

Quem faz a diferença? Uma abordagem para análise da comunicação entre os *stakeholders* de um projeto de software

Luiz Eugênio Coelho Neto ¹, Gláucia Braga e Silva ¹

¹ Universidade Federal de Viçosa
Rodovia LMG 818, km 06 - Florestal-MG

{glaucia, luiz.eugenio}@ufv.br

Resumo. *Este artigo apresenta uma abordagem para criação de uma rede de comunicação entre os envolvidos em um projeto de software (stakeholders). A rede é gerada a partir dos dados de comunicação presentes em ambientes de issue tracking e repositórios de software, mesclando três abordagens disponíveis na literatura: relevância temática, análise de sentimento e análise de rede social. O intuito da rede é possibilitar a identificação de stakeholders chave no processo de comunicação, a partir do cálculo da centralidade dos nós. A fim de automatizar a aplicação da abordagem, a ferramenta ColMiner 2.0 foi desenvolvida e será apresentada. Com a ferramenta, o usuário poderá extrair dados do GitHub, gerar a rede e personalizar os medições presentes nos nós e arestas. Por fim, uma prova de conceito será apresentada como uma forma de avaliar a abordagem proposta, relacionando os stakeholders mais centrais da rede com aqueles que mais realizaram commits de código.*

Palavras-Chave. *Análise da comunicação; Análise de rede Social; Relevância Temática; Sentimento; controle de versão; issue tracking.*

1. Introdução

Considerando-se a natureza colaborativa do desenvolvimento de software, onde os participantes trabalham em conjunto para alcançar o sucesso do projeto, a comunicação tem sido apontada na literatura como um dos fatores críticos a serem considerados [Fan 2010, Han and Jung 2014, da Trindade et al. 2008]. A troca de informações é importante para que os envolvidos no projeto (*stakeholders*) compreendam o trabalho a ser feito e desenvolvam soluções apropriadas. Dessa forma, o gerenciamento das comunicações é umas das áreas-chave recomendadas pelo Project Management Body of Knowledge (PM-BOK), pois é responsável por assegurar que as informações do projeto sejam coletadas, distribuídas e organizadas de maneira oportuna e apropriada [PMI 2013]. Um dos processos envolvidos nesse gerenciamento abrange controlar e monitorar as comunicações de modo a garantir que as necessidades de informação sejam atendidas.

No contexto de desenvolvimento de software, a comunicação ocorre a medida que as ferramentas de *issue tracking* e de controle de versão são usadas. Em ferramentas como

essas, registra-se um conjunto valioso de informações que podem ser analisadas. Segundo Bertran et al. [Bertram et al. 2010], repositórios de *issue tracking* são repositórios de conhecimento, que concentram grande parte da comunicação e da colaboração dos projetos, além de serem um canal de comunicação em si. Por essa razão, analisar seu conteúdo pode gerar informações importantes sobre a comunicação, apoiar o monitoramento e prevenir falhas.

Dentre as diversas análises que podem ser aplicadas sobre tais repositórios de software, tem-se a análise de sentimento e da relevância temática presente nos comentários postados pelos *stakeholders*. A utilidade dessas análises são comprovadas em diversos trabalhos na literatura [Ortu et al. 2015, Yang et al. 2017, Yu et al. 2016]. Dentre os estudos, alguns apresentam o impacto do sentimento em outros fatores no processo de desenvolvimento de software como o tempo de resolução dos *issues* [Ortu et al. 2015, Yang et al. 2017], outros identificam *stakeholders* que escrevem comentários mais relevantes [Eugênio Coelho Neto and Braga e Silva 2018].

Outra forma de análise a ser aplicada sobre repositórios consiste na construção de redes com as relações entre os *stakeholders* e a aplicação de abordagens para a análise de rede social, como cálculo dos nós centrais. No ambiente de *issue tracking* cada nó da rede pode representar um *stakeholder* e uma relação seria formada quando um *stakeholder* comenta uma *issue* reportada por outro, por exemplo [Yu et al. 2016]. As relações também podem ser qualificadas através de pesos calculados a partir da relevância e/ou sentimento dos comentários. Um nó central em uma rede como essa poderia indicar um *stakeholder* que participou ativamente do projeto como comunicador.

Este artigo apresenta uma abordagem que utiliza análise de rede sociais com o intuito de descobrir relações entre os *stakeholders* e identificar quais são os principais no âmbito da comunicação. Essa análise será integrada com outras métricas de análise da comunicação, usadas para atribuir pesos às relações, tais como: o sentimento dos comentários, a relevância temática e a quantidade de comentários ponderada pelo tempo. A fim de automatizar o uso da abordagem, este trabalho apresentará ainda uma nova versão da ferramenta ColMiner, desenvolvida no contexto de outro trabalho [Eugênio Coelho Neto and Braga e Silva 2018], cujo foco era analisar a relevância dos comentários postados em um ambiente de *issue tracking*. O ColMiner 2.0 permitirá que o usuário visualize a rede por meio de grafos e escolha métricas a serem utilizadas como peso das relações e algoritmos para calcular a centralidade dos nós.

A principal contribuição é a integração de três abordagens, cujos trabalhos apresentam resultados interessantes para apoiar o gerenciamento das comunicações em projetos de software. Não menos importante é a criação de uma nova versão da ferramenta ColMiner, como forma de oferecer uma análise mais completa e potencializar os resultados obtidos. A abordagem proposta e o ColMiner 2.0 visam auxiliar no monitoramento da comunicação em projetos, identificando os *stakeholders* que fazem a diferença por participarem ativamente da comunicação e serem contribuidores importantes para o projeto.

Este trabalho será organizado da seguinte forma: primeiramente os termos importantes para compreensão do trabalho serão definidos na Seção 2; os estudos que serviram de base para desenvolvimento serão descritos na Seção 3; a metodologia de pesquisa uti-

lizada durante desenvolvimento é apresentada na Seção 4; a formalização da abordagem para criação da rede de comunicação é descrita na Seção 5; a Seção 6 detalha a nova versão e arquitetura da ferramenta ColMiner; uma prova de conceito aplicando abordagem é descrita na Seção 7; por fim, a conclusão e trabalhos futuros são apresentados na Seção 8.

2. Referencial Teórico

Com o intuito de facilitar a compreensão, nesta seção serão melhor definidos e discutidos termos frequentes de cada uma das formas de análise que serão integradas neste trabalho.

2.1. Análise de sentimentos

A análise de sentimentos tem sido abordada na literatura como forma de investigar o conteúdo textual de uma mensagem com intuito de descobrir o seu peso sentimental (negativo ou positivo)[Yang et al. 2017]. Se o mesmo fala sobre algo de forma mais negativa, seu peso sentimental será negativo. Caso o contrário, seu peso será positivo. Vale ressaltar que um comentário também pode ter um peso sentimental neutro, quando a forma como o conteúdo textual foi escrito não é suficientemente negativa ou positiva. Normalmente, o sentimento de um texto é calculado tendo como base um dicionário com os termos de um idioma e seus respectivos pesos sentimentais. O cálculo pode ser de forma mais simples, medindo estatisticamente o resultado com base na soma ou na média dos pesos das palavras, por exemplo; ou de forma mais complexa, utilizando redes neurais e treinando as mesmas com diferentes conjuntos de dados [Ribeiro et al. 2016, Jongeling et al. 2015].

