UNIVERSIDADE FEDERAL DO PAMPA

PÂMELA DA CUNHA SARAIVA

IDENTIFICAÇÃO DAS REDES CRIMINAIS DE BAGÉ

Bagé

2016

PÂMELA DA CUNHA SARAIVA

IDENTIFICAÇÃO DAS REDES CRIMINAIS DE BAGÉ

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia de Computação da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Computação.

Orientador: Prof. Dr. Sandro da Silva Camargo

Coorientador: Prof. MSc. Carlos Michel Betemps

Bagé

2016

Ficha catalográfica elaborada automaticamente com os dados fornecidos
pelo(a) autor(a) através do Módulo de Biblioteca do
Sistema GURI (Gestão Unificada de Recursos Institucionais) .

S243i Saraiva, Pâmela da Cunha Saraiva
Identificação das Redes Criminais de Bagé / Pâmela da Cunha Saraiva.
114 p.

Trabalho de Conclusão de Curso(Graduação)-- Universidade Federal do Pampa, ENGENHARIA DE COMPUTAÇÃO, 2016.
"Orientação: Sandro da Silva Camargo".

1. Análise de Redes Sociais. 2. Processamento de Linguagem Natural. I. Título.

PÂMELA DA CUNHA SARAIVA

IDENTIFICAÇÃO DAS REDES CRIMINAIS DE BAGÉ

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia de Computação da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Computação.

Trabalho de Conclusão de Curso defendido e aprovado em: 04 de julho de 2016.

Banca examinadora:

Prof. Dr. Sandro da Silva Camargo

Orientador

UNIPAMPA

Prof.ª Dr.ª Ana Paula Lüdtke Ferreira

UNIPAMPA

Prof.ª Drª. Sandra Dutra Piovesan

UNIPAMPA

AGRADECIMENTOS

Aos meus orientadores, pelas correções e incentivo durante o trabalho.

À minha família por todo amor e apoio incondicional.

Ao meu namorado, pelo apoio e ajuda nos momentos difíceis.

À Brigada Militar de Bagé, pela disponibilização dos dados essenciais para a realização desse trabalho.

E a todos que direta ou indiretamente contribuíram para a realização desse trabalho, muito obrigada.

RESUMO

Com o crescimento da criminalidade e, consequentemente, aumento da quantidade de Boletins de Ocorrência registrados nos órgãos de segurança pública, uma grande quantidade de dados sobre ações criminosas vem sendo gerada a uma velocidade maior que os analistas de segurança conseguem explorar. Além disso, existe uma certa tendência desses criminosos a agirem de forma conjunta e organizada na execução dos crimes. Devido a isso, este trabalho propõe a utilização de uma técnica de Análise de Redes Sociais para possibilitar a análise automatizada da grande quantidade de ocorrências digitais registradas nos órgãos de segurança de uma cidade. Com a utilização dessa técnica, e com base em dados reais disponibilizados por um órgão de segurança pública, foi realizada a identificação e análise da rede social criminal da cidade de Bagé. Além disso, foi desenvolvida uma ferramenta que seguirá auxiliando órgãos de segurança pública na identificação de redes criminais no futuro.

Palavras-Chave: Análise de Redes Sociais, segurança pública, rede criminal, Processamento de Linguagem Natural, *Gephi*.

ABSTRACT

With the growth of crime and consequently increase of the number of police reports recorded in the public security agencies a lot of data on criminal actions is being generated at a higher speed than the security analysts can explore. Also, there is a tendency that these criminals act jointly and organized manner in the execution of crimes. Due to this, this paper proposes the use of a technique on Social Network Analysis to enable automated analysis of a large number of incidents recorded in digital security agencies of a city. Using this technique, and based on actual data provided by a public security organ, it was carried out the identification and analysis of criminal social network of the city of Bagé. In addition, it was developed a tool that will continue assisting law enforcement agencies in the identification of criminal networks in the future.

Keywords: Social Network Analysis, public security, criminal network, Natural Language Processing, Gephi.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AEF – Autômato de Estado Finito

AP - Application Programming Interface

BM – Brigada Militar

BO – Boletim de ocorrência

CNA – *Criminal Network Analysis*

CoNLL - *Conference on Computational Natural Language Learning*

COR – Número de respostas corretas

CPU – (*Central Processing Unit*)

FBI - *Federal Bureau of Investigation*

GLC – Gramática Livre de Contexto

LN – Linguagem Natural

MUC - *Message Understanding Conference*

NER – *Named Entity Recognition*

NERC - *Named Entity Recognition and Classification*

PLN – Processamento de Linguagem Natural

SNA – *Social Network Analysis*

TA – Tradução Automática

TCC – Trabalho de Conclusão de Curso

SUMÁRIO

1 INTRODUÇÃO

Com o crescente desenvolvimento da população, os indivíduos vêm formando diversos tipos de sociedades e, com a organização dessas sociedades, muitos dados vêm sendo produzidos. Devido à necessidade de manipulação dessa grande quantidade de dados, surgiu um campo de pesquisa chamado *Social Network Analysis,* ou Análise de Redes Sociais (SNA). Esse campo de pesquisa está focado mais na estrutura dos relacionamentos dos indivíduos da rede social do que nos indivíduos em si. Ele provê uma abordagem onde se pode abstrair e representar um fenômeno de interação e relacionamento entre unidades. Segundo Rostami (2015), SNA tem sido usada para lidar com uma variedade de aplicações de pesquisa, como controle de doenças sexualmente transmissíveis, prevenção de mudanças da população após desastres naturais, assistência em operações humanitárias e, particularmente, vem crescendo na área de análise de redes criminais. Todos esses campos de pesquisa têm como foco um grupo de pessoas.

Há muito tempo tem sido evidenciado que pessoas propensas a cometer crimes raramente trabalham de forma isolada, mas agem como parte de uma rede criminal (Haynie, 2001, *apud* Seidler *et al*, 2013). Com os avanços significativos dos meios de comunicação, transporte e a expansão da tecnologia em geral, as redes criminais vêm se tornando um fenômeno global. Pesquisadores da área têm concluído que as organizações criminais do século XXI são caracterizadas por um amplo grau de fluidez e complexidade estrutural. Com isso, técnicas de SNA vem sendo utilizadas na análise de redes criminais e, nesse campo, as aplicações vão desde o estudo da criminalidade em geral até o monitoramento de redes terroristas, crime organizado e gangues de rua. O conhecimento contido em dados de redes sociais pode ser utilizado como um recurso importante nas investigações criminais, pois pode ajudar a descobrir a estrutura e a forma de organização de uma rede criminal. Porém, a grande quantidade de dados obtidos pelos órgãos de segurança muitas vezes dificulta sua análise, além de ser uma tarefa que demanda muito tempo, caso seja feita de forma manual. Além disso, quando a quantidade de indivíduos atuantes na rede é muito grande, é mais difícil de determinar seus criminosos importantes.

Nesse contexto, o presente trabalho visa propor uma abordagem automática para criação da rede criminal da cidade, a partir de dados obtidos dos sistemas de informação de um órgão de segurança local. Como resultado desta pesquisa, espera-se permitir a geração da rede criminal com o mínimo de intervenção humana e, a partir desta rede, dar aos órgãos de segurança pública um conhecimento mais claro a respeito das organizações criminais atuantes na cidade e, consequentemente, melhores condições de combater o crime organizado.

1.1 Objetivos

1.1.1 Objetivo Geral

O objetivo geral deste trabalho é desenvolver uma abordagem automática para a geração da rede criminal de uma cidade a partir dos dados oriundos de ocorrências criminais dos sistemas de informação de órgãos de segurança pública, para contribuir no combate ao crime organizado.

1.1.2 Objetivos Específicos

- Obter dados digitais de boletins de ocorrência da BM da cidade de Bagé-RS;
- Escolher uma ferramenta de SNA para montar as redes sociais criminais;
- Implementar uma ferramenta para automatizar o processo de geração de redes sociais criminais a partir de dados de boletins de ocorrência;
- Determinar algoritmicamente quais são os criminosos chave por meio de métricas de Análise de Redes Sociais;
- Fazer um estudo de caso com dados reais obtidos com a BM de Bagé;

1.2 Metodologia

A análise do estado da arte foi feita por meio de uma pesquisa exploratória por trabalhos correlatos publicados em eventos e periódicos disponíveis em plataformas online, tais como as Bibliotecas Digitais da IEEE, ScienceDirect,

Springer e ACM, além do Portal de Periódicos da CAPES. Ademais, foram utilizados livros que abordem os assuntos pesquisados. Também foram feitas pesquisas experimentais e de campo.

Com relação à obtenção dos dados, a BM de Bagé disponibilizou dois conjuntos de dados, que foram utilizados em dois momentos diferentes. Em um primeiro momento foi disponibilizado um conjunto de dados que foi utilizado para uma montagem e análise preliminar de uma rede social. Em um segundo momento, foi disponibilizado um novo conjunto de dados, um pouco maior, e mais completo. Esse conjunto foi utilizado para fazer montagens e análises mais aprofundadas das redes, além de montar uma rede criminal georreferenciada. Foram realizadas, também, diversas reuniões junto à BM para a solicitação de dados, discussão de requisitos e análise de resultados.

Além disso, foram estudados algoritmos de Processamento de Linguagem Natural (PLN), com foco em técnicas de *Named-Entity Recognition* (NER), que é um ramo desta área que tem como tarefa a classificação e busca de expressões em textos de linguagem natural (Silveira, 2014). Pois, antes de ser estabelecido o contato com a BM, acreditava-se que as ocorrências estariam registradas em texto puro. Porém, após ser estabelecido o contato, foi constatado que os dados não estavam em formato de texto puro e que os nomes dos criminosos presentes nas ocorrências já estavam pré-processados em uma tabela. Devido a isso, não foi necessária a utilização de PLN ou *Named-Entity Recognition* neste trabalho até o presente momento.

Posteriormente, foram analisadas ferramentas de montagem de redes sociais. Dentre estas ferramentas, a que pareceu ser uma boa opção para ser utilizada neste trabalho, foi a ferramenta *Gephi*. *Gephi* é uma plataforma de visualização e exploração interativa para todos os tipos de redes e sistemas complexos, gráficos dinâmicos e hierárquicos[1].

Já a determinação dos criminosos chave foi feita algoritmicamente por meio de métricas de Análise de Redes Sociais e de centralidade nodais. Métricas de centralidade nodais quantificam a importância de um nodo em uma rede ou o

1 http://gephi.github.io/

quanto um nodo é “central” em um grafo (Mieghem, 2014). Por fim, foram feitos dois estudos de caso com dados reais disponibilizados pela BM e os resultados foram avaliados por ela.

1.3 Organização deste trabalho

Este trabalho está organizado da seguinte maneira: no Capítulo 2 são apresentados os aspectos teóricos que foram estudados para o desenvolvimento do trabalho. São abordados temas como Processamento de Linguagem Natural, Reconhecimento de Entidade Nomeada e Análise de Redes Sociais. Já o Capítulo 3, descreve a abordagem proposta, a implementação do trabalho e a ferramenta desenvolvida para a Brigada Militar. Por fim, o capítulo 4 apresenta as conclusões obtidas neste trabalho.

2 REFERENCIAL TEÓRICO

Para a realização deste trabalho, foi necessário estudar alguns aspectos teóricos. Neste capítulo é apresentada a teoria estudada. Dentre os assuntos que foram estudados pode-se citar Processamento de Linguagem Natural, Análise de Redes Sociais e a ferramenta *Gephi.*

2.1 Processamento de Linguagem Natural

Processamento de Linguagem Natural (PLN) é o nome dado à área de pesquisa que se dedica a investigar, propor e desenvolver formalismos, modelos, técnicas, métodos e sistemas computacionais que têm a linguagem natural como objeto primário (Nunes, 2008). De modo geral, PLN tem como um de seus focos principais o estudo da Linguagem Natural (LN), tal como os idiomas português ou inglês. A língua pode ser escrita na forma de texto ou falada, na forma de onda sonora. Além disso, algumas linguagens naturais alternativas, como a linguagem de sinais, também têm sido alvo de estudos que buscam novas formas de automatização.

No caso da fala, os trabalhos de PLN costumam ser desenvolvidos na área de Processamento de Sinais, pois os principais problemas da linguagem falada estão mais relacionados à síntese e reconhecimento do som e menos a questões linguísticas. Por esse motivo, segundo Nunes (2008), PLN é quase sinônimo de processamento de linguagem escrita.

Devido à complexidade de PLN, as pesquisas na área geralmente tratam as etapas do processamento de forma separada, investigando os fenômenos e características de cada uma delas. Assim, tem sido possível determinar a complexidade e o papel de cada uma das etapas nesse processamento.

Sabe-se que a primeira grande área de atuação de PLN foi a Tradução Automática (TA) (Nunes, 2008). Porém, atualmente, esse ramo vem sendo mais utilizado para o tratamento de problemas pontuais, como reconhecer ou classificar um sintagma ou uma sentença, escolher o sentido correto de uma palavra ambígua, rotular uma ocorrência textual com sua categoria gramatical no contexto, etc. Geralmente, o procedimento utilizado pelos pesquisadores de PLN é fragmentar o problema e solucioná-lo por meio da combinação das partes.

Contudo, essa solução não é completa devido à ambiguidade da linguagem natural, dentre outros fatores. Aliado a este fato, a avaliação da solução não é trivial, pois é difícil definir, após as partes terem sido fragmentadas, quais das soluções intermediárias mais contribuíram para o desempenho geral ou a falta do mesmo. Além disso, a avaliação dos sistemas de PLN são complexas, pois, geralmente, envolvem avaliações humanas, o que causa uma certa subjetividade no processo.

Segundo Nunes (2008), algumas das tarefas realizadas através de PLN são:

- **Pré-processamento de textos**: o texto é subdividido em unidades fonéticas, lexicais, gramaticais, semânticas ou discursivas, de acordo com o objetivo da tarefa em questão;

- **Classificação automática de texto segundo classes pertinentes à tarefa**: morfossintáticas (*PoS-tagger*), sintáticas (*Parser*), semânticas (*Parser* Semântico ou Interpretador), discursivas (*Parser* discursivo). Em cada um dos casos se deve definir linguagens de anotação para representar as classes, como, por exemplo, árvore sintática;

- **Mapeamento de representações**: as representações são mapeadas da Linguagem Natural para uma representação sintática, semântica ou discursiva e vice-versa;

As tarefas citadas acima são realizadas em aplicações de PLN, como: Tradução Automática, Sumarização Automática, Categorização de Textos, Recuperação da Informação, Extração de Informação, Sistemas de Diálogos, Sistemas de Auxílio à Escrita, etc. A primeira aplicação de PLN foi a Tradução Automática, tendo surgido concomitantemente com o computador. Já, quando surgiu a Internet e sua interface Web, as possibilidades de processar LN se ampliaram bastante devido à grande quantidade de texto na Web.

Para que as aplicações de PLN sejam possíveis são necessários recursos linguísticos-computacionais. Alguns exemplos importantes desses recursos, segundo Nunes (2008), são:

- **Corpora**: Representa um conjunto de informações sobre um texto, como estilo, padrões, desvios, etc. Para a extração dessas informações é necessária uma grande quantidade de textos, que devem ser exemplos variados e representativos. Além disso, é necessário que esses textos estejam em formato adequado para que a extração ocorra de forma automática e eficiente. Sistemas de manipulação de corpora permitem consultas de ocorrências no corpus, que é o conjunto de textos escritos em uma determinada língua que serve como base de análise, assim como a definição de estatísticas e padrões. A partir de informações de frequência de ocorrência de unidades em um determinado contexto, é possível gerar representações formais de conhecimento ou modelos probabilísticos, de modo que essas representações ou modelos possam ser usados por aplicações variadas.

- **Léxicos Computacionais**: São estruturas de dados em formato digital e adequado para consultas eficientes. Elas contêm informações sobre o conjunto e unidades lexicais (léxico) de uma LN. Usualmente essas unidades lexicais constituem palavras, porém a granularidade das unidades pode variar. Assim, reconhecidamente, apenas uma lista das palavras de uma LN não é suficiente para a maioria das aplicações de PLN. É necessário, também, que se tenha informações fonéticas, morfológicas, sintáticas, semânticas, dentre outras, sobre esses léxicos computacionais. Cabe ainda destacar a forma em que os léxicos são representados computacionalmente, pois são estruturas que serão consultadas frequentemente e, por isso, é necessário que não ocupem muito espaço e que a consulta seja efetuada de forma rápida. A teoria que permite isso, então, é a dos Autômatos Finitos Mínimos, que são estruturas que armazenam palavras de forma que prefixos comuns são representados uma única vez.

- **Ontologias**: São modelos de dados que representam um domínio digital e que têm um papel importante, principalmente, em aplicações Web. Como exemplo podem ser citados *websites* de vendas que possibilitam uma entrada linguística. Assim, uma ontologia desse domínio permite que a intenção do usuário seja percebida com mais facilidade. Como exemplo, quando o usuário faz referência, por exemplo, a uma TV como um "equipamento", a ocorrência desse termo após a menção da TV, resolve um problema de referência anafórica. Esse problema seria difícil de resolver se não houvesse essa relação entre ambos os termos.

- **Gramáticas Computacionais**: São formalismos para a representação de regras de formação de unidades da LN. Gramáticas Computacionais podem definir desde sintagmas até sentenças. No contexto da geração de texto, uma gramática define sentenças superficiais a partir de formalismos de representação semântica.

A área de investigação de PLN em língua inglesa é mais madura que em língua portuguesa, havendo um grande potencial de crescimento em português, apesar da significativa evolução apresentada nos últimos anos. A Linguateca, por exemplo, é um repositório digital disponível *on-line* no *website* http://linguateca.pt. Trata-se de um centro de recursos para processamento computacional da língua portuguesa. Várias ferramentas estão disponíveis para utilização, além de publicações de projetos em andamento e fórum de discussões para pesquisadores da área. Essa base de dados serve como exemplo do quanto a pesquisa em PLN em língua portuguesa encontra-se em expansão (Camara Junior, 2013).

Tradicionalmente, o trabalho em PLN tende a considerar o processo de análise da linguagem como uma decomposição em estágios, tais quais as distinções teóricas da linguística, quais sejam a sintaxe, a semântica e a pragmática (DALE, 2010 *apud* Camara Junior 2013). A sintaxe trata da ordem e da estrutura, a semântica aborda o significado e a pragmática se refere ao significado contextualizado. Essa estratificação tem um propósito pedagógico, uma vez que geralmente é bastante penoso separar o processamento da linguagem nas suas respectivas caixas. A estratificação constitui uma base para

modelos arquiteturais que tornam o PLN mais gerenciável do ponto de vista da engenharia de *software.* A figura 1 mostra os estágios da abordagem linguística de análise em PLN, conforme apresentado em Dale (2010 *apud* Cama Junior, 2013), partindo do exame na superfície do texto incrementando gradativamente a profundidade da análise.

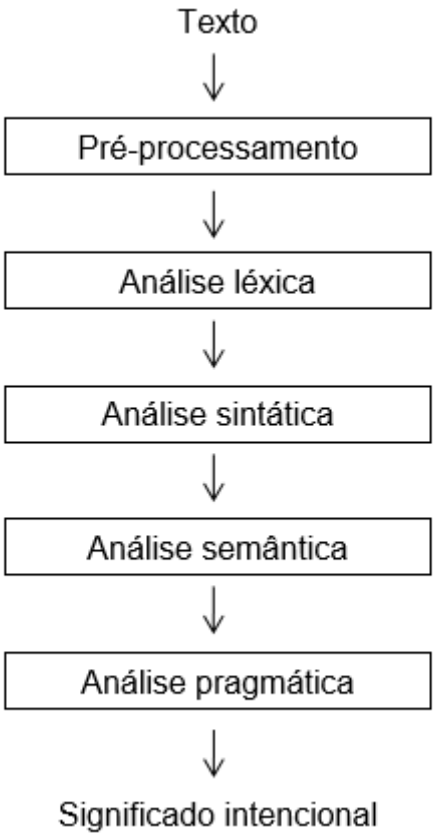


Figura 1 – Estágios de análise em PLN. Fonte: Dale, 2010 apud Camara Junior (2013)

Existem também outros modelos de estágios de análise de PLN, como o de Nierenburg e Raskin (2014 *apud* Camara Junior, 2013), onde os estágios são os mesmos da Figura 1. Porém, é criado o ciclo que pode ser visto na Figura 2. Onde os resultados do passo anterior são as entradas do passo seguinte, indicados pelas setas maiores. Além disso, existe a possibilidade de o conhecimento gerado por um módulo posterior retroalimentar passos anteriores, indicados pelas setas menores, com o objetivo de facilitar a desambiguação.

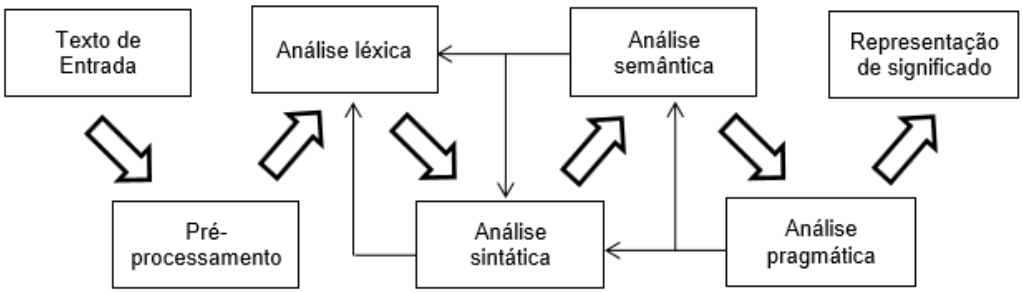


Figura 2 – Estágios retroalimentados de análise em PLN. Fonte: Nirenburg; Raskin, 2004 *apud* Camara Junior (2013)

Pode ser feita, também, uma abordagem dita plana, onde todos os estágios operam simultaneamente, sem esperar pelo resultado do passo anterior. Assim como ocorre em aplicações de tratamento de sons, na área de fonética e fonologia, podem haver passos anteriores a esses.

Em PLN, diferentes linguagens apresentam diferentes dificuldades. No caso do português ou inglês, o reconhecimento de palavras é facilitado pelo espaço em branco. Por outro lado, em línguas orientais, é bem mais difícil de se determinar um segmento.

Para que o Processamento da Linguagem Natural seja realizado de maneira correta, existem diferentes estágios pelo qual ele deve passar. A seguir são apresentados os estágios:

a) Pré-processamento textual

O pré-processamento do texto é a fase onde um arquivo de texto bruto, normalmente apresentado como uma sequência de bits, é reconhecido como uma sequência bem definida de unidades linguisticamente significativas, como grafemas. Um grafema é uma letra, símbolo gráfico utilizado para construir palavras do sistema de escrita, eles representam o nível mais baixo de uma linguagem. Evoluindo para um nível superior existem as unidades lexicais, que são unidades formadas por um ou mais grafemas. Por fim, subindo de nível, existem as sentenças, que contêm uma ou mais palavras. A etapa de pré-processamento textual é muito necessária, uma vez que as unidades linguísticas

(grafemas, unidades lexicais e palavras) são utilizadas por todas as etapas subsequentes do processamento.

Geralmente o pré-processamento é dividido em dois processos. O primeiro processo é chamado de triagem documental, onde um documento de texto bem formado é gerado a partir de um conjunto de arquivos digitais. O formato original dos documentos e a codificação dos caracteres são relevantes para esse passo. Além disso, é necessário o reconhecimento da macroestrutura textual para que possa haver o descarte de elementos não desejados, tais como imagens, cabeçalhos, marcação HTML, dentre outros. Segundo Palmer (2010, *apud* Camara Junior, 2013), a conclusão desse estágio é um *corpus* bem definido, pronto para ser utilizado para análises mais profundas. Esse *corpus* precisa ter um tamanho adequado para o treinamento ou aplicação para o qual se destina, assim como a amostragem deve ser estatística e representativa de todo o conteúdo.

Já a segunda etapa da fase de pré-processamento do texto é a segmentação. Nesse estágio o *corpus* é convertido em suas unidades lexicais e sentenças. Cada unidade lexical é denominada *token*, e esse processo é denominado *tokenização*. Em língua portuguesa o delimitador geralmente utilizado na segmentação das palavras é o espaço em branco, porém outros caracteres especiais também podem ser utilizados para isso.

Por fim, é importante lembrar que o sucesso do sistema de PLN depende bastante da fase de pré-processamento, pois com o aprofundamento da análise, os erros cometidos anteriormente se potencializam e impedem o progresso do processamento. Portanto, devem ser tratadas todas as questões e conseguir uma boa taxa de acerto na *tokenização* e separação de sentenças.

b) Análise Léxica

O objetivo da análise léxica é estudar a morfologia das unidades lexicais, ou palavras, e recuperar informação que será útil em níveis mais profundos de análise (Hippisley, 2010 *apud* Camara Junior, 2013). Na fase anterior, elas já foram devidamente *tokenizadas* e reconhecidas, o que facilita o aprofundamento de sua análise.

Percebe-se, portanto, que a análise léxica é responsável por dois processos. O primeiro deles é o *parse* da unidade lexical, que consiste em encontrar uma palavra e canonizá-la em seu lema. Logo após ela é armazenada no dicionário com as suas respectivas regras de formação. Já o segundo processo, que é denominado geração, é o contrário do primeiro, ou seja, a partir do lema e do conjunto de regras é necessário chegar à palavra derivada.

Sabe-se que a decomposição de palavras e a detecção de regras de formação torna possível uma economia de espaço de armazenamento e aumenta a velocidade de processamento. Com isso, são armazenados os lemas das palavras e aproveitadas as diversas terminações e modificações. Morfologistas reconhecem três abordagens clássicas para estruturação de palavras. A primeira, denominada de item e arranjo, atende ao caso ideal onde uma unidade lexical é a derivação de seu lema somado a um sufixo. A segunda abordagem se chama item e processo. Nela se leva em consideração o processo por meio do qual palavras complexas são geradas pela variação do sufixo, por tipo de lema. Esta abordagem está focalizada no processo fonológico que está associado à operação morfológica. Por fim, a última abordagem é chamada de palavra e paradigma, onde o lema é endereçado em uma tabela que associa a variante morfológica do lema com o conjunto das propriedades morfossintáticas. Essa tabela normalmente é implementada como uma árvore de derivação (Hippisley, 2010 *apud* Camara Junior, 2013).

. Após as palavras serem reconhecidas, é realizado o reconhecimento das orações. Para isso, reconhecem-se os limites dos períodos, que geralmente são o ponto final. Porém, deve se ter cuidado com outros pontos utilizados no meio das frases, como abreviações e partes fracionárias. Além disso, uma frase pode ter um único ponto final mas possuir mais de uma oração dentro dela.

O modelo computacional mais efetivo na construção de ferramentas de PLN é a utilização de Autômatos de Estado Finito (AEF). Qualquer pesquisa textual bem implementada é suportada por um AEF, o que demonstra sua enorme utilidade prática. Winter (2010, *apud* Camara Junior, 2013) coloca que os AEF são dispositivos computacionais que geram linguagens regulares, no entanto eles também podem ser vistos como reconhecedores. Um AEF que gere uma linguagem e uma palavra qualquer pode determinar, em tempo linear, se uma dada palavra pertence àquela linguagem. Isso justifica o uso de AEF para

a procura em dicionários. Uma aplicação de PLN que só é possível devido a esse formalismo.

Em suma, a análise léxica é uma importante etapa no PLN e fornece insumo à análise sintática, que é a fase posterior do processamento.

c) Análise Sintática

A análise sintática é a etapa de PLN onde uma sequência de unidades lexicais, tipicamente uma oração, será decomposta para determinar sua descrição estrutural de acordo com uma gramática formal (Ljunglof, Wirén, 2010 *apud* Camara Junior, 2013). O resultado da análise sintática é uma hierarquia sintaticamente estruturada preparada para interpretação semântica.

A análise sintática de uma linguagem natural é contextualizada, o que atribui uma complexidade exponencial ao problema. A ambiguidade, por exemplo, existe e demanda tratamento, pois a distribuição das possibilidades aumenta o espaço amostral. Além disso, uma linguagem livre possui ilimitadas possibilidades de construção, inclusive incorretas, o que dificulta a análise, pois pode ocorrer um erro que foi causado por construção incorreta ou falta de cobertura da gramática.

Segundo Winter (2010) *apud*. Camara Junior (2013), a forma padrão de representação da estrutura sintática de uma sentença gramatical é uma árvore sintática, árvore de derivação ou árvore *parse*. Clark (2010 *apud* Camara Junior, 2013) explica que há diversos formalismos complexos para modelagem de linguagens naturais, tais como a gramática de generalização frase-estrutura, gramática léxico-funcional, gramática de junção-arbórea, gramática categorial combinatória, dentre outras. Contudo, foi demonstrado por indução infinita que todas elas são equivalentes à Gramática Livre de Contexto (GLC).

d) Análise Semântica

A análise semântica trata do significado da sentença, não sendo possível dar significado ao conteúdo, mas analisando as relações válidas entre as palavras, a partir de seus conceitos. Essa análise demanda um forte esquema de representação do conhecimento para ser efetivada. É necessária uma ontologia

que determine traços semânticos das unidades lexicais para reconhecimento da validade das relações. Além disso, segundo Nierenburg e Raskin (2004 *apud* Camara Junior, 2013), é necessário um formalismo para representação do significado textual e conhecimento estático para processamento semântico, o qual inclua o mapeamento das estruturas de dependências sintáticas e semânticas, tratamento de referências e regras de estruturação textual.

A análise semântica é dividida em semântica léxica e semântica composicional, que também pode ser chamada de combinatória ou gramatical. A primeira análise tem como foco o estabelecimento do significado de unidades lexicais ou de combinações fixas de palavras. Sob essa perspectiva, palavras com conteúdo, como substantivos ou adjetivos, são mais importantes do que as não têm conteúdo em si, como preposições ou artigos. Já a segunda análise procura compreender as infinitas possíveis combinações de unidades léxicas em frases que obedeçam às regras gramaticais.

Sabe-se que um grande problema da análise semântica em PLN é o tratamento da ambiguidade, pois do ponto de vista computacional, uma declaração pode ser interpretada de várias maneiras devido a algumas palavras terem mais de um significado. Podem ser citados como exemplos a homonímia, onde duas palavras que possuem a mesma grafia ou a mesma pronúncia, mas com significação distinta, e a polissemia, onde uma determinada palavra ou expressão que pode ter mais de um significado dependendo do contexto.

e) Análise Pragmática

Na análise pragmática se procura incluir o contexto à análise linguística, com o objetivo de permitir a geração de um significado. O componente pragmático utiliza uma base de dados construída em um esquema de representação de conhecimento para representar o contexto externo do texto e permitir a utilização desse conhecimento para inferências automatizadas (Mellish; Pan, 2008 *apud* Camara Junior, 2013). A exploração da análise pragmática ocorre através do reconhecimento de que as sentenças de um texto não são isoladas, mas se relacionam entre si e formam um discurso. O conhecimento de padrões desse discurso que sejam comuns a determinados tipos de textos pode ser muito útil.

