar as seguintes condições:

- A caixa de inserção de texto não possui qualquer valor;
- O valor inserido está fora do intervalo de valores especificado;
- Não existe um clique no Grafo Social com o tamanho inserido.

Caso o valor inserido pelo utilizador satisfaça alguma das opções anteriores, o input é considerado inválido. Nas duas primeiras opções é apresentado uma mensagem de erro e na última uma mensagem de aviso. No exemplo da figura 4.27a, o utilizador inseriu o tamanho de clique 12. Como o valor inserido está fora do intervalo especificado, entre 2 e 10, surge uma mensagem de erro. No exemplo da figura 4.27b, o utilizador inseriu o tamanho de clique 9, que se encontra dentro do intervalo especificado. Contudo, como não existe no Grafo Social qualquer clique de tamanho 9, surge uma mensagem de aviso.

Choose a clique size
between 2 and 10:

Wrong Input!
Must choose a number
between 2 and 10

(a) Mensagem de erro devido a tamanho de clique inválido

Choose a clique size
between 2 and 10:

**Sorry, there are no
cliques of size 9**
Please, choose a different
number between 2 and 10

(b) Mensagem de aviso devido à inexistência de cliques com o tamanho inserido

Figura 4.27: Mensagens de erro e de aviso da região “Cliques” do Dashboard

Por outro lado, se o texto inserido for válido, a região dos cliques é complementada com informações sobre o número de cliques que existem no Grafo Social com o tamanho especificado

e a lista dos nós que compõem o primeiro clique. No exemplo da figura 4.28a, o utilizador inseriu o tamanho de clique 4. Após validado o texto inserido, surge uma caixa de informação azul, indicando ao utilizador que existem 11 cliques de tamanho 4 no Grafo Social, e a lista dos nós que compõem o clique em questão. Para o exemplo da figura 4.28a, os nós que compõem o primeiro clique são: “Valjean”, “Myriel”, “Mlle.Baptistine” e “Mme.Magloire”.

Choose a clique size
between 2 and 10:

Exist 11 cliques of size 4

◀ Previous Next ▶

Nodes of the 1º clique:
Valjean
Myriel
Mlle.Baptistine
Mme.Magloire

(a) Região “Cliques” com informações sobre o primeiro clique de tamanho 4 do Grafo Social



(b) Representação no Grafo Social do primeiro clique com tamanho 4

Figura 4.28: Procurar cliques com um tamanho predefinido no Grafo Social

Juntamente com as alterações na área dos cliques do Dashboard, descritas anteriormente, o Grafo Social altera a sua representação de modo a destacar os nós que compõem o primeiro clique com o tamanho indicado. A figura 4.28b representa o Grafo Social, destacando primeiro clique de tamanho 4 da figura 4.28a. Os botões de navegação ilustrados na figura 4.28a, quando acionados, para além de alterar a lista dos nós para o clique atual, alterarem a representação do clique no Grafo Social.

Em suma, com esta representação, o processo de iterar sobre os cliques é dinâmico e intuitivo, permitindo ao utilizador saber sempre quais são nós que compõem o clique em questão e observar a sua localização no Grafo Social. Em contexto educativo, o professor pode escolher um tamanho para um grupo máximo e completo de alunos, e ver os diferentes grupos com esse tamanho no Grafo Social. Por fim, caso o professor deseje escolher outro tamanho, pode apagar o tamanho inserido e inserir outro valor. Neste processo a região dos cliques do Dashboard volta à sua configuração inicial, presente na figura 4.26, e o Grafo Social volta à configuração escolhida na região “Social Graph Configuration” do Dashboard.

4.3 Resumo

No presente capítulo foi apresentada a interface do utilizador e os seus componentes, bem como todas as ações que o utilizador pode exercer no sistema proposto na página de um curso do Moodle.

A área de desenho do Grafo Social permite representar interações ou recursos de um ambiente de ensino-aprendizagem virtual. O utilizador pode interagir diretamente com esta área, de modo a obter mais informação sobre cada nó, fazer zoom, movimentar o grafo como um todo e movimentar nós particulares.

A secção do Dashboard é composta por pequenas regiões agrupadas por funções específicas. O Dashboard permite ao utilizador escolher diferentes configurações de visualização para o Grafo social, identificar a posição de nós no grafo, calcular e visualizar comunidades, calcular diferentes caminhos e identificar cliques.

Abaixo das duas secções referidas anteriormente, está presente a secção das propriedades e métricas do Grafo Social, apresentando valores estáticos como o número de nós e arestas do grafo, o número de comunidades detetadas, o número de cliques máximos e to tamanho do maior clique, e a média dos graus dos nós e dos seus coeficientes de agrupamento.

Capítulo 5

Aplicação prática do Sistema

Neste capítulo é apresentado um exemplo prático do sistema proposto. Os dados do Grafo Social derivam de um estudo, na qual foram capturadas as interações de um grupo de 24 pessoas num fórum. Nesta discussão, o utilizador Vasilis Daloukas apresenta um novo módulo para o Moodle que ele próprio criou, capaz de correr jogos. Outros utilizadores comentam este módulo e oferecem conselhos ao criador do módulo, de modo a que este o possa melhorar. Neste exemplo de estudo, é feita a interpretação da representação das interações no Grafo Social, bem como das suas propriedades e métricas aplicadas.

5.1 Análise do Grafo Social

A figura 5.1a representa o Grafo Social, na sua configuração predefinida, ou seja, o grafo é não-direcionado e pesado, com o tamanho dos nós proporcional ao seu *in-degree*. Numa primeira abordagem ao grafo, é possível ver que existe um nó central que possui um tamanho superior em relação aos restantes nós, o que significa que possui o maior valor do *in-degree*. A figura 5.1b representa o Grafo Social após a deteção de comunidades. Pelas cores ilustradas no grafo, podemos aferir que foram calculadas 5 comunidades diferentes e que o nó central está inserido na comunidade com o maior número de nós.

Ao passar com o cursor por cima de um nó, surge uma caixa com informações adicionais sobre o nó em questão. A figura 5.2a representa a caixa de informações para o nó “Vasilis Daloukas”, posicionado ao centro do grafo.

