Desambiguação de autoria em listas de discussão de projetos de software livre

José Teodoro da Silva

DISSERTAÇÃO DE MESTRADO
APRESENTADA AO INSTITUTO DE
MATEMÁTICA E ESTATÍSTICA DA
UNIVERSIDADE DE SÃO PAULO

Programa: Mestrado em Ciência da Computação
Orientador: Prof. Dr. Marco Aurélio Gerosa

Durante o desenvolvimento deste trabalho o autor recebeu
auxílio financeiro do CNPq

São Paulo, agosto de 2015

Desambiguação de autoria em listas de discussão de projetos de software livre

José Teodoro da Silva

Esta é a versão original da dissertação elaborada pelo candidato José Teodoro da Silva, tal como submetida à Comissão Julgadora.

Comissão Julgadora:

- Prof. Dr. Marco Aurélio Gerosa (orientador) – IME-USP
- Prof. Dr. Nazareno Ferreira de Andrade – DSC - Universidade Federal de Campina Grande
- Prof. Dr. Fernando Figueira Filho – Universidade Federal do Rio Grande do Norte

Resumo

DA SILVA, J. S. Desambiguação de autoria em listas de discussão de projetos de software livre. 2015. 34f. Dissertação (Mestrado) – Instituto de Matemática e Estatística, Universidade de São Paulo, 2015.

Listas de discussão possibilitam a comunicação entre várias pessoas utilizando a estrutura do e-mail. Listas são utilizadas para discutir diversos assuntos, desde entretenimento até desenvolvimento de software. Elas constituem uma fonte rica de informações sobre a comunicação de seus membros e o histórico das interações é utilizado para estudos quantitativos sobre o comportamento, organização e evolução da comunidade ali existente. Entretanto, usuários utilizam múltiplos endereços de e-mail, que acabam sendo interpretados como diferentes pessoas em muitos estudos, distorcendo os resultados das análises de redes sociais e levando a conclusões equivocadas. Para evitar esse tipo de problema, alguns trabalhos propõem heurísticas para determinação única do autor das mensagens, porém pouco se sabe sobre o quão efetiva são essas heurísticas. O objetivo deste trabalho é comparar 6 heurísticas de desambiguação de autores utilizadas na literatura. Neste estudo, utilizamos as listas de discussão de 150 projetos de software livre da Fundação Apache e encontramos indícios de que o número de endereços de e-mails utilizados na comunidade pode influenciar a qualidade dos resultados das heurísticas e que a escolha da heurística de desambiguação de autores depende do conjunto de dados a ser utilizado. Construíamos uma base de referência com base em dados disponíveis no gerenciador de funcionalidades, no repositório de chaves públicas, nos sites dos projetos e na literatura. Nossos resultados apresentam indícios de que o tamanho da comunidade influencia a qualidade dos resultados dessas heurísticas e que todas as heurísticas produzem melhores resultados quando utilizam intervalos de tempo menores em vez de utilizar todo o histórico das listas de discussão. Os resultados deste trabalho podem servir de base para pesquisadores que investigam listas de discussão de comunidades abertas com grande número de participantes.

Palavras chaves: Desambiguação de autores; listas de discussão; Fundação Apache; sistemas de comunicação; mineração de repositórios; desambiguação de e-mails.

Abstract

DA SILVA, J. S. Joining identities on Open Source Project mailing lists. 2015. 34f. Dissertação (Mestrado) – Instituto de Matemática e Estatística, Universidade de São Paulo, 2015.

Mailing lists enable communication using the structure of the e-mail. We can use these lists to discuss about various topics, from entertainment to software development. These lists and are a valuable source of information about the community communication. Researchers had used their history of interactions for quantitative studies on behavior, organization and evolution of existing community there. However, the users use multiple e-mail addresses ant this can affect the results of studies using this data. To avoid problems with multiples addresses, some researchers proposed heuristics to join multiple e-mail addresses. There are few studies about how effective are these heuristics. This work compares 6 heuristics from the literature on 150 mailing lists of open source project of the Apache Foundation. We found evidences that the data set may influence the quality and the disambiguation heuristics work better with lower data sets. Our results can help researches to choose a heuristic.

Keywords: E-mail address disambiguation; Mailing lists; Apache Software Fundation; data mining.

Sumário

1 Introdução

Uma lista de discussão possibilita a comunicação e a difusão de informação para os assinantes utilizando a estrutura do e-mail (Pimentel & Fuks 2011). Listas são usadas para discutir vários assuntos, desde arte e entretenimento até o desenvolvimento de software. Listas são usadas tanto por pequenos grupos quanto por grandes comunidades abertas de produção coletiva, com centenas ou milhares de assinantes. Nas comunidades de software livre, por exemplo, essas listas são de grande importância, sendo utilizadas para informar sobre o status do projeto, discutir sobre problemas no software, procurar por instruções de uso, coordenar os membros do projeto, enviar avisos e normas, etc. (Guzzi et al. 2013a).

Os históricos de listas de discussão são uma fonte rica de informações para pesquisas que exploram a comunicação e interação social (Hassan 2008). Listas são utilizadas para entender a estrutura de liderança e relacionamentos entre os membros da comunidade (Squire 2013) e para analisar padrões de discurso de estudantes (Overbaugh 2002), por exemplo. Técnicas de análise de redes sociais são aplicadas, extraindo de forma algorítmica os participantes (nós da rede) e as trocas de mensagens (arestas). Em projetos de desenvolvimento de software, listas de discussão são utilizadas para estudar diferentes aspectos do desenvolvimento de software, por exemplo, explorar a estrutura da comunidade (Xu et al. 2005a), analisar a rede social da comunidade a partir da sua comunicação (Roberts et al. 2006), entender a evolução do software livre a partir de discussões (D'Ambros et al. 2008), estudar os papéis dos membros na lista (Oliva et al. 2012), analisar seu processo e práticas de desenvolvimento (Bacchelli et al. 2012; Guzzi et al. 2013a), explorar a comunicação entre seus colaboradores (Nia et al. 2010a) e analisar como a participação na lista pode afetar os novos membros da comunidade (Steinmacher et al. 2012).

Entretanto, a extração de dados por meio de algoritmos não é uma tarefa trivial. Entre outros, há o problema de desambiguação dos autores das mensagens, conforme relatado por Bettenburg et al. (Bettenburg et al. 2009) e Bird et al. (Bird et al. 2006). Falhas na identificação geram atribuição incorreta das mensagens a um autor e inserem ameaças à validade dos resultados de análises quantitativas (Bettenburg et al. 2009). A dificuldade na extração de dados das listas se dá pelo modo como os membros utilizam o e-mail: usuários criam endereços de e-mail relativamente curtos[1]; não existe padronização na criação de contas de e-mail; endereços de e-mail empresariais são abandonados quando o usuário deixa a empresa; membros da comunidade usam seus endereços de e-mail pessoais e profissionais para participar na lista; e os clientes de e-mail são, em muitos casos, configurados com o nome do remetente de maneira incorreta.

Para solucionar o problema da desambiguação dos autores das mensagens, pesquisadores como Bird et al. (Bird et al. 2006), Oliva et al. (Oliva et al. 2012), Goeminne e Mens (Goeminne 2013) e Kouters et al. (Kouters et al. 2012) propuseram heurísticas que utilizam as informações existentes na própria lista de discussão para identificar múltiplos endereços de um participante.

[1] Um estudo preliminar aponta indícios de que 54% dos membros dos 150 projetos avaliados neste trabalho utilizam os prefixos de endereços de e-mail com até sete caracteres. Os detalhes desse estudo estão na seção 3.2.

Ainda assim, muitos trabalhos da literatura não realizam um pré-processamento dos dados, considerando cada endereço de e-mail como sendo de uma pessoa distinta. Parte disso pode advir da falta de conhecimento do problema ou de seus efeitos. Há uma carência de trabalhos que avaliem e comparem a eficácia dessas heurísticas.

O objetivo deste trabalho é avaliar os resultados do uso de 6 heurísticas para desambiguação de autores encontradas na literatura. Utilizamos informações publicadas pela Fundação de Software Apache para a construção de uma base de referência com endereços de e-mail dos participantes da lista de discussão de 150 projetos.

1.1 Questões de pesquisa

A partir dos problemas na desambiguação de autores e da existência das várias heurísticas na literatura para tratar estes problemas, elaboramos as seguintes questões de pesquisa para direcionar esta pesquisa:

- Os resultados das heurísticas são diferentes quando se utiliza todo o histórico das listas do que quando se utiliza intervalos de tempo menores?
- O número de endereços de e-mail (tamanho da comunidade) influencia na qualidade dos resultados das heurísticas?
- Há uma heurística mais eficaz que as demais nos casos avaliados?

1.2 Objetivo

O objetivo geral deste trabalho é avaliar os resultados de 6 heurísticas de desambiguação de autores em listas de discussão de desenvolvedores para 150 projetos de software livre para auxiliar na condução de estudos que utilizem essas listas como fonte de dados. Os objetivos específicos são:

I. Revisar a literatura para encontrar métodos de extração de informação a partir de listas de discussão;
II. Conceber um método de construção de uma base de referência para verificação dos resultados de heurísticas de desambiguação de autoria para listas de discussão;
III. Avaliar os resultados das heurísticas selecionadas da literatura;
IV. Avaliar a ocorrência de falhas na desambiguação de autores utilizando as heurísticas selecionadas na literatura;
V. Avaliar a relação entre qualidade dos resultados das heurísticas e tamanho das amostras utilizada;
VI. Avaliar a relação entre qualidade dos resultados das heurísticas e o tamanho das comunidades analisadas;
VII. Reproduzir um estudo que utilize as listas de discussão e avaliar os impactos que a mudança de heurística proporciona.

1.3 Contribuições

As contribuições deste trabalho são:

I. Elaboração de um método para construção de bases de referência para avaliar a quantitativamente os resultados de heurísticas de desambiguação de autoria em listas de discussão;
II. Avaliação dos resultados de seis heurísticas para desambiguação de autores em listas de discussão;
III. Quantificação da relação entre os períodos de análise, o tamanho das comunidades e as falhas nos resultados das heurísticas;
IV. Comparação dos resultados de um estudo utilizando as diferentes heurísticas avaliadas nesta dissertação.

1.4 Trabalhos publicados durante o período do programa de mestrado

DA SILVA, José Teodoro et al. An Extensible Service for Experts Recommendation on Distributed Software Development Projects. In: Global Software Engineering Workshops (ICGSEW), 2012 IEEE Seventh International Conference on. IEEE, 2012. p. 18-21.

OLIVA, Gustavo Ansaldi; DA SILVA, José Teodoro; GEROSA, Marco Aurélio; SANTANA, Francisco; WERNER, Claudia M. L; DE SOUZA, Cleidson R. B.; OLIVEIRA, Kleverton C. M. Characterizing key developers: a case study with apache ant. Em processo de revisão e avaliação para o C&I Special Issue Journal.

DA SILVA, José Teodoro; GEROSA, Marco Aurélio; STEINMACHER, Igor F.; WIESE, Igor S. Quem é quem na lista de discussão? Identificando diferentes e-mails de um mesmo participante. Trabalho aceito no Simpósio Brasileiro de Sistemas Colaborativos - SBSC 2015.

1.5 Organização do texto

Este trabalho está organizado da seguinte forma: a Seção 2 apresenta o estado da arte das listas de discussão e da desambiguação de autores; a Seção 3 apresenta detalhes do método de construção da base de referência e da comparação das heurísticas; a Seção 4 apresenta os resultados da comparação; a Seção 5 apresenta a discussão sobre o uso das listas, a ética dos estudos utilizando essas listas e os impactos das diferentes heurísticas na replicação de um estudo da literatura; a Seção 6 apresenta as ameaças à validade do estudo e a Seção 7 apresenta as conclusões e trabalhos futuros.

2 Estado da arte

A desambiguação de informações que representam um mesmo conceito ocorre quando múltiplos formatos, ou a ausência de qualquer padrão, são utilizados para representar um mesmo atributo. Este problema é maior quando são utilizados dados de um longo período de tempo (Dendek et al. 2013). Exemplos de usos para a desambiguação são a desambiguação de autores em coleções de artigos científicos (Dendek et al. 2013), a identificação de indivíduos por seus nomes em coleções de documentos (Godby et al. 2010) e a desambiguação de autores de mensagens em listas de discussão (Bird et al. 2006). Neste trabalho, quando falarmos sobre desambiguação de autores estaremos nos referindo à identificação no contexto das listas de discussão.

Os trabalhos encontrados na literatura sobre a desambiguação de autores podem ser divididos em quatro grupos: trabalhos que alertaram sobre as dificuldades de extração de informações das listas de discussão e desambiguação dos autores das mensagens; trabalhos que propuseram soluções para desambiguação de identidades; trabalhos que utilizaram as heurísticas e trabalhos que avaliaram as soluções propostas.

2. 1 Listas de discussão

Repositórios de comunicação de projetos de software livre são uma fonte de informações de processo e práticas de desenvolvimento de software (Guzzi et al. 2013b). Nesse contexto, as listas de discussão possuem informações que possibilitam analisar o comportamento dos desenvolvedores e usuários, as interações sociais entre os participantes e correlacionar as atividades de desenvolvimento com a lista de discussão (Bacchelli et al. 2010) .

A lista de discussão possibilita a comunicação e a difusão das mensagens para um grande número de pessoas utilizando a estrutura do e-mail. Uma lista de discussão funciona como um endereço de e-mail que agrupa os endereços de todos os assinantes dessa lista. Qualquer mensagem enviada para o e-mail do grupo será replicado automaticamente para todos seus assinantes (Nia et al. 2010a; Pimentel & Fuks 2011). As primeiras implementações de servidores automáticos de listas de discussão datam do fim dos anos 80, contudo essas listas já existiam antes disso, mas eram gerenciadas manualmente (Pimentel & Fuks 2011). Quando se trata de uma lista de discussão de um projeto de software, os remetentes de mensagens nas listas de desenvolvedores de um software livre, por exemplo, podem ser repórteres de falha, colaboradores esporádicos, usuários e os desenvolvedores do projeto (Nia et al. 2010b). Eles utilizam a lista para gerenciar atividades e organizar a comunidade do projeto.

Essas listas facilitam a disseminação das mensagens para todos os membros e usuários do projeto. A lista de discussão também possibilita o armazenamento do histórico das mensagens. A partir do histórico da comunicação é possível explorar as interações humanas e o processo de desenvolvimento do projeto. D'Ambros et al. indicam que a utilização de redes sociais geradas a partir de listas de discussão são uma fonte rica para entendimento da evolução de software (D'Ambros et al. 2008) e pesquisadores utilizam essa fonte de informação para entender aspectos

relacionados a colaboração, organização e evolução das comunidades de software (Guzzi et al. 2013b; Valverde & Solé 2007; Xu et al. 2005b).

2.2 Dificuldades de extração de informações das listas de discussão

Com relação a trabalhos que alertam sobre as dificuldades de extração de informações das listas de discussão e sua atribuição de autoria, encontramos o trabalho de Bettenburg et al. (Bettenburg et al. 2009). Esse trabalho enumera os desafios existentes no processamento de dados oriundos das listas de discussão, dentre eles a atribuição de autoria e a remoção de mensagens automáticas. Os autores alertam sobre os possíveis erros de análise que ocorrem devido a uma má determinação dos autores das mensagens.

Hemmati et al. (Hemmati et al. 2013) também descrevem as dificuldades de recuperação de dados das listas. Eles realizaram um levantamento das boas práticas utilizadas em artigos publicados em conferências e workshops. Hemmati et al. apontam para as dificuldades existentes na recuperação das características da comunidade e apresentam várias abordagens utilizadas pelos pesquisadores na tentativa de mitigar os erros de atribuição incorreta de endereços de e-mail a participantes no processo de desambiguação de autores.

2.3 Propostas de desambiguação

Uma das abordagens mais citadas[2] na literatura é a de Bird et al. (Bird et al. 2006), que apresenta uma heurística para recuperação desambiguação de autores da lista de discussão. Sua abordagem utiliza o agrupamento de endereços de e-mail a partir da similaridade entre os endereços e nomes encontrados nos cabeçalhos das mensagens da lista de discussão. Sua heurística mapeia padrões comuns de criação de endereços de e-mail e procura identificar similaridades entre os vários endereços de e-mail de uma mesma pessoa.

Essa heurística utiliza a similaridade de Levenshtein (Navarro et al. 2001) para avaliar a semelhança entre dois nomes/endereços. Quaisquer endereços/nomes que obtiverem uma similaridade acima de um limite de tolerância (0,93) são consideradas como pertencentes à mesma pessoa. Antes de realizar a identificação, a heurística remove acentos e pontuações nos nomes e divide o nome completo em duas partes: nome e sobrenome. Dados a função de similaridade *simil*, o *prefixoDoEmail* do endereço de e-mail (sem o domínio), o limite de tolerância *t* e o *nomeCompleto* dividido entre nome e sobrenome, a heurística considera os *endereçoA* e *endereçoB* como pertencentes à mesma pessoa nos seguintes casos:

- *simil(nomeCompletoA, nomeCompletoB) ≥ t;*
- *simil(nomeA, nomeB) ≥ t E simil(sobrenomeA, sobrenomeB) ≥ t;*
- *prefixoDoEmailB contém nomeA e sobrenomeA;*

[2] O trabalho de Bird et al. possui 398 citações. Essa consulta foi realizada no site https://scholar.google.com.br em agosto de 2015.

- *prefixoDoEmailB contém nomeA e a primeira letra do sobrenomeA;*
- *prefixoDoEmailB contém a primeira letra do nomeA e sobrenomeA completo; ou*
- *simil(prefixoEmailA, prefixoEmailB) ≥ t.*

A equação de similaridade simil é definida por:

$$simil(termoA, termoB) = 1 - \frac{levenshteinDistance(termoA, termoB)}{max(tamanho(termoA), tamanho(termoB))}$$

Uma abordagem similar à de Bird et al. (Bird et al. 2006) é utilizada por Canfora et al. (Canfora et al. 2011) para analisar as características sociais dos membros da comunidade durante a correção de falhas nos projetos FreeBSD e OpenBSD (Canfora et al. 2011). O trabalho de Canfora et al. utiliza a abordagem de inicias de nomes e sobrenomes de Bird et al., mas não utiliza a similaridade entre nomes e e-mails para evitar a ocorrência de falsos positivos.

Essa heurística é composta por duas estratégias de desambiguação dos autores: a identificação através dos nomes existentes no cabeçalho das mensagens; e a identificação através do endereço de e-mail quando o nome não está presente no cabeçalho da mensagem.

1. A estratégia que utiliza o nome existente no cabeçalho é composta por três casos:
    - O primeiro caso consiste em verificar a igualdade entre um endereço elegível criado a partir do nome existente no cabeçalho da mensagem. Esse endereço é criado a partir da primeira letra do primeiro nome seguido do último sobrenome. Canfora et al. reconhecem que essa característica adiciona a potencial ocorrência de falsos positivos (Canfora et al. 2011). Ela considera ainda apenas o primeiro e último nome. Todos os nomes do meio são desconsiderados e os caracteres especiais e acentos são removidos. Por exemplo, tanto o nome *José Teodoro da Silva* como *João Silva* terão como endereço elegível o conjunto de caracteres *jsilva*.
    - O segundo caso identifica as inicias de todos as partes do nome encontrado no cabeçalho. Se existir apenas um nome em toda a comunidade que possa ser identificado por essas iniciais, a ocorrência dessas iniciais serão consideradas como pertencentes à mesma pessoa e os endereços de e-mail que possuírem apenas essas iniciais no prefixo também serão considerados como pertencentes à essa pessoa. Por exemplo, se existirem *José Teodoro Silva* e *João Teodoro dos Santos* geram as iniciais *jts*. Neste caso, esse passo não pode ser utilizado, contudo se apenas um desses nomes for encontrado, e nenhum outro gerar a sequência de caracteres jts, então toda ocorrência desse prefixo será considerada como pertencendo à mesma pessoa.

- No terceiro caso, as iniciais de todos as partes do nome são concatenadas com a última parte do nome. Se existir apenas um nome capaz de gerar essa cadeia de caracteres, os endereços de e-mail cujos prefixos forem idênticos à essa formação, tanto a pessoa que apresentar esse nome quanto os endereços que apresentarem esse prefixo serão considerados como sendo a mesma pessoa. Por exemplo, se existirem *José Teodoro Silva* e *João Teodoro dos Santos*, o primeiro gera a cadeia *jtsilva*, enquanto o segundo gera *jtsantos*. Se não existirem outros nomes capazes de formar a cadeia jtsilva, então os endereços que possuírem esse prefixo serão considerados como pertencentes à mesma pessoa.

2. A estratégia que utiliza apenas o endereço de e-mail substitui os caracteres especiais (vírgulas, traços e sublinhados) existentes no prefixo do endereço por espaços e constrói um prefixo elegível a partir da inicial da primeira parte concatenada com a última parte extraída do endereço. Ela será considerada como pertencente à mesma pessoa que possuir nomes e endereços que possam ser representados por esse conjunto de caracteres dado o primeiro passo da estratégia que utiliza os nomes. Por exemplo, o endereço *j_t_silva@email.com* gerará um prefixo *jtsilva*. Qualquer nome que gerar esse prefixo e os endereços de e-mail que (por esta estratégia) gerarem esse mesmo nome serão considerados como pertencentes à mesma pessoa.