A análise pragmática ainda é a mais imatura das etapas de PLN. Porém, é possível se perceber que já existem aplicações que utilizam seus resultados.

Em suma, o Processamento de Linguagem Natural é uma área computacional de pesquisa que tem a linguagem como objeto primário. Sendo assim, possui uma vasta gama de ramos a serem explorados. Dentre essas áreas, está o Reconhecimento de Entidade Nomeada, que será descrito a seguir.

2.1.1 Reconhecimento de Entidade Nomeada

O *Named Entity Recognition* (NER), em português Reconhecimento de Entidades Nomeadas, é um importante ramo no processamento de linguagem natural que tem como tarefa a classificação e busca de expressões em textos em linguagem natural (Silveira, 2014). O NER pode ser utilizado para diversos fins, dentre eles se pode citar uma ferramenta de busca e filtro dentro de um texto. Segundo Silveira (2014), as tarefas que utilizam NER são a tradução automática, resposta automática a uma pergunta, sumarização, modelagem de linguagem natural e análise de opinião em textos. Estas tarefas são melhor executadas explorando-se o uso de entidades nomeadas (NE) e trabalhando com as entidades individualmente.

O termo Entidade Nomeada foi utilizado primeiramente em 1996 na sexta Conferência de Compreensão de Mensagem, *Message Understanding Conference* (MUC). Entidades nomeadas se referem a “identificadores únicos de entidades”, onde a importância da semântica da identificação de pessoas, organizações e localizações, assim como expressões numéricas, como tempo e quantidades, estava clara (Marrero *et al*, 2012). Ainda assim, um dos primeiros estudos realizados nesse campo foi apresentado por Lisa F. Rau em 1991 na sétima Conferência em Aplicações de Inteligência Artificial. O estudo de Rau descrevia um sistema para “extrair e reconhecer nomes de companhias” (Nadeau *and* Sekine, 2007).

Muitas ferramentas ainda consideram uma entidade nomeada como o que foi definido no MUC. Apesar disso, ainda há variações sobre o que efetivamente é uma entidade nomeada. Sistemas de NER genéricos tendem a focar em encontrar nomes de pessoas, lugares e organizações que são mencionados em

textos comuns de notícias. Aplicações práticas têm sido construídas também para detectar desde nomes de genes e proteínas até nomes de cursos universitários (Jurafsky *and* Martin, 2008).

Existem diversos fatores que alteram o desempenho de tarefas relacionadas a entidades nomeadas, sendo os mais comuns dentre eles o idioma, o domínio e a informação analisada. O idioma é um fator importante a ser considerado porque os sistemas de NER são desenvolvidos para uma linguagem específica, pois a adaptação de um idioma para outro é difícil. Segundo Silveira (2014), a diferença de desempenho entre idiomas pode chegar a 20%, mesmo com o uso de sistemas de aprendizado automático que permitem a escolha de funcionalidades independentemente do idioma utilizado. Já o domínio da *corpora* utilizada tem importância porque o tamanho do *corpus* influencia no desempenho dos sistemas de NER. Além disso, existem *corpus* que são mais fáceis de serem utilizados do que outros, como é o caso de artigos de jornais e textos de redes sociais. Ademais, a informação analisada também tem a sua influência no desempenho do NER, pois algumas entidades são mais fáceis de se encontrar do que outras. Deve ser levado em consideração que isto depende da definição das classes. Um exemplo de como esta definição tem impacto na busca das entidades pode ser mostrado nas entidades do tipo *datetime*, estas entidades podem conter tanto datas em sua forma absoluta, tal como “15 de janeiro de 2013”, ou datas em uma forma relativa, tal como “próximo sábado” (Silveira, 2014).

A habilidade de reconhecer entidades previamente desconhecidas é uma parte essencial do sistema NER. O ponto principal dessa competência está no reconhecimento e classificação de regras desencadeadas por características distintas associadas com exemplos negativos e positivos. Existem dois tipos de implementação para essa competência. Ela pode ser implementada através da criação manual de regras ou através de aprendizado supervisionado de máquina. Atualmente, o que tem sido mais utilizado é o aprendizado supervisionado por máquina. A ideia desse método é estudar as características dos exemplos positivos e negativos de entidade nomeada com base de um grande acervo de documentos e desenvolver regras que capturem exemplos de um determinado tipo. A principal deficiência do aprendizado supervisionado é que ele requer um grande *corpus.* A indisponibilidade desse recurso e o custo

proibitivo de criá-lo leva a duas alternativas de métodos de aprendizado: o aprendizado semisupervisionado e o aprendizado não supervisionado (Nadeau *and* Sekine, 2007). Esses métodos serão apresentados a seguir:

a) Aprendizado Supervisionado

A abordagem desse método tipicamente consiste de um sistema que lê um grande *corpus,* memoriza listas de entidades e cria regras sem ambiguidade baseadas em características discriminativas. Um sistema de base que é frequentemente proposto consiste em marcar palavras de um *corpus* de teste quando elas estão marcadas como entidades no *corpus* de treino. A performance do sistema de base depende da transferência de vocabulário, que é a proporção de palavras, sem repetições, aparecendo nos dois *corpus* (de treino e de teste).

b) Aprendizado Semisupervisionado

A técnica principal para o aprendizado semisupervisionado é chamada *bootstrapping* e envolve um pequeno grau de supervisão, tal como um conjunto de “sementes”, para começar o processo de aprendizagem. Por exemplo, um sistema destinado a identificar “nomes de doenças” pode pedir para o usuário fornecer um pequeno número de nomes como exemplo. Então o sistema procura por sentenças que contenham esses nomes e tenta identificar algumas pistas contextuais que são comuns aos exemplos. Logo após, o sistema tenta encontrar outras instâncias de nomes de doenças que apareçam em contextos similares. O processo de aprendizado é então reaplicado aos novos exemplos encontrados, a fim de descobrir novos contextos relevantes. Repetindo esse processo, um grande número de nomes de doenças e um grande número de contextos vão, eventualmente, ser coletados.

c) Aprendizado Não Supervisionado

A principal abordagem do aprendizado não supervisionado é o agrupamento. Por exemplo, pode-se tentar coletar entidades nomeadas de grupos que foram agregados baseados na similaridade de contexto. Porém, há outros tipos de

métodos não supervisionados também. Basicamente, as técnicas dos mesmos dependem de recursos léxicos, padrões léxicos e estatísticas computadas em *corpus.*

Logo, diversas tarefas dentro de PLN utilizam Reconhecimento de Entidade Nomeada. Porém, na área de NER são necessárias algumas regras para que as tarefas possam ser cumpridas corretamente, essas regras são as *features*, que serão descritas a seguir.

Features

Segundo Nadeau *and* Sekine (2007), os problemas de NER são resolvidos através da aplicação de um sistema de regras sobre certas características pré-determinadas pelo sistema, usualmente chamadas de *features*. Essas características são descritores ou atributos de palavras designados para o uso do algoritmo de NER. Um exemplo de uma *feature* é uma variável booleana com o valor verdadeiro se a palavra começar com uma letra maiúscula, e falso caso contrário. Uma representação de vetor de *features* é uma abstração do texto onde, tipicamente, cada palavra é representada por um ou vários valores booleanos, numéricos ou nominais.

As *features* mais usadas no reconhecimento e classificação de entidades nomeadas, segundo Nadeau *and* Sekine (2007), estão divididas em três categorias: *Features em nível de* palavra, *Features* de procura em lista e *Features* de documento e *corpus*.

a) *Features* em nível de palavra

Esses tipos de *features* estão relacionadas à construção de caracteres de palavras. A Tabela 1 lista alguns exemplos de *features em* nível de palavra:

Features	**Exemplos**
Caixa	-Começa com letra maiúscula -Toda a palavra está em letra maiúscula -A palavra contém letras maiúsculas e minúsculas (ex.: eBay, ProSys)
Pontuação	-Termina com ponto ou tem ponto no meio (ex.: Av., I.B.M.) -Apostrofe, hífen ou "e comercial" (ex.: O'Connor)
Dígito	-Dígito padrão -Cardinal e Ordinal -Número Romano -Palavras com dígitos (ex.: W3C, 3M)
Caracter	-Letras gregas
Morfologia	-Prefixo, sufixo ou versão singular -Final em comum
Parte da linguagem	-Nome próprio, verbo, substantivo ou palavra estrangeira
Função	-Alpha, não-alpha ou n-gram -Versão de caixa de letra maiúscula, caixa de letra minúscula -Padrão, padrão resumido -Comprimento do token ou comprimento da expressão

Tabela 1 - *Features* em nível de palavra. Fonte: Nadeau and Sekine (2007)

b) *Features* de procura em lista

Esse tipo de *feature* também pode ser chamado de procura em "dicionário". Inclusão em lista, ou dicionário, é uma maneira de expressar relação do tipo: Paris é uma cidade. A Tabela 2 lista alguns exemplos de *features* de procura em lista:

Features	Exemplos
Lista Geral	-Dicionário Geral -Abreviações comuns -Substantivos que começam com letra maiúscula (ex.: Janeiro, Fevereiro)
Lista de entidades	-Organização, governo ou educacional -Nome, sobrenome ou celebridade -Corpo astral, continente, país, estado ou cidade
Lista de sugestão de entidades	-Palavras típicas em organizações -Título de pessoa, prefixo de nome ou letras pós-nominais -Palavra típica de localização ou ponto cardinal

Tabela 2 - *Features* de procura em lista. Fonte: Nadeau and Sekine (2007)

c) *Features* de documento e *corpus*

As *features* de documento são definidas em cima do conteúdo e da estrutura do documento. Uma grande quantidade de documentos (*corpora*) são, também, uma excelente fonte de *features*. A Tabela 3 lista alguns exemplos de *features* de documento e *corpus*:

Features	Exemplos
Múltiplas ocorrências	-Outras entidades no contexto -Ocorrências em caixas de letras maiúscula e minúscula -Anáfora, correferência
Sintaxe local	-Enumeração ou aposição -Posição na sentença, no parágrafo e no documento
Meta informação	-Leitor de e-mail ou seção XML -Listas numeradas, tabelas ou figuras
Frequência de *corpus*	-Frequência de palavra e expressão -Co-ocorrências

Tabela 3 - *Features* de documento e *corpus*. Fonte: Nadeau and Sekine (2007)

Portanto, as *features* representam um sistema de regras pré-determinadas para o funcionamento do sistema de Reconhecimento de Entidade Nomeada. Entretanto, é necessário que haja uma avaliação das regras e do

sistema em si. A seguir serão discutidos os métodos de avaliação dos sistemas NER.

Avaliação dos sistemas de NER

Eventos competitivos são organizados para a avaliação de sistemas NERC (Reconhecimento e Classificação de Entidade Nomeada), onde a habilidade de identificação e classificação das entidades existentes em um *corpus* são analisadas (Marrero *et al,* 2009). Dentre esses eventos, as mais citadas são as conferências MUC e CoNLL (*Conference on Computational Natural Language Learning)*, porém também existem outros eventos que tratam desse assunto. Em NERC, os sistemas são usualmente avaliados baseados em uma comparação de seus resultados com a saída dos mesmos com a resposta gerada pelos linguistas humanos.

Segundo Nadeau e Sekine (2007), no MUC um sistema é pontuado em dois eixos: sua habilidade em encontrar o "tipo" correto e sua habilidade em encontrar o "texto" correto. Um "tipo" correto é creditado se uma entidade é atribuída ao tipo correto, independentemente de limites, desde que exista uma justaposição. Um "texto" correto é creditado se os limites da entidade estão corretos, independentemente do tipo. Para ambos (tipo e texto), três medidas são dadas: o número de respostas corretas (COR), o número de suposições reais do sistema (ACT) e o número de possíveis entidades na solução (POS). A nota final do MUC é a "micromédia" medida *f* (MAF – ou, do inglês, *f-measure*), que é a média harmônica da precisão (do inglês, *precision*) e revocação (do inglês, *recall*) calculadas através de todas as entidades nos dois eixos. Uma "micromédia" é realizada em todos os tipos de entidade, sem distinção, erros e sucessos para todas as entidades são somados juntos. A média harmônica de dois números nunca é maior que a média geométrica. Ela também tende para o número mínimo, minimizando o impacto de valores extremos e maximizando o impacto de valores pequenos. A medida *f* tende, então, a privilegiar sistemas balanceados. No MUC, a precisão é calculada como COR / ACT e a revocação como COR / POS.

Já a CoNLL, possui um protocolo de pontuação simples. Os sistemas são comparados baseados na MAF com a precisão sendo a porcentagem de entidades nomeadas encontradas pelo sistema que estão corretas e a revocação sendo a porcentagem de entidades nomeadas presentes na solução que são encontradas pelo sistema. Uma entidade nomeada está correta somente se ela corresponder exatamente à entidade na solução.

Portanto, se pode concluir que existem diversos métodos de avaliação de um sistema NER e que é importante que haja uma avaliação desses sistemas. No contexto desse trabalho, pretendia-se desenvolver um sistema NER para que fossem retiradas entidades nomeadas das ocorrências digitais. E, assim, a seguir, essas entidades nomeadas pudessem ser utilizadas na técnica de Análise de Redes Sociais. Portanto, na próxima seção será apresentada a teoria relacionada a essa técnica.

2.2 Análise de Redes Sociais

Uma rede social é uma estrutura formada por indivíduos ou organizações, baseada em um grafo, onde cada indivíduo é representado por um nodo e as ligações entre eles são representadas por arestas (Lavrac *et al*,2010), como pode ser visto na Figura 3. Nessa figura, os números representam apenas rótulos dos nodos. Esse tipo de estrutura geralmente é montada para facilitar a análise de uma grande quantidade de dados e os grafos-base resultantes dela podem ser bastante complexos.

Segundo Lavrac *et al* (2010), a análise de rede sociais foca em interpretar padrões de laços entre pessoas, grupo de pessoas, organizações ou países. Um domínio típico de SNA é um grupo de indivíduos com suas características e a estrutura de seus laços. Mais especificamente, o valor de SNA reside na estrutura dos relacionamentos em uma rede, em vez das características individuais dos participantes. As redes sociais vêm sendo usadas para lidar com uma variedade de questões de pesquisas em diferentes contextos. Como exemplo de áreas onde seus métodos vêm sendo usados se pode citar a predição do deslocamento de uma população após desastres naturais, o estudo de controle de doenças e o comportamento social online. Já um ramo onde a análise de redes sociais vem crescendo ultimamente é na

análise de redes criminais (CNA - *Criminal Network Analysis*). Em CNA, as aplicações vão da criminalidade geral até redes terroristas, crime organizado e gangues de rua (Rostami e Mondani, 2015). A essência dessas estratégias são fazer com que a polícia seja mais proativa em resposta ao crime, processando informações sobre os criminosos e formando estratégias de redução e prevenção à criminalidade.

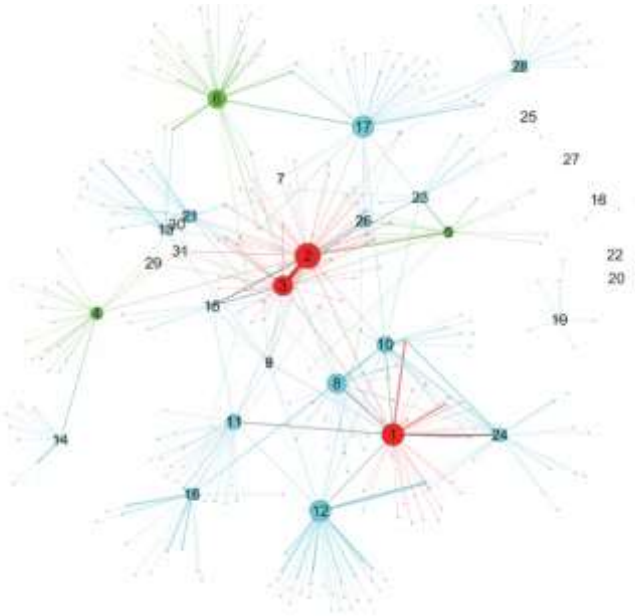


Figura 3 – Exemplo de rede social. Fonte: Rostami e Mondani (2015)

A ideia de que a análise de redes sociais pode oferecer métodos e padrões para a análise de crimes e mais aplicações práticas durante investigações criminais tem sido sugerida em pesquisas por quase 20 anos (Morselli, 2010). Diversos autores já demonstraram que utilizar uma perspectiva de rede criminal permite a administração de uma investigação complexa que contenha muitos indivíduos com incontáveis interações entre eles. Além disso, a rede criminal fornece técnicas visuais e analíticas que levam a um conhecimento baseado em evidências da estrutura geral de uma rede criminal e ao posicionamento de uma variedade de participantes-chave.

SNA emergiu como uma técnica-chave na sociologia moderna mas também ganhou significância em áreas como a antropologia, biologia, estudos sociais, economia, geografia, ciência da informação, dentre outras. Nesse

contexto, ela tem sido uma ferramenta chave de análise da inteligência criminal desde os anos 70 (Mondani, 2015). Essa área vem avançando desde uma técnica básica de análise de ligações na inteligência criminal e investigações, até os conceitos mais recentes de núcleo de uma rede e medidas de SNA. Segundo Mondani (2015), na Dinamarca a polícia está utilizando SNA para detectar membros de gangues que têm potencial a serem induzidos a deixar suas gangues. Essa técnica também tem ganhado popularidade dentre as agências policiais da Suécia. Além disso, a Europol (Serviço Europeu de Polícia) e o FBI *(Federal Bureau of Investigation)* dos Estados Unidos também estão utilizando análise de redes sociais como uma técnica para planejar operações e prever eventos criminais.

Porém, sabe-se que apesar da técnica de análise de redes criminais ser de grande ajuda no combate à criminalidade, ela também apresenta algumas dificuldades e limitações. Uma dessas limitações diz respeito aos dados que serão utilizados para montar a rede. Visto que, eles podem estar incompletos ou com algum tipo de erro, como no caso de algum policial perder os dados, cometer algum erro, ou ainda no caso dos crimes que não foram descobertos ou registrados. Outra possibilidade, ainda, é a pessoa que foi registrada pelo crime ser inocente, acarretando em um erro, onde uma pessoa inocente será inserida na rede criminal e o verdadeiro culpado não. Porém, isso não será um problema no caso de não ser recorrente, pois se um suposto inocente for inserido na rede mais de uma vez é provável que ele não seja inocente. Ademais, outra dificuldade de se analisar uma rede criminal é que a interação entre a agência policial e a população observada afeta a estrutura e a dinâmica da rede. Prisões, incapacitações ou vazamentos de informações podem ter um efeito nos padrões de relacionamento dentro de uma rede criminal. Além disso, a polícia pode mudar a estrutura e o conjunto de membros da rede observada por meio de vigilância ativa e intervenções (Morselli, 2010). Essa vigilância pode, também, resultar na prisão de um ator principal ou membros de uma rede criminal, implicando em uma reestruturação das interações internas.

2.2.1 Conceitos e representação matemática

Segundo Scott (2000), seja qual for a forma física em que os dados utilizados para gerar a rede social estejam, a estrutura lógica da matriz dos dados é sempre a de uma tabela. Na análise de variáveis, os dados dos atributos podem ser organizados em uma matriz caso-por-variável, como mostra a Figura 4, onde cada caso estudado, tal como um criminoso, é representado por uma linha na matriz, enquanto as colunas se referem as variáveis em que seus atributos são medidos.

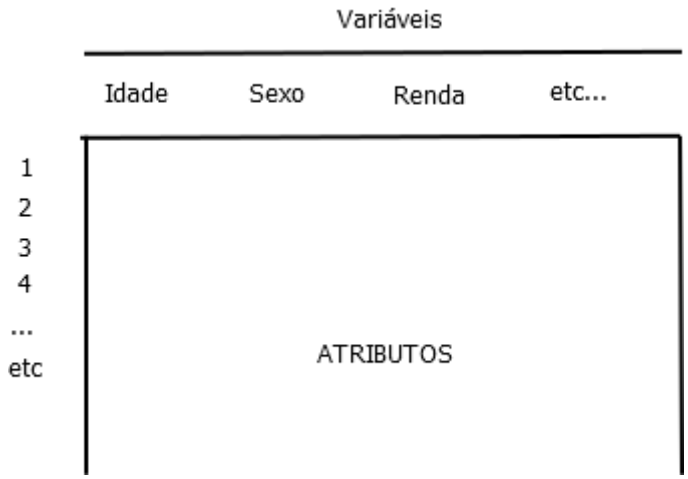


Figura 4 - Uma matriz de dados para análise de variáveis. Adaptado de Scott (2000)

Porém, pode levar tempo e ser trabalhoso construir redes sociais para até mesmo um conjunto de dados de tamanho médio, com até dez participantes e cinco relações (Scott, 2000). Pois, as linhas se cruzam umas com as outras em todos os tipos de ângulos, formando um emaranhado de conexões e dificultando a visualização da rede. Por essa razão, SNA vem tentando encontrar maneiras alternativas de registrar as conexões. Uma dessas maneiras seria separar em duas tabelas os casos estudados das ligações, como mostra a Figura 5.

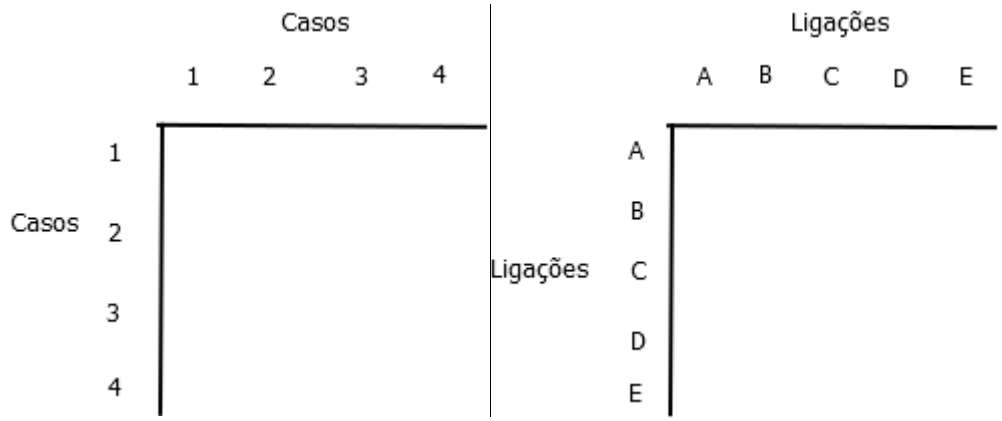


Figura 5 – Matrizes para análise de rede social. Adaptado de Scott (2000)

Na matriz dos casos, as linhas e as colunas representam os casos, e as células individuais vão mostrar quais pares de indivíduos estão relacionados em uma ligação em comum. Essa matriz mostra, portanto, as ligações reais entre os indivíduos, a informação que ela contém é equivalente à apresentada na rede social. Já a matriz das ligações também contém ligações nas linhas e colunas e as células individuais mostram quais pares de indivíduos estão ligados por agentes comuns. Essa matriz também é importante na análise da rede, pois ela pode conter informações que não estão explícitas na primeira tabela. Em SNA, uma matriz retangular, como a da Figura 4, é chamada de matriz de incidência e uma matriz quadrada, como a da Figura 5, é chamada de matriz de adjacência. Esses termos derivam da Teoria dos Grafos.

O uso dessas matrizes de incidência e adjacência na montagem de uma rede social pode ser demonstrado em um exemplo. Na Figura 6 é apresentada uma matriz de incidência com um conjunto de dados fictícios que relacionam criminosos dentro de ocorrências. O número 1 é utilizado em uma célula se um criminoso está presente em uma ocorrência e 0 se ele não está presente.

		Criminosos				
		A	B	C	D	E
Ocorrências	1	1	1	1	1	0
	2	1	1	1	0	1
	3	0	1	1	1	0
	4	0	0	1	0	1

Figura 6 – Matriz de incidência. Adaptado de Scott (2000)

Já nas Figuras 7 e 8 os mesmos dados foram separados em duas matrizes de adjacência que relacionam criminosos com criminosos e ocorrências com ocorrências. Na matriz da Figura 7 é apresentado o número de criminosos em comum entre um par de ocorrências. Por exemplo, a ocorrência 1 e a ocorrência 4 têm apenas um criminoso em comum (criminoso C), por outro lado a 2 e a 3 têm dois criminosos em comum (o B e o C). E na Figura 8 é apresentada uma ocorrência em comum entre um par de criminosos. Por exemplo, os criminosos A e B tem duas ocorrências em comum (1 e 2).

	1	2	3	4
1	-	3	3	1
2	3	-	2	2
3	3	2	-	1
4	1	2	1	-

Figura 7 – Matriz de Adjacência criminosos-por-criminosos. Adaptado de Scott (2000)

	A	B	C	D	E
A	-	2	2	1	1
B	2	-	3	2	1
C	2	3	-	2	2
D	1	2	2	-	0
E	1	1	2	0	-

Figura 8 – Matriz de Adjacência ocorrências-por-ocorrências. Fonte: Scott (2000)

As Figuras 9 e 10 ilustram, respectivamente, as redes sociais montadas a partir das matrizes de adjacência das ocorrências e dos criminosos.

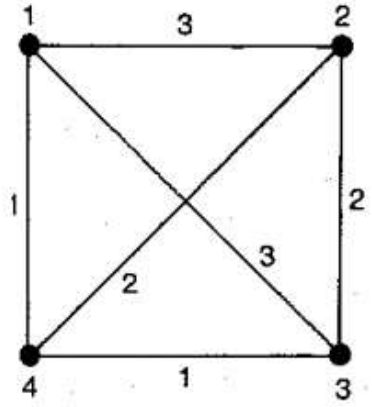


Figura 9 – Rede social de ocorrências. Fonte: Scott (2000)

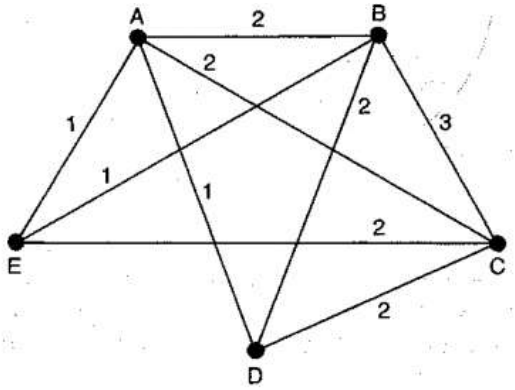


Figura 10 – Rede social de criminosos. Fonte: Scott (2000)

A partir de uma análise simples nessas redes sociais se pode inferir que a força das relações entre duas ocorrências ou dois criminosos pode ser medida a partir do número de ligações que eles possuem entre si. Como, por exemplo, na primeira rede, Figura 9, que sugere que as ocorrências 1 e 2, e 1 e 3, são aquelas que possuem as ligações mais fortes.

Essas redes sociais poderiam ter sido desenhadas de outras maneiras também. Na Figura 10, entre A e B, foi colocado o número 2 para representar o número de ligações. Em vez disso, a aresta poderia ter sido desenhada como uma linha mais grossa, ou, até mesmo, como duas linhas.

Também é possível se verificar na rede social da Figura 10, que os criminosos D e E são mais “periféricos” na rede em comparação aos outros criminosos, pois eles possuem menos conexões. Suas conexões são geralmente fracas e eles não estão conectados entre si.

Foi possível constatar, então, que grande parte dos conceitos e representações matemáticas da área de Análise de Redes Sociais deriva da Teoria dos Grafos. Devido a isso, será brevemente discutida a Teoria dos Grafos a seguir.

2.2.2 Teoria dos grafos

Na matemática, um grafo é uma tupla G = (V, E), onde representa de forma abstrata uma série de objetos, onde alguns pares desses objetos estão conectados por ligações (Passmore, 2015). Os objetos interconectados são representados por abstrações matemáticas chamados vértices, nodos ou nós e as ligações que conectam alguns pares são chamadas de arestas. Tipicamente, um grafo é representado em um diagrama onde um nó é simbolizado por um círculo preenchido ou não e as arestas são simbolizadas por linhas, como pode ser visto na Figura 11.

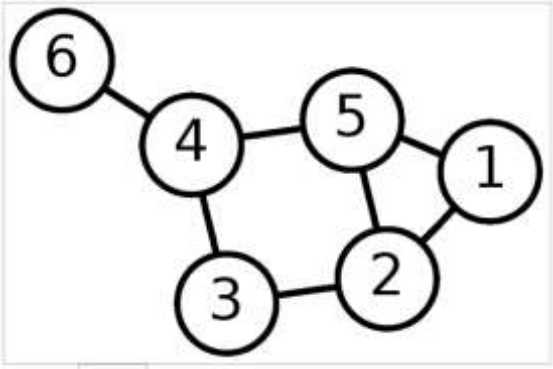


Figura 11 – Grafo com 6 vértices e 7 arestas. Fonte: Passmore (2015)

As arestas de um grafo podem ser direcionadas ou não, como, por exemplo, se os vértices representam pessoas em uma festa e as arestas representam duas pessoas que se apertam as mãos. Se a pessoa A aperta a mão da pessoa B, a pessoa B também aperta a mão da pessoa A, nesse caso a aresta é não direcionada. Já, no caso de uma pessoa conhecer a outra, a aresta é direcionada, pois a pessoa A pode conhecer a pessoa B enquanto que a pessoa B não conhece a pessoa A. Um grafo com arestas direcionadas é chamado de grafo direcionado e um grafo com arestas não direcionadas é chamado de grafo não direcionado. Quando um grafo é direcionado, as arestas

são representadas por uma seta saindo de um nó e chegando em outro, já, quando ele é não direcionado, elas são representadas apenas por uma linha.

A Teoria dos Grafos é uma área ampla e um dos objetos de estudo da Matemática Discreta. Dentro dessa área os grafos podem ser classificados em diversas categorias além dos já mencionados grafos direcionados e não direcionados, são elas: infinitos ou finitos, regular ou completo, fortemente conectado ou fracamente conectado, dentre outros. Porém, uma classificação importante dos grafos dentro da área de análise de redes sociais são os grafos ponderados (ou valorados). Um grafo é ponderado se um número (peso) é atribuído a cada aresta, esse peso pode significar um custo, comprimento, capacidade ou qualquer outra propriedade. No caso da análise de uma rede criminal, geralmente o peso da aresta representa a força da ligação entre dois indivíduos, e pode ser obtido, por exemplo, pelo número de vezes em que os dois foram vistos em uma mesma ocorrência.