Diversas ferramentas e métodos foram propostos para calcular o sentimento e existem estudos comparativos com o intuito de identificar qual método apresenta o melhor desempenho [Ribeiro et al. 2016, Jongeling et al. 2015]. Nos estudos, um dos métodos que mais se destaca é o *Sentistrength*, obtendo uma alta precisão nos diversos conjuntos de dados analisados. Com base nesses estudos e na facilidade de integração com o ColMiner, a abordagem utilizará o *Sentistrength* como forma de calcular o sentimento dos comentários postados em ambientes de *issue tracking* e repositórios de software.

2.2. Relevância Temática

O cálculo da relevância é outra forma de se investigar o conteúdo textual de um comentário. O objetivo é descobrir o quão relevante é um comentário para o tema da discussão de um tópico ou *issue* [Azevedo 2011]. Se um comentário estiver aderente ao assunto da discussão a sua relevância tende a ser alta, entretanto, se o comentário estiver abordando outro assunto, a relevância será penalizada.

Dentre as formas para cálculo da relevância, [Eugênio Coelho Neto and Braga e Silva 2018] propuseram uma abordagem para cálculo de relevância temática dos comentários postados em ambientes de *issue tracking*. Essa abordagem foi baseada no trabalho de [Azevedo 2011] para o contexto de troca de mensagens em fóruns educacionais. Em ambas as abordagens, utiliza-se a representação dos textos por meio de grafos, onde os termos frequentes no texto se tornam vértices e as relações entre eles se tornam arestas. Também é necessário um grafo de referência a ser comparado o grafo do comentário para medir a relevância a partir da similaridade

entre os grafos. A similaridade é calculada identificando os vértices equivalentes entre os dois grafos e verificando a distância entre eles em cada grafo. Quanto mais vértices equivalentes os grafos possuírem e quanto mais similar for a distância entre eles, maior será a similaridade.

2.3. Análise de redes sociais

A análise de redes sociais aplicada no contexto de desenvolvimento de software visa identificar relações entre os *stakeholders* [Yu et al. 2016]. Como exemplo, pode-se citar as relações entre o criador de um *issue* e o *stakeholder* que comentou ou que foi atribuído ao *issue*. Pesos também podem ser atribuídas às relações, a fim de definir o quão forte elas são. Para visualizar as relações entre *stakeholders*, uma rede pode ser construída utilizando grafos, onde os nós representam os indivíduos e as arestas as relações entre eles [Scott 2017]. Posteriormente, métricas, como as de centralidade, podem ser aplicadas sobre a rede, a fim de descobrir informações relevantes sobre os nós e suas relações.

2.3.1. Centralidade de um nó

A centralidade é uma medida de importância de um nó na rede. Existem diferentes métricas de centralidade que permitem, por exemplo, quantificar a influência de uma pessoa em uma rede social. Dentre as métricas para calcular a centralidade tem-se:

- Centralidade de grau, considera somente a quantidade de relações que um nó possui;
- Centralidade de proximidade, considera o quão próximo o nó é de todos os outros nós na rede. Para isso é necessário calcular as distâncias entre o nó e todos os outros nós, e quanto menor for o somatório das distâncias, maior será a centralidade;
- Centralidade por intermediação (*Betweenness*), considera a quantidade de caminhos mínimos entre pares de outros nós da rede que passam por um determinado nó. Utilizando essa métrica, um nó terá uma centralidade alta se ele for um intermediador das outras relações e a maioria dos caminhos mínimos entre pares de outros nós passarem por esse nó central/intermediador.

3. Trabalhos Relacionados

Nesta seção, serão discutidos os trabalhos relacionados a cada uma das três formas de análise da comunicação que serão integradas neste trabalho.

Em relação à análise de sentimentos, alguns trabalhos aplicam a abordagem no contexto do desenvolvimento de software, analisando os comentários de *commits* ou de *issues*, por exemplo, e demonstram sua utilidade ao identificar impactos do sentimento em outros fatores presentes no ambiente de desenvolvimento de software. Ortu et al. 2015 calcula o impacto do sentimento no tempo de resolução de um *issue* e descobre que o mesmo pode ser atrasado por comentários com sentimentos negativos. Em vista desses resultados, a análise de sentimentos foi escolhida para ser integrada como forma de calcular pesos para as relações entre os *stakeholders*.

Sobre a relevância temática, Eugênio Coelho Neto and Braga e Silva 2018 adaptaram, para contexto de ambientes de *issue tracking*, uma abordagem para cálculo da

relevância dos comentários voltada para o contexto educacional. As modificações realizadas foram validadas a partir de pesquisas realizadas com *stakeholders* e consistem, no geral, em tratamentos especiais para textos curtos e com poucas repetições de palavras, segundo as características das mensagens trocadas no ambiente em questão. A abordagem adaptada demonstrou ser mais apropriada e foi utilizada na ferramenta ColMiner, que permite identificar quais *stakeholders* produziam comentários relevantes e investigar o perfil de um *stakeholder* por meio de algumas medições como a quantidade de *issues* reportados. Com a ferramenta, falhas na comunicação podem ser identificadas mais facilmente, bem como os *stakeholders* que mais ou menos participaram. Neste trabalho, a relevância temática será utilizada na rede de comunicação como forma de calcular pesos para as relações entre os *stakeholders* e uma nova versão da ferramenta ColMiner (ColMiner 2.0) será apresentada, a fim de tornar a ferramenta mais abrangente.

Sobre a aplicação da análise de redes sociais em repositórios de software, Yu et al. 2016 relacionam os *stakeholders* envolvidos em um mesmo *pull request* e define uma métrica para calcular os pesos das relações, a partir da data e em quais *pull requests* os comentários foram postados. Basicamente, dentre o conjunto de comentários que compõem a relação entre dois *stakeholders*, a métrica penaliza os comentários mais antigos e os que foram feitos em somente um *pull request* ao invés de em *pull requests* diferentes. Dessa forma é considerado que comentários recentes são mais importantes e que é melhor um *stakeholder* comentar em vários *pull requests* do que comentar várias vezes em um só. A rede e a métrica foram propostas a fim de se criar um modelo para prever quem seria o melhor revisor para um *pull request*. A abordagem que será descrita neste trabalho se baseia na forma como a rede é gerada por Yu et al. 2016 e oferece a métrica proposta como uma opção para atribuir o peso das relações, entretanto, também são oferecidas outras opções para pesos, como a análise de sentimentos ou a Relevância Temática.