Robles e Gonzalez-Barahona (Robles & Gonzalez-Barahona 2005) adotam uma abordagem também utilizando nomes. Os autores consideram as possíveis combinações de nome e sobrenome para formação dos endereços de e-mail. Num segundo passo para desambiguação dos autores, eles buscam chaves públicas que identificam um usuário e seus e-mails. Este segundo passo limita o número de projetos para avaliação porque nem todos os projetos possuem a política de uso dessas chaves. A heurística é apresentada em um conjunto de métodos para integração de informações advindas da lista de discussão e do repositório de códigos fontes para exploração e análise de dados da comunidade Gnome.

Utilizamos o repositório de chaves públicas da comunidade como fonte de informações para a avaliação da qualidade das heurísticas, por este motivo consideraremos apenas o segundo passo do método. Esse segundo passo gera uma coleção de endereços elegíveis a partir da permutação das partes do nome encontrado no cabeçalho das mensagens. Todos os endereços que contiverem esses mesmos prefixos serão considerados como pertentes à mesma pessoa. Por exemplo, o nome José Teodoro Silva gera os seguintes prefixos de endereços elegíveis: *jose.teodoro.silva*, *silva.teodoro.jose*, *j.t.silva*, *silva.t.j*, e as demais permutações de iniciais e partes do nome. O método de Robles se difere do método de Bird por utilizar um número arbitrário das partes do nome, ao contrário de Bird que utiliza apenas a primeira e última parte.

Oliva et at. (Oliva et al. 2012) adotam uma abordagem distinta. O autor desta dissertação foi coautor deste trabalho. A heurística agrupa os endereços de e-mail a partir da reincidência do uso do nome do membro com seus possíveis endereços de e-mail no cabeçalho das mensagens da lista.

Eles utilizam essa heurística para explorar a caracterização dos papéis dos desenvolvedores no projeto Apache Ant. Eles partem do pressuposto de que as pessoas utilizam o mesmo nome na configuração de seus clientes de e-mail, apesar de poderem utilizar endereços de e-mail distintos. Esta heurística considera que dois endereços pertencem à mesma pessoa utilizando o nome de usuário (incluindo o domínio) e o nome utilizado na mensagem. Por exemplo, dados os quatro pares de nomes e endereços de e-mail: <José Teodoro Silva, jteodoro@usp.br>, <José, jteodoro@usp.br>, <José Teodoro de Oliveira, jteodoro@hotmail.com> e <José Teodoro de Oliveira, joliveira@meumail.org>, temos que o terceiro e o quarto pares serão igualmente identificados como pertencendo à mesma pessoa devido ao nome idêntico "José Teodoro de Oliveira". O primeiro e segundo pares serão considerados pertencentes à mesma pessoa devido ao mesmo endereço de e-mail "jteodoro@usp.br". Contudo, o primeiro e segundo pares não serão vinculados com os dois últimos porque a heurística considera os domínios do endereço de e-mail.

Goeminne e Mens (Goeminne 2013) avaliou três métodos de desambiguação de autores incluindo a heurística utilizada por Bird et al. e compilou uma heurística melhorada a partir dos resultados encontrados. Tal como a heurística de Bird et al. Goeminne e Mens utilizam distância de Levenshtein para avaliar a similaridade entre duas identidades. Essa heurística utiliza os cabeçalhos das mensagens para identificar os autores das mensagens. Considerando que duas mensagens A e B possuem em seus cabeçalhos um nome e um endereço de e-mail, as duas mensagens serão consideradas como pertencentes à mesma pessoa nos seguintes casos:

- O nome existente no cabeçalho da mensagem A for similar ao prefixo do endereço de e-mail existente no cabeçalho B de acordo com a equação apresentada no método de Bird et al. $simil(nomeA, prefixoEmailB) \geq t$;
- As partes do nome existente no cabeçalho da mensagem A estiver contida no prefixo do endereço de e-mail existente no cabeçalho B;
- O nome existente na mensagem A for similar ao nome existente no cabeçalho da mensagem B de acordo com a equação apresentada no método de Bird et al. $simil(nomeA, nomeB) \geq t$ e cada um destes nomes contiver mais que três caracteres;
- O prefixo do e-mail da mensagem B for idêntica a um dos endereços elegíveis a partir do nome encontrado no cabeçalho da mensagem A. Esses endereços elegíveis são gerados pela permutação das partes existentes no nome da mensagem A.

Kouters (Kouters 2013) propõe o uso de Análise semântica latente (LSA - Latent Semantic Analysis) para desambiguação dos autores. Essa técnica é utilizada para determinar a similaridade entre nomes e endereços de e-mail, identificando assim agrupamentos de endereços que potencialmente pertencem a uma mesma pessoa. Essa heurística é apresentada em sua dissertação de mestrado. Para avaliar seus resultados, Kouters conduziu um estudo de caso no projeto Gnome. Ele compara seus resultados com os de Bird et al. (Bird et al. 2006) e com um algoritmo ingênuo. Entretanto, ele utiliza uma base de referência criada a partir de três inspeções manuais do repositório de códigos fontes do projeto, não incluindo outras fontes de dados possíveis para desambiguação dos autores. A heurística baseada em LSA, como foi documentado pelo próprio

autor nos códigos fonte, não é escalável para grandes volumes de dados. Por este motivo não foi possível utilizá-la em nosso estudo.

A heurística ingênua documentada por Kouters et al. (Kouters et al. 2012) e Goeminne e Mens (Goeminne 2013) considera que dois endereços pertencem à mesma pessoa utilizando o prefixo do endereço de e-mail e o nome utilizado na mensagem. Considerando o mesmo exemplo dos quatro pares de endereços de e-mail e nomes, o algoritmo ingênuo vincula todos esses endereços à mesma pessoa. O primeiro, o segundo e o terceiro pares são vinculados devido ao prefixo "jteodoro" do endereço de e-mail. O terceiro e quarto pares serão vinculados devido ao nome "José Teodoro de Oliveira" em comum.

A **Tabela 1** apresenta uma comparação da desambiguação de autores dados os quatro pares de nomes e endereços de e-mail: <José Teodoro Silva, jteodoro@usp.br>, <José, jteodoro@usp.br>, <José Teodoro de Oliveira, jteodoro@hotmail.com> e <José Teodoro de Oliveira, joliveira@meumail.org>. Os itens em negrito indicam o motivo de cada item ser agrupado para aquele membro.

Heurística	**Primeiro Usuário**	**Segundo Usuário**
Bird	<José Teodoro Silva, **jteodoro**@usp.br>, <José, **jteodoro**@usp.br> e <José Teodoro de Oliveira, **jteodoro**@hotmail.com>	<**José** Teodoro de **Oliveira**, **joliveira**@meumail.org>
Goeminne	<José Teodoro Silva, **jteodoro**@usp.br>, <José, **jteodoro**@usp.br> e <José Teodoro de Oliveira, **jteodoro**@hotmail.com>	<José Teodoro de Oliveira, joliveira@meumail.org>
Robles	<**José Teodoro** Silva, **jteodoro**@usp.br>, <**José Teodoro** de Oliveira, **jteodoro**@hotmail.com> e <**José** Teodoro de **Oliveira**, **joliveira**@meumail.org>	<José, **jteodoro**@usp.br>
Canfora	<José Teodoro Silva, **jteodoro**@usp.br> e <José, **jteodoro**@usp.br>	<**José** Teodoro de **Oliveira**, jteodoro@hotmail.com> e <**José** Teodoro de **Oliveira**, **joliveira**@meumail.org>
Oliva	<José Teodoro Silva, **jteodoro@usp.br**>, <José, **jteodoro@usp.br**>	<**José Teodoro de Oliveira**, jteodoro@hotmail.com> e <**José Teodoro de Oliveira**, joliveira@meumail.org>
Ingênua	<José Teodoro Silva, **jteodoro**@usp.br>, <José, **jteodoro**@usp.br>, <**José Teodoro de Oliveira**, jteodoro@hotmail.com> e <**José Teodoro de Oliveira**, joliveira@meumail.org>	

Tabela 1: Comparativo das desambiguações de cada uma das heurísticas.

Diferentemente das heurísticas mediadas por computador, Guzzi et al. (Guzzi et al. 2013a) realizaram a desambiguação de autores manualmente para um conjunto de 506 discussões da lista, perfazendo um total de 2400 mensagens. Essa abordagem se torna inviável quando o número de mensagens a serem utilizadas é muito grande. Eles a utilizaram para analisar quantitativa e qualitativamente a lista de discussão do projeto Lucene para caracterizar os assuntos tratados na lista e explorar a participação dos membros centrais do projeto nessas discussões. Ela não foi considera em nossa comparação devido ao número de projetos e quantidade de mensagens que estamos utilizando na comparação das heurísticas.

Yin et al. (Yin et al. 2011) utilizam uma abordagem que utiliza tanto as informações do cabeçalho das mensagens quanto o conteúdo da mensagem em si. Eles identificam existência de saudações no corpo do e-mail e extraem o usuário e o endereço de e-mail dessa saudação. Esse endereço é então agrupado juntamente com os endereços existentes no cabeçalho da mensagem. Eles utilizaram esse método para avaliar a frequência e importância de uso de cada um dos múltiplos endereços de e-mail dos usuários da lista de discussão. Para avaliar a eficácia de seu método, eles utilizaram o conjunto de mensagens Enron (Enron Email Dataset 2003).

Poncin et al. apresentaram sua ferramenta FRASR como um framework para análise de repositórios de software (Poncin et al. 2011). Eles constroem vínculos entre os usuários das diversas comunidades através do prefixo do endereço de e-mail e das identificações únicas de cada usuário para as respectivas comunidades. Neste sentido, sua heurística de desambiguação opera do mesmo modo que a heurística ingênua documenta por Kouters et al. (Kouters et al. 2012) e Goeminne e Mens (Goeminne 2013). Contudo, Poncin et al. alertam sobre a ocorrência de falsos positivos e deixa a cargo do usuário do FRASR realizar qualquer desambiguação necessária.

Vasilescu et al. (Vasilescu et al. 2014) analisaram a migração dos usuários de listas de discussão para comunidades como Stack Exchange[3]. Eles investigam o comportamento dos usuários que inicialmente interagem tanto na lista quanto na Stack Exchange e que com o decorrer do tempo optam por utilizar apenas uma das ferramentas. Novamente, Vasilescu et al. alertam sobre a ocorrência de falsos positivos nos métodos existentes. Eles comparam nomes e endereços de e-mail para realizar a desambiguação de autores, mas a descrição do método não é suficientemente clara para a implementação de seu método.

Os trabalhos que definem abordagens para resolução de autores de mensagens nem sempre são claros sobre todos os parâmetros envolvidos na implementação das heurísticas. Comparamos neste trabalho heurísticas para as quais obtivemos acesso ao código fonte ou que possuíam a definição suficientemente clara na literatura para possibilitar a implementação. Para evitar equívocos de implementação, entramos em contato com os autores dos trabalhos para utilizar a mesma implementação criada pelos autores, contudo apenas Bird et al., Oliva et al. e Kouters responderam em tempo hábil.

A implementação enviada por Kouters não é escalável para listas de discussão com vários anos de histórico e os trabalhos publicados por Kouters et al. não deixam claros os passos para se

[3] http://stackexchange.com/

construir uma solução que fosse capaz de escalar para os históricos dos projetos avaliados neste trabalho. Por este motivo, deixamos a comparação com essa heurística como um trabalho futuro.

As heurísticas de Robles e Gonzalez-Barahona e a heurística ingênua possuem uma descrição clara o suficiente para serem implementadas de acordo com a descrição dos trabalhos encontrados na literatura. Por outro lado, as heurísticas de Canfora et al. e Goeminne e Mens são claras, mas possuem pontos passíveis de interpretação. Para estas duas heurísticas construímos uma implementação utilizando os mesmos limites de tolerância utilizados por Bird et al. Para as demais heurísticas, não fomos capazes de construir uma implementação a partir da descrição existente nos trabalhos existentes na literatura.

Devido à dificuldade de conseguir as implementações dos demais trabalhos, deixamos como trabalho futuro a construção das demais heurísticas para garantir que os resultados sejam compatíveis com os idealizados por seus respectivos autores. Assim, nossa comparação se limita aos seis métodos: Bird et al.; Oliva et al.; Canfora et al.; Goeminne e Mens; Robles et al. e uma heurística ingênua documentada por Goeminne e Mens e por Kouters et al.

A Tabela 2 apresenta as principais características encontradas em cada heurística a serem avaliadas.

Heurística	Utiliza similaridade	Infere prefixo a partir do nome	Infere nome a partir do prefixo	Considera todas as partes do nome	Considera os domínios dos endereços de e-mail	Considera a reincidência dos pares de nome e endereço de e-mail
Bird et al.	X	X				
Canfora et al.		X	X	X		
Robles e Gonzalez-Barahona		X		X		
Oliva et al.				X	X	X
Goeminne e Mens	X	X		X		
Ingênua						X

Tabela 2: Principais características das heurísticas avaliadas neste trabalho.

2.4 Usos da desambiguação de autores

Dentre os trabalhos que utilizam as heurísticas para desambiguação de autores de listas de discussão, podemos citar Panichella et al. (Panichella, Bavota, et al. 2014), que construíram redes sociais a partir das interações extraídas de várias ferramentas de comunicação (listas de discussão, fóruns do projeto e gerenciador de tarefas). Eles utilizaram a abordagem proposta por Bird com algumas modificações na tentativa de reduzir os falsos positivos. Tal abordagem também foi utilizada por Xuan e Filkov (Xuan & Filkov 2014), construindo uma variação própria daquela

proposta por Bird et al. (Bird et al. 2006) para resolver a identidade dos autores dos e-mails da lista de discussão. Eles analisam métodos quantitativos para determinação de ocorrência de atividades sincronizadas na comunidade e para explorar a produtividade e comunicação dos membros do projeto.

Rigby et al. (Rigby et al. 2008) se ativeram à implementação original de Bird et al. (Bird et al. 2006) e utilizaram a mesma ferramenta criada por Bird para identificar os autores em seu estudo sobre as práticas de revisão de código no projeto httpd Apache Server. Essa mesma ferramenta foi utilizada por Nia et al. (Nia et al. 2010a) para examinar a estabilidade das métricas de redes sociais quando expostas a dados ruidosos e esparsos advindos de listas de discussão. Essa ferramenta ainda foi utilizada por Bird et al. (Bird et al. 2008) para analisar a estrutura complexa de comunidades de software livre e explorar o modo como essas comunidades se auto organizam.

2.5 Avaliação das heurísticas na literatura

Dentre os trabalhos que realizam a avaliação dos resultados das heurísticas propostas, podemos citar o trabalho de Goeminne e Mens (Goeminne 2013). Eles avaliaram os resultados de algoritmos de desambiguação de autores para quatro heurísticas: um algoritmo ingênuo de identificação; Bird; Robles e uma versão melhorada de Bird incluindo partes do método de Robles. Essa avaliação foi realizada para três projetos de software livre: Evince, Brasero e Subversion. Contudo, a construção de uma base de referência é limitada e realizada mediante trabalho manual. Essa base também não considera informações advindas dos sites dos projetos e do gerenciador de tarefas, que possuem informações oficiais do projeto sobre os membros.

Com exceção dos trabalhos de Kouters (Kouters 2013) e de Goeminne e Mens (Goeminne 2013), os pesquisadores não possuem comparação ou avaliação das heurísticas existentes na literatura de desambiguação de autores. Entretanto, os trabalhos de Kouters e de Goeminne e Mens utilizam poucos projetos e não exploram as consequências que a mudança no tamanho do conjunto de dados pode causar.

Neste contexto, esta pesquisa apresenta uma ampliação do trabalho realizado por Kouters e Goeminne e Mens na comparação entre os resultados das heurísticas para desambiguação de autores nas listas de discussão. Medimos também a relação existente entre a qualidade dos resultados, o tamanho da comunidade e o período utilizado em cada heurística. Apresentamos estudo de caso em que o problema da desambiguação de autoria pode ocorrer e o impacto nos resultados gerados a partir dos dados recuperados de listas de discussão.

2.6 Exemplos de uso das heurísticas na literatura

As falhas na desambiguação de autores podem afetar os resultados de estudos que utilizam a comunicação existente nas listas de discussão. A consequência mais direta está relacionada aos trabalhos que utilizam as listas de discussão para construção de redes sociais. Essas redes são construídas utilizando modelagem de grafos em que os nós são os membros com seus múltiplos endereços de e-mail e as arestas são as mensagens que esses membros enviam. Falhas na

identificação desses múltiplos endereços de e-mail, por exemplo, agrupam mensagens de diferentes membros a um participante da lista. Esse tipo de erro afeta a construção de arestas da rede social que acaba por possuir mais comunicação do que esses membros possuem. Na literatura encontramos trabalhos que podem ser influenciados pelas falhas na desambiguação de autores nas listas. A seguir citamos alguns dos casos em que essas falham podem influenciar os resultados das análises:

Oliva et al. (Oliva et al. 2012) caracterizam os desenvolvedores centrais do projeto Apache Ant. Eles utilizam a rede social construída a partir da lista de discussão. Foram caracterizados cerca de 25% dos membros como sendo participantes deste grupo de desenvolvedores centrais do projeto. Falhas na desambiguação dos autores podem atribuir conexões erradas na rede social que foi construída, influenciando os resultados desta caracterização.

Vasilescu et al. (Vasilescu et al. 2014) analisaram o uso de duas redes de comunicação pelo mesmo grupo de usuários: a lista de discussão r-help e a comunidade de perguntas e respostas Stack Exchange. Eles encontraram indícios de que com o passar do tempo os usuários se mantém em apenas uma das redes. Para avaliar essa migração de usuários de uma rede de comunicação para outra, foram inferidos os vínculos entre os diferentes usuários dessas duas redes. Falhas na desambiguação dos membros podem inferir vínculos erroneamente entre usuários que não são necessariamente a mesma pessoa. Essas falhas podem influenciar os resultados da avaliação dessa migração de usuários.

Canfora et al. (Canfora et al. 2011) avaliaram as interações sociais entre os sistemas de correção de erros do OpenBSD e FreeBSD. Estes dois sistemas não possuem um gerenciador de falhas específico e utilizam a lista de discussão para que os desenvolvedores reportem e discutam os erros do software e parte do código do primeiro software deu origem ao segundo. Canfora et al. encontraram indícios de que os usuários envolvidos na discussão dos erros desempenham um papel importante na disseminação de conhecimento em ambas as listas de discussão e realizam a ponte entre as comunidades das listas de discussão. Erros na desambiguação dos autores afetam a identificação destes membros que realizam as pontes e atribuir mensagens incorretamente, incluindo arestas inexistentes na rede social de comunicação que foi gerada a partir dessas listas.

Goeminne (Goeminne 2013) avaliou como as comunidades de software crescem e evoluem com o tempo em diferentes vias de comunicação. Ele explorou a comunicação dos membros e como a rede social se desenvolveu no projeto GNOME. Ele encontrou indícios de que a comunidade GNOME é dividida em sub-comunidades e que os membros utilizam essas sub-comunidades de formas diferentes. Além disso, ele reportou que os membros utilizam múltiplos endereços de e-mail para se comunicar. As falhas na desambiguação dos autores podem influenciar a construção dessa rede social que representa as comunidades e dificultar a identificação das diferentes formas que um mesmo usuário se comporta uma vez que pode atribuir mensagens que não pertencem aquele usuário.

Bernardi et al. (Bernardi et al. 2012) exploraram a comunicação das listas de discussão para avaliar se os membros da comunidade que adicionar falhas no software se comunicam pouco nas listas de discussão. Eles encontraram indícios de que esses membros que adicionaram falhas

possuem grande importância social e se comunicam muito com os demais membros, apesar de se comunicar pouco entre si. A falha na desambiguação dos autores pode afetar a identificação da comunicação e verificação da importância dos membros, uma vez que a relevância dos membros é avaliada através das arestas de comunicação existentes na rede social.

Abreu e Premraj (Abreu & Premraj 2009) estudaram como a comunicação se relaciona com a inclusão de falhas no código fonte do projeto JDT da Fundação Eclipse. Eles encontraram indícios de que os membros centrais da comunidade se comunicam mais quando existem mais falhas no software. Esses desenvolvedores centrais foram identificados utilizando métricas de rede social construída a partir da lista de discussão. Isso implica que falhas na desambiguação dos autores podem influenciar a identificação desse grupo de desenvolvedores centrais no sentido de que pode atribuir a um membro mais comunicação do que ele realmente realizou.