Abaixo do nome do nó, está presente o valor do seu *in-degree*, 14, o valor da seu *out-degree*, 10, e o valor do seu grau, 17. O *in-degree* representa o número de arestas que entram num nó, e o *out-degree* representa o número de arestas que saem de um nó. O grau, por outro lado, representa o número de nós ligados diretamente a um nó, considerando o grafo como não-direcionado. Os nós do grafo ilustrado 5.2a tem o seu tamanho proporcional ao seu *in-degree*, o que confirma a suposição de que o nó “Vasilis Daloukas” é o nó que possui o valor de *in-degree* mais elevado da rede. A figura 5.2b representa o Grafo Social, realçando a direção das relações na rede, onde

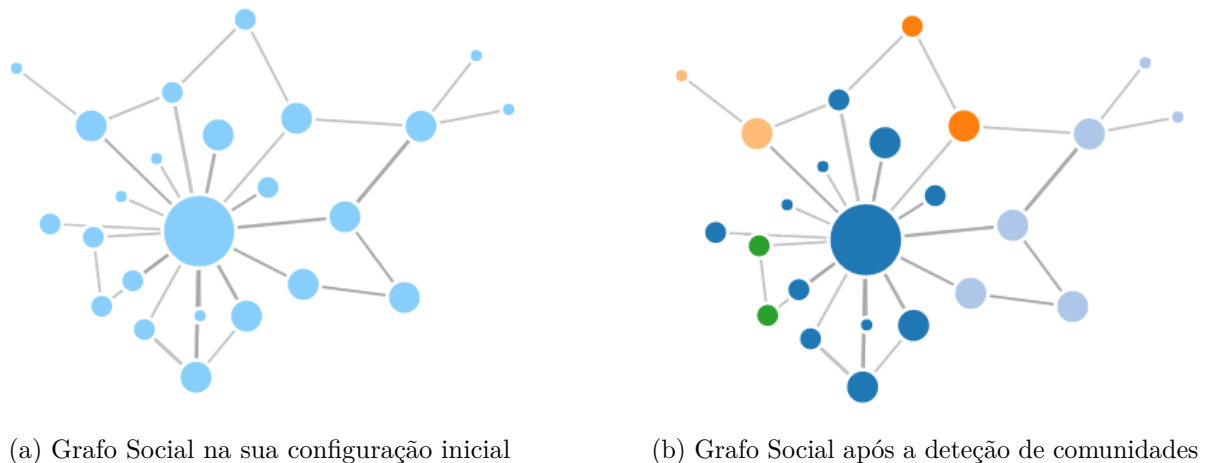


Figura 5.1: Deteção de comunidades no Grafo Social

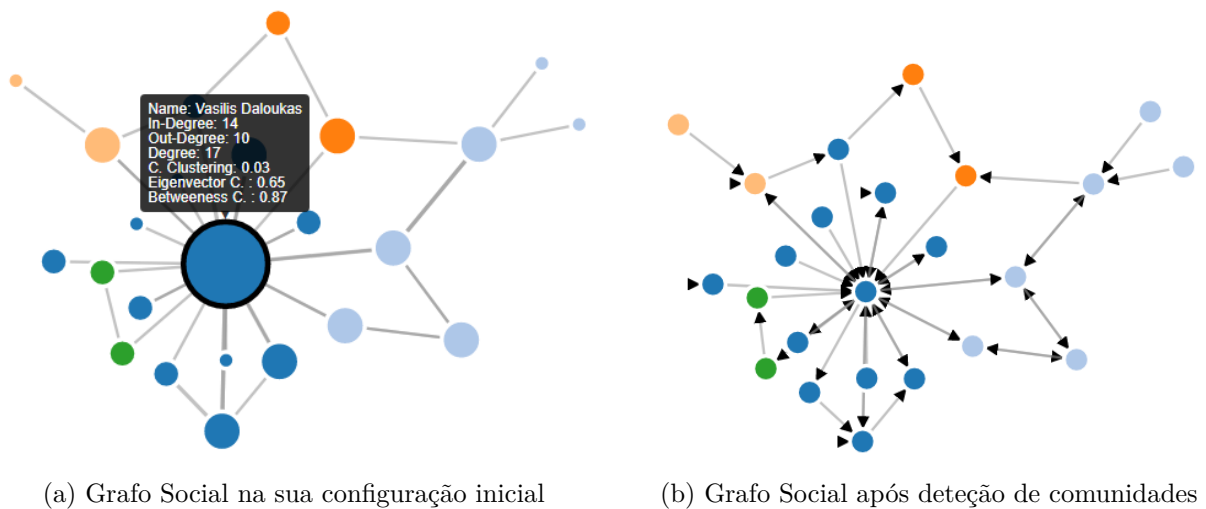


Figura 5.2: Deteção de comunidades no Grafo Social

podemos confirmar que o nó em questão possui *in-degree* 14 e o *out-degree* 10.

De seguida, surge o valor do seu coeficiente de agrupamento, que mede o grau com que um determinado nó tende em agrupar-se a outros nós, formando grupos altamente coesos. O nó “Vasilis Daloukas” apresenta um coeficiente de agrupamento muito próximo de 0, o que indica que não tem tendência a formar grupos coesos. Isto deve-se ao facto de a maioria dos seus nós vizinhos possuírem poucos ou nenhuns relacionamentos, não se agrupando também a outros nós. O próximo valor do quadro informativo é a Eigenvector Centrality, com o valor 0.65. Esta medida de centralidade mede a influência de um nó na rede. O nó “Vasilis Daloukas” apresenta um valor de Eigenvector Centrality elevado, o que se justifica pela sua importante posição central na rede. Por último, é apresentado o coeficiente de intermediação, com o valor 0.87. De uma forma simples, o coeficiente de intermediação mede o grau com que outros nós utilizam um nó para chegarem a outros nós. O nó “Vasilis Daloukas” apresenta um valor para este coeficiente bastante elevado, pois devido à sua posição central acaba por servir de intermediário para a

maioria dos nós chegarem a outros nós.

Em interações online, seja em fóruns, redes sociais ou blogs, é comum uma pessoa comentar o próprio *post* que publicou, ou questão que levantou. Em contexto de Social Network Analysis (SNA), estas pessoas representam-se por um nó que possui uma aresta a apontar para ele mesmo. No grafo ilustrado na 5.3, está destacado o nó “Toan Tran”, que possui uma aresta para ele mesmo.

