

Sfohart: Um Sistema de Recomendação Multicritério para Adoção de Software Livre

Leandro Soriano Ferreira¹, Roberto Almeida Bittencourt²,
Christina von Flach Garcia Chavez¹

¹ Departamento de Ciência da Computação
Universidade Federal da Bahia (UFBA)
Av. Adhemar de Barros, s/n - Campus de Ondina
Salvador - Bahia – Brasil – CEP 40.170-110

² Universidade Estadual de Feira de Santana (UEFS)
Feira de Santana - Bahia – Brasil

{leandro.soriano,flach}@ufba.br, roberto@uefs.br

Abstract. *This paper presents a survey on adoption factors of systems and components of open source software. Based on results of this survey, we define a set of free software's evaluation criteria to reviews more consistently and objectively. We build a multi-criteria recommender of open source systems and components using this criteria set and both colaborative and content-based filtering techniques.*

Resumo. *Este trabalho apresenta uma pesquisa de campo sobre fatores de adoção para sistemas e componentes de software de código aberto na comunidade brasileira de TI. Com base nos principais fatores apontados pela pesquisa, definimos um conjunto de critérios de avaliação de software livre cujo intuito é orientar a comunidade brasileira de TI a avaliar este tipo de software de modo coerente e objetivo. A partir deste conjunto de critérios, construímos um recomendador multicritério para sistemas e componentes de software de código aberto, utilizando tanto técnicas de filtragem baseada em conteúdo como filtragem colaborativa.*

1. Introdução

No processo de escolha de um software livre que atenda aos requisitos desejados, pessoas e organizações costumam iniciar a busca com um software proprietário para, somente em seguida, buscar uma alternativa livre [Verheesen 2010]. Isto leva a uma lógica própria para a escolha de um software livre: em vez de uma especificação precisa de requisitos, procura-se a solução livre mais aproximada de uma solução proprietária.

Uma variedade enorme de produtos sócio-culturais (e.g., filmes, livros, canções) está hoje disponível em repositórios online de informação. Em muitos casos, é necessário fazer escolhas sem experiência pessoal suficiente sobre as alternativas existentes [Resnick et al. 1997, Reategui and Cazella 2005]. Esta mesma necessidade de escolha vale também para o universo de software livre. Diariamente, confiamos nossas escolhas a recomendações vindas de outras pessoas, seja de forma direta¹, seja através de revistas

¹”forma direta” aqui corresponde a uma tradução livre para a expressão da língua inglesa ”word of mouth”, também entendida no contexto da frase como ”propaganda boca a boca”

ou sites especializados, dentre várias alternativas disponíveis [Resnick et al. 1997]. Sistemas de recomendação são uma alternativa recente que procuram melhorar a capacidade e a eficácia deste processo de identificação e seleção, já bastante disseminado nas relações sociais entre pessoas [Reategui and Cazella 2005].

Recomendar software, no entanto, é um processo complexo, potencialmente dependente de uma diversidade de critérios de avaliação. Portanto, para construir sistemas de recomendação de software livre, faz-se necessário definir critérios de avaliação ou fatores de influência para a sua adoção/aquisição. Morgan e Finnegan (2007) foram pioneiros em investigar fatores de adoção de software livre de ordem tecnológica, organizacional, ambiental/mercadológica e individual. Dentre os fatores encontrados, pode-se mencionar a complexidade do software, o apoio da alta gerência no uso de software livre, a adoção prévia de software livre por grandes empresas, além do contato com pessoas experientes no uso ou no desenvolvimento de software livre [Morgan and Finnegan 2007].

Este trabalho procura também investigar os fatores para adoção de software livre no contexto do mercado brasileiro de tecnologia de informação (TI). Mais especificamente, são investigados aqui os fatores que poderiam ser de potencial impacto na construção de um sistema de recomendação de software livre. Uma survey foi realizada com 152 respondentes. A partir dos resultados desta survey e de entrevistas com potenciais usuários deste sistema de recomendação, foi desenvolvido um protótipo de sistema de recomendação de software livre baseado em critérios múltiplos. Este protótipo encontra-se disponível gratuitamente para a comunidade de TI, e será descrito em detalhes neste artigo.

O restante deste artigo está organizado da seguinte forma: a Seção 2 revisa trabalhos relacionados sobre fatores de adoção de software livre, bem como sobre recomendação de sistemas e demais componentes de software de código aberto; a Seção 3 descreve a survey realizada e os resultados encontrados sobre fatores de adoção de software livre passíveis de serem utilizados em sistemas de recomendação; finalmente, a Seção 4 descreve a ferramenta protótipo de busca e recomendação de software livre desenvolvida.

2. Trabalhos Relacionados

Um estudo prévio, realizado em empresas de software norueguesas, procurou investigar a adoção de software livre neste contexto [Hauge et al. 2008]. Os resultados mostraram que 50% dos participantes utilizavam software de código aberto em soluções verticais². Além disso, 30% dos participantes obtiveram pelo menos 40% dos seus lucros oriundos de software de código aberto, ou de serviços relacionados a este tipo de software.

Morgan e Finnegan (2007), por sua vez, elencaram e classificaram fatores de adoção de software livre em quatro diferentes contextos: tecnológico, organizacional, ambiental (de mercado) e individual. Este trabalho procurou entender quais fatores são os mais relevantes para uma avaliação concisa e coerente de software livre e de sua adoção. No contexto *tecnológico*, foram encontrados fatores relacionados à *complexidade*, à *compatibilidade*, à *testabilidade* e a *custos e outras vantagens relativas*. Já no contexto *organizacional*, foram revelados fatores relacionados ao *apoio da alta gerência* e ao *expertise* dos colaboradores. No contexto *ambiental*, foram destacados fatores relacionados às

²”Soluções verticais” correspondem a soluções de software aplicadas a um nicho específico de mercado.

condições de mercado do software e/ou da própria organização. Finalmente, no contexto *individual*, foram elencados fatores relativos a *competências e habilidades pessoais*, bem como a *experiências prévias* no uso de software livre. Os autores também procuraram relacionar os fatores de adoção encontrados através da influência (positiva ou negativa) que cada um exerce sobre o outro [Morgan and Finnegan 2007]. Nosso trabalho utiliza esta classificação de fatores de adoção como ponto de partida de uma survey sobre fatores de adoção de software livre com potencial impacto em um sistema de recomendação.