4. Metodologia de Pesquisa

Com o intuito de garantir a solidez da pesquisa conduzida neste trabalho foi adotada uma metodologia científica baseada no método de *Design Science Research* [Peffer et al. 2007]. A metodologia, representada na Figura 1, é dividida em cinco passos principais:

- **Conscientização:** Neste passo, um problema a ser solucionado é identificado e compreendido. Como saída tem-se a formalização do mesmo e definição de suas fronteiras. O problema em questão neste trabalho é auxiliar na identificação dos principais envolvidos em um projeto (*stakeholders*) no âmbito da comunicação.
- **Sugestão:** consiste em desenvolver, de forma criativa, uma ou mais propostas de solução para o problema encontrado. No contexto do problema citado, foram propostas as seguintes soluções: criação de uma ferramenta para análise estatística dos dados, por meio de gráficos com informações sobre os *stakeholders*; e criação de uma ferramenta para a construção de uma rede com base nas relações entre os *stakeholders* e identificação dos nós centrais, mesclando diversas análises disponíveis na literatura. A primeira proposta de solução foi desenvolvida no primeiro ciclo da pesquisa com a criação da ferramenta ColMiner

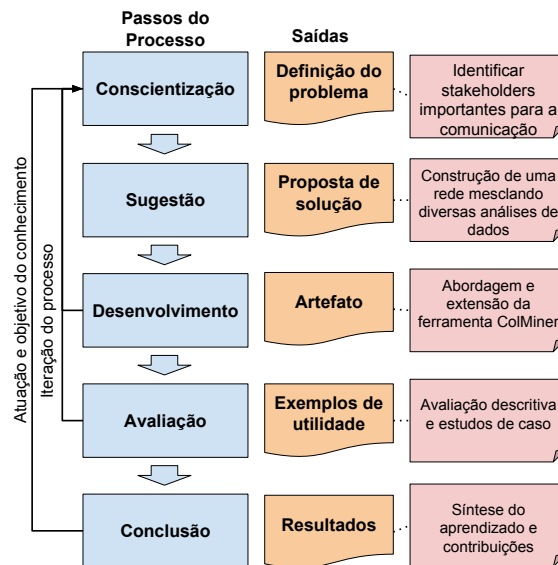


Figura 1. Metodologia adotada na Pesquisa, baseada em [Peffer et al. 2007] e [Takeda et al. 1990]

[Eugênio Coelho Neto and Braga e Silva 2018]. A segunda proposta de solução será conduzida neste trabalho como uma nova versão dessa ferramenta. Em ambos os casos, tem-se formas de se obter informações sobre a comunicação e auxiliar na identificação de *stakeholders* importantes.

- **Desenvolvimento:** Corresponde ao processo de construção do artefato e tem como saída o mesmo em estado funcional. O artefato a ser desenvolvido se caracteriza como uma abordagem para construção de uma rede com as relações entre os *stakeholders*, ponderadas com outras métricas, como análise de sentimento e relevância temática; e, posteriormente, a criação de uma nova versão da ferramenta ColMiner utilizando essa abordagem.
- **Avaliação:** definida como o processo de verificação do comportamento do artefato no ambiente para o qual foi desenvolvido. A avaliação do artefato mais adequada para este estudo é a descritiva, que utiliza argumentos com informações de bases de conhecimento e cenários em torno do artefato, a fim de demonstrar sua utilidade. Será averiguada a importância dos nós centrais apontados pela análise da rede por meio de dados disponíveis que remetem a impactos causados no projeto, como a quantidade de *commits*.
- **Conclusão:** consiste na formalização geral do processo e na síntese do aprendizado e resultados obtidos. Como resultado deste passo, tem-se a formalização da abordagem proposta, descrição da nova versão da ferramenta e estudos de caso com os principais resultados de sua utilização.

A iteração do processo presente na Figura 1 é composta pela identificação de inconsistências durante o desenvolvimento e pela avaliação da solução proposta. Inconsistências que, quando solucionadas, causam um refinamento da formalização do problema e/ou sua solução. Já o processo de atuação e objetivo do conhecimento, consiste

em investigar o quão satisfatória foi a solução desenvolvida e quais são os trabalhos futuros, realimentando o passo de conscientização de um próximo ciclo de condução da pesquisa [Peffer et al. 2007].

5. Abordagem para Construção da Rede de Comunicação

Esta seção apresentará a formalização da abordagem para construção da rede de comunicação entre os envolvidos em um projeto. O objetivo da abordagem é descobrir como os *stakeholders* se relacionam no âmbito da comunicação e auxiliar na identificação de nós centrais na rede, ou seja, aqueles com certa importância para o projeto enquanto comunicadores. A utilização do termo "*stakeholder*" se dá devido ao fato do mesmo ser mais abrangente que o termo "desenvolvedor". Na abordagem são considerados indivíduos que participaram da comunicação comentando ou reportando *issues* e/ou *pull requests*, mesmo se os mesmos não participam do grupo de desenvolvedores e não contribuíram para o projeto fazendo *commits*. A Figura 2 mostra a visão geral da abordagem, composta por 5 etapas que serão detalhadas nas próximas subseções.

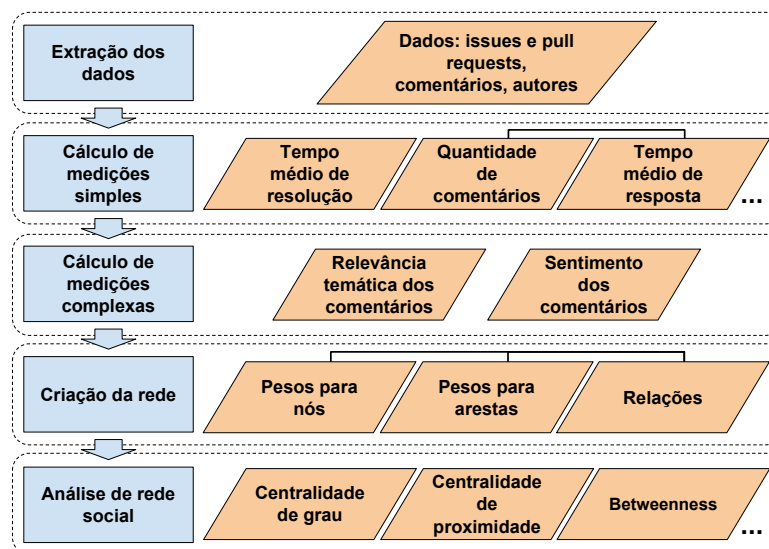


Figura 2. Etapas da abordagem proposta

5.1. Extração dos dados

A extração dos dados consiste em recuperar os dados de repositórios de software, como o GitHub, e armazená-los em um banco de dados ou em arquivos para que possam ser analisados posteriormente. Os dados a serem extraídos são referentes ao ambiente de *issue tracking*, a saber *issues*, *pull requests* e seus respectivos comentários:

- Título e descrição da *issue*;
- Data de criação da *issue*;
- Data de resolução, ou data em que a *issue* foi fechada;
- *stakeholder* autor e *stakeholder* atribuído;
- Conteúdo dos comentários;
- Autor dos comentários;
- Data de criação dos comentários.