Em síntese, a Teoria dos Grafos tem grande significância dentro da área de Análise de Redes Sociais. Além disso, outro assunto importante dessa área, são as métricas utilizadas para a análise das redes, esse assunto será discutido a seguir.

2.2.3 Métricas de SNA

Existem vários tipos de métricas com as quais é possível analisar as redes sociais conforme a sua utilização. No livro *Social Network Analysis: Theory and Applications*, de Passmore (2015), são definidas as seguintes métricas:

- ***Betweeness*** – Representa a extensão de um nodo entre outros nodos na rede. Essa medida leva em conta a conectividade dos nodos vizinhos, designando um valor mais alto para nodos que contenham um aglomerado de “pontes”. Ela reflete o número de pessoas a quem uma pessoa está conectada indiretamente por meio dos seus *links* diretos. Essa métrica pode ser definida matematicamente por:

$$C_B(k)_{norm} = \frac{\sum_{i<}^{n}\sum_{j}^{n} g_{ij}(k)}{n-1} \qquad i \neq k$$, onde $g_{ij}(k)$ indica se o caminho mais curto entre dois outros nós *i* e *j* passa através do nodo *k*.

- **"Ponte"** – Uma aresta é dita como sendo uma *ponte* se, no caso de retirada da mesma, resultaria em que seus pontos finais ficassem em diferentes componentes de um grafo.

- **Centralidade** – Essa medida provê uma indicação bruta do poder social de um nodo baseado no quão bem eles "conectam" a rede. *Betweeness*, proximidade e grau são medidas de centralidade.

- **Centralização** – É a diferença entre o número de ligações para cada nodo pela máxima possível soma de diferenças. Uma rede centralizada terá muitos *links* dispersos em volta de um ou poucos nodos, enquanto que uma rede descentralizada ocorre no caso em que há uma pequena variação entre o número de *links* que cada nó possui.

- **Proximidade** – Consiste no grau em que um indivíduo está perto de outros indivíduos em uma rede (direta ou indiretamente). Proximidade é o inverso da soma das distâncias mais curtas entre cada indivíduo e outra pessoa na rede. Essa métrica pode ser definida matematicamente por:

$$C_C(k)_{norm} = \frac{\sum_{i=1}^{n} l(i,k) - C_c\min}{C_c\max - C_c\min} \qquad i \neq k$$, onde *l(i,k)* é o comprimento do caminho mais curto conectando os nodos *i* e *k.* C_cmin e C_cmax são os comprimentos mínimos e máximos dos caminhos mais curtos, respectivamente.

- **Coeficiente de aglomeramento** - É uma medida da probabilidade de dois nodos associados a um nodo estarem associados entre eles mesmos.

- **Coesão** – Representa o grau em que indivíduos estão conectados entre eles por vínculos coesivos. Grupos são identificados como “cliques” se cada indivíduo está diretamente ligado a cada um dos outros indivíduos. E “círculos sociais” se há menos rigor de contato direto, o que é impreciso, ou como blocos estruturalmente coesivos se é requerido precisão. A coesão pode ser definida matematicamente por:

$$\text{Coesão} = \frac{\frac{\sum_{i \in G} \sum_{j \in G} w(i,j)}{n(n-1)}}{\frac{\sum_{i \in G} \sum_{j \notin G} w(i,j)}{N(N-n)}}$$

Onde *n* é o tamanho do grupo de indivíduos, *N* é o tamanho da rede, *G* representa o conjunto do grupo de nós, *w(i,j)* é o peso da conexão entre os nodos *i* e *j.*

- **Grau** – É a contagem do número de ligações a outros indivíduos da rede. O grau de um indivíduo dá a proporção de ligações que ele tem com outros indivíduos. Essa métrica pode ser definida pela seguinte fórmula matemática:

$$C_D(k)_{norm} = \frac{\sum_{i=1}^{n} a(i,k)}{n-1}, \qquad i \neq k$$

, onde *k* representa um nodo, *n* é o número total de nodos e *a(i,k)* é uma variável binária que indica se existe uma conexão entre os nós *i* e *k.*

- **Fluxo da centralidade de *Betweeness*** - Consiste no grau que um nodo contribui para a soma do fluxo máximo entre todos os pares de nodos (sem contar o nodo em questão).

- **Centralidade de auto-vetor** – Mede a importância de um nodo em uma rede. Essa medida atribui pontuações relativas para todos os nodos na

rede baseando-se no princípio que conexões com nodos que têm uma pontuação maior fazem que o nodo em questão aumente sua pontuação.

- **“Ponte” local** – Uma aresta pode ser considerada uma “ponte” local se os seus pontos finais não têm vizinhos em comum. Ao contrário de uma “ponte”, uma “ponte” local está contida em um ciclo.

- **Comprimento de caminho** – São as distâncias entre pares de nodos em uma rede. O comprimento médio de caminho é a média dessas distâncias entre todos os pares de nodos.

- **Prestígio** – Em um grafo direcionado, prestígio é o termo utilizado para descrever a centralidade de um nodo.

- **Radialidade** – Grau que uma rede individual alcança dentro da rede e provê informação nova e influência.

- **Alcance** – Grau em que um membro de uma rede consegue alcançar qualquer outro membro da rede.

- **Coesão estrutural** – Número mínimo de membros que, se removidos de um grupo, desconectarão o grupo.

- **Equivalência estrutural** – Se refere a extensão que cada nodo possui um conjunto de *links* em comum com outros nodos do sistema. Os nodos não precisam ter nenhuma ligação entre si para serem estruturalmente equivalentes.

- **Furo estrutural** – São furos estáticos que podem ser estrategicamente preenchidos através da conexão entre um ou mais nodos.

- **Modularidade** – Mede a força da divisão de uma rede em módulos, também chamados de grupos, aglomerados ou comunidades.

Ainda não há um consenso na área de Análise de Redes Sociais sobre qual o melhor método para se usar ao fazer a análise de uma rede. Doreian e Stokman (*apud* Xu *et al*, 2004) classificaram abordagens existentes em três categorias: descritiva, estática e métodos de simulação. Essas classificações são descritas a seguir:

- **Métodos descritivos** – O propósito de uma análise descritiva é, frequentemente, detectar mudanças estruturais em redes sociais e testar o quão bem uma teoria sociológica é suportada por dados empíricos. Com métodos descritivos, propriedades estruturais de uma rede social são medidas por várias métricas e índices e comparadas através do tempo para descrever a dinâmica em nodos, ligações ou grupos na rede. Essas medidas frequentemente têm como foco a mudança na centralidade, influência e outras características dos indivíduos. Já as medidas relacionadas a ligações, dizem respeito a estabilidade dessas ligações, com relação ao rompimento e substituição das mesmas. E, por fim, medidas relacionadas a grupos, têm como foco a estabilidade de um grupo de indivíduos na rede e os processos de balanço do mesmo.

- **Métodos estatísticos** – A análise estatística da dinâmica de redes sociais tem como objetivo, não só detectar e descrever mudanças na rede, como também explicar porque essas mudanças ocorrem. Com métodos estatísticos, mudanças estruturais são assumidas como resultado de alguns processos estocásticos de efeitos na rede, como reciprocidade, transitividade e balanço. Nesse tipo de análise, ligações são modeladas como variáveis aleatórias que podem estar em diferentes estados (positivo, negativo ou neutro) ao mesmo tempo. O propósito e examinar qual efeito da rede se encaixa nos dados empíricos e descreve melhor as mudanças estruturais observadas.

- **Métodos de Simulação** – Ao contrário dos métodos descritivos e estatísticos, que examinam a dinâmica da rede social quantitativamente, os métodos de simulação dependem de tecnologia multi-agente para analisar a dinâmica de uma rede. Nesse método, membros de uma rede

> social são frequentemente modelados e implementados como agentes computacionais que têm a habilidade de se comportar e tomar decisões baseados em certos critérios. O comportamento coletivo de todos os membros da rede determinará como a rede evolui de uma estrutura para outra.

Porém, além dos métodos apresentados, pesquisas têm empregado técnicas de visualização para estudar a dinâmica de uma rede. Essa abordagem consiste em apresentações visuais de redes sociais e é bastante diferente dos métodos descritivos, estatísticos e de simulação (Xu *et al*, 2004).

Para visualizar uma rede social, um algoritmo apropriado de *layout* deve ser escolhido para determinar localizações para os nodos. Dentre diversas propostas de cálculo de distribuição de localização dos nodos, uma das que vem se destacando é a que trata a rede como um sistema no qual os nodos são conectados por molas contendo energia. Os nodos são atraídos e repelidos entre si até que a energia no sistema seja minimizada.

Como resultado do estudo realizado nessa subseção, pode-se concluir a importância das métricas na Análise de Redes Sociais, assim como a vasta quantidade de métricas existentes. Algumas dessas métricas estão incluídas na ferramenta *Gephi* que foi utilizada neste trabalho e será discutida a seguir.

2.3 *Gephi*

A análise de um conjunto muito pequeno de dados é usualmente direta. Uma rede social para um grupo de quatro ou cinco pessoas, por exemplo, pode ser facilmente construída à mão. Porém, isso se torna mais difícil quando o tamanho da rede é maior do que isso. Quando se lida com um conjunto de dados que tem mais do que cerca de dez indivíduos e cinco relações, é essencial o uso de um computador (Scott, 2000). O processamento dos dados em um computador não só poupa tempo, como também possibilita alguns tipos de análise que seriam muito mais difíceis de se fazer à mão devido ao tempo que se levaria e ao trabalho que daria.

Em SNA, o processo envolvido desde a coleta de dados até a descoberta de informações requer uma ferramenta completa para adquirir, analisar e filtrar

os dados e então apresentá-los por meio de uma visualização interativa. Devido a isso, a ferramenta *Gephi* foi escolhida. Segundo Heymann e Le Grand (2013), a *Gephi* é uma ferramenta livre e genérica que pode ser utilizada para a exploração de todos os tipos de redes sociais.

Essa ferramenta, utiliza um módulo especial de visualização 3D para apresentar grafos em tempo real (Bastian *et al,* 2009). Essa técnica utiliza a placa de vídeo do computador, assim como os videogames, e deixa a CPU (*Central Processing Unit*) livre para outras computações. A ferramenta pode lidar com grandes redes, alguns dos algoritmos inseridos nela permitem até um milhão de participantes em uma rede, e, pelo fato dela ser construída com um modelo multitarefas, ela toma vantagem de processadores multi-core. A *Gephi* também permite a personalização do *design* dos nodos, em vez de um formato clássico, eles podem ser uma textura, um painel ou uma foto. Além disso, algoritmos altamente configuráveis podem ser executados em tempo real na janela do grafo.

A *Gephi* possui algumas opções de formatos e extensões de arquivos de entrada. Por exemplo, o formato de arquivos utilizado neste trabalho foram duas planilhas eletrônicas, uma representando as informações das arestas e outra a dos nós, ambas com extensão *csv* (*Comma-Separated Values*). Essa ferramenta também oferece a implementação de uma série de algoritmos para a aplicação das métricas de SNA, além de diversas opções de interação com o *layout* da rede para facilitar a visualização. Todas estas funcionalidades, disponíveis em tempo real, permitem que o usuário possa mudar os parâmetros dos algoritmos enquanto ele ainda está rodando, e parar a sua execução quando desejar.

Portanto, devido ao fato da *Gephi* ser uma ferramenta julgada completa e que preenche todos os requisitos para adquirir, analisar e filtrar os dados de uma rede social, ela foi escolhida para utilização neste trabalho. Nessa ferramenta, estão inseridos diversos algoritmos para a montagem e manipulação do *layout* da rede, esses algoritmos serão discutidos a seguir.

2.3.1 Algoritmos da *Gephi*

Force Atlas* e *Force Atlas* 2 *layout

O *Force Atlas* foi desenvolvido por Mathieu Jacomy em 2007, é um algoritmo de força dirigida com a complexidade de $O(N^2)$. Um algoritmo de força dirigida simula um sistema físico para espacializar a rede (Jacomy *et al*, 2014). Os nós repelem uns aos outros como partículas carregadas, enquanto as arestas atraem seus nós, como molas. Essas forças criam um movimento que converge para um estado balanceado. O desenho de força dirigida tem a especificidade de determinar o lugar de cada nodo dependendo da localização de outros nodos. Esse processo depende apenas das conexões entre os nodos e eventuais atributos entre os nodos nunca são levados em consideração. Essa técnica tem a vantagem de possibilitar uma interpretação visual da estrutura da rede. Sua essência é tornar proximidades estruturais em proximidades visuais, facilitando a análise das redes sociais.

O *Force Atlas* suporta um grafo de 1 a 10.000 nodos e utiliza peso em suas arestas. Segundo Jacomy (2011), foi desenvolvido para espacializar redes sem escala. Redes sem escala são redes complexas, nas quais a maioria dos nodos tem poucas ligações, havendo, porém alguns nodos com muitas ligações, diz-se que esses nodos com muitas ligações têm um Grau alto. Nesse tipo de rede, os nodos com Grau alto têm tendência a se ligarem a outros nodos com Grau alto.

Já o algoritmo *Force Atlas* 2, que também é um algoritmo de força dirigida, nada mais é do que uma versão aperfeiçoada do algoritmo *Force Atlas*. As duas versões, *Force Atlas* e *Force Atlas 2*, estão disponíveis na *Gephi*. O *Force Atlas* 2 foi desenvolvido por Mathieu Jacomy em 2011. Ele tem uma complexidade O(N*log(N)) e suporta um grafo de 1 a 1.000.000 nodos, além de utilizar peso nas arestas. Na Figura 12 pode ser visto um exemplo de rede social gerada utilizando-se esse algoritmo.

Segundo Jacomy (2014), o *Force Atlas* 2 tem um foco maior na usabilidade do que na originalidade, não foi feita uma pesquisa bibliográfica sistemática sobre os algoritmos já existentes. Ele foi desenvolvido pensando-se

na fluência e qualidade, porque fluência é um requisito necessário para a experiência interativa do usuário com a *Gephi*, e, porque os pesquisadores preferem qualidade acima de desempenho.

Existem certas configurações no *Force Atlas* que permitem ao usuário afetar o posicionamento dos nodos, e, consequentemente, a forma da rede social. Algumas das configurações que foram implementadas no *Force Atlas* 2 são listadas a seguir:

- **Modo LingLong** – LingLong é um modelo de energia que faz os aglomerados ficarem mais estreitos. Mudar do modo normal para o modo LingLong faz com que seja necessário um ajuste na escala do grafo, pois, como ele deixa os aglomerados mais próximos, é necessário que se afaste os mesmos aumentando a escala.

- **Gravidade** – É uma melhoria comum dos algoritmos de força dirigida. Representa uma força que previne componentes desconectados de se afastarem, como pode ser visto na Figura 12. Ela atrai os nodos para o centro do espaço.

- **Escala** – Um algoritmo de força dirigida deve conter um par de constantes *ka* e *kr* tendo um papel contrário na espacialização do grafo. A constante de atração *ka* ajusta a força de atração e a constante de repulsão *kr* ajusta a força de repulsão. No *Force Atlas* 2, o usuário pode modificar o valor de *kr*, quanto maior for o valor de *kr*, maior será o grafo.

- **Peso da aresta** – Se a aresta tem peso, esse peso será levado em consideração na computação da força de atração. Se o campo “Influência do peso na aresta for definido como 0, os pesos são ignorados e, se esse campo for definido como 1, a atração será proporcional ao peso.

- **Dissuadir pontos centrais** – Quando essa opção está ativada, ela afeta a forma do grafo, dividindo a força de atração de cada nó pelo seu grau mais 1 para cada nodo que ele aponta.

- **Evitar sobreposição** – Com esse modo ativado, a repulsão é modificada para que os nodos não se sobreponham. O objetivo é produzir uma imagem mais esteticamente legível.

Figura 12 - Efeitos da gravidade. Fonte: Jacomy et al (2014)

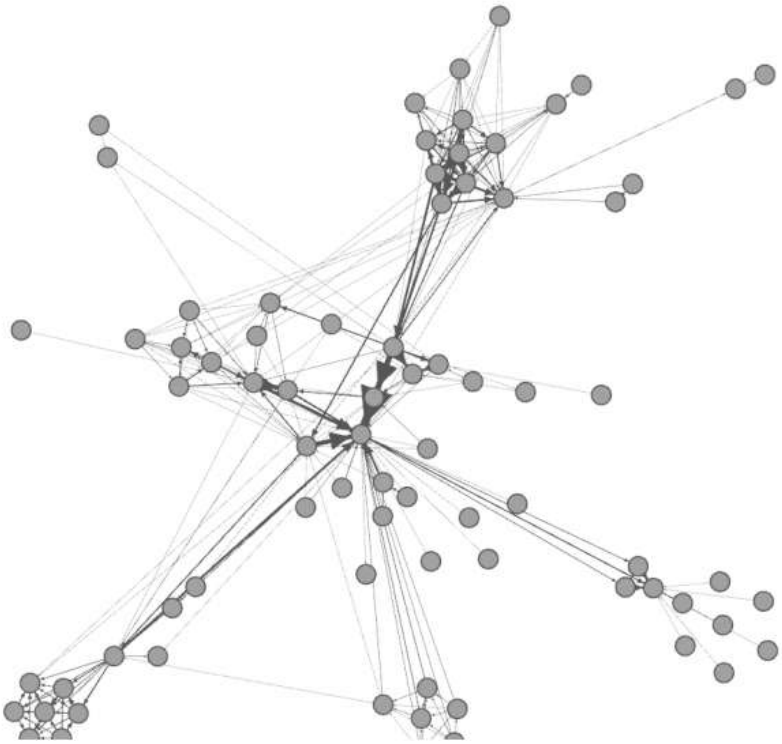


Figura 13 - Rede social gerada utilizando-se o ForceAtlas2. Fonte: Jacomy (2011)

Fruchterman-Reingold layout

Foi desenvolvido por Thomas Fruchterman e Edward Reingold em 1991, é um algoritmo de força dirigida com a complexidade de $O(N^2)$. Esse algoritmo suporta um grafo de 1 a 1.000 nodos e não utiliza peso em suas arestas. Segundo Jacomy (2011), ele se tornou um algoritmo padrão para a geração de redes sociais, mas ainda permanece bastante lento.

O algoritmo de *Fruchterman-Reigold* simula um grafo como um sistema de partículas de massa, sendo os nodos as partículas de massa e as arestas "molas" entre as partículas. O algoritmo tenta minimizar a energia desse sistema físico. Segundo Hu (2005), ele modela o grafo, desenhando o problema por um sistema de molas entre nodos vizinhos do grafo, aproximando esses nodos uns dos outros. Ao mesmo tempo, forças repulsivas elétricas que existem, empurram os nodos para longe uns dos outros. Na Figura 14 pode ser vista uma rede social gerada utilizando-se o algoritmo *Fruchterman-Reingold.*

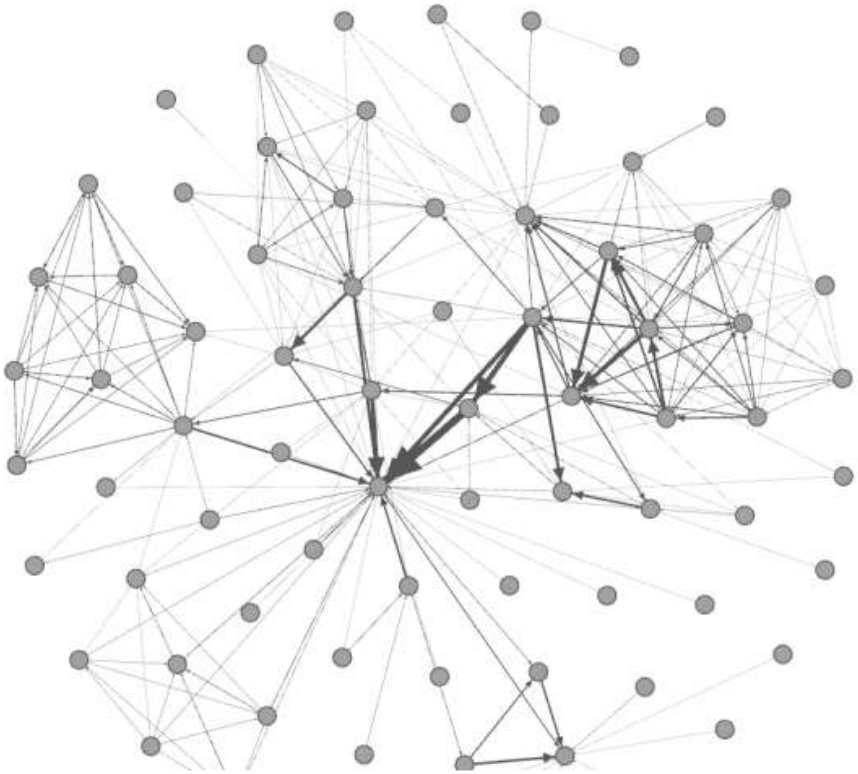


Figura 14 - Rede social gerada utilizando-se o algoritmo Fruchterman-Reingold.
Fonte: Jacomy (2011)

Yifan Hu Multinível *layout*

Foi desenvolvido por Yifan Hu em 2005, é um algoritmo de força dirigida e multinível com a complexidade de O(N*log(N)). Esse algoritmo suporta um grafo de 100 a 100.000 nodos e não utiliza peso em suas arestas. Segundo Jacomy (2011), é um algoritmo muito rápido com uma boa qualidade em grafos grandes. Ele combina um modelo de força dirigida com uma técnica de algoritmo multinível para diminuir a complexidade do grafo. Ao contrário da maioria dos algoritmos presentes na *Gephi,* que precisam ser inicializados e parados, esse algoritmo para de executar automaticamente quando encontra a posição mais adequada para cada nó. Segundo Hu (2005), esse algoritmo modela o grafo desenhando o problema por um sistema físico de corpos, com forças agindo entre eles. O algoritmo encontra uma boa localização dos corpos, onde não estejam nem aglomerados e nem muito afastados uns dos outros, minimizando a energia do sistema. Na Figura 15 é possível ver uma rede gerada utilizando-se o algoritmo Yifan Hu.

Em um algoritmo de força dirigida, a energia do sistema é tipicamente minimizada movimentando iterativamente os nodos ao longo da direção da força ao qual eles estão sendo submetidos (Hu, 2005). Inicialmente, a energia do sistema pode ser grande, porém, ela diminui gradualmente baseada em um “cronograma de resfriamento”. Segundo Hu (2005), existem dois fatores limitantes para algoritmos de força dirigida comuns desenharem grandes grafos. O primeiro é que o modelo físico tipicamente contém muitos mínimos locais, particularmente para grandes grafos. O segundo fator limitante, é a complexidade computacional dos algoritmos de força dirigida padrão. Para subjugar o primeiro fator limitante, foi proposta uma abordagem multinível.

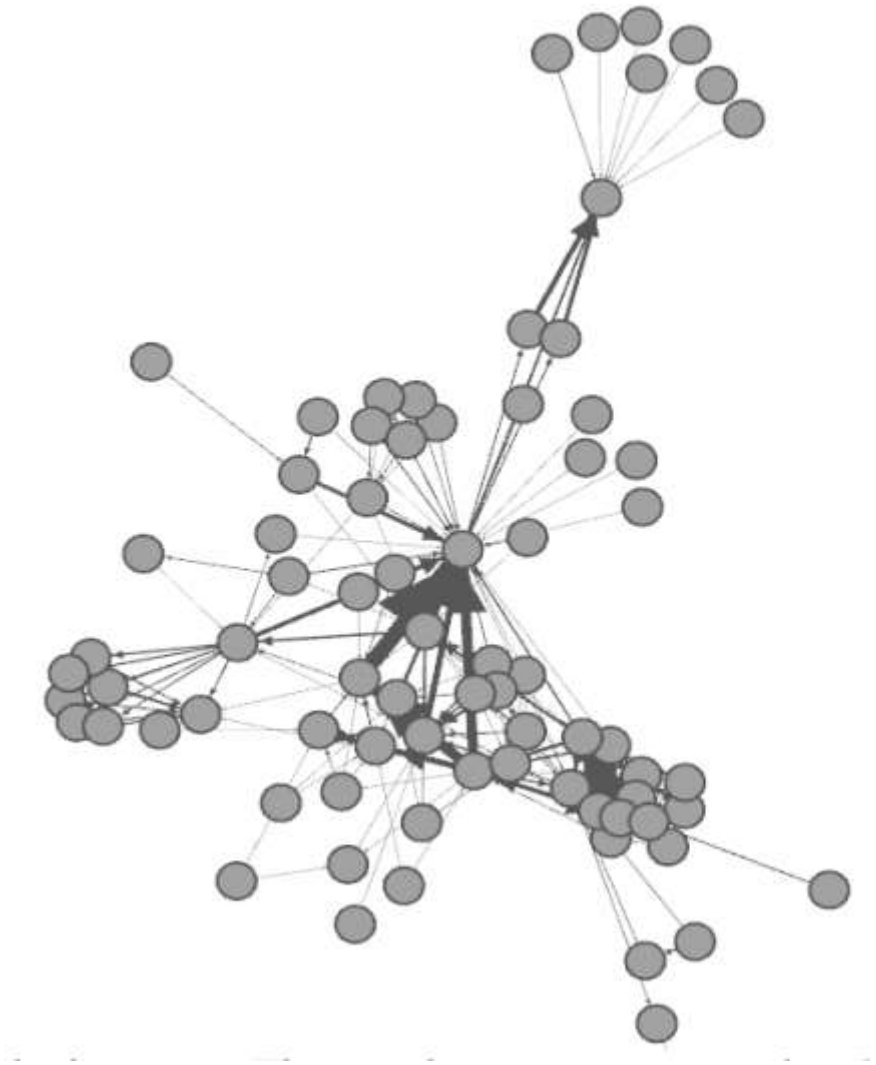


Figura 15 – Rede gerada utilizando-se o algoritmo Yifan Hu. Fonte Jacomy (2011)

OpenOrd layout

Foi desenvolvido por S. Martin, W. M. Brown, R. Klavans e K. Boyack em 2010, é um algoritmo de força dirigida e arrefecimento simulado com a complexidade de O(N*log(N)). Esse algoritmo suporta um grafo de 100 a 1.000.000 de nodos e utiliza peso em suas arestas. Esse algoritmo tem um desempenho melhor com grafos não-direcionados e com peso e tem como objetivo distinguir melhor os aglomerados. Ele pode ser executado em paralelo para acelerar a computação e termina a sua execução automaticamente quando determina a melhor

localização de cada nó. Esse algoritmo é originalmente baseado no algoritmo *Frutcherman-Reingold* e trabalha com um número fixo de iterações controlado via um tipo de cronograma de arrefecimento simulado (líquido, expansão, esfriamento, estalo e ebulição). Na Figura 16 é possível ver uma rede gerada utilizando-se o algoritmo *OpenOrd*.

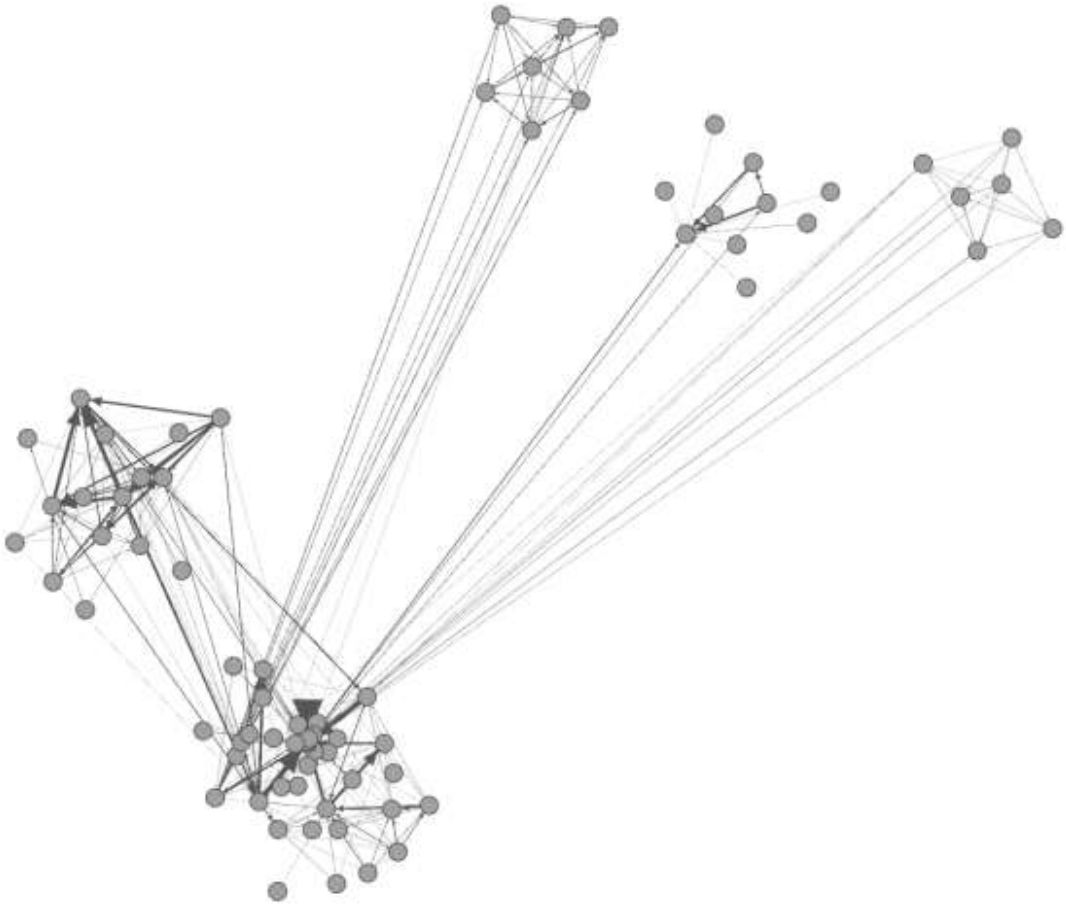


Figura 16 - Rede gerada utilizando-se o algoritmo *OpenOrd*. Fonte Jacomy (2011)

Layout **Circular**

Foi desenvolvido por Matt Groeninger em 2010 e é do tipo circular, ele tem complexidade O(N) e suporta um grafo de 1 a 1.000.000 de nodos. Ele simplesmente desenha os nodos em um círculo ordenados por ID, por alguma métrica ou por um atributo. Na Figura 17 é possível ver uma rede gerada utilizando-se o algoritmo Circular.