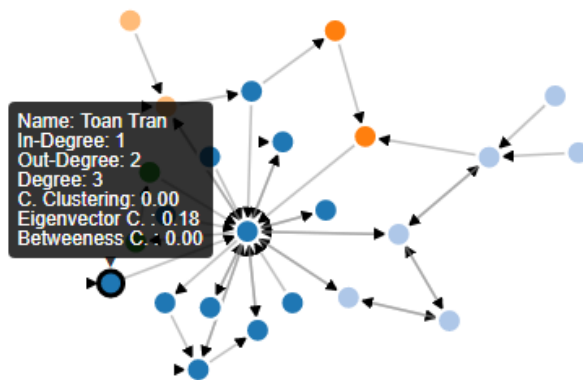


Figura 5.3: Representação de uma aresta para o mesmo nó no Grafo Social

O exemplo de layout da biblioteca D3.js que serve de base para o desenvolvimento deste projeto, não possui a funcionalidade de representar este tipo de relacionamentos com uma aresta curva, uma vez que as arestas são sempre desenhadas com uma linha reta. Contudo, é possível desenhar a seta. Se for analisado em detalhe o grafo ilustrado na figura 5.2b, é possível observar que existem 4 pessoas que comentaram a própria questão que levantaram no fórum. Para efeitos do cálculo do *in-degree* e *out-degree* de um nó com esta característica, como sai uma aresta e volta a entrar para o mesmo nó, é contabilizado adicionalmente o valor 1 para o *out-degree* e o valor 1 para o *in-degree*. Para além de uma aresta para ele mesmo, o nó “Toan Tran” apresenta uma aresta para o nó central “Vasilis Daloukas”, logo o seu *out-degree* tem valor 2 e o *in-degree* tem valor 1.

A figura 5.4a representa os vizinhos diretos do nó “Vasilis Daloukas”. Esta funcionalidade é obtida através um duplo clique sobre o nó.

Com esta representação podemos ter uma visão explícita dos 17 nós que compõem o seu grau. A figura 5.4b representa os mesmos vizinhos, mas com a direção dos relacionamentos de/para o nó “Vasilis Daloukas”.

A figura 5.5a e a figura 5.5b representam 2 configurações de visualização do Grafo Social diferentes, em que o grafo ilustrado na primeira figura apresenta o tamanho dos nós proporcional ao valor do seu *in-degree* e o grafo ilustrado na segunda figura apresenta o tamanho dos nós proporcional ao seu *out-degree*, respetivamente.

Na figura 5.5a está destacado o nó “Claire Browne”, que possui o valor 1 para o seu *in-degree* e apresenta um tamanho médio no Grafo Social. Por outro lado, na figura 5.5b o seu tamanho cresceu consideravelmente, porque o valor o seu *out-degree* é 2. Adicionalmente, podemos ver que

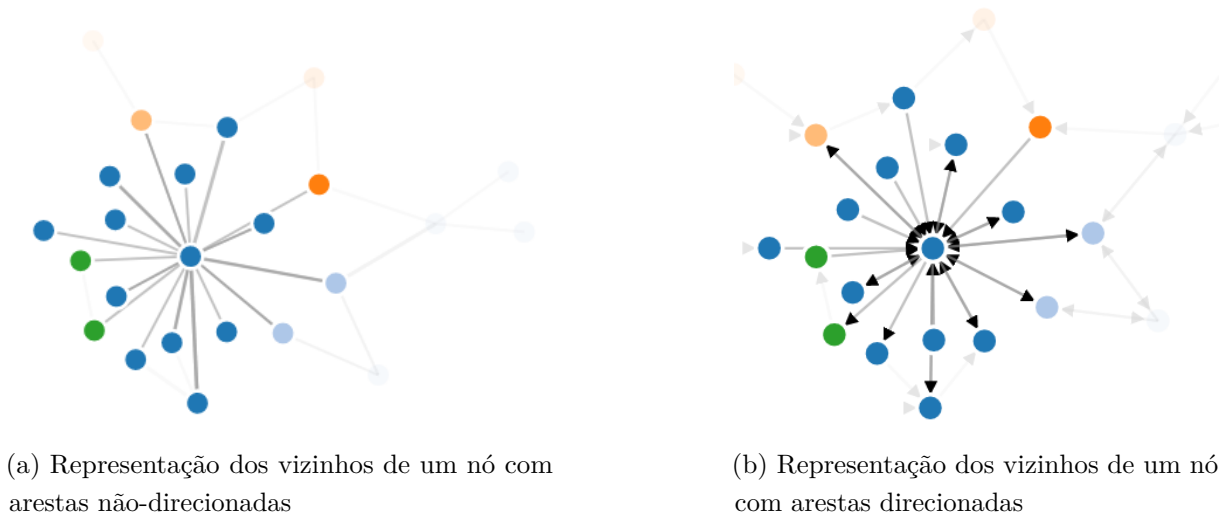


Figura 5.4: Representação dos vizinhos diretos de um nó no Grafo Social

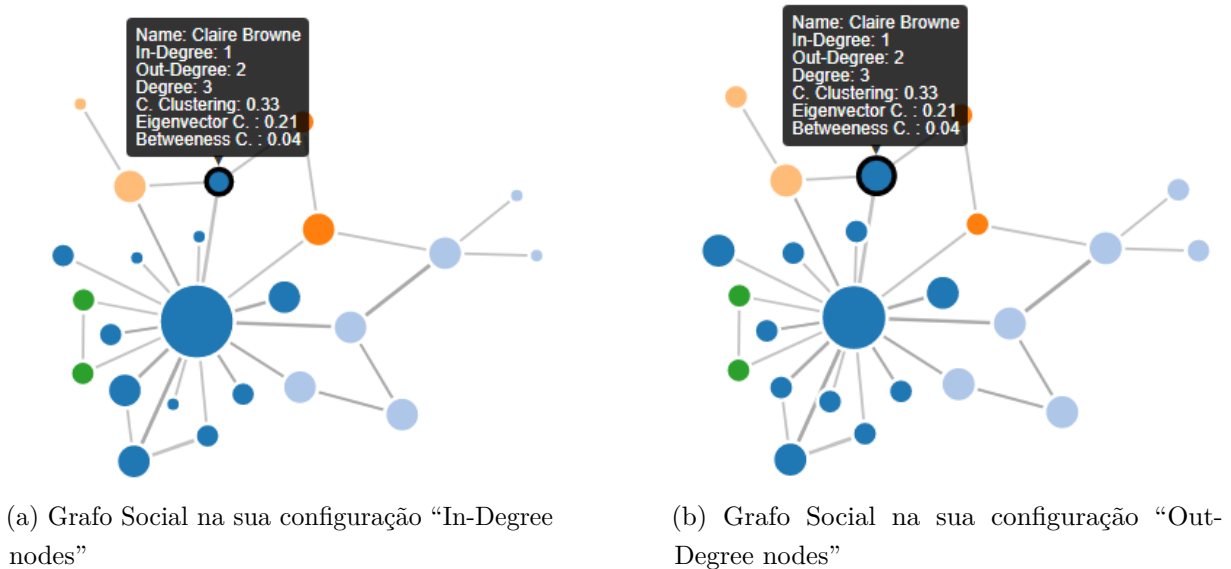


Figura 5.5: Comparação entre as configurações de visualização “In-Degree nodes” e “Out-Degree nodes”

o nó central “Vasilis Daloukas”, analisado anteriormente, diminui o seu tamanho da primeira configuração para a segunda, devido ao facto de o seu *in-degree* ter o valor 14 e o seu *out-degree* o valor 10. Estas duas configurações de visualização do Grafo Social transmitem ao utilizador um *feedback* imediato das pessoas mais comunicativas do fórum e quais são as pessoas que recebem mais atenção, comparando o tamanho dos seus nós no grafo.