Por outro lado, recomendação de software livre não parece ser um tema extensamente investigado. Dos poucos trabalhos encontrados, pode-se citar o AppRecommender, um recomendador de aplicativos GNU/Linux, que utiliza como base os aplicativos disponíveis para a distribuição Debian Linux [Araujo 2011]. A proposta do presente trabalho é mais flexível que o AppRecommender, pois recomenda soluções que podem ser utilizadas não apenas em distribuições GNU/Linux, mas em qualquer sistema operacional. Além disso, o AppRecommender é um recomendador *monocritério*, enquanto que a proposta deste trabalho é *multicritério*. Rascal é um outro sistema de recomendação baseado em filtragem colaborativa voltado para o reuso ágil de componentes de software [Mccarey et al. 2005]. Este recomendador baseia-se também numa avaliação monocritério, diferente da proposta deste trabalho, além de estar voltado especificamente para o objetivo de reuso de software. Finalmente, OhLoh é um diretório público de software e demais componentes de código aberto, com foco especial na busca de software, embora possua algumas características de um sistema de recomendação híbrido (mesclando filtragem colaborativa e baseada em conteúdo). Entretanto, o OhLoh também não permite avaliar os sistemas de software com base em critérios múltiplos [Black Duck Software 2013].

3. Survey sobre Fatores de Adoção de Software Livre

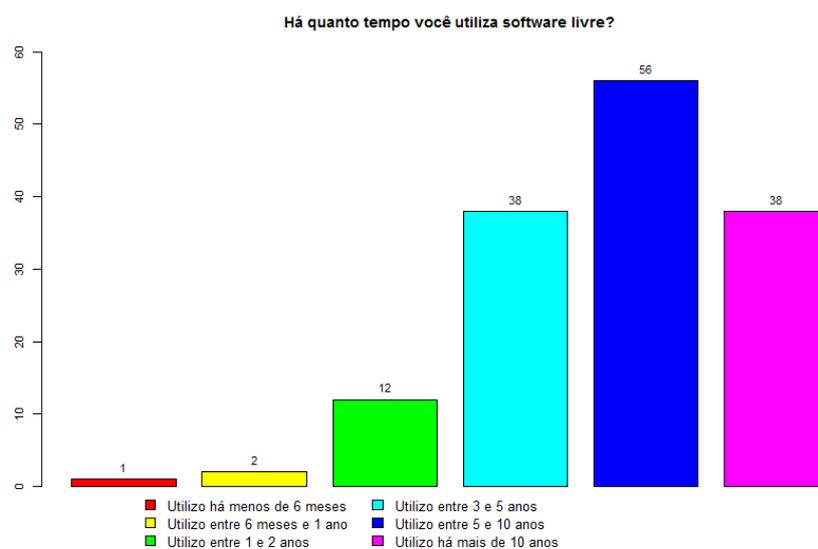


Figura 1. Tempo de uso de software livre dos participantes

Antes de criarmos um recomendador de software, era necessário compreender melhor quais são os fatores que mais influenciam na adoção de sistemas e componen-

tes de software de código aberto. Este entendimento era essencial para a definição dos critérios de avaliação de software livre de um sistema de recomendação multicritério. Para tanto, realizamos um levantamento de campo através de um questionário *online*, divulgado através de redes sociais, listas de discussão, dentre outros veículos da comunidade brasileira de TI. Os participantes desta pesquisa, em sua maioria, eram profissionais e/ou estudantes da área de TI, atuando quase exclusivamente no mercado brasileiro. Após descreverem sua experiência no uso/customização de sistemas e/ou componentes de software de código aberto, os participantes foram questionados a definir o grau de influência de uma série de fatores para a adoção de software livre. Estes fatores foram agrupados nas mesmas categorias definidas pela literatura [Morgan and Finnegan 2007].

3.1. Análise dos Resultados da Survey

Dois meses após a divulgação do questionário de pesquisa, obtivemos 152 respostas completas. A experiência de uso de software livre dos participantes pode ser vista na Figura 1. Cerca de 86.84% dos participantes possuíam experiência acima de dois anos na utilização de software livre. Dadas as características desta amostra do universo de profissionais de TI brasileiros, assumimos que os participantes da pesquisa possuíam relativa maturidade para definir, com coerência, quais fatores são de fato relevantes para a adoção de software livre.

Os resultados sobre os fatores tecnológicos, organizacionais, mercadológicos e individuais podem internamente apresentar alguma superposição, já que as respostas às perguntas no questionário de pesquisa permitiam múltipla escolha dos fatores investigados. Para quantificar a relevância de cada fator na adoção de software livre, definimos a seguinte escala: **(a)** *nenhuma influência*, **(b)** *pouca influência*, **(c)** *influência considerável*, e **(d)** *muita influência*. Em uma primeira análise, avaliamos especialmente a quantidade de respostas na categoria *muita influência*, ou no agregado de respostas nas categorias *muita influência* e *influência considerável*.

A Figura 2 ilustra os resultados da pesquisa sobre fatores tecnológicos. Os principais fatores tecnológicos foram: **(a)** *vantagem(ns) em relação a custos com hardware, requisitos mínimos menos exigentes, custos com licenças ou suporte, dentre outros*, com 52% dos participantes considerando-o um fator de *muita influência*; **(b)** *facilidade de entender, testar ou adaptar*, com 48%; **(c)** *compatibilidade com a infraestrutura existente, com os requisitos/necessidades/demandas e/ou com a tecnologia em vigor* com 43.4% e **(d)** *maior confiabilidade em relação à solução, livre ou não, utilizada atualmente*, com 42%.

Tais fatores, segundo a pesquisa, foram mais relevantes que fatores como *utilização de modelos de desenvolvimento e de qualidade bem definidos* (19.7%), ou *facilidade de testar e de ser avaliado por usuários de um modo geral* (28.2%). Ao contrário do que esperávamos, *maior eficiência em relação à solução, livre ou não, utilizada anteriormente* (36.84%) foi apenas o quinto fator mais relevante, dentre os sete elencados para o contexto tecnológico.