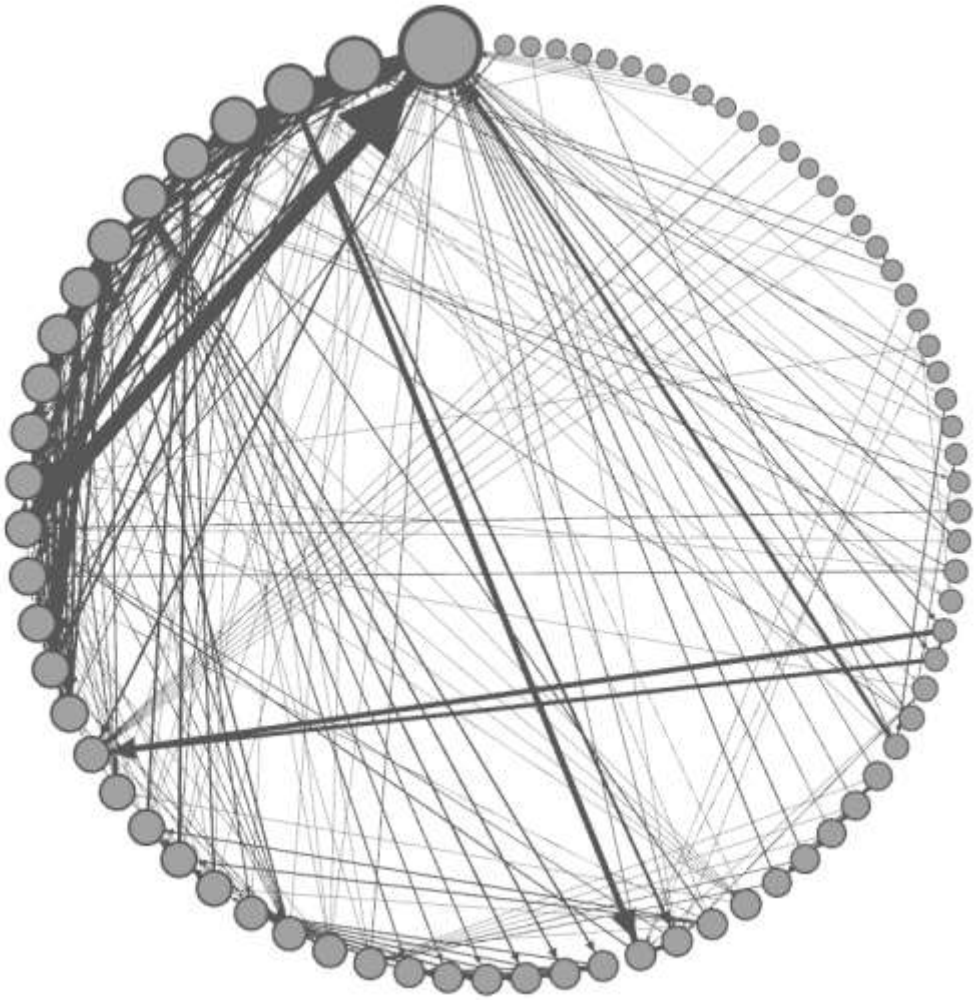


Figura 17– Rede gerada utilizando-se o algoritmo Circular. Fonte Jacomy (2011)

Layout **de Eixo Radial**

Segundo Jacomy (2011), é um algoritmo do tipo circular, foi desenvolvido por Matt Groeninger em 2011 e tem complexidade O(N), além de suportar um grafo de 1 a 1.000.000 de nodos. Esse algoritmo é provido com o *plugin* do algoritmo do *layout* Circular. Ele agrupa nodos e os desenha em grupos em eixos que irradiam de um círculo central para fora. Estes grupos são gerados utilizando alguma métrica ou atributo. Na Figura 18 é possível ver um exemplo de como pode ficar uma rede gerada utilizando-se esse algoritmo.

Segundo Tominski *et al* (2004), visualizações baseadas em eixos são ferramentas úteis para a análise de conjuntos de dados multidimensionais. Esse

tipo de projeção tem a vantagem de não ter perdas de visualização na projeção de dados que constituam uma rede com n-dimensões em uma tela de computador com duas dimensões.

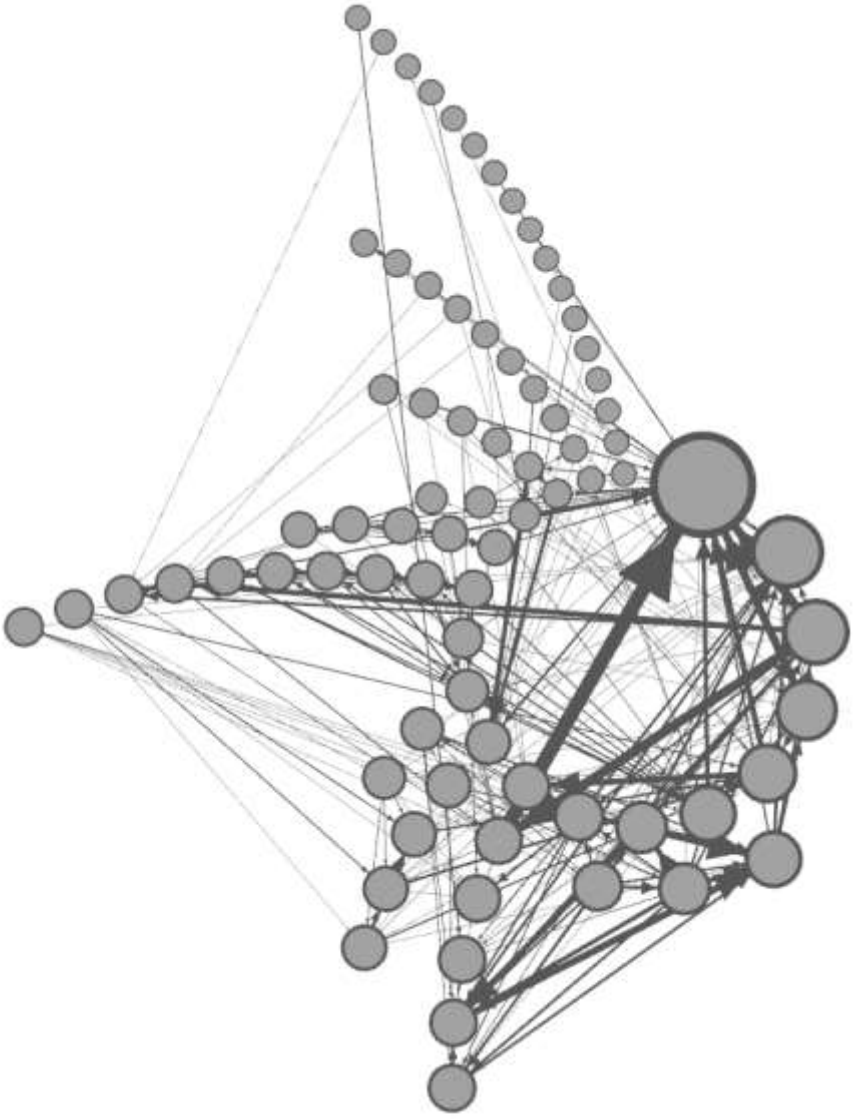


Figura 18 – Rede social gerada utilizando-se o algoritmo de Eixo Radial

Geo*Layout*

Segundo Jacomy (2011), o *GeoLayout* foi desenvolvido por Alexis Jacomy em 2010, é do tipo geográfico com uma complexidade O (N), ele suporta um grafo com um tamanho de 1 a 1.000.000 de nodos. Esse algoritmo utiliza coordenadas geográficas, latitude e longitude, para definir a posição dos nodos na rede. Várias

projeções estão disponíveis, incluindo *Mercator*, que é utilizada pelo *Google Maps* e outros serviços online.

2.4 Trabalhos Correlatos

Alguns trabalhos têm abordado a área de na Análise de Redes Sociais Criminais, como em Pereira *et al* (2015), que apresenta uma análise de padrões de tatuagens associadas às ocorrências de crimes registrados pela Secretaria de Segurança Pública do Estado da Bahia. O trabalho aponta dados relevantes para a compreensão de padrões de criminalidade que podem ajudar os órgãos de segurança pública na tomada de decisão.

Outro trabalho relacionado com a Análise de Redes Sociais voltado para a área criminal é o de Rostami e Mondani (2015), que estuda uma gangue sueca a partir de três diferentes conjuntos de dados. Neste trabalho, os dados obtidos foram utilizados para construir redes e compará-las através da aplicação de métricas de redes sociais. O estudo mostra um mesmo grupo de pessoas analisado por meio de três diferentes fontes de dados, destacando a importância da fonte dos dados na Análise de Redes Criminais. Além disso, ele tem como objetivo contribuir para fortalecer a Análise de Redes Sociais como uma ferramenta confiável para entender e analisar a criminalidade e as redes criminais.

Ademais, outra área onde as técnicas de SNA são fortemente aplicadas é no combate ao terrorismo. Podendo-se citar como exemplo o trabalho de Ressler (2006), que faz uma revisão da área de Análise de Redes Sociais e seu uso na pesquisa sobre o terrorismo. Esse trabalho discute o uso de SNA antes dos atentados de 11 de setembro aos Estados Unidos e o aumento da utilização dessa técnica após os atentados. Ele divide os estudiosos da área em dois grupos, os coletores de dados e os modeladores. Onde os coletores de dados têm como principal função coletar a maior quantidade de dados possível sobre uma rede criminal e os modeladores modelam a rede. O trabalho tem como objetivo demonstrar a importância de SNA e que de fato ela está sendo usada pelos governos, principalmente governo federal dos Estados Unidos, para prevenir e combater o terrorismo.

Além desses trabalhos, é válido citar também o trabalho de Koschade (2006), que teve como objetivo contribuir para o estudo de SNA através da Análise de Rede Social do celular de Jemaah Islamyah, que foi responsável pelas bombas de Bali, Indonésia, em 2012. Além disso, o trabalho forneceu um potencial *framework* para a análise inteligente dos celulares de terroristas. Esse *framework* tem como objetivo ajudar na compreensão da estrutura dos celulares e ajudar a prever os prováveis resultados dos celulares dos terroristas quando empregado em análise de inteligência em tempo real.

A partir dos conhecimentos adquiridos com esses trabalhos correlatos e, principalmente, do estudo realizado na área de Análise de Redes Sociais, foi possível desenvolver uma abordagem e implementação para esse este trabalho. A abordagem da identificação e da análise de redes sociais na cidade de Bagé, assim como desenvolvimento de uma ferramenta que possa auxiliar a BM de Bagé na prevenção e combate à criminalidade, são descritas na seção a seguir.

3 ABORDAGEM E IMPLEMENTAÇÃO

Neste capítulo serão apresentadas a abordagem proposta para o trabalho, como o tipo de dados e o formato dos mesmos disponibilizados pela BM de Bagé. Assim como a identificação e análise das redes sociais se utilizando os dados recebidos. Além disso, será descrita a implementação da ferramenta desenvolvida para a BM de Bagé.

3.1 Visão geral da abordagem

A abordagem proposta para a montagem das redes sociais criminais de Bagé baseia-se em receber da Brigada Militar da cidade dados sobre a participação de indivíduos em ocorrências criminais e aprimorar esses dados para que os mesmos possam ser manipulados pela ferramenta *Gephi* (gerar os arquivos de entrada da ferramenta). Além disso, realizar a montagem dos grafos e aplicar as métricas de SNA nas redes.

3.2 Descrição da abordagem

O setor de inteligência da Brigada Militar de Bagé disponibilizou, em um primeiro momento, os dados que foram previamente retirados por eles de boletins de ocorrência, como os apresentados na Tabela 4. Estes são dados reais e permitiram que já fosse possível na primeira parte do Trabalho de Conclusão de Curso (TCCI) o desenvolvimento de uma rede criminal da cidade de Bagé.

A Tabela 4 apresenta trechos de uma tabela de 164 registros disponibilizada pela BM de Bagé para a utilização neste trabalho. A primeira coluna da tabela contém o número (o identificador) de uma ocorrência policial, e, na sua linha correspondente na segunda coluna, o número de identificação de um criminoso. Esses números estão cadastrados no banco de dados da BM e só esta sabe qual ocorrência e qual criminoso eles representam. A partir da referida tabela, é possível verificar quais criminosos encontram-se em quais ocorrências e, consequentemente, identificar quais foram citados em uma mesma ocorrência.

	A	B
1	**idOc**	**idDel**
2	627	149
3	627	541
4	627	542
5	627	1016
6	627	1034
7	627	1035
8	627	1036
9	1092	401
10	1092	464
59	7495	193
60	7600	193
61	7600	345
62	7600	952
63	7600	1000
64	7600	1083
65	7600	1309
66	7600	1409
67	7600	1435
68	7600	1436
69	7600	1437
70	7730	193
71	7730	952
72	7730	1000
73	7730	1409
74	7730	1432
75	7730	1433
76	7730	1434
77	7860	193

Tabela 4 – Tabela de ocorrências e delinquentes disponibilizada pela BM.
Fonte: Brigada Militar de Bagé.

Porém, é necessário que se manipule essa tabela para que se obtenha as duas tabelas (de nós e de arestas), que são as entradas para a ferramenta *Gephi.* Para isso, foi desenvolvida uma ferramenta que, a partir dos dados disponibilizados no referido formato, se possa obter as tabelas de nós e arestas necessárias para a *Gephi*. Porém, na primeira parte do trabalho, essa ferramenta ainda não havia sido desenvolvida. Mas, para que fosse possível uma montagem e análise preliminar da rede, as tabelas de nós e arestas correspondentes aos dados disponibilizados pela BM foram geradas manualmente. Na Tabela 5, a tabela dos nós, é possível visualizar na primeira coluna o ID do criminoso e na segunda o rótulo (*Label*). O ID representa o

número identificador do criminoso, já o *Label* é um campo onde, geralmente, coloca-se o nome de um indivíduo da rede. Porém, a fim de preservar a confidencialidade dos dados, foi inserido novamente o ID do criminoso, para que ao visualizar a rede, apareça o ID em vez do nome.

	A	B
1	ID	Label
2	149	149
3	541	541
4	542	542
5	1016	1016
6	1034	1034
7	1035	1035
8	1036	1036
9	401	401
10	464	464
11	824	824
12	931	931
13	858	858
14	1442	1442
15	1443	1443
16	1444	1444
17	1197	1197
18	1198	1198
19	1199	1199
20	1427	1427
21	973	973
22	974	974
23	975	975
24	890	890
25	1440	1440
26	297	297
27	1242	1242
28	1080	1080
29	1000	1000
30	826	826

Tabela 5 – Tabela dos nós. Fonte: Arquivo pessoal

Já na Tabela 6, que é a tabela das arestas, é possível visualizar 4 colunas. As colunas *source* e *target* correspondem, respectivamente, à fonte e ao alvo (ambos nodos do grafo da rede) de uma aresta no grafo. Ou seja, são dois nodos que possuem uma ligação, onde a ligação vai da fonte ao alvo. Porém, como é possível verificar na coluna 4, coluna que corresponde ao tipo do grafo, o grafo

utilizado não é direcionado (*undirected*). Portanto, *source* e *target* correspondem somente a dois nós que possuem uma ligação, que representam dois criminosos que foram citados em uma mesma ocorrência. Neste caso, não importa qual criminoso irá no campo *source* e qual irá no campo *target*. Só é necessário que o ID do criminoso presente na tabela dos nós esteja em um dos campos e o ID de outro criminoso que foi citado junto em uma mesma ocorrência esteja no outro campo da linha correspondente. Na terceira coluna (*Weight)*, está contido o peso da ligação formada pelos ID's nos campos *Source* e *Target*. Este peso representa o número de vezes em que o par de criminosos desses campos foram encontrados juntos em uma mesma ocorrência. Como, por exemplo, os criminosos de ID 193 e 1000, que são vistos juntos nas ocorrências 7600 e 7730 (como pode ser visto na Tabela 4). E, por isso, receberam peso 2 (como pode ser visto na Tabela 6 – linha 64).

	A	B	C	D
1	Source	Target	Weight	Type
2	149	541	1	undirected
3	149	542	1	undirected
4	149	1016	1	undirected
5	149	1034	1	undirected
6	149	1035	1	undirected
7	149	1036	1	undirected
8	541	542	1	undirected
9	541	1016	1	undirected
10	541	1034	1	undirected

45	975	1197	1	undirected
46	890	1440	1	undirected
47	297	1242	4	undirected
48	464	1080	1	undirected

61	108	193	1	undirected
62	193	345	1	undirected
63	193	952	3	undirected
64	193	1000	2	undirected
65	193	1309	1	undirected

Tabela 6 – Tabela das arestas. Fonte: Arquivo pessoal

Depois de se obter as tabelas dos nós e das arestas, essas tabelas foram utilizadas como entrada para a ferramenta *Gephi* para que fosse gerada e analisada uma rede social preliminar. Essa primeira análise está descrita na seção de implementação.

Em um segundo momento (TCCII), foi disponibilizado pela Brigada Militar, um novo conjunto de dados. O formato desses dados é semelhante ao da Tabela 4, a única diferença é que possui uma coluna a mais que contém o endereço da ocorrência. Esses endereços foram disponibilizados para que fosse possível se montar uma rede criminal georreferenciada, eles estão escritos em uma coluna no formato “rua, número, bairro”. Por questão de confidencialidade, pois a BM enviou dados de ocorrências reais, não será mostrada uma imagem desse arquivo.

Esse segundo arquivo foi utilizado como entrada na ferramenta desenvolvida, que já estava finalizada, para gerar os arquivos de nós e arestas. Esses arquivos possuem o formato dos arquivos mostrados nas Tabelas 5 e 6, porém, com uma pequena diferença. Na implementação realizada na primeira parte do trabalho, foi possível perceber que a *Gephi* coloca automaticamente os pesos das arestas. Portanto, a ferramenta foi desenvolvida para que os pesos das arestas sejam sempre 1 e tenham quantas correspondências forem geradas de fonte e alvo. Como, por exemplo, se em uma mesma ocorrência do arquivo de entrada forem encontrados os delinquentes de ID 1 e 2, gerando uma aresta no arquivo de arestas onde o 1 estará na fonte e o 2 no alvo, ou vice-versa. E, em outra ocorrência, esses mesmos dois delinquentes foram encontrados juntos novamente. Será gerada então uma outra linha no arquivo de arestas onde esses dois ID’s estarão na fonte e no alvo, em vez de ser adicionado 1 no peso da aresta já existente. Após o arquivo de arestas ser inserido na *Gephi*, ela montará uma aresta só para os ID’s 1 e 2 e colocará o peso de acordo com a quantidade de vezes em que eles apareceram na fonte e no alvo do arquivo de arestas.

Após serem gerados os arquivos necessários, de nós e arestas, eles foram inseridos na *Gephi*. Os dados contidos nesses arquivos foram utilizados para fazer a montagem e análise de doze diferentes redes sociais criminais, utilizando-se os algoritmos *Force Atlas*, Circular, Eixo Radial, *Yifan Hu*, *Open*

Ord e *Fruchterman-Reingold* descritos na seção 2.3.1. Essa análise pode ser vista na seção de implementação.

3.4 Ferramenta Desenvolvida para a Brigada Militar

Segundo a Brigada Militar de Bagé, são registradas cerca de 450 ocorrências por mês na cidade e, em cada ocorrência, participam em média de 3 a 4 indivíduos. Devido a essa significativa quantidade de dados, há uma grande dificuldade em fazer uma análise de quais indivíduos costumam ser encontrados juntos em uma mesma ocorrência. As técnicas de Análise de Redes Sociais solucionam o problema da análise, porém, a manipulação manual dessa grande quantidade de dados, ainda demandaria muito tempo e pessoal. Devido a isto, foi proposta a utilização da ferramenta *Gephi* para realizar a montagem e análise da rede. Ainda assim, a *Gephi* demanda um certo padrão para os dados de entrada, tornando necessário que se manipule os dados que a BM possui para que se adequem a esse padrão. Portanto, uma das etapas deste trabalho, consistiu em desenvolver uma ferramenta para auxiliar a Brigada Militar. Tal ferramenta tem como objetivo processar os dados exportados pelo banco de dados da Brigada Militar, de forma que se adequem ao formato de entrada da *Gephi*. O formato de entrada da *Gephi* demanda a construção de duas tabelas *xls* com estrutura bem definida. Por isso, esse processamento é necessário, uma vez que, com um grande número de ocorrências demandaria uma quantidade muito grande de tempo a construção, de forma manual, as tabelas de entrada da ferramenta. Além disso, através da construção de tais tabelas via software se pode garantir que não serão inseridos erros na rede social criminal. Logo, justifica-se a implementação desta ferramenta.

3.4.1 Ambiente de Desenvolvimento

A ferramenta foi desenvolvida se utilizando a linguagem de programação Java, devido a familiarização da autora com essa linguagem e da facilidade na implementação de interfaces. Além disso, ela possui as API'S necessárias ao

trabalho. A IDE utilizada para a programação foi o Netbeans[2], pois possui vasta gama de bibliotecas e *plug-ins* que dão suporte ao programador.

3.4.2 Análise de Requisitos

Foram realizadas reuniões com a Brigada Militar de Bagé e analisados os dados disponíveis para definir os requisitos necessários à ferramenta. Como requisitos funcionais do *software* foram definidos:

- A ferramenta deve processar os dados provenientes do banco de dados da Brigada Militar de Bagé;
- A ferramenta deve apresentar uma interface intuitiva e de fácil utilização do ponto de vista da equipe do setor de inteligência da BM de Bagé;
- A ferramenta deve exportar os arquivos necessários para entrada na ferramenta *Gephi*.

Como requisito não funcional do *software* foi definido:

- A ferramenta deve ser desenvolvida se utilizando ferramentas gratuitas.

3.4.3 Diagrama de Arquitetura

O Diagrama de arquitetura da ferramenta desenvolvida pode ser visto na Figura 19. Primeiramente, o usuário faz a seleção do diretório onde está localizado o arquivo de entrada, nesse mesmo diretório serão gerados os arquivos de saída. Logo após, o usuário seleciona o arquivo de entrada. Por fim, a ferramenta de Redes Criminais processará o arquivo de entrada e exportará os dois arquivos de saída, *Nós.xls* e *Arestas.xls*.

[2] https://netbeans.org/

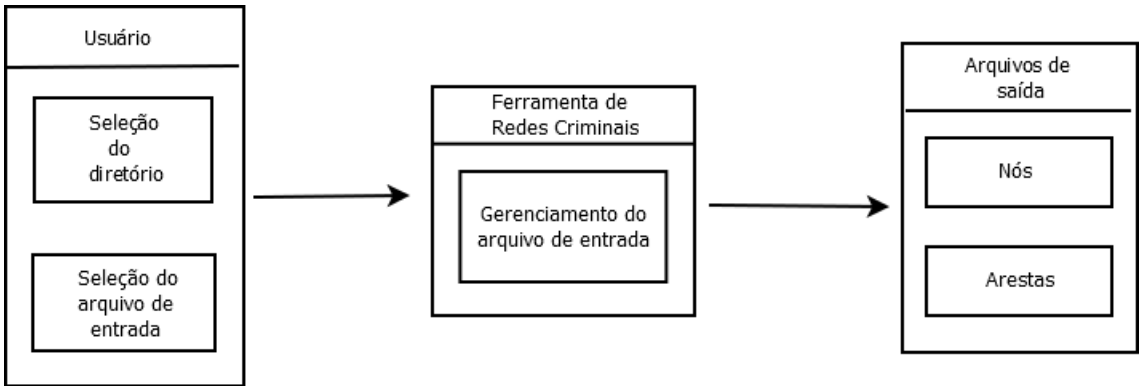


Figura 19 – Diagrama de arquitetura. Fonte: Arquivo pessoal

3.4.4 Diagrama de Classes

O Diagrama de classes da ferramenta é mostrado na Figura 20. A classe *Redes Criminais* realiza a inicialização do *software,* já a classe *Interface* implementa a parte gráfica do *software*, onde ficam localizados os botões para abrir diretórios e selecionar arquivos e onde se escolhe entre a rede normal ou georreferenciada. A classe GeocodeImplementation é chamada para o caso de ser escolhida a rede georreferenciada, ela implementa uma função que recebe endereços e retorna coordenadas geográficas (latitude e longitude) com o auxílio da API (Application Programming Interface) *Google Maps Geocoding*[3]. As classes Workbook, Sheet e Cell implementam funções para lidar com planilhas *xls*, páginas e células. Já a IDNome define um objeto que armazena o nome e o ID de cada delinquente durante o processo de leitura e geração dos arquivos de parâmetros da *Gephi.*

[3] https://developers.google.com/maps/documentation/geocoding/intro?hl=pt-br

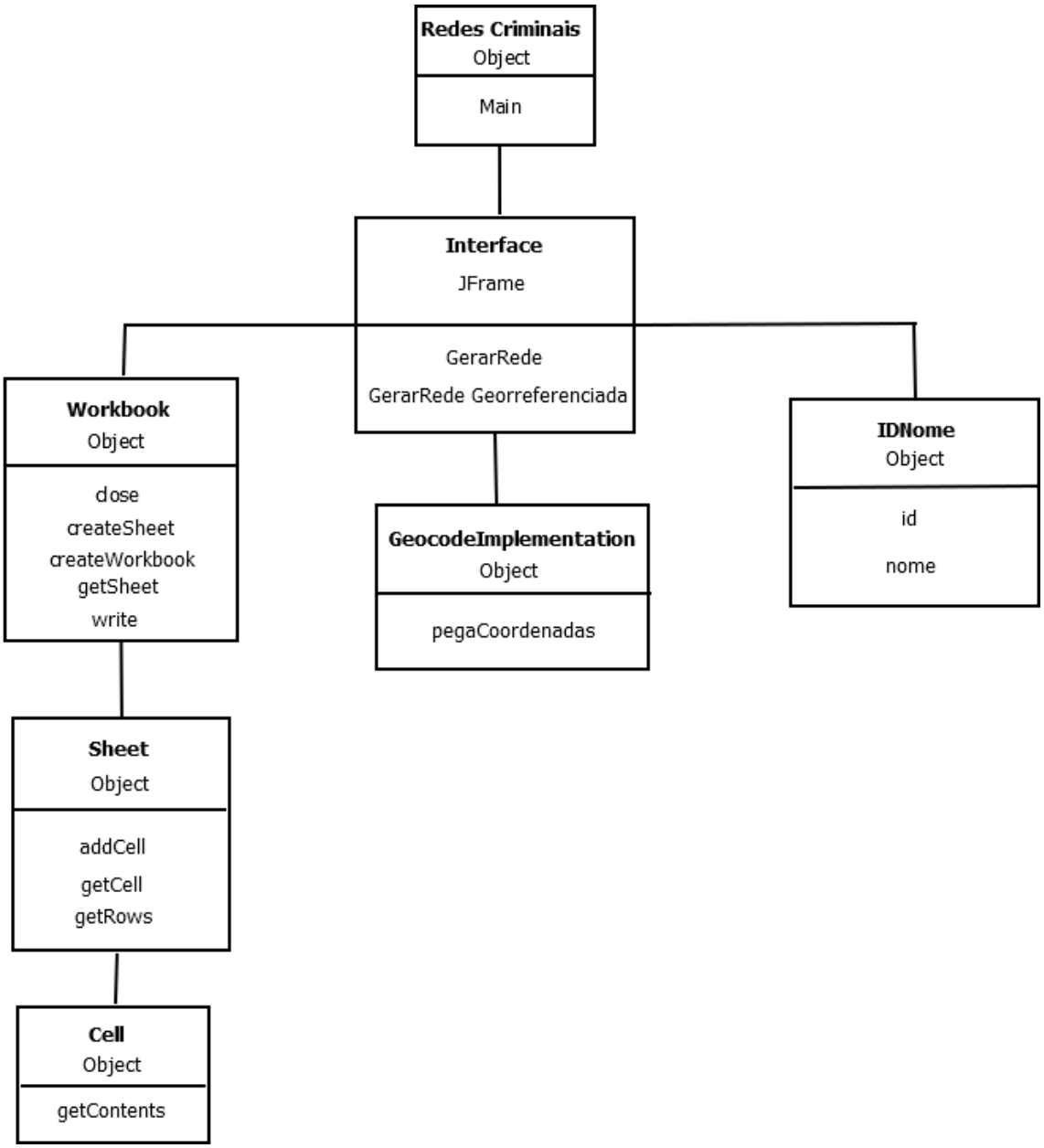


Figura 20 – Diagrama de classes. Fonte: Arquivo pessoal

3.4.5 Desenvolvimento

Para facilitar a utilização da ferramenta por parte da BM, foi desenvolvida uma interface de entrada simplificada, como pode ser vista na Figura 21. Essa interface contém apenas dois botões, um para rede normal e outro para rede georreferenciada, e permite de forma prática que, com apenas alguns cliques, o usuário consiga obter as entradas padrão da *Gephi*.

Figura 21 - Interface de entrada da ferramenta desenvolvida para a Brigada Militar de Bagé. Fonte: Arquivo pessoal

Na Figura 21, é possível observar os dois botões, o “Normal”, que é responsável por iniciar a execução do gerenciamento do arquivo para as redes que não são georreferenciadas. E o botão “Georreferenciada”, que inicia a execução do gerenciamento do arquivo para as redes georreferenciadas. Quando qualquer um desses botões é acionado, um *jFileChooser* será prototipado, como pode ser visto na Figura 22, onde o usuário pode navegar entre os diretórios do seu computador e selecionar o arquivo desejado. Para fins de praticidade o *jFileChooser*[4] foi configurado para encontrar apenas diretórios e arquivos no formato xls.