5.2. Cálculo de medições simples

Após a extração dos dados, o próximo passo é o cálculo das medições simples, que são cálculos realizados, com o auxílio de contadores, percorrendo pelas *issues* e comentários. A partir dos dados das *issues* ou *pull requests* serão calculadas as seguintes medições:

- MS01: Tempo de resolução de cada *issue*;
- MS02: Tempo médio de resolução das *issues*;
- MS03: Quantidade de *issues* reportadas por um *stakeholder*;
- MS04: Quantidade de *issues* atribuídas a um *stakeholder*;

A partir dos dados dos comentários, serão calculados as seguintes informações:

- MS05: Tempo de resposta de um comentário;
- MS06: Tempo médio de resposta de cada *stakeholder*;
- MS07: Quantidade de *issues* que um *stakeholder* se envolveu postando um comentário;
- MS08: Quantidade de comentários postados por um *stakeholder*;
- MS09: Média do tempo de resolução das *issues* em que um *stakeholder* esteve envolvido;

5.3. Cálculo de medições complexas

As medições complexas utilizam mineração de textos, a partir do conteúdo dos comentários, e dependem da integração com outras abordagens presentes na literatura para calcular a relevância temática [Eugênio Coelho Neto and Braga e Silva 2018] e o sentimento dos comentários [Thelwall et al. 2010].

A relevância temática mede o quão relevante um comentário é para o tema da discussão [Eugênio Coelho Neto and Braga e Silva 2018]. A abordagem utilizada nesse caso é a presente em trabalhos anteriores, pois foi adaptada para o ambiente de *issue tracking* e apresenta melhores resultados. Primeiramente, é necessário gerar grafos a partir dos textos como forma de representação estrutural de seu conteúdo. O grafo possuirá os conceitos mais frequentes no texto, formados por uma ou mais palavras, e as ligações entre os conceitos que aparecem próximos uns dos outros. Um grafo deve ser gerado para cada comentário presente na *issue* (grafo do comentário), outro para a descrição da *issue* (grafo da *issue*) e um a partir de todo o conteúdo textual da *issue* (grafo da discussão), ou seja, a partir do texto de todos os comentários juntos. Posteriormente, o grafo do comentário será comparado com o grafo da *issue* e o grafo da discussão e a relevância será calculada a partir da similaridade entre os grafos. Os valores da relevância variam no intervalo de 0 (zero) a 5 [Eugênio Coelho Neto and Braga e Silva 2018].

O sentimento visa medir qual o peso sentimental dos comentários, se o mesmo representa algo de forma negativa ou positiva. Será utilizado o *SentiStrength* neste trabalho pois apresenta melhores resultados em estudos comparativos [Ribeiro et al. 2016]. A partir de um dicionário com o peso sentimental das palavras, o sentimento de um texto é calculado, utilizando redes neurais e treinando das mesmas com diferentes conjuntos de dados disponibilizados junto com a biblioteca do *SentiStrength* [Thelwall et al. 2010]. Como resultado tem-se o sentimento negativo, positivo e total de um texto ou comentário,

sendo que o sentimento total é o utilizado na abordagem e consiste na soma dos sentimentos negativo e positivo. Os valores do sentimento, no caso do sentimento total, variam de -4 a 4.

5.4. Criação da rede

Com todas as medições calculadas, procede-se com a construção da rede de comunicação a partir da identificação de relações entre *stakeholders*. Na rede, cada nó representa um *stakeholder* que criou uma *issue* ou *pull request* ou postou um comentário. Uma relação entre dois *stakeholders* é criada quando um deles comenta uma *issue* ou *pull request* postado por outro. Para representar a relação, uma aresta é adicionada à rede, saindo do nó criador para o nó que postou o comentário. A maneira como as relações são criadas se baseia no trabalho de Yu et al. 2016, embora outras formas de relacionar os nós pudessem ser propostas e/ou utilizadas, por exemplo relacionando também o autor com *stakeholder* atribuído à *issue*. A escolha dessa forma de criar relações ocorre devido ao fato da mesma estar mais associada com a comunicação em si e possuir embasamento em outros trabalhos.

Com o intuito de identificar o quão forte é uma relação ou o quão influente é um nó na rede, os nós e as arestas devem ser ponderadas com base nas medições calculadas e outras medições presentes na literatura. Para cada relação, os pesos podem ser calculados realizando a soma ou média das medições sobre os comentários trocados entre um par de *stakeholders*. Dessa forma, os pesos que serão atribuídos as arestas são:

- PA01: Relevância temática média dos comentários;
- PA02: Sentimento médio dos comentários;
- PA03: Quantidade de comentários ponderados pelo tempo [Yu et al. 2016];
- PA04: Média da relevância somada ao sentimento dos comentários;
- Outras medições simples;

O peso PA03 foi utilizado com base na métrica retirada do trabalho de Yu et al. 2016. Tal métrica foi validada pelos autores e pondera os comentários de acordo com a data e a *issue* ou *pull request* em que foram postados. Basicamente, dentre os comentários de uma relação, os comentários antigos possuem um peso menor do que aqueles mais recentes, e comentários postados em uma mesma *issue* possuem pesos menores do que aqueles postados em *issues* diferentes.

O peso PA04 foi proposto como forma de considerar tanto a relevância quanto o sentimento ao se avaliar uma relação, pois ambos são características referentes ao conteúdo textual dos comentários. A métrica referente ao PA04 é definida pela equação 1.

$$PA04 = \sum_{i=1}^n (R_i + S_i) \quad (1)$$

Onde n é o número de comentários presentes na relação, R_i a relevância do comentário i e S_i o sentimento do comentário i .

Para tentar identificar a influência ou participação dos *stakeholders* no projeto, são calculados os pesos dos nós a partir das medições disponíveis. Os seguintes pesos serão atribuídos aos nós:

- PN01: Quantidade de *commits* realizados;
- PN02: Relevância média das *issues*;
- PN03: Sentimento médio das *issues*;
- PN04: Tempo de resolução médio das *issues* que o *stakeholder* esteve envolvido;
- Outras medições simples;

5.5. Análise de rede social

Após as relações serem formadas e o peso dos nós e arestas serem calculados, a abordagem prevê a aplicação de uma análise de rede social. A análise da rede, no contexto deste trabalho, visa identificar facilmente propriedades de um nó em relação à estrutura da rede de comunicação. A fim de medir a influência de um nó na rede, são utilizadas diversas métricas de centralidade (grau, proximidade e intermediação). Vale ressaltar que, nesta etapa, também são calculados pesos para o nós, mas de forma diferente da etapa anterior. Anteriormente, os pesos foram calculados em relação a informações do *stakeholder* no projeto, nesta etapa, o cálculo ocorre em relação à estrutura da rede em si. Os pesos calculados nessa etapa são:

1. PN05: Centralidade de grau;
2. PN06: Centralidade por proximidade;
3. PN07: Centralidade por intermediação (Betweenness);

No caso da centralidade por proximidade, os pesos das arestas são utilizados para cálculo. Entretanto, devido a restrições do algoritmo de proximidade utilizado (Dijkstra [Frana and Misa 2010]), esses pesos não podem ser negativos. Como o sentimento dos comentários varia em um intervalo de -4 a 4, alguns pesos não atendem a essa restrição. Para contornar esse problema, antes de calcular a centralidade por proximidade, a escala do sentimento é convertida para valores de 0 a 8, correspondentes.