Já os resultados destacados na Figura 3, apontaram, como principais fatores organizacionais, os seguintes: **(a)** *custo total de propriedade de software*, com 53.2% dos participantes, **(b)** *relevância para a organização*, com 46.7% e **(c)** *apoio à disseminação e uso de software livre por parte da alta gerência*, com 36.8%.

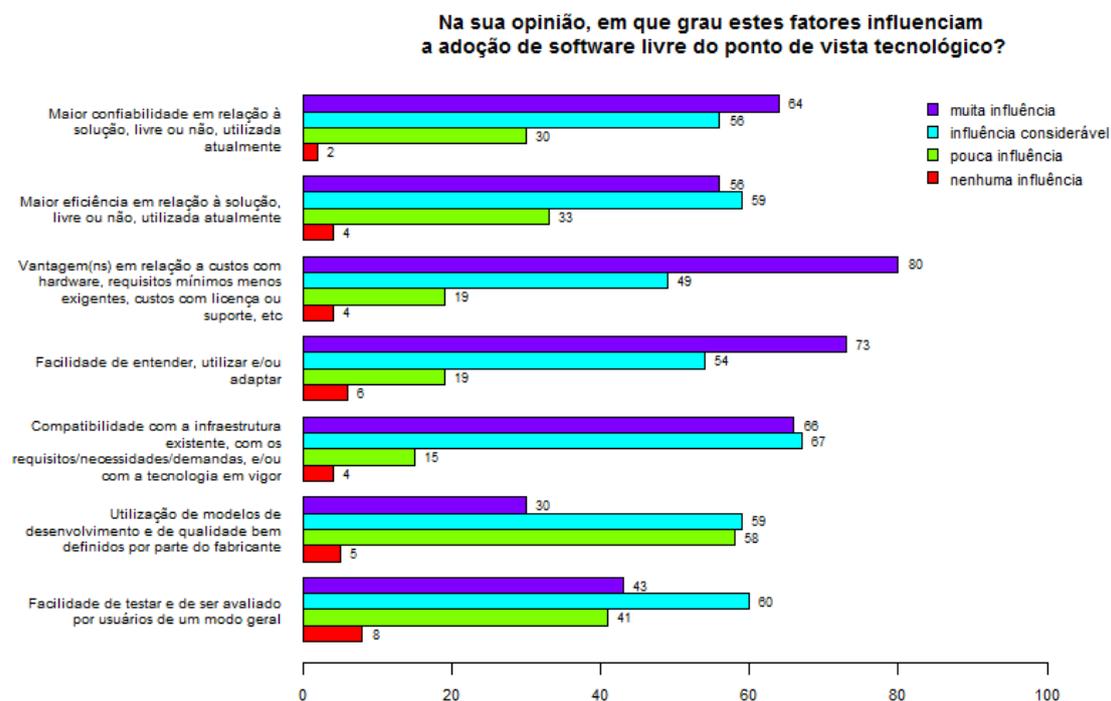


Figura 2. Fatores tecnológicos

Esperávamos que *experiência dos membros da equipe com uso/customização de software livre* (28.94%) fosse um fator importante para os participantes, o que não foi confirmado pelos resultados. Isso talvez possa ser explicado pela disponibilidade dos participantes em adquirir treinamento ou suporte/consultoria especializada para suprir esse tipo de deficiência entre os membros da equipe. *Disponibilidade de recursos para pesquisa/estudo de soluções que façam uso de software livre* (25%) aparece em último lugar, também possivelmente justificado pela possibilidade da contratação de consultoria externa.

Sobre os fatores mercadológicos, a Figura 4 aponta, como mais relevantes, os seguintes: **(a) existência de um caso de sucesso no mercado**, com 44.7% dos participantes; **(b) condições de mercado do software**, com 43.4% e **(c) disponibilidade de especialistas e/ou serviços de suporte externos à organização**, com 36.8%.

Ao contrário do que esperávamos, o *incentivo do governo* (19.7%) não foi considerado um fator importante para a adoção de software livre pelos participantes da pesquisa. *Marketing negativo por parte dos vendedores de software proprietário* (16.4%) também não representou um fator decisivo em relação ao mercado de software. *Condições de mercado de uma organização (fôlego financeiro, tamanho das equipes, distribuição geográfica, dentre outros)*, com 24.34%, também não se destacou como fator de adoção relevante. Isto sugere que, para os participantes da pesquisa, mesmo equipes pequenas e iniciantes podem ser capazes de tomar decisões maduras e acertadas sobre o uso (ou não) de software livre.

Finalmente, a Figura 5 aponta como principais fatores do contexto individual os seguintes: **(a) adoção prévia com sucesso de um software de código aberto**, com 43.4%