5.2 Interação através do Dashboard

Para além de alterar a configuração visual do Grafo Social, procurar nós na rede, e detetar e ver comunidades, a secção do Dashboard permitem também calcular caminhos e verificar a existência de cliques no Grafo Social.

A figura 5.6a ilustra o caminho mais curto entre o nó “Tim Cooper”, desenhado a verde, e o nó “Kerry Enns”, desenhado a vermelho, considerando o Grafo Social direcionado e pesado. A figura 5.6b, por sua vez, ilustra a secção “Shortest path” do Dashboard, para o exemplo descrito anteriormente.

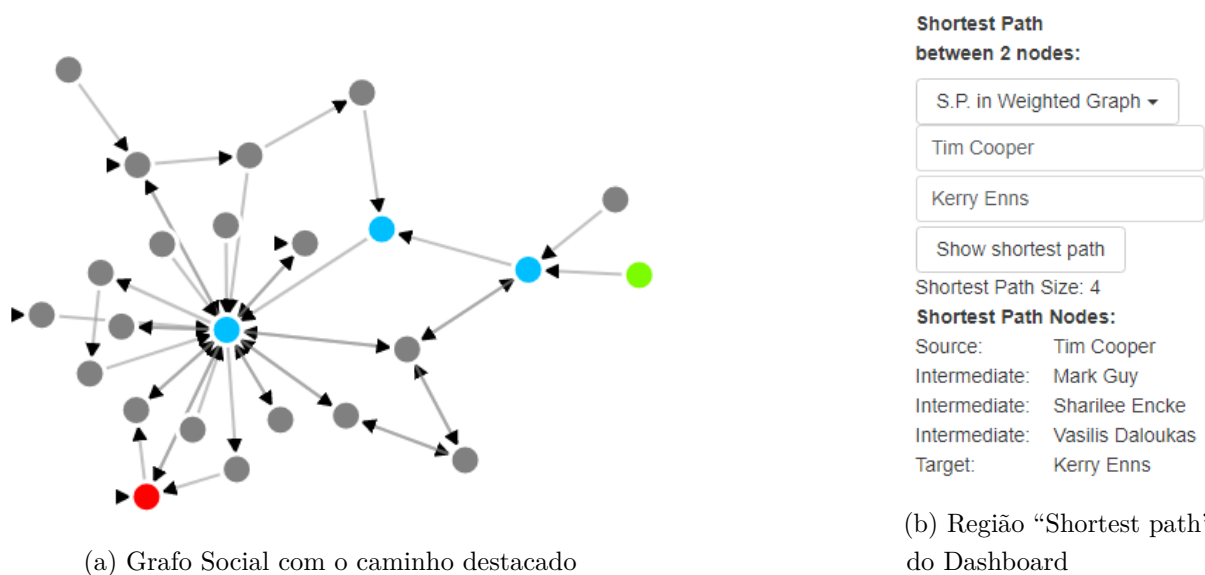


Figura 5.6: Caminho mais curto entre dois nós no Grafo Social direcionado e pesado

É possível observar que o caminho tem tamanho 4 e os seus nós intermédios são “Mark Guy”, “Sharlee Encke” e “Vasilis Daloukas”. Como o nó “Vasilis Daloukas” é uma pessoa influente e central neste fórum, a grande maioria das interações passam por ele, e este caminho não é exceção. Este facto vem reforçar o valor elevado do seu coeficiente de intermediação.

A região “Shortest path” do Dashboard permite, adicionalmente, detetar os nós alcançáveis a partir de um determinado nó na rede, utilizando a opção “Single Source Dijkstra Path”. O grafo ilustrado na figura 5.7a representa todos os nós alcançáveis a partir do nó “Vasilis Daloukas”.

Através da análise do grafo, é possível aferir a grande importância e influencia do nó “Vasilis Daloukas” no fórum, uma vez que este consegue alcançar uma grande parte das pessoas nele presentes. Na totalidade das pessoas que compõem o fórum, apenas não consegue alcançar 7 pessoas. A figura 5.7b representa o mesmo grafo, mas com a visualização da direção das interações entre as pessoas. Nesta figura é possível confirmar todos os caminhos possíveis a partir do nó “Vasilis Daloukas”, e identificar quais os nós terminais no fluxo de informação. Os nós desenhados a cinza, ou se relacionam diretamente com o nó “Vasilis Daloukas” ou se relacionam com outros nós periféricos, não sendo alcançáveis a partir deste.

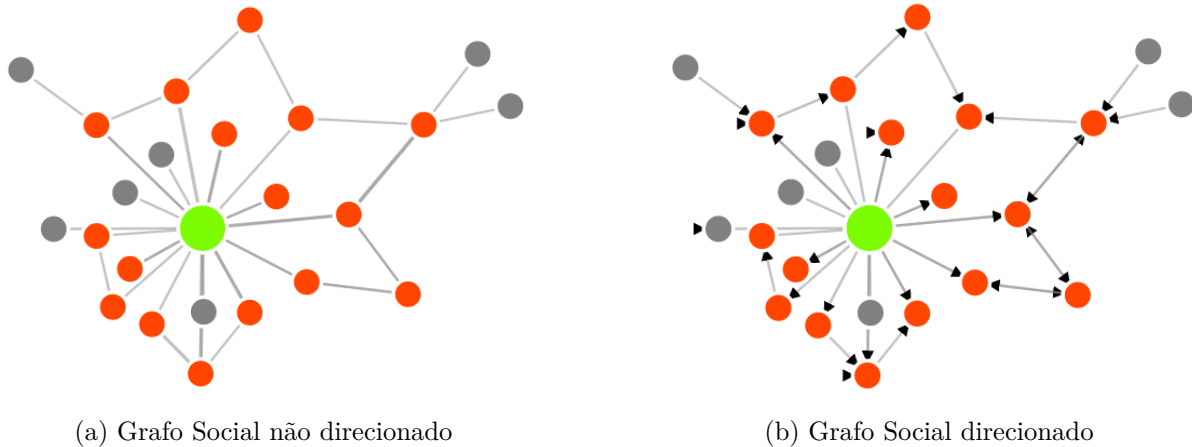


Figura 5.7: Nós alcançáveis a partir de um nó no Grafo Social

A última região do Dashboard, ilustrada na figura 5.8, permite ao utilizador verificar a existência de cliques no Grafo Social, que possuem um tamanho específico. No contexto deste exemplo prático, um clique consiste num grupo coeso de pessoas, na qual cada pessoa tem uma ligação para todas outras pessoas do grupo. Os cliques são calculados tendo em conta o grafo como sendo não-direcionado, ou seja, apenas a existência ou não de uma ligação entre os nós tem impacto, e não a direção dos seus relacionamentos.