[4] https://docs.oracle.com/javase/7/docs/api/javax/swing/JFileChooser.html

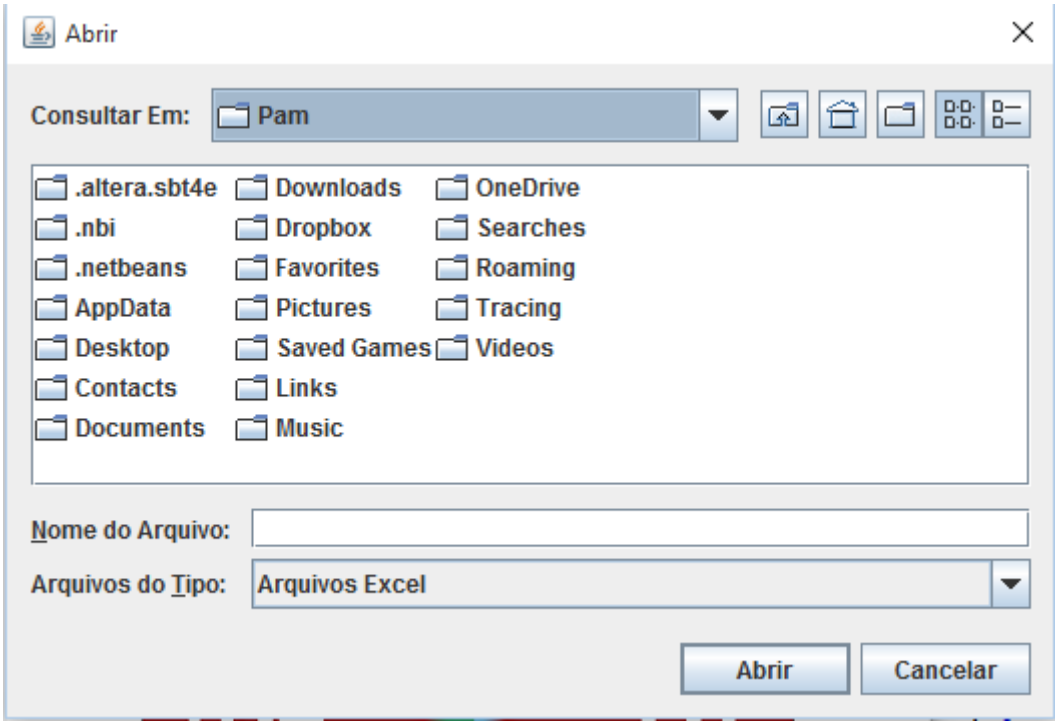


Figura 22 - Janela de navegação para escolha de arquivo. Fonte: Arquivo pessoal

O arquivo de entrada para o caso “Normal” será um arquivo com extensão *xls* contendo as colunas de *id* da ocorrência, *id* do delinquente e nome do delinquente, nessa ordem. Já o arquivo de entrada para o caso “Georreferenciada” será um arquivo com extensão *xls* contendo as colunas *id* do delinquente, *id* da ocorrência, endereço da ocorrência. Sendo que o endereço estará no formato “rua, número – bairro, Bagé – RS, Brasil”, porém, o número não é essencial para o funcionamento. Pois, caso não tiver o número, serão buscadas coordenadas centrais da rua. Nas duas alternativas, após o usuário escolher o arquivo desejado e clicar em *Abrir,* a ferramenta inicia sua execução automaticamente, incrementando uma barra de progresso e emitindo uma mensagem de conclusão tão logo o processo é encerrado, como pode ser visto na Figura 23.

O processamento de planilhas do tipo xls em Java requer a implementação de diversas funções como leitura, escrita e alteração de arquivos. Para facilitar a implementação da ferramenta, foi utilizada a API Java Excel[5], pois se trata de uma API madura, de simples utilização e com ampla documentação.

A execução da ferramenta no caso “Normal” inicia pela leitura de todas as ocorrências presentes no arquivo de entrada. A partir desses dados é realizada uma ordenação pela coluna *idDel*, que contém os ids dos delinquentes. Além da ordenação são excluídos os valores repetidos, fazendo o mesmo com os seus respectivos nomes. Após a ordenação, um novo arquivo chamado nós.xls é criado, no mesmo diretório em que está o executável da ferramenta, contendo o resultado da ordenação anterior. Este arquivo possui uma coluna com *ids* e outra com os nomes correspondentes dos delinquentes.

Após o arquivo nós.xls ser criado, a coluna de ocorrências do arquivo inicial é percorrida. A primeira ocorrência é selecionada e são percorridos os *ids* dos delinquentes encontrados nessa ocorrência. Os *ids* dos delinquentes são escritos em um novo arquivo de maneira que todos os *ids* de delinquentes presentes em uma mesma ocorrência tenham uma ligação uns com os outros, estando presentes na coluna *source* ou *target.* Por exemplo, se os delinquentes 4, 5 e 6 estiverem presentes na ocorrência 1, serão geradas 3 linhas no arquivo com “4 e 5”, “4 e 6” e “5 e 6” nas colunas *source* e *target* respectivamente. O arquivo inicial é percorrido até o final selecionando cada ocorrência por vez e repetindo o mesmo processo. Por fim, é gerado um arquivo arestas.xls e é emitido o aviso de encerramento de execução do programa. O arquivo de arestas possui as colunas *source, target, weight* e *type*, onde *source* e *target* contêm os *ids* de dois delinquentes, *weight* contém o valor 1 para todos os casos e *type* contém a informação *undirected* para todos os casos.

Para o caso de ter sido escolhida a opção “Georreferenciada”, o arquivo de entrada será um arquivo com extensão *xls* com as colunas *id* do delinquente, *id* da ocorrência e endereço. O processamento das duas primeiras colunas funciona da mesma maneira que no caso “Normal”, a única diferença é que,

[5] http://jexcelapi.sourceforge.net/

como a entrada das colunas está invertida, elas serão processadas de forma invertida. Assim, os nós serão representados pelas ocorrências e as arestas serão representadas pelos delinquentes. Já o processamento da coluna endereços, é realizado pela função “pegaCoordenadas”, que recebe os endereços lidos da coluna e retorna duas informações, a latitude e a longitude. Os arquivos “nós “ e “arestas”, nesse caso, também terão a extensão *xls.* O arquivo “nós” irá conter uma coluna com o *id* da ocorrência e outra com o endereço correspondente. Já o arquivo “arestas”, irá conter as colunas *source, target, weight, type,* latitude e longitude. *Source* e *target* contêm os *ids* de duas ocorrências, *weight* contém o valor 1 para todos os casos, *type* contém a informação *undirected* para todos os casos, latitude contém o valor da latitude da ocorrência e longitude contém o valor da longitude da ocorrência.

Figura 23 - Barra de progresso e mensagem de encerramento de execução do programa. Fonte: Arquivo Pessoal

3.3 Implementação

A implementação foi realizada em duas etapas. Na primeira etapa do trabalho, (TCCI) foi utilizado um primeiro conjunto de dados disponibilizado pela Brigada Militar de Bagé. Esse conjunto de dados foi manipulado manualmente e foram gerados os arquivos de entrada necessários para a *Gephi.* Foi feita uma geração e análise superficial de uma rede social criminal utilizado um algoritmo para a manipulação do *layout* da rede. Assim como, foram aplicadas, para a análise da rede, quatro métricas que mais se adaptavam às necessidades da análise de uma rede social criminal e às necessidades da BM de Bagé. Essa primeira análise possibilitou a determinação dos melhores formatos para os arquivos de entrada na *Gephi* e, principalmente, os requisitos necessários para o

desenvolvimento da ferramenta que iria automatizar o processo da criação desses arquivos.

Já, na segunda etapa (TCCII), foi recebido da Brigada Militar de Bagé um novo conjunto de dados um pouco mais completo. Esses dados foram inseridos na ferramenta desenvolvida e foram gerados automaticamente os arquivos de nós a arestas necessários. Esses arquivos foram utilizados como entrada na *Gephi* e, a partir da pesquisa previamente realizada sobre os algoritmos inseridos nela, foram geradas diversas redes criminais com diferentes *layouts.* Assim como, nessas redes também foram aplicadas as métricas necessárias para fazer a análise das mesmas. Além disso, foi possível, também, utilizar os endereços das ocorrências disponibilizadas pela BM para gerar uma rede criminal georreferenciada. Essa segunda etapa foi importante para fazer uma análise mais profunda dos algoritmos da *Gephi* e determinar os mais adequados para a utilização da BM de Bagé. Ela também foi essencial para a determinação dos requisitos necessários para que a ferramenta implementada conseguisse, além de gerar os arquivos necessários para entrada na *Gephi*, também obter latitudes e longitudes a partir de endereços para que fosse possível a geração de redes georreferenciadas. Nessa seção, serão descritas as duas etapas.

3.3.1 Primeiro conjunto de dados

Utilizando-se como entrada na ferramenta *Gephi* a tabela dos nós e a tabela das arestas, geradas a partir do primeiro conjunto de dados disponibilizado pela BM, o primeiro *layout* da rede social gerada foi o da Figura 24. É possível observar as diferentes espessuras das arestas da rede, o que demonstra que existem algumas ligações entre nós que são mais fortes que outras. Porém, o *layout* da rede apresenta uma difícil visualização, com um emaranhado de arestas sobrepostas e sem identificação dos nós.

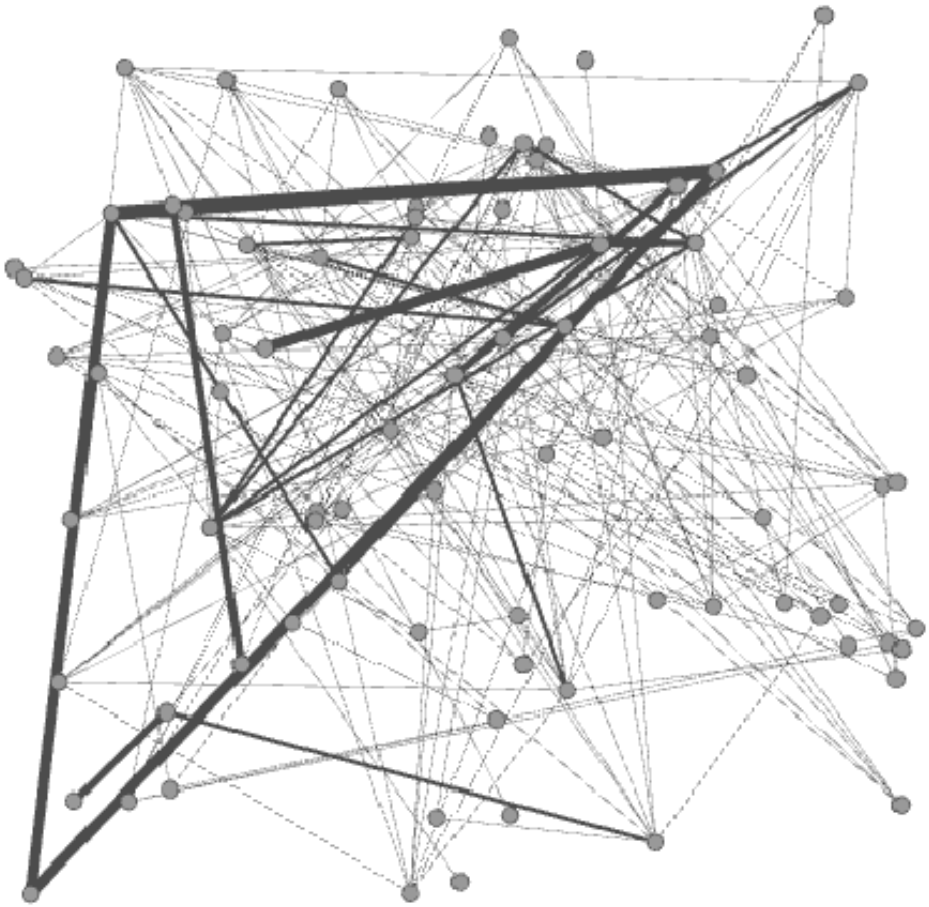


Figura 24 - Primeiro layout da rede social. Fonte: Arquivo Pessoal

Após executar o algoritmo *Force Atlas,* que é um algoritmo para melhorar o *layout* da rede que vem como opção na ferramenta *Gephi*, a rede criminal ficou como é apresentada na Figura 25.

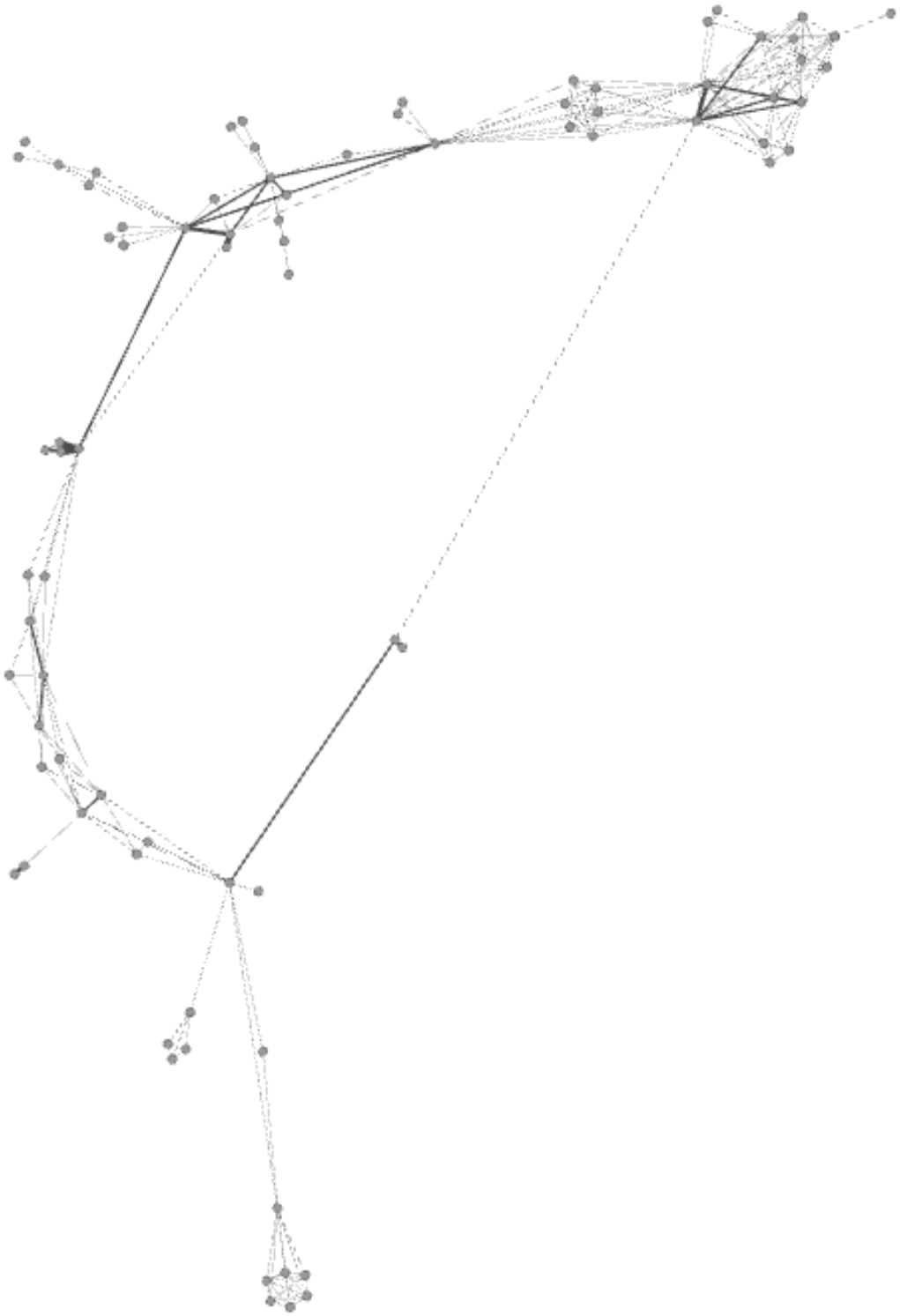


Figura 25 - Rede social após o algoritmo Force Atlas. Fonte: Arquivo pessoal

Após a mudança no *layout* realizada pelo algoritmo *Force Atlas* já foi possível ter uma melhor visualização da rede e perceber alguns aglomerados e ligações mais fortes. A seguir, foi aplicada a métrica Grau da *Gephi* e, com isso, foi possível obter uma tabela com uma classificação dos nós (criminosos) com

mais ligações na rede. A Tabela 7 apresenta os primeiros resultados dessa classificação.

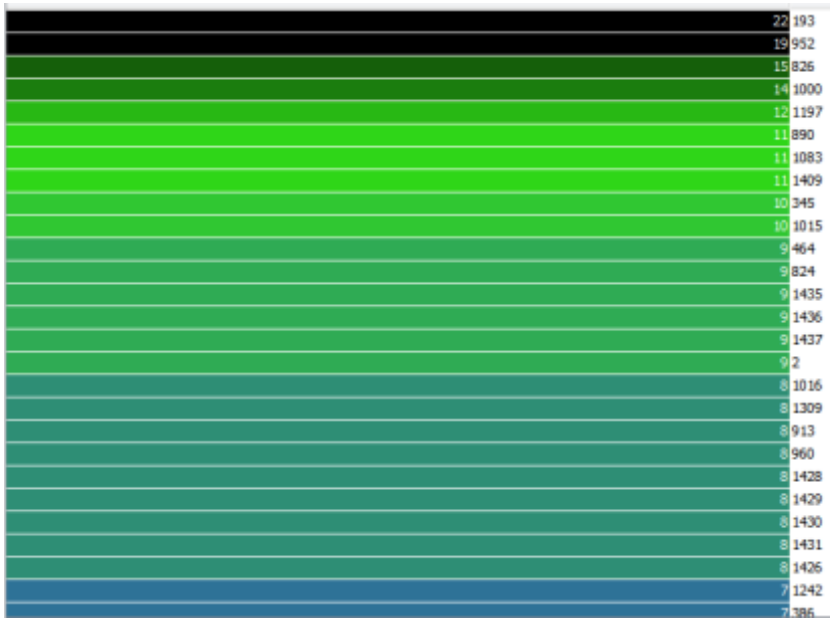


Tabela 7 – Classificação dos nós com mais ligações na rede. Fonte: Arquivo pessoal

Na Tabela 7 é possível visualizar, à direita, o número identificador do criminoso, caso tivessem sido usados nomes nesse campo apareceriam os nomes dos mesmos. E, no retângulo colorido, é possível ver a quantidade de ligações que ele possui com outros nós. Portanto, foi determinado que o criminoso com mais ligações, ou seja, com um Grau maior, é o de ID 193. Este, por sua vez, está conectado com 22 nós. Além disso, ao aplicar essa métrica, é possível selecionar uma série de cores para nós com diferentes Graus, isso facilita a visualização tanto na tabela, quanto no grafo em si, que fica colorido.

O passo seguinte foi aplicar a métrica Comprimento de Caminho Médio da rede. Essa opção calcula o comprimento do caminho para todas as possibilidades de pares de nós e informa o quanto os nós estão próximos uns dos outros. Quando o cálculo termina, ele mostra um relatório. Para este caso, o Comprimento de Caminho Médio foi, aproximadamente, 3,8. Esse valor não possui uma importância visual na rede, ele apenas é necessário para que seja possível se aplicar a métrica Centralidade de *Betweeness.* Pois, após o resultado

dessa métrica, são “liberadas” novas métricas para se aplicar, dentre elas estão Centralidade de *Betweeness* e Centralidade de Proximidade. Neste caso, foi aplicada a Centralidade de *Betweeness* para determinar os nós mais conectados a aglomerados, que são os nós mais influentes da rede. Depois dessas etapas o grafo ficou como é apresentado na Figura 26.

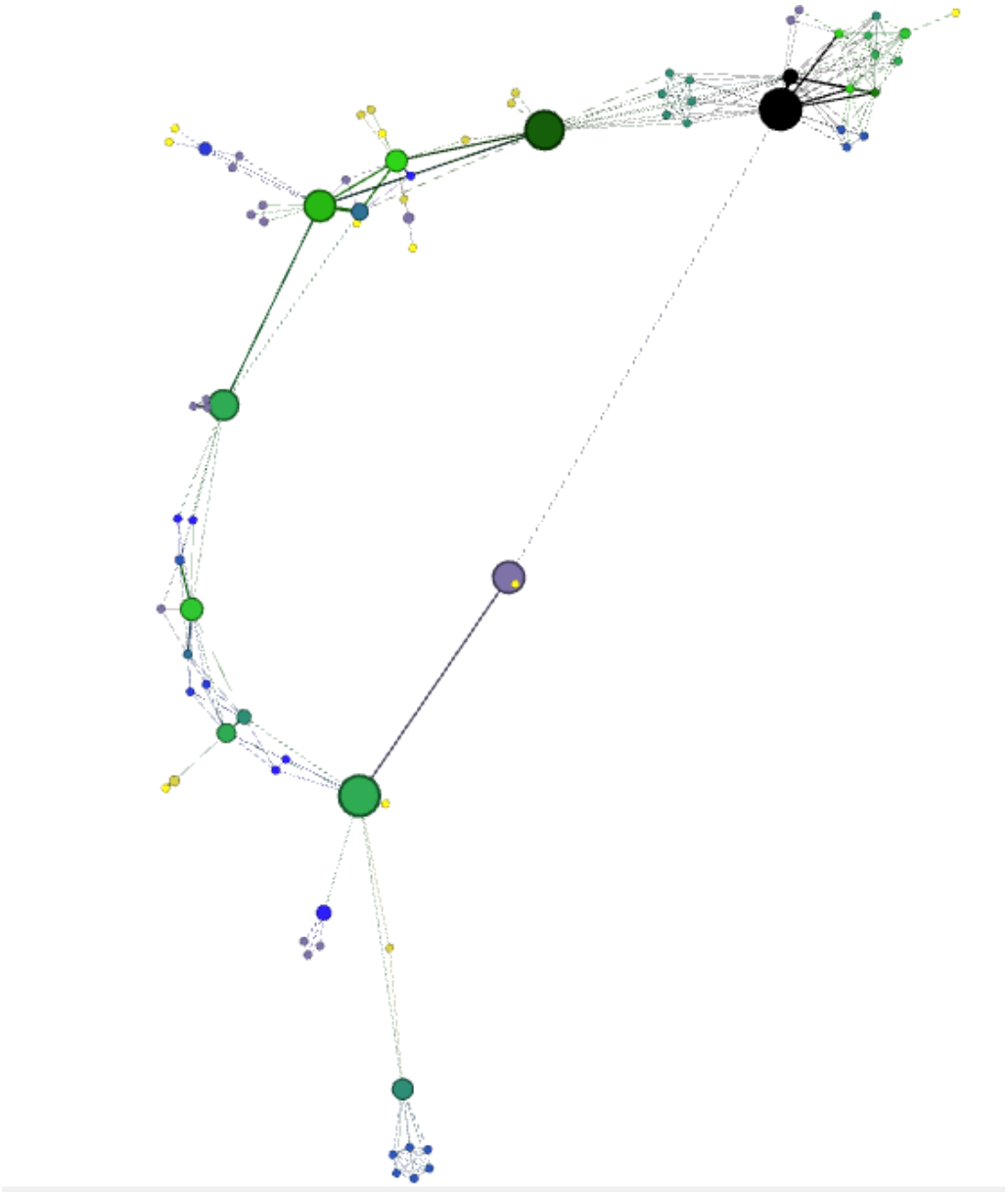


Figura 26 - Grafo após aplicar as métricas Grau e Centralidade de Betweeness.
Fonte: Arquivo pessoal

Na Figura 26, os nodos estão coloridos conforme os valores obtidos pela métrica Grau, e estão com tamanhos diferentes conforme os valores obtidos pela métrica Centralidade de *Betweeness.* O valor relacionado a cada cor pode ser

conferido na Tabela 7. Já o tamanho dos nós é equivalente ao valor obtido na Centralidade de *Betweeness*, e a ferramenta *Gephi* também mostra o valor obtido por essa métrica para cada nó em um relatório.

Na Figura 26 também pode ser visto que alguns nós se sobrepõem aos outros, para que isso não aconteça, o algoritmo *Force Atlas* na *Gephi* possui uma opção de ajuste por tamanho. Após selecionar essa opção, os nós não se sobrepõem mais.

Outra métrica, que também é muito útil da ferramenta *Gephi,* é a Modularidade, ela diferencia os nós mais conectados entre si por cores, destacando, assim, comunidades dentro da rede. Além disso, ela mostra um relatório com a porcentagem de cada comunidade dentro da rede. Na Figura 27 podem ser vistas as porcentagens das 7 comunidades calculadas no trabalho. O número à esquerda representa o número da comunidade designado pela ferramenta e à direita está a sua porcentagem na rede.

Figura 27 - Porcentagem de cada comunidade encontrada na rede. Fonte: Arquivo pessoal

Por fim, um último ajuste para ajudar na visualização da rede, é selecionar a opção para que o *Label* do nó seja mostrado, que, neste caso, é o número identificador do criminoso. Assim, a rede criminal final obtida com base no primeiro conjunto de dados disponibilizada pela BM de Bagé, e em uma primeira implementação, é mostrada na Figura 28.

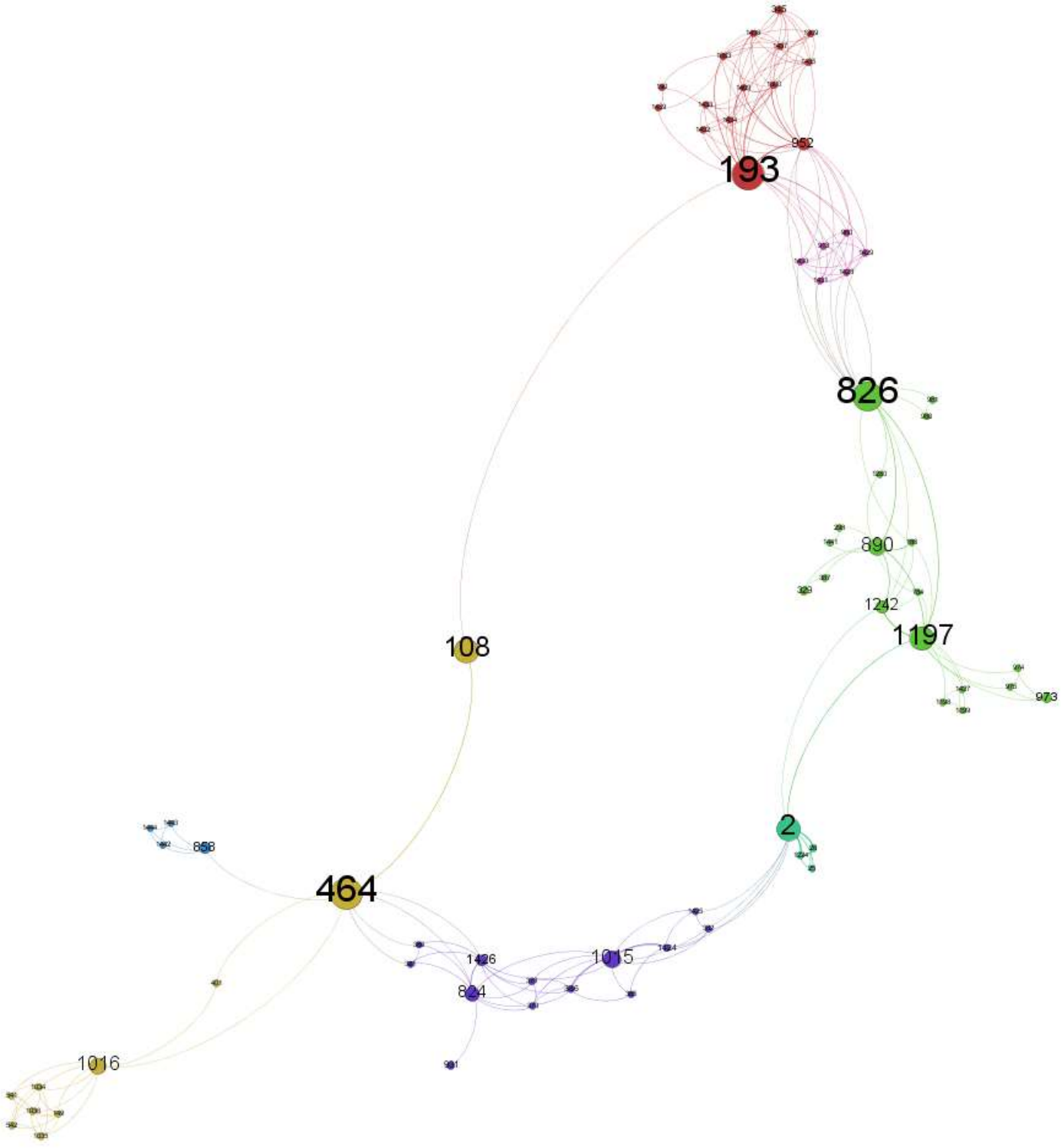


Figura 28 - Rede social final. Fonte: Arquivo pessoal

Como pode ser visto na Figura 28, na rede criminal construída a partir do primeiro conjunto de dados enviados pela BM de Bagé, existem 7 comunidades dentro dessa rede. Onde, dentro de cada comunidade um ou mais indivíduos têm mais influência que os outros. Também pode ser visto dentro desta rede criminal, a partir do tamanho do nó, que os nós mais conectados a aglomerados são os de ID 193, 464 e 826. Porém, isso não significa, necessariamente, que esses são os indivíduos de maior importância dentro da rede criminal. Pois, como pode ser observado, o indivíduo de ID 108 está conectado a apenas dois indivíduos, porém esses indivíduos estão conectados a grandes aglomerados, isto sugere grande influência na rede, também, por parte do ID 108. Podendo-se concluir, a partir disso, que é sempre necessário levar em consideração mais de um tipo de métrica para fazer a análise de uma rede criminal. Em uma interpretação voltada mais para a realidade, porém sem saber mais aprofundadamente a natureza dos dados iniciais, que são sigilosos, poderia-se inferir que os nós maiores representam, por exemplo, um distribuidor de drogas e os nós menores da mesma rede representam os vendedores. Ou, ainda, os indivíduos com nós maiores representam receptadores de uma quadrilha de ladrões de carros, onde os ladrões são representados pelos nós menores.

3.3.2 Segundo conjunto de dados

Os diferentes tipos de algoritmos já citados, que estão contidos na *Gephi,* têm diferentes ênfases. O *OpenOrd* enfatiza as divisões da rede social enquanto que os algoritmos Circular e Eixo Radial têm como principal ênfase um *ranking* dentro da rede. O *GeoLayout*, por sua vez, enfatiza repartições geográficas. Por fim, os algoritmos *ForceAtlas*, *Yifan Hu* e *Fruchterman-Reingold* têm complementaridades como ênfase.

Apesar dos algoritmos presentes na *Gephi* serem de extrema importância para darem formato a rede social e facilitar a visualização da mesma, é necessário que se apliquem as métricas, também presentes na *Gephi*, para fazer a análise da rede. Devido a isso, e às necessidades da Brigada Militar de Bagé na análise de uma rede social criminal, foram selecionadas algumas métricas para serem feitas as análises das diferentes redes sociais criminais montadas

pelos diferentes algoritmos. As métricas selecionadas foram Grau, Comprimento de Caminho Médio, Centralidade de *Betweeness* e Modularidade. A métrica Grau consiste na contagem do número de nós ao qual um nó está conectado, podendo, assim, mostrar quais os indivíduos dentro da rede social criminal com maior número de conexões. Já o Comprimento de Caminho Médio não é uma métrica que tenha um resultado visual na rede, porém, na *Gephi* é necessário calcular-se o Comprimento de Caminho Médio da rede antes de se poder utilizar a métrica Centralidade de *Betweeness.* O Comprimento de Caminho médio calcula a média entre as distâncias entre todos os pares de nós da rede. A métrica Centralidade de *Betweeness* foi escolhida porque, não só é necessário saber o número de conexões de um indivíduo dentro de uma rede, como também é necessário saber a influência desses indivíduos dentro dessa rede. Portanto, essa métrica indica quais são os indivíduos mais influentes dentro da rede social. E, por fim, uma métrica muito importante para a análise da rede social, que foi aplicada, foi a Modularidade. Essa métrica calcula a força das divisões de uma rede, permitindo a detecção de comunidades dentro da rede social.