Outros exemplos de trabalhos que podem ser impactados pelos erros de desambiguação de autores são os trabalhos de Bosu e Carver (Bosu & Carver 2014), Panichella et al. (Panichella, Canfora, et al. 2014), Bird e Nagappan (Bird & Nagappan 2012), Bird et al. (Bird et al. 2008), (Bird et al. 2007), Neu et al. (Neu et al. 2011) e Canfora et al. (Canfora et al. 2012).

3 Método

O design da pesquisa é apresentado na Figura 1 e está discutido nas subseções a seguir. Os passos do método podem ser sumarizados em:

- Conceber uma base de referência para avaliação dos resultados de cada heurística;
- Comparar os resultados gerados com as informações existentes na base de referência;
- Analisar a relação existente entre os erros de identificação e o tamanho da comunidade;
- Analisar a relação entre o período de tempo utilizado e os erros de identificação.

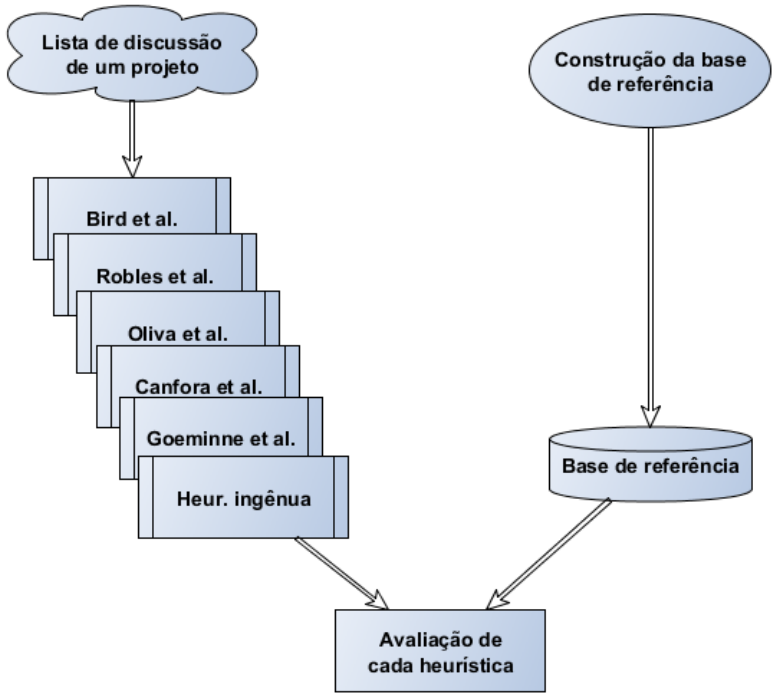


Figura 1: Design da pesquisa

A avaliação das heurísticas de desambiguação de autores não pode contar unicamente com as informações advindas da lista, uma vez que estamos tentando justamente validar os resultados de heurísticas que utilizem apenas as informações da lista na identificação de usuários. Precisamos de outras fontes de dados que possibilitem a construção de uma base de referência para a avaliação das heurísticas. Escolhemos projetos que disponibilizam outras fontes de informações para que possamos construir essa base para a avaliação das heurísticas.

3.1 Fonte de dados

A comunidade Apache foi escolhida pela diversidade de projetos que possui e por ser frequentemente estudada empiricamente (Bird et al. 2006; Oliva et al. 2012; Steinmacher et al.

2012). Além disso, a comunidade está dispersa por vários países e conta com ajuda de profissionais de diferentes empresas. A escolha dos projetos foi realizada mediante a disponibilidade das informações para construção de uma base de referência para avaliar os resultados das heurísticas.

A Fundação Apache mantém o histórico das listas de discussão de seus projetos em sua página[4]. Utilizamos os históricos das mensagens de 150 listas de desenvolvedores de projetos da Fundação Apache. Foram recuperados o histórico das mensagens até maio de 2013. Esses dados totalizam aproximadamente 3,85 milhões de mensagens e mais de 315 mil endereços de e-mail.

Assim como muitas comunidades de software livre, o histórico das listas da Fundação Apache são armazenados no formato mbox (Robles et al. 2009). Esse formato contém os cabeçalhos completos dos e-mails enviados. As heurísticas utilizam essas informações para identificar os vários endereços de e-mail utilizados por um membro da comunidade.

Tal como Squire (Squire 2013), utilizamos várias fontes de dados para recuperar informações e criar um conjunto de dados contendo os endereços de e-mail dos membros da comunidade. Porém, utilizamos apenas os dados existentes nos perfis de usuários encontrados no gerenciador de tarefas dos projetos (Jira) e os sites dos projetos que continham informações sobre os membros. Essas fontes foram utilizadas para a criação de uma base contendo os endereços de e-mail e nomes dos membros.

Os resultados das heurísticas podem ser expressos por agrupamentos de endereços de e-mail e nomes de usuários. Utilizamos a base de referência contendo nomes e endereços formalmente atribuídos para avaliar a capacidade das heurísticas de realizar esse agrupamento corretamente.

3.2 Construção da base de referência de endereços

As ferramentas utilizadas pelos membros da comunidade podem ser utilizadas como fonte de dados confirmatórios sobre as identidades dos membros da lista de discussão. Ferramentas como gerenciador de versões, sites do projeto e gerenciador de tarefas (Jira) contêm os endereços e nomes dos participantes. O gerenciador de tarefas (Jira) é uma ferramenta para auxiliar no desenvolvimento e organização de projetos de software. A ferramenta possui um perfil para cada usuário, em que encontramos um endereço de e-mail, o nome e uma identificação única do usuário na comunidade. Por sua vez, os sites dos projetos contêm informações sobre o software e sua equipe de desenvolvimento. Alguns sites disponibilizam os nomes, identificações únicas e endereços de e-mail dos membros da equipe do software. Por fim, o repositório de chaves públicas da comunidade contém uma assinatura digital acompanhada dos endereços de e-mail assinados pelo próprio membro da comunidade. Recuperamos todas essas informações que estavam disponíveis para os projetos analisados e compilamos nossa base para conduzir as avaliações.

Neste trabalho utilizamos quatro processos distintos para a construção da base de referência de identidades dos membros: um conjunto de dados criado a partir do site da comunidade por Squire et al. (Squire 2013); o conjunto de dados inspecionados manualmente por nosso grupo de pesquisa a partir dos sites dos projetos e utilizado no experimento aceito para publicação no SBSC

[4] http://mail-archives.apache.org/

2015 (Da Silva et al. 2015); a recuperação das informações existentes no repositório de chaves públicas da comunidade Apache; e a recuperação de informações existentes na ferramenta Jira da comunidade Apache.

Para utilizar as informações existentes nos sites da comunidade, fizemos uso do conjunto de dados publicado por Squire et al. (Squire 2013). Eles construíram um conjunto de dados utilizando scripts para recuperação e interpretação das páginas e arquivos existentes no site da comunidade Apache. Essas páginas contêm as informações de todos os membros da comunidade. Essas páginas são estáticas, foram construídas e são mantidas manualmente pelos membros. O conjunto de dados resultante possui identificações dos membros na comunidade, seus respectivos nomes e algumas dessas identificações possuem endereços de e-mail. Utilizamos apenas as identidades que possuem um endereço de e-mail que não pertença ao domínio da própria comunidade, uma vez que podemos inferir esse endereço a partir da identificação do usuário. Do conjunto de dados publicado por Squire et al. conseguimos utilizar 507 endereços de e-mail agrupados em 476 identidades de membros.

Para utilizar os dados existentes nos sites dos projetos, utilizamos uma base construída por nosso grupo de pesquisa durante a realização de nosso experimento (Da Silva et al. 2015). Na ocasião, construímos uma base de dados a partir da inspeção manual das páginas de 16 projetos da comunidade Apache e ao contrário de Squire et al., que utilizaram a página dos usuários da comunidade. Optamos por utilizar os sites dos projetos partindo do racional de que os sites dos projetos possuem manutenção mais frequente do que as páginas de documentação sobre os usuários da comunidade. Essa nossa inspeção resultou em 720 endereços de e-mail agrupados em 309 usuários. Como apresentado na Figura 3, apesar de existirem intersecções com as informações encontradas por Squire et al., localizamos endereços de e-mail que só foram encontrados inspecionando manualmente os sites dos próprios projetos.

A Fundação Apache possibilita o acesso ao repositório de chaves públicas da comunidade[5]. Neste repositório é possível obter os arquivos da chave de cada um dos membros da comunidade. Esses arquivos contêm informações sobre cada membro e um conjunto de endereços de e-mail reconhecidos pelo proprietário da chave pública. A partir destes arquivos extraímos as identidades e o conjunto de endereço de e-mail de cada membro. Essa coleta resultou em 1.358 endereços de e-mails agrupados em 722 membros da comunidade.

Por fim, para recuperar as informações do gerenciador de versões e da ferramenta Jira, criamos scripts que realizaram a recuperação automatizada das informações dos membros dessas duas ferramentas. Primeiramente, listamos as identidades utilizadas para submeter códigos para todos os repositórios. A partir dessa listagem, construímos os endereços de e-mail pertencentes à comunidade Apache a partir da identificação única de cada membro seguida do sufixo "@apache.org". Ainda a partir da identificação única de cada usuário, solicitamos à ferramenta Jira que fornecesse o endereço de e-mail registrado para a respectiva identificação. Essa ferramenta só possibilita que um endereço de e-mail seja cadastrado para cada membro, por este motivo esse processo consegue fornecer no máximo dois endereços de e-mail de um mesmo indivíduo.

[5] https://people.apache.org/keys/committer/

Em todo o repositório da comunidade Apache, localizamos 2.850 identidades para serem verificadas junto à ferramenta Jira. Essa requisição junto à ferramenta foi realizada com sucesso para 1.992 identificações resultando num total de 3.984 endereços de e-mail. Alguns endereços de e-mail registrados na ferramenta Jira eram os próprios endereços da comunidade Apache. Isso ocorreu em 523 casos. Por este motivo, apesar de verificarmos 1.992 identificações - com possíveis dois endereços de e-mail para cada uma - fomos capazes de recuperar no total 3.461 endereços de e-mail utilizando esse método.

Por fim, compilamos uma base de referência utilizando os endereços de e-mail de todas as essas fontes. A Figura 2 ilustra o processo de fusão das identidades das várias fontes. Para mesclar as diferentes identidades encontradas em cada uma das fontes de dados, utilizamos a identificação única dos membros da comunidade Apache. Cada usuário possui uma identificação única no domínio apache.org que possibilita a união das informações de um mesmo membro advindas de quaisquer dos quatro processos supracitados. Após essa junção das identidades, nossa base de referência apresentou 3.672 endereços de e-mail agrupados em 1.639 identidades. A Tabela 3 apresenta a estatística descritiva de nossa base de referência final.

Número total de endereços de e-mail	3.672
Média de endereços de e-mail por membro	2,24
Mediana de endereços de e-mail por membro	2
Número mínimo de endereços de e-mail distintos por membro	2
Número máximo de endereços de e-mail distintos por membro	15
Desvio padrão de endereços de e-mail por membro	0,95
Endereços com prefixo de até sete caracteres	54,00%
Endereços com prefixo de até seis caracteres	39,86%
Endereços com prefixo de até cinco caracteres	24,56%
Desvio padrão do tamanho dos endereços de e-mail	3,49
Média do tamanho dos endereços de e-mail	7,87
Mediana do tamanho dos endereços de e-mail	7
Tamanho mínimo dos endereços de e-mail	1
Tamanho máximo dos endereços de e-mail	25

Tabela 3: Estatística descritiva da base de referência.

Uma vez que utilizamos o repositório de códigos fontes e a ferramenta Jira para determinar os endereços de e-mail dos membros com permissão de escrita, os membros que são identificados

possuem apenas dois endereços de e-mail: o primeiro pertence à comunidade Apache que é equivalente à identificação única do membro seguida do domínio apache.org; e o segundo é um endereço pessoal ou profissional registrado pelo usuário na ferramenta Jira. Devido à quantidade de membros que foram recuperados utilizando esse método ser quase um terço do total de endereços da comunidade, é possível que essa fonte de dados influencie os resultados sobre o número de endereços de e-mail que os membros da comunidade possuem em média. Por este motivo, adicionamos Tabela 4 que foi gerada a partir das demais fontes de dados, excluindo as informações advindas do Jira para avaliar melhor a média de endereços de e-mail por membro.

Como podemos observar na Tabela 4, a mediana do número de endereços de e-mail por membro permanece dois e a média aumenta muito pouco. Isso apresenta indícios de que em geral os membros da comunidade utilizam de dois a três endereços de e-mail. Contudo, estamos cientes de que nosso processo de geração da base de referência pode favorecer os usuários com mais de um endereço de e-mail.

Número total de endereços de e-mail	1406
Média de endereços de e-mail por membro	2,70
Mediana de endereços de e-mail por membro	2
Desvio padrão de endereços de e-mail por membro	1,52

Tabela 4: Estatística descritiva dos endereços de e-mail excluindo as informações advindas da ferramenta Jira.

A Figura 2 ilustra como foi realizada a mescla dos endereços de cada uma das fontes de dados. Podemos observar no exemplo que o usuário 1 recebe endereços de e-mail apenas do repositório de chaves públicas, enquanto que o usuário 2 recebe informações advindas do servidor de chaves públicas, da base de dados compilada por Squire et al. e da base que compilamos para o artigo do SBSC. Em contrapartida, o usuário 3 recebe informações de todas as fontes: Jira, repositório de chaves públicas e Squire et al.

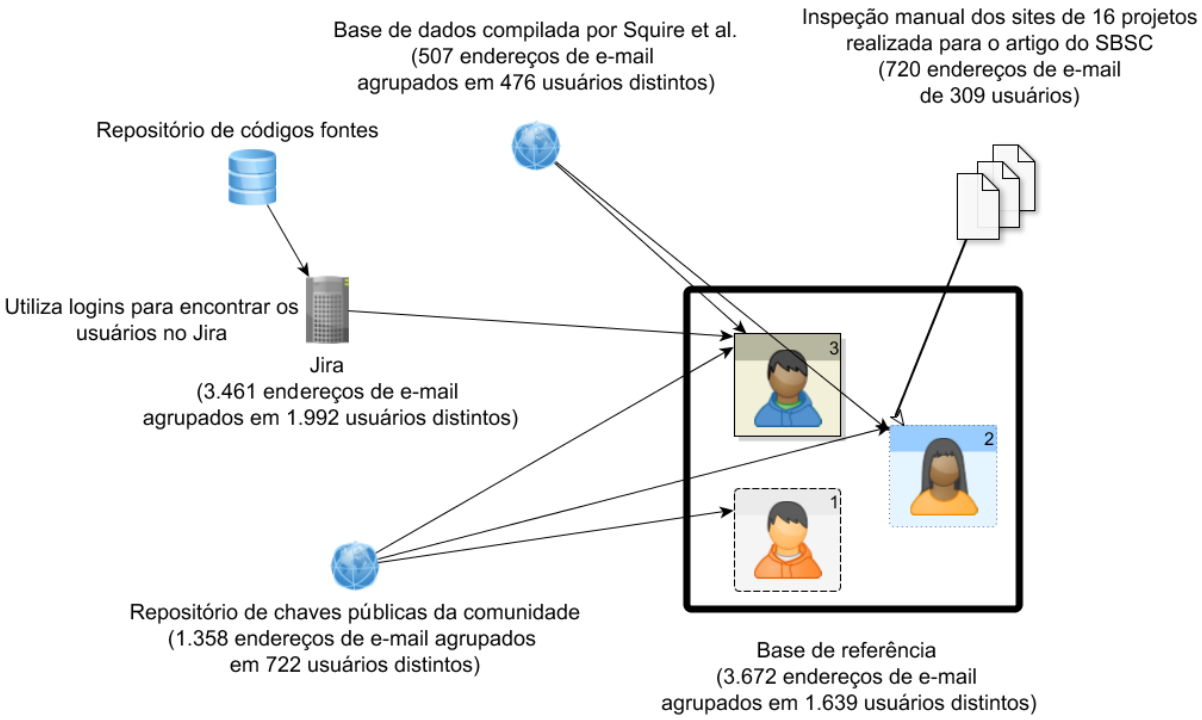


Figura 2: Construção da base de referência utilizando várias fontes de informações.

A mescla das diferentes fontes de dados apresentou intersecções entre os endereços de e-mail advindos das três fontes utilizadas. A Figura 3 apresenta as intersecções dos endereços de e-mails encontrados em cada fonte de dados. Podemos observar que 192 endereços de e-mail foram identificados apenas pela inspeção assistida realizada por Squire et al. e que outros 27 endereços de e-mail só foram encontrados mediante a inspeção manual dos sites dos projetos. Estes são indícios de que a abordagem automatizada de geração da base de referência não consegue identificar todos os endereços de e-mail dos membros da comunidade. Porém, notamos que os endereços que só foram identificados manualmente representam apenas 4% do total dos endereços da base de referência. Observamos ainda que na intersecção de todas as fontes utilizadas para a construção da base de referência existem apenas 35 endereços.

A baixa quantidade de endereços existentes nas intersecções de todas as fontes é um indício de que é necessário utilizar várias fontes de dados para verificar as identidades dos membros da comunidade a ser analisada. Podemos observar que as várias fontes de dados se completam para formar um conjunto mais amplo dos endereços de e-mail dos membros e que 96% desse conjunto pode ser obtido por scripts de mineração de dados automatizados.

Apesar de nossa base de referência possuir apenas endereços de e-mail oficiais dos membros da comunidade, não podemos afirmar que todos os endereços dos membros da comunidade estão inclusos nesta base. Os conjuntos dos endereços que recuperamos do Jira e do repositório de chaves públicas não garantem que sabemos todos os endereços que os membros utilizaram para enviar e-mail para a lista. Além disso, apenas os usuários da apache com permissão de escrita no repositório de códigos fontes possuem identidades no repositório de chaves públicas da comunidade. Neste sentido, nossa base é precisa, porém incompleta.

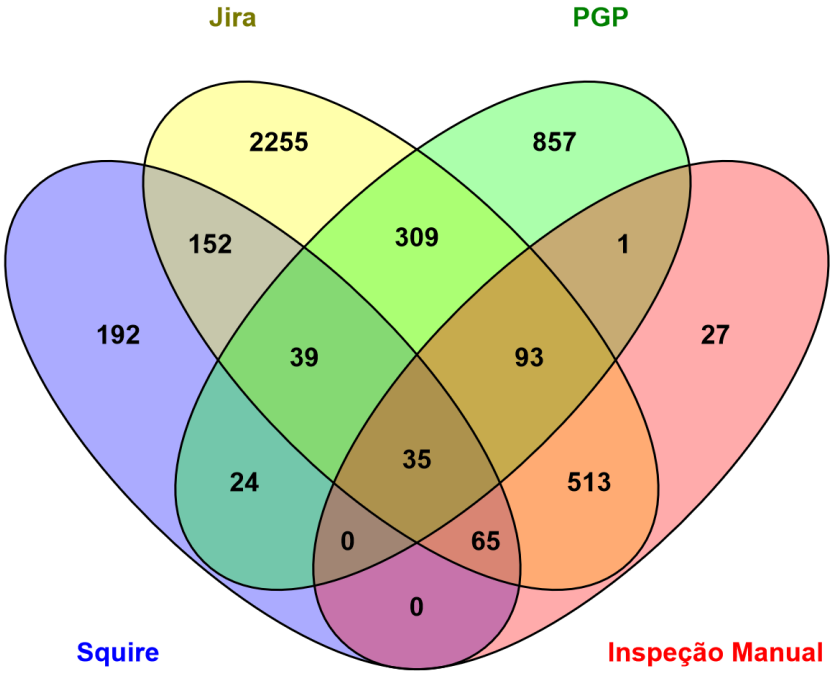


Figura 3: Diagrama da intersecção das fontes de dados.

3.3 Avaliação das heurísticas

Para avaliar cada heurística, comparamos os conjuntos de endereços de e-mail identificados pela heurística com os endereços de e-mail existentes na nossa base de referência. Essa comparação foi realizada mediante verificação dos conjuntos de endereços de e-mail do resultado de uma heurística com o conjunto de endereços de e-mail existentes na base de referência.

Devido à incompletude de nossa base, não pudemos avaliar todos os conjuntos de endereços que as heurísticas identificaram nas listas de discussão e comparamos apenas os endereços que estão presentes tanto em nossa base de referência quanto nas listas de discussão. Figura 3 apresenta as proporções entre o número de endereços da base de referência e os endereços de e-mail encontrados na lista de discussão de cada projeto que foram comparados para cada intervalo de tempo. Os valores existentes nessa tabela representam o tamanho do conjunto de endereços de e-mail que conseguimos avaliar mediante nossa base de referência. Os demais endereços fazem parte

do conjunto não-comparável sobre o qual não possuímos informação suficiente para julgar se o agrupamento foi realizado corretamente.

Para os intervalos de até 48 meses conseguimos comparar até um quarto do total dos endereços de cada lista de discussão, com um desvio padrão de aproximadamente 16%. Contudo, quando utilizamos todo o histórico das listas de discussão a proporção de endereços passíveis de comparação caiu para 12,57% com desvio padrão de 8,18%.