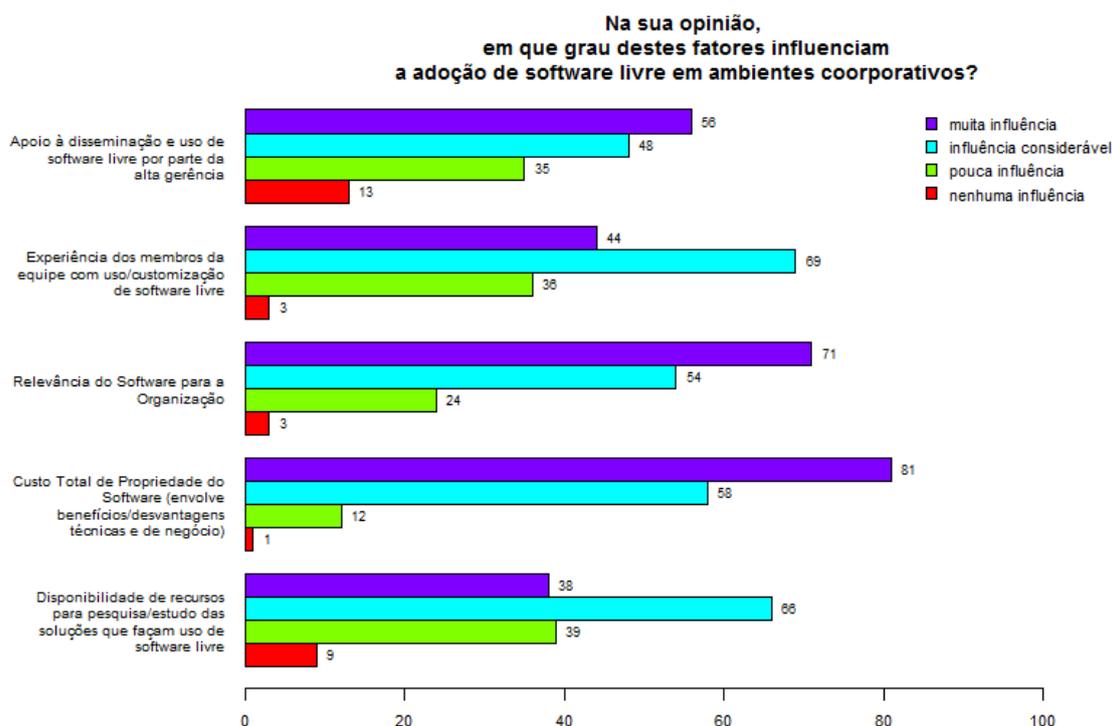


Figura 3. Fatores organizacionais

dos participantes e **(b)** existência de um caso de sucesso entre as pessoas do círculo de contatos com 41.4%.

Esperávamos que *perfil para pesquisa de soluções em software livre* (30.2%) fosse um fator mais relevante para os participantes da pesquisa. Isso também pode ser explicado pela possibilidade de treinamento e consultoria. *Aspectos políticos e/ou ideológicos* aparecem apenas com 26.9% das respostas para *muita influência*, sugerindo que os participantes consideram outras questões além do “*uso pelo uso*” de software livre.

3.2. Discussão dos Resultados da Survey

Os resultados da nossa pesquisa confirmam os resultados descritos por Morgan e Finnegan (2007), embora com algumas variações no grau de influência de alguns fatores de adoção. No contexto *tecnológico*, ambas as pesquisas avaliam *vantagens relativas* e *compatibilidade* como fatores importantes para a adoção de software livre. Entretanto, a *facilidade de testar e ser avaliado por usuários em geral* (um correspondente para o termo “*trialability*”³) aparece com menor importância nos resultados de nossa pesquisa.

No contexto *organizacional*, ambas as pesquisas concordam que *apoio da alta gerência ao uso de software livre* e *custo total de propriedade* são fatores importantes para a adoção de software de código aberto. Entretanto, nossa pesquisa considera que *relevância para a organização* corresponde a um fator de maior peso em relação à *experiência dos membros da equipe com uso/customização de software livre*⁴, ao contrário

³“*Trialability*” pode ser entendido, numa tradução livre, como “testabilidade”, ou capacidade de ser testado

⁴“*experiência dos membros da equipe com uso/customização de software livre*”, neste caso, cor-

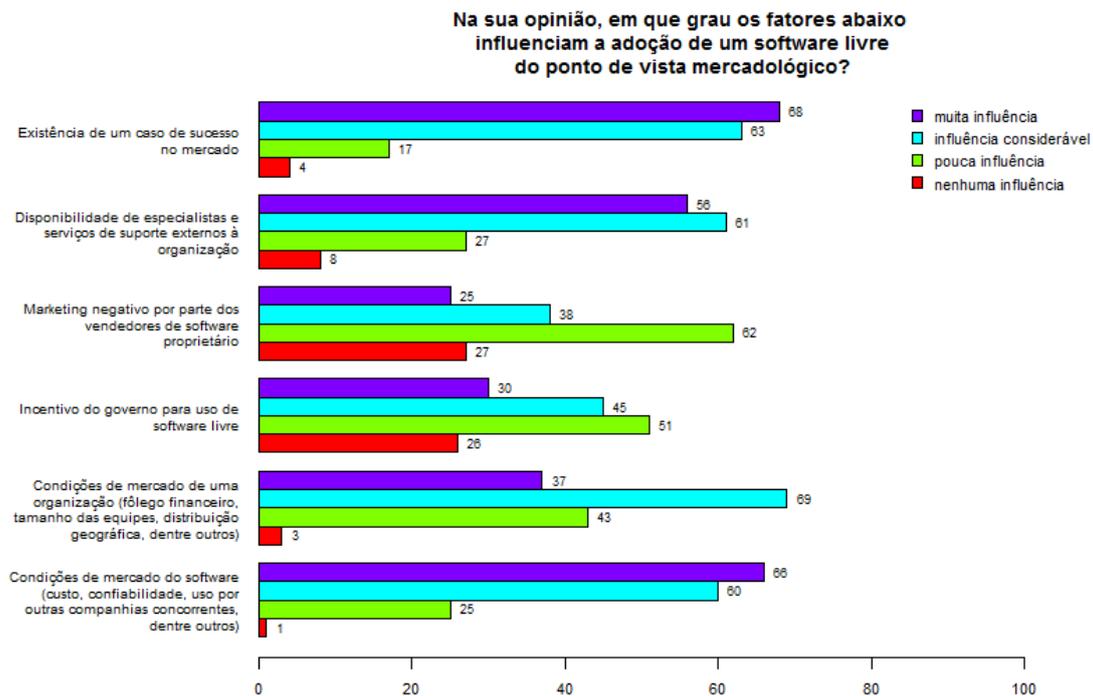


Figura 4. Fatores mercadológicos

do que sugerem Morgan e Finnegan (2007).

Para os contextos *ambiental* e *individual*, consideramos os resultados coincidentes para ambas as pesquisas, sem alterações na ordem de importância dada aos fatores de adoção.