O segundo conjunto de dados disponibilizado pela BM de Bagé foi inserido na *Gephi* e o primeiro *layout*, antes de qualquer algoritmo ou métrica ser aplicado, pode ser visto na Figura 29. A seguir, será mostrada cada etapa da montagem, modificação do *layout* e análise das redes sociais pelos algoritmos *ForceAtlas*, *OpenOrd*, Circular, Eixo Radial, *Yifan Hu* e *Fruchterman-Reingold*. Além disso, após os algoritmos serem executados, serão aplicadas as métricas Grau, Comprimento de Caminho médio e Centralidade de *Betweeness* em cada uma das redes obtidas.

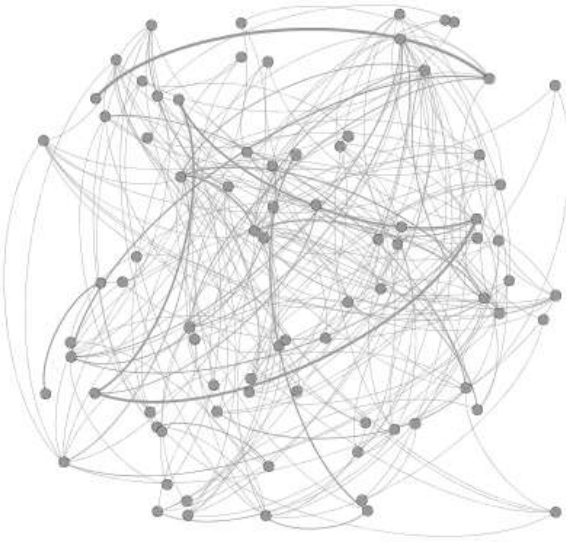


Figura 29 - Layout inicial, sem executar nenhum algoritmo ou métrica. Fonte: Arquivo pessoal

Force Atlas

Após a execução do Force Atlas com configuração padrão no primeiro *layout*, a rede criminal ficou como pode ser vista na Figura 30. Dentre as principais opções de configuração estão as forças de atração, repulsão e gravidade, que foram iniciadas com os valores de 10, 200 e 30, respectivamente. Esses valores foram sendo modificados e a cada modificação o algoritmo foi novamente executado. As mudanças ocorridas no *layout* da rede social devido as modificações nos valores das configurações foram registradas.

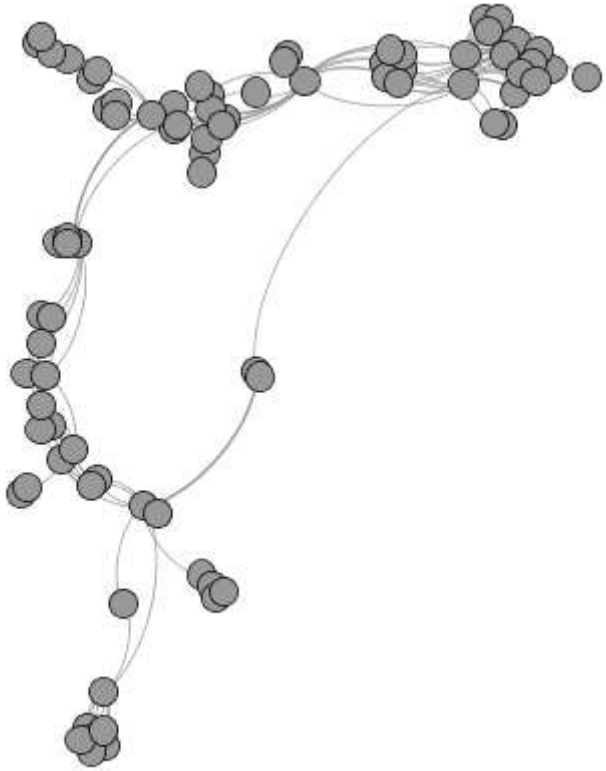


Figura 30 - Layout da rede social após o Force Atlas ser executado com as configurações padrão. Fonte: Arquivo pessoal

Aumentando-se a força de repulsão, que faz os nós se repelirem uns aos outros, de 200 para 1000, o *layout* da rede alterou para o formato que pode ser visto na Figura 31. Nesse *layout,* já é possível ver uma melhoria na configuração da rede, pois o espaçamento entre nós aumentou de forma considerável, facilitando assim a análise da rede.

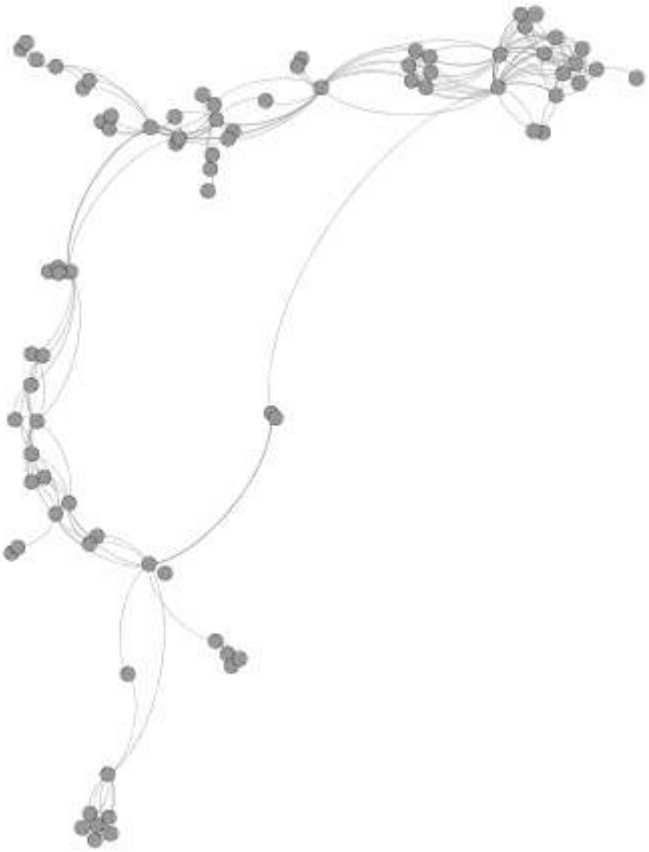


Figura 31 - *Layout* da rede social após a força de repulsão ser aumentada para 1000. Fonte: Arquivo pessoal

Diminuindo a força de atração, que faz com que os nós atraiam uns aos outros, de 10 para 1, o *layout* da rede ficou como pode ser visto na Figura 32. Pode ser observado que a configuração da rede ficou ainda mais aperfeiçoada, os nós continuam posicionados de maneira que seja possível enxergar-se as ligações entre eles, porém, os agrupamentos de nós estão distribuídos de forma mais eficaz, evitando que se sobreponham.

Figura 32 - *Layout* da rede social após a força de atração ser diminuída para 1.
Fonte: Arquivo pessoal.

O próximo passo consistiu em alterar a gravidade de 30 para 100, resultando no *layout* da rede que pode ser observada na Figura 33. A gravidade tem como objetivo atrair todos os nós para o centro para evitar dispersão de componentes desconectados. Pode-se notar nesse *layout* a importância da gravidade, pois, após ser aplicada com mais força, os nós ficaram menos dispersos e a estrutura da rede ficou com uma visualização melhor. Porém, se a gravidade for aplicada com uma força muito grande, os nós podem acabar se aglomerando no centro da rede, dificultando, assim, a visualização.

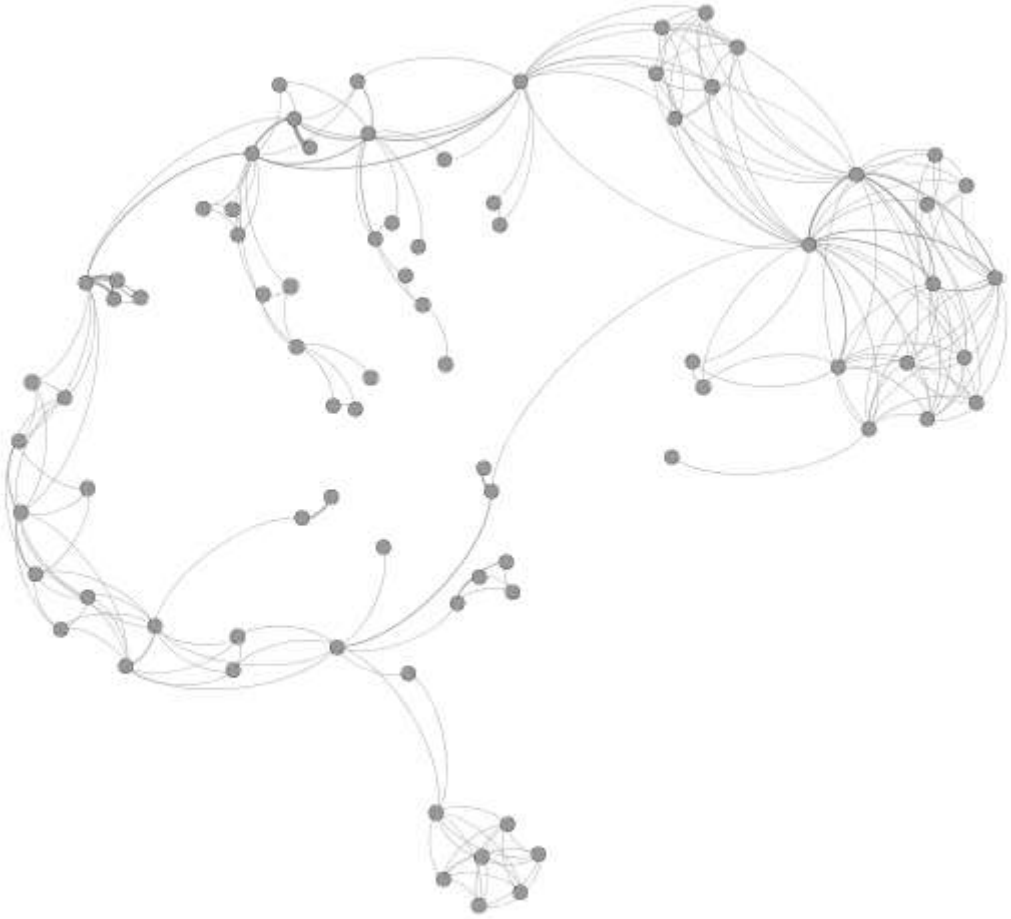


Figura 33 - *Layout* da rede após a gravidade ser aumentada para 100. Fonte: arquivo pessoal.

Uma outra configuração do algoritmo ForceAtlas que pode ser utilizada, também para melhorar o *layout* da rede, é a distribuição de atração. Essa configuração faz com que os nós com maiores números de ligações sejam empurrados para o centro da rede. Essa configuração não pode ser regulada com um valor numérico, pois se trata de uma caixa de seleção que somente pode ser marcada ou desmarcada. O *layout* final, após ser marcada a opção de distribuição de atração, e, apenas com a utilização do algoritmo ForceAtlas, sem a aplicação de nenhuma métrica, pode ser visto na Figura 34.

Figura 34 - Layout final da rede social após ser aplicado o algoritmo *ForceAtlas*.
Fonte: Arquivo pessoal.

Após a finalização do *layout* da rede com a execução do algoritmo ForceAtlas, foi aplicada a métrica Grau. Essa métrica faz uma contagem das ligações de cada nó com outros nós, gerando uma tabela de cores aleatórias e colore os nós conforme a sua quantidade de ligações. A rede social, após ser aplicada a métrica Grau, pode ser vista na Figura 35. E, parte da tabela com a legenda das cores, pode ser vista na Figura 36. A tabela apresenta uma classificação decrescente dos nós com mais ligações, onde o número de ligações do nó é mostrado dentro do retângulo colorido e, ao lado desse retângulo, é mostrada a *label* que identifica o nó. Nesse caso o *label* é mostrado como o número identificador do criminoso para preservar o sigilo, porém, o usual é que se mostre um nome.

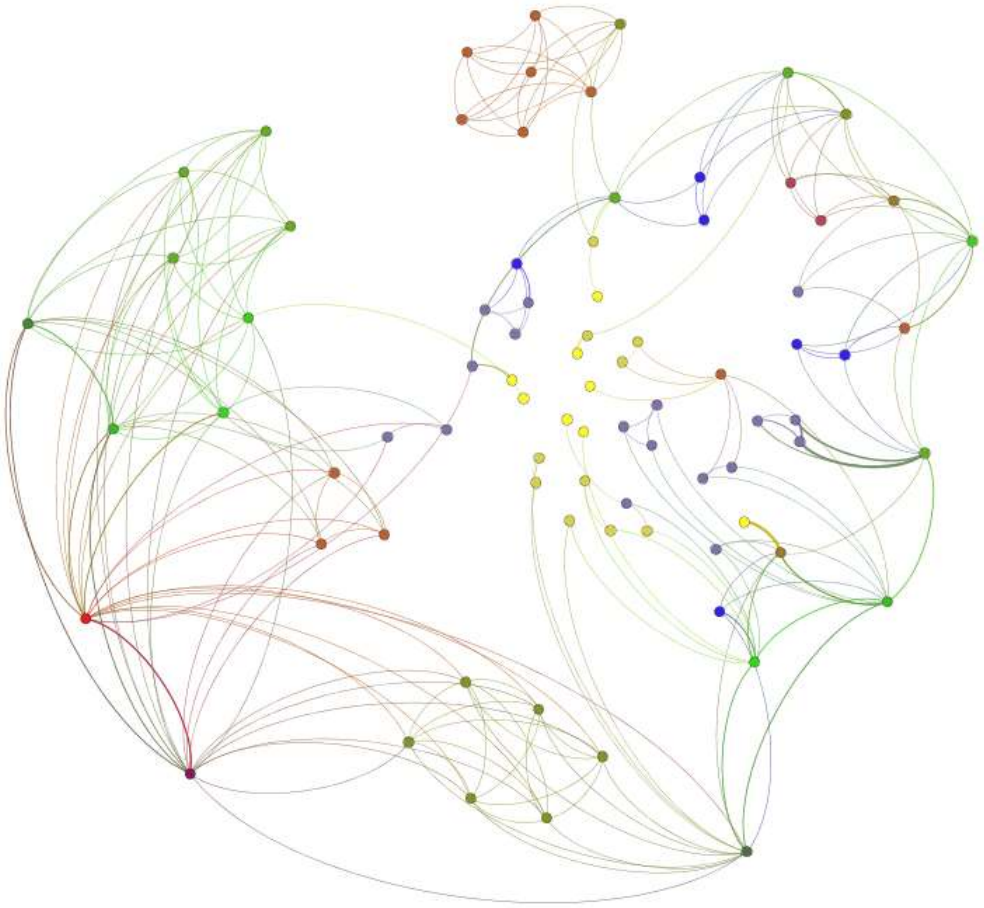
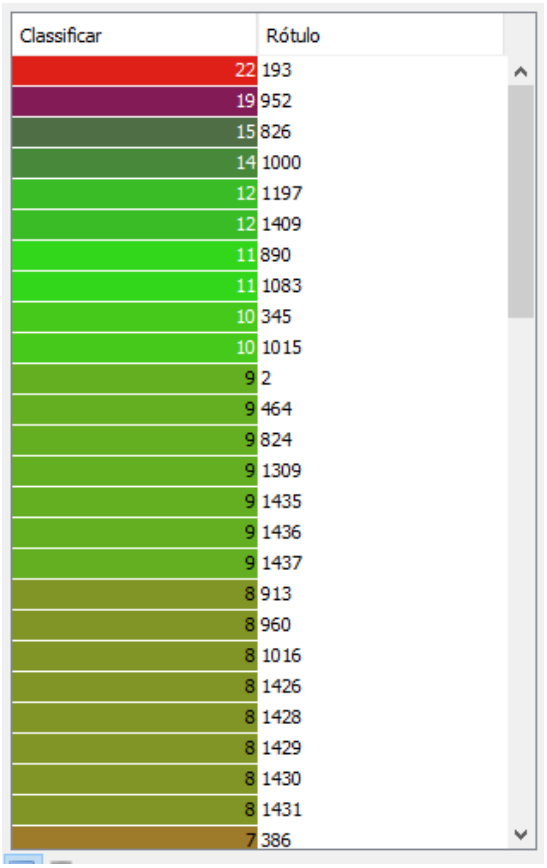


Figura 35 - Rede Social após ser aplicada a métrica Grau. Fonte: Arquivo pessoal

Logo após, foram aplicadas as métricas Comprimento de Caminho médio e Centralidade de *Betweeness*. O tamanho dos nós foi modificado conforme a influência de cada nó na rede calculada pela métrica Centralidade de *Betweeness.* A nova rede, após essas métricas serem aplicadas, pode ser vista na Figura 37. E, após a opção das *labels* dos nós ficarem visíveis, a rede ficou como pode ser vista na Figura 38. Já, na Figura 39, pode ser vista a mesma rede social, porém, colorida pela métrica Modularidade. A Modularidade detecta comunidades dentro da rede e, na rede mostrada na Figura 39, cada comunidade está colorida com uma cor diferente. As cores são determinadas aleatoriamente pelo algoritmo, porém, além de mostrar as comunidades, ele também mostra a porcentagem de cada comunidade dentro da rede, ou seja, o tamanho de cada comunidade dentro da rede. A tabela com essas porcentagens pode ser vista na Figura 40.



Classificar	Rótulo
22	193
19	952
15	826
14	1000
12	1197
12	1409
11	890
11	1083
10	345
10	1015
9	2
9	464
9	824
9	1309
9	1435
9	1436
9	1437
8	913
8	960
8	1016
8	1426
8	1428
8	1429
8	1430
8	1431
7	386

Figura 36 - Tabela da gerada pela métrica Grau. Fonte: Arquivo pessoal.

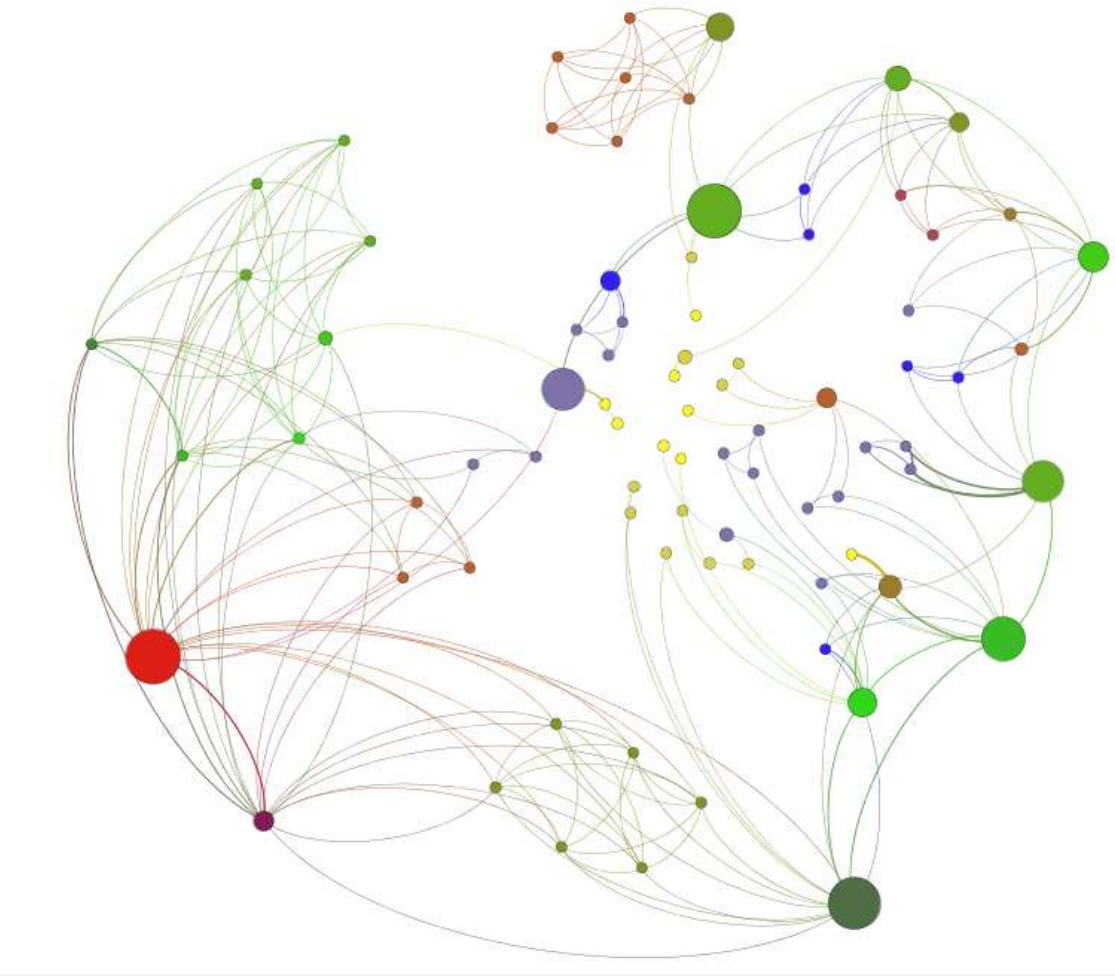


Figura 37 - Rede Social após as métricas Comprimento de Caminho médio e Centralidade de *Betweeness* serem aplicadas. Fonte: Arquivo pessoal.

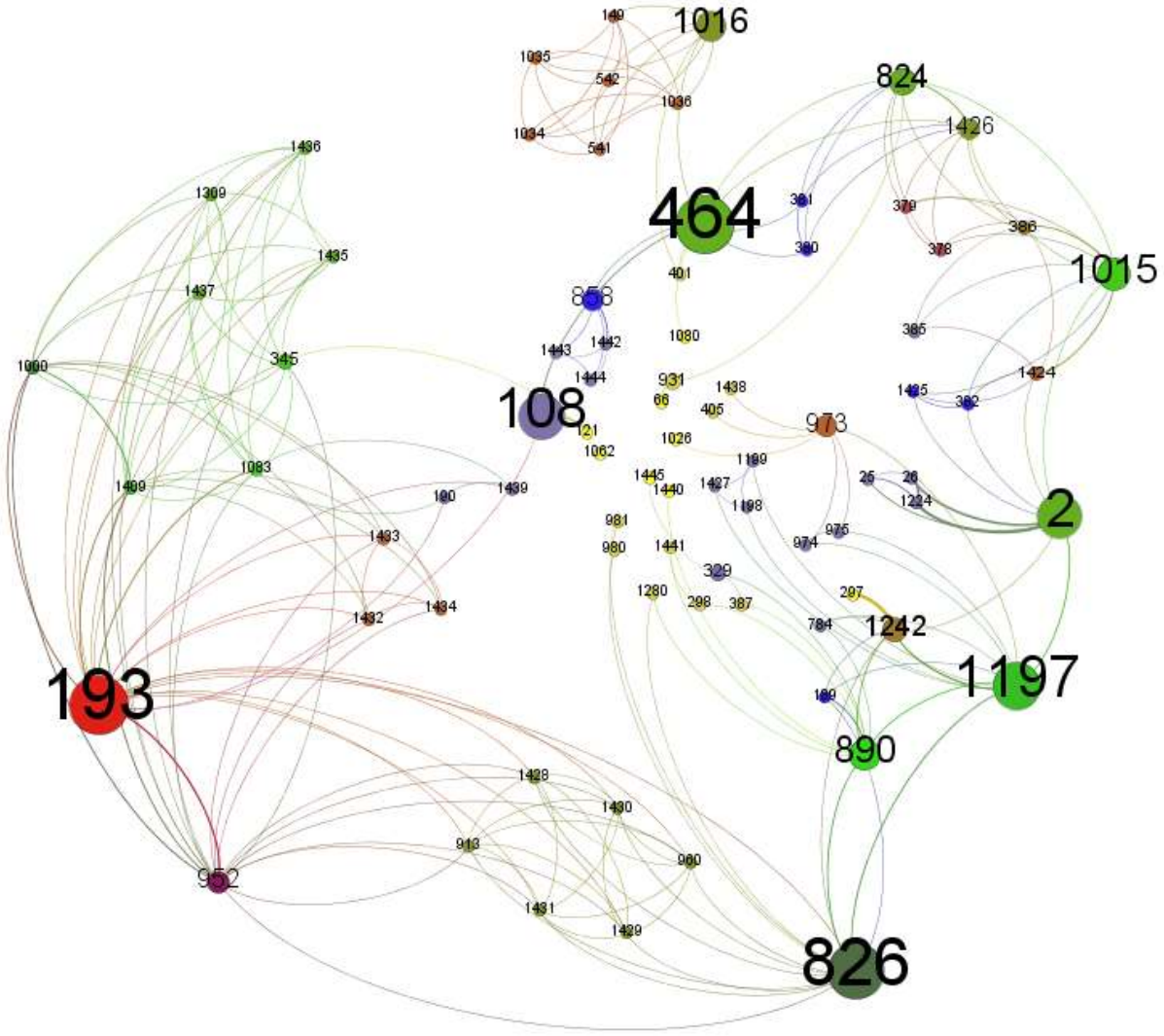


Figura 38 - Rede Social final, utilizando o algoritmo *Force Atlas* e as métricas Grau, Comprimento de Caminho médio e Centralidade de *Betweeness*. Fonte: Arquivo pessoal

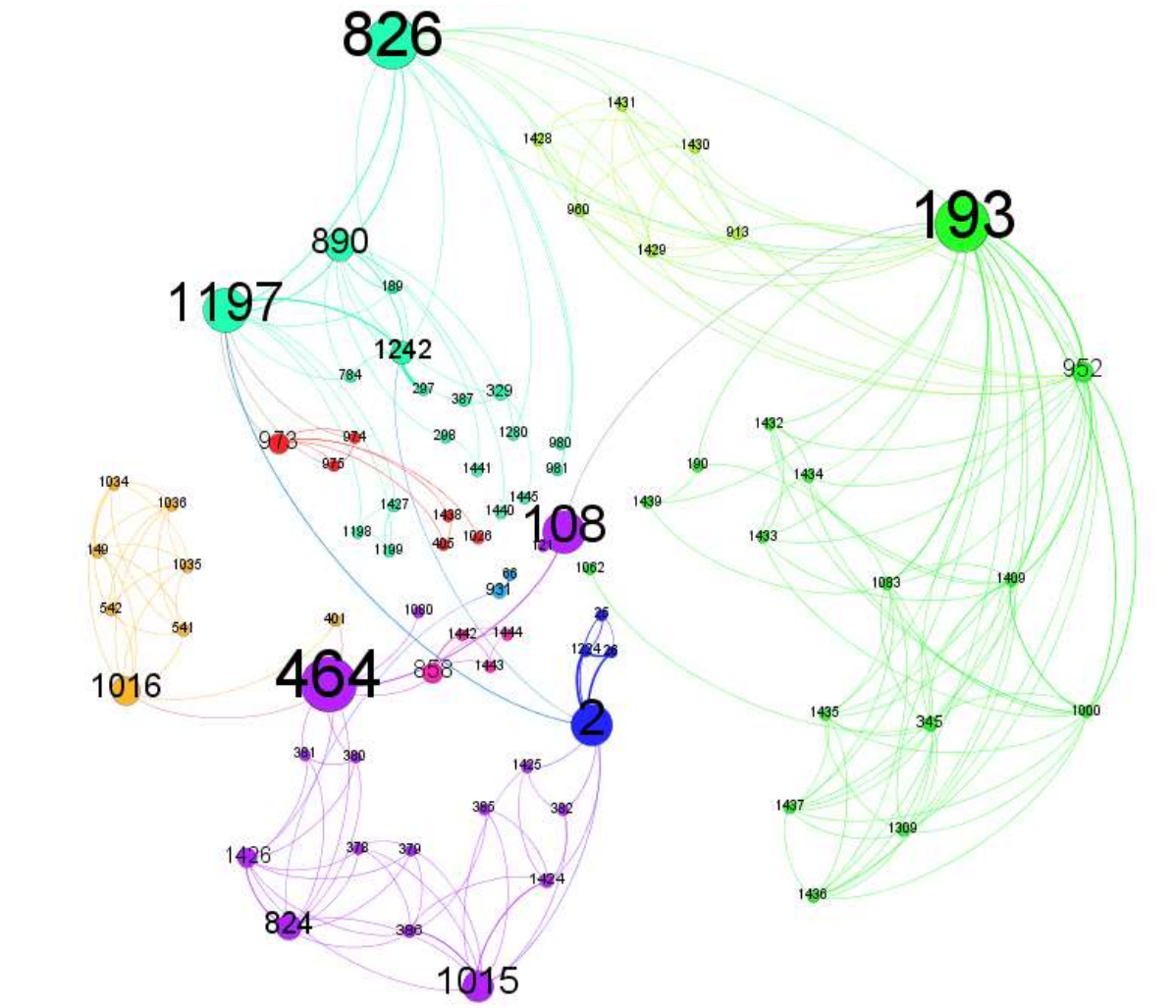


Figura 39 - Rede social gerada pelo algoritmo *ForceAtlas* e colorida pela métrica Modularidade. Fonte: Arquivo pessoal.

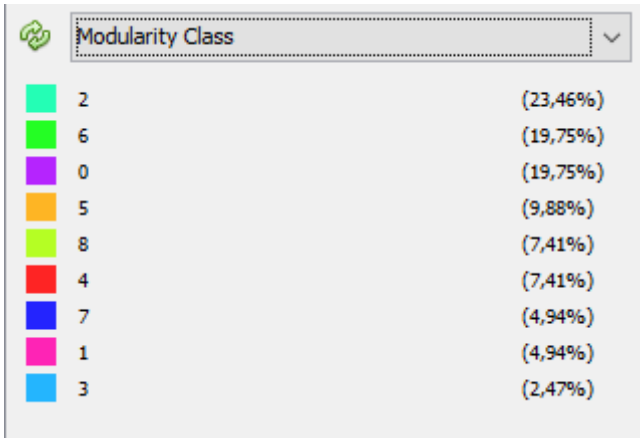


Figura 40 - Tabela com a porcentagem de cada comunidade. Fonte: Arquivo pessoal

OpenOrd

O *OpenOrd* é um algoritmo que tem como ênfase as divisões da rede social. Este algoritmo possui configurações para a determinação do *layout* da rede. Uma delas é o "*Edge Cut",* que pode ser traduzido como o corte da aresta. Essa configuração é um campo onde pode ser digitado um valor de 0 até 1 e representa a porcentagem da maior distância entre dois nodos na rede. Quanto maior o número digitado nesse campo, mais aglomerado será o *layout* da rede. Outra configuração importante que pode ser definida é o número de iterações do algoritmo. Essa configuração não apresenta número mínimo ou máximo, porém, quanto maior for esse número, mais expandidos ficarão os nodos e, quanto menor for o número de iterações, maior a aglomeração dos mesmos.