Intervalo de tempo	Média	Desvio Padrão
3 meses	25,59%	16,16
6 meses	25,88%	16,56
12 meses	25,69%	16,66
24 meses	25,12%	16,19
36 meses	26,53%	16,03
48 meses	27,91%	16,83
Todo o histórico	12,57%	8,18

Tabela 5: Proporções entre os endereços comparados e o total de endereços da lista de discussão.

Para analisar os resultados de cada heurística, utilizamos a medida de reconhecimento de padrões e de recuperação de informações Medida F. Essa medida é calculada a partir das medidas de Precisão e Sensibilidade. As medidas de Precisão e Sensibilidade foram utilizadas por Kouters (Kouters 2013) e por Goeminne e Mens (Goeminne 2013) para avalição das heurísticas de desambiguação de autores. Essas duas medidas possibilitam a avalição dos acertos e erros das heurísticas em relação às informações existentes na base de referência e sua representatividade em relação ao número total de endereços de e-mail encontrados em cada lista de discussão. A precisão é a proporção dos endereços de e-mail que foram corretamente identificados de acordo com nosso conjunto verdade. Por outro lado, a medida de sensibilidade avalia a proporção dos endereços que identificamos como pertencentes a um membro da comunidade e que realmente pertencem a esse membro, de acordo com a base de referência. Um índice baixo de precisão significa que a heurística está atribuindo endereços de e-mail para os membros da comunidade que não os pertencem. Um índice baixo de sensibilidade significa que a heurística está errando por omissão e está deixando de atribuir endereços que notoriamente pertencem a um membro da comunidade.

A Medida F consolida as medidas de Precisão e Sensibilidade, por este motivo estamos utilizando-a para avaliar os resultados das heurísticas neste trabalho. Deste modo, consideramos numa mesma medida a eficácia na desambiguação dos autores a partir da ocorrência de falsos positivos e falsos negativos.

Para calcular essas medidas, utilizamos uma matriz de confusão que possibilita a avaliação da qualidade dos resultados das heurísticas.

	Positivo (conhecido)	**Negativo (conhecido)**
Positivo (resultado da heurística)	Verdadeiro Positivo (vp)	Falso Positivo (fp)
Negativo (resultado da heurística)	Falso Negativo (fn)	Verdadeiro Negativo (vn)

Tabela 6: Matriz de confusão.

Neste trabalho, conforme Tabela 6, temos:

- O número de verdadeiros positivos (vp): endereços que a heurística identificou como pertencendo a uma mesma pessoa e que estão de acordo com os endereços existentes na base de referência;
- O número de falsos negativos (fn): endereços de e-mail que a heurística não atribuiu corretamente à pessoa que o possui na base de referência;
- O número de verdadeiros negativos (vn): endereços que a heurística identificou corretamente como não pertencentes à mesma pessoa e que estão de acordo com a base de referência;
- O número de falsos positivos (fp): endereços de e-mail que a heurística atribuiu incorretamente a uma pessoa, sendo que este endereço pertence a outra pessoa na base de referência.

A partir da matriz de confusão, calculamos a Precisão e a Sensibilidade utilizando as seguintes equações:

$$Precisão = \frac{\sum vp}{\sum vp + fp} \qquad Sensibilidade = \frac{\sum vp}{\sum vp + fn}$$

Consideramos a precisão e a sensibilidade simultaneamente para determinar a qualidade dos resultados das heurísticas. Para isso, utilizamos a Medida F que proporciona uma média harmônica entre precisão e sensibilidade. Essa medida possibilita a análise de perdas e ganhos existentes nos resultados gerados pelas heurísticas. Para calcular a Medida F utilizamos a seguinte equação:

$$Medida\ F = 2 * \frac{Precisão * Sensibilidade}{Precisão + Sensibilidade}$$

Com essas medidas, realizamos duas avaliações dos resultados das heurísticas, a primeira utiliza os dados de todo o histórico das listas de discussão. A segunda utiliza este mesmo histórico fracionado em períodos de 3 meses, 6 meses, 12 meses, 24 meses, 36 meses, 48 meses e todo o histórico da lista de discussão. Avaliamos esses diferentes períodos para não favorecer a heurística de Oliva et al. (Oliva et al. 2012) que faz uso da reincidência do nome do remetente da mensagem existente na configuração de seus clientes de e-mail e para avaliar como o tamanho do período utilizado pode afetar na qualidade dos resultados das heurísticas.

3.4 Comparação da heurística em relação à base de referência

Para cada endereço de e-mail existente no resultado da heurística, verificamos se o mesmo existe na base de referência. Se existir, verificamos se o agrupamento de endereços que o contém está coerente com o agrupamento existente na base de referência. A Figura 4 ilustra a comparação dos agrupamentos de endereços gerados pela heurística contra os agrupamentos existentes na base de referência. Cada agrupamento de endereços de e-mail representa uma identidade de um usuário e seus múltiplos endereços de e-mail. Lembremos que nossa base de referência é incompleta e, portanto, nem todos os agrupamentos de endereços podem ser verificados. Em nosso exemplo, só seria possível comparar os endereços de e-mail das identidades #1, #4 e #7 porque não temos informações sobre as demais identidades em nossa base de referência.

Quando comparamos duas entidades sempre consideramos como ponto de partida o endereço de e-mail da comunidade Apache. Isso é possível porque nossa base de referência foi construída utilizando as identificações únicas da comunidade, que uma vez concatenadas com o sufixo "@apache.org" constituem o endereço de e-mail daquele membro na comunidade. Para cada identidade gerada pela heurística, identificamos o endereço de e-mail com o sufixo "@apache.org" e localizamos na base de referência a identidade que possui esse endereço. Em nosso exemplo, percebemos que a identidade #7A possui um endereço da comunidade e buscamos na base de referência pela identidade que possua esse mesmo endereço (#7B).

Quando encontramos essa segunda identidade iniciamos a comparação dos endereços de e-mail. Para cada endereço de e-mail de #7A verificamos se o mesmo existe em #7B. Todos os endereços que forem encontrados em ambas as identidades são marcados como sendo VP (verdadeiros positivos). Em nosso exemplo, VP = 3.

Continuando a comparação, avaliamos se os endereços existentes em #7A e que não existem em #7B podem ser encontrados na base de referência. Esses endereços são marcados como FP

(falso positivo). Em nosso exemplo, o endereço adalberto@hotmail.com pode ser encontrado na identidade #6B da base de referência, portanto FP = 1.

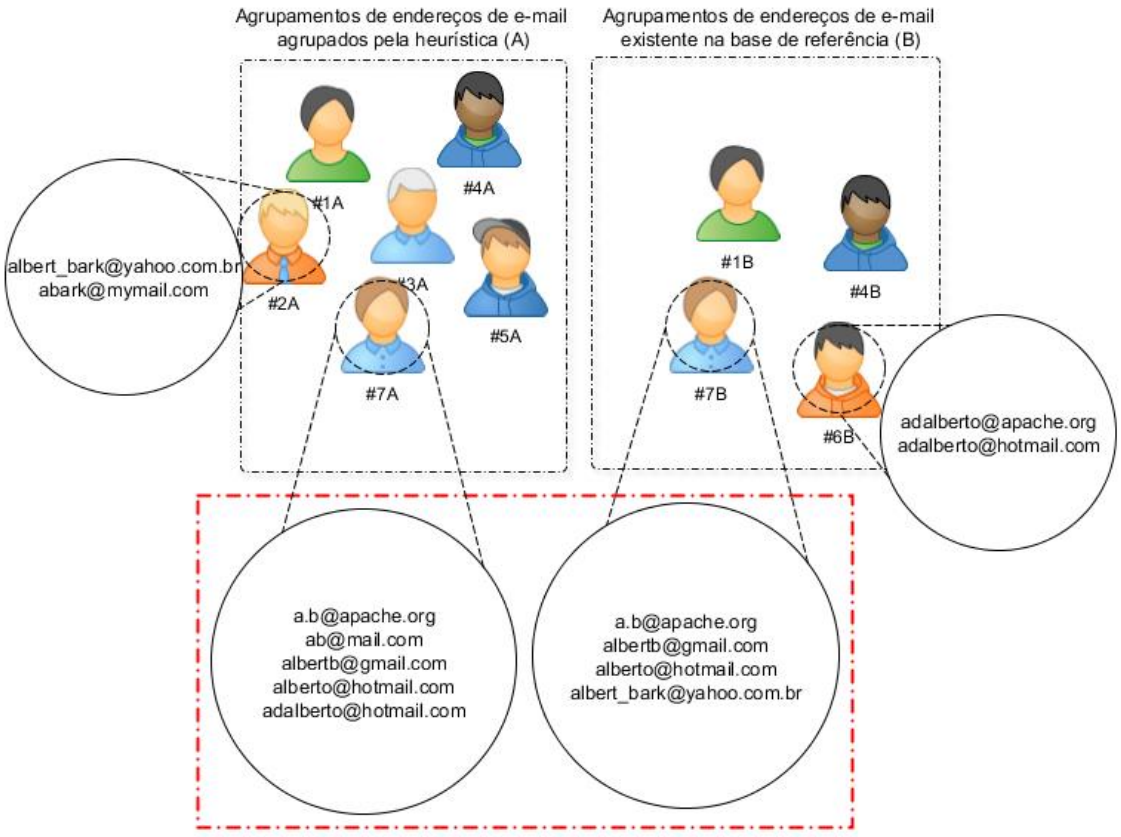


Figura 4: Comparação do resultado da heurística com a base de referência.

Para determinar os falsos negativos (FN) realizamos a busca por todos os endereços de e-mail existentes em #7B e verificamos se algum desses endereços existe em algum outro agrupamento da heurística. Em nosso exemplo, observamos que o endereço albert_bark@yahoo.com.br não foi agrupado com #7A. E sim foi agrupado na identidade #2A. Então em nosso exemplo, FN = 1.

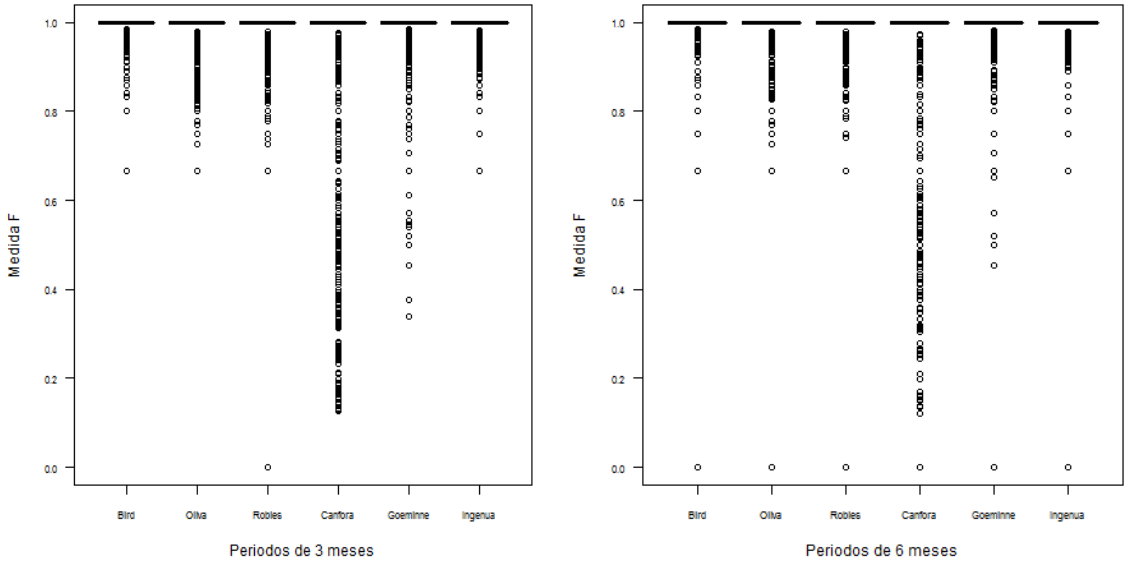
Para determinar os VN (verdadeiros negativos), recuperamos todos os endereços de e-mail existentes na base de referência, exceto aqueles encontrados em #7B. Para cada um destes endereços, verificamos quantos podem ser encontrados nos agrupamentos gerados pela heurística e que não estejam presentes em #7A. Essa medida tem por objetivo identificar os endereços que foram corretamente atribuídos como não pertencentes à #7A.

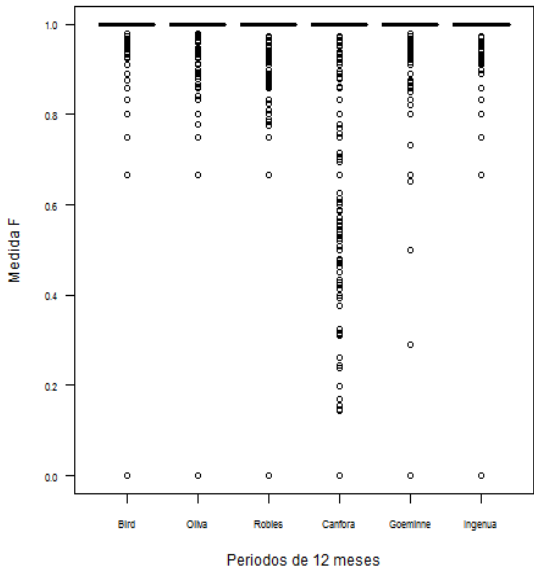
4 Resultados da comparação das heurísticas

Conforme descrito na seção anterior, avaliamos 6 heurísticas de desambiguação de autores em 150 listas de discussão de desenvolvedores de projetos de software livre da Fundação Apache. Esta seção apresenta os resultados obtidos a partir da avaliação dessas heurísticas. As subseções a seguir apresentam a avaliação da relação entre o intervalo de tempo e a eficácia das heurísticas (Seção 4.1), a avaliação da relação entre o tamanho da comunidade e as falhas nas heurísticas (Seção 4.2) e a comparação da eficácia de cada heurística em diferentes intervalos de tempo (Seção 4.3).

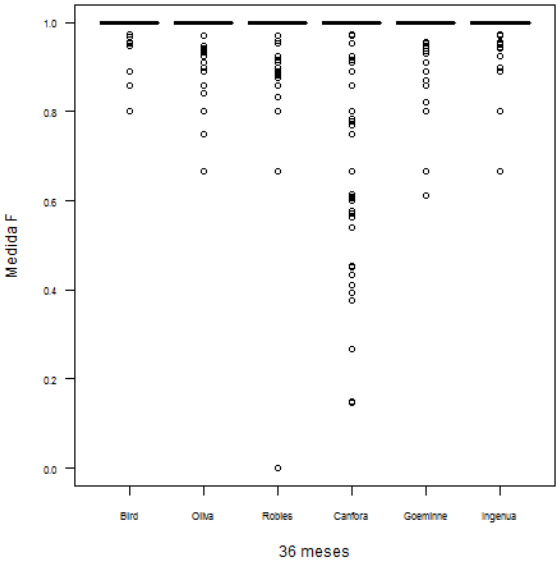
4.1 O intervalo de tempo importa?

Avaliamos a qualidade dos resultados das heurísticas mediante diferentes períodos de tempo (3, 6, 12, 24, 36, 48 meses e todo o histórico da lista de discussão). O objetivo desta análise é verificar se o período de tempo influencia a qualidade dos resultados das heurísticas. Para tal, construímos os gráficos das Medidas F de cada heurística para cada um dos períodos. A Figura 5 apresenta esses gráficos.

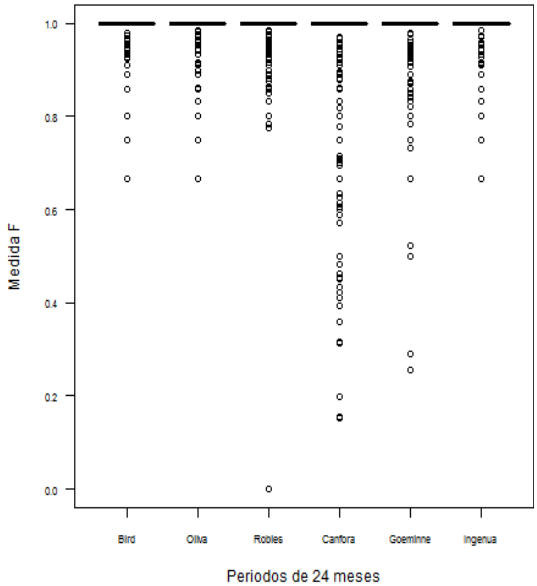




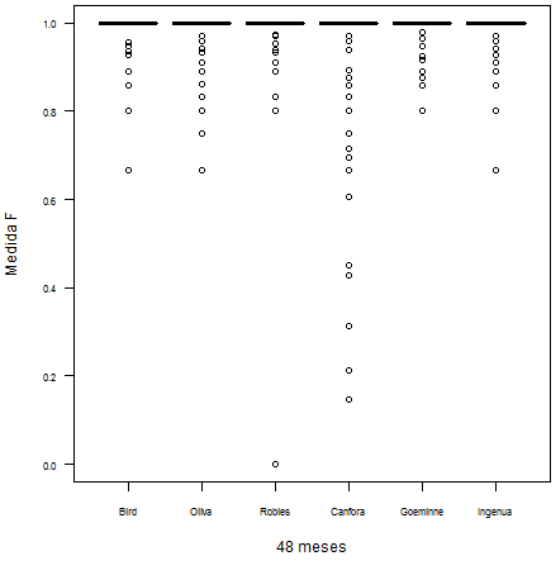
Periodos de 12 meses



36 meses



Periodos de 24 meses



48 meses

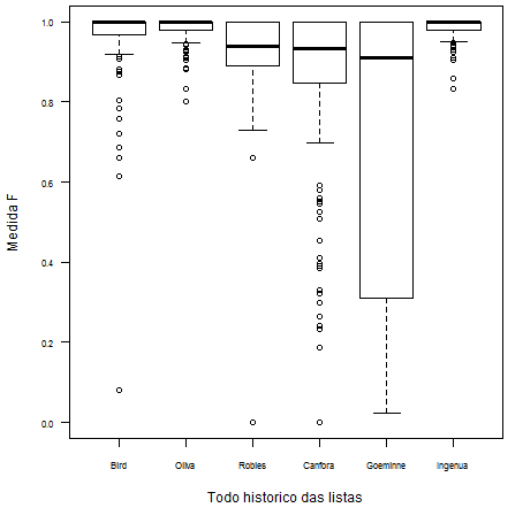


Figura 5: Boxplot das Medidas F para cada heurística em cada intervalo de tempo.

As Medidas F de todas as heurísticas apresentam médias próximas de 1 para os períodos de 3, 6, 12, 24, 36 e 48 meses. Contudo, quando consideramos todo o histórico observamos que as médias aumentam muito. Isso é um indício de que o tamanho do período considerado na lista de discussão pode influenciar a qualidade dos resultados de cada heurística. Argumentamos em favor do uso de intervalos que não utilizem todo o histórico da lista de discussão nos experimentos que utilizam essas listas como fontes de dados. O uso de todo histórico pode acarretar problemas em quaisquer das heurísticas de desambiguação de autores que foram avaliadas. Em contrapartida, todas as heurísticas apresentam bons resultados se forem considerados intervalos de tempo de até quatro anos.

Quando consideramos um período maior de tempo, a lista de discussão pode conter mais mensagens e mais endereços de e-mail em decorrência do crescimento da comunidade que utiliza a lista. Avaliamos a relação entre a eficácia das heurísticas e a quantidade de endereços de e-mail na seção 4.2.

4.2 O número de endereços de e-mail importa?

Avaliamos a qualidade dos resultados das heurísticas mediante o número de endereços de e-mails existentes em cada lista de discussão. Estamos consideramos apenas o histórico completo das listas para analisar a relação existente entre os valores da Medida F para cada heurística e o número de endereços de e-mail distintos encontrados em cada lista. Estamos cientes de que ao utilizar todo o histórico das listas estamos expondo todas as heurísticas ao pior cenário de desambiguação.

Para avaliar a relação entre a qualidade dos resultados das heurísticas e o tamanho da comunidade, verificamos a correlação de Spearman entre o número de endereços de e-mail únicos encontrados na lista de discussão de um projeto e a Medida F do resultado da desambiguação de autores gerado por cada heurística para o mesmo projeto. O uso da correção de Spearman se justifica pelo fato de que o conjunto das Medidas F não possui uma distribuição Normal. Assim, necessitamos de uma correlação não-paramétrica para estudar a relação entre o número de endereços e a Medida F. Os resultados dessa análise estão apresentados na Tabela 7.

Heurística	Correlação de Spearman entre número de endereços de e-mail e Medida F considerando todo o histórico de cada projeto
Bird	-0.5800478
Oliva	-0.1545838
Robles	-0.3952133
Canfora	-0.6569058
Goeminne	-0.8077635
Ingênua	-0.1491139

Tabela 7: Correlação de Spearman entre o número de endereços de e-mail únicos de cada projeto e a Medida F de cada heurística para o respectivo projeto.