3.3. Definição dos Critérios de Avaliação de Software Livre

A partir desta análise dos dados do levantamento de campo, bem como pelos *insights* oferecidos em entrevistas informais por desenvolvedores de software e outros pesquisadores, definimos o conjunto de dez itens a seguir como os critérios de avaliação de software livre de nosso sistema de recomendação: **(i) Facilidade de Entender, Testar e/ou Customizar;** **(ii) Desempenho/Eficiência;** **(iii) Atende às Necessidades;** **(iv) Facilidade de Instalação/Configuração;** **(v) Qualidade da Documentação;** **(vi) Custo Total de Propriedade do Software;** **(vii) Relevância/Utilidade do Software;** **(viii) Existência de Suporte Externo;** **(ix) Curva de Aprendizado para Utilização da Solução** e **(x) Popularidade**

Qualidade da documentação, facilidade de instalação/configuração e curva de aprendizado para a utilização da solução foram critérios oriundos de entrevistas informais com estudantes e profissionais da área de TI para fins de validação de conjuntos anteriormente pré-selecionados. *Atende às necessidades* é uma reescrita do fator *relevância para a organização*, no sentido de que um software pode ser relevante também para desenvolvedores comuns. Apesar de ter sido apontado como pouco relevante pela pesquisa, acreditamos que o *desempenho/eficiência* de um software (uma reescrita do fator *maior*

responde a uma adaptação para o termo *“boundary spanners”*, que segundo [Depietro et al. 1990, Morgan and Finnegan 2007] diz respeito à existência de pessoas que conectam a organização com o conhecimento exterior e possíveis inovações.

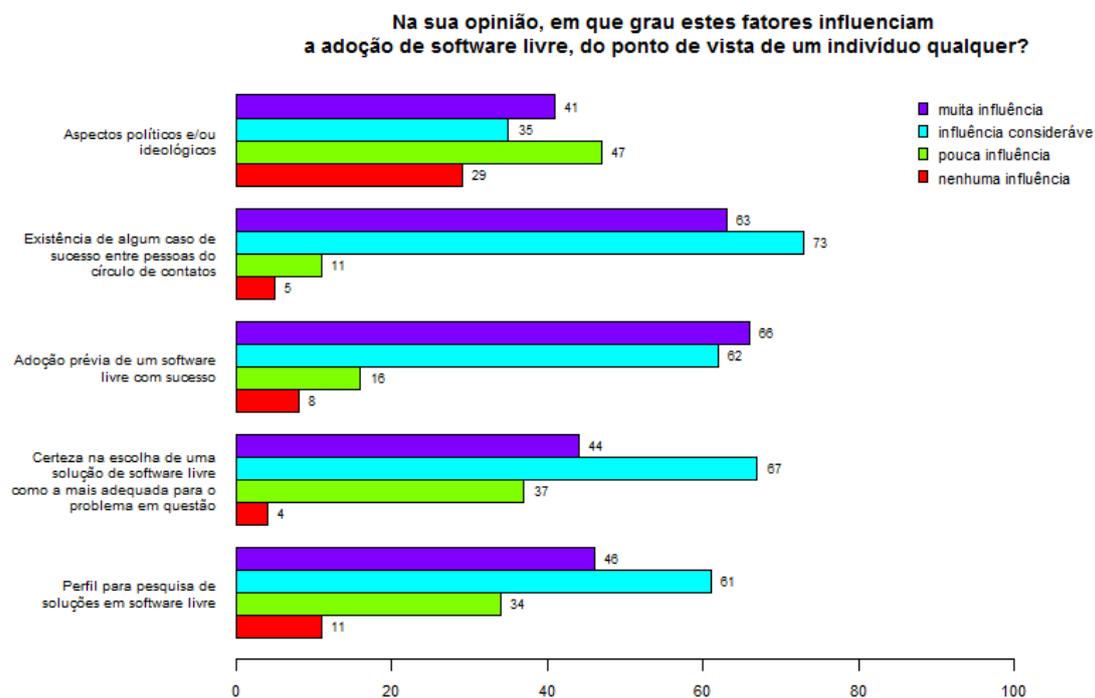


Figura 5. Fatores individuais

eficiência em relação à solução, livre ou não, utilizada atualmente) é crucial para a adoção do software.

Existência de suporte externo é uma reescrita para o fator *disponibilidade de especialistas e serviços de suporte externos à organização*. O critério *popularidade* corresponde à junção dos fatores *existência de algum caso de sucesso entre pessoas do círculo de contatos* e *existência de algum caso de sucesso no mercado*.

4. Recomendando Software Livre

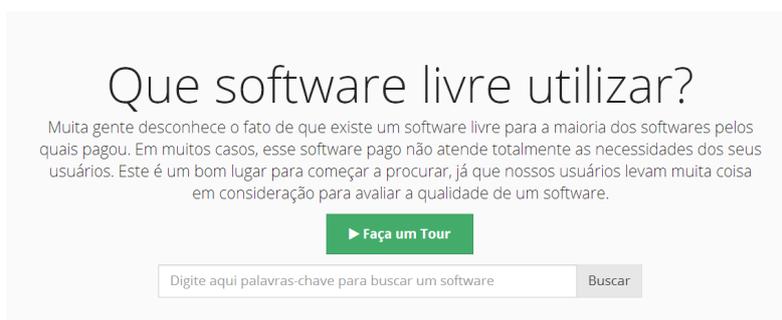


Figura 6. Tela inicial da ferramenta de busca

Após a análise dos resultados apontados pelo levantamento de campo (Seção 3.1), pudemos entender que existem diferentes fatores a serem considerados para a adoção de um software livre. Tais fatores podem se traduzir em critérios de avaliação durante a

procura pela solução de código aberto mais adequada às necessidades dos usuários. Entretanto, usuários diferentes levam em consideração conjuntos de critérios também diferentes durante suas buscas. Partindo deste princípio, consideramos interessante a criação de uma ferramenta que reunisse um conjunto relativamente abrangente dos critérios de avaliação discutidos na Seção 3.1, resumidos no conjunto descrito na Seção 3.3.

Em seguida, construímos um protótipo de recomendador de sistemas e componentes de software livre. Os dados sobre projetos de software livre foram obtidos através de requisições REST ao site OhLoh ⁵ [Black Duck Software 2013]. Utilizamos essencialmente a técnica de filtragem colaborativa para sugerir software livre aos usuários. Em sistemas de recomendação por filtragem colaborativa [Ricci et al. 2011], recomenda-se, ao usuário em questão, itens associados a outros usuários com perfis semelhantes. A similaridade entre os perfis é calculada com base no histórico de avaliação dos itens pelos usuários.