Choose a clique size
between 2 and 3:

Clique size

Show clique

Figura 5.8: Região “Cliques” do Dashboard

Através do rótulo da região, podemos ver que apenas existem cliques de tamanho 2 e de tamanho 3 no Grafo Social, logo o tamanho do maior clique é 3. A figura 5.9b representa esta região, após se ter inserido o tamanho 3 na caixa de inserção de texto. De seguida, o utilizador é informado que existem 4 cliques de tamanho 3 no Grafo Social, e os nós que compõem o primeiro clique com este tamanho são: “Vasilis Daloukas”, “Claire Browne” e “Alissa Parrish”.

O grafo ilustrado na figura 5.9b apresenta os nós que compõem o primeiro clique destacados. Através dos botões de navegação presentes na região dos cliques do Dashboard, o utilizador pode iterar sobre os cliques que possuem o tamanho 3, e verificar que o nó “Vasilis Daloukas” está presente em todos os cliques, reforçando deste modo a sua influência na rede.

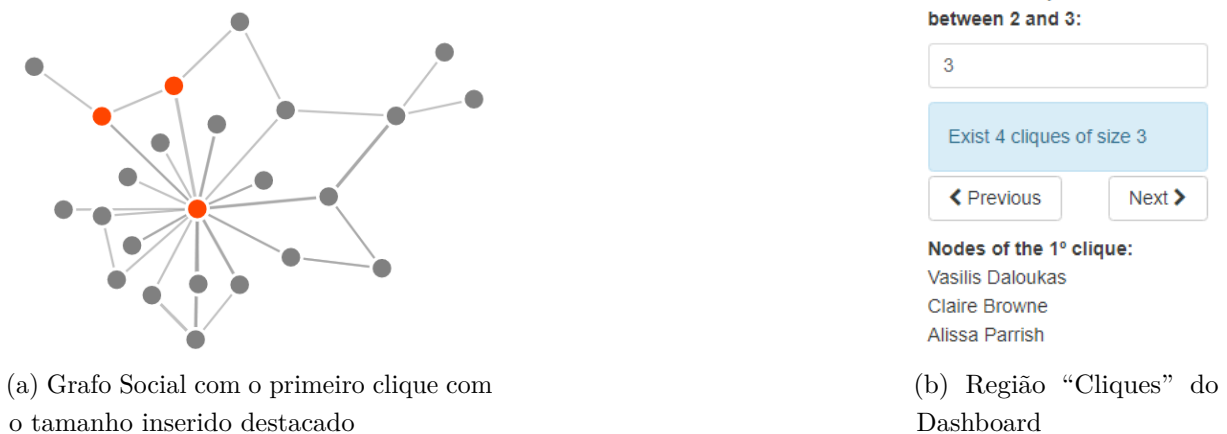


Figura 5.9: Detecção de cliques no Grafo Social com um tamanho específico

5.3 Análise das propriedades do Grafo Social

Através da secção das propriedades e métricas do Grafo social, o utilizador pode ver propriedades estáticas do grafo, ou seja, valores que não se alteram com o tempo ou com as interações com o Grafo Social. Na primeira coluna, é possível ver que o grafo é composto por 24 pessoas, que mantém 45 relacionamentos entre elas, e que existem 5 comunidades.



Figura 5.10: Propriedades do Grafo Social

Na segunda coluna, é apresentada a densidade do Grafo Social, com o valor 0.12. A densidade mede a proporção entre as arestas presentes no grafo e o número máximo de arestas que o grafo pode ter, ou seja, um grafo completo. O valor da densidade varia entre 0 e 1 e o valor do Grafo Social é próximo de 0, o que nos indica que o grafo é esparso e pouco coeso. Pelo grafo da figura 5.1a, é possível ver que o nó com o nome “Vasilis Daloukas” é um nó central do grafo, e a grande maioria dos nós interage maioritariamente com ele. Deste modo, a grande parte dos relacionamentos do grafo estão centrados maioritariamente num nó, e os restantes nós interagem pouco entre si, o que justifica o valor baixo da sua densidade. Abaixo do valor da densidade, são apresentados os valores do número de cliques máximos: 23, e o tamanho do maior clique: 3. A biblioteca JSNetworkX.js considera cliques de tamanho superior ou igual a 2, e é por esse motivo que é apresentado um número de cliques máximos tão elevado. O rótulo da região “Cliques” do Dashboard, presente na figura 5.9b, indica que existem cliques de tamanho 2 e tamanho 3 no Grafo Social, logo 3 é o tamanho do maior clique. Adicionalmente, pelo exemplo da figura 5.9b,

podemos ver que existem 4 cliques de tamanho 3, logo existem 19 cliques de tamanho 2, que tem pouca importância na formação de relações coesas entre indivíduos.

A terceira coluna é composta por médias de valores de centralidade do Grafo Social. O primeiro valor representa a média do grau dos nós, com o valor 2.85. Este valor indica que em média, cada pessoa se relaciona aproximadamente com 3 pessoas. O segundo valor apresenta a média do coeficiente de agrupamento, com o valor 0.22. Como o Grafo Social é altamente centralizado no nó “Vasilis Daloukas”, os nós do grafo não tendem a agrupar-se e a formar grupos coesos caracterizados por uma alta densidade. Em resultado, o coeficiente de agrupamento apresenta um valor baixo.

5.4 Discussão

Neste capítulo foi analisado um exemplo prático da aplicação do sistema proposto, que compreende a visualização e a análise da interação entre 24 pessoas num fórum. Nesta secção pretendemos responder a algumas questões que ficaram em aberto na análise feita e tirar algumas conclusões.

A discussão apresentada neste fórum foi criada pelo utilizador “Vasilis Daloukas”. Neste discussão, “Vasilis Daloukas” apresenta um módulo para o Moodle que ele próprio criou, capaz de executar vários jogos. Vários outros utilizadores comentaram este módulo e ofereceram *feedback* e conselhos ao criador do Módulo, de modo a que ele pudesse melhorá-lo.