Existem indícios de que a quantidade de endereços de e-mail na lista de discussão utilizada está relacionada negativamente com a qualidade dos resultados das heurísticas. Quanto mais endereços de e-mail na lista, piores serão os resultados das heurísticas. Contudo, podemos observar que a heurísticas de Oliva et al. e a heurística ingênua estão menos relacionadas com o número de endereços. Esses indícios sugerem melhores resultados por estas duas heurísticas nas situações em que não for possível realizar estudos com intervalos menores de tempo e se fizer necessário o uso de todo o histórico das listas de discussão.

Esses resultados também apresentam semelhanças com o mapeamento da característica de Inferência do prefixo do endereço de e-mail a partir do nome existente no cabeçalho da mensagem, conforme apresentado na Tabela 2. Contudo, são necessários mais experimentos e análises para determinar a relação entre essa característica e a influência do número de endereços de e-mail da comunidade estudada. Como trabalho futuro vamos investigar em mais detalhes esse comportamento para mapear quais características influenciam os resultados dos métodos negativamente.

4.3 Eficácia das heurísticas com diferentes intervalos de tempo

Para avaliarmos a eficácia de cada heurística nos diferentes intervalos de tempo, realizamos o teste não paramétrico de delta de Norman Cliff descritos por Macbeth et al. (Macbeth et al. 2011) e por Norman Cliff (Cliff 1996). Esse teste quantifica a diferença entre duas amostras de dados. Em nosso caso, quantificamos a diferença entre as Medidas F de cada heurística para cada intervalo de tempo. Buscamos assim, avaliar a eficácia de cada heurística em cada um dos diferentes intervalos de tempo. Como esse teste é realizado par-a-par, conduzimos 15 testes para cada intervalo de tempo, uma vez que precisamos avaliar cada heurística com seus pares para ordenarmos as heurísticas por eficácia de resultados em cada período. Essas avaliações nos possibilitam investigar qual heurística apresentou resultados estatisticamente melhores em cada intervalo de tempo.

O Anexo 2 apresenta as tabelas comparativas das heurísticas para cada intervalo de tempo e a Tabela 8 apresenta a síntese dos resultados das comparações dos métodos.

Intervalos de tempo	Heurísticas ordenadas por sua eficácia
3	Bird > Goeminne, Ingênua > Oliva, Robles > Canfora
6	Bird > Goeminne, Ingênua > Oliva, Robles > Canfora
12[6]	Bird > Ingênua, Goeminne, Oliva, Roble, Canfora
24	Bird, Ingênua > Goeminne, Oliva > Canfora, Robles
36	Bird, Ingênua, Goeminne > Oliva, Robles > Canfora
48	Bird, Ingênua, Goeminne, Robles > Oliva, Canfora
Todo histórico[7]	Bird, Ingênua, Oliva > Robles, Canfora > Goeminne

Tabela 8: Ordenação das heurísticas por sua eficácia mediante o teste de delta de Norman Cliff utilizando as Medidas F de cada intervalo de tempo.

Podemos observar na Tabela 8 que a heurística de Bird et al. possui resultados estatisticamente melhores que as demais heurísticas nos intervalos de até 12 meses e também se apresenta bons resultados em comparação às demais para os intervalos de 24, 36, 48 meses e todo o histórico. Esses indícios sugerem o uso desta heurística para realizar a desambiguação de autores.

[6] De acordo com os resultados do teste de Cliff, Bird et al. não apresenta diferença significativa contra a heurística ingênua e esta não apresenta diferenças significativas contra Goeminne e Mens. Contudo, Bird et al. possui melhores resultados que Goeminne e Mens. Como a diferença entre a heurística ingênua e Goeminne e Mens foi menor que a diferença entre Bird et al. e a heurística ingênua, optamos por considerar a heurística ingênua na mesma posição que Goeminne e Mens.

[7] De acordo com os resultados do teste de Cliff, Robles et al. é equivalente a Canfora et al. e este último é equivalente a Goeminne e Mens. Contudo, Robles et al. apresenta melhores resultados que Goeminne e Mens. Como o delta para Robles et al. e Goeminne e Mens foi o maior entre essas 3 comparações, optamos por considerar Robles et al. mais eficaz que Goeminne e Mens. E como o delta entre Robles et al. e Canfora et al. foi menor que entre Canfora et al. e Goeminne e Mens, consideramos que Robles et al. é equivalente a Canfora et al.

No sentido de avaliar a proporção e variação da qualidade dos resultados de cada heurística nos diferentes períodos, definimos três categorias para a eficácia das heurísticas, as amostras da Medida F para cada heurística estão: abaixo do segundo quartil (amostras com resultados 25% inferiores) consideramos como resultados ruins; entre o segundo e quarto quartil consideramos como resultados razoáveis e por fim acima do terceiro quartil das amostras da heurística (amostras com Medida F 25% superiores) consideramos que são bons resultados.

Os gráficos 1, 2 e 3 apresentam o percentual de amostras que cada heurística possui em cada categoria para cada intervalo de tempo. O Gráfico 1 ilustra os casos em que cada heurística apresentou os piores resultados.

Na visualização deste gráfico, quanto menor o valor, melhores foram os resultados da heurística. Neste sentido, o gráfico corrobora com os resultados da Tabela 8 no sentido de que as heurísticas Ingênua, Bird et al. e Goeminne e Mens apresentam resultados mais consistentes e possuem uma variação menor entre os intervalos de até 48 meses. Podemos observar que nos casos médios, a heurística ingênua está melhor posicionada que as demais.

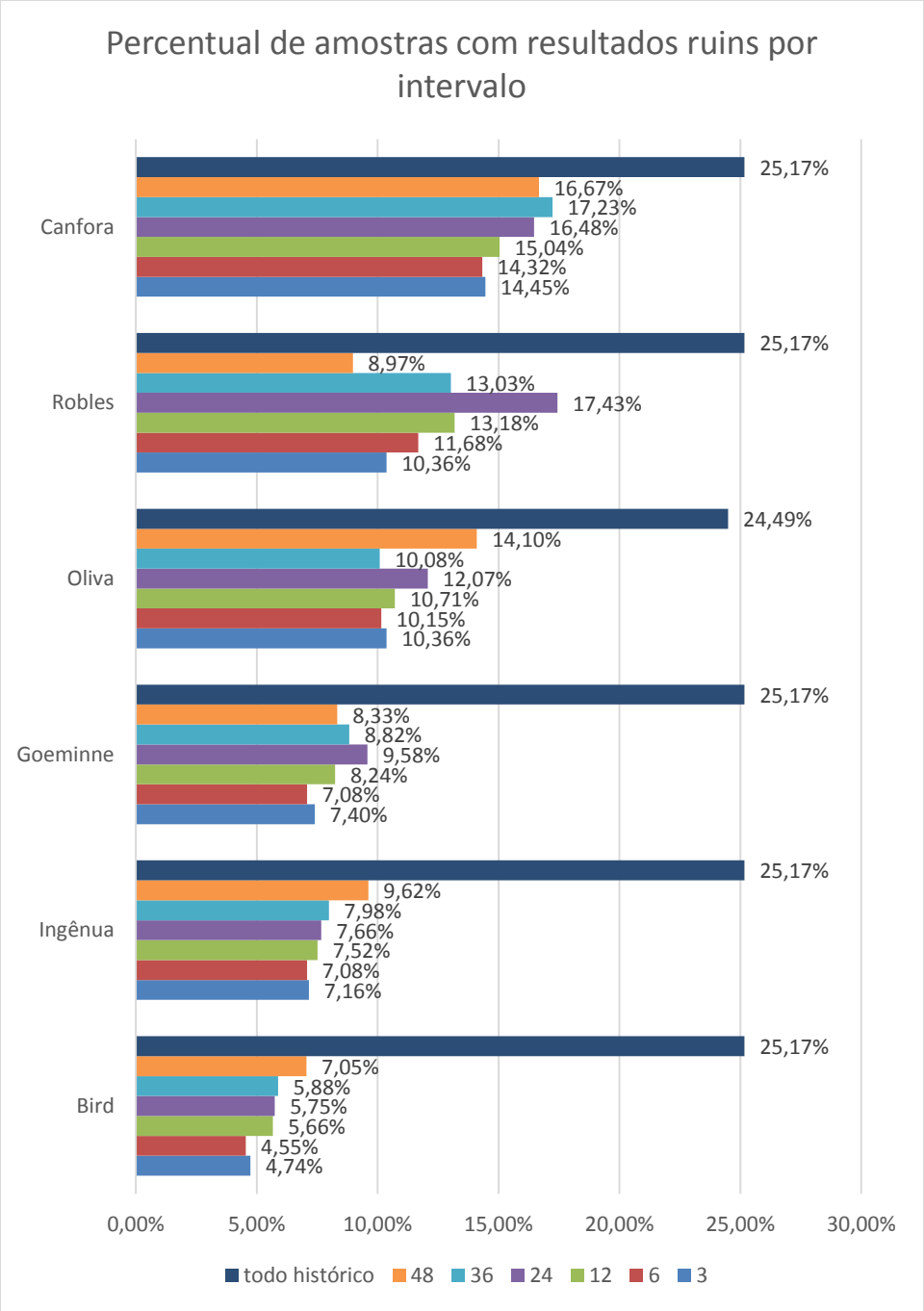


Gráfico 1: Casos em que as heurísticas de desambiguação apresentaram piores resultados.

O Gráfico 2 ilustra os casos médios de cada heurística. Como os demais intervalos estão todos zerados, apresentamos apenas o intervalo de todo o histórico.

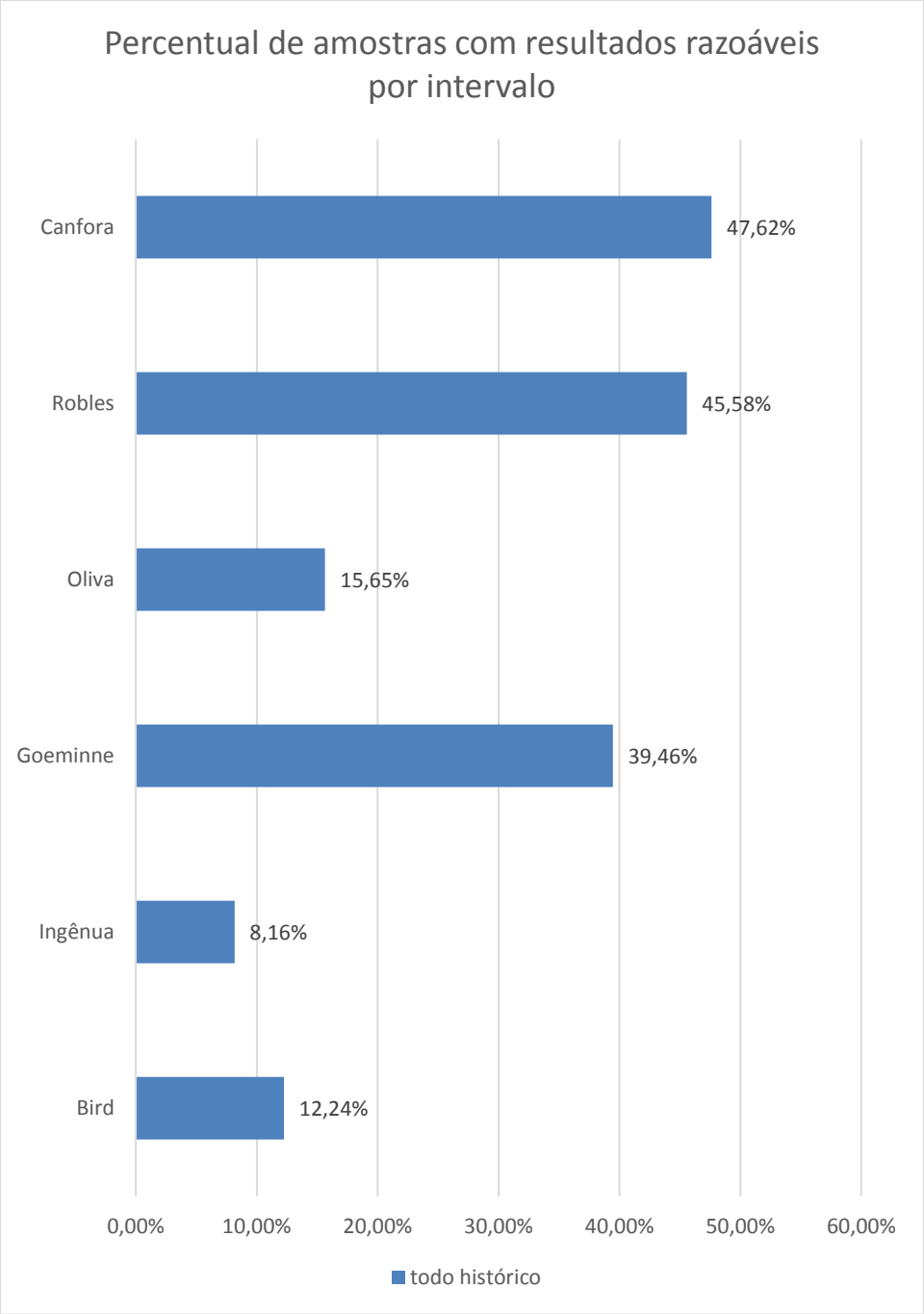


Gráfico 2: Casos em que as heurísticas de desambiguação apresentaram resultados razoáveis.

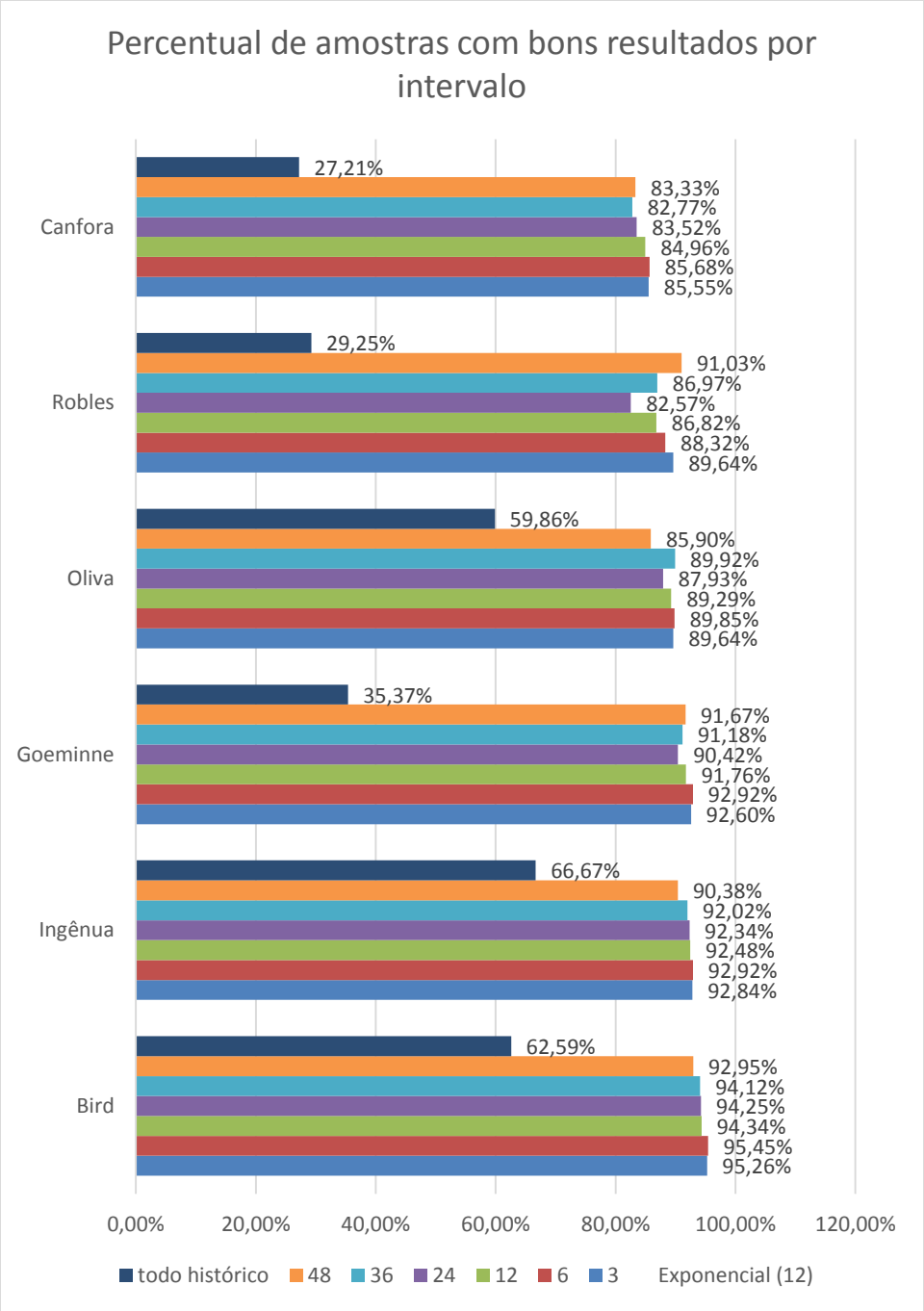


Gráfico 3: Casos em que as heurísticas de desambiguação apresentaram melhores resultados.

No Gráfico 3, quanto maior o percentual, melhor foi o desempenho da heurística. Existem indícios de que se for necessário o uso de todo o histórico das listas de discussão, a heurística ingênua apresenta melhores resultados que as demais, apesar de seus resultados para intervalos menores ficarem aquém da heurística de Bird et al.

5 Discussão

As listas de discussão das comunidades de software livre são utilizadas para informar sobre o status do projeto, discutir sobre problemas no software, procurar por instruções de uso, coordenar os membros do projeto, enviar avisos e normas, etc. (Guzzi et al. 2013a). Os históricos dessas listas são disponibilizados publicamente na internet e podem ser utilizados para explorar as interações humanas e o processo de desenvolvimento do projeto. Porém precisamos utilizar heurísticas de desambiguação de autores para solucionar os problemas dos múltiplos endereços de e-mail utilizados para cada membro (Bettenburg et al. 2009; Bird et al. 2006).

Neste contexto discutimos como essas listas são utilizadas atualmente pelas comunidades do software livre (Seção 5.1), a ética no uso das informações advindas destas listas para condução de estudos científicos (Seção 5.2) e o impacto das falhas de identificação das heurísticas na reprodução de um estudo da literatura (Seção 5.3).

5.1 Uso das listas de discussão pelas comunidades de software livre

A função das listas de discussão depende do projeto, do processo e do modo como a comunidade decide utilizá-la. A comunidade pode discutir problemas, novas funcionalidades do projeto, realizar votação entre os membros e aceitar patches de correção. Contudo, existem ferramentas como Jira e Bugzilla que favorecem a discussão de falhas e funcionalidades do projeto e a comunidade pode optar por utilizar ferramentas específicas para cada função. Além disso, com o passar do tempo a função da lista na comunidade pode mudar. O projeto Apache Httpd é um exemplo de como a função da lista foi modificada de acordo com as decisões da comunidade no decorrer do tempo. O projeto possui uma lista de discussão iniciada em março de 1996, contudo o primeiro registro no gerenciador de bugs do projeto é datado de março de 2002. Antes disso as falhas e funcionalidades do projeto eram discutidas na própria lista de discussão de desenvolvedores. A partir de 2002, as falhas passaram a ser discutidas no serviço gerenciador de bugs. Segundo Guzzi et al. (Guzzi et al. 2013a) a utilidade da lista de discussão mudou nos últimos anos.

A verificação da utilidade das listas na comunidade de software livre requer a inspeção manual das mensagens trocadas na lista. Neste contexto, realizamos um experimento para determinar se as listas de discussão de desenvolvedores ao menos existem em dois ecossistemas de desenvolvimento de software livre: Fundação Eclipse e Fundação Apache. Este experimento objetivou a verificação da existência dessas listas no contexto das comunidades de software livre.

Para cada comunidade foram inspecionados manualmente os sites de cada dos projetos e verificado se existiam informações sobre a existência de listas de discussão para os desenvolvedores daquele projeto. Verificamos apenas a existência da lista de discussão e não avaliamos a finalidade de cada lista de discussão.

Dos 228 projetos inspecionados da Fundação Eclipse, 196 possuíam links para lista de discussão. No caso da Fundação Apache 182 dos 200 projetos listas utilizam listas de discussão.

Isso indica que 91% dos projetos da Fundação Apache e 86% dos projetos da Fundação Eclipse utilizam listas de discussão. Apesar da afirmação de Guzzi et al. (Guzzi et al. 2013a) sobre a mudança dos usos das listas de discussão, existem indícios que essas listas ainda são muito utilizadas nas comunidades de software livre. Por este motivo, defendemos uma inspeção amostral de mensagens para determinar a função da lista para a comunidade a ser utilizado em estudos que utilizem as listas de discussão.

5.2 Ética no uso das informações de listas de discussão

Zimmer (Zimmer 2010) discute um conjunto de princípios éticos necessários durante a realização de experimentos utilizando informações existentes na internet. Os desafios da ética na condução de pesquisas com informações de redes sociais incluem a tradicional natureza de consentimento, respeito às regras de privacidade da comunidade estudada e aplicar técnicas para mascarar as identidades antes de publicar informações dos membros.