6. 0 ColMiner 2.0

A ferramenta ColMiner tem como objetivo auxiliar gerentes no processo de monitoramento da comunicação, fornecendo diversas formas de análise sobre dados de projetos armazenados no GitHub. A primeira versão da ferramenta foi desenvolvida no contexto de um outro trabalho [Eugênio Coelho Neto and Braga e Silva 2018]. Essa versão permitia o cálculo da relevância temática dos comentários postados em um ambiente de *issue tracking*, além de outras medições simples, possibilitando a geração de gráficos e tabelas a fim apoiar a gerência das comunicações.

A fim de tornar a ferramenta mais abrangente, este trabalho implementa uma nova versão da mesma, incluindo o suporte à abordagem proposta, com a criação e a análise da rede de comunicação. O ColMiner 2.0 permitirá que o usuário gere e visualize a rede por meio de um grafo, possibilitando também a customização por meio da escolha dos pesos exibidos para os nós e arestas. A nova versão junto com os outros recursos já implementados tornará mais fácil investigar o perfil dos *stakeholders* e identificar aqueles mais importantes no âmbito da comunicação.

6.1. Visão Geral do ColMiner 2.0

A ferramenta tem como principais usuários: gerentes de projeto, *Scrum Masters*, gerentes de configuração e demais interessados no monitoramento das comunicações. Dentre as funcionalidades do ColMiner, destacam-se:

1. Extrair dados de comunicação de ambientes de *issue tracking*, referentes aos comentários postados em *issues* e *pull requests* de um projeto¹;
2. Calcular medições simples;
3. Calcular medições complexas;
 - (a) Calcular a relevância temática dos comentários; e
 - (b) Calcular o sentimento dos comentários.
4. Realizar análises estatísticas por meio de gráficos e tabelas;
 - (a) Correlacionar a relevância temática e o sentimento com as outras medições por meio de gráficos de dispersão;
 - (b) Investigar quais são os *stakeholders* com os maiores valores de uma determinada medição através de gráficos de barras;
 - (c) Listar todos os *stakeholders* em uma tabela com a possibilidade de pesquisar por um *stakeholder* e ver mais informações; e
 - (d) Apresentar informações de um *stakeholder* específico, permitindo a investigação do perfil do mesmo enquanto comunicador no projeto.
5. Realizar criação e análise da rede de comunicação;
 - (a) Gerar grafo com a rede de comunicações, onde cada *stakeholder* será representado por um nó com um valor e as relações entre eles serão representadas por arestas ponderadas;
 - (b) Calcular peso das arestas e nós de acordo com as medições primitivas, complexas e medições de centralidade;
 - (c) Permitir que o usuário escolha quais serão os pesos nos nós e arestas e atualize o grafo;
 - (d) Permitir a visualização da rede em forma de tabela, onde cada linha seria um nó e cada coluna uma medição do nó (centralidade, quantidade de comentários etc).

Em relação à segunda funcionalidade, todas as medições simples descritas na abordagem (Seção 5) são calculadas pelo ColMiner, exceto a quantidade de *commits* de um desenvolvedor, que ainda não pode ser recuperada pelo ColMiner. Entretanto, para esse trabalho essa informação foi coletada manualmente no estudo de caso a fim de avaliar a abordagem e a ferramenta.

6.2. Arquitetura

A arquitetura da ferramenta ColMiner é dividida em componentes que se comunicam, a fim de realizar o processamento necessário para atender às funcionalidades especificadas. O diagrama de componentes é apresentado na Figura 3. Os componentes adicionados na nova versão desenvolvida neste trabalho são o *Dados da Rede de Comunicação* e o subcomponente *Rede*. Os demais componentes foram implementados na primeira versão da ferramenta [Eugênio Coelho Neto and Braga e Silva 2018].

As próximas subseções descrevem os componentes da ferramenta ColMiner.

¹Nesta versão, a ferramenta suporta apenas a extração de dados de repositórios do GitHub

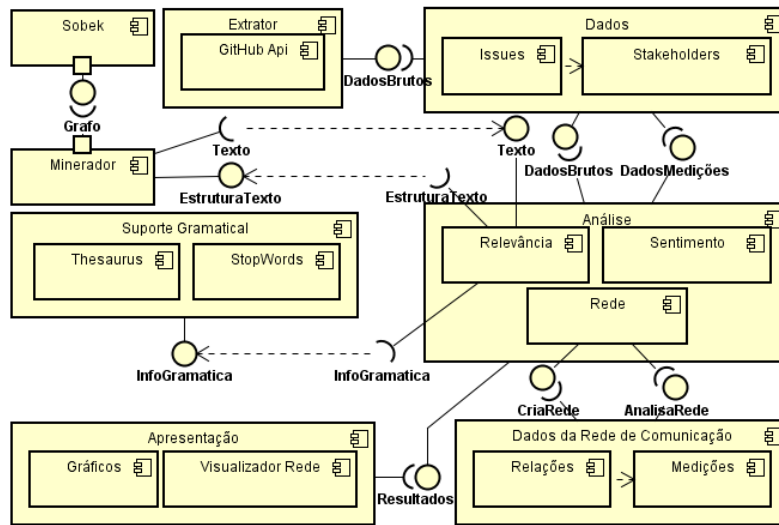


Figura 3. Componentes do ColMiner

6.2.1. Extrator

No componente *Extrator*, realiza-se a integração com os repositórios de software, a fim de extrair os dados de um projeto. Atualmente, a ferramenta suporta somente a integração com o GitHub. Esse componente representa o primeiro passo da abordagem para criação da rede (Figura 2). A partir dele, os dados relacionados à comunicação são recuperados com auxílio da API do GitHub² e fornecidos para o componente *Dados*. No caso de futuras extrações de dados de outros repositórios, basta adicionar subcomponentes no *Extrator* ou fazer com que ele se comunique com componentes externos.

6.2.2. Dados

O componente *Dados* é responsável pela representação interna e armazenamento em um banco de dados interno. Os dados sobre as *issues*, *pull Requests* e comentários são armazenados no subcomponente *Issues* e os dados sobre os autores dos mesmos ficam no subcomponente *Stakeholders*. Inicialmente, são armazenados apenas os dados brutos, sem nenhuma análise. Nesse caso, os dados representam a informação resultante da primeira etapa da abordagem (Figura 2). Os dados brutos são fornecidos para o componente de *Análise*, que calcula as medições simples e complexas. Posteriormente, o componente *Dados* consome do componente *Análise* os dados com as medições que também conterão a relevância e sentimento dos comentários e representarão as informações resultantes da segunda e do terceira etapa da abordagem.

²<https://developer.github.com/v3/libraries/>

6.2.3. Suporte Gramatical

O componente *Suporte gramatical* é responsável por fornecer informações sobre os termos de um idioma. Para isso, esse componente usa um *thesaurus* e visa fornecer os sinônimos de um termo, com o intuito de descobrir se um par de termos são sinônimos ou não. Já as *stopwords* são listas de palavras que não adicionam significado ao texto, ou seja, preposições e conjunções que pouco caracterizam os textos [Matsubara et al. 2003]. Esse suporte gramatical é requerido pelo componente *Relevância*, para se comparar dois grafos na atual abordagem para cálculo da relevância presente no ColMiner [Eugênio Coelho Neto and Braga e Silva 2018].