Projetos recomendados para você



Figura 7. Exemplo de recomendação colaborativa na tela inicial do sistema

A baixa quantidade de avaliações dos projetos de software livre na nossa base de dados nos fez cair no problema da "partida fria", inerente aos recomendadores colaborativos [Ricci et al. 2011]. Assim, o objetivo inicial de fornecer recomendações com base em múltiplos critérios de avaliação foi postergado para trabalhos futuros (mais detalhes na seção 4.2). Para fornecer algum nível de recomendação, inicialmente armazenamos informações de navegação do usuário, ainda que ele não tenha se autenticado no sistema. Neste caso, o usuário é identificado pelo endereço IP da máquina na qual ele acessa o sistema. Na página inicial, recomendamos um máximo de seis projetos de software livre, de acordo com os dados capturados, como mostrado na Figura 7.

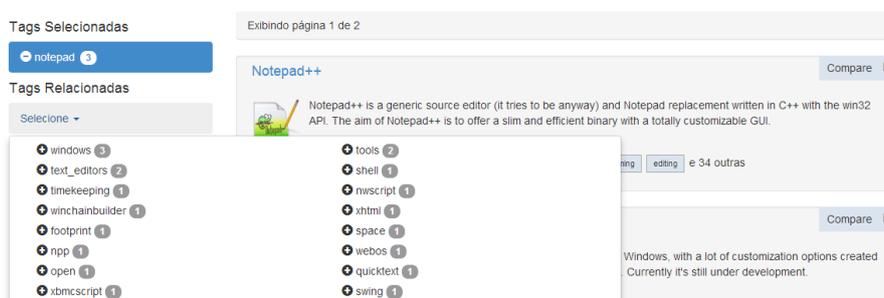


Figura 8. Navegação facetada nos resultados da busca pelo termo "notepad"

Uma vez que o usuário tenha pesquisado por algum tipo de software, seja pelo nome ou por alguma palavra chave relacionada (veja a Figura 6), é possível realizar

⁵<http://www.ohloh.net>

uma navegação facetada entre as *tags* associadas ao(s) projetos de software livre contidos no resultado da busca [Tunkelang 2009]. O novo resultado de busca representa uma conjunção destas *tags* (facetadas). A navegação facetada torna a busca por informações mais eficaz para os usuários do que as abordagens comuns [Tunkelang 2009]. A interface do sistema para o filtro de busca por *tags* pode ser visto na Figura 8.

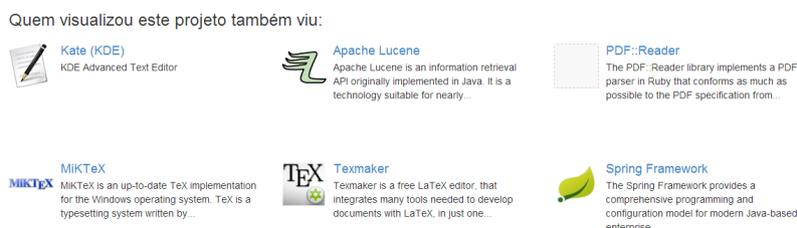


Figura 9. Recomendação colaborativa na tela de detalhe para o software Notepad++

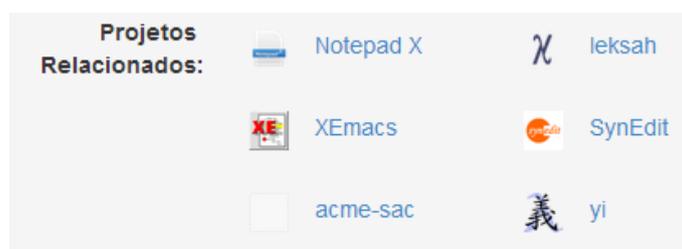


Figura 10. Lista de projetos de software livre similares ao software Notepad++

Tendo o usuário se interessado pela descrição de um software qualquer, ele poderá clicar no seu respectivo *link* na página de resultados da busca. Uma segunda página com detalhes sobre o software em questão é então exibida, onde pode-se ver uma recomendação por filtragem colaborativa, conforme mostrado na Figura 9. Projetos de software similares também são exibidos, conforme mostrado na Figura e 10. Nesta mesma página, também é possível observar a avaliação média da comunidade de usuários do sistema sobre o software em questão (como mostra a Figura 11), tanto de modo geral quanto pelos critérios previamente definidos. Ao clicar num ícone de ajuda posicionado ao lado de cada critério de avaliação, o usuário pode ler uma breve descrição sobre o objetivo deste critério, auxiliando na sua interpretação. Dados sobre a atividade do projeto (em termos de *commits* dos desenvolvedores), porcentagem de linguagens de programação utilizadas no código fonte, *links* para *download* dos executáveis, do código fonte e da documentação do software livre também podem ser visualizados.

Também há a possibilidade de se ler e escrever avaliações textuais sobre um dado software livre, levando em consideração os mesmos critérios de avaliação, como exemplificado na Figura 12. As notas dadas pelo usuário a cada critério de avaliação também podem ser visualizadas, como mostrado na Figura 13. A utilidade de tais avaliações é definida pelos próprios usuários que as leem. Na tela de "Avaliações e Opiniões", as avaliações tidas como mais úteis são exibidas primeiro, na coluna definida como "Opiniões mais úteis".

Comunidade

1 pessoa avaliou este projeto
★★★★★ 5.00 / 5



Figura 11. Avaliação média dos usuários do sistema para o software Notepad++

4.1. Técnicas de Recomendação

Em nosso protótipo de ferramenta, utilizamos a técnica de filtragem colaborativa *baseada em itens*, a partir dos dados capturados da navegação dos usuários pelos diferentes projetos cadastrados (vide Figura 9). De acordo com [Ricci et al. 2011, p. 120], recomendações feitas utilizando esse tipo de técnica são mais facilmente justificáveis ao usuário, apresentando-lhe uma lista dos itens "vizinhos" a um item recomendado em conjunto com os seus respectivos pesos de similaridade. Para contornar o problema da serendipidade⁶ apontado pelos autores, utilizamos a taxa de verossimilhança logarítmica (*log-likelihood ratio score*) como peso de similaridade entre os itens [Dunning 1993].