Neste contexto, precisamos distinguir entre dois ambientes distintos de privacidade durante a realização de pesquisas com dados advindos da internet: comunidades públicas e comunidades privadas. As pesquisas que utilizam as informações de comunidades privadas exigem o consentimento de todos os membros envolvidos na comunidade que será estudada. Em contrapartida, comunidades públicas podem ser realizadas com registros públicos sem o consentimento dos membros desde que obedeçam às políticas de privacidade da comunidade envolvida no estudo (Eysenbach & Till 2001).

No caso da Fundação Apache, As informações existentes nas listas de discussão são públicas[8] e os e-mails enviados para essas listas ficam submetidas às regras da Política de Arquivos de Fóruns Públicos[9]. Essa política deixa claro que quaisquer informações enviadas para essas listas se tornam públicas para promover o espírito de transparência e abertura da comunidade (Fundation 2015). A comunidade Apache entende que manter o livre acesso aos históricos da comunicação é de vital importância para o funcionamento da comunidade porque possibilita a existência de um histórico público de suas atividades e de um repositório pesquisável sobre o que acontece na história dos projetos (Fundation 2015).

A partir das fontes analisadas, as pesquisas que utilizam informações advindas das listas de discussão da comunidade Apache devem obedecer aos princípios existentes na política de privacidade da comunidade. Porém, defendemos que é necessário mascarar as identidades e endereços de e-mail para que os endereços os membros não sejam expostos a ataques de SPAM.

[8] http://apache.org/foundation/mailinglists.html
[9] http://apache.org/foundation/public-archives.html

5.3 Redes sociais da comunicação dos desenvolvedores do projeto Apache Ant

Realizamos um estudo de caso (Oliva et al. 2012) para caracterizar os grupos de centro / periferia no projeto Apache Ant. Para explorar a rede de comunicação dos membros foi necessária a extração dos dados da lista de discussão de desenvolvedores do projeto. Após a extração dos dados foram criadas redes sociais a partir da comunicação para condução da pesquisa sobre os grupos de centro / periferia.

A ferramenta OSSNetwork (Balieiro et al. 2007) foi utilizada para iniciar os experimentos, mas a inspeção das redes sociais geradas revelou a existência de máscaras nos endereços de e-mails. Isso motivou a condução deste trabalho para possibilitar a extração de informações das listas de discussão.

Para contornar os problemas de e-mails mascarados utilizamos os arquivos originais que armazenam o histórico das listas de discussão da Fundação Apache. A Apache mantém esses arquivos em um repositório na internet. Esse repositório é formado por arquivos divididos por projetos, listas, anos e meses. Todas as mensagens de um determinado mês estão armazenadas um arquivo específico.

Após a obtenção dos arquivos da lista de discussão de desenvolvedores do Apache Ant, extraímos as mensagens existentes nesses arquivos para aplicação de filtros de ruído e duplicatas. Os filtros foram aplicados considerando os títulos e remetentes das mensagens extraídas. As aplicações desses filtros possibilitaram a criação de uma comunidade livre dos endereços utilizados por ferramentas automáticas e duplicações que influenciam o número de mensagens de um membro.

O processo iterativo resultou em uma comunidade com endereços de e-mails mais precisos. A rede social foi criada a partir dessa comunidade e das trocas de e-mails entre os membros. Essa rede foi utilizada para explorar e analisar as características da comunidade dos desenvolvedores do projeto Apache Ant por meio de métricas de SNA (Social Network Analysis). Uma atribuição de autoria pouco precisa resulta na adição de nós e arestas duplicados. Essa duplicação adiciona ruído que distorce os resultados dessas análises.

Os resultados deste trabalho indicam que apenas 25% dos desenvolvedores do projeto podem ser classificados como membros do grupo central e que esses desenvolvedores estão entre aqueles que mais colaboram com códigos fontes no projeto. Além disso, apontam para a participação ativa destes membros nas listas de discussão.

Após o levantamento das heurísticas existentes na literatura e comparação das mesmas, reproduzimos o experimento com as informações da lista de discussão da comunidade Apache Ant e utilizamos o mesmo período informado por Oliva et al. Utilizamos ainda o mesmo código fonte utilizado na condução do trabalho original, alterando apenas a heurística que realiza a desambiguação dos autores. A Tabela 9 apresenta os resultados da reprodução deste experimento. Podemos observar que as únicas heurísticas que apresentam resultados semelhantes para este experimento são: Bird et al., Canfora et al. e Robles et al.

As diferenças de resultado entre as heurísticas comprovam nossa afirmação de que problemas na desambiguação de autores podem influenciar os resultados na identificação dos endereços de e-mail dos membros da comunidade e as medidas da rede social construídas a partir dessas informações.

Para proteger a privacidade dos membros da comunidade, mascaramos os domínios dos endereços de e-mail dos membros que foram identificados pelas diferentes heurísticas.

Bird et al.	**Canfora et al.**	**Robles et al.**	**Goeminne e Mens**	**Heurística Ingênua**	**Oliva et al.**
ajack@*	ajack@*	ajack@*	ajack@*	ajack@*	ajack@*
ajoo.email@*	ajoo.email@*	ajoo.email@*	ajoo.email@*	ajoo.email@*	ajoo.email@*
alexeys@*	alexeys@*	alexeys@*	alexeys@*	alexeys@*	alexeys@*
anthony.goubard@*	anthony.goubard@*	anthony.goubard@*	anthony.goubard@*	anthony.goubard@*	anthony.goubard@*
antoine@*	antoine@*	antoine@*	-	–	–
aurele.venet@*	aurele.venet@*	aurele.venet@*	aurele.venet@*	aurele.venet@*	aurele.venet@*
bodewig@*	bodewig@*	bodewig@*	bodewig@*	bodewig@*	bodewig@*
brett@*	brett@*	brett@*	brett@*	brett@*	brett@*
bruce@*	bruce@*	bruce@*	bruce@*	bruce@*	bruce@*
carej@*	carej@*	carej@*	carej@*	carej@*	carej@*
carnold@*	carnold@*	carnold@*	carnold@*	carnold@*	carnold@*
caster@*	caster@*	caster@*	caster@*	caster@*	caster@*
cmanolache@*	cmanolache@*	cmanolache@*	cmanolache@*	cmanolache@*	cmanolache@*
conor@*	conor@*	conor@*	conor@*	conor@*	conor@*
cstrong@*	cstrong@*	cstrong@*	cstrong@*	cstrong@*	cstrong@*
dak@*	dak@*	dak@*	dak@*	dak@*	dak@*
danson@*	danson@*	danson@*	danson@*	danson@*	danson@*
davidsouther@*	davidsouther@*	davidsouther@*	davidsouther@*	davidsouther@*	davidsouther@*
ddevienne@*	ddevienne@*	ddevienne@*	ddevienne@*	ddevienne@*	ddevienne@*
ehatcher@*	ehatcher@*	ehatcher@*	ehatcher@*	ehatcher@*	ehatcher@*
flamefew@*	flamefew@*	flamefew@*	flamefew@*	flamefew@*	flamefew@*
groboclown@*	groboclown@*	groboclown@*	groboclown@*	groboclown@*	groboclown@*
hunterp@*	hunterp@*	hunterp@*	hunterp@*	hunterp@*	hunterp@*
jalberto@*	jalberto@*	jalberto@*	jalberto@*	jalberto@*	jalberto@*
james.fuller.2007@*	james.fuller.2007@*	james.fuller.2007@*	james.fuller.2007@*	-	james.fuller.2007@*
jax@*	jax@*	jax@*	jax@*	jax@*	jax@*
jefft@*	jefft@*	jefft@*	jefft@*	jefft@*	jefft@*
jessh@*	jessh@*	jessh@*	jessh@*	jessh@*	jessh@*
jglick@*	jglick@*	jglick@*	jglick@*	jglick@*	jglick@*
-	–	–	–	jim.fuller@*	-
jkf@*	jkf@*	jkf@*	jkf@*	jkf@*	jkf@*
kevin.jackson@*	kevin.jackson@*	kevin.jackson@*	kevin.jackson@*	kevin.jackson@*	kevin.jackson@*

king@*	king@*	king@*	king@*	king@*	king@*
lists@*	lists@*	lists@*	lists@*	lists@*	lists@*
mariano@*	mariano@*	mariano@*	mariano@*	mariano@*	mariano@*
mark@*	mark@*	mark@*	mark@*	mark@*	mark@*
mbenson@*	mbenson@*	mbenson@*	mbenson@*	mbenson@*	mbenson@*
mcconnell@*	mcconnell@*	mcconnell@*	mcconnell@*	mcconnell@*	mcconnell@*
mharp@*	mharp@*	mharp@*	mharp@*	mharp@*	mharp@*
neeme@*	neeme@*	neeme@*	neeme@*	neeme@*	neeme@*
nicolaken@*	nicolaken@*	nicolaken@*	nicolaken@*	nicolaken@*	nicolaken@*
peterreilly@*	peterreilly@*	peterreilly@*	peterreilly@*	peterreilly@*	peterreilly@*
phil.weighill-smith@*	phil.weighill-smith@*	phil.weighill-smith@*	phil.weighill-smith@*	phil.weighill-smith@*	phil.weighill-smith@*
rainer@*	rainer@*	rainer@*	rainer@*	rainer@*	rainer@*
richard.evans@*	richard.evans@*	richard.evans@*	richard.evans@*	richard.evans@*	richard.evans@*
roxspring@*	roxspring@*	roxspring@*	roxspring@*	roxspring@*	roxspring@*
russgold@*	russgold@*	russgold@*	russgold@*	russgold@*	russgold@*
-	-	-	sbailliez@*	sbailliez@*	sbailliez@*
scohen@*	scohen@*	scohen@*	scohen@*	scohen@*	scohen@*
sergey.yevtushenko@*	sergey.yevtushenko@*	sergey.yevtushenko@*	sergey.yevtushenko@*	sergey.yevtushenko@*	sergey.yevtushenko@*
sstirling@*	sstirling@*	sstirling@*	sstirling@*	sstirling@*	sstirling@*
steve_l@*	steve_l@*	steve_l@*	steve_l@*	steve_l@*	steve_l@*
teknopaul@*	teknopaul@*	teknopaul@*	teknopaul@*	teknopaul@*	teknopaul@*
umagesh@*	umagesh@*	umagesh@*	umagesh@*	umagesh@*	umagesh@*
vegorov@*	vegorov@*	vegorov@*	vegorov@*	vegorov@*	vegorov@*
wascallywabbit@*	wascallywabbit@*	wascallywabbit@*	wascallywabbit@*	wascallywabbit@*	wascallywabbit@*
whaefelinger@*	whaefelinger@*	whaefelinger@*	whaefelinger@*	whaefelinger@*	whaefelinger@*
wikidiffs@*	wikidiffs@*	wikidiffs@*	wikidiffs@*	wikidiffs@*	wikidiffs@*
xavier@*	xavier@*	xavier@*	xavier@*	xavier@*	xavier@*
yves.martin@*	yves.martin@*	yves.martin@*	yves.martin@*	yves.martin@*	yves.martin@*
yyamano@*	yyamano@*	yyamano@*	yyamano@*	yyamano@*	yyamano@*

Tabela 9: Resultado da identificação de membros centrais da lista de discussão utilizando diferentes heurísticas.

A Figura 3 ilustra as intersecções dos resultados gerados por cada heurística durante a reprodução do experimento. Como o resultado de Bird et al., Robles et al. e Canfora et al. são idênticos no experimento que foi reproduzido, representamos eles como um único conjunto nomeado de Bird et al.

Podemos observar que 97% dos resultados são semelhantes, mas existem diferenças entre os resultados gerados pelas heurísticas de desambiguação. Neste experimento, a diferença apesar de existente é relativamente pequena. Como trabalho futuro reproduziremos outros trabalhos da

literatura para avaliar as divergências que podem ocorrer com o uso das diferentes heurísticas de desambiguação de autores.

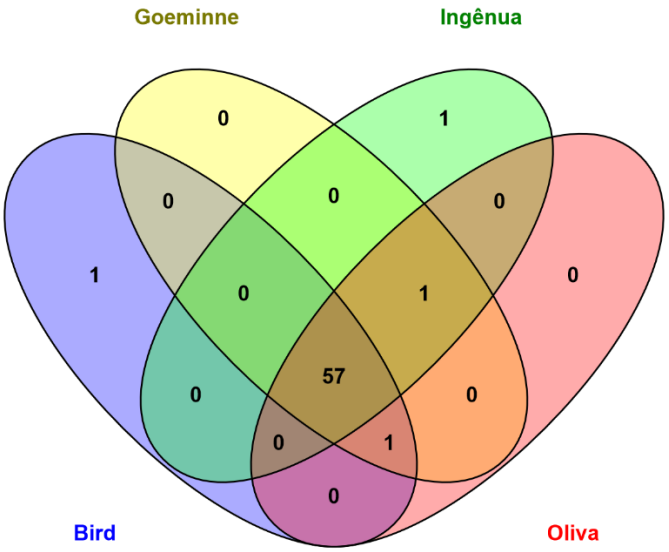


Figura 6: Intersecção dos resultados do experimento utilizando as heurísticas.

6 Ameaças à validade

A comparação considera apenas os endereços de e-mail que foram possíveis de serem adicionados à base de referência. Ela contém em média menos de 25% dos endereços de e-mail existentes na lista de discussão para cada projeto (vide Tabela 5 na seção 3.4). Essa incompletude adiciona o viés sobre os falsos positivos identificados por cada heurística. Como trabalho futuro, ampliaremos nossa base de referência para conter uma proporção maior dos endereços de e-mail utilizados na lista e verificar os resultados obtidos pela nossa amostra atual do conjunto verdade.

Este trabalho considera os nomes de usuários do domínio apache.org como sendo equivalentes aos usuários do repositório de códigos fontes da Fundação Apache. Essa decisão se baseia na documentação existente na página da comunidade que afirma que "toda identificação única de submissão está vinculada a um endereço de e-mail da própria comunidade"[10] e na experiência adquirida durante a verificação manual dos nomes de usuários do repositório para o projeto Apache Ant.

A Tabela 2 denota as principais características das heurísticas. Essas características possibilitam que uma heurística seja beneficiada em detrimento de outra de acordo com o conjunto de dados utilizado para a desambiguação de autores. Por exemplo, a heurística de Oliva et al. possui a característica de utilizar a recorrência dos pares de endereços de e-mail e nomes. Então, essa heurística se beneficia do uso de um volume maior de mensagens, enquanto que as demais sofrem com tal volume. Por este motivo, para amenizar os vieses das características de cada uma das heurísticas, realizamos análises fracionando o histórico da lista de discussão em vários períodos.

Se por um lado a diversidade de projetos favorece a generalização dos resultados, por outro pode servir de viés pelas diferentes características de cada comunidade. Devido à diversidade de número de usuários e tempo de vida de cada projeto, as heurísticas podem ser beneficiadas ou prejudicadas por estas características próprias de cada comunidade. Como trabalho futuro, avaliaremos o comportamento de cada heurística mediante a segregação das comunidades em conjuntos de listas de discussão com características semelhantes para mitigar o viés que essa diversidade pode incluir.

Este trabalho compara apenas heurísticas que utilizam as informações existentes nos cabeçalhos das mensagens das listas de discussão. Como trabalho futuro, avaliaremos heurísticas que utilizem outras formas de desambiguação, como estilo de escrita e saudações existentes nas mensagens. E para explorar como a diversidade dos endereços de e-mail afetam os resultados, como trabalho futuro avaliaremos os tipos de endereços de e-mail que mais são usados nas listas: empresarial, e-mail gratuito, etc.

[10] https://reference.apache.org/committer/email

7 Conclusões

As heurísticas de desambiguação de autores em listas de discussão encontradas na literatura, em geral, utilizam apenas as informações das listas porque existem ocasiões em que informações adicionais sobre os membros não estão disponíveis. Nesse sentido, essas heurísticas podem ser utilizadas para desambiguação dos autores das mensagens em qualquer lista de discussão.

Utilizamos informações adicionais extraídas de diferentes locais (Jira, sites e repositório de chaves públicas da comunidade) para construir uma base de referência dos endereços de e-mail da comunidade e avaliar a capacidade de identificação de seis dessas heurísticas: Bird et al.; Oliva et al., Canfora et al. Goeminne e Mens, Robles et al. e a heurística ingênua considerada por Kouters e Goeminne e Mens em seus trabalhos. Realizamos duas análises para cada uma dessas heurísticas: a primeira utilizando todo o histórico da lista de discussão e a segunda fragmentando este histórico em períodos anuais. A divisão em períodos da segunda análise objetiva reduzir a vantagem que a heurística de Oliva et al. possui ao utilizar todo o histórico da lista de discussão. Lembramos que após a identificação automática realizada pelas heurísticas, é necessária a inspeção manual dos resultados para mitigar possíveis erros de atribuição. Contudo, quanto menor for o índice de erros da heurística utilizada, menor serão o esforço e tempo necessários para correção de possíveis falhas geradas na desambiguação de autores.

Encontramos indícios de que os resultados de todas as heurísticas são afetados pelo uso de todo o histórico das listas de discussão. Apesar disso, a heurística de Bird et al. apresenta bons resultados em todos os casos. A qualidade de cada heurística está relacionada com o número de endereços existentes na lista de discussão e com o intervalo de tempo utilizado na análise. Neste sentido, os intervalos de tempo menores aparentam ser vantajosos em todas as heurísticas. Sendo preferível evitar o uso de todo o histórico da lista de discussão em estudos que utilizem essa fonte de dados. Apesar de todas as heurísticas se beneficiarem do uso de intervalos menores, verificamos uma eficácia maior pode parte da heurística de Bird et al. quando não estamos expostos ao histórico completo. E encontramos indícios de que esses erros podem alterar os resultados de pesquisas já realizadas.

Para os casos em que não é possível fracionar o histórico em períodos menores, encontramos indícios de que a heurística ingênua oferece uma eficácia melhor por estar menos sujeita a ruídos gerados pelo uso de similaridade. A heurística de Bird et al. utiliza similaridade de nomes e endereços de e-mail para identificar os autores, enquanto que a heurística ingênua evita tal uso. Existe a expectativa de que a primeira é mais bem-sucedida em conjuntos menores de dados que gerem menos ruídos durante a avaliação de similaridade dos autores e que a segunda alcance melhores resultados em conjuntos de dados maiores porque é menos afetada pelo ruído existente.

Nossos resultados corroboram com os resultados de Goeminne e Mens (Goeminne 2013) que identificaram que um aumento nos endereços corretamente identificados implica no aumento de falsos positivos durante a desambiguação dos autores. Contudo, encontramos indícios de que o

problema dos falsos positivos pode ser amenizado com a utilização de um conjunto de dados menor no processo de desambiguação dos autores.

Os resultados deste trabalho implicam em ameaças à validação dos trabalhos que utilizaram listas de discussão e apontam para a necessidade de que os autores levem em consideração a existência das demais heurísticas para desambiguação de autores.

Como trabalhos futuros, iremos explorar o modo como as heurísticas estão relacionadas com as características de cada comunidade (como tamanho e tempo de vida), e como essas características podem influenciar a capacidade de desambiguação dessas heurísticas. Queremos com isso, facilitar ainda mais a escolha das heurísticas utilizadas pelos pesquisadores que utilizam as listas de discussão na condução de estudos científicos. Esse tipo de trabalho se torna cada vez mais relevante na medida em que aparecem numerosas comunidades de produção coletiva (e.g., MOOCs, projetos de software livre, comunidades de prática, etc.).

8 Referências

Abreu, R. & Premraj, R., 2009. How developer communication frequency relates to bug introducing changes. In *Proceedings of the joint international and annual ERCIM workshops on Principles of software evolution (IWPSE) and software evolution (Evol) workshops*. ACM, pp. 153–158.

Bacchelli, A. et al., 2012. Content classification of development emails. In *Software Engineering (ICSE), 2012 34th International Conference on*. IEEE, pp. 375–385.

Bacchelli, A., Lanza, M. & Robbes, R., 2010. Linking e-mails and source code artifacts. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1*. ACM, pp. 375–384.

Balieiro, M.A. et al., 2007. Ossnetwork: Um ambiente para estudo de comunidades de software livre usando redes sociais. In *Experimental Software Engineering Latin America Workshop*. pp. 33–424.

Bernardi, M.L. et al., 2012. Do developers introduce bugs when they do not communicate? the case of eclipse and mozilla. In *Software Maintenance and Reengineering (CSMR), 2012 16th European Conference on*. IEEE, pp. 139–148.

Bettenburg, N., Shihab, E. & Hassan, A.E., 2009. An empirical study on the risks of using off-the-shelf techniques for processing mailing list data. In *Software Maintenance, 2009. ICSM 2009. IEEE International Conference on*. IEEE, pp. 539–542.

Bird, C. et al., 2008. Latent social structure in open source projects. In *Proceedings of the 16th ACM SIGSOFT International Symposium on Foundations of software engineering*. ACM, pp. 24–35.