6.2.4. Minerador

O *Minerador* é responsável por transformar o texto não estruturado em uma representação estruturada, utilizando um minerador de textos. Para isso, ele recebe o conteúdo textual dos comentários fornecidos pelo subcomponente *Relevância* e fornece como resultado o texto estruturado. Segundo a abordagem para cálculo da relevância implementada [Eugênio Coelho Neto and Braga e Silva 2018], grafos são usados como estrutura de dados para representar os textos. Para gerar os grafos, o componente se comunica o minerador Sobek [Reategui et al. 2011], um componente externo, pois o Sobek não é uma biblioteca e sim um *webservice*. De forma resumida, o processo para estruturação do texto utilizado pelo Sobek primeiramente remove as *StopWords*, identifica os termos mais frequentes e cria relações entre eles. Ao criar as relações de um termo, considera-se os 5 termos mais próximos a ele, de acordo com a ordem no texto, e a relação é criada desde que não haja pontuação entre os termos.

6.2.5. Dados da Rede Comunicação

Dados da Rede Comunicação é o componente responsável por representar e armazenar a rede de comunicação, junto com as *relações* entre os *stakeholders* e as *medições* que serão pesos para as arestas e nós. Em relação à abordagem para criação da rede, o componente contém a informação resultante do passo *Criação da Rede*. Os *Dados da Rede Comunicação* é dividido nos subcomponentes das relações entre os *stakeholders* e das medições/pesos das arestas e nós. Tal divisão ocorre devido ao baixo acoplamento entre os dois componentes, facilitando a adição ou substituição das medições ou a forma como a rede é construída.

A representação da rede está de acordo com o diagrama de classes da Figura 4. A interface *Medição* e as classes abstratas representadas no diagrama foram criadas de forma a facilitar a expansão da rede com a adição de novas medições. Cada nova medição será implementada por uma classe que estende uma classe abstrata de medição (*MedicaoAresta* ou *MedicaoNo*). Um nó possuirá uma lista de *MedicaoNo* e um *stakeholder*. Já uma aresta possuirá uma lista de *MedicaoAresta*. Uma aresta terá também dois nós, que são o nó de origem da aresta e o nó de destino. Caso o sistema passe a utilizar uma

mantido da versão anterior da ferramenta [Eugênio Coelho Neto and Braga e Silva 2018]. O subcomponente *Visualizador Rede* faz parte da nova versão do ColMiner.

Neste componente, para facilitar a visualização da rede, um grafo é gerado como na Figura 5, onde cada *stakeholder* é representado por um nó com um peso e as relações são representadas por arestas orientadas e ponderadas. O sentido da aresta é definido de acordo com a abordagem, sendo a origem o *stakeholder* que reportou um *issue/pull request* e o destino o *stakeholder* que fez um comentário. O valor do peso é exibido ao lado da aresta, e a espessura da mesma também varia em relação ao peso. Vale ressaltar que o peso da aresta aparece mais perto do nó de origem para possibilitar a visualização em caso de relações bidirecionais. Dessa forma, na rede da Figura 5 o peso da aresta saindo *stakeholder* "iziki" para o "buptkang" é 2 e o peso para a aresta na direção oposta é 1. Caso um *stakeholder* reporte um *issue/pull request* que não foi comentado por outro *stakeholder*, seu nó na rede não possuirá relações com outros nós e sua cor será cinza. A apresentação deve permitir que usuário altere os pesos presentes como nas arestas e nós. Para isso a caixa no canto direito da interface é exibida, onde o usuário pode escolher as medições e clicar em *Generate* para atualizar a rede.

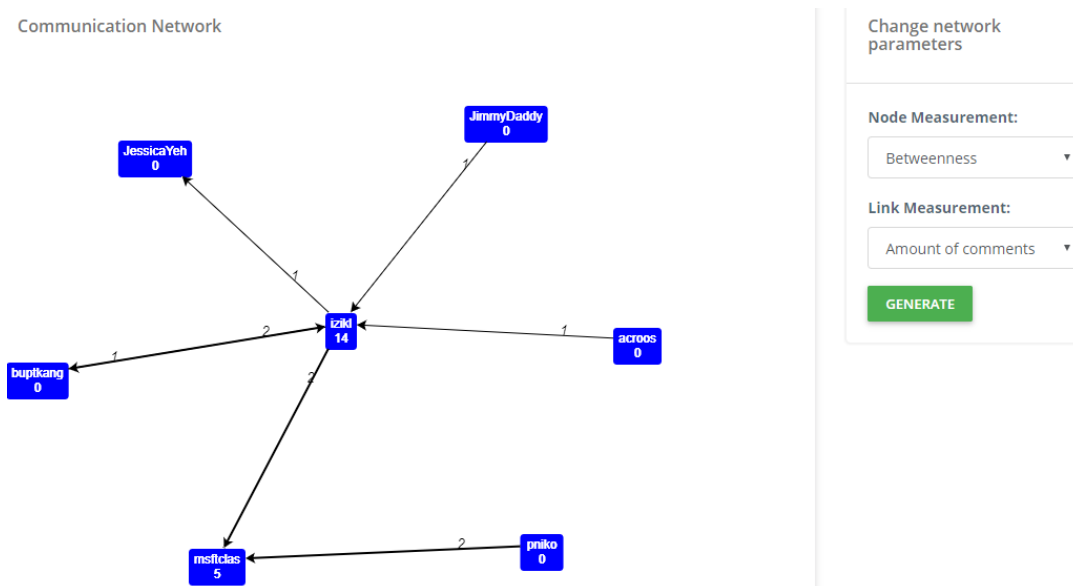


Figura 5. Interface do componente de apresentação

6.3. Detalhes Técnicos

O ColMiner é uma ferramenta web desenvolvida em Java com auxílio do Spring Framework³. Ela está disponível no site do Núcleo de Pesquisa em Engenharia de Software e Sistemas Colaborativos (NuPESSC)⁴.

No subcomponente *Sentimento*, utiliza-se a biblioteca Java do SentiStrength⁵ para calcular o sentimento dos comentários. Tal biblioteca possui métodos de inicialização

³<https://spring.io/>

⁴<http://nupessc.caf.ufv.br/>

⁵<http://sentistrength.wlv.ac.uk/>

com os arquivos que contêm os pesos de cada termo de um idioma, e métodos que recebem um conteúdo textual e retornam o sentimento negativo, o positivo e o sentimento total que é a soma dos dois anteriores.

Para auxiliar na criação da rede e calcular as medições de centralidade de um nó, o subcomponente *Rede* utiliza da biblioteca JUNG 2⁶. A mesma fornece uma representação de grafos orientada a objetos, com recursos úteis como métodos para encontrar uma aresta ou nó na rede, ou algoritmos de análise de rede social. Vale ressaltar que o cálculo da centralidade por intermediação (*Betwenness*) realizado pelo JUNG é custoso e por essa razão considera que todos os pesos das arestas são iguais a 1.