A partir do *layout* inicial da Figura 29, o algoritmo *OpenOrd* foi executado, utilizando um valor de 0.2 como corte de aresta para aumentar a distância entre os nós. Além disso, foi utilizado um valor de 1000 para a quantidade de iterações com objetivo de expandir os nodos e, assim, facilitar a visualização a rede. Após a execução o algoritmo, foram aplicadas as métricas Grau, Comprimento de Caminho médio e Centralidade de *Betweeness* da mesma maneira que foram aplicadas na rede gerada pelo algoritmo Force Atlas. A rede social final gerada pelo algoritmo *OpenOrd* pode ser vista na Figura 41. Em seguida, foi aplicada a métrica Modularidade. Como os dados iniciais inseridos na *Gephi* foram os mesmos para todos os algoritmos e, cada algoritmo apenas muda o *layout* da rede, as porcentagens das comunidades detectadas pela Modularidade serão sempre as mesmas, que podem ser vistas na Figura 40. Portanto, a rede social final gerada pelo *OpenOrd* e colorida pela Modularidade pode ser vista na Figura 42.

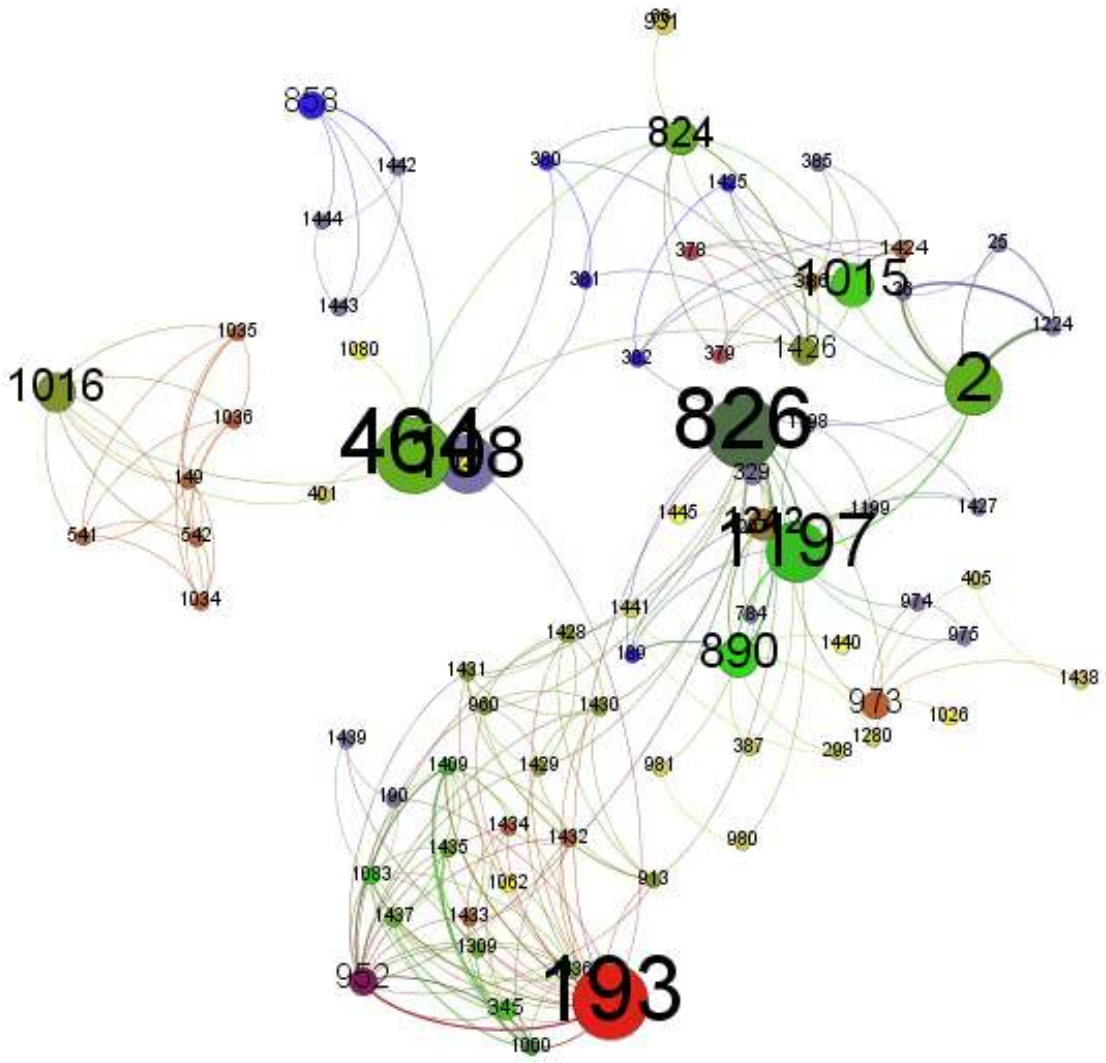


Figura 41 - Rede social final gerada pelo algoritmo *OpenOrd* e colorida por Grau. Fonte: Arquivo pessoal

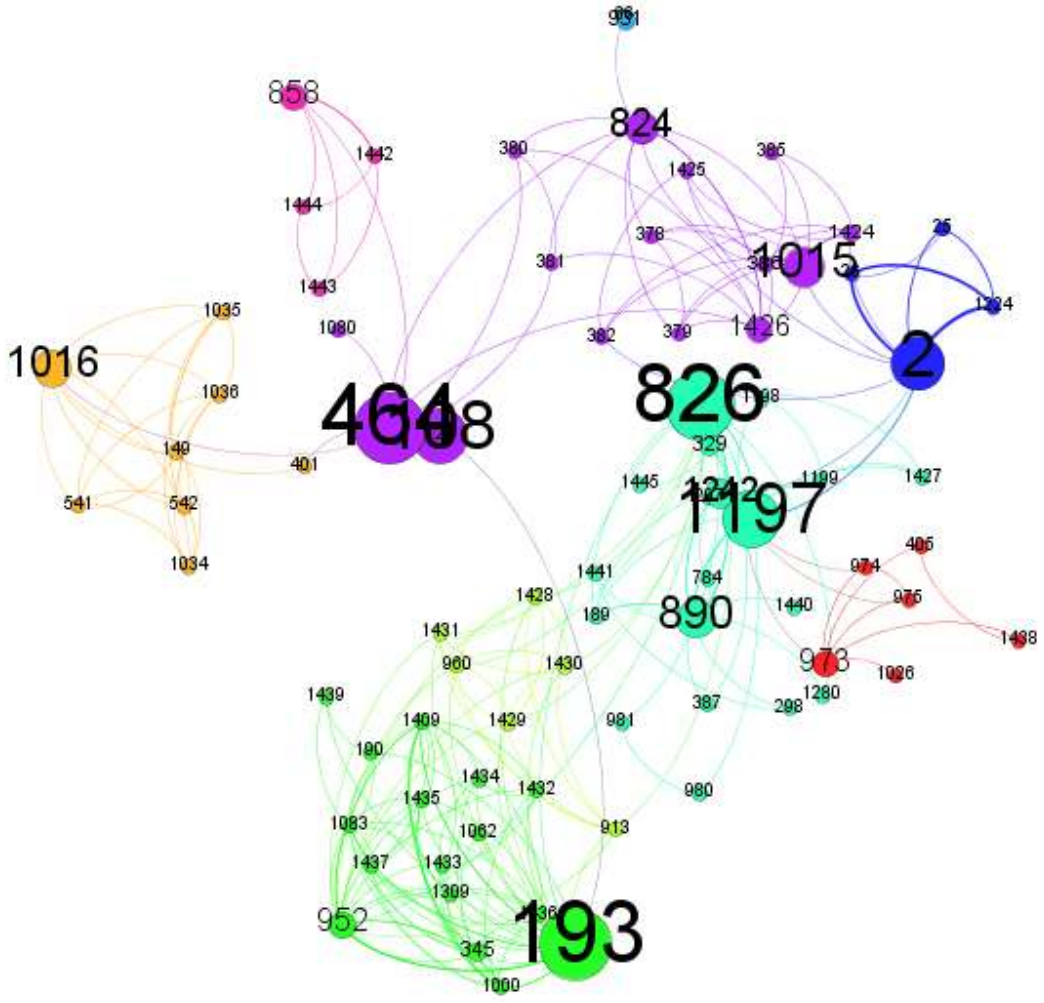


Figura 42 - Rede Social Final gerada pelo algoritmo *OpenOrd* e colorida por Modularidade. Fonte: Arquivo pessoal

Circular

O algoritmo de *layout* Circular tem como principal objetivo fazer uma classificação circular dos nós. Algumas das principais opções de configuração deste algoritmo são a direção da classificação dos nós, que pode ser no sentido horário ou anti-horário, o tamanho do diâmetro do círculo se for escolhida a opção de diâmetro fixo e a métrica que será utilizada na classificação dos nós. Na Figura 43 pode ser observada a rede social gerada com o algoritmo Circular. Os nodos estão classificados no sentido horário pela métrica Grau, assim como, coloridos pela mesma. Já os tamanhos dos nós foram definidos pela métrica

Centralidade de *Betweeness.* O tamanho, as cores e a classificação dos nós podem ser determinadas por qual métrica o usuário achar mais adequada para a análise da sua rede. Na Figura 44 pode ser vista outra rede gerada pelo algoritmo Circular, os nós seguem coloridos pela métrica Grau e com o tamanho determinado pela métrica Centralidade de *Betweeness,* porém eles estão ordenados pelos ID's dos criminosos. Já na Figura 45, pode ser vista a rede após a aplicação da métrica Modularidade, com as comunidades destacadas por cores, porém, com os nodos ainda classificados por Grau.

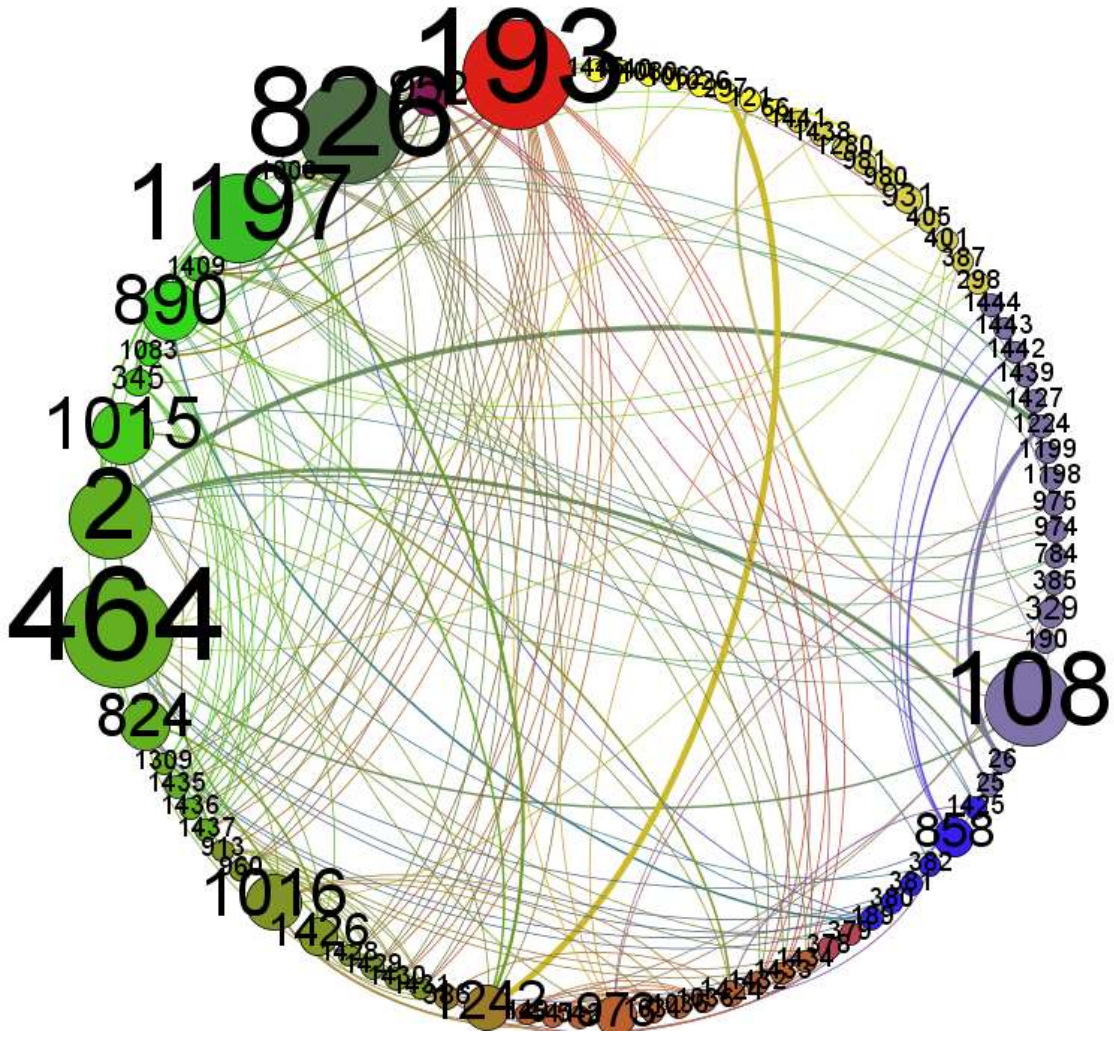


Figura 43 - Rede social gerada pelo algoritmo Circular com os nodos ordenados por Grau. Fonte: Arquivo pessoal

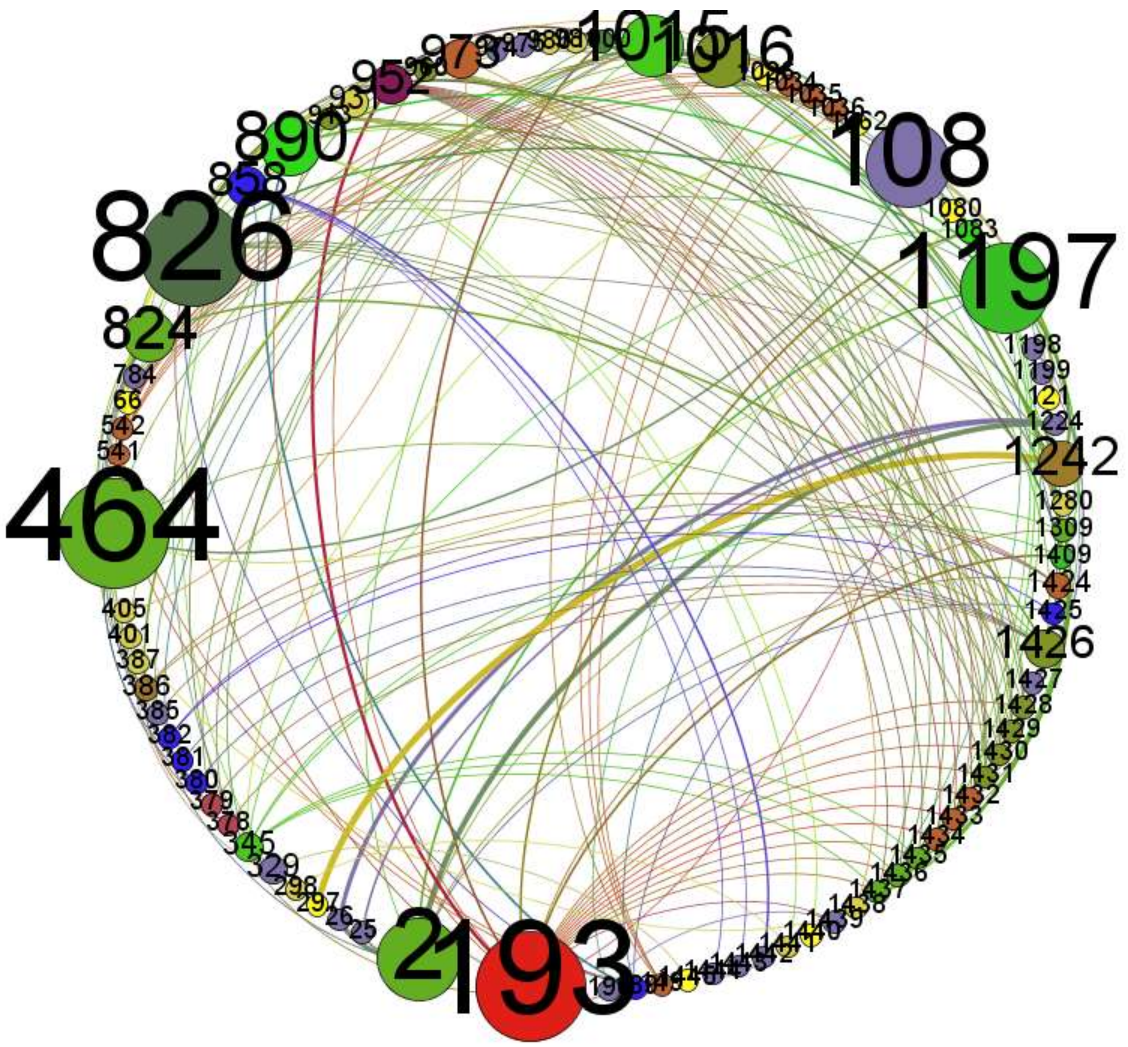


Figura 44 - Rede social gerada pelo algoritmo Circular com os nodos ordenados pelos ID's. Fonte: Arquivo pessoal

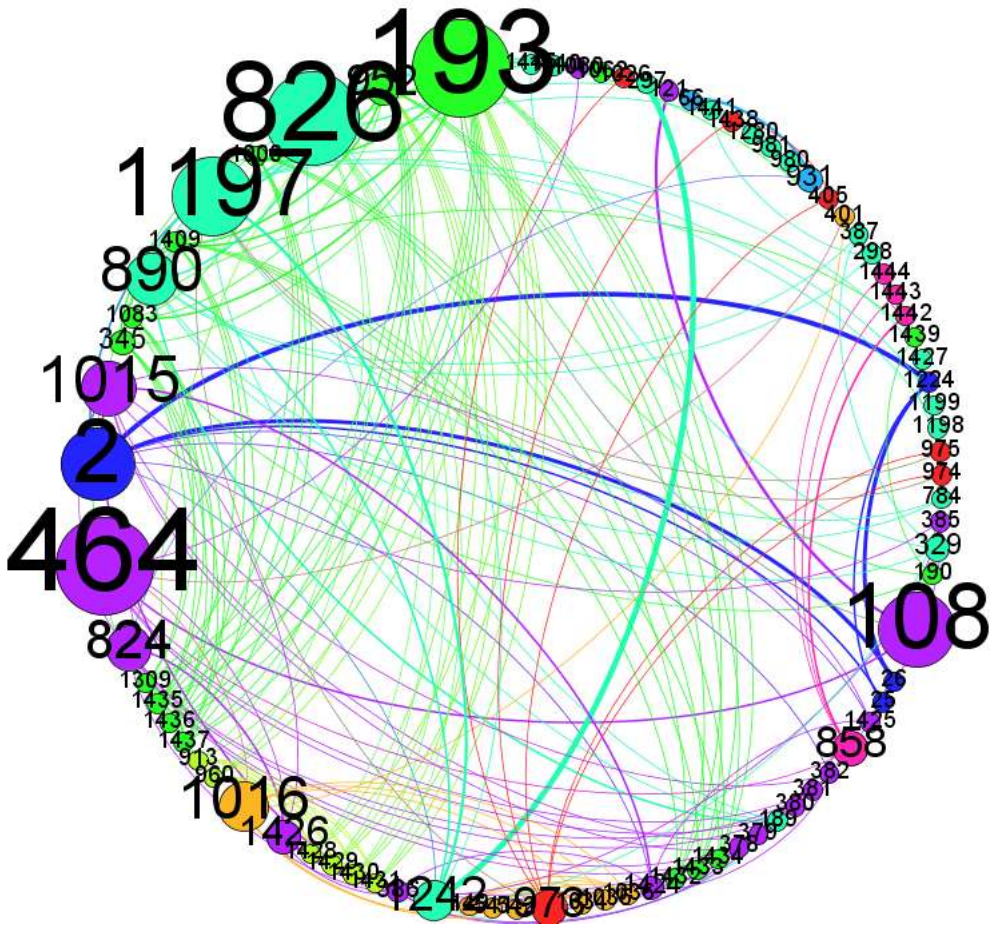


Figura 45 - Rede social gerada pelo algoritmo Circular com os nodos classificados por Grau e coloridos por Modularidade. Fonte: Arquivo pessoal.

Eixo Radial

O algoritmo de Eixo Radial, assim como o Circular, tem como objetivo realizar uma classificação dos nós. Porém, tem um *layout* radial em vez de circular. Algumas das opções de configuração desse algoritmo são a métrica pela qual os nodos serão agrupados, a direção dos nodos no *layout* e a largura da escala. Esse algoritmo foi aplicado no *layout* da Figura 29 e a rede gerada pode ser vista na Figura 46. Os nodos foram agrupados e coloridos por Grau e a direção dos mesmos foi configurada em sentido horário, assim como os tamanhos dos nós foram definidos pela métrica Centralidade de *Betweeness*. A largura da escala foi deixada em 1.2, que foi o padrão gerado pelo algoritmo. Na Figura 47 pode ser observada a mesma rede, só que colorida e com os nós agrupados pela métrica Modularidade.

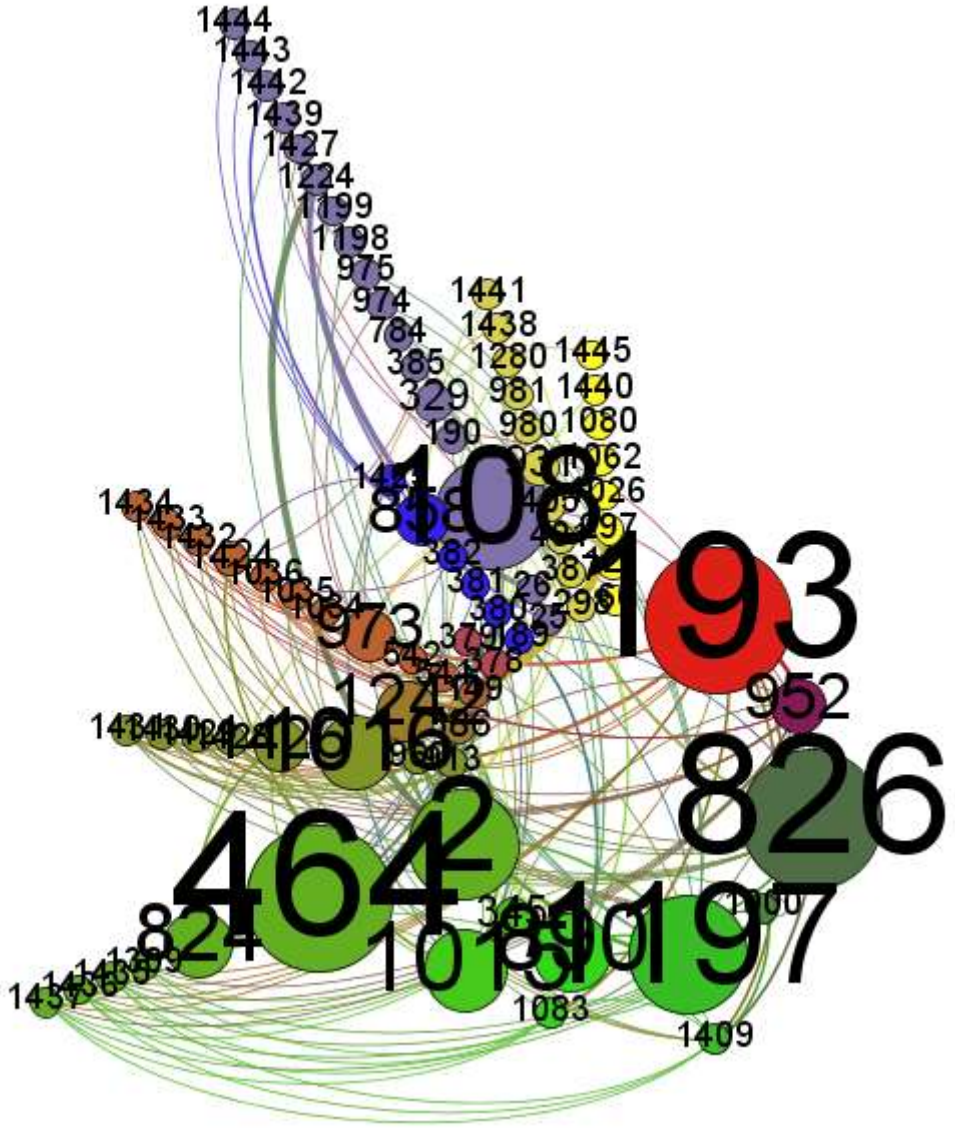


Figura 46 - Rede social gerada pelo algoritmo Eixo Radial com nós coloridos por Grau. Fonte: Arquivo pessoal

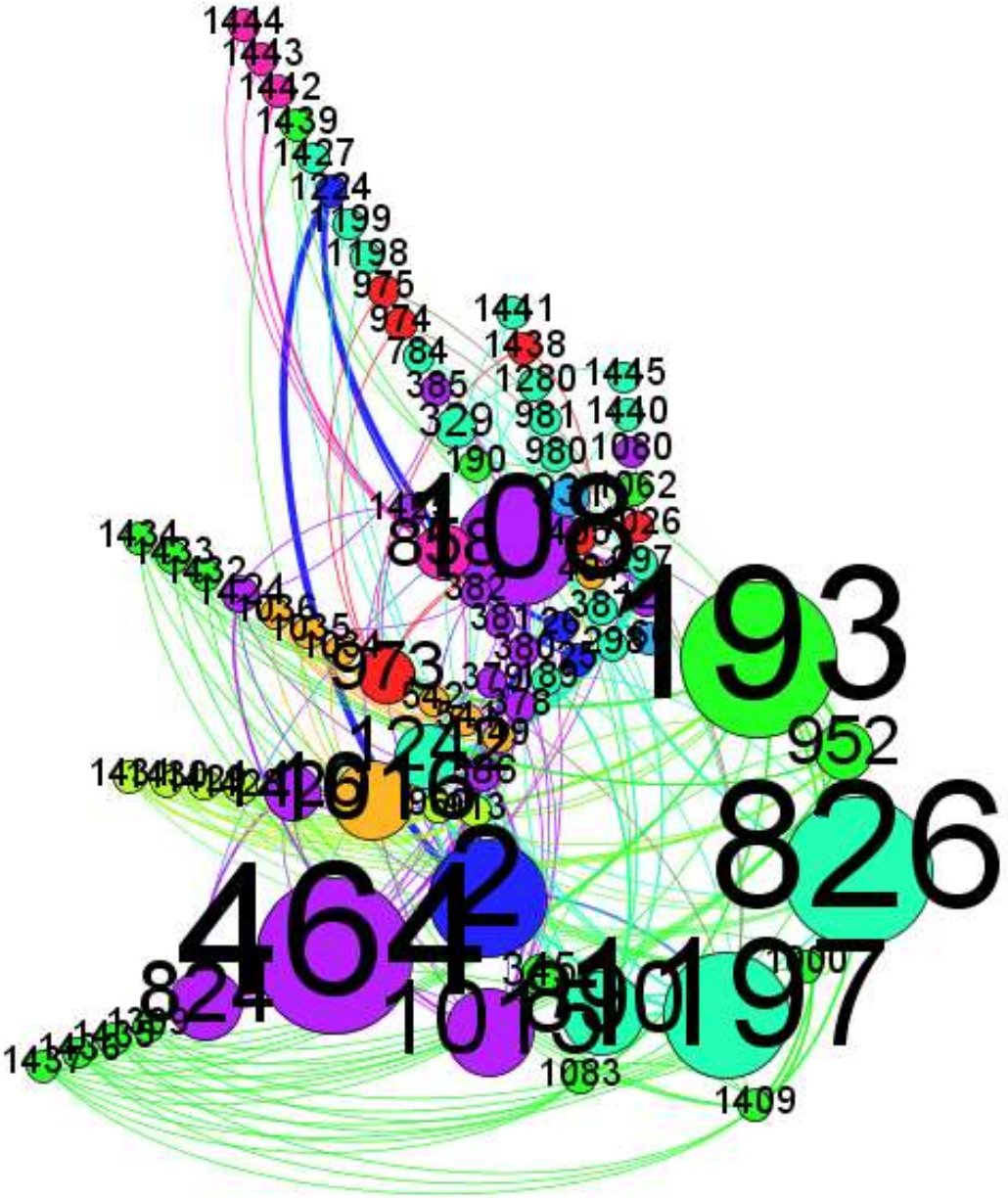


Figura 47 - Rede social gerada pelo algoritmo Eixo Radial e com os nós coloridos e agrupados pela métrica Modularidade. Fonte: Arquivo pessoal

Yifan Hu

O próximo algoritmo que foi executado na rede inicial da Figura 28 foi o *Yifan Hu*. Esse algoritmo possui duas opções de configuração que se destacam que são a relação de passo e a distância ótima. A relação de passo vai de 0 até 1 e representa o raio usado para atualizar o passo do algoritmo. Esse número deve ser aumentado para se conseguir uma qualidade melhor, porém, a velocidade de execução do algoritmo pode ser comprometida. Já, a distância ótima representa o comprimento natural das molas na simulação do sistema de energia, ela deve ser aumentada para que os nodos sejam posicionados mais afastados. A relação de passo utilizada foi de 0.99 para aumentar a qualidade e a distância ótima de 200. Após a execução do algoritmo, as métricas Grau, Comprimento de Caminho Médio e Centralidade de *Betweeness* foram aplicadas. A rede social final gerada pelo algoritmo Yifan Hu pode ser vista na Figura 48. Os nodos estão coloridos pela métrica Grau e o tamanho dos mesmos foi definido pela métrica Centralidade de *Betweeness.* Já na Figura 49, pode ser vista a mesma rede, porém com os nós coloridos pela métrica Modularidade.