Bird, C. et al., 2006. Mining email social networks. In *Proceedings of the 2006 international workshop on Mining software repositories*. ACM, pp. 137–143.

Bird, C. et al., 2007. Open borders? immigration in open source projects. In *Mining Software Repositories, 2007. ICSE Workshops MSR'07. Fourth International Workshop on*. IEEE, pp. 6–6.

Bird, C. & Nagappan, N., 2012. Who? where? what?: examining distributed development in two large open source projects. In *Proceedings of the 9th IEEE Working Conference on Mining Software Repositories*. IEEE Press, pp. 237–246.

Bosu, A. & Carver, J.C., 2014. How do social interaction networks influence peer impressions formation? a case study. In *Open Source Software: Mobile Open Source Technologies*. Springer, pp. 31–40.

Canfora, G. et al., 2011. Social interactions around cross-system bug fixings: the case of FreeBSD and OpenBSD. In *Proceedings of the 8th working conference on mining software repositories*. ACM, pp. 143–152.

Canfora, G. et al., 2012. Who is going to mentor newcomers in open source projects? In *Proceedings of the ACM SIGSOFT 20th International Symposium on the Foundations of Software Engineering*. ACM, p. 44.

Cliff, N., 1996. Ordinal methods for behavioral data analysis.

D'Ambros, M. et al., 2008. Analysing Software Repositories to Understand Software Evolution. In T. Mens & S. Demeyer, eds. *Software Evolution*. Springer, pp. 37–67. Available at: http://dx.doi.org/10.1007/978-3-540-76440-3.

Dendek, P.J., Wojewódzki, M. & Bolikowski, L., 2013. Author disambiguation in the YADDA2 software platform. In *Intelligent Tools for Building a Scientific Information Platform*. Springer, pp. 131–143.

Enron Email Dataset, C.P., 2003. Enron Email Dataset. Available at: https://www.cs.cmu.edu/ ./enron/.

Eysenbach, G. & Till, J.E., 2001. Ethical issues in qualitative research on internet communities. *Bmj*, 323(7321), pp.1103–1105.

Fundation, A.S., 2015. Public Forum Archive Policy. Available at: http://apache.org/foundation/public-archives.html.

Godby, C.J. et al., 2010. Who's who in your digital collection: developing a tool for name disambiguation and identity resolution. In *Journal of the Chicago Colloquium on Digital Humanities and Computer Science*.

Goeminne, M., 2013. *Understanding the Evolution of Socio-technical Aspects in Open Source Ecosystems: An Empirical Analysis of GNOME*. Ph. D. dissertation, UMONS.

Guzzi, A. et al., 2013a. Communication in open source software development mailing lists. In *Proceedings of the 10th Working Conference on Mining Software Repositories*. MSR '13. San Francisco, CA, USA: IEEE Press, pp. 277–286. Available at: http://dl.acm.org/citation.cfm?id=2487085.2487139.

Guzzi, A. et al., 2013b. Communication in open source software development mailing lists. In *Proceedings of the 10th Working Conference on Mining Software Repositories*. MSR '13. San Francisco, CA, USA: IEEE Press, pp. 277–286. Available at: http://dl.acm.org/citation.cfm?id=2487085.2487139.

Hassan, A.E., 2008. The road ahead for mining software repositories. In *Frontiers of Software Maintenance, 2008. FoSM 2008.* IEEE, pp. 48–57.

Hemmati, H. et al., 2013. The msr cookbook: Mining a decade of research. In *Mining Software Repositories (MSR), 2013 10th IEEE Working Conference on*. IEEE, pp. 343–352.

Kouters, E., 2013. Identity matching and geographical movement of open-source software mailing list participants.

Kouters, E. et al., 2012. Who's who in Gnome: Using LSA to merge software repository identities. In *Software Maintenance (ICSM), 2012 28th IEEE International Conference on*. IEEE, pp. 592–595.

Macbeth, G., Razumiejczyk, E. & Ledesma, R.D., 2011. Cliff's Delta Calculator: A non-parametric effect size program for two groups of observations. *Universitas Psychologica*, 10(2), pp.545–555.

Navarro, G. et al., 2001. Indexing methods for approximate string matching. *IEEE Data Eng. Bull.*, 24(4), pp.19–27.

Neu, S. et al., 2011. Telling stories about GNOME with Complicity. In *Visualizing Software for Understanding and Analysis (VISSOFT), 2011 6th IEEE International Workshop on*. IEEE, pp. 1–8.

Nia, R. et al., 2010a. Validity of network analyses in open source projects. In *Mining Software Repositories (MSR), 2010 7th IEEE Working Conference on*. IEEE, pp. 201–209.

Nia, R. et al., 2010b. Validity of network analyses in open source projects. In *Mining Software Repositories (MSR), 2010 7th IEEE Working Conference on*. IEEE, pp. 201–209.

Oliva, G.A. et al., 2012. Characterizing key developers: a case study with apache ant. In *Collaboration and Technology*. Springer, pp. 97–112.

Overbaugh, R.C., 2002. Undergraduate education majors' discourse on an electronic mailing list. *Journal of Research on Technology in Education*, 35(1), pp.117–138.

Panichella, S., Bavota, G., et al., 2014. How Developers' Collaborations Identified from Different Sources Tell Us about Code Changes. In *Software Maintenance and Evolution (ICSME), 2014 IEEE International Conference on*. IEEE, pp. 251–260.

Panichella, S., Canfora, G., et al., 2014. How the evolution of emerging collaborations relates to code changes: an empirical study. In *Proceedings of the 22nd International Conference on Program Comprehension*. ACM, pp. 177–188.

Pimentel, M. & Fuks, H., 2011. *Sistemas colaborativos*, Ed Campus.

Poncin, W., Serebrenik, A. & Brand, M. van den, 2011. Process mining software repositories. In *Software Maintenance and Reengineering (CSMR), 2011 15th European Conference on*. IEEE, pp. 5–14.

Rigby, P.C., German, D.M. & Storey, M.-A., 2008. Open source software peer review practices: a case study of the apache server. In *Proceedings of the 30th international conference on Software engineering*. ACM, pp. 541–550.

Roberts, J., Hann, I.-H. & Slaughter, S., 2006. Communication networks in an open source software project. In *Open Source Systems*. Springer, pp. 297–306.

Robles, G. et al., 2009. Tools for the study of the usual data sources found in libre software projects. *International Journal of Open Source Software and Processes (IJOSSP)*, 1(1), pp.24–45.

Robles, G. & Gonzalez-Barahona, J.M., 2005. Developer identification methods for integrated data from various sources. *ACM SIGSOFT Software Engineering Notes*, 30(4), pp.1–5.

Da Silva, J.T. et al., 2015. Quem é quem na lista de discussão? Identificando diferentes e-mails de um mesmo participante. In *Collaborative Systems (SBSC), 2015 Brazilian Symposium on*.

Squire, M., 2013. Project Roles in the Apache Software Foundation: A Dataset. In *Proceedings of the 10th Working Conference on Mining Software Repositories*. MSR '13. San Francisco, CA, USA: IEEE Press, pp. 301–304. Available at: http://dl.acm.org/citation.cfm?id=2487085.2487142.

Steinmacher, I. et al., 2012. Newcomers withdrawal in open source software projects: Analysis of Hadoop Common project. In *Collaborative Systems (SBSC), 2012 Brazilian Symposium on*. IEEE, pp. 65–74.

Valverde, S. & Solé, R.V., 2007. Self-organization versus hierarchy in open-source social networks. *Physical Review E*, 76(4), p.046118.

Vasilescu, B. et al., 2014. How social Q&A sites are changing knowledge sharing in open source software communities. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*. ACM, pp. 342–354.

Xu, J. et al., 2005a. A topological analysis of the open souce software development community. In *System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on*. IEEE, p. 198a–198a.

Xu, J. et al., 2005b. A topological analysis of the open souce software development community. In *System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on*. IEEE, p. 198a–198a.

Xuan, Q. & Filkov, V., 2014. Building it together: synchronous development in OSS. In *Proceedings of the 36th International Conference on Software Engineering*. ACM, pp. 222–233.

Yin, M. et al., 2011. User Name Alias Extraction in Emails. *International Journal of Image, Graphics and Signal Processing (IJIGSP)*, 3(3), p.1.

Zimmer, M., 2010. But the data is already public": on the ethics of research in Facebook. *Ethics and information technology*, 12(4), pp.313–325.

9 Anexos

Anexo A – Lista das listas de discussão dos projetos avaliados

Avaliamos 150 listas de desenvolvedores no total. Abaixo seguem os repositórios de cada uma das listas utilizadas nas análises. Note-se que existem mais de 150 repositórios porque alguns históricos foram movidos. Nestes casos adicionamos os históricos existentes em todos esses repositórios para que fosse possível a reconstrução de todo a vida da lista de discussão do projeto.

Projeto	**Endereço da lista de discussão**
Projeto Apache abdera	http://mail-archives.apache.org/mod_mbox/abdera-dev/.
Projeto Apache Accumulo	http://mail-archives.apache.org/mod_mbox/accumulo-dev/.
Projeto Apache accumulo	http://mail-archives.apache.org/mod_mbox/incubator-accumulo-dev/.
Projeto Apache ace	http://mail-archives.apache.org/mod_mbox/ace-dev/.
Projeto Apache ace	http://mail-archives.apache.org/mod_mbox/incubator-ace-dev/.
Projeto Apache ActiveMQ	http://mail-archives.apache.org/mod_mbox/activemq-dev/.
Projeto Apache airavata	http://mail-archives.apache.org/mod_mbox/airavata-dev/.
Projeto Apache airavata	http://mail-archives.apache.org/mod_mbox/incubator-airavata-dev/.
Projeto Apache allura	http://mail-archives.apache.org/mod_mbox/allura-dev/.
Projeto Apache allura	http://mail-archives.apache.org/mod_mbox/incubator-allura-dev/.
Projeto Apache ambari	http://mail-archives.apache.org/mod_mbox/ambari-dev/.
Projeto Apache ambari	http://mail-archives.apache.org/mod_mbox/incubator-ambari-dev/.
Projeto Apache Ant	http://mail-archives.apache.org/mod_mbox/ant-dev/.
Projeto Apache AntUnit	http://mail-archives.apache.org/mod_mbox/ant-dev/.
Projeto Apache Archiva	http://mail-archives.apache.org/mod_mbox/archiva-dev.
Projeto Apache archiva	http://mail-archives.apache.org/mod_mbox/archiva-dev/.
Projeto Apache Aries	http://mail-archives.apache.org/mod_mbox/aries-dev/.
Projeto Apache aries	http://mail-archives.apache.org/mod_mbox/incubator-aries-dev/.
Projeto Apache aurora	http://mail-archives.apache.org/mod_mbox/aurora-dev/.
Projeto Apache Avro	http://mail-archives.apache.org/mod_mbox/avro-dev/.
Projeto Apache Axis2	http://mail-archives.apache.org/mod_mbox/axis-java-dev/.
Projeto Apache axis-c	http://mail-archives.apache.org/mod_mbox/axis-c-dev/.
Projeto Apache batchee	http://mail-archives.apache.org/mod_mbox/incubator-batchee-dev/.
Projeto Apache batik	http://mail-archives.apache.org/mod_mbox/xmlgraphics-batik-dev/.
Projeto Apache bcel	http://mail-archives.apache.org/mod_mbox/jakarta-bcel-dev/.

Projeto Apache bigtop	http://mail-archives.apache.org/mod_mbox/bigtop-dev/.
Projeto Apache bigtop	http://mail-archives.apache.org/mod_mbox/incubator-bigtop-dev/.
Projeto Apache bloodhound	http://mail-archives.apache.org/mod_mbox/bloodhound-dev/.
Projeto Apache bloodhound	http://mail-archives.apache.org/mod_mbox/incubator-bloodhound-dev/.
Projeto Apache bookkeeper	http://mail-archives.apache.org/mod_mbox/bookkeeper-dev/.
Projeto Apache bsf	http://mail-archives.apache.org/mod_mbox/jakarta-bsf-dev/.
Projeto Apache buildr	http://mail-archives.apache.org/mod_mbox/buildr-dev/.
Projeto Apache bval	http://mail-archives.apache.org/mod_mbox/bval-dev/.
Projeto Apache bval	http://mail-archives.apache.org/mod_mbox/incubator-bval-dev/.
Projeto Apache Camel	http://mail-archives.apache.org/mod_mbox/camel-dev/.
Projeto Apache cassandra	http://mail-archives.apache.org/mod_mbox/cassandra-dev/.
Projeto Apache Cayenne	http://mail-archives.apache.org/mod_mbox/cayenne-dev/.
Projeto Apache celix	http://mail-archives.apache.org/mod_mbox/celix-dev/.
Projeto Apache celix	http://mail-archives.apache.org/mod_mbox/incubator-celix-dev/.
Projeto Apache Chemistry	http://mail-archives.apache.org/mod_mbox/chemistry-dev/.
Projeto Apache chemistry	http://mail-archives.apache.org/mod_mbox/incubator-chemistry-dev/.
Projeto Apache chukwa	http://mail-archives.apache.org/mod_mbox/chukwa-dev/.
Projeto Apache chukwa	http://mail-archives.apache.org/mod_mbox/incubator-chukwa-dev/.
Projeto Apache clerezza	http://mail-archives.apache.org/mod_mbox/clerezza-dev/.
Projeto Apache clerezza	http://mail-archives.apache.org/mod_mbox/incubator-clerezza-dev/.
Projeto Apache Click	http://mail-archives.apache.org/mod_mbox/click-dev/.
Projeto Apache cloudstack	http://mail-archives.apache.org/mod_mbox/cloudstack-dev/.
Projeto Apache cloudstack	http://mail-archives.apache.org/mod_mbox/incubator-cloudstack-dev/.
Projeto Apache Cocoon	http://mail-archives.apache.org/mod_mbox/cocoon-dev/.
Projeto Apache commons	http://mail-archives.apache.org/mod_mbox/commons-dev/.
Projeto Apache continuum	http://mail-archives.apache.org/mod_mbox/continuum-dev/.
Projeto Apache Continuum	http://mail-archives.apache.org/mod_mbox/maven-continuum-dev/.
Projeto Apache couchdb	http://mail-archives.apache.org/mod_mbox/couchdb-dev/.
Projeto Apache crunch	http://mail-archives.apache.org/mod_mbox/crunch-dev/.
Projeto Apache crunch	http://mail-archives.apache.org/mod_mbox/incubator-crunch-dev/.
Projeto Apache cTAKES	http://mail-archives.apache.org/mod_mbox/ctakes-dev/.

Projeto Apache ctakes	http://mail-archives.apache.org/mod_mbox/incubator-ctakes-dev/.
Projeto Apache CXF	http://mail-archives.apache.org/mod_mbox/cxf-dev/.
Projeto Apache db-derby	http://mail-archives.apache.org/mod_mbox/db-derby-dev/.
Projeto Apache drill	http://mail-archives.apache.org/mod_mbox/drill-dev/.
Projeto Apache drill	http://mail-archives.apache.org/mod_mbox/incubator-drill-dev/.
Projeto Apache esme	http://mail-archives.apache.org/mod_mbox/esme-dev/.
Projeto Apache esme	http://mail-archives.apache.org/mod_mbox/incubator-esme-dev/.
Projeto Apache etch	http://mail-archives.apache.org/mod_mbox/etch-dev/.
Projeto Apache Etch	http://mail-archives.apache.org/mod_mbox/incubator-etch-dev/.
Projeto Apache falcon	http://mail-archives.apache.org/mod_mbox/falcon-dev/.
Projeto Apache Felix	http://mail-archives.apache.org/mod_mbox/felix-dev/.
Projeto Apache flink	http://mail-archives.apache.org/mod_mbox/flink-dev/.
Projeto Apache flume	http://mail-archives.apache.org/mod_mbox/flume-dev/.
Projeto Apache flume	http://mail-archives.apache.org/mod_mbox/incubator-flume-dev/.
Projeto Apache FOP	http://mail-archives.apache.org/mod_mbox/xmlgraphics-fop-dev/.
Projeto Apache Forrest	http://mail-archives.apache.org/mod_mbox/forrest-dev/.
Projeto Apache ftpserver	http://mail-archives.apache.org/mod_mbox/incubator-ftpserver-dev/.
Projeto Apache Geronimo	http://mail-archives.apache.org/mod_mbox/geronimo-dev/.
Projeto Apache Gora	http://mail-archives.apache.org/mod_mbox/gora-dev/.
Projeto Apache gora	http://mail-archives.apache.org/mod_mbox/incubator-gora-dev/.
Projeto Apache Hadoop	http://mail-archives.apache.org/mod_mbox/hadoop-common-dev/.
Projeto Apache hadoop-chukwa	http://mail-archives.apache.org/mod_mbox/hadoop-chukwa-dev/.
Projeto Apache hadoop-common	http://mail-archives.apache.org/mod_mbox/hadoop-common-dev/.
Projeto Apache hadoop-hdfs	http://mail-archives.apache.org/mod_mbox/hadoop-hdfs-dev/.
Projeto Apache hadoop-mapreduce	http://mail-archives.apache.org/mod_mbox/hadoop-mapreduce-dev/.
Projeto Apache hadoop-pig	http://mail-archives.apache.org/mod_mbox/hadoop-pig-dev/.
Projeto Apache hadoop-yarn	http://mail-archives.apache.org/mod_mbox/hadoop-yarn-dev/.
Projeto Apache hama	http://mail-archives.apache.org/mod_mbox/hama-dev/.
Projeto Apache harmony	http://mail-archives.apache.org/mod_mbox/harmony-dev/.
Projeto Apache HBase	http://mail-archives.apache.org/mod_mbox/hbase-dev/.
Projeto Apache hc	http://mail-archives.apache.org/mod_mbox/hc-dev/.

Projeto Apache helix	http://mail-archives.apache.org/mod_mbox/helix-dev/.
Projeto Apache Hive	http://mail-archives.apache.org/mod_mbox/hbase-dev/.
Projeto Apache hive	http://mail-archives.apache.org/mod_mbox/hive-dev/.
Projeto Apache httpd	http://mail-archives.apache.org/mod_mbox/httpd-dev/.
Projeto Apache isis	http://mail-archives.apache.org/mod_mbox/incubator-isis-dev/.
Projeto Apache isis	http://mail-archives.apache.org/mod_mbox/isis-dev/.
Projeto Apache Jackrabbit	http://mail-archives.apache.org/mod_mbox/jackrabbit-dev/.
Projeto Apache JAMES	http://mail-archives.apache.org/mod_mbox/james-mime4j-dev/.
Projeto Apache james-server	http://mail-archives.apache.org/mod_mbox/james-server-dev/.
Projeto Apache jena	http://mail-archives.apache.org/mod_mbox/incubator-jena-dev/.
Projeto Apache Jena	http://mail-archives.apache.org/mod_mbox/jena-dev/.
Projeto Apache jmeter	http://mail-archives.apache.org/mod_mbox/jakarta-jmeter-dev/.
Projeto Apache jmeter	http://mail-archives.apache.org/mod_mbox/jmeter-dev/.
Projeto Apache JSPWiki	http://mail-archives.apache.org/mod_mbox/incubator-jspwiki-dev/.
Projeto Apache jspwiki	http://mail-archives.apache.org/mod_mbox/jspwiki-dev/.
Projeto Apache juddi	http://mail-archives.apache.org/mod_mbox/juddi-dev/.
Projeto Apache kandula	http://mail-archives.apache.org/mod_mbox/ws-kandula-dev/.
Projeto Apache Karaf	http://mail-archives.apache.org/mod_mbox/karaf-dev/.
Projeto Apache knox	http://mail-archives.apache.org/mod_mbox/knox-dev/.
Projeto Apache lenya	http://mail-archives.apache.org/mod_mbox/lenya-dev/.
Projeto Apache log4j	http://mail-archives.apache.org/mod_mbox/logging-log4j-dev/.
Projeto Apache lucene	http://mail-archives.apache.org/mod_mbox/lucene-dev/.
Projeto Apache lucene-solr	http://mail-archives.apache.org/mod_mbox/lucene-solr-dev/.
Projeto Apache lucy	http://mail-archives.apache.org/mod_mbox/incubator-lucy-dev/.
Projeto Apache lucy	http://mail-archives.apache.org/mod_mbox/lucy-dev/.
Projeto Apache Mahout	http://mail-archives.apache.org/mod_mbox/mahout-dev/.
Projeto Apache marmotta	http://mail-archives.apache.org/mod_mbox/marmotta-dev/.
Projeto Apache Maven	http://mail-archives.apache.org/mod_mbox/maven-dev/.
Projeto Apache mesos	http://mail-archives.apache.org/mod_mbox/incubator-mesos-dev/.
Projeto Apache mesos	http://mail-archives.apache.org/mod_mbox/mesos-dev/.
Projeto Apache mina	http://mail-archives.apache.org/mod_mbox/mina-dev/.
Projeto Apache MyFaces	http://mail-archives.apache.org/mod_mbox/myfaces-dev/.
Projeto Apache nuvem	http://mail-archives.apache.org/mod_mbox/incubator-nuvem-dev/.
Projeto Apache ode	http://mail-archives.apache.org/mod_mbox/ode-dev/.
Projeto Apache ofbiz	http://mail-archives.apache.org/mod_mbox/ofbiz-dev/.
Projeto Apache oltu	http://mail-archives.apache.org/mod_mbox/oltu-dev/.