Para sugerir projetos de software livre similares a um projeto particular (vide Figura 10), utilizamos a técnica de filtragem baseada em conteúdo. Dentre as diversas informações catalogadas para um projeto, utilizamos apenas as palavras chave (*tags*) informadas para um projeto. Neste caso, consideramos que o uso da descrição dos projetos tende a introduzir informações pouco relevantes no espaço de busca.

4.2. Recomendação Multi-critério

A partir de uma melhor compreensão dos fatores de adoção de software livre considerados como mais importantes do ponto de vista da comunidade de TI (Seção 3.1), pudemos transformá-los em critérios para a avaliação de software livre (Seção 3.3). Com tais critérios, podemos orientar o usuário a avaliar o software livre de modo mais coerente do que fornecer uma mera nota (*ranking*) geral, apontando aspectos ele deveria levar em consideração para a avaliação.

⁶*Serendipidade* é um anglicismo que se refere às descobertas afortunadas feitas, aparentemente, por acaso

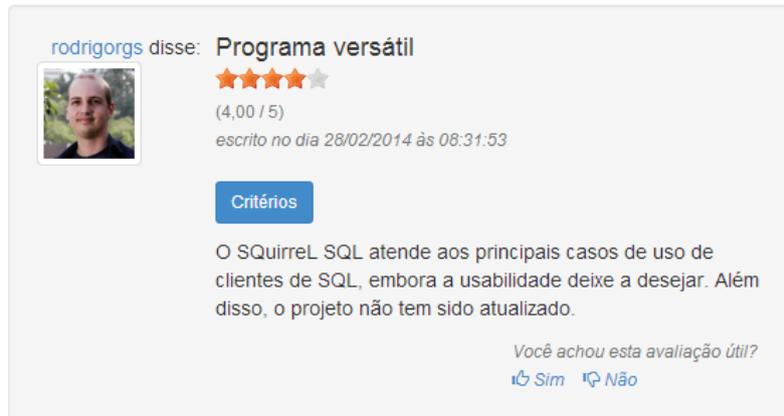


Figura 12. Avaliação textual de um usuário sobre o Squirrel SQL Client

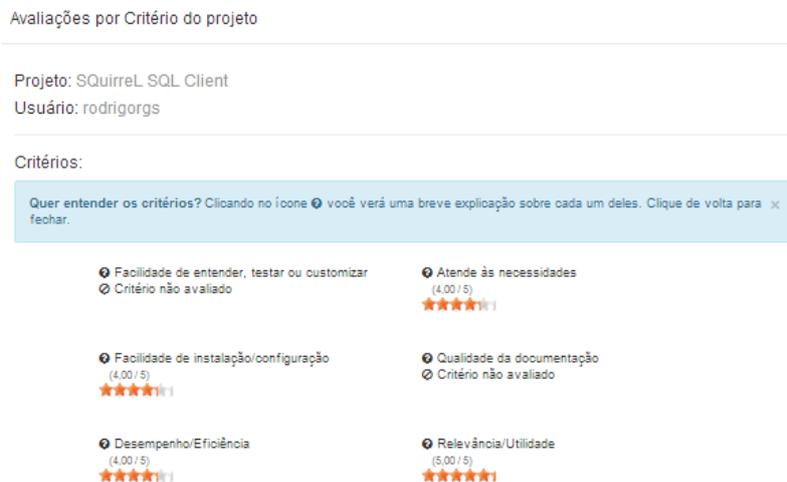


Figura 13. Notas dadas por um usuário em cada critério de avaliação para o Squirrel SQL Client

Um dos objetivos deste trabalho é utilizar os critérios de avaliação de software livre para construir um recomendador multi-critério, sugerindo itens de interesse do usuário na página inicial. Este é um trabalho em andamento, onde pretendemos utilizar duas estratégias de avaliação, como mostrado na Figura 14 [Adomavicius et al. 2011]:

- Obter uma função de agregação $f(c_1, c_2, \dots, c_n)$, onde c_i pertence ao conjunto de critérios de avaliação C , a partir das avaliações multi-critério dos usuários, utilizando técnicas estatísticas ou de aprendizado de máquina. A partir disso, prever avaliações gerais desconhecidas, utilizando f e avaliações previstas para cada critério individualmente;
- Prever avaliações desconhecidas separadamente para cada critério c_i do conjunto de critérios C , usando qualquer técnica de recomendação. Posteriormente, utilizar a função de agregação $f(c_1, c_2, \dots, c_n)$ para prever avaliações gerais desconhecidas.

Em nosso sistema, o usuário pode reavaliar um sistema ou componente de software de código aberto. Portanto, para efeitos de aprendizado de máquina, devemos coletar

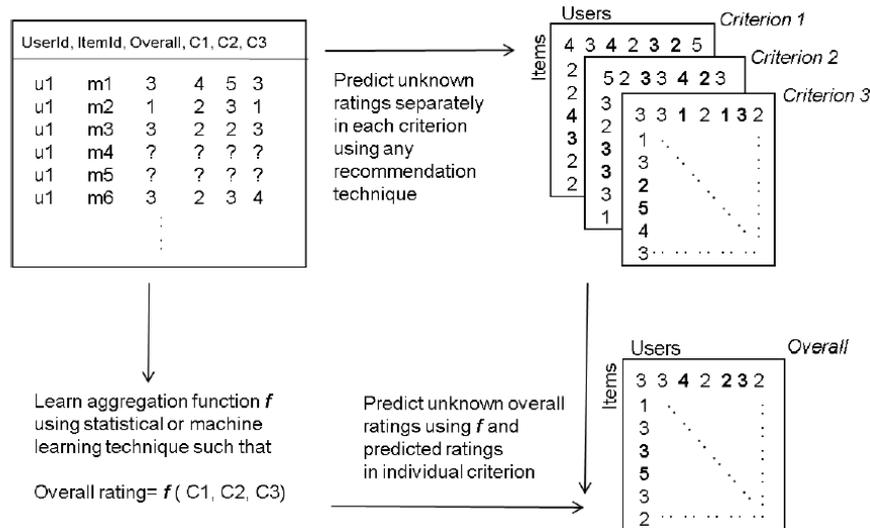


Figura 14. Abordagem utilizando funções de agregação. Exemplo para um sistema de recomendação com três critérios [Adomavicius et al. 2011].

apenas a avaliação mais recente do usuário para cada sistema ou componente.