A fim de exibir o grafo que representa a rede no subcomponente *visualizador Rede*, utiliza-se a biblioteca JavaScript Arbor JS⁷. A mesma contém diversos recursos para plotar grafos e permite que usuário mude a disposição da rede arrastando os nós para uma posição desejada. O Arbor JS executa o processamento para renderizar a rede no lado do cliente, sendo que os dados necessários sobre a rede são transmitidos do servidor para o cliente por meio de arquivos JSON.

7. Prova de Conceito

A fim de se avaliar e demonstrar a utilidade da abordagem, juntamente com a nova versão da ferramenta ColMiner, foram recuperados alguns projetos presentes no GitHub.

Ao todo, foram extraídos oito projetos com conjuntos de dados de diversos tamanhos, de diferentes grupos e linguagens de programação. A variedade das características dos projetos foi mantida com o intuito de identificar em quais casos a abordagem apresenta resultados mais interessantes e em quais casos isto não ocorre. Os projetos recuperados estão presentes na Tabela 1. Nesta seção, será utilizado o termo "contribuidores" para se referir a *stakeholders*/desenvolvedores que contribuíram para o projeto realizando *commits*, e "comunicadores" para se referir a *stakeholders* que participaram da comunicação reportando ou comentando alguma *issue*.

Uma das formas de medir a produtividade de um desenvolvedor em um projeto é o número de *commits* realizados. A Tabela 1 apresenta uma relação entre os *stakeholders* que mais contribuíram fazendo *commits* e aqueles que mais participaram da comunicação, sendo nós centrais na rede. Para cada métrica de centralidade dos nós, é calculada a porcentagem dos *stakeholders* com mais *commits* que também são os mais centrais na rede de comunicação, a fim de descobrir a assertividade da métrica ao comparar os contribuidores com os comunicadores. Tais dados foram separados entre o primeiro colocado (Top 1) e os três primeiros colocados (Top 3). Por exemplo, em relação à centralidade de grau, no projeto "App Metadata", o *stakeholder* que mais fez *commits* também foi o comunicador mais central na rede ($1/1 = 100\%$). Já dentre os 3 principais contribuidores (Top 3), 2 deles foram comunicadores mais centrais na rede ($2/3 = 66\%$). Vale ressaltar que o Top 3 não considera a ordem da colocação, mas apenas se o *stakeholder* está entre os 3 primeiros colocados ou não. Além das métricas de centralidade, as porcentagens também foram calculadas em relação ao *Score* total do *stakeholder*. Esse *score* foi calculado somando-se

⁶<http://jung.sourceforge.net/doc/api/index.html>

⁷<http://arborjs.org/>

o valor das quatro métricas de centralidade. Também é apresentada uma relação entre a quantidade de comunicadores e a quantidade de contribuidores em cada projeto. No caso do "App Metadata", por exemplo, existem 7 contribuidores dentre 7 comunicadores, ou seja, 100% dos contribuidores também foram comunicadores.

Tabela 1. Porcentagem dos principais comunicadores que também foram principais contribuidores

Projeto	Contribuidores /Comunicadores	Grau		Betweenness		Proximidade (#Comentários)		Proximidade (Comentários ponderados)		Score Total	
		Top 1	Top 3	Top 1	Top 3	Top 1	Top 3	Top 1	Top 3	Top 1	Top 3
Microsoft App Metadata	7/7 (100%)	100% (1)	66,6% (2)	100% (1)	33,3% (1)	100% (1)	66,6% (2)	100% (1)	66,6% (2)	100% (1)	66,6% (2)
Jetbrains kotlin	278/412 (67%)	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)
Netflix titus-control	9/14 (64%)	0% (0)	66,6% (2)	0% (0)	66,6% (2)	0% (0)	66,6% (2)	0% (0)	66,6% (2)	0% (0)	66,6% (2)
Plonegovr Brasil.gov.tiles	12/21 (57%)	0% (0)	66,6% (2)	0% (0)	66,6% (2)	0% (0)	66,6% (2)	0% (0)	66,6% (2)	0% (0)	66,6% (2)
Okfn Serenata de Amor	98/235 (41%)	100% (1)	66,6% (2)	100% (1)	66,6% (2)	100% (1)	66,6% (2)	100% (1)	66,6% (2)	100% (1)	66,6% (2)
Google Easyypi	4/11 (36%)	100% (1)	66,6% (2)	100% (1)	66,6% (2)	100% (1)	33,3% (1)	100% (1)	33,3% (1)	100% (1)	66,6% (2)
Oracle Graal	114/409 (27%)	100% (1)	66,6% (2)	0% (0)	66,6% (2)	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)	66,6% (2)
Google go-cloud	28/109 (25%)	0% (0)	66,6% (2)	0% (0)	66,6% (2)	0% (0)	66,6% (2)	0% (0)	33,3% (1)	0% (0)	66,6% (2)
Média	52,1%	50%	58,32%	37,5%	54,1%	37,5%	45,7%	37,5%	41,6%	37,5%	58,2%

Dentre as métricas de centralidade, a que melhor relacionou os principais comunicadores com os principais contribuidores foi a centralidade de grau. No top 1, em 50% das vezes, o contribuidor que mais fez *commits* também foi o apontado como mais central na rede de comunicação. No top 3, 58% dos três contribuidores que mais fizeram *commits* também foram apontados como os três comunicadores mais centrais. A segunda métrica mais assertiva foi a Betweenness, sendo que no top 1 não houve uma relação muito forte. Entretanto, no top 3, 54,1% dos 3 principais contribuidores também foram os 3 mais centrais da rede. A única métrica que considera o peso das arestas é a centralidade de proximidade. Por essa razão, ela foi calculada duas vezes, uma considerando a quantidade de comentários como peso e outra considerando a métrica de Yu et al. 2016 de comentários ponderados como peso. No geral, não foi muito forte a relação entre os principais contribuidores e o nós mais centrais apontados pela centralidade de proximidade e praticamente não houve diferença entre as escolhas de peso da aresta.

Em relação ao *score* total, no conjunto de dados analisado, somente em 37,5% das vezes o contribuidor que mais realizou *commits* também foi o comunicador com maior *score*. Já dentre o top 3, o *score* superou as métricas de centralidade, com exceção da centralidade de grau. Sendo que, em 58,2% das vezes, os 3 contribuidores com mais *commits* também foram um dos 3 comunicadores com maior *score*.

Um fato notável foi que, no projeto "Jetbrains Kotlin", a relação entre principais contribuidores e os principais comunicadores praticamente não existiu. Tal projeto tem muitos *stakeholders* envolvidos e se trata do desenvolvimento de uma linguagem de programação. Também existem muitos comunicadores que não realizaram *commits*. Alguma característica da finalidade do projeto e/ou da comunidade pode ter causado a falta dessa relação, como por exemplo, os papéis dos *stakeholders*.

Em relação à quantidade de contribuidores e comunicadores, observando a Tabela 1, é possível perceber que é frequente a presença de *stakeholders* que participaram da comunicação mas que não fizeram *commits*. Em média, quase metade (47,9%) dos *stakeholders* foram comunicadores sem realizar *commits*. Essa característica é presente em projetos *Open Source* devido ao fato de que qualquer pessoa no *GitHub* pode comentar

ou reportar uma *issue*. Dessa forma, isso pode ser uma interferência na relação entre os principais comunicadores e os principais contribuidores, principalmente quando se trata do top 1, já que existem *stakeholders* ativos na comunicação que não são contribuidores.