Em qualquer discussão dentro de um grupo, existem pessoas mais comunicativas e pessoas que recebem mais atenção. Através da configuração visual do Grafo Social em que o tamanho dos nós é proporcional ao valor do seu *in-degree*, conseguimos saber quais são as pessoas que recebem mais atenção no fórum. Através desta configuração, identificamos “Vasilis Daloukas” como a pessoa que recebe mais atenção no fórum, uma vez que apresenta o maior *in-degree*, com o valor 14. Este valor significa que recebeu vários comentários e sugestões por parte dos outros membros do fórum. Para além de “Vasilis Daloukas”, as pessoas que recebem mais atenção são: “Alissa Parrish”, “Mark Guy”, “John Andrewartha” e “Kerry Enns”, com *in-degree* igual a 3.

Por outro lado, através da configuração visual do Grafo Social em que o tamanho dos nós é proporcional ao valor do seu *out-degree*, conseguimos identificar as pessoas mais comunicativas do fórum. Para além de ser, de forma destacada, a pessoa que recebe mais atenção no fórum, “Vasilis Daloukas” é também a pessoa mais comunicativa, com um valor de *out-degree* igual a 10, indicando deste modo ser uma pessoa ativa na discussão, fortalecendo o seu papel central na rede. Com um valor claramente inferior, “Alissa Parrish”, “Kerry Enns” e “John Andrewartha” posicionam-se em segundo lugar com um valor de *out-degree* igual a 3.

O sistema detetou 5 comunidades para a rede deste fórum, destacadas no Grafo Social com cores diferentes. Os nós que possuem a mesma cor, pertencem à mesma comunidade. Para a deteção destas comunidades foi aplicado o algoritmo “Clauaset, Newman and Moore” ao Grafo Social, que se baseia na otimização do conceito de modularidade que consiste numa medida

estrutural do grafo que mede a força da divisão da rede em grupos ou módulos. Também era possível utilizar um algoritmo de análise semântica das mensagens trocadas no fórum. Deste modo, as pessoas que falam do mesmo tema ou tópico iriam pertencer à mesma comunidade.

A análise visual da rede reflete a centralização da discussão em torno de “Vasilis Daloukas” e, considerando a natureza da discussão, todas as respostas são focadas em comentar o módulo criado por ele. Desta forma, o Grafo Social apresenta valores relativamente baixos para a sua média do coeficiente de agrupamento e densidade, uma vez que não houve a formação de grupos e o facto de a discussão ser focada no mesmo membro o tempo todo contribui para uma rede pouco densa e coesa. Se a rede não tivesse um nível de centralização em “Vasilis Daloukas” tão elevado, e os restantes nós interagissem mais entre si, os valores da densidade e coeficiente de agrupamento seriam claramente superiores.

Capítulo 6

Conclusões

No presente capítulo, é feita uma síntese de todo o trabalho desenvolvido no decorrer desta tese de mestrado uma conclusão das suas contribuições em contexto pedagógico. Por fim, é feita uma lista de propostas de um trabalho futuro, de modo a tornar o sistema proposto mais completo, robusto e versátil.

6.1 Resumo do trabalho desenvolvido

Neste relatório, apresentamos um módulo de visualização e análise de interações online de pessoas e recursos, integrado no Learning Management System (LMS) Moodle. A representação das interações é feito com recurso a um grafo, e apresentado na página de um curso do Moodle, de acordo com os dados do ambiente de ensino-aprendizagem virtual do curso em questão. Os dados de entrada do grafo são armazenados na base de dados do Moodle, em duas tabelas. A primeira tabela guarda todas as informações dos nós do grafo e a segunda tabela guarda as informações das arestas do grafo.

A biblioteca D3.js é responsável pelo desenho automático do Grafo Social utilizando um “Force-Direct” layout, quando se acede à página de um curso. Para este efeito, é feito o *parsing* de um ficheiro JavaScript Object Notation (JSON) que contém a lista dos nós e a listas das arestas do grafo. Para a construção deste ficheiro, são utilizados métodos da Application Programming Interface (API) de manipulação de dados do Moodle, de modo a obter as informações necessárias armazenadas na base de dados do Moodle.

A biblioteca netClustering.js permite aplicar o algoritmo de deteção de comunidades “Clauset, Newman and Moore” ao Grafo Social, alterando a cor dos nós de modo a que os nós que pertençam à mesma comunidade possuam a mesma cor.

A biblioteca JSNetworkX.js fornece todos os métodos e algoritmos de Social Network Analysis (SNA) presentes no painel de controlo, nomeadamente na secção do Dashboard e na secção das propriedades do grafo. De seguida apresentamos as funcionalidades mais relevantes e que foram

implementadas no sistema proposto, com recurso à [API](#) da desta biblioteca:

- Criar grafos direcionados e não direcionados;
- Calcular o número de nós, o número de arestas e o número de cliques máximos do grafo;
- Calcular a densidade, e medidas de centralidade locais e globais, tais como o *in-degree*, o *out-degree*, o grau, o coeficiente de agrupamento, a Eigenvector Centrality e o coeficiente de intermediação;
- Identificação de cliques no grafo;
- Calculo de caminhos mais curtos entre dois nós em grafos direcionados pesados e em grafos não direcionados;
- Identificação dos nós alcançáveis a partir de um nó.

O sistema compreende a integração de todos os componentes referidos anteriormente. Em resultado, é apresentado na página de um curso do Moodle um painel de controlo dividido em 3 secções principais: a área de desenho do Grafo Social, o Dashboard e as Propriedades do grafo.

6.2 Contribuições

O sistema proposto permite representar as interações ou recursos dentro de um ambiente de ensino aprendizagem, fazendo o desenho automático de um Grafo Social. Esta representação fornece uma interpretação rápida e clara dos relacionamentos criados. A aplicação de métodos e técnicas de [SNA](#) permite identificar propriedades importantes da rede para a identificação e análise da turma, detetar estruturas e perceber a dinâmica da rede.

O ponto principal a referir é que [SNA](#) e a correspondente representação gráfica não está restrita às interações entre pessoas. Apesar de este ser o caso tradicional, como o caso deste exemplo de estudo, em muitas situações de Eletronic Learning ([e-Learning](#)) existem relações latentes entre entidades diferentes que, através de uma representação gráfica, emergem como tão importantes e perspicazes (um bom exemplo é [\[29\]](#)).

A representação do caminho mais curto entre dois nós pode servir vários propósitos educativos. Por exemplo, se as interações forem entre estudantes, mostra elementos envolvidos na conversação, por outro lado, se o grafo representa recursos e a duração do seu uso, então o caminho pode exibir a ordem em que os recursos foram “consumidos” e quanto tempo os estudantes gastaram a usá-los.

A deteção de cliques permite identificar quem grupos coesos de estudantes na qual todos comunicam com todos dentro do grupo, medir a proximidade estas interações e identificar quem é que interage com quem.