Projeto Apache oodt	http://mail-archives.apache.org/mod_mbox/incubator-oodt-dev/.
Projeto Apache oodt	http://mail-archives.apache.org/mod_mbox/oodt-dev/.
Projeto Apache oozie	http://mail-archives.apache.org/mod_mbox/incubator-oozie-dev/.
Projeto Apache oozie	http://mail-archives.apache.org/mod_mbox/oozie-dev/.
Projeto Apache openejb	http://mail-archives.apache.org/mod_mbox/openejb-dev/.
Projeto Apache OpenJPA	http://mail-archives.apache.org/mod_mbox/openjpa-dev/.
Projeto Apache openmeetings	http://mail-archives.apache.org/mod_mbox/incubator-openmeetings-dev/.
Projeto Apache openmeetings	http://mail-archives.apache.org/mod_mbox/openmeetings-dev/.
Projeto Apache opennlp	http://mail-archives.apache.org/mod_mbox/incubator-opennlp-dev/.
Projeto Apache opennlp	http://mail-archives.apache.org/mod_mbox/opennlp-dev/.
Projeto Apache openoffice	http://mail-archives.apache.org/mod_mbox/openoffice-dev/.
Projeto Apache openwebbeans	http://mail-archives.apache.org/mod_mbox/openwebbeans-dev/.
Projeto Apache orc	http://mail-archives.apache.org/mod_mbox/orc-dev/.
Projeto Apache PDFBox	http://mail-archives.apache.org/mod_mbox/pdfbox-dev/.
Projeto Apache phoenix	http://mail-archives.apache.org/mod_mbox/phoenix-dev/.
Projeto Apache photark	http://mail-archives.apache.org/mod_mbox/incubator-photark-dev/.
Projeto Apache Pig	http://mail-archives.apache.org/mod_mbox/pig-dev/.
Projeto Apache Pivot	http://mail-archives.apache.org/mod_mbox/pivot-dev/.
Projeto Apache POI	http://mail-archives.apache.org/mod_mbox/poi-dev/.
Projeto Apache Qpid	http://mail-archives.apache.org/mod_mbox/qpid-dev/.
Projeto Apache rampart	http://mail-archives.apache.org/mod_mbox/ws-rampart-c-dev/.
Projeto Apache rampart	http://mail-archives.apache.org/mod_mbox/ws-rampart-dev/.
Projeto Apache rat	http://mail-archives.apache.org/mod_mbox/incubator-rat-dev/.
Projeto Apache rave	http://mail-archives.apache.org/mod_mbox/incubator-rave-dev/.
Projeto Apache Rave	http://mail-archives.apache.org/mod_mbox/rave-dev/.
Projeto Apache river	http://mail-archives.apache.org/mod_mbox/incubator-river-dev/.
Projeto Apache river	http://mail-archives.apache.org/mod_mbox/river-dev/.
Projeto Apache Roller	http://mail-archives.apache.org/mod_mbox/roller-dev/.
Projeto Apache samza	http://mail-archives.apache.org/mod_mbox/samza-dev/.
Projeto Apache sanselan	http://mail-archives.apache.org/mod_mbox/incubator-sanselan-dev/.
Projeto Apache Shindig	http://mail-archives.apache.org/mod_mbox/shindig-dev/.

Projeto Apache shiro	http://mail-archives.apache.org/mod_mbox/incubator-shiro-dev/.
Projeto Apache shiro	http://mail-archives.apache.org/mod_mbox/shiro-dev/.
Projeto Apache sirona	http://mail-archives.apache.org/mod_mbox/incubator-sirona-dev/.
Projeto Apache sis	http://mail-archives.apache.org/mod_mbox/incubator-sis-dev/.
Projeto Apache sis	http://mail-archives.apache.org/mod_mbox/sis-dev/.
Projeto Apache Sling	http://mail-archives.apache.org/mod_mbox/sling-dev/.
Projeto Apache Solr	http://mail-archives.apache.org/mod_mbox/lucene-dev/.
Projeto Apache spamassassin	http://mail-archives.apache.org/mod_mbox/spamassassin-dev/.
Projeto Apache spark	http://mail-archives.apache.org/mod_mbox/spark-dev/.
Projeto Apache sqoop	http://mail-archives.apache.org/mod_mbox/incubator-sqoop-dev/.
Projeto Apache sqoop	http://mail-archives.apache.org/mod_mbox/sqoop-dev/.
Projeto Apache stanbol	http://mail-archives.apache.org/mod_mbox/incubator-stanbol-dev/.
Projeto Apache Stanbol	http://mail-archives.apache.org/mod_mbox/stanbol-dev/.
Projeto Apache stdcxx	http://mail-archives.apache.org/mod_mbox/stdcxx-dev/.
Projeto Apache storm	http://mail-archives.apache.org/mod_mbox/storm-dev/.
Projeto Apache stratos	http://mail-archives.apache.org/mod_mbox/stratos-dev/.
Projeto Apache streams	http://mail-archives.apache.org/mod_mbox/incubator-streams-dev/.
Projeto Apache struts	http://mail-archives.apache.org/mod_mbox/struts-dev/.
Projeto Apache Struts2	http://mail-archives.apache.org/mod_mbox/struts-dev/.
Projeto Apache subversion	http://mail-archives.apache.org/mod_mbox/subversion-dev/.
Projeto Apache Synapse	http://mail-archives.apache.org/mod_mbox/synapse-dev/.
Projeto Apache syncope	http://mail-archives.apache.org/mod_mbox/incubator-syncope-dev/.
Projeto Apache Syncope	http://mail-archives.apache.org/mod_mbox/syncope-dev/.
Projeto Apache tajo	http://mail-archives.apache.org/mod_mbox/tajo-dev/.
Projeto Apache tapestry	http://mail-archives.apache.org/mod_mbox/tapestry-dev/.
Projeto Apache thrift	http://mail-archives.apache.org/mod_mbox/incubator-thrift-dev/.
Projeto Apache thrift	http://mail-archives.apache.org/mod_mbox/thrift-dev/.
Projeto Apache Tika	http://mail-archives.apache.org/mod_mbox/tika-dev/.
Projeto Apache tiles	http://mail-archives.apache.org/mod_mbox/tiles-dev/.
Projeto Apache Tomcat	http://mail-archives.apache.org/mod_mbox/tomcat-dev/.
Projeto Apache tomee	http://mail-archives.apache.org/mod_mbox/tomee-dev/.
Projeto Apache UIMA	http://mail-archives.apache.org/mod_mbox/uima-dev/.
Projeto Apache whirr	http://mail-archives.apache.org/mod_mbox/incubator-whirr-dev/.

Projeto Apache Whirr	http://mail-archives.apache.org/mod_mbox/whirr-dev/.
Projeto Apache Wicket	http://mail-archives.apache.org/mod_mbox/wicket-dev/.
Projeto Apache wink	http://mail-archives.apache.org/mod_mbox/incubator-wink-dev/.
Projeto Apache wink	http://mail-archives.apache.org/mod_mbox/wink-dev/.
Projeto Apache wookie	http://mail-archives.apache.org/mod_mbox/incubator-wookie-dev/.
Projeto Apache wookie	http://mail-archives.apache.org/mod_mbox/wookie-dev/.
Projeto Apache ws-sandesha	http://mail-archives.apache.org/mod_mbox/ws-sandesha-dev/.
Projeto Apache ws-savan	http://mail-archives.apache.org/mod_mbox/ws-savan-dev/.
Projeto Apache xerces-j	http://mail-archives.apache.org/mod_mbox/xerces-j-dev/.
Projeto Apache xmlbeans	http://mail-archives.apache.org/mod_mbox/xmlbeans-dev/.
Projeto Apache ZooKeeper	http://mail-archives.apache.org/mod_mbox/zookeeper-dev/.

Anexo 2 – Avaliação estatística dos métodos por período

Comparações para 3 meses: Bird > Goeminne, Kouter > Oliva, Rouble > Canfora

Comparações	Delta	Intervalo
bird_fmeasure, oliva_fmeasure	0.0573586	inf sup 0.04539460 0.06930616
bird_fmeasure, robles_fmeasure	0.05775799	inf sup 0.04579460 0.06970481
bird_fmeasure, canfora_fmeasure	0.1017989	inf sup 0.08857454 0.11498730
bird_fmeasure, goeminne_fmeasure	0.02700956	inf sup 0.01615593 0.03785682
bird_fmeasure, kouterBaseLine_fmeasure	0.02426785	inf sup 0.01351321 0.03501687
oliva_fmeasure, robles_fmeasure	0.0008859358	inf sup -0.01300531 0.01477684
oliva_fmeasure, canfora_fmeasure	0.04940235	inf sup 0.03438740 0.06439501
oliva_fmeasure, goeminne_fmeasure	-0.03056562	inf sup -0.04349994 -0.01762105
oliva_fmeasure, kouterBaseLine_fmeasure	-0.03374534	inf sup -0.04659404 -0.02088548
robles_fmeasure, canfora_fmeasure	0.04755882	inf sup 0.03254148 0.06255470
robles_fmeasure, goeminne_fmeasure	-0.03128324	inf sup -0.04421688 -0.01833911
robles_fmeasure, kouterBaseLine_fmeasure	-0.03446759	inf sup -0.04731626 -0.02160752
canfora_fmeasure, goeminne_fmeasure	-0.07699585	inf sup -0.09109045 -0.06287040
canfora_fmeasure, kouterBaseLine_fmeasure	-0.07998196	inf sup -0.09399765 -0.06593457
goeminne_fmeasure, kouterBaseLine_fmeasure	-0.002966403	inf sup -0.014795830 0.008863854

Comparações para 6 meses: Bird > Goeminne, Kouter > Oliva, Rouble > Canfora

Comparações	Valores	Intervalo
bird_fmeasure, oliva_fmeasure	0.05646117	inf sup 0.03989716 0.07299415
bird_fmeasure, robles_fmeasure	0.07248194	inf sup 0.05519219 0.08972824
bird_fmeasure, canfora_fmeasure	0.1018123	inf sup 0.08341441 0.12014090
bird_fmeasure, goeminne_fmeasure	0.02542334	inf sup 0.01051230 0.04032306
bird_fmeasure, kouterBaseLine_fmeasure	0.0251437	inf sup 0.01023168 0.04004454
oliva_fmeasure, robles_fmeasure	0.0167404	inf sup -0.003181417 0.036648930
oliva_fmeasure, canfora_fmeasure	0.05022846	inf sup 0.02930904 0.07110389
oliva_fmeasure, goeminne_fmeasure	-0.03129875	inf sup -0.04917142 -0.01340603
oliva_fmeasure, kouterBaseLine_fmeasure	-0.03193037	inf sup -0.04980145 -0.01403885
robles_fmeasure, canfora_fmeasure	0.0340261	inf sup 0.01250984 0.05551086
robles_fmeasure, goeminne_fmeasure	-0.04782935	inf sup -0.06636542 -0.02926028
robles_fmeasure, kouterBaseLine_fmeasure	-0.04855903	inf sup -0.06709357 -0.02999099
canfora_fmeasure, goeminne_fmeasure	-0.07852609	inf sup -0.09807936 -0.05891220
canfora_fmeasure, kouterBaseLine_fmeasure	-0.07912474	inf sup -0.09867728 -0.05951114
goeminne_fmeasure, kouterBaseLine_fmeasure	-0.00044669	inf sup -0.01681589 0.01592275

Comparações para 12 meses: Bird > Kouter, Goeminne, Oliva, Roble > Canfora

Pelo intervalo, aparece que Bird = Kouter, Kouter=Goeminne, mas Bird > Goeminne. Como a diferença entre Kouter e Goeminne foi menor que a diferença entre Bird e Kouter, decidimos por Kouter = Goeminne).

Comparações	Valores	Intervalo
bird_fmeasure, oliva_fmeasure	0.05115603	inf sup 0.02679754 0.07545380
bird_fmeasure, robles_fmeasure	0.07653995	inf sup 0.05067092 0.10230634
bird_fmeasure, canfora_fmeasure	0.09872821	inf sup 0.07186632 0.12544698
bird_fmeasure, goeminne_fmeasure	0.02597363	inf sup 0.003327889 0.048592737
bird_fmeasure, kouterBaseLine_fmeasure	0.01860229	inf sup -0.003499204 0.040685615
oliva_fmeasure, robles_fmeasure	0.02593226	inf sup -0.003013542 0.054834647
oliva_fmeasure, canfora_fmeasure	0.05213604	inf sup 0.02226119 0.08191787
oliva_fmeasure, goeminne_fmeasure	-0.02542953	inf sup -0.0515005737 0.0006761583
oliva_fmeasure, kouterBaseLine_fmeasure	-0.03284647	inf sup -0.058441341 -0.007208447
robles_fmeasure, canfora_fmeasure	0.02752638	inf sup -0.003619097 0.058618505
robles_fmeasure, goeminne_fmeasure	-0.05122073	inf sup -0.07866930 -0.02369455
robles_fmeasure, kouterBaseLine_fmeasure	-0.05864403	inf sup -0.08563901 -0.03156301
canfora_fmeasure, goeminne_fmeasure	-0.07498295	inf sup -0.10333905 -0.04650506
canfora_fmeasure, kouterBaseLine_fmeasure	-0.08189504	inf sup -0.10980746 -0.05385355
goeminne_fmeasure, kouterBaseLine_fmeasure	-0.00739043	inf sup -0.03139452 0.01662218

Comparações para 24 meses: Bird, Kouter > Goeminne, Oliva > Canfora, Rouble

Comparações	Delta	Intervalo
bird_fmeasure, oliva_fmeasure	0.06392302	inf sup 0.02941627 0.09827759
bird_fmeasure, robles_fmeasure	0.1183299	inf sup 0.07987014 0.15643785
bird_fmeasure, canfora_fmeasure	0.1126855	inf sup 0.0749251 0.1501232
bird_fmeasure, goeminne_fmeasure	0.03914358	inf sup 0.006840281 0.071365273
bird_fmeasure, kouterBaseLine_fmeasure	0.01933325	inf sup -0.01107803 0.04970878
oliva_fmeasure, robles_fmeasure	0.05434081	inf sup 0.01111276 0.09736612
oliva_fmeasure, canfora_fmeasure	0.05421603	inf sup 0.01157829 0.09665697
oliva_fmeasure, goeminne_fmeasure	-0.02437574	inf sup -0.06213846 0.01345667
oliva_fmeasure, kouterBaseLine_fmeasure	-0.04471455	inf sup -0.080840851 -0.008470925
robles_fmeasure, canfora_fmeasure	0.00306807	inf sup -0.0428726 0.0489958
robles_fmeasure, goeminne_fmeasure	-0.07829818	inf sup -0.11955247 -0.03677396
robles_fmeasure, kouterBaseLine_fmeasure	-0.09930858	inf sup -0.1390293 -0.0592688
canfora_fmeasure, goeminne_fmeasure	-0.07573289	inf sup -0.11634740 -0.03486552
canfora_fmeasure, kouterBaseLine_fmeasure	-0.09512485	inf sup -0.13420001 -0.05575434
goeminne_fmeasure, kouterBaseLine_fmeasure	-0.01995714	inf sup -0.05405814 0.01419037

Comparações para 36 meses: Bird, Kouter, Goeminne > Oliva, Robles > Canfora

Comparações	Delta	Intervalo
bird_fmeasure, oliva_fmeasure	0.0430584	inf sup -0.005806916 0.091718568
bird_fmeasure, robles_fmeasure	0.07238189	inf sup 0.01976746 0.12459654
bird_fmeasure, canfora_fmeasure	0.120048	inf sup 0.06300963 0.17630473
bird_fmeasure, goeminne_fmeasure	0.02999435	inf sup -0.01709915 0.07695507
bird_fmeasure, kouterBaseLine_fmeasure	0.02107902	inf sup -0.02476490 0.06683447
oliva_fmeasure, robles_fmeasure	0.02987077	inf sup -0.02791783 0.08746037
oliva_fmeasure, canfora_fmeasure	0.08156204	inf sup 0.01961969 0.14288065
oliva_fmeasure, goeminne_fmeasure	-0.01345244	inf sup -0.06622237 0.03939253
oliva_fmeasure, kouterBaseLine_fmeasure	-0.0224737	inf sup -0.07412345 0.02929631
robles_fmeasure, canfora_fmeasure	0.05433938	inf sup -0.0106059 0.1188281
robles_fmeasure, goeminne_fmeasure	-0.04309371	inf sup -0.09916193 0.01324728
robles_fmeasure, kouterBaseLine_fmeasure	-0.05246805	inf sup -0.107477663 0.002861844
canfora_fmeasure, goeminne_fmeasure	-0.09312549	inf sup -0.15300170 -0.03256803
canfora_fmeasure, kouterBaseLine_fmeasure	-0.1012993	inf sup -0.16017567 -0.04170478
goeminne_fmeasure, kouterBaseLine_fmeasure	-0.009144834	inf sup -0.05919331 0.04094950

Comparações para 48 meses: Bird, Goeminne, Robles, Kouter > Oliva, Canfora

Comparações	Delta	Intervalo
bird_fmeasure, oliva_fmeasure	0.07223866	inf sup 0.00370006 0.14010173
bird_fmeasure, robles_fmeasure	0.01894313	inf sup -0.04172186 0.07946896
bird_fmeasure, canfora_fmeasure	0.1019888	inf sup 0.03019425 0.17273644
bird_fmeasure, goeminne_fmeasure	0.01199869	inf sup -0.04754301 0.07145542
bird_fmeasure, kouterBaseLine_fmeasure	0.02547666	inf sup -0.03628079 0.08704026
oliva_fmeasure, robles_fmeasure	-0.05382972	inf sup -0.12468802 0.01757488
oliva_fmeasure, canfora_fmeasure	0.03583169	inf sup -0.04510145 0.11629753
oliva_fmeasure, goeminne_fmeasure	-0.06188363	inf sup -0.131721583 0.008565604
oliva_fmeasure, kouterBaseLine_fmeasure	-0.04750164	inf sup -0.11928845 0.02477923
robles_fmeasure, canfora_fmeasure	0.08288133	inf sup 0.008265923 0.156578903
robles_fmeasure, goeminne_fmeasure	-0.007190993	inf sup -0.06995429 0.05562901
robles_fmeasure, kouterBaseLine_fmeasure	0.006656805	inf sup -0.05822552 0.07148313
canfora_fmeasure, goeminne_fmeasure	-0.09196252	inf sup -0.164635 -0.018297
canfora_fmeasure, kouterBaseLine_fmeasure	-0.07905983	inf sup -0.153601284 -0.003623655
goeminne_fmeasure, kouterBaseLine_fmeasure	0.01397107	inf sup -0.04990173 0.07773006

Comparações para todo o histórico: Bird, Kouter, Oliva > Roubles, Canfora > Goeminne

(Pelo intervalo, aparece que Robles = Canfora e Canfora = Goeminne, mas Robles > Goeminne. Como o delta para Robles e Goeminne foi o maior entre as 3 comparações, escolhi Robles > Goeminne. E como o delta entre Robles e Canfora foi menor que Canfora e Goeminne, escolhi Robles = Canfora.)

Comparações	Delta	Intervalo
bird_fmeasure, oliva_fmeasure	-0.02850664	inf sup -0.14461534 0.08837642
bird_fmeasure, robles_fmeasure	0.4045074	inf sup 0.2826083 0.5135492
bird_fmeasure, canfora_fmeasure	0.4601324	inf sup 0.3423468 0.5636795
bird_fmeasure, goeminne_fmeasure	0.4261187	inf sup 0.3086271 0.5308027
bird_fmeasure, kouterBaseLine_fmeasure	-0.08607525	inf sup -0.19757495 0.02762316
oliva_fmeasure, robles_fmeasure	0.4893331	inf sup 0.3725205 0.5908818
oliva_fmeasure, canfora_fmeasure	0.5345458	inf sup 0.4202579 0.6321390
oliva_fmeasure, goeminne_fmeasure	0.4569392	inf sup 0.3379991 0.5615130
oliva_fmeasure, kouterBaseLine_fmeasure	-0.0642325	inf sup -0.17724226 0.05044909
robles_fmeasure, canfora_fmeasure	0.1023185	inf sup -0.02948831 0.23062783
robles_fmeasure, goeminne_fmeasure	0.1886714	inf sup 0.0515113 0.3188455
robles_fmeasure, kouterBaseLine_fmeasure	-0.5317229	inf sup -0.6275679 -0.4200021
canfora_fmeasure, goeminne_fmeasure	0.1171271	inf sup -0.01787755 0.24793733
canfora_fmeasure, kouterBaseLine_fmeasure	-0.5748068	inf sup -0.6668369 -0.4655068
goeminne_fmeasure, kouterBaseLine_fmeasure	-0.4893331	inf sup -0.5888346 -0.3752212