No geral, parece existir uma relação entre os *stakeholders* mais ativos na comunicação e os que mais contribuíram para o projeto realizando *commits*. Há uma tendência dos principais contribuidores estarem mais envolvidos com a comunicação, principalmente ao considerar a centralidade de grau.

8. Conclusão

Este trabalho apresentou uma abordagem para construção de um rede de comunicação a partir dos dados presentes em ambientes de *issue tracking*. A abordagem integrou três outras formas de análise da comunicação presentes da literatura, a fim de oferecer uma análise mais abrangente, sendo duas análises do conteúdo textual dos comentários (relevância temática e sentimento) e uma análise de rede social. As etapas para criação e análise da rede foram descritas detalhadamente, de forma a facilitar a replicação da abordagem.

O ColMiner 2.0 foi desenvolvido para implementar a abordagem, com o intuito de tornar a aplicação da mesma mais fácil. Com a nova versão, a ferramenta se tornou mais abrangente ao auxiliar o monitoramento da comunicação por meio de análise dos dados dos projetos. A nova versão junto aos recursos já disponíveis anteriormente na ferramenta possibilitaram a identificação de *stakeholders* que fazem a diferença contribuindo e participando ativamente da comunicação em um projeto.

Os resultados da aplicação da abordagem em alguns projetos foram satisfatórios. No geral, parece existir uma relação entre os *stakeholders* principais na comunicação e aqueles que fizeram mais *commits*. Um *stakeholder* apontado pela abordagem como central na rede e chave para comunicação tende a ser também um dos *stakeholders* mais produtivos. Com base nesse resultado, pode-se dizer que esses *stakeholders* fazem a diferença, seja como comunicadores, contribuidores ou ambos. Tais *stakeholders* possuem um papel importante seja no esclarecimento e/ou repasse de informações referentes às *issues* ou na realização de *commits* e desenvolvimento de artefatos. Dessa forma, eles causam um impacto positivo ao participarem de um projeto e devem ter uma importância significativa para o sucesso do mesmo.

A análise de rede social realizada neste trabalho baseou-se no uso de métricas de centralidade dos nós, embora outras métricas possam ser utilizadas, como a detecção de comunidades. Tais métricas poderiam ser integradas em trabalhos futuros e gerar resultados úteis. Entretanto, faz-se necessário investigar quais seriam suas implicações no contexto deste trabalho.

Devido à variedade de características de um projeto e sua equipe de *stakeholders*, existem dificuldades para uma análise mais assertiva em alguns projetos. Estudos sobre essas características poderiam ser conduzidos em trabalhos futuros, a fim de investigar melhor essa questão. Poderia ser gerado um mapeamento dos diversos tipos de projetos, analisando os resultados das métricas, a fim de identificar diferenças na estrutura da comunicação dos mesmos.

Referências

- [Azevedo 2011] Azevedo, B. F. T. (2011). Minerafórum : um recurso de apoio para análise qualitativa em fóruns de discussão. Tese de doutorado em informática na computação, Universidade Federal do Rio Grande do Sul.
- [Bertram et al. 2010] Bertram, D., Volda, A., Greenberg, S., and Walker, R. (2010). Communication, collaboration, and bugs: The social nature of issue tracking in small, collocated teams. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work, CSCW '10*, pages 291–300, New York, NY, USA. ACM.
- [da Trindade et al. 2008] da Trindade, C. C., de Moraes, A. K. O., and Meira, S. L. (2008). Comunicação em equipes distribuídas de desenvolvimento de software: Revisão sistemática. *5th Experimental Software Engineering Latin American Workshop (ESELAW)*.
- [Eugênio Coelho Neto and Braga e Silva 2018] Eugênio Coelho Neto, L. and Braga e Silva, G. (2018). Colminer: uma ferramenta de apoio ao gerenciamento das comunicações em um ambiente de issue tracking. *XIV Simpósio Brasileiro de Sistemas de Informação - SBSI*.
- [Fan 2010] Fan, D. (2010). Analysis of critical success factors in it project management. In *Industrial and Information Systems (IIS), 2010 2nd International Conference on*, volume 2, pages 487–490. IEEE.
- [Frana and Misa 2010] Frana, P. L. and Misa, T. J. (2010). An interview with edsgar w. dijkstra. *Commun. ACM*, 53(8):41–47.
- [Han and Jung 2014] Han, J. and Jung, W. (2014). Extracting communication structure of a development organization from a software repository. *Personal and Ubiquitous Computing*, 18(6):1413–1421.
- [Jongeling et al. 2015] Jongeling, R., Datta, S., and Serebrenik, A. (2015). Choosing your weapons: On sentiment analysis tools for software engineering research. In *2015 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pages 531–535. IEEE.
- [Matsubara et al. 2003] Matsubara, E. T., Martins, C. A., and Monard, M. C. (2003). Pretext: Uma ferramenta para pré-processamento de textos utilizando a abordagem bag-of-words. *Technical Report*, 209.
- [Ortu et al. 2015] Ortu, M., Adams, B., Destefanis, G., Tourani, P., Marchesi, M., and Tonelli, R. (2015). Are bullies more productive?: empirical study of affectiveness vs. issue fixing time. In *Proceedings of the 12th Working Conference on Mining Software Repositories*, pages 303–313. IEEE Press.
- [Peppers et al. 2007] Peppers, K., Tuunanen, T., Rothenberger, M. A., and Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of management information systems*, 24(3):45–77.
- [PMI 2013] PMI (2013). *PMBok. A Guide to the Project Management Body of Knowledge*. 5th edition.
- [Reategui et al. 2011] Reategui, E., Klemann, M., Epstein, D., and Lorenzatti, A. (2011). Sobek: A text mining tool for educational applications. In *Proceedings of the International Conference on Data Mining (DMIN)*, page 1. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).

- [Ribeiro et al. 2016] Ribeiro, F. N., Araújo, M., Gonçalves, P., Gonçalves, M. A., and Benevenuto, F. (2016). Sentibench-a benchmark comparison of state-of-the-practice sentiment analysis methods. *EPJ Data Science*, 5(1):1–29.
- [Scott 2017] Scott, J. (2017). *Social network analysis*. Sage.
- [Takeda et al. 1990] Takeda, H., Veerkamp, P., and Yoshikawa, H. (1990). Modeling design process. *AI magazine*, 11(4):37.
- [Thelwall et al. 2010] Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., and Kappas, A. (2010). Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology*, 61(12):2544–2558.
- [Yang et al. 2017] Yang, B., Wei, X., and Liu, C. (2017). Sentiments analysis in github repositories: An empirical study. In *Software Engineering Conference Workshops (AP-SECW), 2017 24th Asia-Pacific*, pages 84–89. IEEE.
- [Yu et al. 2016] Yu, Y., Wang, H., Yin, G., and Wang, T. (2016). Reviewer recommendation for pull-requests in github: What can we learn from code review and bug assignment? *Information and Software Technology*, 74:204–218.