Outra questão importante para enfatizar é que outros estudos representaram e mostraram interações online que ocorreram em atividades do Moodle, nomeadamente em atividades de fórum ([64], [65]), mas isso também é possível ser alargado às redes sociais que são usadas em contextos pedagógicos, como mostrado em [31]. No entanto, a teoria por detrás do desenvolvimento do sistema proposto é sólida, as representações gráficas são genéricas e, para além disso, existe uma variedade fontes de dados de que podem ser usadas pelo nosso sistema para serem analisadas pela ferramenta proposta.

De modo a documentar e dar crédito ao trabalho desenvolvido, foi redigido um *short paper* intitulado “A System for Visualization and Analysis of Online Pedagogical Interactions”, no qual descrevemos o sistema desenvolvido até ao momento, suas contribuições e propostas para o trabalho futuro. Este paper foi submetido, aceite e apresentado durante a “International Conference on E-Education, E-Business and E-Technology (ICEBT 2017)”, realizada ente 10 a 12 de Setembro em Toronto, Canadá. Posteriormente, o paper foi publicado, e encontra-se disponível na plataforma ACM Digital Library.

Em suma, o sistema permite combinar as funcionalidades de um LMS para propósitos educativos com as técnicas de SNA para o estudo das relações humanas em ambientes virtuais de ensino-aprendizagem. O sistema proposto é uma mais valia para que professores e estudantes recebem o melhor diagnóstico educativo do seu ambiente de ensino-aprendizagem, descobrindo deste modo novas oportunidades e fraquezas. Podemos concluir, deste modo, que o sistema proposto cumpriu com sucesso os objetivos assumidos no início deste desenvolvimento.

6.3 Trabalho futuro

Conforme mostrado em [22], integrar SNA e plataformas de e-Learning é uma área que requer pesquisa. O sistema proposta ainda está numa fase inicial, por isso ambicionamos, num trabalho futuro, implementar novas funcionalidades importantes.

A API da biblioteca JSNetworkX.js ainda está numa fase inicial, por isso é expectável que ao longo do tempo surjam novos algoritmos e métodos de SNA, que podem ser integrados no sistema proposto.

Também é possível implementar análise temporal do grafo, de modo a visualizar a evolução da rede ao longo do tempo, comparando o estado da rede entre os períodos temporais capturados.

A biblioteca D3.js contém imensos componentes de visualização poderosos, tais como gráficos temporais e sequenciais, histogramas ou outros layouts de grafos, que podem facilmente ser introduzidas no sistema a fim de implementar novas funcionalidades, tais como estatísticas ou sistemas de *ranking*.

Finalmente, pretendemos tornar o sistema o mais genérico possível. O modelo de dados é passível de ser alterado a fim de poder receber dados de várias fontes, como redes sociais, grupos, blogs ou fóruns.

Bibliografia

- [1] Csv format. <https://gephi.org/users/supported-graph-formats/csv-format/>. Acedido em: 29/05/2017.
- [2] Gexf file format. <https://gephi.org/gexf/format/>. Acedido em: 29/05/2017.
- [3] Alchemy.js. <https://graphalchemist.github.io/Alchemy/>. Acedido em: 19/12/2016.
- [4] Arangodb. <https://www.arangodb.com/>. Acedido em: 21/12/2016.
- [5] Cytoscape. <https://www.cytoscape.org/>, . Acedido em: 17/12/2016.
- [6] Cytoscape.js. <https://js.cytoscape.org/>, . Acedido em: 19/12/2016.
- [7] D3.js. <https://d3js.org/>. Acedido em: 18/12/2016.
- [8] Gephi. <https://gephi.org/>. Acedido em: 15/12/2016.
- [9] Graphviz. <https://www.graphviz.org/>. Acedido em: 17/12/2016.
- [10] Jung. <https://jung.sourceforge.net/>. Acedido em: 18/12/2016.
- [11] Neo4j. <https://neo4j.com/>. Acedido em: 21/12/2016.
- [12] Oracle rdbms. <https://www.oracle.com/database/index.html>. Acedido em: 21/12/2016.
- [13] Pajek. <https://pajek.imfm.si/>. Acedido em: 15/12/2016.
- [14] Sigma.js. <https://sigmajs.org/>. Acedido em: 18/12/2016.
- [15] Tulip. <https://www.tulip.software.org/>. Acedido em: 17/12/2016.
- [16] Tasleem Arif. The mathematics of social network analysis: Metrics for academic social networks. *Int. J. Comput. Appl. Technol. Res.*, 4(12):889–892, 2015.
- [17] David Auber. Tulip—a huge graph visualization framework. In *Graph drawing software*, pages 105–126. Springer, 2004.
- [18] Vladimir Batagelj and Andrej Mrvar. Pajek-program for large network analysis. *Connections*, 21(2):47–57, 1998.

-
- [19] Stephen P Borgatti, Martin G Everett, and Jeffrey C Johnson. *Analyzing social networks*. SAGE Publications Limited, 2013.
- [20] Ulrik Brandes, Markus Eiglsperger, Jürgen Lerner, and Christian Pich. Graph markup language (graphml). *Handbook of graph drawing and visualization*, 20007:517–541, 2013.
- [21] N. Cavus and T. Zabadi. A comparison of open source learning management systems. *Procedia-Social and Behavioral Sciences*, 143, pages 521–526, 2014.
- [22] Sicilia M. Á. Sánchez S. Cela, K. L. Social network analysis in e-learning environments: A preliminary systematic review. *Educational Psychology Review*, 27(1), pages 219–246, 2015.
- [23] Aaron Clauset, Mark EJ Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical review E*, 70(6):066111, 2004.
- [24] Carlos D. Correa and Kwan-Liu Ma. *Visualizing Social Networks*. Springer, 2011.
- [25] Matthew Denny. Social network analysis. *Institute for Social Science Research, University of Massachusetts Amherst*, Friday 26th September, 2014.
- [26] P. Eades. A heuristic for graph drawing. *Congressus Numerantium*, 42:149–160, 1984.
- [27] Geoffrey Fairchild and Jason Fries. *Social networks: Models, algorithms, and applications*. 2012.
- [28] A Figueira. Predicting results from interaction patterns during online group work. *In Design for Teaching and Learning in a Networked World (pp. 414-419)*, 2015.
- [29] Álvaro Figueira. Communication and resource usage analysis in online environments: An integrated social network analysis and data mining perspective. pages 1027–1032, 2017.
- [30] Á. Figueira and J. Laranjeiro. Work in progress—igraphs for characterization of online communities. *FIE 2008. 38th Annual (pp. F4E-13)*. *IEEE.*, 2008, October.
- [31] Álvaro Figueira and Luciana Oliveira. Edubridge social - bridging social networks and learning management systems. pages 162–171, 2016.
- [32] Linton C Freeman. A set of measures of centrality based on betweenness. *Sociometry*, pages 35–41, 1977.
- [33] Linton C Freeman. Centrality in social networks conceptual clarification. *Social networks*, 1(3):215–239, 1978.
- [34] Linton C Freeman. The gatekeeper, pair-dependency and structural centrality. *Quality & Quantity*, 14(4):585–592, 1980.
- [35] Thomas MJ Fruchterman and Edward M Reingold. Graph drawing by force-directed placement. *Software: Practice and experience*, 21(11):1129–1164, 1991.