4.2.1. Desafios em Aprendizado de Máquina

De acordo com [Alag 2009, p. 294], talvez a maneira mais simples de se construir um modelo preditivo seja utilizar técnicas padrão da estatística, tais como regressão linear ou quadrática. Em nosso caso, necessitamos construir um modelo preditivo para a avaliação geral de um projeto de software livre, partindo de avaliações reais ou estimadas para cada critério (vide Figura 14).

As avaliações por critério assumem valores dentro do intervalo inteiro $[0, 5]$, onde 0 indica um critério que não foi levado em consideração na avaliação. As avaliações gerais assumem valores dentro do intervalo inteiro $(0, 5]$, já que a avaliação geral é considerada obrigatória. Podemos considerar os valores possíveis para uma avaliação geral como sendo *categoricos*, isto é, cada valor pode ser considerado como uma categoria. Algoritmos de regressão podem ser utilizados para problemas de classificação [Alag 2009, p. 295]. Tendo isso em vista, pretendemos utilizar regressão logística *online* para obter o modelo preditivo. Também pretendemos utilizar características de programação genética para otimizar o modelo preditivo obtido.

5. Conclusões

Nosso levantamento de campo revelou que a comunidade brasileira de TI leva em consideração muito mais que aspectos políticos, ideológicos, e vantagens em relação a custo com licenças quando da adoção de software livre. Casos de sucesso entre pessoas do círculo de contatos e no mercado, além da facilidade de uso/customização e a compatibilidade com tecnologias e soluções já utilizadas podem influenciar bastante o processo de adoção de sistemas e componentes de código aberto, tanto por usuários/desenvolvedores comuns quanto pelas organizações.

A partir dos resultados do levantamento de campo, definimos um conjunto de critérios de avaliação de software livre, levando em consideração os fatores apontados pela pesquisa como os mais influentes no processo de adoção. Este conjunto leva em consideração fatores de adoção relevantes tanto para usuários/desenvolvedores comuns quanto organizações de TI, públicas, privadas e sem fins lucrativos. A partir deste conjunto, podemos orientar os usuários, desenvolvedores e organizações a avaliarem software livre de modo coerente e objetivo.

Construímos um protótipo de recomendador multicritério de sistemas e componentes de código aberto. Este recomendador baseia-se numa interface web, e também conta com opiniões textuais dos seus usuários, reforçando as avaliações feitas para os critérios pré-selecionados. Estas opiniões podem ser avaliadas como úteis ou não pelos próprios usuários. Utilizamos tanto técnicas de filtragem baseadas em conteúdo como filtragem colaborativa para recomendar software livre ao usuário. O uso de técnicas de recomendação aplicadas ao domínio de sistemas e componentes/frameworks de código aberto é de grande relevância, na medida em que o público em geral ainda possui certa resistência ou desconhece a existência deste tipo de software.

Como trabalhos futuros, podemos destacar a avaliação *offline* e *online* dos itens recomendados pela nossa ferramenta. Para a avaliação *offline*, utilizaremos o próprio recomendador para estimar notas de sistemas e componentes já avaliados pelos usuários, medindo o quão próximo o valor estimado se aproxima do valor real. Para a avaliação *online*, entrevistaremos usuários do sistema, a fim de saber se a recomendação feita pela ferramenta foi, de fato, útil para eles. Ainda há a possibilidade de se criar perfis de avaliação diferentes para usuários/desenvolvedores comuns e organizações de TI, com critérios de avaliação diferentes para cada um, tornando tanto a avaliação quanto a recomendação mais direcionada.

Referências

- [Adomavicius et al. 2011] Adomavicius, G., Manouselis, N., and Kwon, Y. (2011). Multi-criteria recommender systems. In *Recommender systems handbook*, pages 769–803. Springer.
- [Alag 2009] Alag, S. (2009). *Collective intelligence in action*. Manning New York.
- [Araujo 2011] Araujo, T. C. (2011). *AppRecommender: um recomendador de aplicativos GNU/Linux*. PhD thesis, Universidade de São Paulo.
- [Black Duck Software 2013] Black Duck Software (2013). Ohloh. Online. Acessado em 14 de Fevereiro de 2013.
- [Depietro et al. 1990] Depietro, R., Wiarda, E., and Fleischer, M. (1990). The context for change: Organization, technology and environment. *The processes of technological innovation*, pages 151–175.
- [Dunning 1993] Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *COMPUTATIONAL LINGUISTICS*, 19(1):61–74.
- [Hauge et al. 2008] Hauge, Ø., Sørensen, C.-F., and Conradi, R. (2008). Adoption of open source in the software industry. In *Open Source Development, Communities and Quality*, pages 211–221. Springer.

- [Mccarey et al. 2005] Mccarey, F., Cinnéide, M. Ó., and Kushmerick, N. (2005). Rascal: A recommender agent for agile reuse. *Artificial Intelligence Review*, 24(3-4):253–276.
- [Morgan and Finnegan 2007] Morgan, L. and Finnegan, P. (2007). How perceptions of open source software influence adoption: An exploratory study. In *Proceedings of the 15th European Conference on Information Systems (ECIS 2007)*, pages 7–9.
- [Reategui and Cazella 2005] Reategui, E. B. and Cazella, S. C. (2005). Sistemas de recomendação. In *XXV Congresso da Sociedade Brasileira de Computação*, pages 306–348.
- [Resnick et al. 1997] Resnick, P., Varian, H. R., and Editors, G. (1997). Recommender Systems. *Communications of the ACM*, 40(3):56–58.
- [Ricci et al. 2011] Ricci, F., Rokach, L., and Shapira, B. (2011). *Recommender Systems Handbook*. Springer US, Boston, MA.
- [Tunkelang 2009] Tunkelang, D. (2009). Faceted search (synthesis lectures on information concepts, retrieval, and services). *Morgan and Claypool Publishers*.
- [Verheesen 2010] Verheesen, M. (2010). How to buy something that is free? Master's thesis, Technische Universiteit Eindhoven, Eindhoven, The Netherlands.