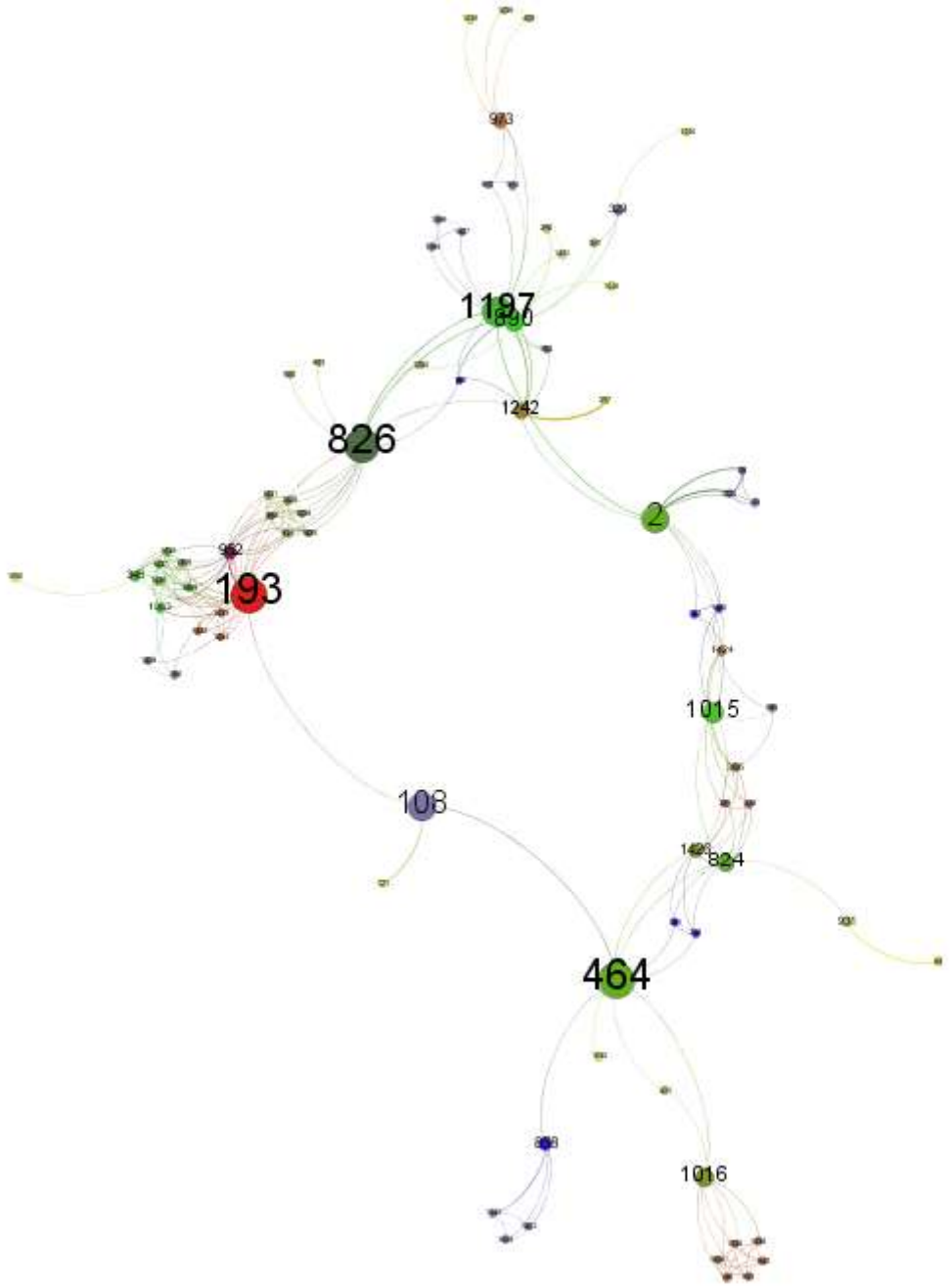


Figura 48 - Rede social final gerada pelo algoritmo *Yifan Hu* e colorida por grau.

Fonte: Arquivo pessoal

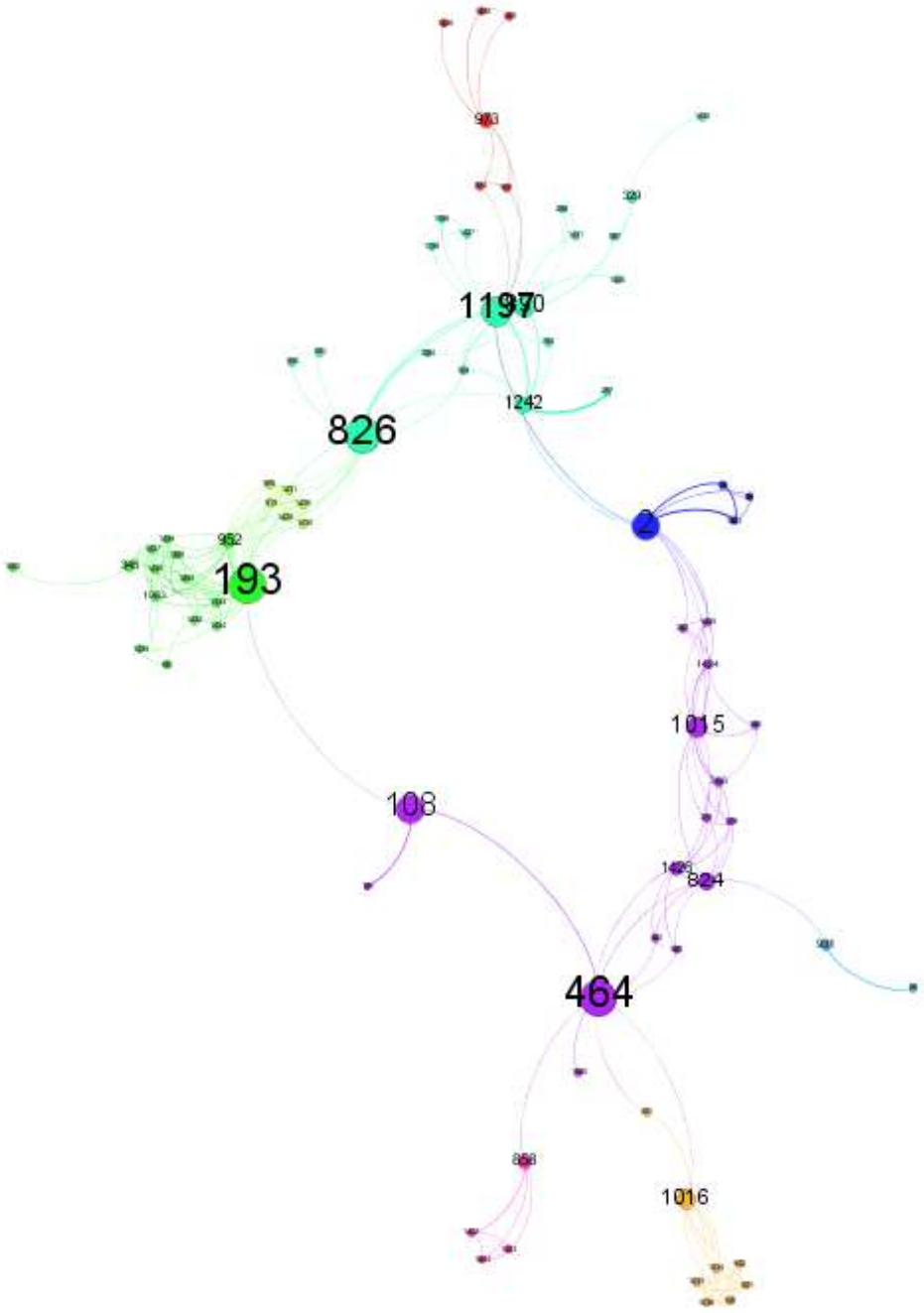


Figura 49 - Rede social gerada pelo algoritmo *Yifan Hu* colorida pela métrica Modularidade. Fonte: Arquivo pessoal

Fruchterman-Reingold

As principais opções de configuração desse algoritmo são a área e a gravidade. A área determina o tamanho da área ocupada pela rede e a gravidade determina o quanto os nodos serão atraídos para o centro da rede, evitando dispersão dos nós. O algoritmo foi executado várias vezes com diferentes valores de área e gravidade, porém, os valores que geraram um *layout* com uma visualização mais agradável, com os nodos mais visíveis e menos aglomerados, foi a área 1000 e a gravidade 100. Após isso, as mesmas métricas aplicadas nas outras redes foram aplicadas nessa rede e, a rede final, colorida pela métrica Grau e com os tamanhos dos nós determinados pela métrica Centralidade de *Betweeness*, pode ser vista na Figura 50. Já na Figura 51, pode ser vista

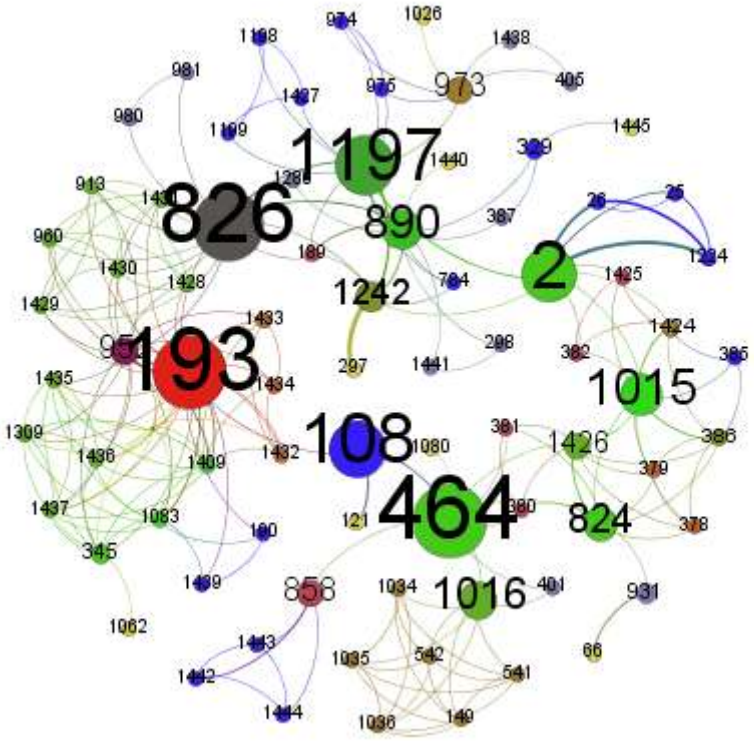


Figura 50 - Rede social gerada pelo algoritmo *Fruchterman-Reingold* colorida pela métrica Grau. Fonte: Arquivo pessoal

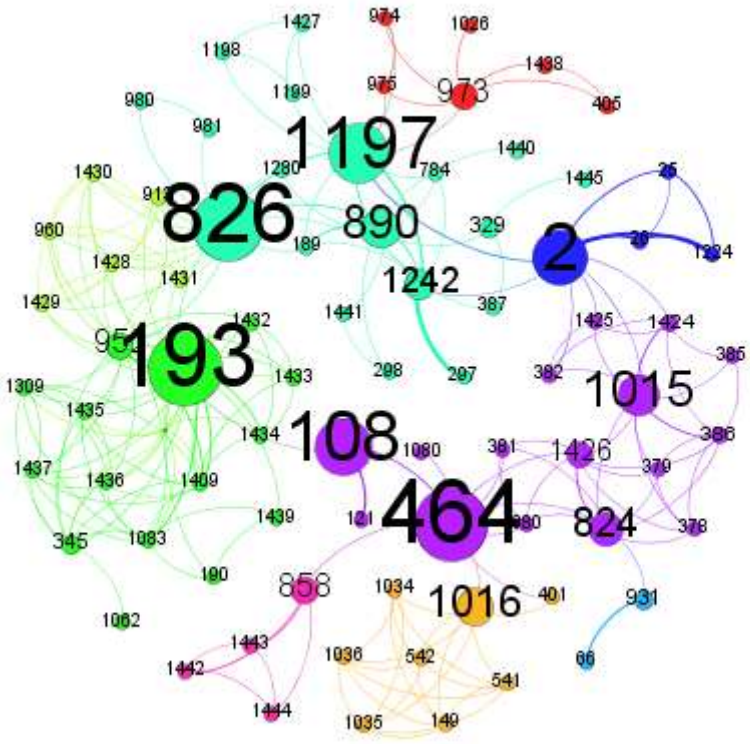


Figura 51 - Rede social gerada pelo algoritmo *Fruchterman-Reingold* com os nodos coloridos pela métrica Modularidade. Fonte: Arquivo pessoal

Geolayout

A partir do conjunto de dados recebido da Brigada Militar, que foi uma tabela no formato *xls* contendo colunas com *id* do delinquente, *id* da ocorrência e endereço, foi possível realizar a montagem de uma rede criminal georreferenciada. Os dados recebidos foram utilizados como entrada na ferramenta desenvolvida e os arquivos de saída foram inseridos na *Gephi*. O *layout* da primeira rede gerada, antes de qualquer manipulação, pode ser visto na Figura 52. Logo após, foi executado na rede o algoritmo *Geolayout*, cuja principal configuração é a escala, na Figura 53 pode ser vista a rede gerada com a escala de 100.00. Como pode ser visualizado nessa figura, os nós ficaram muito próximos uns dos outros, alguns até mesmo se sobrepondo. Devido a isso, foram testadas diversas escalas, até que na escala de 500.000 os nós ficaram com uma boa visualização, sem se sobreporem. Após ser definida a escala, foram aplicadas as métricas Grau e centralidade de *Betweeness*. Na Figura 54,

pode ser vista a rede georreferenciada com escala de 500.000, com os nós coloridos pela métrica Centralidade de *Betweeness* e com os tamanhos dos mesmos destacados pela métrica Grau.

Figura 52- Primeiro *layout* da rede social georreferenciada. Fonte: Arquivo pessoal

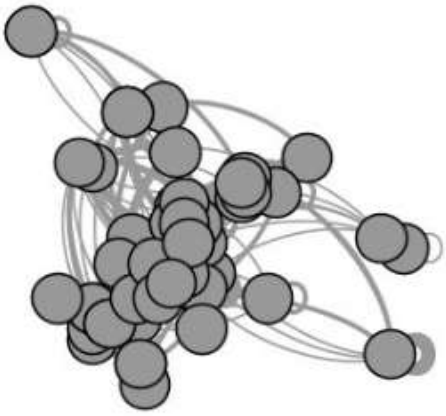


Figura 53 - Rede georreferenciada após aplicar o *Geolayout* com escala de 100.000. Fonte: Arquivo pessoal

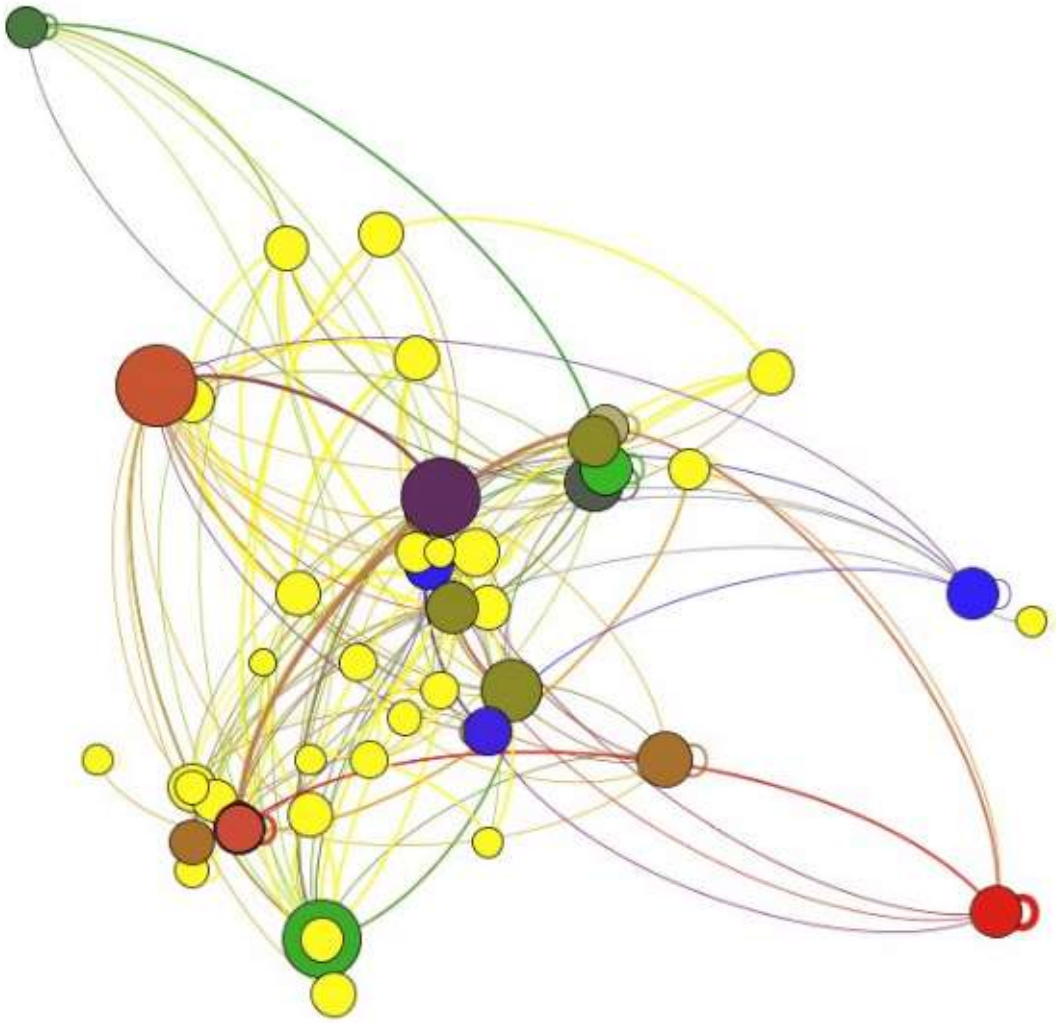


Figura 54 - Rede georreferenciada após o *Geolayout* com escala de 500.000, tamanho por Grau e cor por Centralidade de *Betweeness.* Fonte: Arquivo pessoal

Nas Figuras 52, 53 e 54 é possível de se visualizar que a posição relativa de cada nó com relação aos outros nós não mudou, isso se deve ao fato de o *Geolayout* posicionar os nós de acordo com as suas latitudes e longitudes. Para que seja possível a verificação se, de fato, os nós estão dispostos de acordo com as suas latitudes e longitudes e, além disso, possibilitar uma visualização melhor da rede, o ideal é que se coloque a rede em cima de um mapa. A *Gephi* possui um *plugin* chamado *Map of Countries*[6]*,* que tem como objetivo disponibilizar

[6] https://marketplace.gephi.org/plugin/maps-of-countries/

mapas de regiões para que, combinado com o algoritmo *Geolayout,* seja possível a montagem de redes georreferenciadas (Jacomy, 2011). Porém, até o presente momento, não são disponibilizados nesse *plugin* mapas de todas as regiões do mundo, incluindo Bagé, assim como ainda não é possível de se carregar novos mapas. Devido a isso, a imagem da Figura 54 foi exportada na extensão *png* com um fundo transparente e foi colada em cima do mapa da cidade de Bagé. Pelo fato de o mapa da cidade ser muito grande, quando ele é mostrado em tamanho A4 a rede criminal acaba ficando muito pequena para ser visualizada, em virtude disso, a rede social georreferenciada pode ser vista no *link* *https://www.dropbox.com/s/atr8p3cf58fsv82/Rede%20Georreferenciada.png?dl=0.* Nessa imagem, se pode observar as ocorrências, representadas pelos nós, localizadas em cima de cada lugar em que elas de fato ocorreram, pois, através do arquivo inicial, que continha os endereços, é possível conferir a localização de cada ocorrência. Foi possível notar, também, que a necessidade de se colar a rede em cima de um mapa influenciou um pouco na rede final, pois as ocorrências ficaram um pouco deslocadas do seu lugar exato. Apesar disso, a localização de um nó com relação ao outro continuou correta, apenas houve uma dificuldade de ajuste de escala do mapa com a rede.

Na imagem do *link* também é possível de se ver uma grande quantidade de nós amarelos. Como esses nós foram coloridos pela métrica Centralidade de *Betweeness*, que representa a extensão e o poder social de um grupo de nodos baseado no quão bem eles se conectam. E, além disso, os nós são ocorrências policiais conectadas por criminosos presentes nas mesmas ocorrências, é possível de se interpretar dessa rede que o mesmo grupo de criminosos age nesses locais coloridos em amarelo. Pode-se interpretar também dessa rede, a partir dos nós que estão em tamanho maior, porque foram coloridos pela métrica Grau, que há uma maior incidência de ocorrências criminais nessas áreas. Pois, a métrica Grau contabiliza a quantidade de ligações de um nó com outros nós. Portanto, a partir do desenvolvimento de uma rede criminal georreferenciada, foi possível de se retirar informações dos dados disponibilizados, como locais onde a incidência de ocorrências é maior e locais por onde um mesmo grupo de criminosos atua.

4 CONCLUSÕES

Segundo Schwartz e Rouselle (2008), identificar participantes centrais em redes simples, com relativamente poucos criminosos participantes e poucas conexões, pode frequentemente ser feito “de olho”. Mas, quando redes criminais se tornam mais complexas, envolvendo um grande número de participantes e conexões, SNA pode ser necessária para determinar quais são os criminosos mais importantes.

Baseando-se nisso e no fato da Brigada Militar de Bagé ter cerca de 450 ocorrências registradas por mês e, em média, de 3 a 4 participantes em cada ocorrência, o que gera um grande número de dados, este trabalho foi proposto para que se utilizasse técnicas de SNA a favor da segurança pública. Primeiramente, foram realizas análises do estado da arte de técnicas que poderiam contribuir para o trabalho, como PLN e SNA. A seguir, foram pesquisadas ferramentas de SNA e foi determinado que a ferramenta *Gephi* se adequava às necessidades do trabalho.

Após a análise do estado da arte, foi recebido, em um primeiro momento, um conjunto de dados da BM de Bagé. Esses dados foram manipulados manualmente para que pudessem ser inseridos na ferramenta *Gephi,* o que possibilitou a identificação de uma rede criminal e a aplicação de métricas de SNA. Após a rede criminal ter sido montada, analisada e ter-se chegado a um resultado final (Figura 28), ela foi encaminhada para que os oficiais do setor de Inteligência da BM de Bagé que disponibilizaram os dados fizessem uma análise dos resultados. Eles confirmaram que a rede criminal construída mostra, de fato, as ligações corretas entre os criminosos presentes nos dados encaminhados e que os indivíduos mais influentes na rede também estão devidamente destacados. A partir dessa análise, foi possível desenvolver uma ferramenta que facilitasse e agilizasse a manipulação dos dados para que possam ser utilizados como entrada na ferramenta *Gephi* e, sempre que a Brigada necessitar, possam ser geradas novas redes criminais.

Além disso, foi recebido mais um conjunto de dados para que fosse possível fazer uma análise mais aprofundada sobre as opções de *layouts* de redes sociais disponíveis na *Gephi.* Adicionalmente, foram solicitados nesses dados, os endereços das ocorrências, e, com isso, foi possível montar uma rede

criminal georreferenciada. Essa rede, cujos nodos representam as ocorrências, foi colocada sobre o mapa da cidade de Bagé, possibilitando, assim, uma visão mais clara do que nos dados tabelados sobre onde são as regiões onde o número de ocorrências é maior e mais incidente.

Com relação ao estudo de técnicas de PLN, que foi realizado, porém não foi utilizado neste trabalho. Essas técnicas podem ser utilizadas em trabalhos futuros, pois, foram realizados contatos com a Polícia Civil de Bagé, que demonstrou interesse neste trabalho. Esse órgão possui inquéritos policiais em texto puro digital, onde podem ser aplicados Processamento de Linguagem Natural e Reconhecimento de Entidade Nomeada para retirar os nomes dos criminosos e, a seguir, realizar a identificação de redes criminais.

Por fim, este trabalho teve como resultado o desenvolvimento de uma ferramenta que possibilitará à Brigada Militar de Bagé o processamento automatizado de uma grande quantidade de dados. Assim como, foram feitas análises com dados reais de redes criminais existentes na cidade, comprovando que o método de Análise de Redes Sociais Criminais funciona e que pode continuar sendo utilizado por órgãos de segurança pública para ajudar na prevenção e combate à criminalidade.

4.1 Trabalhos Futuros

Com relação à possibilidade de continuação desse trabalho, foram realizadas reuniões entre o orientador deste trabalho e o delegado da Polícia Civil da cidade de Bagé. Este órgão demonstrou interesse no trabalho realizado e foram disponibilizadas informações de que a Polícia Civil possui inquéritos em formato de texto puro digital. Esses inquéritos são ainda mais completos que ocorrências, pois apresentam, além de nomes de participantes, nomes de suspeitos. Devido a isso, existe a possibilidade deste trabalho ser continuado utilizando-se informações disponibilizadas pela Polícia Civil de Bagé. Essas informações seriam inquéritos em formato de texto puro digital, nos quais poderia se aplicar técnicas de Reconhecimento de Entidades-Nomeadas para se extrair automaticamente nomes de criminosos e suspeitos. Após isso, seria possível identificar novas redes criminais da cidade de Bagé, que possivelmente seriam ainda mais completas que as já identificadas. Além do uso de novas informações

para a identificação de redes sociais, outra abordagem possível para a continuação deste trabalho seria a utilização de técnicas de mineração de dados sobre as redes sociais. Essas técnicas possibilitariam a descoberta de novas informações, como, por exemplo, se os delinquentes costumam cometer crimes perto de onde moram. Portanto, a continuação deste trabalho seria interessante para se contribuir ainda mais com os órgãos de segurança pública.

REFERÊNCIAS

BASTIAN, Mathieu; HEYMANN, Sebastien, JACOMY, Mathieu. Gephi: An Open Source Software for Exploring and Manipulating Networks. **International AAAI Conference on Weblogs and Social Media**, 2009, San Jose, California. Press, 2009. p. 361-362. Disponível em: < https://gephi.org/publications/gephi-bastian-feb09.pdf> Acesso em: 12 abr. 2016.

CAMARA JUNIOR, A. T.. **Processamento de Linguagem Natural para a indexação automática Semântico-Ontológica.** Universidade de Brasília, 2013. Disponível em: <http://repositorio.unb/bitstream/10482/13768/1/2013_AutoTavaresDaCamaraJunior.pdf>. Acesso em: 05 out. 2015.

CHRISTIAN, Tominski; JAMES, Abello; HEIDRUN, Schumann. **Axes-based visualizations with radial layout.** Nicosia, Cypros. Disponível em: <http://www.mgvis.com/Papers/Visualization/Greece_MMV-06.pdf>. Acesso em: 13 abr. 2016.

HEYMANN, Sebástian; LE GRAND, Bénédicte. Visual Analysis of Complex Networks for Business Intelligence with Gephi. **International Conference on Information Visualisation.** 2013.

HU, Yifan. Efficient and high quality force-directed graph drawing. **The Mathematica Journal.** 2005. Disponível em: <http://yifanhu.net/PUB/graph_draw_small.pdf> Acesso em: 14 abr. 2016.

JACOMY, Mathieu. **Gephi Tutorial Layouts.** Disponível em: <https://gephi.org/tutorials/gephi-tutorial-layouts.pdf>. Acesso em: 12 abr. 2016.

JACOMY, Mathieu; *et al.* ForceAtlas2, a Continuous Graph Layout Algorithm for Handy Network Visualization Designed for the Gephi Software. **PLoS ONE**, 2014. Disponível em: <http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0098679>. Acesso em: 12 abr. 2016.

JURAFSKY, Daniel; MARTIN, James. **Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition.** 2 ed. New Jersey: Pearson, 2008.

KOSCHADE, Stuart. A Social Network Analysis of Jemaah Islamiyah: The Applications to Counter-Terrorism and Intelligence. **Studies in Conflict and Terrorism,** 2006**.** Vol. 29(6):pp. 559-575.

LAVRAC, Nada; *et al.* Exploratory Analysis of the Social Network of Researchers in Inductive Logic Programming. **Computer Communications and Networks.** Springer: 2010.

MARRERO, Mônica; *et al*. Evaluation of Named Entity Extraction Systems. **Advances in Computational Linguistics.** Research in Computing Science 41, 2009, p. 47-58.

MARRERO, Mônica; *et al.* Named Entity Recognition: Fallacies, challenges and opportunities. **Computer Standards and Interfaces.** Elsevier, out. 2012.

MIEGHEM, P. **Graph eigenvectors, fundamental weights and centrality metrics for nodes in networks**. Cornell University Library, 2014. Disponível em: < http://arxiv.org/abs/1401.4580>. Acesso em: 23 jun. 2015.

MORSELLI, Carlo. Assessing Vulnerable and Strategic Positions in a Criminal Network. **Journal of Contemporary Criminal Justice.** Sage, set. 2010.

NADEAU, D; SEKINE, S. **A survey of named entity recognition and classification.** National Research Council Canada and New York University, 2009. Disponível em: <http://brown.cl.uniheidelberg.de/~sourjiko/NER_Literatur/survey.pdf>. Acesso em: 15 out. 2015.

NUNES, M. **O Processamento de Línguas Naturais: para quê e para quem?** Instituto de Ciências Matemáticas e de Computação, 2008. Disponível em: <http://wiki.icmc.usp.br/images/1/10/Nunes2008.pdf>. Acesso em: 09 out. 2015.

ROSTAMI, Amir; MONDANI, Hernan. **The Complexity of Crime Network data: A Case Study of Its Consequences for Crime Control and the Study of Networks.** Plos One, 2015. Disponível em: <http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0119309>. Acesso em: 02 abr. 2015.

RESSLER, Steve. “Social Network Analysis as an Approach to Combat Terrorism: Past, Present, and Future Research.” **Homeland Security Affairs *2***, Article 8 (July 2006). Disponível em: <https://www.hsaj.org/articles/171>. Acesso em: 22 jun. 2016.

SCHWARTZ, Daniel; ROUSELLE, Tony. Using social network analysis to target criminal networks. **Trends in Organized Crime.** Springer, out. 2008.

SCOTT, John. **Social Network Analysis: A Handbook.** 2 ed. Torquay: Sage, 2000.

SCOTT, John. Social network analysis: developments, advances, and prospects. **Social Network Analysis and Mining.** v. 1, p. 21-26. Springer, out. 2010.

SEIDLER, Patrick; ADDERLEY, Rick. Criminal Network Analysis inside law enforcement agencies: a data-mining system approach under the National Intelligence Model. **International Journal of Police Science and Management**, Worcestershire, v. 15, n. 4, p. 323-337, out/dez. 2013.

SILVEIRA, Matheus. **Named Entity Recognition.** Universidade de Évora, 2014. Disponível em: <http://www.researchgate.net/publication/264129652_Named_Entity_Recognition> Acesso em: 20 jun. 2015.

Social Network Analysis: Theory and Applications, 2015. Disponível em: < http://train.ed.psu.edu/WFED-543/SocNet_TheoryApp.pdf>. Acesso em: 25 out. 2015.

XU, Jennifer; *et al*. Analyzing and Visualizing Criminal Network Dynamics: A Case Study. **Intelligence and Security Informatics: Second Symposium on Intelligence and Security Informatics**, Tucson, p. 359-377, jun. 2004. Disponível em: <http://img1.wikia.nocookie.net/__cb20140306215732/vroniplag/de/images/0/01/Nm-20140306.pdf> Acesso em: 28 abr. 2016.