- [36] Neveen Ghali, Mrutyunjaya Panda, Aboul Ella Hassanien, Ajith Abraham, and Vaclav Snasel. Social networks analysis: Tools, measures and visualization. pages 3–23, 2012.
- [37] Daniel Z Grunspan, Benjamin L Wiggins, and Steven M Goodreau. Understanding classrooms through social network analysis: a primer for social network analysis in education research. *CBE-Life Sciences Education*, 13(2):167–178, 2014.
- [38] Robert A. Hanneman and Mark Riddle. [Introduction to social network methods](#). Online, 2005. Acedido em 11/12/2016.
- [39] Nathalie Henry and Jean-Daniel Fekete. Matlinl: enhanced matrix visualization for analyzing social networks. *INTERACT'07: Proceedings of the 11th IFIP TC 13 international conference on Human-computer interaction*, pages 288–302, 2007.
- [40] Nathalie Henry, Jean-Daniel Fekete, and Michael J. McGuffin. Nodetrix: a hybrid visualization of social networks. *IEEE Transactions on Visualizarion and Computer Graphics*, pages 1302–1309, 2007.
- [41] Michael Himsolt. Gml: A portable graph file format. *Html page under http://www.fmi.uni-passau.de/graphlet/gml/gml-tr.html*, Universität Passau, 1997.
- [42] Mohsen Jamali and Hassan Abolhassani. Different aspects of social network analysis. In *2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2006 Main Conference Proceedings)(WI'06)*, pages 66–72. IEEE, 2006.
- [43] Tomihisa Kamada and Satoru Kawai. An algorithm for drawing general undirected graphs. *Information processing letters*, 31(1):7–15, 1989.
- [44] Hyunmo Kang, Lise Getoor, and Lisa Singh. Visual analysis of dynamic group membership in temporal social networks. *ACM SIGKDD Explorations Newsletter*, 9(2):13–21, 2007.
- [45] Leo Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.
- [46] R Duncan Luce and Albert D Perry. A method of matrix analysis of group structure. *Psychometrika*, 14(2):95–116, 1949.
- [47] Michael J McGuffin. Simple algorithms for network visualization: A tutorial. *Tsinghua Science and Technology*, 17(4):383–398, 2012.
- [48] James Moody, Daniel McFarland, and Skye Bender-deMoll. Dynamic network visualization1. *American journal of sociology*, 110(4):1206–1241, 2005.
- [49] James Moody, Daniel McFarland, and Skye Bender-deMoll. Dynamic network visualization. *American journal of sociology*, 110(4):1206–1241, 2005.
- [50] Chris Muelder and Kwan-Liu Ma. Rapid graph layout using space filling curves. *IEEE Transactions on Visualizarion and Computer Graphics*, pages 1301–1308, 2008.

- [51] Mayur Narkhede, ST Patil, and Vrushali Inamdar. Comparative study of various graph layout algorithms. *International Journal of Emerging Trends Technology in Computer Science (ITETTCS)*, 3, March - April 2014.
- [52] Takao Nishizeki and Md Saidur Rahman. *Planar graph drawing*, volume 12. World Scientific Publishing Co Inc, 2004.
- [53] Luciana Oliveira. Social student relationship management in higher education: extending educational and organisational communication into social media. *9th Annual International Technology, Education and Development Conference*, 2015.
- [54] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. 1999.
- [55] Helen C. Purchase. Effective information visualisation: a study of graph drawing aesthetics and algorithms. *Interacting with computers*, 13(2):147–162, 2000.
- [56] Ijaz A Qureshi, Hassan Raza, and Michael Whitty. Facebook as e-learning tool for higher education institutes. *Knowledge Management & E-Learning: An International Journal (KM&EL)*, 6(4):440–448, 2015.
- [57] Khairi Reda, Chayant Tantipathananandh, Andrew Johnson, Jason Leigh, and Tanya Berger-Wolf. Visualizing the evolution of community structures in dynamic social networks. 30(3):1061–1070, 2011.
- [58] Peter Rodgers. Graph drawing techniques for geographic visualization. *Exploring geovisualization*, pages 143–158, 2005.
- [59] Martin Rosvall and Carl T Bergstrom. Mapping change in large networks. *PloS one*, 5(1): e8694, 2010.
- [60] John Scott. *Social network analysis*. Sage, 2012.
- [61] Paul Shannon, Andrew Markiel, Owen Ozier, Nitin S Baliga, Jonathan T Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome research*, 13(11):2498–2504, 2003.
- [62] Zeqian Shen and Kwan-liu Ma. Path visualization for adjacency matrices. *Euro Vis*, pages 83–90, 2006.
- [63] Zeqian Shen and Kwan-Liu Ma. Mobivis: A visualization system for exploring mobile data. In *2008 IEEE Pacific Visualization Symposium*, pages 175–182. IEEE, 2008.
- [64] A. Silva and Á. Figueira. Depicting online interactions in learning communities. *International Business and Management*, In Global Engineering Education Conf. (EDUCON), 2012 (pp. 1-8). IEEE, 2012, April.

-
- [65] A. Silva and Á. Figueira. Visual analysis of online interactions through social network patterns. In *Advanced Learning Technologies (ICALT), 2012 IEEE 12th International Conference on* (pp. 639-641). IEEE, 2012, July.
- [66] Carsten Thomassen. Kuratowski's theorem. *Journal of Graph Theory*, 5(3):225–241, 1981.
- [67] William Thomas Tutte. How to draw a graph. *Proceedings of the London Mathematical Society*, 3(1):743–767, 1963.
- [68] Renata Vadera, Gordana Milosavljević, and Igor Dejanović. Graph layout algorithms and libraries: Overview and improvements. pages 55–60, 2015.
- [69] Fernanda B Viégas, Danah Boyd, David H Nguyen, Jeffrey Potter, and Judith Donath. Digital artifacts for remembering and storytelling: Posthistory and social network fragments. In *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*, pages 10–pp. IEEE, 2004.
- [70] Chounta I.A. Ziebarth, S. and H.U. Hoppe. Resource access patterns in exam preparation activities. In *Design for Teaching and Learning in a Networked World* (pp. 497-502), 2015.